

Understanding Quantum Technologies

Ninth edition – 2026
Part 3/5 – Quantum Software

Olivier Ezratty



le lab quantique

le lab quantique

Le Lab Quantique is a supporter and promoter of this book, but not its publisher or editor.



Understanding Quantum Technologies

Ninth edition - 2026 - Version 9.0

Part 3/5 – Quantum Software

Olivier Ezratty

Generated on September 28, 2026

About the author

Olivier Ezratty



0000-0003-3944-2896

olivier (at) oezratty.net
www.oezratty.net, @olivez
+33 6 67 37 92 41

Olivier Ezratty is an author, academic, teacher and advisor on quantum technologies. He started this endeavour in 2018. He is the author of the reference book “Understanding Quantum Technologies” (September 2021, 2022, 2023, 2024, 2025 and 2026) following three previous editions in French in 2018, 2019 and 2020. The 2025 edition is also available in paperback version on Amazon. This 2026 edition’s Amazon print version will be available before the end of 2026.

Since 2021, he has been teaching quantum technologies at EPITA, an IT engineering school in France, at Ecole Normale Supérieure Paris-Saclay since 2024 and CentraleSupélec since 2005. He’s affiliated with the EPITA Research lab for his various research writings.

He is a cofounder of the Quantum Energy Initiative in 2022 with Alexia Auffèves (CNRS MajuLab Singapore), Robert Whitney (CNRS LPMCM) and Janine Splettstoesser (Chalmers University, Sweden), and member of its scientific board.

He acts as an independent expert for Bpifrance and at the Agence Nationale de Recherche (ANR) to evaluate quantum R&D projects, as a member of the European Quantum Computing Advisory Board (Q-CAB) since April 2026, and of the Inria expert group for the selection of risky research projects (Programme PIQ) since June 2024.

Since 2018, he has spoken in a large number of quantum technology events such as the Q2B Paris and Santa Clara organized by QC Ware, France Quantum, Lindau Nobel Laureate meeting, DPG, Q-Expo and other events, on top of presentations at Société Générale, BNPParibas, Crédit Agricole, Michelin, Adéo, L’Oréal, FIECC, IHEDN, Business France, CentraleSupélec, Avolta Partners, IHEDN, etc.

He coproduces two series of podcasts on quantum technologies along with Fanny Bouton (in French): a monthly “Quantum ” on tech news (since September 2019) and Decode Quantum (2020-2026), with entrepreneurs and researchers, with a total of about 170 episodes as of September 2026.

He also lectures or lectured in various universities such as CentraleSupélec, ENSTA, Telecom Paristech, Les Gobelins, HEC, Neoma Rouen and SciencePo, on artificial intelligence, entrepreneurship and product management (until 2020) and on quantum technologies (since 2018), in French and English as needed. He is also the author of many open-source ebooks in French on entrepreneurship (2006-2019), the CES of Las Vegas yearly report (2006-2020) and on artificial intelligence (2016-2021).

Beforehand, Olivier Ezratty was specialized in many other topics since 2005, like digital television, internet of things and artificial intelligence. As such, he carried out various strategic advisory missions, conferences, or training assignments in different verticals and domains such as media and telecoms, finance and insurance, industry and services and the public sector.

Olivier Ezratty started in 1985 at Sogitec, a subsidiary of the Dassault group, where he was successively Software Engineer, then Head of the Research Department in the Communication Division. He initiated developments under Windows 1.0 in the field of editorial computing as well as on SGML, the ancestor of HTML and XML. Joining Microsoft France in 1990, he gained an extensive experience in many areas of the marketing mix: products, channels, markets and communication. He launched in France the first version of Visual Basic in 1991 and Windows NT in 1993. In 1998, he became Marketing and Communication Director of Microsoft France and in 2001, of the Developer Division, which he created in France to launch the .NET platform and promote it to developers, higher education and research, as well as to startups. Olivier Ezratty is a software engineer from Centrale Paris (1985), which became CentraleSupélec in 2015.

This document is provided free of charge and is licensed under a “Creative Commons” license.

in the variant “Attribution-Noncommercial-No Derivative Works 2.0”. ISSN 2680-0527



Cover illustration credits: personal creation associating a Bloch sphere describing a qubit and the symbol of peace (my creation, first published in 2018) above a long list of terms from the book glossary. This book contains about 1,325 illustrations. I have managed to give credit to their creators as much as possible. Most sources are credited in the captions. Only scientists’ portraits are not credited since it’s quite hard to track their original source. I have added my own credit in most of the illustrations I have created or modified. In some cases, I have redrawn some third-party illustrations to create clean vector versions or used existing third-party illustrations and added my own text comments. The originals are still credited in that case.

Table of contents

Table of contents	iii	1.4.8 Quantum Singular Value Decomposition	112
Foreword	v	1.4.9 Barren plateaus	112
Contents	vii	1.4.10 Analog quantum computing and simulations	113
1 Quantum algorithms	1	1.4.11 Digitized Counter-Diabatic Quantum Optimization	117
1.1 Quantum algorithms classes	4	1.4.12 NISQ distributed computing	118
1.1.1 Classes and use cases	6	1.4.13 Bibliography and notes	118
1.1.2 Algorithms and quantum computing paradigms	7	1.5 Quantum speedups	130
1.1.3 Algorithms process and compilation	7	1.5.1 Complexity class vs speedup gains	130
1.1.4 Algorithms toolbox	9	1.5.2 Non-Clifford gates	130
1.1.5 How is a quantum algorithm created?	11	1.5.3 Maximally entangled states	130
1.1.6 Algorithms figures of merit	11	1.5.4 Superpolynomial and exponential speedups	132
1.1.7 Bibliography and notes	14	1.5.5 Theoretical vs practical speedups	133
1.2 Basic algorithms toolbox	15	1.5.6 Bibliography and notes	136
1.2.1 Data encoding	15	1.6 Complexity classes	138
1.2.2 Quantum State Transfer	18	1.6.1 Generic complexity classes	139
1.2.3 Black boxes and oracles	19	1.6.2 Quantum complexity classes	144
1.2.4 Output encoding	20	1.6.3 Bibliography and notes	147
1.2.5 Quantum phase kickback	20	1.7 Quantum inspired algorithms	149
1.2.6 Quantum arithmetic	21	1.7.1 Applications	149
1.2.7 Uncompute trick	21	1.7.2 Tensor networks	151
1.2.8 Quantum Fourier Transform	23	1.7.3 Other quantum inspired techniques	154
1.2.9 Amplitude Amplification	26	1.7.4 Bibliography and notes	155
1.2.10 Hamiltonian simulations	27	1.8 Quantum algorithms key takeaways	160
1.2.11 Quantum Phase Estimation (QPE)	27	2 Quantum software development tools	161
1.2.12 Quantum Imaginary Time Evolution (QITE)	33	2.1 Development tool classes	161
1.2.13 Quantum Amplitude Estimation (QAE)	34	2.1.1 Graphical programming tools	161
1.2.14 Quantum Signal Processing (QSP)	35	2.1.2 Scripting languages	164
1.2.15 Linear and differential equations (HHL, QLSA, PDE)	35	2.1.3 Frameworks	165
1.2.16 Bibliography and notes	39	2.1.4 Machine languages	165
1.3 Higher level algorithms	47	2.1.5 Compilers, transpilers and optimizers	165
1.3.1 Oracle-based algorithms	47	2.1.6 Application-specific frameworks	169
1.3.2 Shor integer factoring	52	2.1.7 Emulators and simulators	169
1.3.3 Shor dlog	55	2.1.8 Resource estimators	180
1.3.4 Hidden Subgroup Problems (HSP)	56	2.1.9 Bibliography and notes	185
1.3.5 Computational Fluid Dynamics (CFD)	56	2.2 Academic quantum development tools	193
1.3.6 Quantum Artificial Intelligence (QAI)	59	2.2.1 Bibliography and notes	197
1.3.7 Quantum physics simulation	71	2.3 Vendor quantum-development tools	199
1.3.8 Optimization algorithms	73	2.3.1 D-Wave	199
1.3.9 Bibliography and notes	80	2.3.2 IBM	199
1.4 NISQ algorithms	96	2.3.3 Rigetti	202
1.4.1 Algorithms classes	96	2.3.4 Google	202
1.4.2 Qubit quality	97	2.3.5 Microsoft	205
1.4.3 Computing time	100	2.3.6 Amazon	206
1.4.4 Variational Quantum Eigensolvers	102	2.3.7 IonQ	206
1.4.5 Quantum Approximate Optimization Algorithms and other optimizations	105	2.3.8 Intel	206
1.4.6 Quantum Machine Learning	110	2.3.9 Huawei	207
1.4.7 Variational Quantum Linear Solver	112	2.3.10 Bull	207
		2.3.11 Bibliography and notes	208
		2.4 Quantum software engineering	210
		2.4.1 Certification and verification	210
		2.4.2 Testing and debugging	211
		2.4.3 Verification and validation	211
		2.4.4 Code porting	213

2.4.5	Code refactoring	213	3.4	Life sciences	297
2.4.6	Securing quantum computing	213	3.4.1	Protein dynamics	302
2.4.7	Machine learning in quantum computing	215	3.4.2	Diagnosis	302
2.4.8	Bibliography and notes	217	3.4.3	Drug discovery and retargeting	304
2.5	Benchmarking	222	3.4.4	Treatments	306
2.5.1	Classical computing benchmarking	222	3.4.5	Healthcare	306
2.5.2	Quantum computing benchmarking	222	3.4.6	Bibliography and notes	307
2.5.3	IBM Quantum Volume	223	3.5	Energy	312
2.5.4	IBM CLOPS	227	3.5.1	Power grid	312
2.5.5	IBM EPLG	227	3.5.2	Renewable energies	313
2.5.6	QLOPS	228	3.5.3	Oil exploration	313
2.5.7	IonQ algorithmic qubits	228	3.5.4	Nuclear energy	313
2.5.8	Other systems level benchmarks and metrics	228	3.5.5	Bibliography and notes	314
2.5.9	Hardware specific benchmarks	230	3.6	Transportation and logistics	317
2.5.10	Eviden Q-score	230	3.6.1	Automotive	317
2.5.11	Other application level benchmarks	231	3.6.2	Rail	319
2.5.12	International standard organizations	232	3.6.3	Air	320
2.5.13	Benchmarking tools	233	3.6.4	Sea	321
2.5.14	Quantum supremacy and advantage	234	3.6.5	Supply chain	322
2.5.15	Bibliography and notes	242	3.6.6	Bibliography and notes	322
2.6	Cloud quantum computing	248	3.7	Financial services	326
2.6.1	Cloud rationale	248	3.7.1	Analysts roadmaps	326
2.6.2	Cloud offerings	249	3.7.2	Bank approaches	326
2.6.3	HPC-quantum hybrid computing	252	3.7.3	Portfolio optimization	328
2.6.4	Space as the new cloud?	255	3.7.4	Derivatives and options pricing	331
2.6.5	Bibliography and notes	255	3.7.5	Risk analysis	331
2.7	Quantum software development tools key takeaways	258	3.7.6	Currency	332
3	Quantum computing applications	259	3.7.7	Credit scoring	333
3.1	Case studies and use cases evaluation	259	3.7.8	Fraud detection	333
3.1.1	Problem sizing	261	3.7.9	Other use cases	333
3.1.2	Resource estimates	262	3.7.10	Cybersecurity	334
3.1.3	Quantum advantage nature	263	3.7.11	Quantum money	334
3.1.4	Classical comparisons	264	3.7.12	Insurance	335
3.1.5	Algorithm advancement	264	3.7.13	Bibliography and notes	336
3.1.6	Quantum computer type	264	3.8	Other user segments	341
3.1.7	Documentation	266	3.8.1	Industry	341
3.1.8	Example	266	3.8.2	Civil engineering and construction	342
3.1.9	Market	266	3.8.3	Retail	342
3.1.10	Bibliography and notes	268	3.8.4	Telecommunications	343
3.2	Market forecasts	270	3.8.5	Content and media	343
3.2.1	Quantitative assessments	270	3.8.6	Marketing	346
3.2.2	Qualitative assessments	272	3.8.7	Defense and aerospace	347
3.2.3	Market variables	272	3.8.8	Intelligence services	349
3.2.4	Bibliography and notes	274	3.8.9	Bibliography and notes	349
3.3	Chemistry and material design	276	3.9	Science	353
3.3.1	Quantum chemistry 101	276	3.9.1	Climate change mitigation	353
3.3.2	Chemical engineering	283	3.9.2	Astrophysics	354
3.3.3	Fertilizer production	285	3.9.3	High-energy physics	355
3.3.4	Cement production	289	3.9.4	Quantum physics	357
3.3.5	Material sciences and design	289	3.9.5	Bibliography and notes	357
3.3.6	Carbon capture	290	3.10	Software and tools vendors	360
3.3.7	Batteries	291	3.11	Service vendors	395
3.3.8	Bibliography and notes	291	3.11.1	Bibliography and notes	399
			3.12	Quantum computing business applica- tions key takeaways	407

Foreword

The birth of the second quantum revolution may be reasonably settled to 1982, with two celebrated articles: “Experimental Test of Bell’s Inequalities Using Time-Varying Analyzers” by Alain Aspect and colleagues, and the famous theoretical paper “Simulating Physics with Computers” by Richard Feynman. They were quickly followed in 1985 by David Deutsch’s “Quantum theory, the Church–Turing principle and the universal quantum computer”, providing the birth certificate of what is known now as quantum computing. Incidentally, I got a position at CNRS also in 1982, a good spot to witness these developments. However these subjects, including the surprising quantum key distribution scheme proposed by Charlie Bennett and Gilles Brassard in 1984, were still quite confidential at that time. It took ten more years for the topic to attract broad international attention, with the articles “Algorithms for quantum computation: Discrete logarithms and factoring” (1994) and “Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer”, both by Peter Shor.

This started an explosive development, which has not stopped since then. The big impact of Shor’s algorithm is that it may break cryptographic codes that were considered fully secure at that time, and are still broadly used on internet - so each layman gets concerned. This, and other reasons, triggered a huge international effort to build a quantum computer. Since it was clear that such a computer was still far away, this effort was mostly academic during the next 20 years. But another earthquake happened in 2014, when large US companies, Google and IBM as the most visible ones, decided to get into the race. They set up heavily funded research programs, focused on a specific technology, superconducting quantum bits. This move stimulated a high level of private investment, leading to the creation of a flurry of dedicated companies, initially startups, but then quite respectable enterprises in terms of funding. They were not limited to superconductors, but included all quantum objects likely to become a quantum bit, a qubit : ions, atoms, photons, and even more exotic ones. One must honestly say that, from this time, the language used in this research field changed significantly: the way to speak with investors is not the same as the way to publish articles in reputable scientific journals. The result was a mixture of authentic scientific and technological progress, coming together with hype and overselling. This may be seen either as an unavoidable evil, or as the appropriate way to communicate in an extremely competitive high-tech environment.

A good example is the way to name the “ultimate goal”, the point where a quantum computer can really achieve a useful task that no classical computer can do in any reasonable time. The first name that popped up was “quantum supremacy”, but it did not make it, both for technical and political reasons. The next one was

“quantum advantage”, more politically correct, but still difficult to warrant. The next one is “quantum utility”, clearly more modest, but more likely to be reached in a reasonable time span. In the current language, “utility” is used as a step towards “advantage”, with some possible oscillations between the two due to progress in classical computing.

Then a question arises : given the very fast progress of the devices, their technical complexity, and the strange mixture of science and publicity used by companies to communicate their results, how to find one’s way in the jungle, without getting lost or eaten ? One would need somebody with good scientific and technological knowledge, but also a huge talent for global accounts, given the many facets of these developments, involving physics, hardware, software, middleware, logistics, and finance. We are lucky enough that this person exists, his name is Olivier Ezratty, and you hold his book in your hands or on your computer screen. This is not a modest enterprise, and $\approx 1,822$ pages carefully written in L^AT_EX is a minimal container. Go for it, it’s free ! But how to approach such a monument ? Fortunately there is a reading guide in the first pages, able to orient you depending on your background and interests. It includes also an illustrated “Pandora’s box”, listing the main high level topics (physics, mathematics, human sciences, engineering, computer science, geopolitics), together with their inner content, opening doors towards many stimulating questions for science, technology, and society.

So where do we stand now, already 44 years after the baby was born ? The general idea is that the real stuff, the one needed for supremacy / advantage / utility for practical problems of industrial interest, requires what is called now “Fault Tolerant Quantum Computing” (FTQC), in the line of Deutsch’s ideas; this is quite attractive for computer scientists, but very hard to achieve. Physicists tend to prefer quantum simulators, in line with Feynman’s ideas, potentially useful for more fundamental physics problems. Myself, as a physicist, will go for the second trail, because one should not dismiss fundamental physics problems: it is by solving them that science made progress, and not necessarily by jumping right now into the market. Actually, it is not clear that all fundamental questions to be answered before being able to implement quantum computing have been solved, as summarized in the old joke : “Where is quantum computing? Somewhere between cold fusion and hot fusion”, a physicist’s warning that nature’s laws are not to be fooled around.

Anyway, as we say in French, “*on ne discute pas des goûts et des couleurs*” (there’s no accounting for tastes): whatever your personal interest is - physicist, computer scientist, or investor -, and wherever you may want to look at - computing, communications, sensors or metrology - the development of quantum technologies is a fascinating arena where much more is expected to come.

I cannot close this foreword without a few words about another ongoing revolution, artificial intelligence and large language models generative artificial intelligence (simplified as “LLM”). The most straightforward step forward, taking place currently in all domains of scientific and technical activities, is to use LLM as super-efficient assisting tools; their progress during the 2 or 3 last years has been astonishing, and it cannot be listed

here. Another more speculative idea is to think about a “quantum AI”, an AI running on a quantum computer. The mismatch between the current performances of quantum computers, and what is used now on supercomputers running AI, makes this objective extremely far-fetched. Still, blue-sky thinking about what might be achieved by such machines - and even the question of whether they will still be “machines”, is attracting attention by some quantum theoreticians. Where will these revolutions lead humanity ? Only time will tell.

Philippe Grangier

Director of Research emeritus at CNRS, Institut d’Optique Graduate School (IOGS), and incoming President of the French Optical Society (SFO).

September 2026

Contents

This is **Part 3/5 – Quantum Software** of the book “Understanding Quantum Technologies 2026”, which is also downloadable using [this link](#).

The downloadable PDFs are available in a single A4 and Letter version, containing the five parts in sequential order. You can also download compressed PDFs for the five parts in A4 and Letter formats. Their size fits the constraints of ebook readers like the Kindle from Amazon.

The e-reader and printed version of the book separates parts 1, 2, 3, 4 and 5.

- The **first part** contains a history of quantum physics, some quantum physics 101 and everything about quantum computing basics and engineering including quantum error correction, the energetics of quantum computing, a view on pessimism and optimism with regard to the advent of scalable and useful quantum computers, as well as some economic considerations.
- This **second part** contains the sections dedicated to quantum computing hardware with all types of qubits and their related industry vendors, enabling technologies, and unconventional computing solutions which are potential alternate routes between classical and quantum computing.
- The **third part** contains the sections dedicated to algorithms, software tools, and applications across multiple markets. It also contains an inventory of quantum computing software and service vendors.
- The **fourth part** covers quantum communications, quantum cryptography, post-quantum cryptography and quantum sensing.
- The **fifth part** describes the quantum ecosystem at the entrepreneurial level, the country level, on various societal topics, corporate adoption methodologies, quantum pseudosciences. It ends with a glossary, and book bibliography.

The book is split into five parts to make its printing easier, some online printing services including Amazon being limited to a maximum of 600 pages.

You can order the printed version of this book in five parts on all Amazon sites by searching for the book title, edition 2026, when it is released around October-November 2026.

1 Quantum algorithms

Once you know what is a qubit, a quantum gate, a quantum circuit and quantum measurement, you are ready to move up in the quantum computing food chain and learn about quantum algorithms. Quantum computers use quantum algorithms which describe series of operations that are executed in quantum circuits. This is the case of gate-based quantum computing. With analog quantum computers, the software process is different. It consists of converting a problem formulation into another formulation that the machine will understand. A formulation is a bit different than an algorithm. It looks more like a graph with relationships between elements. We will also see how this works. In both cases, quantum computer circuits or reformulated problems can lead to some theoretical and practical speedups compared with their best equivalents designed for classical computers. There are not so many quantum algorithms available their relative performance compared with classical algorithms is not always easy to prove or generalize. It is even sometimes contested. The assertion “*quantum computers are faster than classical computers*” is therefore debatable and must be discussed and analyzed on a case-by-case basis. On top of that, we will see that the benefit of quantum computing may not just be about speed but also about the quality of the results.

Richard Feynman described the idea of creating quantum simulators in 1982 [1]. His idea was to create devices using the effects of quantum mechanics to simulate other quantum systems, which would be almost impossible with traditional computers. This corresponds today to analog quantum simulators like the ones from Pasqal and QuEra. In this part of the book, we are dealing mainly with gate-based quantum computing, based on **Yuri Manin** and **Paul Benioff**’s ideas from 1979-1981, which were later refined by **David Deutsch** between 1985 and 1992.

The first quantum algorithms were published in the early 1990s long before any quantum hardware was available. The first two-qubit quantum computers appeared around 2000/2002. Researchers have been regularly creating new algorithms for the past three decades, regardless of the relatively slow progress with hardware (Figure 2). Launched in 2011, the Quantum Algorithm Zoo [2] identified about 65 algorithms in the scientific literature as of September 2026 when it was last updated, organized into four algorithm groups (algebraic and number theory, oracle-based problems, approximation and simulations, optimization - numerics and machine learning). The list is maintained by Stephen Jordan, then a Microsoft Quantum researcher, who has worked at Google AI since 2023. This is still a modest number compared to the thousands of non-quantum algorithms [3–6]. Even though most classical computing developers don’t know or use many algorithms in practice! Quantum algorithms creation is thus a parallel research path with hardware progress. This is not the first time in history.

Hardware and algorithms parallel development

The emblematic **Ada Lovelace** did formalize the first algorithms and lines of code to run on **Charles Babbage**’s machine, which only saw the light of day in 2002

in London, 153 years after its creation (video) (see the sample program in Figure 1). In 1842/1843, she had annotated a translation of her own of a paper by the Italian **Luigi Federico Menabrea** describing Babbage’s machine.

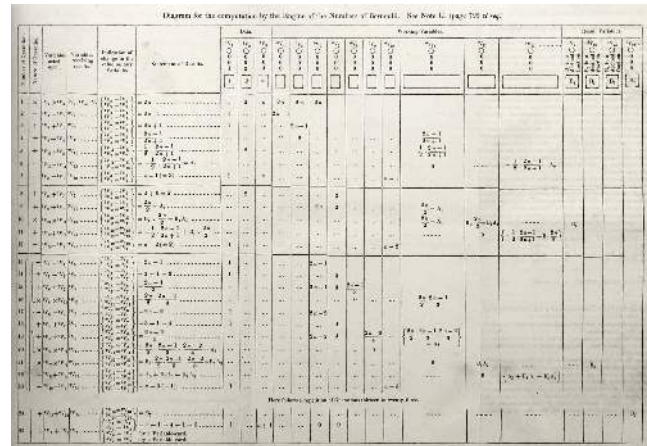


Figure 1: Ada Lovelace’s 1842 Note G, the first algorithm and program computing Bernoulli numbers designed for Charles Babbage’s mechanical analytical machine which didn’t exist then and was manufactured in the UK only in 2003!

It took 102 years for the first electronic computers to see the light of day at the end of World War II, the ENIAC. A beautiful game... of patience! It is reminiscent of **Leonardo da Vinci**’s helicopter designs dating from 1487-1490. A first human-powered helicopter created by the University of Toronto flew in 2013, AeroVelo (video) followed by another fairly close specimen from the University of Maryland flying in 2018 (video)! So, more than five centuries apart. And even considering the flight of the first motorized helicopter in 1907, the time lag is still over four centuries. This same University of Maryland is one of the most advanced in the world in quantum computers based on trapped ions!

After the war, history repeated itself in part for much of the work in the vast field of artificial intelligence, where researchers were also working on algorithms, especially neural network-based algorithms, before any computers could execute them properly on a useful scale such as for object recognition in images. The first computers running perceptrons, the ancestors of today’s artificial neural networks, were rudimentary.

The rise of deep learning since 2012 is partly linked to the power of machines and GPUs able to train such neural networks. Hardware has once again joined algorithms that were ahead of their time.

Even today, many of the quantum algorithms that are invented are not yet executable on a large problem scale on current quantum computers or on classical computing quantum emulators. There are not enough quality qubits available to be of any use and, more importantly, to be more powerful in any dimension than classical computers. Supercomputers can emulate about 50 qubits in state vector mode, but no operational quantum computer can reach this number with error-corrected qubits.

Quantum computing abstraction

In another analogy with the history of computer science, we are still programming quantum computers with rather low layers of machine language, a bit like machine language or macro-assembler used 30 to 50 years ago, or more recently, for those who program low-level embedded systems or peripheral drivers. Today’s quantum algorithms are mid- to low-level logical chunks of quantum code. Their assembly is not even yet done in practice.

The creation of quantum algorithms requires a capacity for abstraction that is beyond that of classical algorithms and programs, even taking into account classical object-oriented or event-driven programming. We will have to groom a new generation of mathematicians and developers capable of reasoning with the mathematical formalism of quantum programming as quantum computers mature.

They need to conceptualize algorithms that are not easy to mentalize, first using the Dirac notation formalism (Figure 3), using graphical tools like Quirk and then Python code.

Most of the time, though, quantum algorithms will not be simple language translations from classical programming language. They are also bound to solve problems that classical computers and classical programming languages cannot solve efficiently.

There is currently an interesting difference between classical computing and gate-based quantum software development. While classical computing can use various levels of abstraction for both coding and data structures, quantum computing currently uses only one coding abstraction level, gate-based circuits, as shown in Figure 4. From the highest coding level using frameworks down to

gate-based code generated by compilers and transpilers, qubits and qubit gates are always used, while in classical computing, nearly nobody cares about bits [8].

Quantum software tools can get rid of qubit formalism only with high-level libraries and tools that are fed with user data. One day, the abstraction level of quantum programming may rise to a point where it is no longer necessary to understand the low-level intricacies of quantum gates, Hilbert spaces, Hamiltonians and quantum interferences. In an interim stage, we will decompose quantum algorithms into subroutines [9, 10]. Some startups like **Classiq** and the **Qrisp project** [11] have started to play in that realm!

One could wonder how to write the first “**Hello, World!**” program using a quantum computer. One example can be found in Microsoft’s C# code samples [12] but the “Hello, World!” part is entirely classical! The real “Hello World!” template of quantum programming is to create your first Bell state by coupling a H and CNOT gate and generate a two-qubit entangled state $(|00\rangle + |11\rangle)/\sqrt{2}$.

Quantum computing paradigms

Today’s classical quantum algorithms use quantum gates. But there are other quantum computing paradigms:

- **Quantum annealing problem solving** such as those for D-Wave machines which are based on the initialization of relations between average quality qubits and on the search for a minimum energy based on the tunnel effect. The basic approach contains two steps: reformulating an initial problem into a QUBO or Ising problem and then solving it by a dedicated quantum approach [13]. We will describe it when discussing D-Wave.
- **Analog quantum simulators** are used to simulate quantum phenomena, for example to predict the organization of atoms in molecules. These include cold atom quantum simulators. An algorithm here is about preparing the state of the qubits in the system and their link weights. It is a process similar to D-Wave quantum annealing, with variations on the degrees of freedom handled in the system and on qubit coherence.
- **Continuous variable quantum computers** that use quantum objects whose physical quantity can be measured as a continuous, not binary, quantity. This creates yet another programming model. They are mainly based on photons [14, 15].
- **Topological quantum computers**, which do not yet exist. This is the research path of Microsoft and some research laboratories, especially in China. It should still be programmable with gate-based classical code.
- **Variational algorithms** combining traditional algorithms and quantum algorithms running on any of

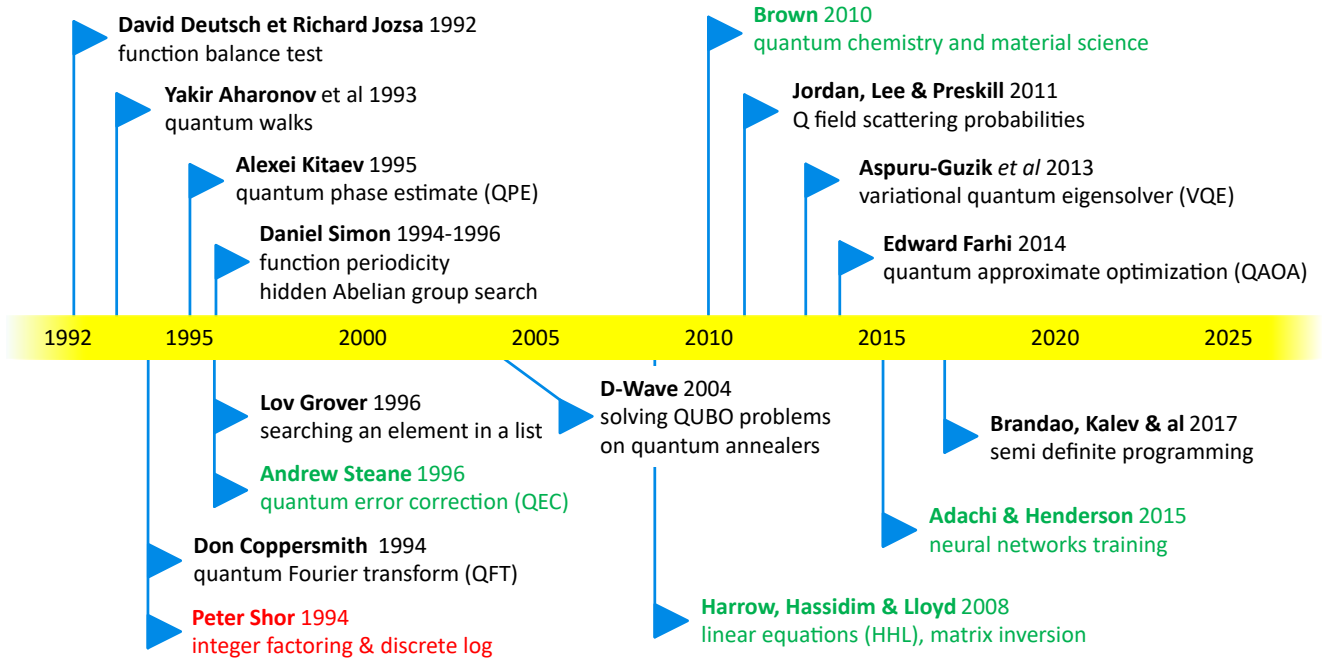


Figure 2: A quantum computing algorithms creation timeline. It is a three-decade story. © Olivier Ezratty, 2021-2024

uniform preparation or Hadamard-Walsh transform

$$H^{\otimes n} |0\rangle = \frac{1}{2^{n/2}} \sum_{j=0}^{2^n-1} |j\rangle = |+\rangle^{\otimes n}$$

expressed as a simple tensor product

modular exponentiation

$$|\psi_1\rangle = \sum_{x=0}^{2^n-1} \alpha_x |x\rangle \otimes |a^x \bmod N\rangle$$

α_x are the amplitudes of the input state
a is an integer (see Shor's algorithm description)

Quantum Fourier Transform

$$|\psi_1\rangle = \frac{1}{\sqrt{N}} \bigotimes_{j=1}^n \left(|0\rangle + e^{-2\pi i \left(\sum_{k=n-j}^n x_k 2^{-k} \right)} |1\rangle \right)$$

expressed as a complicated tensor product

$$|\psi_1\rangle = \frac{1}{\sqrt{N}} \begin{bmatrix} 1 & 1 & 1 & 1 & \dots & 1 \\ 1 & \omega & \omega^2 & \omega^3 & \dots & \omega^{N-1} \\ 1 & \omega^2 & \omega^4 & \omega^6 & \dots & \omega^{2(N-1)} \\ 1 & \omega^3 & \omega^6 & \omega^9 & \dots & \omega^{3(N-1)} \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \omega^{N-1} & \omega^{2(N-1)} & \omega^{3(N-1)} & \dots & \omega^{(N-1)(N-1)} \end{bmatrix} |\psi_0\rangle$$

expressed as a unitary matrix

$\omega_N = e^{-2\pi i/N}$, $\omega_N^n = \omega_N^{1/n}$, $\omega^n = \omega_1^n = (\omega_1)^{1/n} = e^{-2\pi i/n}$

Figure 3: Examples of algorithms expressed in Dirac notation, using tensor products, and expressed as a unitary matrix, for the case of a quantum Fourier transform. One key learning exercise, not done here, is to find out what each of the coefficients mean and where the input and output data are for these algorithms of circuits. For more on the structure of a quantum circuit, see the lecture notes in [7]. © Olivier Ezratty, August 2024.

the above systems [16]. These are algorithms mostly designed for gate-based NISQ QPUs and a part of the related section is dedicated to them.

We will mention **quantum inspired algorithms** which are algorithms running on classical computers that are inspired by quantum algorithms, formalism or mathematics for solving complex problems. Their creation started long before the first experimental quantum computers were created.

In practice, NISQ systems will dominate the landscape until the late 2020's.

Namely, because quantum gates and readout error rates are too high and limit the number of quantum gates

that can be chained, even when using quantum error mitigation techniques.

This is the case for **VQE** (Variational Quantum Eigensolver), **QAOA** (Quantum Approximate Optimization Algorithm), and various **Quantum Machine Learning** algorithms (Support Vector Machine, Principal Component Analysis and Quantum Variational Autoencoder). We will study some of them in a dedicated part related to NISQ algorithms. We will also look at how quantum software handles data input and output and how it is being debugged.

We explain in Figure 6 and Figure 7 how to differentiate gate-based and analog quantum computing models, first with the three main models and then with mak-

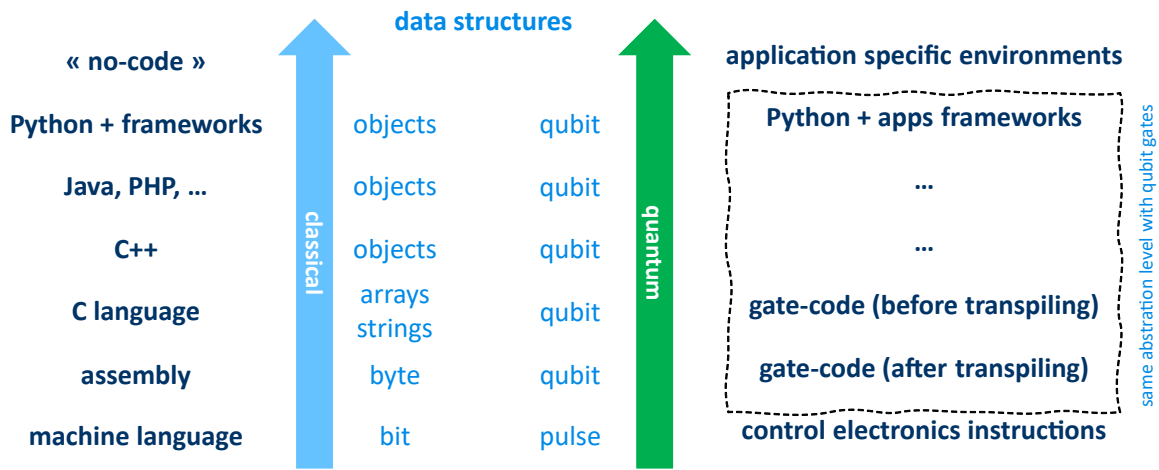


Figure 4: Quantum languages and data abstraction levels. © Olivier Ezratty, 2023. Some proposals for higher level of abstraction gate-based quantum coding have already been made, by Classiq and with the Qrisp project which led to Qmod.

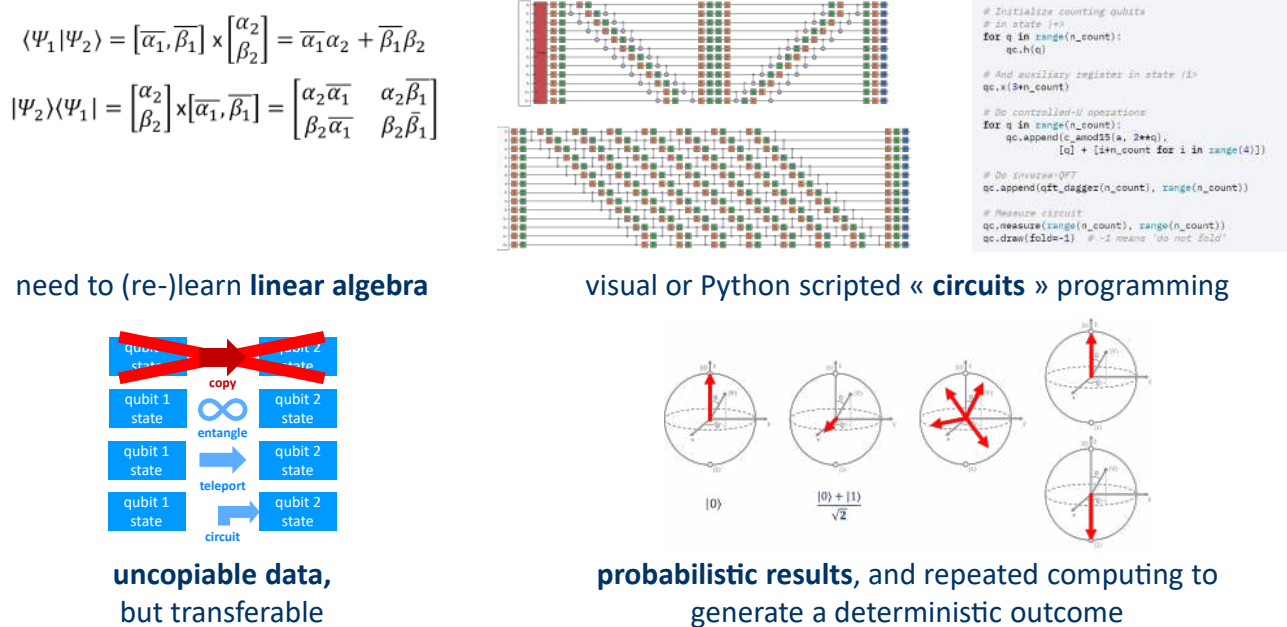


Figure 5: Key differences of quantum programming: a need to understand linear algebra and do some maths, different debugging techniques and coping with the impossibility to copy data and playing with the probabilistic nature of quantum measurement. Gate-based programming can use graphical tools like Quirk for code learning and to visualize interactively qubits values (Bloch sphere, vector state, density matrix) in emulation mode. Python based scripted code is used for professional programming. © Olivier Ezratty, 2022-2026.

ing analogies between analog quantum computing and neural networks training.

1.1 Quantum algorithms classes

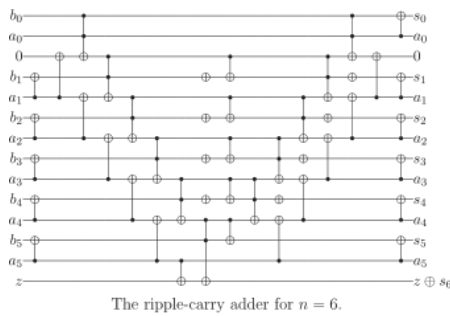
Before getting deep into quantum algorithms, let's take a detour by covering their practical usefulness known to date and for which category of quantum hardware they are designed. Then, how they are organized and what is the basic algorithms toolbox available to developers.

As a reminder, a quantum algorithm is all the algorithms components solving a given problem. This comprises, of course, the quantum circuit to run on a quantum

computer which contains a set of gate and readout operations implementing unitary transformations in the case of gate-based quantum computing. It also contains all the related classical algorithms which generate the quantum circuit and handles its pre- and post-processing tasks.

A quantum circuit acting on N qubits is represented by a unitary operator in $SU(2^N)$, corresponding to a matrix of size $2^N \times 2^N$, containing 2^{2N} complex entries. Due to the unitary constraints, the unitary has $2^{2N} - 1$ real numbers degrees of freedom. In comparison, a pure quantum state on N qubits requires $2^{N+1} - 2$ real parameters once normalization and global phase are taken into account.

gates-based quantum computers

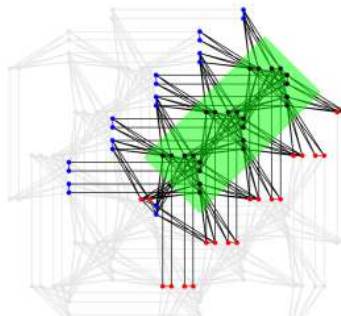


sequential programming of quantum gates, can implement any algorithm and Hamiltonian transformation



QUANDELA

quantum annealers

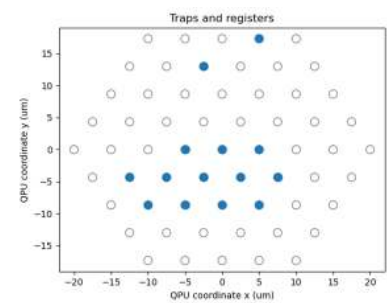


problems are embedded in a BQM model to solve Ising or QUBO problems, using **static qubit connectivity** and **local control**



The Quantum Computing Company™

quantum simulators



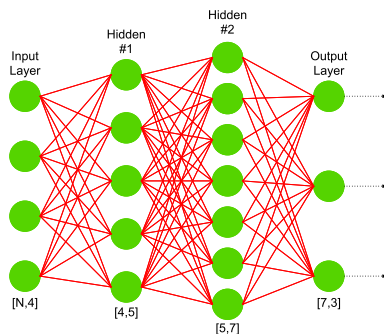
problems are embedded in a graph to solve Ising or QUBO problems, using **dynamic qubit positioning** but **no or poor local qubit control**



Pasqal | QuEra

Figure 6: How to differentiate the three main quantum computing paradigms (gate-based, annealing, and simulations). The main difference between the first and two other deals with sequential programming in gate-based model while the two others use a classical preparation of sorts of networks of qubits for which we are searching for an energy minimum. © Olivier Ezratty, 2023-December 2024.

deep learning



variables

neural network hyperparameters
neuron activation functions
training data

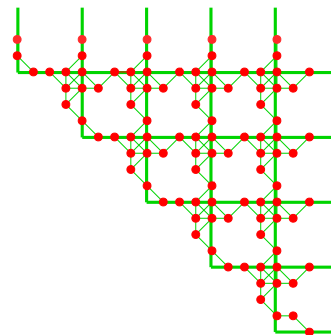
unknowns

neurons link weights

training

minimize cost function with gradient descent and backpropagation

analog quantum computing



link weights between qubits and qubit weight
qubit connectivity / graph

training data that conditions the above values in classical computing

qubit value (-1 or +1 / 0 or 1) with Hamiltonian ground state

solving BQM, QUBO or Ising models, classical-quantum hybrid

Figure 7: Here is a way to understand the similarities and differences between deep learning neural networks and analog computing (annealing and simulations). The main difference sits with the variables (green) and unknowns (red). They are reversed! With analog quantum computing, the variables are the links between qubits while these are the unknowns in neural networks. The unknowns in analog quantum computing are the qubit weights (0 or 1, or -1 and -1 depending on the model, Ising or QUBO) while with neural networks, the neuron characteristics like their activation function are their variables. © Olivier Ezratty, 2023.

One such unitary can be decomposed in different series of gates, meaning several algorithms can potentially exist that solve a given problem, the most efficient implementation generally minimizing the relevant resources such as gate count, circuit depth or execution time. It is a bit like finding the shortest path in a hyperspace

(the Hilbert space) using specific steps (the quantum gates) [17].

Generic software tools are then using the generated quantum circuit to compile, optimize, transpile it into physical operations run on the quantum computer. This transformation will be significant in the case of FTQC,

which adds a lot of quantum error correction overhead with their wealth of error syndrome measurements and corrections.

1.1.1 Classes and use cases

Here's a simple classification of high-level algorithms by use cases [18]. There are many such classifications around. This is the most common one.

Quantum physics simulation is a broad field with applications in inorganic and organic chemical processes optimization and new material designs. This is based on simulating at the lowest level the interactions between atoms in molecules and crystal structures or magnetism, which themselves depend on the laws of quantum mechanics. This may someday help invent new solutions such as more efficient batteries that can be charged more quickly and with greater energy density, craft chemical processes for carbon capture or nitrogen fixation or create superconducting materials operating at room temperature.

Biological molecule simulations require a much larger number of qubits and are therefore positioned in the longer term. Quantum simulation may eventually help run simulations of biological molecules. This will start with the simulation of peptides, then polypeptides, and finally protein folding and interactions. Biological molecules have the particularity of being overly complex, with structures that can reach tens of thousands of atoms. The top of the line would be the ability to simulate the assembly and then the operation of a ribosome, which is more than 100,000 atoms. It is the most magical molecular structure in living organisms, the one that assembles amino acids to build proteins from the messenger RNA code resulting from the transposition of gene DNA. This would be followed by the simulation of the functioning of a whole cell. But we are here bordering on science fiction. Practical chemical simulations need to deal with finding the ground or excited states of many-body quantum systems, computing their spectra, simulating rotational and vibrational states, and various chemical pathways and reactions.

Oracle function-based algorithms can speed up the search of a needle in a haystack and find a solution to some decision problems. Some are useful, some are not. The most famous oracle-based algorithms are Deutsch-Jozsa, Simon, Bernstein-Vazirani and Grover. They usually solve decision and combinatorial optimization problems.

Other algorithms to solve complex optimization problems, particularly combinatorial problems like finding an optimal route for deliveries or automated driving, aka the traveling salesman problem. Such algorithms can also optimize the design of integrated circuits where one generally seeks to minimize the links between functional blocks, transistors and to minimize energy consumption, forms of sub-constrained optimiza-

tion adapted to quantum processing. This category of algorithms can find solutions to combinatorial optimization problems like discrete log, traveling salesperson problem (TSP), QUBO problems, and continuous optimization problems used in many other fields (linear programming, gradient descent, convex optimizations, and semidefinite programming).

Quantum machine learning algorithms which under some circumstances could be more efficient than machine learning algorithms running on classical hardware, including GPGPUs and TPUs. This can impact both machine learning and deep learning models training and running inferences.

Cryptanalysis problems relate to breaking public encryption keys like RSA keys with Shor's integer factoring algorithm. These may be implemented over a very long term, when highly scalable gate-based quantum computers are available. Their economic value is questionable!

Hybrid algorithms, as already mentioned, use a mix of quantum and classical algorithms with some variations presented in Figure 8. As a matter of fact, all practical quantum computations require classical control, since they are driven by some classical computer that prepares data and quantum code. But the label hybrid quantum-classical algorithm usually designates workflows with substantive classical and quantum computation, often in an iterative loop. These are most often variational algorithms, mostly in the NISQ realm, that have a classical and quantum loop that ends at some point when a cost function converges to an optimum. But oracle-based algorithms with the oracle retrieving classical data and even Shor's integer factoring algorithm also have a significant classical part and are thus also hybrid classical/quantum algorithms.

Otherwise, many quantum algorithms require some classical data preparation and loading that can be a heavy-weight task handled by classical computers before tasking a QPU with a simpler task using prepared data. This is particularly true with quantum machine learning algorithms, whether they rely or not on some variational process.

There are also vertical and horizontal hybridization techniques like parallelizing processing on several specialized QPUs [19]. Likewise, a QPE algorithm to find the ground state of a molecule must be fed with an energy estimation prepared classically using a DFT or equivalent method.

Quantum inspired algorithms are classical algorithms like genetic algorithms that are inspired by quantum properties, particularly interferences which are key characteristics of quantum algorithms. These algorithms are usually based on tensor network frameworks implementing matrix/vector optimized classical computing. The "quantum" in quantum inspired is viewed as a misnomer by many.

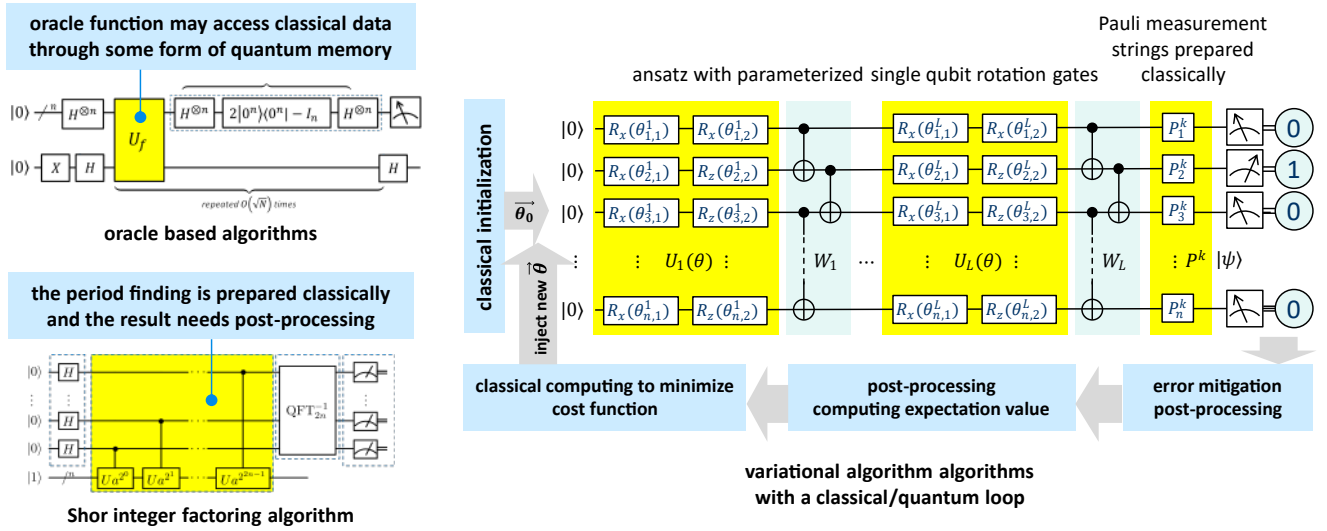


Figure 8: There are many sorts of hybrid algorithms. The dominant one corresponds to variational algorithms using an ansatz function that is prepared classically to compute a function cost that is then used classically in a loop to optimize the ansatz. But oracle based algorithms need to access classical data, and even Shor’s integer factoring algorithm has a preprocessing part prepared classically and dynamically, for the implementation of its period finding function. © Olivier Ezratty, 2023-2024

1.1.2 Algorithms and quantum computing paradigms

As shown in Figure 9, there’s a relation between these broad classes of algorithms and the class of quantum computers they can run on. Reusing the quantum algorithms paradigm classification used earlier in this book, this gives an idea of what works where and when, given these computer classes span from available systems (quantum annealing), to NISQ systems in the short to mid-term, to very long-term availability (large scale quantum computing). It is still an open question to find relevant and useful algorithms running on NISQ systems, if not reaching a quantum advantage with these [20, 21].

You can also create analogies to figure out what quantum computing is from a mind model standpoint, as shown in Figure 10. The first one is simple: it is all about mathematics and linear algebra, manipulating large matrices and vectors. A quantum gate is a (unitary) matrix multiplied by a (state) vector. You then decompose a large problem into a series of such unitaries. The second model is related to analog electronics. With quantum computing, you happen to reuse similar concepts, manipulating signals phase, amplitude, decomposition (QFT) or recomposition (inverse QFT) and interferences.

Finally, quantum annealing, and quantum analog simulations have some similarities with neural networks, with some differences, already shown in Figure 7.

1.1.3 Algorithms process and compilation

As we have seen in a previous section describing the structure of gate-based quantum computing, a quantum

algorithm is built with three key parts: data initialization or preparation, computing and qubits readouts. This is always done on a data structure called a quantum register, made of N qubits. Data initialization, preparation and computing are implemented with quantum gates.

Results. The algorithm result comes from the classical measurement of some qubits, which returns one bit string per shot. Repeated shots estimate the output distribution, so it is generally necessary to run the whole algorithm several times and aggregate the generated results. How many times must it be done? It depends on the nature of the algorithm and the speed at which we’ll move from a probabilistic output (one run) to a deterministic result (average of several runs).

Time constraints. The algorithm run must be compatible with the quantum computer characteristics. The main ones are qubits numbers, gates and readout fidelities and coherence time. These parameters will condition the usable depth of computing, *aka*, the number of series of gates that can be executed. This verification is generally performed by quantum code compilers.

It will also have to consider the error correction codes that will be implemented in the hardware, either autonomously or through the control of the code compiler that will drive all logical qubits programming.

Gates conversion. Compilers play another key role: they translate the qubit gates used by the programmer into the set of physical qubit gates implemented at the hardware layer. Many quantum gates used by developers will be converted by the compiler into a set of universal quantum gates supported by the quantum computer. This will multiply the number of executed

quantum advantage some useful cases no useful case so far 	classical computers	quantum annealers	analog quantum simulators	gate-based quantum computers	
				NISQ	FTQC
search algorithms	hybrid algorithms				polynomial speedup
optimization algorithms	hybrid algorithms			QAOA	may require qRAM polynomial speedup
quantum machine learning	hybrid algorithms				may require qRAM advantage still TBD
physics simulation	hybrid algorithms	Ising models	Ising models	VQE, Ising models	100 to 10K logical qubits
organic chemistry simulation	hybrid algorithms	hard for fermionic simulations	hard for fermionic simulations	VQE not scaling well	100 to 10K logical qubits
integer factoring	hybrid algorithms	6-digit record to date		15-digit record to date (QAOA, China)	4K-6K logical qubits
quantum inspired algorithms					
	now	now	soon	later	much later
« quantumness » and arrow of time availability →					

Figure 9: Classes of problems and the quantum computing paradigms (gate-based, analog simulation annealing) which can solve them, and a related time scale for their practical availability. Surprisingly, integer factoring algorithms are also available on quantum annealers and simulators, but it may not scale as well as future FTQC systems. There are subtle nuances on the scalability of all these platforms, including analog quantum computers. Analog quantum computing platforms can theoretically solve a lot of problems, but are particularly adapted to solve graph problems and Ising models in fundamental physics, while the FTQC paradigm is the most generic one. But its advent is still uncertain and in a time window between 5 and 15 years depending on the estimates. In the FTQC regime, search, optimization and machine learning algorithms have an uncertain fate due to either a polynomial speedup that may not turn into a practical speedup, or unproven speedups. © Olivier Ezratty, 2021-2025.

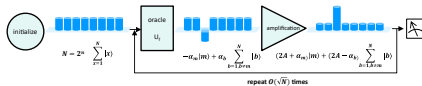
unitary transformation for a 3-qubit gate

$$\begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \\ \alpha_5 \\ \alpha_6 \\ \alpha_7 \\ \alpha_8 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \\ \alpha_5 \\ \alpha_6 \\ \alpha_8 \\ \alpha_7 \end{bmatrix}$$

linear algebra

vector and matrix computing,
state vectors, density matrices,
parallelism

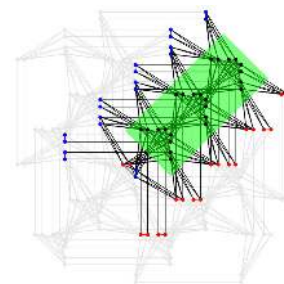
Grover amplitude amplification



analog electronics

signals phase and amplitude,
interferences, amplification, signal
analysis (QFT), phase detection

D-Wave qubits connectivity



neural networks

analog quantum computing,
solving Ising models

Figure 10: Three analogies to understand how quantum computing works. The mathematical approach is quite simple in principle: quantum computing is about multiplying vectors by high-dimensional matrices. The analog electronics approach is related to many similar concepts used in quantum algorithms which are akin to electronic signal processing (phase estimation, amplitude estimation, Fourier transform and inverse transform, etc.). The last one is related to solving graph problems and has some similarities with neural networks. The two first relate to gate-based quantum computing and the last one to quantum annealing and quantum simulation. © Olivier Ezratty, 2023-2024.

physical quantum gates compared to what shows up in the initial algorithm.

Geometry. They also take into account the physical geometry of qubits, i.e. how are they connected together. A simple two-qubit gate might require chaining a lot of

SWAP gates because the two related qubits are far from each other in the quantum register physical layout.

Efficiency. An important consideration in creating quantum algorithms is to ensure that they are more efficient than their optimized counterparts for traditional computers or supercomputers. There are theories to

verify this to evaluate the exponential, polynomial, logarithmic or linear rise in computing time as a function of the size of the problem to be solved, or a combination of all four. But nothing can replace experience!

Nearly everything is linear. Quantum algorithms are practical applications of linear algebra, the branch of mathematics that handles vector spaces and matrix-based linear transformations. They are applied in large dimension spaces of vectors that define the states of qubit registers. Mathematically speaking, a qubit is in a 2-dimensional vector space using complex numbers and an N -qubit register manipulates a vector in a 2^N -dimensional space of complex numbers. Their manipulation is based on matrix-based calculations that allow the qubit state to be modified without reading the content of the qubits. One way to look at a gate-based quantum algorithm would be the following: it is about finding the shortest path on a hypersphere of dimension 2^N from an initialized register to the problem solution [22]. But it is possible to introduce non-linearity in quantum programming, particularly with mid-circuit measurement and feed-forward.

Conditional programming. Since quantum algorithms usually prohibit reading intermediate results, conditional programming is not obvious. Like running a given calculation depending on the value of some intermediate. However, multi qubits quantum gates (CNOT & co) are tools allowing conditional programming, but in another fashion than with classical computing. Conditional branching can be implemented in some situations and is used with hybrid algorithms and with ancilla qubits for intermediate values measurements (mid-circuit measurement). It also plays a role in the MBQC programming paradigm.

1.1.4 Algorithms toolbox

All these algorithms are based on a small set of classical low-level algorithms that we'll describe in detail in the basic algorithm's toolbox section (see also Figure 11):

- **Quantum Fourier Transforms (QFT)**, which helps find periods in a signal. It is used in Shor integer factoring and discrete logarithm (dlog), in many other algorithms (HHL, QML), for solving the hidden subgroup problem (HSP) and even for simple reversible arithmetic (Table 1) [23].
- **Quantum Phase Estimation (QPE)** is relying on a QFT and a modular exponentiation to find the eigenvalues or eigenvalues' phase of a unitary matrix or quantum subcircuit. It is used in HHL and many other quantum linear algebra algorithms.
- **Quantum Amplitude amplification (QAE)** is used to amplify and select the desired state of a quantum superposition. It is used in the Grover algorithm and with combinatorial searches like the traveling salesperson problem search (TSP).
- **Linear equations solving** and the famous HHL algorithm and various partial derivative equation solving algorithms.
- **Hamiltonian simulations** are used to find a point of equilibrium of a complex system such as in quantum physics simulation, neural networks training, the search for optimal paths in networks or process optimization. It can be implemented in all quantum paradigms: annealing, simulation and gate-based computing (with a FTQC QPE or a NISQ VQE).
- **BQM** or binary quadratic model is the mechanism used to drive quantum annealers. This is the underlying model used with D-Wave machines. Many physics simulation (Ising models), combinatorial and optimization problems (QUBO problems) can be translated or converted into BQM, as explained in Figure 13.

We can add here several other key basic algorithms components:

- **Data preparation:** how is data loaded in an algorithm? This is particularly important for quantum machine learning and optimization algorithms.
- **Uncompute trick:** with reversing some parts of an algorithm after it is run. It allows you to get rid of garbage states and clean up ancilla qubits by disentangling them from other qubits.
- **Oracles:** are binary functions implemented as unitaries that can be used for parallelizing their operation on all computational state basis (all combinations of 0s and 1s in part of a qubits register). They are used in Grover, Deutsch-Jozsa, Bernstein-Vazirani, Simon and QAE algorithms.
- **Quantum teleportation** is an algorithm basic, used mostly in cryptography and telecommunication. It will also play a key role in distributed quantum computing and in some non-telecom related algorithms. It is used in the "gate teleportation" trick which is hard to understand.
- **Quantum phase kickback** is an interference trick used in most oracle-based search algorithms and quantum walks, and then in quantum machine learning.

Classifying quantum algorithms is a tedious task due to the many dependencies they have on each other. For example, a QFT is used in HSP and phase-estimation algorithms which themselves are used in integer factoring and linear equations solving.

There are many different if not inconsistent algorithms classifications in the available literature ([Wikipedia](#), [John Preskill](#), [Algorithm Zoo](#), etc.). Some consider oracle-based algorithms as a separate algorithm class when other split these algorithms in various classes depending on the sub-algorithms they are using. A 2024 exhaustive classification work done by researchers from

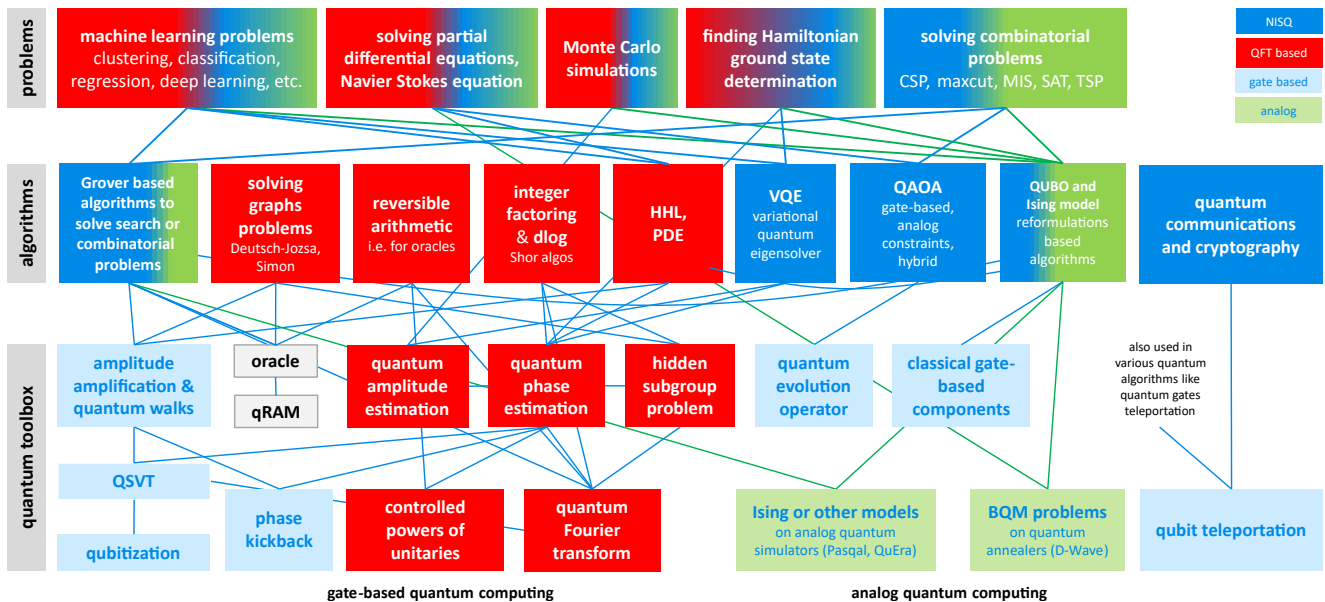


Figure 11: Simplified quantum algorithms inventory with problems (top), key algorithms (middle) and basic algorithm components (bottom). The color code is red for FTQC problems/algorithms relying on the QFT (quantum Fourier transform) and/or controlled powers of unitaries or modular exponentiation in the case of Shor algorithms, blue for problems/algorithms corresponding to other gate-based algorithms that may work in NISQ QPUs and green for problems/algorithms related to analog quantum computing paradigms like quantum annealing (*ala* D-Wave) and quantum simulations (*ala* Pasqal). Many high-level problems can be solved using an FTQC QFT based version and some analog equivalents, usually based on a QUBO or Ising problem formulation. © Olivier Ezratty, 2023-2025, initially inspired by a schema found on [24].

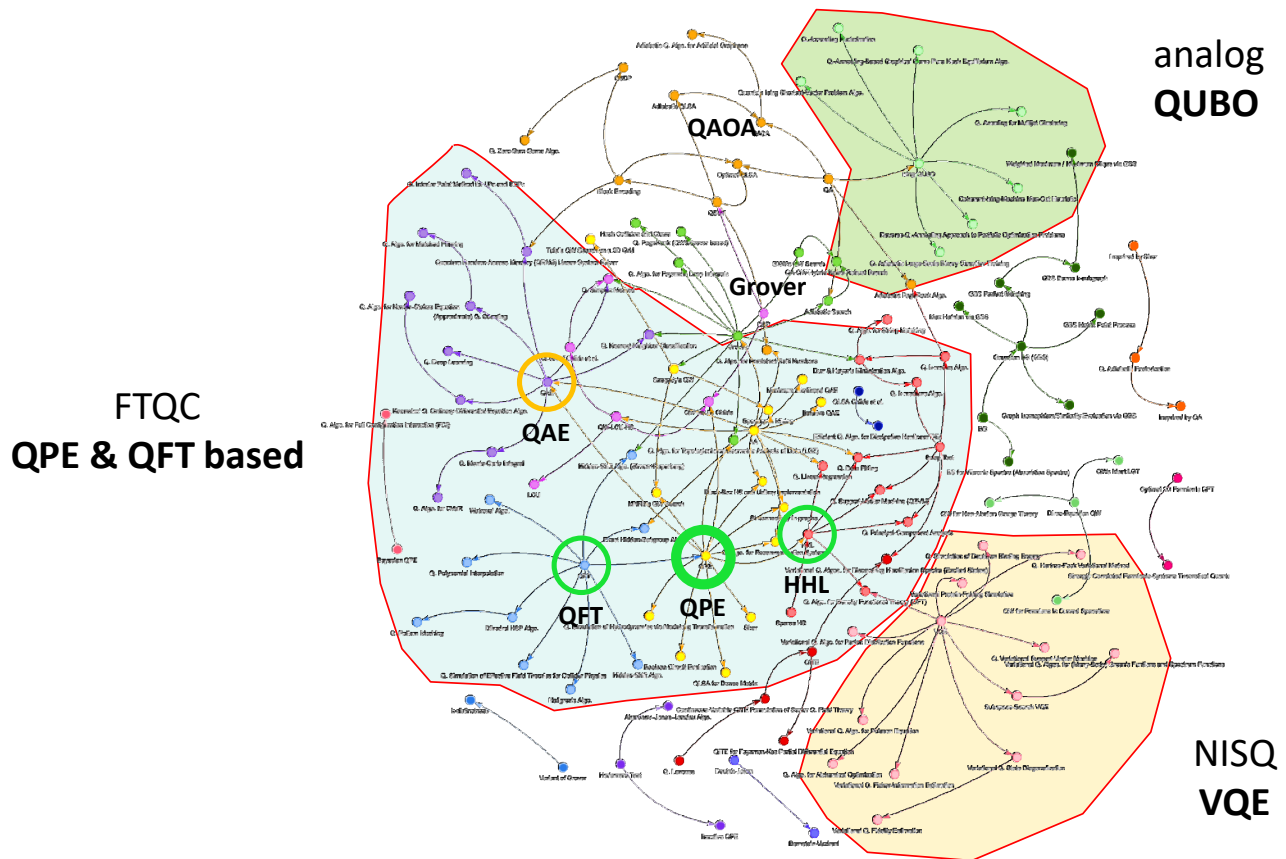


Figure 12: A classification of the main quantum algorithms classes with an overview of the three main buckets: analog algorithms (QUBO), NISQ algorithms (VQE/VQA based) and FTQC algorithms mostly all dependent on the QPE. Source: [25]. With additions from Olivier Ezratty, January 2025.

France and Quantinuum is more thorough although its
Quantum algorithms

graphic representation is not easy to grasp (Figure 12Fig-
ure 12, zoom on the PDF to look at it).

They define 6 algorithms classes (Hidden-Subgroup Problems, linear algebra, dynamical systems, stochastic processes and statistics, optimization and combinatorics) and 6 application domains (cryptanalysis, machine learning and data science, first-principle quantum simulation, classical scientific computing, operational research and quantum enablement). It shows qRAM for quantum RAM, which is not an algorithm per se, but a hardware tool that is indispensable to run the related algorithms, particularly Grover's search algorithm and a many quantum machine learning algorithms.

Quantum algorithms are classifiable and explainable at a high level, but their detailed understanding requires some conceptual capacity in a rather analog world [3, 26–30]. One key thing that a developer must learn is how to translate customer needs into existing quantum algorithms. How to assemble various quantum algorithms, frequently combined with classical algorithms, is another key skill.

1.1.5 How is a quantum algorithm created?

As an end-user, you may wonder about how to create an algorithm corresponding to your need. One way to respond to this question is to look at the common practices in classical computing. Are users creating algorithms or software? You usually reuse existing algorithms and build software chains around them. Common examples are database searches, quick sort, FFT in signal processing, linear and non linear regressions, the zoo of machine learning algorithms, Monte Carlo simulations, linear and integer programming to solve combinatorial algorithms, and so on.

With quantum computing, you can anticipate the emergence of similar practices. Fundamental algorithms and tools are created by scientists in this field and end-

user will integrate multiple algorithms in their solution. These will combine classical algorithms and quantum circuits. The customization will frequently come from the way the problem is converted into a circuit including its data encoding. Data will need to be prepared and cleaned up. We can also expect to see the expansion and emergence of multiple high-level software tools and SDK coming from software and hardware vendors.

We will probably see the advent of quantum software architects who will design software solutions assembling multiple components. They will also be involved in selecting the supporting hardware, some HPC to handle pre- and post-processing as well as one of multiple QPUs. They will design solutions pipelines, like in the case of chemical simulations where multiple questions need answers: computing ground states, excited states, free energies and molecular dynamics, and where the solution will always be a mix of classical and quantum computing. The magic will be more in the recipe than in the ingredients.

1.1.6 Algorithms figures of merit

It is rarely talked about, but what are the key figures or merits for quantum algorithms? In a generic way, they should showcase either a provable speedup compared with classical algorithms running on classical computers or some other quantitative advantage (precision, error rates for quantum machine learning, energy consumption advantage). They should generate a relatively small-size data output since N qubits generate only N useful bits of information. Finally, there should be some correctness guarantee on the results and of course, it should solve some useful problems. In the case of NISQ algorithms, they should add two requirements: a shallow depth circuit (not many quantum gate cycles) and a resilience to qubit noise [42].

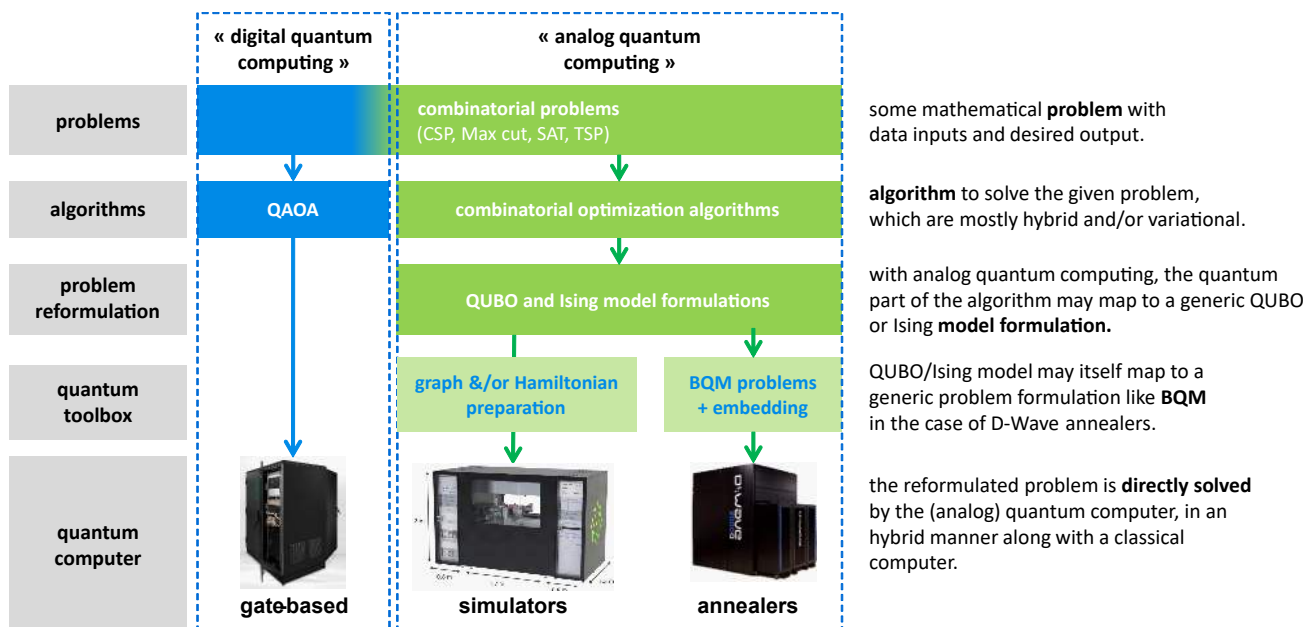
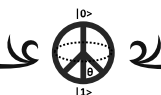


Figure 13: Some clarification between the notion of problem, algorithms, and models, particularly in the analog quantum computing domain. In that domain, the quantum part of a hybrid algorithm is mainly a problem reformulation into another generic problem that is directly solved by the quantum computer in a single step. © Olivier Ezratty, 2023-2024.

Algorithm	Problem	Quantum complexity	Classical comparison	Main quantum primitive	Reference
Algorithms Based on Fourier Transforms and Hidden Structure					
Shor	Integer factorization and discrete logarithms through period/order finding.	For $n = \lceil \log_2 N \rceil$: $O(n^3)$ with standard arithmetic.	Polynomial quantum time versus no known polynomial-time classical algorithm for general factoring or the relevant discrete logarithm problems.	QPE, QFT, modular exponentiation.	Shor [31].
Simon	Find the hidden string s such that $f(x) = f(x \oplus s)$.	$O(n)$ oracle queries.	Classical randomized query complexity: $\Omega(2^{n/2})$. Exponential oracle separation.	Hadamard transform $H^{\otimes n}$ and linear algebra over $\mathbb{F}_2$.	Simon [32].
Deutsch-Jozsa	Determine whether a promised Boolean function is constant or balanced.	Exactly 1 oracle query.	Exact deterministic classical complexity: $2^{n-1} + 1$. Bounded-error randomized complexity is $O(1)$.	Hadamard transforms and quantum interference.	Deutsch and Jozsa [33].
Bernstein-Vazirani	Recover $s \in \{0, 1\}^n$ from $f_s(x) = s \cdot x \pmod{2}$.	Exactly 1 oracle query.	Classical deterministic complexity: n queries. Linear query advantage.	Hadamard transforms and phase kickback.	Bernstein and Vazirani [34].
Kitaev	Abelian stabilizer problem, encompassing factoring and discrete logarithms.	Quantum polynomial time.	Includes problems with no known general polynomial-time classical solution.	QPE and Fourier transforms over finite Abelian groups.	Kitaev [35].
van Dam / Hallgren	Recover hidden shifts of quadratic characters.	For the principal problems: 2 oracle queries and polynomial running time.	General classical computational complexity remains unknown. No unconditional asymptotic separation is established.	QFT, character phases and Gauss sums.	Dam and Hallgren [36].
Watrous	Black-box solvable groups: group order, membership, equality and normality.	Quantum polynomial time.	No polynomial classical algorithms are known in general for the considered black-box group problems.	Quantum group states, Fourier techniques and order finding.	Watrous [37].
Hallgren	Pell's equation and the principal ideal problem in real quadratic number fields.	Quantum polynomial time in the input size.	No polynomial-time classical algorithm is known for the general problems.	Quantum period finding and Fourier sampling.	Hallgren [38].
Algorithms Based on Quantum Search and Amplitude Amplification					
Grover	Find a marked item among N unstructured possibilities.	$O(\sqrt{N})$ oracle queries.	Classical unstructured search: $\Theta(N)$. Quadratic speedup.	Amplitude amplification and inversion about the mean.	Grover [39].
Traveling Salesman Problem	Find the minimum-cost Hamiltonian tour over n cities.	Hybrid quantum dynamic programming: $O^*(1.728^n)$.	Held-Karp classical dynamic programming: $O^*(2^n)$. The quantum algorithm remains exponential.	Quantum search combined with classical dynamic programming.	Ambainis [40].

Table 1: Representative quantum algorithms using a QFT with their computational advantages, and their main quantum primitives. For Shor's algorithm, $n = \lceil \log_2 N \rceil$ is the bit length of the integer being factored. For Simon, Deutsch-Jozsa and Bernstein-Vazirani, n is the oracle input size. For TSP, n is the number of cities. $O^*(\cdot)$ suppresses polynomial factors. Query complexity counts oracle calls and is distinct from total gate or runtime complexity. Source: corrections on a table from 2018 found in [41].



1.1.7 Bibliography and notes

These are the bibliographical references and notes for the **Algorithms classes** subsection.

- [1] [Simulating Physics with Computers](#), by Richard Feynman, International Journal of Theoretical Physics, 1982 (22 pages).
- [2] [Quantum Algorithm Zoo](#), by Stephen P. Jordan, Quantum Algorithm Zoo as of May 2025.
- [3] [Lecture Notes on Quantum Algorithms](#), by Andrew M. Childs, 2017 (174 pages).
- [4] [Quantum Computing Lecture Notes](#), by Ronald de Wolf, 2021 (184 pages).
- [5] [Why Haven't More Quantum Algorithms Been Found?](#), by Peter Shor, 2003 (4 pages).
- [6] [Quantum Algorithms: A Call To Action](#), by Robbie King, Quantum Frontiers, April 2025.
- [7] [Rethinking Quantum Circuits](#), by Steven Rayan, arXiv, August 2026 (75 pages).
- [8] [Neither Software Developers nor Users Need to Know about Bits](#), by Jack Krupansky, March 2023.
- [9] [Quantum Subroutine Composition](#), by Stacey Jeffery, arXiv, February 2025 (49 pages).
- [10] [Why Are We Unrolling? The Importance of Structured Quantum Programs for Compilation](#), by Damian Rovara, Robert Wille, David Ittah, et al., arXiv, September 2026 (8 pages).
- [11] [Qrisp: A Framework for Compilable High-Level Programming of Gate-Based Quantum Computers](#), by Raphael Seidel, Manfred Hauswirth, et al., arXiv, June 2024 (19 pages).
- [12] [Structure of a Q sharp program](#), by Microsoft, Microsoft.
- [13] [Ising formulations of many NP problems](#), by Andrew Lucas, arXiv, January 2014 (27 pages).
- [14] [Toward large-scale fault-tolerant universal photonic quantum computing](#), by Shuntaro Takeda and Akira Furusawa, APL Photonics, June 2019 (11 pages).
- [15] [Continuous-variable quantum neural networks](#), by Nathan Killoran, Thomas R. Bromley, Juan Miguel Arrazola, Maria Schuld, Nicolás Quesada, and Seth Lloyd, arXiv, June 2018 (21 pages).
- [16] [Hybrid Quantum Computation](#), by Arun Sehrawat, Daniel Zemann, and Berthold-Georg Englert, arXiv, September 2010 (13 pages).
- [17] [Grover's algorithm is an approximation of imaginary-time evolution](#), by Yudai Suzuki, Marek Gluza, Jeongrak Son, Bi Hong Tiang, Nelly H. Y. Ng, and Zoë Holmes, arXiv, July 2025 (28 pages).
- [18] [Quantum algorithms for scientific computing](#), by R. Au-Yeung, B. Camino, O. Rathore, and V. Kendon, arXiv, October 2024 (43 pages).
- [19] [Classification of Hybrid Quantum-Classical Computing](#), by Frank Phillipson, Niels Neumann, and Robert Wezeman, arXiv, October 2022 (8 pages).
- [20] [Noisy intermediate-scale quantum \(NISQ\) algorithms](#), by Kishor Bharti, Hermanni Heimonen, Leong-Chuan Kwek, Alán Aspuru-Guzik, et al., arXiv, October 2021 (91 pages).
- [21] [Simultaneous execution of quantum circuits on current and near-future NISQ systems](#), by Yasuhiro Ohkura, Takahiko Satoh, and Rodney Van Meter, arXiv, December 2021 (10 pages).
- [22] [Wasserstein Complexity of Quantum Circuits](#), by Lu Li, Kaifeng Bu, Dax Enshan Koh, Arthur Jaffe, and Seth Lloyd, arXiv, August 2022 (7 pages).
- [23] [A New Approach to Multiplication Opens the Door to Better Quantum Computers](#), by Kevin Harnett, 2019.
- [24] [Quantum Computing Algorithms](#), by Andreas Baertschi, 2019 (45 slides).
- [25] [A typology of quantum algorithms](#), by Pablo Arnault, Pablo Arrighi, Steven Herbert, Evi Kasnetsi, and Tianyi Li, arXiv, July 2024 (60 pages).
- [26] [Quantum Computing Applications](#), by Ashley Montanaro, 2013 (69 slides).
- [27] [An Introduction to Quantum Computing](#), by Phillip Kaye, Raymond Laflamme, and Michele Mosca, 2017 (284 pages).
- [28] [Quantum Computation and Quantum Information](#), by Michael Nielsen and Isaac Chuang, Cambridge University Press, 2010 (710 pages).
- [29] [A Course in Quantum Computing for the Community College](#), by Michael Locef, 2016 (742 pages).
- [30] [Quantum algorithms for optimizers](#), by Giacomo Nannicini, arXiv, July 2025 (252 pages).
- [31] [Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer](#), by Peter W. Shor, SIAM Journal on Computing, October 1997 (26 pages).
- [32] [On the Power of Quantum Computation](#), by Daniel R. Simon, SIAM Journal on Computing, October 1997 (10 pages) 📄.
- [33] [Rapid solution of problems by quantum computation](#), by David Deutsch and Richard Jozsa, Royal Society Proceedings A, December 1992 📄.
- [34] [Quantum Complexity Theory](#), by Ethan Bernstein and Umesh Vazirani, SIAM Journal on Computing, October 1997 (63 pages) 📄.
- [35] [Quantum measurements and the Abelian Stabilizer Problem](#), by Alexei Yu. Kitaev, arXiv, November 1995 (22 pages).
- [36] [Efficient Quantum Algorithms for Shifted Quadratic Character Problems](#), by Wim van Dam and Sean Hallgren, arXiv, January 2001 (15 pages).
- [37] [Quantum algorithms for solvable groups](#), by John Watrous, arXiv, January 2001 (13 pages).
- [38] [Polynomial-Time Quantum Algorithms for Pell's Equation and the Principal Ideal Problem](#), by Sean Hallgren, 2006 (21 pages).
- [39] [A Fast Quantum Mechanical Algorithm for Database Search](#), by Lov K. Grover, Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing (STOC '96), May 1996 (8 pages).
- [40] [Quantum Speedups for Exponential-Time Dynamic Programming Algorithms](#), by Andris Ambainis, Kaspars Balodis, Jānis Iraids, Martins Kokainis, Krišjānis Prūsis, and Jevgēnijs Vihrovs, arXiv, July 2018 (17 pages).
- [41] [Quantum computing \(QC\) Overview](#), by Sunil Dixit, PHM Society 2018 presentation, Northrop Grumman, September 2018 (94 slides).
- [42] [Topological data analysis on noisy quantum computers](#), by Ismail Yunus Akhalwaya, Lior Horeh, et al., arXiv, March 2024 (34 pages).

This **Algorithms classes** subsection contains 42 bibliographical references and notes containing at least 3,071 pages.

1.2 Basic algorithms toolbox

We will describe here the overall structure of basic low-level gates-based quantum algorithms. We separate three stages: **data preparation**, **unitary transformations** (caveat: data preparation also relies on unitaries) and **measurement**. We have already covered **error correction** in a previous section of this book.

1.2.1 Data encoding

The data encoding stage is also named data loading or data preparation. Its complexity ranges from uniform-superposition preparation in oracle algorithms such as Deutsch-Jozsa, Simon and Grover to costly arbitrary-state preparation in most quantum linear-algebra and quantum machine-learning algorithms. A quantum processor stores its working quantum state in qubits while gates transform that state and measurements extract classical information.

As illustrated in Figure 15, we can distinguish several high-level key scenarios describing where the problem data is encoded in a quantum circuit:

- **Oracle based algorithms** which start with a uniform superposition of some of the register’s qubits and use an oracle that is either entirely implemented with quantum gates or accesses data through an explicit quantum data-access model. A qRAM is one such model, but no large-scale fault-tolerant qRAM has been demonstrated [43].
- **Hamiltonian encoding into a unitary.** In Hamiltonian simulation and quantum phase estimation (QPE), a Hamiltonian H is represented through a unitary evolution such as $U = e^{-iHt}$, or through a block-encoding from which such transformations can be implemented. QPE then applies controlled powers of this unitary using U^{2^k} , with $k = 0$ to $n - 1$, n being the number of qubits in the phase register.
- **Ansatz encoding.** This is the process of choosing and parameterizing the circuit used in a variational algorithm such as VQE or QAOA. Hardware-efficient ansatzes often alternate parameterized single-qubit rotations and entangling gates like a CNOT. Other ansatz families use problem-inspired excitation operators, symmetry-preserving gates or different entanglers.
- **Quantum sensing data** which is sent directly as qubits in superposed states, if not entangled states in some cases, to the quantum computer [44–46]. This is called “quantum computational sensing”. It has been experimented at the MIT in 2026 for oscillating magnetic field sensing with amplitude and frequency classification [47]. It could also be used in astrophysics imaging [48].

Data loading may be a long process for large sets of inputs and a significant number of qubits.

Quantum data access may require a dedicated memory or data-loading architecture in algorithms whose input is stored externally. qRAM is one proposed architecture. Hamiltonian simulation and QPE do not generically require qRAM, since structured Hamiltonians can instead be compiled or accessed through sparse, LCU, QROM or block-encoding constructions. When a separate quantum memory is used, its state can be transferred to processing qubits through SWAP networks, coherent state-transfer protocols or teleportation. This is a quantum state-transfer (QST) operation rather than a copying operation because an unknown quantum state cannot be cloned. Architectures can also operate directly on memory qubits, so a separate transfer stage is not mandatory.

Let’s first look at the various techniques used for simple data encoding [49–54] (Figure 15).

Uniform superpositions correspond to the simplest qubits register initialization with a register state where all the computational basis states have the exact same amplitude. It is used by oracle-based algorithms where the “real data” sits in the oracle function $f(x)$ that outputs such and such values depending on the entry (usually, a 1 for a single entry and 0 for all the others). The oracle can evaluate this function simultaneously for all superposed computational basis values in the prepared superposed register. This superposition is done by applying Hadamard gates on all computing qubits where some data must be prepared. This preparation is also known as Hadamard-Walsh Transform (HWT), with the following effect on n qubits initialized at $|0\rangle$ the result being an equal superposition of the states $|k\rangle$, k being all the possible values of the computational basis states of these n qubits:

$$H^{\otimes n} |0\rangle^{\otimes n} = (H|0\rangle)^{\otimes n} = \frac{1}{\sqrt{2^n}} \sum_{k=0}^{2^n-1} |k\rangle$$

Figure 14: Uniform superposition encoding. It creates equal amplitudes in the qubit register state vector. This is used for oracle-based algorithms like Grover. © Olivier Ezratty, 2021-2025.

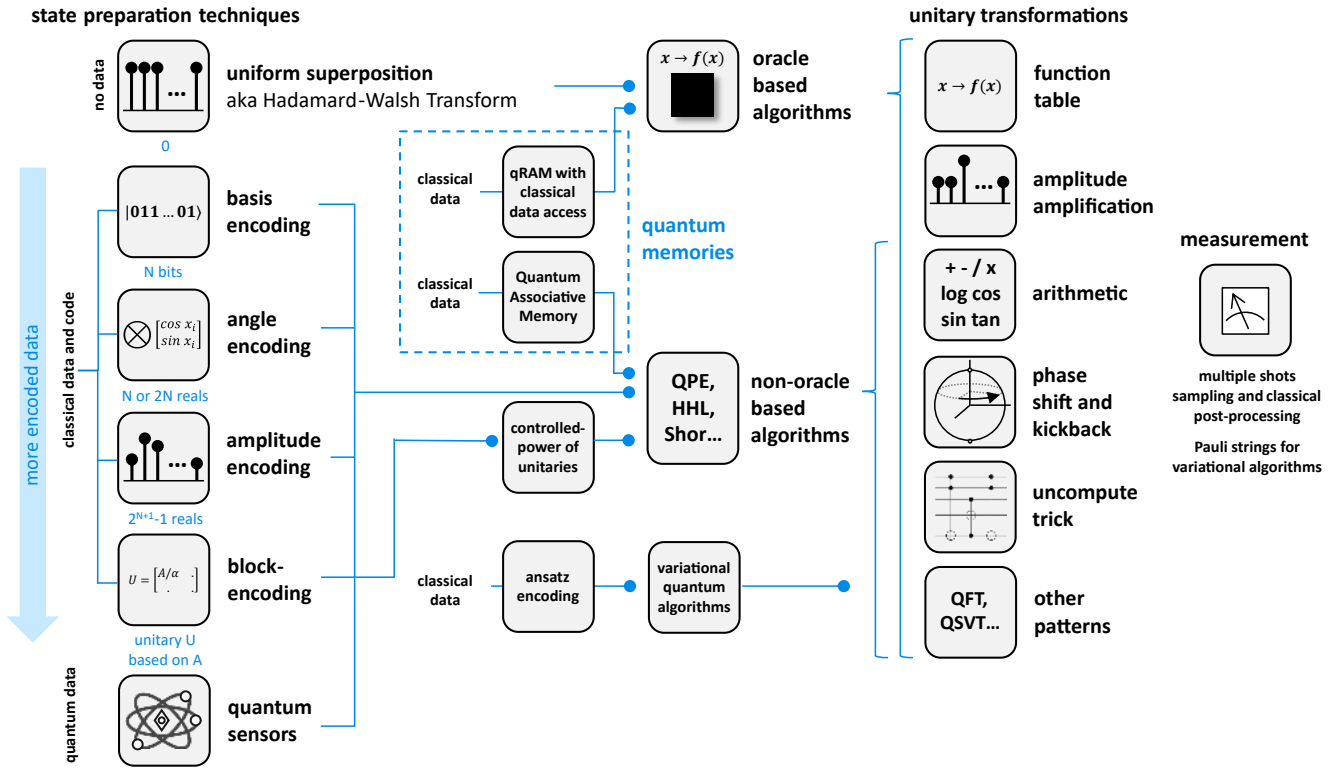


Figure 15: How data is fed into a quantum algorithm, showcasing several scenarios: oracle-based algorithms, non-oracle algorithms using state preparation, Hamiltonian encoding or block-encoding followed by controlled unitary transformations such as those used by QPE, and ansatz preparation and tuning in variational algorithms such as VQE or QAOA. It also includes quantum data fed directly by quantum sensors, which can avoid classical-to-quantum loading in some QML settings [46]. © Olivier Ezratty, 2021-September 2026.

Basis encoding transfers N classical bits into N qubits by applying X gates to the qubits that must change from $|0\rangle$ to $|1\rangle$. It prepares one computational-basis state matching the classical bit string. The 2^N -dimensional state vector therefore contains a single nonzero amplitude, up to an overall phase.

Before this encoding, the classical quantity must be represented as a bit string, for example as a fixed-width integer or fixed/floating-point representation. In Shor’s factoring algorithm, the integer N to be factored is a classical parameter compiled into the exponentiated modular-arithmetic circuit rather than a quantum input register that must be amplitude-loaded.

Angle encoding encodes a real vector of dimension N into N qubits and is also called product encoding. Each datum controls a single-qubit rotation such as R_x , R_y or R_z . On present hardware these rotations may be native or compiled into the processor’s native gates. In a fault-tolerant Clifford+T setting, arbitrary rotations are synthesized to a chosen accuracy. Because the resulting register is a product state unless subsequent entangling gates are applied, the encoding itself provides no exponential compression of independent classical parameters.

A dense angle-encoding variation can use two independent angular parameters per qubit, for example polar and azimuthal angles, giving up to $2N$ real parameters for an N -qubit pure product state, apart from irrele-

vant global phases. The N single-qubit preparations require $O(N)$ gates and can have $O(1)$ circuit depth if they are executed in parallel, or $O(N)$ depth if executed serially. The register remains separable until entangling operations are applied.

Amplitude encoding prepares a normalized quantum state whose amplitudes represent a classical vector. For a vector $x = (x_0, \dots, x_{L-1})$, one prepares $|x\rangle = \sum_j x_j |j\rangle / \|x\|_2$, padding with zeros if necessary. It therefore needs $n = \lceil \log_2 L \rceil$ qubits and satisfies $\sum_j |\alpha_j|^2 = 1$. Three qubits provide eight amplitude slots, not seven. Generic arbitrary state preparation requires $\Theta(2^n)$ elementary-gate resources in the standard circuit model, although depth can be reduced with additional ancillas and structured or approximate target states can be much cheaper [55, 56]. A generic target state may be entangled, but amplitude encoding does not imply entanglement for every possible input vector.

Generic exact preparation of an arbitrary n -qubit state uses $\Theta(2^n)$ gate resources. Consequently, amplitude loading of an unstructured classical vector can erase an algorithmic speedup. Sparse, low-entanglement or otherwise structured data may admit substantially cheaper preparation, which is why the data-access model must be included in end-to-end complexity and resource estimates [55, 57].

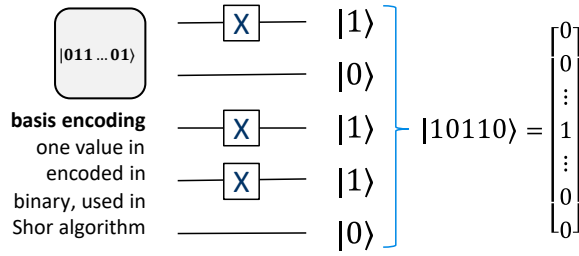


Figure 16: Basis encoding where some qubits are flipped. This is similar to a classical encoding of a number in binary for example. © Olivier Ezratty, 2021-2025.

There are, however, optimized approaches that reduce this cost for structured or approximate target states. One important family is Approximate Amplitude Encoding (AAE), which trains a shallow parameterized quantum circuit to approximate a target amplitude-encoded state [58–61]. Recent frameworks quantify the trade-off between approximation error, circuit resources and target-state entanglement, explaining why structured low-entanglement targets can be easier than generic highly entangled ones [62].

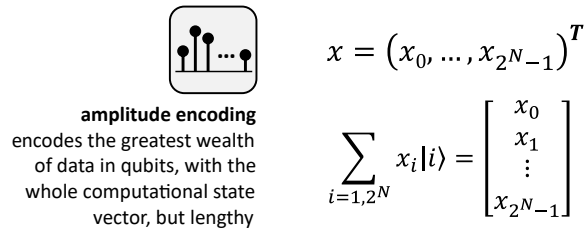


Figure 18: Amplitude encoding creates a normalized state whose computational-basis amplitudes encode a vector. A generic arbitrary target may be entangled and generic exact state preparation has exponential gate complexity in the number of qubits. © Olivier Ezratty, 2021-2026.

A structured variant for high-dimensional distributions is Qvine, whose ansatz mirrors the classical vine copula decomposition into bivariate pair-copulas to stay trainable while keeping circuit depth linear in the distribution dimension for D-vines and at most quadratic for general regular vines [63]. A copula decomposition relies on Sklar’s theorem [64], which factorizes any multivariate distribution into its one-dimensional marginals and a single copula that carries the full dependence structure between the variables. For structured amplitude profiles, exact constructions can be cheaper than the arbitrary case. A degree- d polynomial amplitude vector can be prepared in depth logarithmic in the qubit count n using $\mathcal{O}(n)$ ancillas, via a Pauli-basis block-encoding of an affine diagonal operator combined with a modified LCU (linear combination of unitaries, defined in 1.2.11), improving on earlier linear-depth schemes and demonstrated on a 14-qubit trapped-ion processor [65].

Block-encoding embeds a generally non-unitary matrix A , rescaled by a factor α , into a larger unitary U . It is an efficient representation only when A has

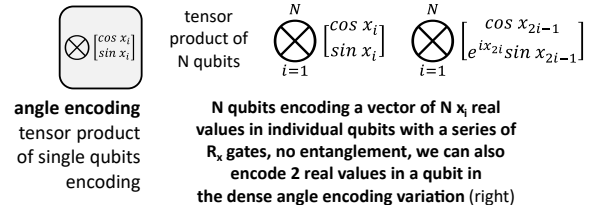


Figure 17: Angle encoding can embed the equivalent of two real numbers for the two angles of the register qubits. These qubits are not entangled. © Olivier Ezratty, 2021-2025.

an efficient access structure, for example sparsity or an efficient oracle, LCU, QROM or arithmetic construction. A generic dense classical matrix is not cheap just by calling the representation a block-encoding [66, 67].

$$U = \begin{bmatrix} A/\alpha & . \\ . & . \end{bmatrix}$$

Once an efficient block-encoding is available, matrix functions can be implemented without explicitly storing every matrix element in qubits. Block-encodings are central to QSVT, QSP-based constructions, qubitization, modern quantum linear-system algorithms and Hamiltonian simulation. QSVT normally consumes a block-encoding and applies a polynomial transformation to its singular values. The cost of constructing the block-encoding itself must remain part of the resource estimate. Techniques also exist for products and polynomial functions of block-encoded matrices [68–72].

Other data encoding methods exist like:

- **Tensor-network based methods** can compress circuits for states with suitable entanglement structure [67, 73–75]. Matrix Product States (MPS) are natural for one-dimensional states and can also approximate selected higher-dimensional or quantum-chemistry states [76, 77]. Tree Tensor Networks (TTN) encode hierarchical correlations and are used in quantum chemistry and strongly correlated systems [78]. MERA (Multiscale Entanglement Renormalization Ansatz) is particularly associated with scale-invariant and quantum-critical states [79].
- **Cosine-Sine Decomposition (CSD)**, a matrix decomposition technique that turns a unitary matrix into a product of simpler unitary diagonal matrices involving cosine and sine diagonal blocks ($\text{diag}(\cos \theta_1, \cos \theta_2, \dots)$ and $\text{diag}(\sin \theta_1, \sin \theta_2, \dots)$). It is useful for decomposing large unitary operations into smaller, structured components [80].
- **Sampling based methods** that consider a target quantum state and treating its squared amplitudes as a probability distribution, then loading that distri-

bution using the classical “alias method” turned into a reversible quantum circuit [81]. A precomputed lookup table in a QROM stores, for each address, a keep threshold and an alias index, and a quantum comparator plus controlled swaps redistribute amplitude coherently to match the target. The benefit is asymptotically lower T-count than a controlled-rotation approach, which is key in FTQC systems where T gates dominate the circuit cost [82]. The QROM used here is a reversible circuit, which converts an address register in superposition into a superposition of the addressed data words [83, 84].

- **Measurement and feedback based methods** which trades circuit depth at the expense of extra ancilla qubits and high-fidelity mid-circuit measurements and classical feedforward steps [85].
- **Walsh-series encoding**, used in quantum signal processing (QSP) and quantum feature maps in quantum machine learning [86].
- **Function/series-based state preparation**, which loads states whose amplitudes follow a smooth analytic function via a series expansion [65, 87].
- **Subspace encoding** which uses variational methods and is relevant for NISQ setups [88].
- **Genetic algorithm based methods** which are adapted to NISQ circuits as well [89].
- **Topology-driven encoding**, which builds the quantum state from the shape of relational data instead of from its coordinates. It remains a proof of concept, being benchmarked only in classical emulation on random graphs with 30 to 50 vertices where a plain edge count already reaches the reported accuracy [90].

The underlying theory on quantum complexity for data loading has interesting ramifications with fundamental quantum physics concepts [91]. Based on this method, various higher abstracted classical data structures can be encoded like images, texts, and time series [92].

Encoding precision. Angle and amplitude encoding use continuous parameters mathematically, but a physical implementation has finite control and gate synthesis precision. On NISQ hardware the relevant limitation is state-preparation error or fidelity, which depends on gate errors, circuit depth and the conditioning of the encoding. A gate infidelity of a few percent cannot be translated directly into a fixed number of decimal digits of the encoded data.

Nonlinearities. Quantum state evolution is linear, but nonlinear classical decision functions can emerge from measurement probabilities, repeated data encoding, tensor-product feature maps and classical post-processing. In data re-uploading, repeated parameterized encodings can generate rich trigonometric or Fourier features of the input rather than a generic rule such as

“one CNOT produces x^2 ”. The accessible polynomial or Fourier degree depends on the encoding and circuit architecture [93–95]. Nonlinear preprocessing can also be performed classically before state preparation. Trainable upload frequencies can in some constructions be replaced by fixed encodings while retaining universal approximation, with a depth/accuracy trade-off [96, 97]. It can also be implemented as a classical Boolean nonlinear circuit embedded in a quantum reversible circuit.

How many registers? A quantum processor exposes a set of physical or logical qubits that an algorithm partitions into logical registers according to function, not to be confused with the notion of logical qubits related to quantum error correction. A circuit can therefore contain several simultaneous registers, for example data, address, phase, work and ancilla registers. “Register” is a logical abstraction rather than a restriction to one monolithic hardware register. Quantum error correction introduces additional physical ancillas for syndrome extraction beyond these algorithm-level logical qubits.

1.2.2 Quantum State Transfer

The no-cloning theorem prevents the copying of an unknown quantum state from one qubit or register to another. It does not prevent moving that state. Quantum state transfer is useful in modular architectures, quantum memories and quantum simulation, including the transport of excitations and entanglement across a lattice or communication link.

Quantum State Transfer (QST) protocols transfer an arbitrary quantum state $|\psi\rangle$ from a sender subsystem to a receiver subsystem while ideally preserving its coherence, phase relations and entanglement with external systems. Perfect preservation is the ideal goal. Real implementations are characterized by a transfer fidelity below one. QST can be implemented with quantum circuits using sequences of SWAP gates or the quantum teleportation protocol. At the physical level, it can exploit natural Hamiltonian evolution to move states or use phonons, photons or cavity modes to mediate transfer. Various protocols and circuits have been proposed to implement QST [98, 99].

The quality of QST is measured by its fidelity, meaning how closely the received state matches the target state. Perfect State Transfer (PST) relates to unit fidelity from the ideal model [100, 101]. Practical protocols trade fidelity against time, control complexity and gate or interaction resources [102]. For short-range, bounded-interaction lattice Hamiltonians, Lieb-Robinson bounds define an effective causal velocity for the propagation of information. This is not a universal hard velocity bound for arbitrary long-range interaction models.

Benchmarks have been proposed to evaluate QST quality [103]. Existing NISQ processors remain challenging. In [104], a four-qubit IBM implementation reports about 72.5% fidelity and 78.1% after Bayesian optimization

Q Hamiltonian encoding

The Hamiltonian H is a Hermitian operator representing the total energy of a system. It governs the time evolution of a quantum state via the Schrödinger equation:

$$i\hbar \frac{d}{dt} |\psi(t)\rangle = H |\psi(t)\rangle$$

In quantum computing, we often set $\hbar = 1$, simplifying the time evolution to this, e^{-iHt} is the unitary time evolution operator: $|\psi(t)\rangle = e^{-iHt} |\psi(0)\rangle$. Hamiltonians are constructed based on the physical interactions in a system. For example with spin systems:

$$H = \sum_{i,j} J_{ij} \sigma_i^z \sigma_j^z + \sum_i h_i \sigma_i^x$$

And with fermionic systems:

$$H = \sum_{i,j} t_{ij} a_i^\dagger a_j + \text{interaction terms}$$

To “simulate” a Hamiltonian on a quantum computer, it is decomposed into Pauli operators:

$$H = \sum_k \alpha_k P_k$$

Where: $\alpha_k \in \mathbb{R}$ are coefficients and P_k are tensor products of Pauli matrices I, X, Y, Z . This decomposition enables implementation using quantum gates. In quantum algorithms such as QAOA, the problem Hamiltonian encodes the computational task. For example, in the MaxCut problem:

$$H_{\text{problem}} = \sum_{(i,j) \in E} \frac{1}{2} (1 - Z_i Z_j)$$

For the MaxCut cost operator written above, larger eigenvalues correspond to larger cuts, so the optimal cut is a *maximum*-eigenvalue state. If one defines the problem Hamiltonian with the opposite sign, $-H_{\text{problem}}$, the optimal solution becomes a ground state. To implement the unitary evolution e^{-iHt} , we use techniques such as Trotter-Suzuki decomposition:

$$e^{-iHt} \approx \left(\prod_k e^{-i\alpha_k P_k t/r} \right)^r$$

Each term $e^{-i\alpha_k P_k t}$ can be implemented using a sequence of quantum gates, typically involving basis rotations, controlled operations and single-qubit and two-qubit gates. This gate-level decomposition allows quantum processors to simulate complex Hamiltonians using their native gate sets.

of coupling parameters. Bayesian optimization here is best described as classical parameter or control optimization rather than, by itself, a general quantum-error-mitigation protocol.

Another technique uses QST to transfer an arbitrary state generated by some unitary, then apply the reverse unitary and transfer the state back to its origin qubits. This “do nothing” approach can help characterize spatial noise in a quantum processor [105] (Figure 19).

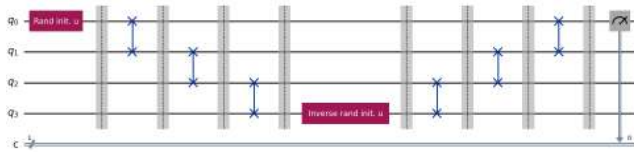


Figure 19: The do-nothing protocol creates an arbitrary state, transfers it through SWAP operations, applies the reverse preparation unitary and transfers it back. Comparing the final state with the expected reference characterizes spatially dependent noise and can help identify better routes through the processor connectivity graph. Source: [105].

1.2.3 Black boxes and oracles

A black-box quantum algorithm queries a function through a unitary oracle, often of the form $U_f|x\rangle|y\rangle = |x\rangle|y \oplus f(x)\rangle$. The input register may be in a superposition, which lets one oracle call act coherently on all basis components. Deutsch-Jozsa, Simon and Grover are the usual examples. A Boolean oracle has a one-bit function value, but the term oracle is not restricted to the case of a single output qubit [106].

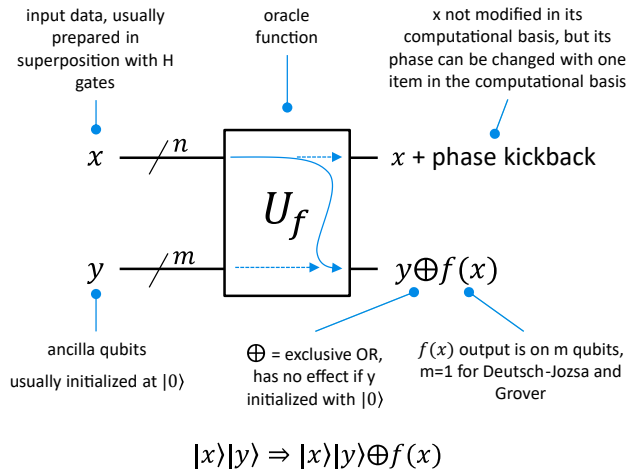


Figure 20: How an oracle function is used in an algorithm, in complement of a phase kickback. © Olivier Ezratty, 2021.

An oracle can be compiled entirely using reversible quantum gates, or it can rely on a specified data-access primitive. qRAM is one proposed primitive for coherent memory access, but no small or large-scale fault-tolerant implementation exists. For a specified function and reversible circuit, the oracle’s gate cost can be estimated. What the query-complexity model does is abstract that internal cost away. Arithmetic-heavy oracles can be expensive even when the number of oracle queries is small [43, 107].

Oracle-based speedups must therefore be interpreted together with the cost of realizing the oracle. Grover’s quadratic query advantage, for example, can be overwhelmed by state preparation, oracle evaluation, routing and fault-tolerance overheads. The threshold theorem gives polylogarithmic overhead in asymptotic idealized models for fixed logical accuracy, but realistic physical-qubit and spacetime overheads can still be very large. Parallelizing an oracle over several data ranges is one possible space-time trade-off [108].

In oracle parlance, an **oracle expansion** considers the oracle quantum circuit to estimate the total algorithm resources like gate cycle, number of Clifford and non-Clifford group gates and the like [54].

Another distinction is between unstructured and promise oracles. In unstructured Grover search, the oracle marks candidate solutions but provides no metric telling how close an incorrect candidate is. Simon’s algorithm instead assumes a global promise. For a nonzero hidden string s , $f(x) = f(y)$ if and only if $y = x$ or $y = x \oplus s$. The quantum algorithm exploits this structure to recover s with exponentially fewer oracle queries than a classical randomized algorithm.

1.2.4 Output encoding

Literature on quantum algorithms rarely describes algorithm output formats. A measurement of an n -qubit register in the computational basis returns one classical n -bit sample, not a full characterization of the underlying

ing quantum state. In algorithms such as Grover, one shot can return the marked item with high probability after the appropriate number of iterations, while repetitions increase confidence and diagnose hardware errors. Algorithms such as HHL instead return a quantum state whose amplitudes encode the solution vector, so their usefulness normally relies on estimating selected observables or passing the state to another quantum subroutine.

Reading every component of an amplitude-encoded output generally destroys the compact representation advantage because the classical output itself contains exponentially many parameters. Full state tomography is even more demanding because phases as well as probabilities must be inferred. Amplitude-encoded outputs are therefore most useful when consumed by another quantum routine or when only aggregate quantities are required [109]. In Shor’s factoring algorithm, repeated measurements of the phase-estimation register are classically post-processed, typically using the continued-fractions classical technique, to infer the order and then obtain a non-trivial factor by a greatest-common-divisor computation.

In VQE, the problem Hamiltonian energy is normally obtained from expectation values of Pauli terms measured after appropriate basis rotations. These are usually projective measurements rather than requiring a generic POVM. Chemistry algorithms often return expectation values estimated by averaging many circuit shots, and measurement cost can become a dominant resource [110, 111]. Its overhead requires significant optimizations [111]. Generically, NISQ QML algorithms also output sampled values and not a single computational base value [112].

1.2.5 Quantum phase kickback

A phase oracle marks selected basis states by changing their relative phase. A conventional bit oracle can be converted into a phase oracle by preparing the target register in an eigenstate such as $|-\rangle$, so that the oracle eigenphase is kicked back onto the input register. Grover’s algorithm uses this phase marking before the diffusion or amplitude-amplification step.

More generally, the target register is an eigenstate $|\psi\rangle$ of a unitary U , with $U|\psi\rangle = e^{i\phi}|\psi\rangle$. If a control qubit is in a coherent superposition, a controlled- U multiplies only its $|1\rangle$ branch by $e^{i\phi}$. The target state is unchanged and the eigenphase becomes a relative phase of the control. Hadamard gates are one common way to create and later interrogate that superposition, but they are not part of the mathematical definition of kickback [113].

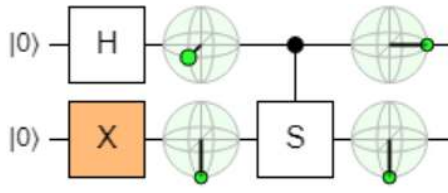


Figure 21: A two-qubit phase kickback shown using the Quirk emulator.

The two-qubit example in Figure 21 illustrates this mechanism. The key point is not that a controlled-phase gate moves a global phase from one qubit to another. Rather, when the target is an eigenstate of the controlled unitary, the joint state remains separable and the corresponding eigenvalue appears as a relative phase on the control branch [114, 115]. Because the target is an eigenstate, controlled- U does not entangle the control and target in the ideal kickback construction. The physically observable information is the relative phase between control branches. A global phase of the full two-qubit state would be unobservable.

In a general case, if $U|\psi\rangle = e^{i\phi}|\psi\rangle$ and the control begins in $a|0\rangle + b|1\rangle$, then controlled- U produces $(a|0\rangle + be^{i\phi}|1\rangle) \otimes |\psi\rangle$. The target remains unchanged while the control carries the eigenphase.

1.2.6 Quantum arithmetic

Arithmetic functions can be implemented in a quantum circuit [116, 117]. It is frequently used in oracle functions but also in the period finding function found in Shor's integer factoring algorithms. Many quantum algorithms exist to implement various arithmetic functions: adders [118, 119], multipliers [120–122], dividers, floating point calculus [123–125], transcendental functions (exponential, logarithm, and trigonometric functions) [126, 127] and matrix operations [128, 129].

Quantum reversible adders and multipliers can be derived from classical reversible logic, for example ripple-carry constructions. X implements NOT and CNOT implements XOR when one input is preserved. A reversible AND generally requires a Toffoli or an equivalent decomposition, not X and CNOT alone. Early adders such as [130] used substantial workspace, while later constructions reduce or eliminate clean ancillas [131]. There is a standard space-time trade-off between ancilla count, circuit depth and non-Clifford gate count [132].

Alternatives consist in using a QFT and inverse QFT which reduce the need for ancilla qubits [133], including the variant aQFT (approximate QFT) which reduces the gate count. A similar process is proposed to compute matrix multiplications [134, 135]. Another approach is using a matrix multiplication formalism and a QSP [136].

Adder and multiplier modules optimizations are proposed from time to time to reduce their overhead [137,

138], their ancillary [139], T gate numbers [140–142] or Toffoli gate count [143].

Quantum arithmetic logic unit (QALU) proposals also target shallow or hardware-efficient arithmetic circuits, including Clifford and CSX ($\sqrt{X}$) constructions [144]. Their usefulness on NISQ hardware remains architecture- and error-dependent rather than established generically.

Shor's integer factoring algorithm uses extensive quantum arithmetic in its order-finding component, particularly in the modular-exponentiation circuit before the inverse QFT, as shown in Figure 23. It is sufficiently compute-intensive that a team in Japan proposed to create special units to run the related multipliers, using fluxonium qubits, without however indicating how Shor's algorithm would be implemented in its entirety in such a system [145].

1.2.7 Uncompute trick

The compute-copy-uncompute pattern is associated with Charles Bennett's 1973 construction of reversible computation, which computes a result, preserves the desired classical output and reverses the computation to erase intermediate garbage reversibly [146]. Bennett's 1989 work later developed reversible time-space trade-offs.

It is used to rewind some parts of an algorithm affecting ancilla or input qubits. It cleans up the state of a qubits register without requiring a qubit reset that may damage the stored values in the qubits with the algorithm results. It is also used to disentangle the ancilla qubits from the input qubits (Figure 22). It then makes it possible to go on using these ancilla qubits for the remainder of the algorithm.

The transformation works if the unitary U_f is a reversible circuit, as is true of any unitary gate sequence without intermediate measurement.

The uncompute trick is especially common in oracle and arithmetic circuits. It does not mathematically presume error-free qubits. On noisy hardware, however, executing the inverse circuit adds further noisy gates, so the ancillas are returned only approximately to $|0\rangle$. This is a practical fidelity limitation, not a failure of the reversible-computation principle.

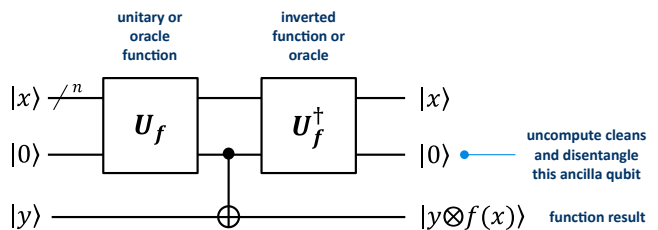


Figure 22: The uncompute trick algorithm cleans up a register and its ancilla qubits with disentangling them from the data qubits while preserving the function result from the computed algorithm. It makes the ancilla qubits available for subsequent computation. It leaves x and $f(x)$ intact. © Olivier Ezratty with various sources. 2022.

algorithm	classical input	quantum input blue for superposed state	quantum output blue for superposed state	classical output	speedup (and number of circuit runs)
Deutsch-Jozsa	constant or balanced Boolean function, hidden in an oracle [33]		all n query qubits measured at $ 0\rangle$ if f is constant, any other bitstring if f is balanced [33, 147]	one bit, “yes or no”	exponential against deterministic classical (1 query against $2^{n-1} + 1$), none against randomized [147] (1 run)
Bernstein-Vazirani	hidden bitstring s carried by $f(x) = s \cdot x \bmod 2$ [34]	oracle function U_f , implemented as a reversible circuit or through an explicit coherent data-access model such as qRAM. Large-scale fault-tolerant qRAM has not been demonstrated [43, 49, 62]	secret string s recovered directly, in basis encoding [34]	integer s over n bits	linear, 1 query against n classical queries, not exponential [34] (1 run)
Grover	oracle marking $M \geq 1$ solutions among $N = 2^n$ entries [39, 106]		marked item index in basis encoding, after $\lceil \frac{\pi}{4} \sqrt{N/M} \rceil$ iterations [39]	integer index of the marked item	quadratic and proven optimal, $O(\sqrt{N})$ oracle calls inside one circuit [148] (a few runs)
Simon	2-to-1 function with a hidden XOR mask s , $f(x) = f(x \oplus s)$ [32]		a random bitstring y orthogonal to s , with $y \cdot s = 0 \bmod 2$, in basis encoding [32]	mask s , from a Gaussian elimination over $\text{GF}(2)$ on the collected y	exponential, $O(n)$ queries against $\Omega(2^{n/2})$ classically [32] ($O(n)$ runs, not one)
Shor factoring	composite integer N , a semi-prime in the RSA case [31]	Hadamard prepared exponent register, modular exponentiation, then an inverse QFT [150]		period r from continued fractions, then a non-trivial factor by gcd	superpolynomial over classical GNFS [149]
Shor dlog	generator g , target h and order r of a finite cyclic group [31]		amplitude peaks regularly spaced by $2^m/r$, starting at 0	discrete logarithm d , from the post-processing of the measured pairs	exponential, and about one run at a 60 to 82% success rate [151]
QFT	vector of 2^n values [152]	complex amplitudes loaded by state preparation, in amplitude encoding [49]	Fourier coefficients in amplitude encoding, used by others stages of an algorithm [153]	dominant frequencies only, $O(1/\epsilon^2)$ shots per amplitude. Recovering all $N = 2^n$ coefficients not efficient.	uses $O(n^2)$ gates, or less in some variants, vs an FFT. State preparation and classical readout can dominate [154, 155].
QPE	unitary $U = e^{-iHt}$ and a classically prepared approximate eigenstate [35]	controlled applications of powers of U with U^{2^k} applied to the eigenstate register [147]	phase-register distribution concentrated around the n -bit estimate of θ after the inverse QFT	phase as a real number over n bits, then the eigenvalue or energy [156]	precision ϵ at $O(1/\epsilon)$ controlled- U calls [157], several runs to improve output confidence.
HHL	sparse Hermitian matrix A and vector b [158]	efficient access to A through sparse/oracle, Hamiltonian-simulation or modern block-encoding constructions, and preparation of $ b\rangle$ [53]	$ x\rangle \propto A^{-1} b\rangle$, normalized and represented in amplitudes	expectation value, overlap or other property. Full classical recovery of all N solution components not efficient [159].	potential exponential-in-log N separation with favorable sparsity or structure, condition number, and compact output [159, 160]
QAOA	objective function recast as a diagonal Ising Hamiltonian [161]	parameters loaded in an ansatz of rotation gates and CNOTs, updated at each classical iteration [163]	superposition biased toward low cost solutions, read out as computational basis bitstrings [161]	best cost value found and updated angles β and γ	no general proven quantum speedup. [161]
VQE	problem Hamiltonian as a weighted sum of Pauli strings [162]		superposed ansatz state, sampled term by term after a basis rotation into $\langle P_i \rangle$ [111]	energy estimate, updated parameters and approximate ground state	no general proven speedup. Electronic-structure Hamiltonians can contain $O(n^4)$ Pauli terms before exploiting structure. [163]
QML classification	training set or single object vector, plus a model [50]	feature vector in amplitude or angle encoding, through a parameterized feature map [51, 164]	class label as an index in basis encoding, from a majority vote over shots [50]	integer identifying the class in a reference table	not proven, and dominated by the data loading cost [112]

Table 2: Main quantum algorithms input and output formats, with classical data fed in the quantum circuit to the read out classical result, with quantum intermediates. Blue marks registers containing a superposed or amplitude encoded state, which cannot be read out directly and must be sampled or consumed by another quantum circuit. Red marks unproven speedups. The last column separates the asymptotic speedup from the number of circuit runs needed to obtain a usable classical result. The FTQC block itself mixes two regimes, query algorithms using oracles (Deutsch-Jozsa to Simon) and primitives leaving their result in amplitudes (Shor to HHL). © Olivier Ezratty, 2021–September 2026.

Q What is a circuit sampling?

In quantum computing, sampling is the process of running a quantum circuit multiple times and measuring the final state of the qubits to obtain classical bitstrings from an underlying probability distribution.

Because one measurement returns only one sample, repeated shots are most of the time needed to estimate probabilities, expectation values or a most-likely result with a chosen statistical confidence. Reconstructing an entire 2^n -outcome distribution can itself require exponentially many samples in the worst case, so many algorithms are designed to extract relevant statistics rather than reconstruct the full state or distribution.

On NISQ systems, additional shots reduce statistical uncertainty. Variational algorithms typically use repeated measurements to estimate expectation values over multiple Pauli string basis. And hardware errors may require error-mitigation techniques which adds another circuit shots overhead.

In FTQC, a simple Grover search algorithm amplifies the probability of target basis state, but repeated runs may still be used to increase the probability of observing the given state. In quantum phase estimation algorithms, repeated runs can improve confidence when the output is probabilistic.

Good NISQ and FTQC papers indicate how many times a circuit must be executed on top of its size and composition in non-Clifford gates.

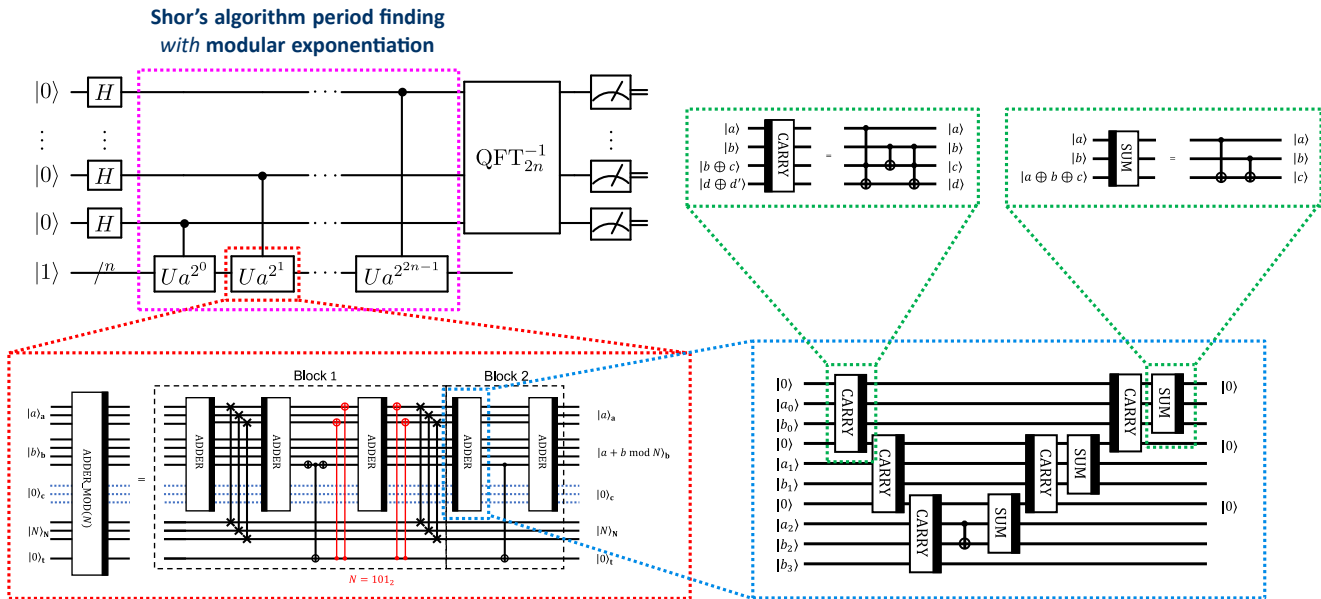


Figure 23: Shor's algorithm period finding decomposition. It uses many ADDER_MOD arithmetic modules which themselves use ADDER modules comprising CARRY and SUM modules. Modules schematics from [150]. Added in 2023.

1.2.8 Quantum Fourier Transform

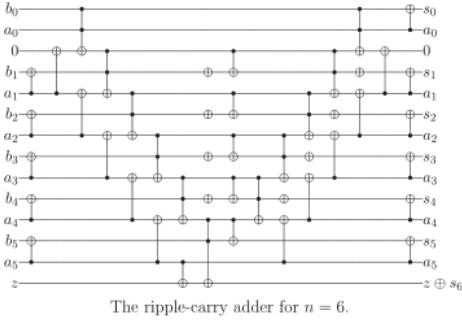
Classical Fourier transforms are used to decompose a signal into its compound frequencies. For example, it is used to identify the basic components of some sound by breaking it down into its frequency components. In astrophysics, the atomic composition of stars is determined by a decomposition of its light spectrum obtained with telescopes. This can be done by using an optical prism for optical telescopes. The same is true for Scio-type near-infrared sensors that determine the composition of food. A prism and the principle of diffraction therefore allow an optical Fourier transform to be performed.

Efficient QFT circuits for power-of-two dimensions emerged in the early quantum-algorithm literature,

and Coppersmith's 1994 manuscript gave an influential approximate-QFT construction [152]. In Shor's algorithms, the inverse QFT converts phase or periodicity information into a distribution of measurable integers from which the order is inferred classically. It does not directly reconstruct a divisor. A QFT is the unitary analogue of a discrete Fourier transform on quantum amplitudes. Any comparison with a classical FFT must include the costs of preparing the input quantum state and extracting the desired classical information from the output.

QFT or inverse-QFT subroutines appear in phase estimation, Shor's factoring and discrete-logarithm algorithms, several arithmetic constructions and some linear-system or signal-processing methods.

arithmetic operations can be useful in many algorithms
quantum reversible adders/multipliers can be derived from
their classical counterparts (ripple-carry adders) or use a QFT
and IQFT (inverse QFT) to reduce the need for ancilla qubits



6-bit ripple-carry addition adder

Adder type	Toffoli/T depth	Toffoli/T gates	Qubits required
Majority ripple [12]	$2n$	$2n$	$2n + 1$
Prefix-ripple [Section A.1]	n	$3n$	$2n + 1$
Carry look-ahead [13]	$4\log_2(n)$	$10n$	$4n - \log_2(n) - 1$
Fourier transform basis [33]	$3\log_2(1/\epsilon)$	$3n\log_2(1/\epsilon)$	n

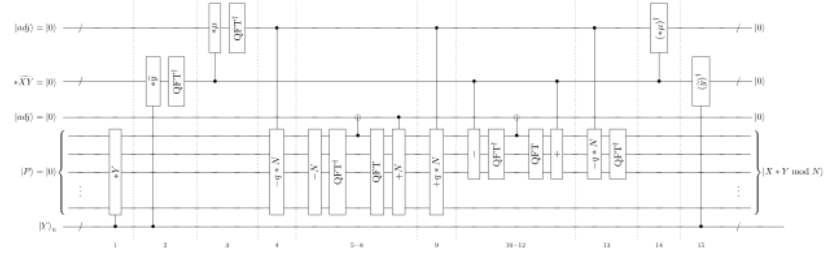


Figure 12: Barrett multiplication circuit using Fourier arithmetic. The numbers in the figure correspond to the steps of Algorithm 2.

Barrett multiplication circuit using QFT

Figure 24: Various arithmetic computing can be implemented with quantum algorithms, mostly using a QFT. Sources: [118, 120].

QFT mathematics. For an n -qubit state, let $N = 2^n$ and write $|\psi\rangle = \sum_{j=0}^{N-1} \alpha_j |j\rangle$. The QFT applies the discrete Fourier transform to the full vector of computational-basis amplitudes $\{\alpha_j\}$, not separately to one value stored in each qubit [153].

The QFT implements a Discrete Fourier Transform (DFT) on these discrete amplitudes and converts it into a new computational state vector with amplitudes being the result of the QFT. The initial vector state can be written as in the formula on the right.

$$|\psi\rangle = \sum_{j=0}^{N-1} \alpha_j |j\rangle$$

The QFT creates a new state vector $QFT_N(|\psi\rangle)$ with N β_k amplitudes of the N computational basis vectors $|k\rangle$. It is a formula similar to the above starting point.

$$QFT_N(|\psi\rangle) = \sum_{k=0}^{N-1} \beta_k |k\rangle$$

The amplitudes β_k are computed with a big sum using all the amplitudes α_j with the coefficient ω^{jk} which is a N^{th} root of 1, meaning that $\omega^{jN} = 1$.

$$\beta_k = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} \omega^{jk} \alpha_j$$

These coefficients $\omega^{jk} = e^{\frac{-2\pi i}{N}jk}$ explain the heavy use of R_n phase rotation gates in the QFT algorithm as described on the right. You remove the minus sign to obtain a reverse QFT.

$$\beta_k = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} e^{-2\pi i \frac{jk}{N}} \alpha_j$$

In the end, the QFT_N is a unitary matrix transformation $[QFT_N]$ with simple coefficients $[QFT_N]_{jk}$, as in a DFT.

$$[QFT_N]_{jk} = \frac{1}{\sqrt{N}} \omega^{jk}$$

When $n = 1$ and $N = 2$, the QFT becomes a Hadamard gate transform. The QFT is indeed presented as a generalization of the Hadamard operation, applied to dimensions $N > 2$.

Since preparing such an arbitrary vector could take an exponential time with regards to the number of qubits, it is usually done through some faster preparation mechanism like in Shor's algorithm.

What are we really getting out of a QFT? Consider $n = 4$ qubits, hence $N = 16$ amplitudes, with a phase that advances by $45^\circ = 2\pi/8$ from one basis index to the next. Across 16 indices the phase completes two cycles, so an ideal QFT concentrates the state on Fourier index $k = 2$ for one sign convention, or the conjugate index for the opposite convention. Which physical qubit reads $|1\rangle$ depends on the final SWAP convention, so the frequency index is an unambiguous statement.

Recovering every complex Fourier coefficient from a QFT output would require extracting $N = 2^n$ classical numbers, with additional measurements needed to reconstruct phases, and therefore annihilating the compact quantum representation. QFTs are consequently valuable mainly as subroutines whose output remains quantum and is used by the remainder of the circuit, or when only a compact property can be estimated efficiently.

The exact n -qubit QFT uses Hadamard gates and controlled phase rotations whose smallest angles scale exponentially with n . Small rotations can be expensive to synthesize fault-tolerantly with high accuracy, but approximate QFTs omit sufficiently small rotations and substantially reduce this cost. The relevant resource is the number and synthesis precision of rotations, not an intrinsic inability to control exponentially small physical pulse angles on every architecture.

QFT circuit complexity. For $N = 2^n$, an exact textbook QFT uses $O(n^2) = O((\log N)^2)$ one- and two-qubit gates, while the classical FFT on an explicit length- N vector uses $O(N \log N)$ arithmetic operations. Approximate QFT constructions can reduce the quantum gate count further for a specified error tolerance.

The QFT on n qubits acts in a Hilbert space of dimension $N = 2^n$. The exact textbook circuit uses $O(n^2)$ gates. Truncating sufficiently small controlled rotations can reduce an approximate QFT to roughly $O(n \log(n/\epsilon))$ gates for common error criteria, with the precise scaling depending on the chosen approximation metric and architecture. Extra ancillas can trade space for depth, but do not by themselves define a single universal $O(n \log n)$ construction.

The “speedup” comes from the fact that for $N = 2^n$:

- Classical FFT: $O(N \log N) = O(2^n \cdot n)$
- Quantum QFT: $O(n^2)$ (or $O(n \log n)$)

This gate-count comparison is sometimes called an exponential reduction when expressed in the qubit count n , but it is not by itself an end-to-end quantum speedup over the FFT. The QFT acts on amplitudes already stored in a quantum state and does not output all N Fourier coefficients as classical numbers. Its importance comes from algorithms such as Shor integer factoring, where the Fourier-transformed state is sampled only for the limited information needed by the surrounding algorithm. It was demonstrated in 2026 with some diffusion model based generative AI that, at a small scale of 5 qubits, a QFT was the optimal circuit to analyze a signal encoded with a vector state. But the related work didn’t recreate the repetitive circuit structure of a QFT at scale [165].

QFT implementation. Small QFTs can and have been executed on NISQ hardware. Large, high-precision QFTs inside algorithms such as fault-tolerant phase estimation require logical error rates low enough for the support of deep circuits. In a Clifford+T architecture, arbitrary controlled rotations are synthesized to the required approximation error and can drive high T costs. QEC suppresses logical errors through repeated syndrome extraction. Its physical-qubit and spacetime overhead can be much larger than one order of magnitude [154, 166].

As a result, there’s a need to minimize the number of CNOT and T gates that are used [167], for example using

a so-called “approximate QFT” or aQFT, that removes rotation gates with angles below a certain threshold, thereby optimizing the number of T gates for the aQFT [154]. There are also ways to get rid of most of the required ancillas [168]. This log-depth construction, also named optimistic QFT since it accepts a bounded average error over the Hilbert space, was evaluated in 2026 at the architecture level on a QuEra system, bringing the rotation count down to $O(n \log n)$ but by adding about 44% more rotations than an approximate QFT [169]. It halves the runtime of a serial QFT on 256 to 2048 bit registers but requires about 500 extra logical ancilla qubits and a peak parallelism of 128 concurrent logical qubits, against 32 for the regular QFT.

For very small QFT rotation angles, a fault-tolerant implementation compares the benefit of retaining the rotation with the allowed approximation error and the cost of synthesizing it. If an angle is below the chosen truncation threshold it is simply omitted in an approximate QFT. If it is retained, the synthesis cost is controlled primarily by the target approximation error and gate set rather than by the angle alone. Modern synthesis bounds are therefore a better description than invoking the Solovay-Kitaev theorem as the practical cost model [170].

The QFT circuit can also be classically simulable in restricted regimes, for example when the input and intermediate states retain sufficiently low entanglement or admit compact tensor-network descriptions [155, 171]. This does not imply that arbitrary QFT-based algorithms are efficiently classically simulable, because their surrounding state preparation and entangling computation can invalidate those assumptions.

As a result, restricted QFT circuits can sometimes be simulated efficiently with tensor-network or other classical methods when the state structure permits it. Such results are best viewed as conditional dequantizations rather than a generic classical replacement for the QFT [155, 172].

Semiclassical QFT was introduced by Griffiths and Niu in 1996 [173]. When a QFT is immediately followed by measurement in the computational basis, controlled phase gates between qubits can be replaced by sequential single-qubit measurements and classically feed-forward phase rotations. The transform still processes and measures n output bits, but it eliminates the two-qubit controlled rotations of the final QFT. In iterative phase-estimation or Shor implementations, one can additionally recycle a single control qubit across the n phase bits, trading qubit count for sequential control and classical feedforward [149, 174].

Exact QFT for arbitrary orders. The standard power-of-two QFT circuit is already exact in the ideal gate model. Mosca and Zalka showed how Kitaev’s approximate construction for more general transform orders can be made exact using amplitude amplification, and applied the idea to an exact discrete-logarithm al-

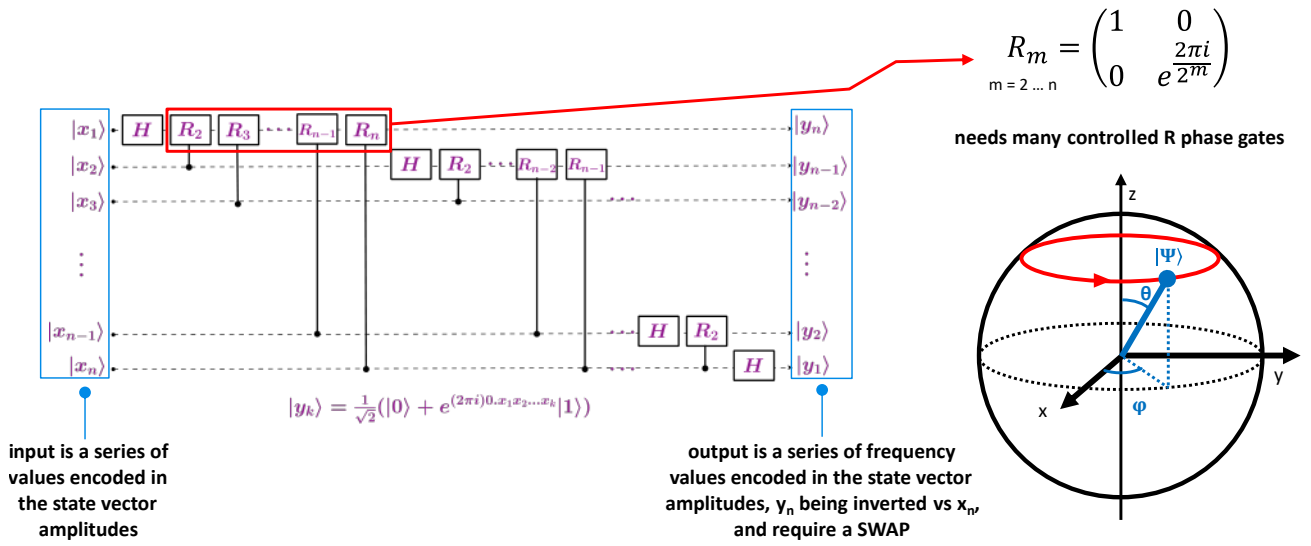


Figure 25: A QFT applies a discrete Fourier transform to the computational-basis amplitudes of a quantum state. Its inverse appears in QPE and Shor’s algorithms, while QFT variants also occur in arithmetic and signal-processing routines. The exact textbook transform uses $O(n^2)$ controlled phase rotations on n qubits. In fault-tolerant implementations, rotation synthesis can be expensive, while approximate QFTs omit sufficiently small rotations. © Olivier Ezratty and various sources. Updated in 2026.

algorithm [175]. The result does not mean that an exact QFT generically produces a single basis-state answer rather than a superposition. A non-uniform QFT was also proposed in 2026 as a quantum analogue of the non-uniform FFT for irregular sampling grids [176].

Non-abelian QFT. The QFT generalizes to finite non-abelian groups, where basis states are transformed into labels associated with irreducible representations and matrix indices rather than ordinary scalar frequencies. Efficient Fourier transforms exist for some groups, including the symmetric group, but strong Fourier sampling did not yield an efficient graph-isomorphism algorithm [177, 178]. The dihedral hidden-subgroup problem is related to hidden-shift problems and lattice cryptography, but Kuperberg’s algorithm remains subexponential rather than polynomial, so it does not provide a generic attack on lattice-based cryptography [179].

QFT implementation records. In the FTQC realm, a record was beaten in 2024 by Quantinuum who implemented a small QFT using 3 logical trapped-ion qubits [180]. In 2026, ParityQC demonstrated a 50-qubit QFT on an IBM Heron r3 processor using a SWAP-free Parity Twine compilation, surpassing the previous 27-qubit ion-trap record, though at a still-low process fidelity of about 10^{-2} [181]. It was followed by a Q-CTRT experiment published in August 2026 with a NISQ QFT reaching 100 qubits, with the use of error suppression, although the process has a fidelity of 1.8% at 80 qubits [182].

A QFT can also be distributed across multiple FTQC systems using quantum interconnect [183].

1.2.9 Amplitude Amplification

Amplitude amplification is a generalization of Grover’s search that increases the total probability amplitude of a designated *good subspace* while decreasing that of its complement. The good subspace can contain one or many basis states. Quantum amplitude estimation builds on this idea to obtain a quadratic improvement in sample or oracle complexity for suitably encoded Monte Carlo estimation tasks, from $O(1/\epsilon^2)$ classical samples to $O(1/\epsilon)$ coherent oracle uses in the ideal model [184, 185].

In 2021, a researcher from the Fermi Lab at the DoE found a way to create an amplitude amplification working on a non-Boolean oracle [186].

Qubitization was introduced by Low and Chuang as a method that embeds a block-encoded Hamiltonian into invariant two-dimensional $SU(2)$ subspaces. Combined with quantum signal processing, it implements e^{-iHt} to error ϵ with optimal query complexity $O(t + \log(1/\epsilon))$ in the normalized projected-unitary oracle model, using only a constant number of additional ancillas [187]. It is a core primitive for modern Hamiltonian simulation and phase-estimation resource estimates, rather than an optimized form of amplitude amplification.

QSVT. Quantum Singular Value Transformation generalizes QSP to block-encoded matrices and implements polynomial transformations of their singular values [188]. It is a powerful unifying framework covering many components in Hamiltonian simulation, linear algebra, amplitude amplification and related algorithms [189]. QSVT itself does not guarantee an exponential end-to-end speedup. Any advantage depends on the polynomial degree, block-encoding cost, condition numbers,

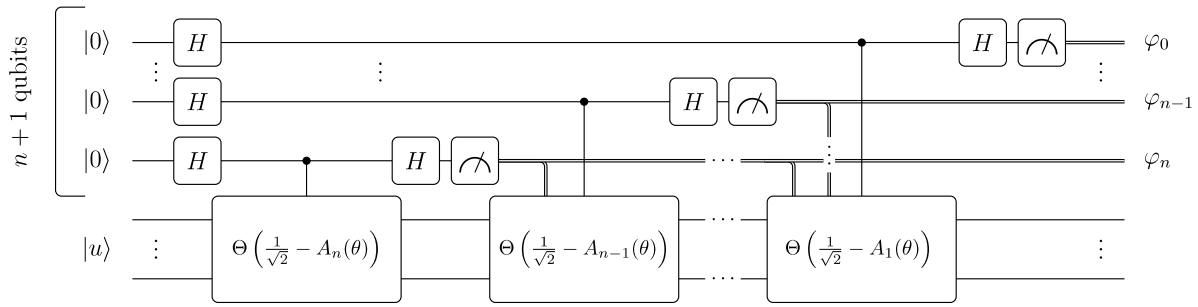


Figure 26: The QSVT algorithm, used here to compute a phase estimation. Source: [189]. Added in 2024.

input-state preparation and what information must be extracted from the output.

1.2.10 Hamiltonian simulations

A Hamiltonian H is a Hermitian observable representing the energy of a closed quantum system and, more fundamentally for simulation, the generator of its time evolution. For a time-independent Hamiltonian, the state evolves according to $|\psi(t)\rangle = e^{-iHt/\hbar}|\psi(0)\rangle$.

Hamiltonian simulation means approximating the action of the time-evolution operator e^{-iHt} , or its time-ordered analogue for a time-dependent Hamiltonian. Estimating ground-state energies, preparing ground states and computing dynamical observables are important applications built on top of this primitive, but they are distinct computational problems.

In quantum chemistry this supports electronic-structure and molecular-dynamics calculations in selected active spaces. Predicting complete three-dimensional protein structures is not a direct consequence of Hamiltonian simulation and generally involves much larger biochemical and thermodynamic modeling problems.

Simulating a Hamiltonian was at the core of Richard Feynman's idea coined in 1981 when he wondered whether a quantum system could simulate another quantum system more efficiently than a classical computer, breaking down the fatal exponential growth of computing resources required to implement this kind of simulation on classical computers.

Such a simulation problem is described by a Hamiltonian which is a Hermitian matrix H of size $2^N \times 2^N$, when working with N qubits or N two-states quantum objects like spin-1/2 particles.

For a time-independent Hamiltonian, the Schrödinger equation and its solution are

$$i\hbar \frac{\partial}{\partial t} |\psi(t)\rangle = H |\psi(t)\rangle, \quad |\psi(t)\rangle = e^{-iHt/\hbar} |\psi(0)\rangle$$

Hamiltonian simulation assumes a description or access model for H and approximates its generated evolution. The *local Hamiltonian problem* is a different complexity-theory problem: given a Hamiltonian written as a sum

of local terms, estimate whether its ground-state energy is below or above promised thresholds. Locality here means that each term acts nontrivially on at most k subsystems. It is a model assumption about the interaction structure, not a consequence of special or general relativity.

A k -local Hamiltonian has the form $H = \sum_j H_j$, where each term H_j acts on at most k of the N qubits. Many physically motivated lattice Hamiltonians are geometrically local, but long-range interactions and nonlocal effective Hamiltonians also occur.

Estimating the ground-state energy of a general 2-local Hamiltonian to inverse-polynomial precision is QMA-complete [190]. QMA-completeness does *not* mean that a quantum computer can efficiently solve every such instance. It means that a proposed low-energy quantum state can be verified efficiently by a quantum verifier, given an appropriate quantum witness. QMA is believed to contain problems harder than BQP.

There are of course many variations of Hamiltonian simulations depending on the type of quantum system to emulate and the characteristics we want to extract from H . It includes hybrid solutions associating classical and gates-based quantum computing including the quantum adiabatic algorithm. QPE is one solution, adapted to FTQC system, which we describe later in 1.2.11.

If the Hamiltonian is of the family of an Ising model, it can be simulated using quantum annealers or quantum simulators. There are other families of Hamiltonians that can be simulated on quantum simulators or coherent quantum annealers with more than one degree of freedom (see what Qilimanjaro is doing there).

1.2.11 Quantum Phase Estimation (QPE)

QPE was proposed by Alexei Kitaev in 1995 [35, 191]. Given a unitary U and an eigenstate $|\psi\rangle$ satisfying $U|\psi\rangle = e^{2\pi i\theta}|\psi\rangle$, QPE estimates the eigenphase θ . If the input is a superposition of eigenstates, measurement projects onto one eigenphase with probability equal to the corresponding squared overlap [192]. In quantum chemistry, QPE is therefore useful for energy estimation only when an initial state with sufficient overlap with the desired eigenstate can be prepared.

Complexity and possible speedup. QPE applies controlled powers U^{2^k} to a system register while a separate phase control register accumulates the eigenphase. The key components of a QPE are an n -qubit phase register corresponding to an output resolution of n bits, and m qubits which holds the initial state $|\psi\rangle$ on which controlled powers of its unitary U are applied. There is, however, a larger number of additional logical ancilla qubits used for the unitary preparation and other tasks. QPE's end-to-end speedup is application-dependent because implementing U , preparing $|\psi\rangle$ and extracting the useful observable can dominate the cost [193].

Initialization. The m -qubit system register must have the best overlap with the eigenstate of interest. In electronic-structure applications a Hartree-Fock determinant is a common inexpensive starting state, while multireference, tensor-network, adiabatic, variational and other correlated state-preparation methods can increase the overlap. A Hartree-Fock state is typically a computational-basis occupation state after a second-quantized fermion-to-qubit mapping, whereas a correlated state generally requires a nontrivial preparation circuit. There is no universal “10 mHa” accuracy for Hartree-Fock relative to exact molecular ground states. Its error is strongly system-dependent [192, 194–196].

Classical preprocessing may be significant because molecular integrals, active spaces, factorizations and the circuit representation of the simulated Hamiltonian must be generated before the quantum run. In second quantization, the m -qubit *system* register represents spin-orbital occupations, while the phase register is the QPE ancilla/control register. For $U = e^{-iHt}$, an energy eigenstate $|E_j\rangle$ satisfies $U|E_j\rangle = e^{-iE_j t}|E_j\rangle$. The measured phase is therefore related to E_j modulo $2\pi/t$, with the sign determined by the chosen convention.

Here is a decomposition of the various parameters of the QPE which are also described in Figure 27:

- **Phase register size:** n is the number of qubits in the phase register, *aka* the phase-register width. These n qubits are related to the n successive exponentiations of the unitary U :

$$U^{2^0}, \quad U^{2^1}, \quad U^{2^2}, \quad \dots, \quad U^{2^{n-1}}.$$

The phase register discretizes the dimensionless eigenphase θ into 2^n possible values, spaced by

$$\Delta\theta = \frac{1}{2^n},$$

when the eigenvalue convention is:

$$U|\psi\rangle = e^{2\pi i\theta}|\psi\rangle.$$

If each U^{2^k} is implemented through repeated applications of the elementary black-box unitary U , the total number of calls to U becomes:

$$1 + 2 + 4 + \dots + 2^{n-1} = 2^n - 1.$$

Thus, there are only n controlled powers of U which correspond to $2^n - 1$ elementary calls to U in the black-box model.

- **Phase accuracy, ϵ_θ ,** conditions the phase register width according to:

$$n \sim \log_2 \left(\frac{1}{\epsilon_\theta} \right).$$

This is why QPE has a logarithmic phase-register width but a linear in $1/\epsilon_\theta$ coherent evolution or query cost. For example, when needing about 10^{-3} precision in the phase estimation, $n \simeq \log_2(10^3) \simeq 10$ phase qubits are sufficient at the resolution level, while the total number of elementary calls to U is approximately $1 + 2 + \dots + 512 = 1023$ for a 10^{-6} phase precision. You then need about $n \simeq 20$ phase qubits and approximately $2^{20} \simeq 10^6$ applications of U . Doubling the phase register from 10 to 20 qubits improves the phase resolution by about 10^3 and increases the total coherent evolution time or black-box query count by roughly the same factor.

When QPE is used for computing a many-body energy estimation with $U = e^{-iHt}$, the phase and energy are related by:

$$E = -\frac{2\pi\theta}{t}$$

up to the sign convention and phase wrapping. Therefore an energy accuracy ϵ_E corresponds approximately to:

$$\epsilon_\theta = \frac{t\epsilon_E}{2\pi}.$$

The number of phase qubits required for a given chemical energy accuracy also depends on the chosen evolution time t and on the spectral range encoded into the $[0, 1)$ phase interval.

- **Circuit size.** The fundamental coherent evolution or black-box query cost scales as:

$$\Theta \left(\frac{1}{\epsilon_\theta} \right)$$

In this notation, Θ is the tight asymptotic bound notation used in computational complexity. It cumulates $O()$ which is an upper bound and $\Omega()$, a lower

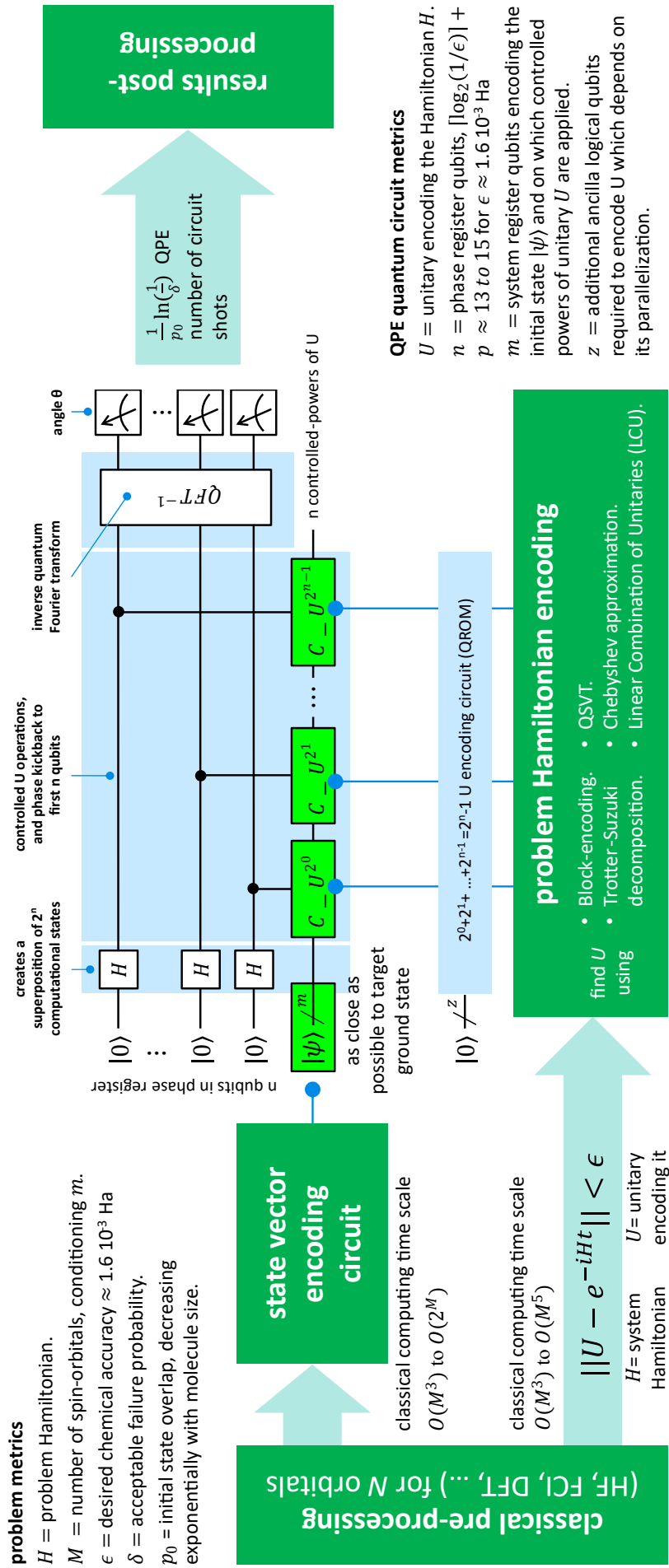


Figure 27: The quantum phase estimation algorithm explained with its classical parts in green and their complexity scale, on the top left are the problem parameters and on the bottom right are the circuit parameters including how to compute the number of logical qubits. This figure has been a lot refined compared to the last edition of this book. © Olivier Ezratty with various sources. 2022-September 2026.

bound. This is different from the total logical circuit depth. If one application of U has a circuit depth D_U and the successive powers are implemented serially through repeated applications of U , then approximately

$$D_{\text{QPE}} \sim D_U \frac{1}{\epsilon_\theta},$$

apart from state preparation, controls, ancilla management and the inverse QFT. The exact inverse QFT uses $O(n^2)$ gates, hence:

$$O \left[\log^2 \left(\frac{1}{\epsilon_\theta} \right) \right]$$

gates, which is asymptotically much smaller than the $O(1/\epsilon_\theta)$ coherent evolution cost. Its actual circuit depth depends on connectivity, compilation and parallelization and should therefore not be identified with its $O(n^2)$ gate count.

A semiclassical or iterative QPE can reduce the phase register to a single recycled control logical qubit. But it still must implement the successive evolution times associated with U^{2^k} . It therefore slightly reduces the logical qubit but does not remove the fundamental $\Theta(1/\epsilon_\theta)$ coherent evolution-time scaling.

- **Number of spin-orbitals** in the electronic-structure representation of the molecule, M , usually after selecting a finite basis set and possibly an active space. In a standard second-quantized encoding such as Jordan–Wigner, each spin-orbital occupation is represented by one logical qubit. The system register encoding $|\psi\rangle$ therefore contains approximately $m \simeq M$ logical qubits, before reductions obtained from symmetries, frozen orbitals or more compact encodings. One should distinguish spin-orbitals from spatial orbitals. In the usual spin representation, one spatial orbital generates two spin-orbitals. M is proportional to the molecular size to simulate Ω in Figure 27.

The number of orbitals affects not only the size of the $|\psi\rangle$ register but also the complexity of implementing $U = e^{-iHt}$. A generic second-quantized electronic Hamiltonian contains $O(M^4)$ two-electron integral coefficients before exploiting sparsity or factorization. Therefore, the cost D_U of one effective Hamiltonian evolution can itself grow polynomially and steeply with M . Modern techniques such as low-rank factorization, tensor hypercontraction, qubitization and optimized block-encodings can substantially reduce this cost. Hence the full QPE resource scaling should conceptually be viewed as:

$$\text{QPE cost} \sim \frac{\text{cost of implementing } U(M)}{\epsilon_\theta},$$

rather than simply as $1/\epsilon_\theta$.

- **Initial-state overlap** defines the gap between the initialized $|\psi\rangle$ and the system ground state. It conditions the probability that a QPE run returns the eigenvalue that is actually sought. With $|\psi_0\rangle$ being the target ground state, the prepared state is:

$$|\psi_{\text{init}}\rangle = \sqrt{p_0}|\psi_0\rangle + \sum_{j>0} c_j |\psi_j\rangle,$$

Then, the squared overlap amplitude p_0 and the overlap S are defined with:

$$p_0 = |\langle\psi_0|\psi_{\text{init}}\rangle|^2 = S^2$$

An ideal QPE run projects onto the ground-state eigenphase with an approximate probability of p_0 .

- **Number of circuit shots.** First, you have the expected number of circuit shots (runs) needed to obtain one successful projection on the eigenstate:

$$N_{\text{runs}} \sim \frac{1}{p_0}$$

The number of circuit shots you need to perform to reach a specified confidence level with failure probability below δ is R with:

$$(1 - p_0)^R \leq \delta$$

so, for small p_0 :

$$R \geq \frac{\ln(\delta)}{\ln(1 - p_0)} \simeq N_{\text{runs}} \ln \left(\frac{1}{\delta} \right) \simeq \frac{1}{p_0} \ln \left(\frac{1}{\delta} \right)$$

If the overlap is instead defined as the amplitude:

$$S = |\langle\psi_0|\psi_{\text{init}}\rangle|,$$

then $p_0 = S^2$ and the ordinary QPE repetition overhead scales as: $1/S^2$. This overlap may itself deteriorate with molecular size and electronic correlation. Therefore, increasing the number of orbitals can affect QPE twice: it increases the cost of representing and evolving the Hamiltonian, and it may reduce the overlap of an easily prepared approximate state, such as a Hartree–Fock state, with the exact many-body ground state.

QPE readout distribution. The QPE phase register used for the inverse QFT does not return a single deterministic value. With an exact input eigenstate $|\psi\rangle$ at the input and a regular n -qubit phase register,

measuring the register after the inverse QFT returns the solution integer k with a probability:

$$P(k) = \frac{1}{2^{2n}} \cdot \frac{\sin^2(2^n \pi \delta_k)}{\sin^2(\pi \delta_k)}, \text{ with } \delta_k = \theta - k/2^n$$

δ_k being the estimation error of candidate outcome k , expressed as a fraction of a full phase turn. This is a periodic sinc² profile, a mathematical distribution defined by squaring the cardinal sine function, mathematically expressed as $f(x) = \left(\frac{\sin x}{x}\right)^2$ (unnormalized) or $f(x) = \left(\frac{\sin(\pi x)}{\pi x}\right)^2$ (normalized). It is centered on the best n -bit approximation of θ . It is the same aliased diffraction pattern that a rectangular window produces in classical spectral analysis, and it was already derived in this form in 1998 [147]. Writing ε for the distance from θ to the nearest grid point in units of 2^{-n} , the profile simplifies for large n into $P = \text{sinc}^2(\varepsilon + m)$ over the integers m , which makes its spread independent of the register size.

The shot count R is then conditioned by three aspects:

- When the phase angle θ falls exactly on the 2^{-n} grid, with $\varepsilon = 0$, the readout is deterministic and one shot is sufficient. But this never happens with a molecular Hamiltonian (the green bar in chart A from Figure 28).
- The worst case is $\varepsilon = 1/2$, with θ sitting halfway between two grid points (the two red bars in chart A from Figure 28). The two neighbors then collect $4/\pi^2 = 40.5\%$ each, hence $8/\pi^2 = 81.1\%$ together and $80/(9\pi^2) = 90.1\%$ for the four best outcomes [147]. Averaged over a phase uniformly distributed inside a grid cell, the most likely outcome collects $2 \int_0^{1/2} \text{sinc}^2(\varepsilon) d\varepsilon = 77.4\%$.
- Individual tail outcomes decay only as $1/d^2$, d being the distance to the peak in grid units, so the cumulative tail decays as $1/d$. This is slow, not fast, and it is the single most consequential feature of the distribution. In the worst case the cumulative tail is worth $2/(\pi^2 d)$, that is 5.0% beyond 4 grid units and 1.3% beyond 16, the asymptotic expression matching the exact sum to better than 1% relative from $d = 4$ on. The corresponding textbook bound is $1/(2(d-1))$ [147], which is loose by a factor of about 3 at $d = 4$.

That slow tail motivates adding p confidence or padding qubits when one wants the leading n phase bits with high success probability. Padding increases the maximum coherent power of U and therefore the evolution/gate cost, not necessarily the number of independent circuit shots. The distribution is also circular, so a raw arithmetic mean of integer labels can be a poor estimator. Modes, circular estimators or maximum-likelihood methods are more natural. More generally, optimized control states

can improve confidence intervals and metrological loss functions [147, 197].

With a perfect eigenstate and a suitably designed phase-estimation protocol, only a small number of independent samples may be needed to identify the relevant phase peak. Additional repetitions arise from imperfect state overlap, hardware noise, confidence requirements and from statistical phase-estimation variants that trade coherent evolution depth for more circuit executions. If the desired eigenstate has weight $p_0 = 5\%$, about 20 independent preparations are required on average for one sample from that component. The fundamental Heisenberg scaling concerns total coherent evolution/query cost, $O(1/\epsilon)$, versus the $O(1/\epsilon^2)$ sample scaling of standard incoherent estimation [192, 193, 199].

Optimizing the control state is an active research topic. A discrete prolate spheroidal sequence minimizes a confidence interval for a fixed controlled-unitary budget. Kristjuhan and Berry showed in 2026 that these states can be approximated by bond-dimension-4 matrix product states up to the tested dimension 2^{24} , prepared by simple three-qubit operations, and that their power-of-two construction can use a three-qubit control register [200]. This reduces control-register width but does not remove the required controlled powers or total evolution-time scaling. Statistical post-processing can further reduce repetition costs in specialized variants [201].

QPE and ground-state energy. For $U = e^{-iHt}$, an eigenenergy E is encoded as an eigenphase e^{-iEt} . With the convention $e^{2\pi i\theta} = e^{-iEt}$, $E = -2\pi\theta/t$ modulo $2\pi/t$. The evolution time t is therefore not an arbitrary constant: it sets both the energy window and the conversion between phase resolution and energy resolution. With an n -bit phase estimate, the ideal grid spacing is of order $\delta E \simeq 2\pi/(t2^n)$. Hence a target such as 1 mHa does not by itself imply a universal number of phase qubits. The Hamiltonian-simulation approximation error, state-preparation overlap and confidence target must be included as well [193, 202].

Unitary encoding. This encoding process of the unitary U based on the problem Hamiltonian H is using techniques like block-encoding (for dense Hamiltonians or when combined with QSVT for high-accuracy transformations, with a cost scaling at least quadratically with the number of molecular orbitals [203, 204]), qubitization (faster, but requiring ancillary qubits and QROM [205, 206]), Trotter-Suzuki decomposition (for sparse Hamiltonians with simple decompositions), Chebyshev approximation (with a better accuracy outcome than with a Trotter-Suzuki decomposition under some circumstances), a QSVT (for high-precision, polynomial-based transformations of block-encoded Hamiltonians), and LCU (Linear Combination of Unitaries) (for Hamiltonians with a natural decomposition into a sum of unitaries, as seen in electronic structure problems expressed with Pauli operator) [207–210]. An LCU can be built with preparing a d -diagonal unitary with d diagonals having non-null values [67], particularly to solve PDEs.

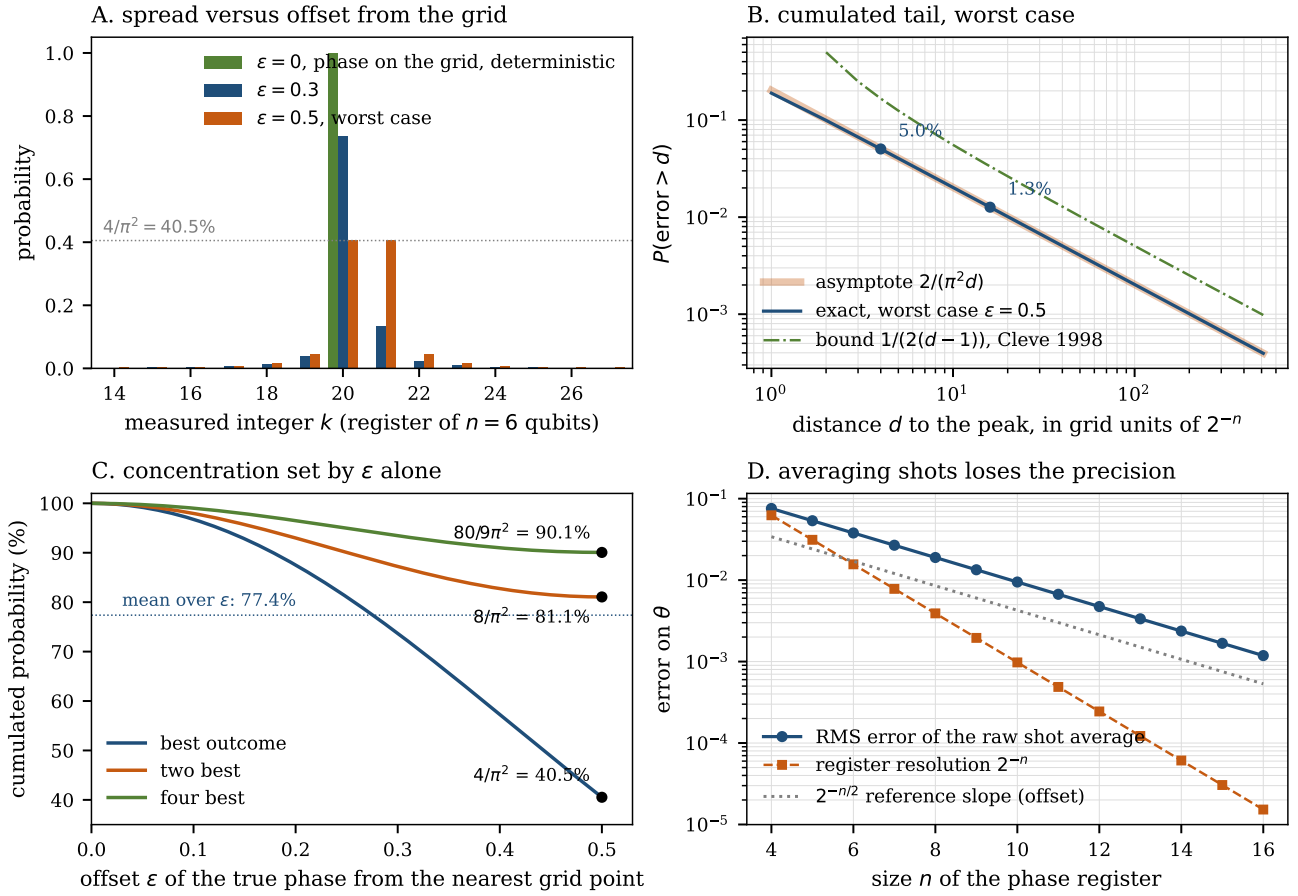


Figure 28: Readout distribution of a textbook QPE. (A) shows the integer-output probability for k using a 6-qubit phase register. With an exact eigenstate and $\epsilon = 0$, the phase register returns a single integer only when θ lies exactly on the 2^{-n} grid, here with integer outcomes from 0 to 63. Otherwise the outcome spreads over a periodic sinc^2 profile and is broadest when θ is halfway between two neighboring grid points. (B) the cumulated tail decays only as $1/d$, d being the distance to the closer peak value in the grid, which is why p padding qubits can be added on top of the n qubits set by the target accuracy. The tail is the set of outcomes that are far from the peak, that is all measured integers more than d grid units away from θ , whose total weight $T(d) \simeq 2 \sin^2(\pi\epsilon)/(\pi^2 d)$ is the probability that the phase estimate is off by more than $d \cdot 2^{-n}$. (C) the cumulative probability or output concentration depends only on the offset ϵ from the grid, not on the register size, with the classic $4/\pi^2$ worst case. (D) the uniform-control-state distribution has long $1/d^2$ probability tails. A naive arithmetic mean of raw integer outcomes therefore need not achieve the same 2^{-n} scaling suggested by the modal grid resolution, and it also ignores the circular nature of phase. Robust or likelihood-based estimators are generally preferable. © Olivier Ezratty, 2026, computed from [147] on August 8th, 2026.

Initial-state overlap. In large correlated systems, the overlap between an inexpensive trial state such as a single Hartree-Fock determinant and the desired eigenstate can become small, making state preparation a central QPE bottleneck because the probability of projecting onto that eigenstate is the squared overlap. Anderson’s orthogonality catastrophe is a specific many-body phenomenon in the thermodynamic limit. In quantum chemistry, that overlap can deteriorate strongly with correlation and system size. Divide-and-conquer, multireference, tensor-network, adiabatic and other state-preparation methods aim to mitigate this problem [192, 211].

QPE is also used in quantum linear algebra (HHL) and quantum random walks. A QPE circuit is also very similar to Shor’s integer factoring algorithm quantum part, the difference being in the way the period finding

part is encoded and the circuit output exploited, using some classical continued-fractions computing [212].

QPE variants. Resource estimates also depend strongly on whether the electronic problem is represented in first or second quantization [213, 214]. In first quantization, registers encode particle coordinates or basis-state labels and the many-fermion wavefunction must be antisymmetrized. In second quantization, qubits encode occupations of single-particle modes or spin-orbitals and creation/annihilation operators are mapped to qubit operators. Neither representation is inherently restricted to “simple” or “complex” systems, and second quantization is not mainly a light-matter formalism. Recent first-quantized spectrum-amplification constructions can reduce QPE block-encoding normalization and resource estimates substantially for selected systems [215].

Q How many logical qubits are needed for a QPE?

The logical-qubit requirement of a QPE-based chemistry is highly architecture- and algorithm-dependent on the encoding, Hamiltonian-simulation algorithm and circuit implementation. For challenging molecular benchmarks, published fault-tolerant resource estimates can range from hundreds to several thousand logical qubits, depending strongly on the chosen space-time trade-off.

Here’s how this is decomposed, as shown in Figure 27:

- The phase-estimation register size is determined by the required energy resolution, the base evolution time t , the spectral interval that must avoid phase aliasing, and the target failure probability. For $U = e^{-iHt}$, the energy grid spacing is approximately $\delta E = 2\pi/(t2^n)$ before additional algorithmic errors. Therefore the chemical-accuracy target alone does not determine a universal value such as 13 to 15 phase qubits. Semiclassical or iterative phase estimation can instead extract the phase information sequentially using a single reusable control qubit, with intermediate measurements and classical feed-forward.
- In a common second-quantized encoding, the system register holding $|\psi\rangle$ typically uses one qubit per spin-orbital before symmetry reductions and more compact encodings are applied. Its size is

therefore tied to the chosen orbital basis, typically twice the number of spatial orbitals in a conventional nonrelativistic representation rather than to the electron count alone. First-quantized encodings use different representations and therefore have different qubit-count and computational-cost trade-offs.

- Additional logical ancillas are required by the chosen Hamiltonian-simulation or block-encoding construction. QROM-based qubitization schemes can trade additional workspace qubits for reduced Toffoli count and/or circuit depth, while other simulation methods require very different amounts of workspace. Tensor hypercontraction and related factorizations can substantially reduce Hamiltonian-representation and simulation costs [198]. Thus, a large ancilla count is a feature of particular resource-optimized constructions, not an intrinsic requirement of QPE itself. We can add here the padding qubits described before.

Table 3 shows how these resources evolved over time with the improvements in QPE for computing the ground state of the FeMoCo compound. It traded time and T-gate numbers against an increase in number of logical qubits. More recent resource estimates also account for the cost of encoding the initial state ψ .

Other ground-state preparation approaches include adiabatic state preparation, dissipative methods and iterative phase-estimation variants. Adiabatic state preparation can provide high overlap when a gapped path from an easy initial Hamiltonian exists, but long coherent evolution can make it demanding rather than generically NISQ-robust [216]. Iterative Phase Estimation (IPEA) reuses a single control qubit and applies feedback phase corrections followed by Hadamard measurements. It does not simply replace the QFT with one Hadamard transform [217, 218]. Other proposals extend eigenvalue estimation to non-Hermitian problems or replace the inverse QFT with QSP-like phase filters [219–221].

The quantum fast multipole method (QFMM) proposed by Google in 2025 is based on the first quantization and lowers gate complexity compared to QPE. It implements a quantum version of the fast multipole method (FMM) to compute Coulomb interactions with quasilinear complexity in the number of particles. It is better suited for computing time evolution and dynamics [222].

Some QPE variations exist that could potentially run on NISQ QPUs for small chemical systems [201, 223–226]. A deterministic single-shot variant of QPE was demonstrated in a photonic system in 2026, with a circuit

complexity linear instead of exponential in the register size. It only applies to the restricted class of unitaries whose eigenphases are exact multiples of $2\pi/2^n$, typical of cyclic group representations and of periodically evolving systems, so it does not help with molecular Hamiltonians [227].

1.2.12 Quantum Imaginary Time Evolution (QITE)

Quantum Imaginary Time Evolution (QITE) is a ground-state preparation method inspired by imaginary-time propagation. Ideal normalized imaginary-time evolution maps a trial state to $|\psi(\tau)\rangle = e^{-\tau H}|\psi(0)\rangle/\|e^{-\tau H}|\psi(0)\rangle\|$, exponentially suppressing excited-state components when the initial state has nonzero overlap with the ground-state subspace [234].

$$|\psi(\tau)\rangle = \frac{e^{-\tau H}|\psi(0)\rangle}{\|e^{-\tau H}|\psi(0)\rangle\|}$$

Because $e^{-\tau H}$ is non-unitary, the original QITE algorithm approximates each short imaginary-time step by a local unitary whose generator is inferred from measured

quantum algorithm	logical qubits	unitary preparation qubits	T or Toffoli count	circuit depth	computing time estimate
Reiher et al. 2017 [228] <i>trotterization</i>	111	~ 2	$\sim 1.0 \times 10^{14}$ (phase estimate)	$\sim 10^{14}$	months
Berry et al. 2019 [229] <i>sparse qubitization</i>	1,232	$\sim 1,123$	$\sim 4.0 \times 10^{10}$ (phase estimate)	$\sim 10^{10}$	weeks
Lee et al. 2021 [230] <i>tensor hypercontraction (THC</i> <i>[198]) and qubitization</i>	2,142	$\sim 2,033$	$\sim 2.1 \times 10^{10}$ (phase estimate)	$\sim 5.3 \times 10^9$	4 days
Low et al. 2025 [231] <i>spectrum amplification</i>	1,137	$\sim 2,033$	$\sim 3.41 \times 10^9$ (phase estimate) $\sim 7.3 \times 10^{10}$ (end-to-end)	N/A	several weeks
Watts et al. 2026 [232] <i>dissipative ground state preparation</i>	1,137	$>2,000$	$\sim 3.41 \times 10^8$ (end-to-end)	N/A	N/A

Table 3: resource estimates for simulating the FeMoco active space $CAS(54e, 54o)$ to chemical accuracy, demonstrating the space-time trade-off between logical qubit width and T-gate depth. The ψ encoding register accounts for ~ 108 logical qubits, and the phase register requires 1 logical qubit across all estimates thanks to the use of a semi-classical QFT. Quantum computing resource estimates are usually based on using superconducting qubits and surface codes. The phase estimates corresponds only to the QPE while the end-to-end includes the ψ state preparation cost. The number of circuits runs are not indicated here. In 2026, Garnet Chan et al. computed FeMoCo’s ground state classically using DMRG tensor networks (defined in 1.7.2) and estimated a computing time of less than a week on a Frontier HPC for 54 and then, 76 orbitals (113-electron). Source: [233]. Added in September 2026.

expectation values and a classically solved linear system [234]. Ideal imaginary-time evolution converges toward the ground-state subspace when the initial overlap is nonzero, but practical QITE inherits errors from locality truncation, finite step size, measurement noise and the conditioning of the classical linear systems. Polynomial resource scaling is therefore conditional on the problem structure and approximation regime, and noise robustness is not guaranteed generically.

There is no demonstrated exponential speedup for QITE although it could showcase some practical speedups in some circumstances [235, 236]. QITE sits in between VQE (NISQ) and QPE (FTQC) and is frequently combined with these algorithms [237–240]. It could potentially be used in Quantum Machine Learning [241].

The DB-QITE variant [242] uses Brockett’s double-bracket flow [243] to iteratively diagonalize or reorder a Hamiltonian in a chosen basis. Under the assumptions of the method, its iterations can increase overlap with a desired eigenstate while reducing circuit resources relative to some QITE constructions. QPE does not require classical knowledge of the ground-state overlap. It only requires that the prepared state have sufficient overlap, because that overlap determines the probability of observing the desired eigenphase.

A variational alternative to measurement-based QITE is Variational Quantum Time Evolution (VarQTE), whose “bowtie” variant proposed by IBM researchers in 2026 reduces quantum resources by classically simulating only the causally relevant subcircuits within the light-cone

intersection used to evaluate its gradient and quantum geometric tensor terms [244].

There are many other proposed algorithms that are intermediate between VQE and QPE to computed many-body quantum systems excited states, which subdivide these systems in subspaces that are computed independently on QPUs: Quantum Subspace Expansion (QSE), Quantum Equation of Motion (qEOM), and Quantum Self-consistent Equation-of-Motion (q-sc-EOM) [245]. They may become relevant in early-stage FTQC QPUs.

1.2.13 Quantum Amplitude Estimation (QAE)

Quantum Amplitude Estimation (QAE) was introduced by Brassard, Høyer, Mosca and Tapp [185]. It estimates a success probability or expectation value encoded as an amplitude. The original algorithm combines amplitude amplification with phase estimation and achieves additive error ϵ using $O(1/\epsilon)$ coherent oracle or Grover-operator uses, compared with $O(1/\epsilon^2)$ independent samples for standard classical Monte Carlo estimation. QFT-free, maximum-likelihood, iterative and Bayesian variants reduce ancilla count or coherent depth, but can still require increasingly high powers of the Grover operator, so describing them generically as NISQ-ready is too strong [246–248]. QAE is relevant to quantum Monte Carlo, finance and signal-processing proposals [249].

1.2.14 Quantum Signal Processing (QSP)

Quantum Signal Processing (QSP), developed by Guang Hao Low and Isaac Chuang, interleaves repeated uses of a signal unitary with parameterized single-qubit phase rotations so that a degree- d polynomial transformation is implemented on the eigenphases or encoded scalar of an invariant $SU(2)$ subspace [250]. The query complexity is $O(d)$ signal-unitary uses, not a constant number. QSP enables asymptotically optimal Hamiltonian simulation and underlies several phase-filtering and signal-processing constructions, but it does not by itself imply an exponential end-to-end speedup.

QSP has been implemented experimentally on small processors [251, 252]. QSVT is the matrix-valued generalization of QSP, not a primitive that QSP itself requires. In practical QSP synthesis, a target polynomial P must satisfy parity and boundedness constraints and is accompanied by a complementary polynomial or equivalent phase-factorization problem, which becomes a nontrivial classical preprocessing step [253]. Extensions have also been proposed for continuous-variable systems and sensing [254, 255].

A Generalized Quantum Signal Processing (GQSP) extension was introduced in 2023 to generalize QSP algorithm with fewer restrictions on the entry matrix [256, 257]. Other quantum digital processing techniques can avoid using a QFT and provide better speedups for lowpass and high-pass signal filtering [258].

A $U(N)$ generalization proposed in 2026 extends the original two-dimensional signal-processing framework so that several polynomial transformations can be realized simultaneously [259]. The authors give applications including bivariate polynomial functions and amplitude estimation with asymptotically optimal query complexity. These are theoretical query-complexity results under the paper’s access model and should not be phrased as a generic demonstrated quantum advantage.

A variant of QSP introduced in 2024 enables the efficient simulation of bosonic systems [260].

Analytical alternatives to numerical angle finding are emerging, with closed-form phase sequences obtained from orthogonal polynomial theory for Hermite, Jacobi and Rogers-Szegő families, requiring $2n+2$ angles for degree n and yielding $O(\log(1/\epsilon))$ gate counts for smooth functions through Hermite series expansions [261].

One QSP application is the analytical design of bounded-error quantum-control and pulse sequences for superconducting qubits [262]. The QSP-derived phase sequence can be computed or compiled classically and then executed by the ordinary control stack.

In the signals processing domain, we can add the technique of forrelation, introduced by Scott Aaronson in 2014 [263]. It estimates the correlation between a function and the Fourier transform of another function. It brings an exponential speedup, but it has no known use case, beyond indirect applications in cryptanalysis in as-

sociation with the Deutsch-Jozsa algorithm [264]. It has nonetheless regained interest as a quantum-advantage benchmark. Its 2-forrelation variant, which realizes the maximal separation between classical and quantum query complexity and underlies the oracle separation of BQP from the polynomial hierarchy, was shown to be solvable by commuting-gate IQP (Instantaneous Quantum Polynomial-time) circuits with only two queries, or a single circuit and query in the signed case. It provides a route to quantum advantage on a verifiable decision problem rather than on a hard-to-check sampling task [265].

1.2.15 Linear and differential equations (HHL, QLSA, PDE)

Many other quantum algorithms exist that allow complex mathematical operations such as solving various linear algebra problems, inverting matrices, or differential equations [266]. They are then used elsewhere as in quantum machine learning (see 1.3.6) and in fluid dynamics simulations (see 1.3.5).

HHL is the 2009 quantum linear-system algorithm of Harrow, Hassidim and Lloyd [53, 158]. Given efficient access to a sparse Hermitian $D \times D$ matrix A and an efficiently preparable state $|b\rangle$, it prepares a normalized quantum state proportional to $A^{-1}|b\rangle$. In the original oracle model, the runtime is polynomial in sparsity s , condition number κ and $1/\epsilon$, but only polylogarithmic in the matrix dimension D . This can yield an exponential-in- $\log D$ separation from algorithms that must explicitly manipulate all D components, but only under strong input-access and output assumptions. qRAM is one possible state-preparation model, not a mandatory physical component of HHL.

A commonly quoted scaling for the original HHL construction is $\tilde{O}(s^2 \kappa^2 \log(D)/\epsilon)$ up to oracle-model and implementation details, where:

- D is the dimension of matrix A . If $D = 2^N$, then $N = \log_2 D$ is the number of index qubits.
- s being its sparsity, i.e., the maximum number of non-zero entries per row.
- κ is the condition number of A , $\kappa(A) = \|A\| \|A^{-1}\|$, which controls the sensitivity of the solution to perturbations in the input data.
- ϵ being the desired precision of the approximation.

The HHL quantum circuit is decomposed in three steps: a QPE, a rotation *aka* an inversion circuit and an inverse QPE (or “QPE uncompute”). The inversion circuit applies a series of controlled rotation gates on register C (referenced in Figure 29). It can be simplified in so-called Hybrid HHL versions [267]. HHL used for matrix inversion could be simplified by replacing its QPE by a continuous time quantum walk [268].

HHL’s original Hamiltonian-simulation step assumes a Hermitian matrix. A non-Hermitian matrix A can be embedded into the Hermitian dilation:

$$\begin{bmatrix} 0 & A \\ A^\dagger & 0 \end{bmatrix}$$

This Hermitian embedding is conceptually distinct from block-encoding, although modern QLSAs often use block-encodings as their access model.

Scott Aaronson emphasized the crucial output caveat in 2015 [159]. HHL prepares a quantum state proportional to the solution vector rather than returning all D components as a classical array. Estimating selected overlaps or observables can preserve a quantum advantage. Reconstructing all components necessarily incurs at least $\Omega(D)$ classical output work and, depending on the required precision and observable access, can require substantially more samples. As a result, HHL is not generically exponentially faster than classical methods to compute a full classical matrix inverse.

HHL is an algorithm that can be piggybacked by other algorithms to solve interesting problems. It is mostly used with quantum machine learning algorithms. It can also be used to solve physics and engineering problems involving Poisson’s equation [269].

HHL is mostly a FTQC-related algorithm, but some researchers are trying to “squeeze” it somehow to run it on NISQ platforms [270, 271]. A variational variant prepares the solution as an eigenvector of a self-adjoint matrix through a von Neumann measurement implemented with a QPE, removing the Pauli string decomposition that confines usual variational solvers to sparse matrices. Its accuracy is set by the number of shots rather than by the condition number κ , at the cost of $O(\log \kappa)$ qubits for the eigenvalue register, but it was only checked on dense random 16×16 matrices [272].

There is another way to invert a matrix, or more specifically, to find values of interest of the inverse of a matrix, a bit like what HHL can yield. It uses a convoluted mathematical workaround that is worth describing. Following Jacobi’s formula, restated by Kenneth Kubota in 1994 [273], the first derivative of the logarithm of the determinant of a matrix X over its component values yields the values of the inverse of the matrix X , $Y = X^{-1}$, transposed. For a symmetric or Hermitian X , which is the case of interest here, the transposition is immaterial and the elements y_{ij} directly compose Y :

$$\frac{\partial}{\partial x_{ij}} \ln \det(X) = y_{ji}$$

As proposed in 2025 in [274], a quantum gradient algorithm based on a quantum Fourier transform (QFT) can focus efficiently on the eigenvectors and eigenvalues

of the matrix. This works for Hermitian matrix with positive eigenvalues (positive definite) having a low effective rank, meaning that only a small number of its eigenvectors (say, k) are relevant for the computation. Also, the matrix must be accessible through a quantum oracle circuit. This algorithm can have applications in quantum machine learning, statistical physics simulations and in application requiring fast matrix inversion or determinant estimation (Figure 30).

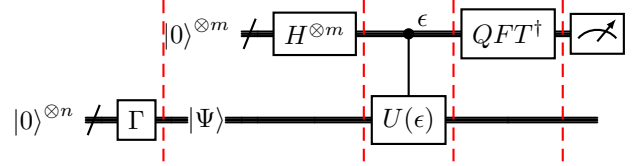


Figure 30: The quantum gradient algorithm quantum circuit proposed in [274].

QLSA (Quantum Linear System Algorithm) is a broad family of quantum algorithms to solve linear system problems. HHL is the first iteration of QLSA, using a phase estimation, controlled rotations and an inverse QPE. Then, more modern versions of linear solvers came out like Variable Time Amplitude Amplification (VTAA) in 2012 which improved the scaling from $\mathcal{O}(\kappa^2)$ to $\mathcal{O}(\kappa)$ [275], the Linear Combination of Unitaries (LCU) in 2017 which used a Chebyshev expansion of A^{-1} to achieve an exponential improvement in precision scaling, bringing the complexity to $\mathcal{O}(\kappa s \text{ polylog}(1/\epsilon))$ [276], Quantum Singular Value Transformation (QSVT) in 2019 which unified these approaches using block encoding with applying a polynomial approximation directly to the singular values of A to synthesize A^{-1} , eliminating the need for the QPE used in HHL [188]. Then, the Shortcut method introduced in 2024 helped reduce the constant prefactors [277, 278].

On real quantum computing hardware, what matters is less the asymptotic query complexity in the condition number κ and the inverse precision, than the degree of the polynomial approximation of $1/y$ that QSVT actually implements. It conditions the circuit depth, and its normalization outside the spectrum, which also sets the number of circuit shots. Constrained optimal polynomials borrowed from classical Krylov subspace theory, either uniform (CUP) or adapted to a spectral density reconstructed from QSVT moments (CAP), cut the noise-induced error, reaching a 1.5% relative error on an IBM processor using fewer than 15 qubits against 18.3% for a standard QSVT solver on the same preconditioned elliptic problem [279]. Another experiment adapted the classical Kaczmarz row-action method [280], which enforces one equation at a time and thus avoids oracle access to matrix entries, with a claimed $\mathcal{O}(\log(s)/\epsilon)$ circuit depth using $\mathcal{O}(s)$ ancilla qubits. Its suggested independence from the matrix condition number remains a conjecture [281].

End-to-end measurements of an industrial linear algebra workflow are hard to find. One example came from IonQ

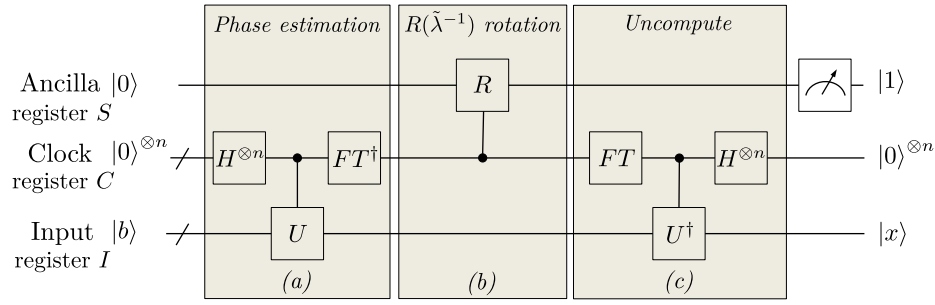


Figure 29: The HHL linear-system circuit. Its Hamiltonian-simulation or block-encoding subroutine requires an efficient access model for matrix A , while the right-hand side must be prepared as $|b\rangle$. qRAM is one possible access model, not a mandatory component. The output is a normalized quantum state proportional to $A^{-1}|b\rangle$, which is most useful when only aggregate observables are extracted or when it feeds another quantum routine. R is a single ancilla qubit. 2022-2026.

and Synopsys who used the multiphysics simulation software Ansys LS-DYNA on meshes of up to 35 million elements along an iterative QAOA handling the graph partitioning step that minimizes the fill-in of the sparse factorization. It was classically emulated on up to 150 qubits with CUDA-Q and with 36 qubits on an IonQ Forte system. The full wall-clock time improved by 7% to 15% depending on the model, a rare full-stack data point. But the quantum part solved an optimization subproblem and not the linear system itself, and the classical partitioning baseline was not state of the art [282].

Other linear equations. In 2025, researchers from Google AI and Macquarie University proposed an efficient way to solve another linear equation, with finding matrix X by solving the Sylvester equation where $AX + XB = C$, A , B and C being given complex number matrices. As with HHL, it does not efficiently return the full solution matrix X classically, but it can estimate selected properties of the encoded solution [283]. The same team extended this block-encoding approach to linear matrix differential equations, where an entry of the solution matrix is obtained with $\tilde{O}(\nu \mathcal{L} t / \epsilon)$ queries, $\mathcal{L}$ being a time integral of norm bounds on the evolution operators. A matching $\Omega(\nu \mathcal{L} t / \epsilon)$ lower bound makes it optimal up to logarithmic factors. Encoding the solution in a matrix rather than in a quantum state avoids the exponentially small amplitudes that make most quantum ODE solvers exhibit exponential runtimes for this task. It was applied to the dissipative dynamics of non-interacting fermions, with a polynomial speedup for lattice systems that grows with the interaction range [284].

Another problem is computing the stationary distribution of Markov chains. It is about finding a stationary distribution π that satisfies the equation $\pi P = \pi$. π is a “left” eigenvector of P (left in the previous equation) with an eigenvalue of 1 ($\pi P = \lambda \pi$, with $\lambda = 1$). P is a Markov chain’s transition input matrix. It is a square $n \times n$ matrix that represents the probability of transitioning from a state i to state j with each row summing to 1 and all entries being non negative. It represents the dynamics of a discrete-time Markov chain. This is the

problem input data. π is a stationary distribution $1 \times n$ row vector which describes the long-term probability of being in state i . P indicates how the system evolves and π describes where the system settles in the long run if it does. Solving this kind of problem is about finding π or characteristics of π . It is useful in many domains, such as economics and market modeling, helping to predict average wait times in call center queues, and in reliability and maintenance engineering.

Ordinary Differential Equations (ODE) are differential equations involving functions of a single independent variable and its derivatives. The dependent variable is usually a function of just one variable like time [288], and the equation involves derivatives with respect to this single variable. Various quantum algorithms are available to solve ODEs. They can be based on HHL, QSVT, other breeds of QLSA [289], and even VQE [290]. One emerging quantum ODE solver is new variants of the linear combination of Hamiltonian simulation (LCHS) algorithm which bring a significant reduction in runtime costs, particularly in terms of constant-factor overheads [291].

Delay differential equations, where the derivative depends on the history of the system, can sometimes be embedded into augmented ODEs and then treated by Hamiltonian-simulation methods under assumptions on the memory kernel and stability [292]. Quantum Koopman Algorithms proposed in 2026 instead work in observable space. For the specific class of N free fermions linearly coupled to a bath, the authors derive $O(\text{polylog}(N))$ gate complexity and an exponential asymptotic separation from the classical methods considered, while also proposing perturbative methods for nonlinear dynamics [293]. This is a structured-model result, not a generic exponential speedup for arbitrary nonlinear dynamical systems.

Partial Differential Equations (PDE) are differential equations involving multiple independent variables and their partial derivatives against some of these variables. The equation unknown is a function with several variables like time and space. Solving a PDE involves finding the unknown function that satisfies the given equation, usually using some boundary or initial condi-

Q Read the fine print, again

An analog quantum walk based quantum algorithm solving the stationary distribution of Markov chains problem was proposed in 2019-2020 by Shantanav Chakraborty in [285]. It is supposed to run on QPU implementing Hamiltonian evolution using neutral atoms or Rydberg arrays, which aligns with the algorithm’s reliance on interpolated quantum walks and continuous-time dynamics. Its speedup was only quadratic.

In August 2025, Ward van der Schoot and Niels M. P. Neumann published a paper where they did show how this speedup was probably overestimated [286]. They made a parallel with Scott Aaronson famous “Fine Print” paper on HHL [159]. Here, the caveat was that the analog algorithm from Chakraborty et

al. can generate some theoretical speedup only if “*certain variables are available in the application of the algorithm*”. An “*in general, the complexity to compute these variables is higher than the complexity of the algorithm*”. More precisely, *this complexity is actually the same as the classical time to prepare the stationary distribution, implying that the claimed speedup does not hold*. And this didn’t even consider the practical hardware implementation details on an analog quantum computer and related scaling aspects. This confirms a point you will see elsewhere in this book, particularly when looking at case studies and use case analysis: it must be analyzed in a full-stack manner with considering all the classical steps involved in solving the given problem.

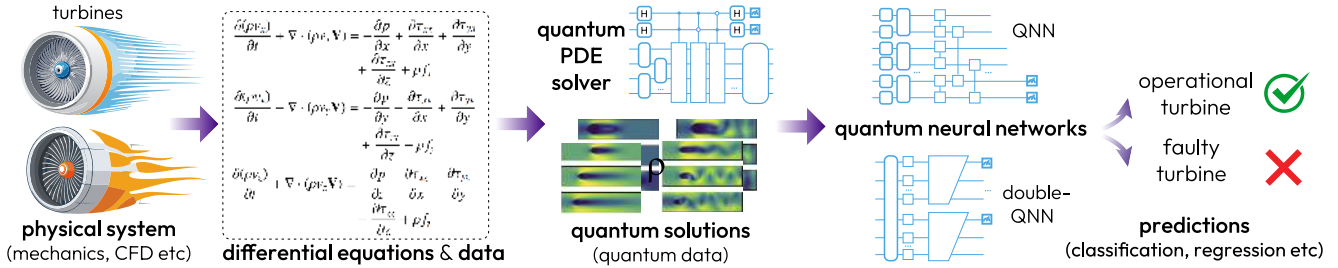


Figure 31: An efficient way to exploit the results of a quantum PDE is to use a quantum neural network to extract useful values, here a simple classification between operational and faulty turbine. Source: [287].

tions [294]. The Schrödinger wave equation is one of the most famous PDEs. PDEs are linear or nonlinear [295]. They are used in physics, fluid mechanics, nuclear fusion, engineering, financial services and applied mathematics [296].

Other famous PDEs are the Laplace and Poisson equations that are used in theoretical physics, the heat equation in thermodynamics [297], the Hamilton-Jacobi-Bellman nonlinear equation used to solve many problems in engineering, economics, portfolio optimization, pricing of derivatives, path planning, for optimizing power distribution and energy storage, and trajectory design, and the advection equation which is used in fluid dynamics, meteorology, and environmental science.

Quantum PDE algorithms use several distinct strategies, including spectral/QFT methods, quantum linear-system algorithms after spatial discretization, Hamiltonian simulation or Schrödingerization, block-encoded differential operators, and variational approaches [298–302]. It is therefore too broad to say that most PDE solvers are based simply on QFT for frequency and QPE for time. Finite-element and finite-difference discretizations can also map PDEs to linear systems or time-evolution problems, often with substantial condition-number and data-loading costs [303].

How the derivative operator is discretized matters as much as the solver itself. Non-local SLAC derivatives, whose Fourier symbol matches the continuum momentum exactly inside the first Brillouin zone and therefore avoid mirror modes, can be block-encoded with an LCU and preconditioned in a Shannon wavelet basis that brings the condition number down to a small constant, with a $\mathcal{O}(dn^3\alpha^{(k)}\log(1/\epsilon))$ gate complexity for a d -dimensional PDE [304]. When the operator becomes diagonal in Fourier space, as with the Poisson, Helmholtz, heat and anisotropic diffusion equations, its inversion can be done with quantum reversible arithmetic on the diagonal symbol instead of a generic QSVT, which was checked in noiseless Qiskit simulations [305]. Direct Trotterized finite difference schemes also benefit from tighter error analysis. Bounding errors in vector norm instead of operator norm cuts the required number of time steps by $\Theta(16^n)$ for the anisotropic diffusion equation and $\Theta(4^n)$ for the convection equation, n being the number of qubits per dimension, an exponential reduction that is a change of analysis and not of algorithm [306]. Lastly, QLSAs lack the residual monitoring that tells classical iterative solvers when to stop, and instead run for a fixed worst-case schedule. Adding residual variables to the dynamics gives a convergence flag that is readable without reconstructing the solution vector. It

halted a two-dimensional finite element Poisson problem well before its conservative runtime estimate [307].

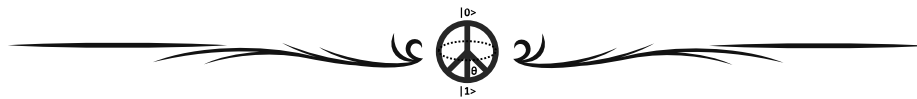
PDEs could also be solved with QPINN (Quantum Physics Informed Neural Networks) which embed physical laws directly into neural network architectures, eliminating the need for labeled training data while penalizing violations of physical constraints in the loss function [308, 309]. Another proposal comes from China, the Quantum Random Feature Method (QRFM), is bringing only a quadratic speedup [310]. And yet another one using the Finite Element Method (FEM) [311].

One problem with solving PDE is how to extract the results from a quantum algorithm. One solution consists of processing it with a quantum neural network [287] (see Figure 31).

Many of the quantum PDE proposals that rely on deep Hamiltonian simulation, QLSA or QSVT circuits are

naturally fault-tolerant algorithms. Their theoretical speedups range from polynomial to exponential under different sparsity, conditioning, input-state and output assumptions, so there is no single generic polynomial-speedup statement [312, 313]. Variational, QAOA-like, analog and annealing approaches have also been explored for small PDE instances, but no quantum advantage has been demonstrated for these NISQ implementations [314–322].

Another linear algebra problem is inverting a covariance matrix using a quantum Gaussian Process Regression. It can be used for PCA (principal component analysis) tasks. But it provides no exponential speedup even when putting aside classical data loading. A covariant matrix is a square matrix giving the covariance between each pair of elements of a given random vector [323].



1.2.16 Bibliography and notes

These are the bibliographical references and notes for the **Basic algorithms toolbox** subsection.

- [31] [Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer](#), by Peter W. Shor, SIAM Journal on Computing, October 1997 (26 pages).
- [32] [On the Power of Quantum Computation](#), by Daniel R. Simon, SIAM Journal on Computing, October 1997 (10 pages) 📄.
- [33] [Rapid solution of problems by quantum computation](#), by David Deutsch and Richard Jozsa, Royal Society Proceedings A, December 1992 📄.
- [34] [Quantum Complexity Theory](#), by Ethan Bernstein and Umesh Vazirani, SIAM Journal on Computing, October 1997 (63 pages) 📄.
- [35] [Quantum measurements and the Abelian Stabilizer Problem](#), by Alexei Yu. Kitaev, arXiv, November 1995 (22 pages).
- [39] [A Fast Quantum Mechanical Algorithm for Database Search](#), by Lov K. Grover, Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing (STOC '96), May 1996 (8 pages).
- [43] [Quantum Random Access Memory](#), by Vittorio Giovannetti, Seth Lloyd, and Lorenzo Maccone, Physical Review Letters, April 2008 (4 pages).
- [44] [Quantum Computational-Sensing Advantage](#), by Saeed A. Khan, Sridhar Prabhu, Logan G. Wright, and Peter L. McMahon, arXiv, July 2025 (27 pages).
- [45] [Quantum advantage in learning from experiments](#), by Hsin-Yuan Huang, Michael Broughton, Jordan Cotler, Sitan Chen, Jerry Li, Masoud Mohseni, Hartmut Neven, Ryan Babbush, Richard Kueng, John Preskill, and Jarrod R. McClean, Science, June 2022 (6 pages).
- [46] [A Primer on Quantum Machine Learning](#), by Su Yeon Chang and M. Cerezo, arXiv, November 2025 (45 pages).
- [47] [Quantum sensors that compute: quantum computational magnetic-field sensing using a superconducting qubit](#), by Purnendu Sen, Mathieu Ouellet, Saeed A. Khan, Wayne Wang, Sridhar Prabhu, Alen Senanian, William P. Ban-
ner, William D. Oliver, and Peter L. McMahon, arXiv, August 2026 (26 pages).
- [48] [Imaging Stars at the Quantum Compatibility Limit](#), by Xinyao Guo, Haixing Miao, Zheng Cai, and Huan Yang, arXiv, August 2026 (23 pages).
- [49] [Loading Classical Data into a Quantum Computer](#), by John Cortese et al., arXiv, 2018 (38 pages).
- [50] [Circuit-centric quantum classifiers](#), by Maria Schuld, Alex Bocharov, Krysta Svore, and Nathan Wiebe, arXiv, April 2018 (17 pages).
- [51] [Robust data encodings for quantum classifiers](#), by Ryan LaRose and Brian Coyle, arXiv, March 2020 (18 pages).
- [52] [Towards a Pattern Language for Quantum Algorithms](#), by Frank Leymann, arXiv, June 2019 (12 pages).
- [53] [Quantum linear systems algorithms: a primer](#), by Danial Dervovic, Simone Severini, Leonard Wossnig, et al., arXiv, February 2018 (55 pages).
- [54] [The bitter truth about gate-based quantum algorithms in the NISQ era](#), by Frank Leymann and Johanna Barzen, Quantum Science and Technology, September 2020 (29 pages).
- [55] [Quantum-state preparation with universal gate decompositions](#), by Martin Plesch and Časlav Brukner, Physical Review A, March 2011 📄 (preprint on [arXiv](#)).
- [56] [Adaptive Circuit Learning of Born Machine: Towards Realization of Amplitude Embedding and Data Loading](#), by Chun-Tse Li and Hao-Chung Cheng, arXiv, November 2023 (14 pages).
- [57] [Quantum Resources Required to Block-Encode a Matrix of Classical Data](#), by B. David Clader, William J. Zeng, et al., arXiv, June 2022 (31 pages).
- [58] [Approximate complex amplitude encoding algorithm and its application to data classification problems](#), by Naoki Mitsuda, Naoki Yamamoto, et al., arXiv, May 2024 (13 pages).
- [59] [Variational preparation of entangled states on quantum computers](#), by Vu Tuan Hai, Nguyen Tan Viet, and Le Bin Ho, arXiv, June 2023 (8 pages).
- [60] [Fast Quantum Amplitude Encoding of Typical Classical Data](#), by Vittorio Pagni, Sigurd Huber, Michael Epping, and Michael Felderer, arXiv, March 2025 (8 pages).

- [61] [Limitations of Amplitude Encoding on Quantum Classification](#), by Xin Wang, Yabo Wang, Bo Qi, and Rebing Wu, arXiv, March 2025 (18 pages).
- [62] [AQER: a scalable and efficient data loader for digital quantum computers](#), by Kaining Zhang, Xinbiao Wang, Yuxuan Du, Min-Hsiu Hsieh, and Dacheng Tao, arXiv, February 2026 (45 pages).
- [63] [Qvine: Vine Structured Quantum Circuits for Loading High Dimensional Distributions](#), by David Quiroga, Hannes Leipold, and Bibhas Adhikari, arXiv, April 2026 (11 pages).
- [64] [Random variables, joint distribution functions, and copulas](#), by Abe Sklar, Kybernetika, 1973.
- [65] [Logarithmic-depth quantum state preparation of polynomials](#), by Baptiste Claudon, Alexis Lucas, Jean-Philip Piquemal, César Feniou, and Julien Zylberman, arXiv, March 2026 (21 pages).
- [66] [On Encoding Matrices using Quantum Circuits](#), by Liron Mor Yosef and Haim Avron, arXiv, October 2025 (50 pages).
- [67] [Quantum compilation framework for data loading](#), by Guillermo Alonso-Linaje, Utkarsh Azad, Jay Soni, Jarrett Smalley, Leigh Lapworth, and Juan Miguel Arrazola, arXiv, December 2025 (23 pages).
- [68] [Sparse Quantum State Preparation for Strongly Correlated Systems](#), by César Feniou, O. Adjoua, B. Claudon, Julien Zylberman, E. Giner, and Jean-Philip Piquemal, The Journal of Physical Chemistry Letters, March 2024 (27 pages)  (preprint on HAL).
- [69] [Randomized Quantum Singular Value Transformation](#), by Xinzhao Wang, Shantanav Chakraborty, et al., arXiv, October 2025 (67 pages).
- [70] [Products between block-encodings](#), by Dekuan Dong, Yingzhou Li, and Jungong Xue, arXiv, September 2025 (23 pages).
- [71] [Reducing C-NOT Counts for State Preparation and Block Encoding via Diagonal Matrix Migration](#), by Zexian Li, Guofeng Zhang, and Xiao-Ming Zhang, arXiv, March 2026 (10 pages).
- [72] [Practical block encodings of matrix polynomials that can also be trivially controlled](#), by Martina Nibbi, Filippo Della Chiara, Yizhi Shen, Aaron Szasz, and Roel Van Beeumen, arXiv, January 2026 (23 pages).
- [73] [Data is often loadable in short depth: Quantum circuits from tensor networks for finance, images, fluids, and proteins](#), by Raghav Jumade and Nicolas PD Sawaya, arXiv, December 2023 (10 pages).
- [74] [Scalable Quantum State Preparation for Encoding Genomic Data with Matrix Product States](#), by Floyd M. Creevey, Sergii Strelchuk, et al., arXiv, August 2025 (10 pages).
- [75] [Quantum State Preparation for Medical Data: Comprehensive Methods, Implementation Challenges, and Clinical Prospects](#), by Nikhil Kumar Rajput and Riya Bansal, arXiv, August 2025 (21 pages).
- [76] [Matrix Product State on a Quantum Computer](#), by Yong Liu, Guangyao Huang, Yizhi Wang, and Junjie Wu, arXiv, June 2025 (8 pages).
- [77] [Renormalization-group-based preparation of matrix product states on up to 80 qubits](#), by Moritz Scheer, Daniel Malz, et al., arXiv, October 2025 (8 pages).
- [78] [Simulating quantum circuits using tree tensor networks](#), by Philipp Seitz, Christian B. Mendl, et al., arXiv, March 2023 (13 pages).
- [79] [Simulating quantum circuits using the multi-scale entanglement renormalization ansatz](#), by A. V. Berezutskii, Physical Review Research, January 2025.
- [80] [History and generality of the CS decomposition](#), by C.C. Paige and M. Wei, Linear Algebra and its Applications, September 1994 (24 pages).
- [81] [An Efficient Method for Generating Discrete Random Variables with General Distributions](#), by Alastair J. Walker, ACM Transactions on Mathematical Software (TOMS), September 1977 (4 pages).
- [82] [Logical Resource Estimation for Quantum State Preparation with Compilation](#), by Diyi Liu, Hanyu Wang, Shuchen Zhu, Jason Cong, Wibe A. de Jong, Di Fang, Zhen Huang, Costin Iancu, and Chao Yang, arXiv, May 2026 (24 pages).
- [83] [Trading T gates for dirty qubits in state preparation and unitary synthesis](#), by Guang Hao Low, Vadym Kliuchnikov, and Luke Schaeffer, Quantum, June 2024 (26 pages).
- [84] [Encoding Electronic Spectra in Quantum Circuits with Linear T Complexity](#), by Ryan Babbush, Craig Gidney, Dominic W. Berry, Nathan Wiebe, Jarrod McClean, Alexandru Paler, Austin Fowler, and Hartmut Neven, Physical Review X, October 2018 (36 pages)  (preprint on arXiv).
- [85] [Constant-Depth Quantum Circuits for Arbitrary Quantum State Preparation via Measurement and Feedback](#), by Wei Zi, Junhong Nie, and Xiaoming Sun, arXiv, March 2025 (20 pages).
- [86] [Efficient Quantum State Preparation with Walsh Series](#), by Julien Zylberman and Fabrice Debbaesch, arXiv, November 2023 (22 pages).
- [87] [Efficient Quantum Circuits for Non-Unitary and Unitary Diagonal Operators with Space-Time-Accuracy Trade-Offs](#), by Julien Zylberman, Ugo Nzongani, Andrea Simonetto, and Fabrice Debbaesch, ACM Transactions on Quantum Computing, May 2025.
- [88] [Trainability and Expressivity of Hamming-Weight Preserving Quantum Circuits for Machine Learning](#), by Léo Monbroussou, Jonas Landman, Elham Kashefi, et al., arXiv, May 2025 (48 pages).
- [89] [GASP – A Genetic Algorithm for State Preparation](#), by Floyd M. Creevey, Charles D. Hill, and Lloyd C. L. Hollenberg, arXiv, February 2023 (8 pages).
- [90] [Quantum Evolution Kernel: Machine Learning on Graphs with Programmable Arrays of Qubits](#), by Louis-Paul Henry, Slimane Thabet, Constantin Dalyac, and Loïc Henriët, Physical Review A, September 2021 (12 pages)  (preprint on arXiv).
- [91] [Quantum complexity in gravity, quantum field theory, and quantum information science](#), by Stefano Baiguera, Nicole Yunger Halpern, et al., arXiv, May 2025 (119 pages).
- [92] [Representation of Classical Data on Quantum Computers](#), by Thomas Lang, Theobald OJ Fuchs, et al., arXiv, December 2024 (16 pages).
- [93] [Quantum Fourier Networks for Solving Parametric PDEs](#), by Nishant Jain, Jonas Landman, Natansh Mathur, and Iordanis Kerenidis, arXiv, June 2023 (17 pages).
- [94] [EHands: Quantum Protocol for Polynomial Computation on Real-Valued Encoded States](#), by Jan Balewski, C. Pestano, Mercy G. Amankwah, E. Wes Bethel, Talita Perciano, and Roel Van Beeumen, arXiv, December 2025 (17 pages).
- [95] [NNQA: Neural-Native Quantum Arithmetic for End-to-End Polynomial Synthesis](#), by Ziqing Guo, Jie Li, Yong Chen, and Ziwen Pan, arXiv, March 2026 (15 pages).
- [96] [Universal Approximation of Continuous Functions with Minimal Quantum Circuits](#), by Adrián Pérez-Salinas, Mahtab Yaghubi Rad, Alice Barthe, and Vedran Dunjko, Physical Review Research, November 2025.
- [97] [The Cost of Removing Tunability in Quantum Data Reuploading](#), by Anthony Yuezhong Liu and Lirandé Pira, arXiv, June 2026 (30 pages).
- [98] [Perfect quantum state transfer via state restoring and ancilla measurement](#), by E.B. Fel'dman, J. Wu, and A.I. Zenchuk, arXiv, September 2025 (41 pages).
- [99] [Continuous-Time Quantum State Transfer with a Generalized Laplacian](#), by Yujia Shi, arXiv, September 2025 (7 pages).
- [100] [A Review of Perfect Quantum State Transfer, from One to Two and Three Dimensional Arrays of Qubits](#), by Marzieh

Asoudeh and Vahid Karimipour, International Journal of Theoretical Physics, July 2025 📄 (preprint on arXiv).

- [101] [Inverse designed Hamiltonians for perfect state transfer and remote entanglement generation, and applications in superconducting qubits](#), by Tian-Le Wang, Guo-Ping Guo, et al., arXiv, October 2025 (18 pages).
- [102] [Robustness-Runtime Tradeoff for Quantum State Transfer](#), by Twesh Upadhyaya, Yifan Hong, T. C. Mooney, and Alexey V. Gorshkov, arXiv, February 2026 (16 pages).
- [103] [Benchmarking quantum state transfer on quantum devices](#), by Yi-Te Huang, Physical Review Research, April 2021.
- [104] [Enhanced Algorithmic Perfect State Transfer on IBM Quantum Computers](#), by Zong-Yuan Ge, Lian-Ao Wu, and Zhao-Ming Wang, arXiv, August 2025 (14 pages).
- [105] [How Far Can You Do Nothing On a Quantum Computer?](#), by Nitay Mayo, Tal Mor, and Aryeh Lev Zabokritskiy, arXiv, August 2026 (16 pages).
- [106] [Inverse Problems, Constraint Satisfaction, Reversible Logic, Invertible Logic and Grover Quantum Oracles for Practical Problems](#), by Marek Perkowski, May 2020 (62 slides).
- [107] [Circuit complexity of quantum access models for encoding classical data](#), by Xiao-Ming Zhang and Xiao Yuan, arXiv, April 2024 (21 pages).
- [108] [Low Depth Phase Oracle Using a Parallel Piecewise Circuit](#), by Zhu Sun, Nikitas Stamatopoulos, et al., arXiv, October 2024 (14 pages).
- [109] [Quantum advantage for differential equation analysis](#), by Bobak T. Kiani, Dirk Englund, Seth Lloyd, et al., arXiv, April 2022 (21 pages).
- [110] [Learning to Measure: Adaptive Informationally Complete Generalized Measurements for Quantum Algorithms](#), by Guillermo García-Pérez, Ivano Tavernelli, Sabrina Maniscalco, et al., arXiv, December 2021 (18 pages).
- [111] [Nearly Optimal Quantum Algorithm for Estimating Multiple Expectation Values](#), by William J. Huggins, Nathan Wiebe, Ryan Babbush, et al., arXiv, October 2022 (18 pages).
- [112] [Single-shot quantum machine learning](#), by Erik Recio-Armengol, Jens Eisert, and Johannes Jakob Meyer, arXiv, June 2024 (6 pages).
- [113] [Phase Kickback](#), by Eduard Smetanin, November 2019 (4 pages).
- [114] [A clever quantum trick](#), by Emilio Peláez, January 2021.
- [115] [Quantum Phase Kickback - What I told you was true...from a certain point of view](#), by Frank Zickert, March 2021.
- [116] [Everything You Always Wanted to Know About Quantum Circuits](#), by Edgard Muñoz-Coreas and Himanshu Thapliyal, arXiv, August 2022 (18 pages).
- [117] [A Comprehensive Study of Quantum Arithmetic Circuits](#), by Siyi Wang, Anupam Chattopadhyay, et al., arXiv, June 2024 (35 pages).
- [118] [A new quantum ripple-carry addition circuit](#), by Steven A. Cuccaro, Thomas G. Draper, Samuel A. Kutin, and David Petrie Moulton, arXiv, October 2004 (9 pages).
- [119] [A Higher radix architecture for quantum carry-lookahead adder](#), by Siyi Wang, Anubhab Baksi, and Anupam Chattopadhyay, Scientific Reports, September 2023.
- [120] [High Performance Quantum Modular Multipliers](#), by Rich Rines and Isaac Chuang, arXiv, January 2018 (48 pages).
- [121] [Arithmetic on Quantum Computers: Multiplication](#), by Sashwat Anagolum, December 2018.
- [122] [Resource Estimation of Quantum Multiplication Algorithms](#), by Ethan R. Hansen, Sanskriti Joshi, and Hannah Rarick, arXiv, February 2024 (4 pages).
- [123] [Efficient Floating Point Arithmetic for Quantum Computers](#), by Raphael Seidel, Nikolay Tcholchev, Sebastian Bock, Colin Kai-Uwe Becker, and Manfred Hauswirth, arXiv, December 2021 (15 pages).
- [124] [Efficient Floating-Point Arithmetic on Fault-Tolerant Quantum Computers](#), by José E. Cruz Serrallés, Riccardo Lattanzi, et al., arXiv, October 2025 (14 pages).
- [125] [Hybrid quantum floating-point method for sharp arithmetic](#), by Gabriele Agliardi and Enrico Prati, arXiv, July 2026 (37 pages).
- [126] [Quantum circuits for floating-point arithmetic](#), by Thomas Häner, Mathias Soeken, Martin Roetteler, and Krysta M. Svore, arXiv, July 2018 (13 pages).
- [127] [QRTlib: A Library for Fast Quantum Real Transforms](#), by Armin Ahmadkhaniha, Lu Chen, Jake Doliskani, and Zhifu Sun, arXiv, October 2025 (13 pages).
- [128] [Quantum Subroutine for Efficient Matrix Multiplication](#), by Anna Bernasconi, IEEE, August 2024 (11 pages).
- [129] [Quantum computational complexity of matrix functions](#), by Santiago Cifuentes, Leandro Aolita, et al., arXiv, April 2025 (30 pages).
- [130] [Quantum Networks for Elementary Arithmetic Operations](#), by V. Vedral, A. Barenco, and A. Ekert, arXiv, November 1995 (7 pages).
- [131] [Quantum Addition Circuits and Unbounded Fan-Out](#), by Yasuhiro Takahashi, Seiichiro Tani, and Noboru Kunihiro, arXiv, October 2009 (17 pages).
- [132] [Quantum adders: on the structural link between the ripple-carry and carry-lookahead techniques](#), by Maxime Remaud, arXiv, October 2025 (11 pages).
- [133] [Quantum Non-Modular Multiplication with QFT-Based Multi Input Parallelized Adder](#), by Murat Kurtand Selçuk Çakmak and Azmi Gençten, arXiv, April 2026 (9 pages).
- [134] [Universal Matrix Multiplication on Quantum Computer](#), by Jiaqi Yao, Tianjian Huang, and Ding Liu, arXiv, May 2025 (21 pages).
- [135] [Quantum algorithms for matrix multiplication](#), by François Le Gall, 2012 (2 pages).
- [136] [Quantum Arithmetic-based on Quantum Signal Processing](#), by Robin Ollive and Stephane Louise, arXiv, March 2025 (28 pages).
- [137] [Optimal Toffoli-Depth Quantum Adder](#), by Siyi Wang, Suman Deb, Ankit Mondal, and Anupam Chattopadhyay, arXiv, May 2024 (16 pages).
- [138] [Fast quantum integer multiplication with zero ancillas](#), by Gregory D. Kahanamoku-Meyer and Norman Y. Yao, arXiv, November 2024 (10 pages).
- [139] [Ancilla-free Quantum Adder with Sublinear Depth](#), by Maxime Remaud and Vivien Vandaele, arXiv, January 2025 (22 pages).
- [140] [Improved T counts and active volume estimates for high-level arithmetic subroutines](#), by Sam Heavey, arXiv, April 2025 (20 pages).
- [141] [A Polylogarithmic-Depth Quantum Multiplier](#), by Fred Sun and Anton Borissov, arXiv, April 2026 (9 pages).
- [142] [A Modular and T-Gate Efficient Architecture for Quantum Leading-Zero/One Counter](#), by Lei-Han Yao, Shang-Wei Lin, Yu-Chung Chen, and Yean-Ru Chen, arXiv, April 2026 (11 pages).
- [143] [A Classical-Quantum Adder with Constant Workspace and Linear Gates](#), by Craig Gidney, arXiv, July 2025 (8 pages).
- [144] [A Feasible Design of Elementary Quantum Arithmetic Logic Units for Near-Term Quantum Computers](#), by Junxu Li, arXiv, September 2024 (29 pages).
- [145] [4-bit Factorization Circuit Composed of Multiplier Units with Superconducting Flux Qubits toward Quantum Annealing](#), by Daisuke Saida, Mutsuo Hidaka, and Yuki Yamanashi, arXiv, August 2023 (9 pages).
- [146] [Logical Reversibility of Computation](#), by Charles H. Bennett, IBM Journal of Research and Development, November 1973 (8 pages) 📄.
- [147] [Quantum Algorithms Revisited](#), by Richard Cleve, Artur Ekert, Chiara Macchiavello, and Michele Mosca, Proceed-

- ings of the Royal Society of London A, January 1998 (16 pages).
- [148] [Strengths and Weaknesses of Quantum Computing](#), by Charles H. Bennett, Ethan Bernstein, Gilles Brassard, and Umesh Vazirani, SIAM Journal on Computing, October 1997 (14 pages).
 - [149] [How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits](#), by Craig Gidney and Martin Ekerå, arXiv, April 2021 (31 pages).
 - [150] [Efficient realization of quantum primitives for Shor’s algorithm using PennyLane library](#), by A.V. Antipov, E.O. Kiktenko, and A.K. Fedorov, arXiv, July 2022 (20 pages).
 - [151] [Revisiting Shor’s quantum algorithm for computing general discrete logarithms](#), by Martin Ekerå, arXiv, May 2019 (32 pages).
 - [152] [An approximate Fourier transform useful in quantum factoring](#), by Don Coppersmith, arXiv, December 1994 (8 pages).
 - [153] [Quantum circuit for the fast Fourier transform](#), by Ryo Asaka, 2020 (20 pages).
 - [154] [Approximate quantum Fourier transform with \$O\(n \log\(n\)\)\$ T gates](#), by Yunseong Nam, Yuan Su, and Dmitri Maslov, npj Quantum Information, March 2020.
 - [155] [Quantum Fourier Transform Has Small Entanglement](#), by Jielun Chen, E.M. Stoudenmire, and Steven R. White, arXiv, October 2023 (30 pages).
 - [156] [Number of bits returned by a quantum estimation](#), by Xi Lu, Lorenzo Maccone, et al., arXiv, September 2024 (10 pages).
 - [157] [An assessment of quantum phase estimation protocols for early fault-tolerant quantum computers](#), by Jacob S. Nelson and Andrew D. Baczewski, arXiv, February 2024 (14 pages).
 - [158] [Quantum algorithm for linear systems of equations](#), by Aram W. Harrow, Avinandan Hassidim, and Seth Lloyd, Physical Review Letters, October 2009 (4 pages).
 - [159] [Read the fine print](#), by Scott Aaronson, Nature Physics, 2015 (3 pages).
 - [160] [A CS guide to the quantum singular value transformation](#), by Ewin Tang and Kevin Tian, arXiv, October 2023 (32 pages).
 - [161] [A Quantum Approximate Optimization Algorithm](#), by Edward Farhi, Jeffrey Goldstone, and Sam Gutmann, arXiv, November 2014 (16 pages).
 - [162] [A variational eigenvalue solver on a photonic quantum processor](#), by Alberto Peruzzo, Jarrod McClean, Peter Shadbolt, Man-Hong Yung, Xiao-Qi Zhou, Peter J. Love, Alán Aspuru-Guzik, and Jeremy L. O’Brien, Nature Communications, July 2014 (7 pages).
 - [163] [The theory of variational hybrid quantum-classical algorithms](#), by Jarrod R. McClean, Jonathan Romero, Ryan Babbush, and Alán Aspuru-Guzik, New Journal of Physics, February 2016 (20 pages).
 - [164] [Supervised learning with quantum enhanced feature spaces](#), by Vojtech Havlicek, Jerry M. Chow, Jay M. Gambetta, et al., arXiv, June 2018 (22 pages).
 - [165] [Synthesis of discrete–continuous quantum circuits with multimodal diffusion models](#), by Florian Furrutter, Zohim Chandani, Ikko Hamamura, Hans J Briegel, and Gorka Muñoz-Gil, Machine Learning: Science and Technology, April 2026 (16 pages).
 - [166] [Efficient decomposition methods for controlled- \$R_n\$ using a single ancillary qubit](#), by Taewan Kim et Byung-Soo Choi, 2018 (7 pages).
 - [167] [The exact lower bound of CNOT-complexity for fault-tolerant quantum Fourier transform](#), by Qiqing Xia, Huiqin Xie, and Li Yang, arXiv, September 2024 (20 pages).
 - [168] [A log-depth in-place quantum Fourier transform that rarely needs ancillas](#), by Gregory D. Kahanamoku-Meyer, John Blue, Thiago Bergamaschi, Craig Gidney, and Isaac L. Chuang, arXiv, May 2025 (11 pages).
 - [169] [Towards Deploying Optimistic Quantum Fourier Transforms: An Architecture-Algorithm Co-Design Study](#), by Pedro L. S. Lopes, arXiv, May 2026 (12 pages).
 - [170] [Optimal ancilla-free Clifford+T approximation of z-rotations](#), by Neil J. Ross and Peter Selinger, arXiv, June 2016 (40 pages).
 - [171] [Efficient classical simulation of the semi-classical Quantum Fourier Transform](#), by Daniel E. Browne, arXiv, December 2006 (4 pages).
 - [172] [De-quantisation of the Quantum Fourier Transform](#), by Alastair A. Abbott, arXiv, January 2020 (14 pages).
 - [173] [Semiclassical Fourier Transform for Quantum Computation](#), by Robert B. Griffiths and Chi-Sheng Niu, Physical Review Letters, April 1996 (7 pages) (preprint on arXiv).
 - [174] [Quantum Fourier Transform using Dynamic Circuits](#), by Elisa Bäumer, Daniel Lidar, Derek S. Wang, et al., arXiv, March 2024 (4 pages).
 - [175] [Exact quantum Fourier transforms and discrete logarithm algorithms](#), by Michele Mosca and Christof Zalka, International Journal of Quantum Information, March 2004 (10 pages) (preprint on arXiv).
 - [176] [Non-Uniform Quantum Fourier Transform](#), by Junaid Aftab, Yuehaw Khoo, and Haizhao Yang, arXiv, March 2026 (37 pages).
 - [177] [The Symmetric Group Defies Strong Fourier Sampling](#), by Cristopher Moore, Alexander Russell, and Leonard J. Schulman, SIAM Journal on Computing, January 2008 (23 pages) (preprint on arXiv).
 - [178] [Limitations of Quantum Coset States for Graph Isomorphism](#), by Sean Hallgren, Cristopher Moore, Martin Rötteler, Alexander Russell, and Pranab Sen, Journal of the ACM, November 2010 (33 pages) (preprint on arXiv).
 - [179] [A Subexponential-Time Quantum Algorithm for the Dihedral Hidden Subgroup Problem](#), by Greg Kuperberg, SIAM Journal on Computing, July 2005 (19 pages) (preprint on arXiv).
 - [180] [Benchmarking logical three-qubit quantum Fourier transform encoded in the Steane code on a trapped-ion quantum computer](#), by Karl Mayer, Russell Stutz, et al., arXiv, April 2024 (13 pages).
 - [181] [Demonstrating Record Fidelity for the Quantum Fourier Transform](#), by Philipp Aumann, Michael Fellner, David Alber, Max Cykier, Christoph Fleckenstein, Roeland ter Hoeven, Leo Stenzel, Riccardo J. Valencia-Tortora, and Wolfgang Lechner, arXiv, April 2026 (6 pages).
 - [182] [Experimental demonstration of the Quantum Fourier Transform on up to 100 qubits using a convolutional compilation strategy](#), by Paul Coote, Michael J. Biercuk, and Yuval Baum, arXiv, August 2026 (9 pages).
 - [183] [Towards an Optimally Distributed Quantum Fourier Transform Circuit](#), by Zachary Vernec, Michael Silver, and Hans-Arno Jacobsen, arXiv, June 2026 (22 pages).
 - [184] [A Survey of Quantum Alternatives to Randomized Algorithms: Monte Carlo Integration and Beyond](#), by Philip Intallura, Jakub Marecek, et al., arXiv, March 2023 (22 pages).
 - [185] [Quantum Amplitude Amplification and Estimation](#), by Gilles Brassard, Peter Høyer, Michele Mosca, and Alain Tapp, Contemporary Mathematics, January 2002 (22 pages) (preprint on arXiv).
 - [186] [Non-Boolean Quantum Amplitude Amplification and Quantum Mean Estimation](#), by Prasanthy Shyamsundar, arXiv, February 2021 (36 pages).
 - [187] [Hamiltonian Simulation by Qubitization](#), by Guang Hao Low and Isaac L. Chuang, Quantum, July 2019.
 - [188] [Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics](#), by András Gilyén, Yuan Su, Guang Hao Low, and Nathan Wiebe, Proceedings of the 51st Annual ACM SIGACT Sym-

- posium on Theory of Computing (STOC 2019), June 2019 (12 pages).
- [189] [A Grand Unification of Quantum Algorithms](#), by John M. Martyn, Zane M. Rossi, Andrew K. Tan, and Isaac L. Chuang, arXiv, December 2021 (39 pages).
 - [190] [The Complexity of the Local Hamiltonian Problem](#), by Julia Kempe, Alexei Kitaev, and Oded Regev, SIAM Journal on Computing, May 2006 (28 pages)  (preprint on [arXiv](#)).
 - [191] [Quantum Algorithm for the Direct Calculations of Vertical Ionization Energies](#), by Kenji Sugisaki, March 2021 (6 pages).
 - [192] [Initial state preparation for quantum chemistry on quantum computers](#), by Stepan Fomichev, Juan Miguel Arrazola, et al., arXiv, February 2024 (30 pages).
 - [193] [Tight Bounds for Quantum Phase Estimation and Related Problems](#), by Nikhil S. Mande and Ronald de Wolf, Quantum, June 2026 (18 pages).
 - [194] [Rapid initial state preparation for the quantum simulation of strongly correlated molecules](#), by Dominic W. Berry, Garnet Kin-Lic Chan, Ryan Babbush, Nicholas C. Rubin, et al., arXiv, September 2024 (47 pages).
 - [195] [Quantum State Preparation Of Multiconfigurational States For Quantum Chemistry](#), by Gabriel Greene-Diniz, Georgia Prokopiou, David Zsolt Manrique, and David Muñoz Ramo, arXiv, March 2026 (22 pages).
 - [196] [VQE as Initial State Preparation for QPE on Heisenberg Spin-Glass Hamiltonians](#), by Elijah Pelofske and Stephan Eidenbenz, arXiv, June 2026 (11 pages).
 - [197] [How to perform the most accurate possible phase measurements](#), by Dominic W. Berry, Brendon L. Higgins, Stephen D. Bartlett, Morgan W. Mitchell, Geoff J. Pryde, and Howard M. Wiseman, Physical Review A, November 2009 (22 pages).
 - [198] [Tensor hypercontraction density fitting. I. Quartic scaling second- and third-order Møller-Plesset perturbation theory](#), by Edward G. Hohenstein, Robert M. Parrish, and Todd J. Martínez, The journal of Chemical Physics, 2012 .
 - [199] [Heisenberg-limited ground state energy estimation for early fault-tolerant quantum computers](#), by Lin Lin and Yu Tong, arXiv, February 2022 (31 pages).
 - [200] [Quantum phase estimation with optimal confidence interval using three control qubits](#), by Kaur Kristjuhan and Dominic W. Berry, arXiv, January 2026 (31 pages).
 - [201] [Statistical Quantum Phase Estimation: Extensions and Practical Considerations](#), by Amit Surana and Brandon Allen, arXiv, May 2026 (9 pages).
 - [202] [Refining Quantum Phase Estimation Precision Conditions on Unitaries for Many-Electron Systems](#), by Jérémie Messud and Wassil Sennane, arXiv, April 2026 (7 pages).
 - [203] [Improving the runtime of quantum phase estimation for chemistry through basis set optimization](#), by Pauline J. Ollitrault, Michael Streif, et al., arXiv, September 2025 (32 pages).
 - [204] [Block encoding with low gate count for second-quantized Hamiltonians](#), by Diyi Liu, Chao Yang, et al., arXiv, October 2025 (32 pages).
 - [205] [Benchmarking Quantum Simulation Methods](#), by Calvin Ku, Yu-Cheng Chen, Alice Hu, and Min-Hsiu Hsieh, arXiv, October 2025 (49 pages).
 - [206] [Hamiltonian Simulation and Linear Combination of Unitary Decomposition of Structured Matrices](#), by Robin Ollive and Stéphane Louise, arXiv, March 2026 (13 pages).
 - [207] [Fault-tolerant quantum algorithms for quantum molecular systems: A survey](#), by Yukun Zhang, Xiao Yuan, et al., arXiv, February 2025 (28 pages).
 - [208] [Hamiltonian dynamics simulation using linear combination of unitaries on an ion trap quantum computer](#), by Michelle Wynne Sze, Nathan Fitzpatrick, et al., arXiv, February 2025 (22 pages).
 - [209] [Tradeoffs between quantum and classical resources in linear combination of unitaries](#), by Kaito Wada, Suguru Endo, et al., arXiv, December 2025 (25 pages).
 - [210] [Efficient Fourier-Based Linear Combination of Unitaries and Applications in Quantum Optimization](#), by Almudena Carrera Vazquez, Daniel J. Egger, and Stefan Woerner, arXiv, May 2026 (18 pages).
 - [211] [Dividing and Conquering the Van Vleck Catastrophe](#), by Sophia Simon, Nathan Wiebe, et al., arXiv, April 2025 (21 pages).
 - [212] [Finding Prime Factors of Integer using Quantum Computer](#), by Saptashwa Bhattacharyya, June 2022.
 - [213] [Quantum Simulations of Chemistry in First Quantization with any Basis Set](#), by Timothy N. Georges, Aleksei V. Ivanov, et al., arXiv, March 2025 (25 pages).
 - [214] [Improved Techniques for Preparing Eigenstates of Fermionic Hamiltonians](#), by Dominic W. Berry, Nathan Wiebe, Craig Gidney, Ryan Babbush, et al., arXiv, March 2018 (16 pages).
 - [215] [Spectral amplification for ground-state energy estimation of electronic structure in first quantization](#), by Alicja Dutkiewicz, Alec F. White, Guang Hao Low, A. Eugene De-Prince III, Matthew P. Harrigan, Marika Kieferova, Ryan Babbush, Dominic W. Berry, and Nicholas C. Rubin, arXiv, July 2026 (38 pages).
 - [216] [Adiabatic ground state preparation of fermionic many-body systems from a two-body perspective](#), by Dyon van Vreumingen and Kareljan Schoutens, arXiv, July 2023 (14 pages).
 - [217] [Iterative phase estimation](#), by Caleb J O’Loan, arXiv, January 2010 (17 pages).
 - [218] [Iterative quantum-phase-estimation protocol for shallow circuits](#), by Joseph G. Smith et al., Physical Review A, December 2022  (preprint on [repository.cam.ac.uk](#)).
 - [219] [Real-Space Chemistry on Quantum Computers: A Fault-Tolerant Algorithm with Adaptive Grids and Transcorrelated Extension](#), by César Fenieu, Jean-Philip Piquemal, Emmanuel Giner, et al., arXiv, July 2025 (19 pages).
 - [220] [Adiabatic Quantum Phase Estimation](#), by Alexander Schmidhuber and Seth Lloyd, arXiv, May 2026 (11 pages).
 - [221] [Programmable Signal Design for Quantum Phase Estimation via Quantum Signal Processing](#), by Zikang Jia, Suying Liu, and Yulong Dong, arXiv, April 2026 (23 pages).
 - [222] [Quantum simulation of chemistry via quantum fast multipole method](#), by Dominic W. Berry, Ryan Babbush, et al., arXiv, October 2025 (23 pages).
 - [223] [On low-depth algorithms for quantum phase estimation](#), by Hongkang Ni, Haoya Li, and Lexing Ying, arXiv, October 2023 (12 pages).
 - [224] [Even shorter quantum circuit for phase estimation on early fault-tolerant quantum computers with applications to ground-state energy estimation](#), by Zhiyan Ding and Lin Lin, arXiv, March 2024 (48 pages).
 - [225] [Quantum phase estimation of multiple eigenvalues for small-scale \(noisy\) experiments](#), by T.E. O’Brien, B. Tarasinski, and B.M. Terhal, arXiv, December 2018 (29 pages).
 - [226] [An adaptive Bayesian quantum algorithm for phase estimation](#), by Joseph G. Smith, Crispin H. W. Barnes, and David R. M. Arvidsson-Shukur, arXiv, March 2023 (7 pages).
 - [227] [Deterministic Quantum Phase Estimation with Linear Circuit Complexity in a Photonic System](#), by M. Midhuna, Ajay Jayachandran, Kanad Sengupta, Akshai T. Krishnan, and C. M. Chandrashekar, arXiv, July 2026 (11 pages).
 - [228] [Elucidating Reaction Mechanisms on Quantum Computers](#), by Markus Reiher, Nathan Wiebe, Krysta M Svore, Dave Wecker, and Matthias Troyer, arXiv, May 2016 (28 pages).
 - [229] [Qubitization of Arbitrary Basis Quantum Chemistry Leveraging Sparsity and Low Rank Factorization](#), by Dominic W. Berry, Craig Gidney, Mario Motta, Jarrod R. McClean, and Ryan Babbush, arXiv, November 2019 (44 pages).

- [230] [Even More Efficient Quantum Computations of Chemistry Through Tensor Hypercontraction](#), by Joonho Lee et al., PRX Quantum, July 2021 (62 pages).
- [231] [Fast Quantum Simulation of Electronic Structure by Spectral Amplification](#), by Guang Hao Low, Robbie King, Dominic W. Berry, Qiushi Han, A. Eugene DePrince III, Alec F. White, Ryan Babbush, Rolando D. Somma, and Nicholas C. Rubin, Physical Review X, October 2025 (41 pages).
- [232] [Rapid Dissipative Ground State Preparation at Chemical Transition States](#), by Thomas W. Watts, Soumya Sarkar, Daniel Collins, Nam Nguyen, Luke Quezada, Michael J. Bremner, and Samuel J. Elman, arXiv, March 2026 (13 pages).
- [233] [Classical solution of the FeMo-cofactor model to chemical accuracy and its implications](#), by Huanchen Zhai, Chenghan Li, Xing Zhang, Zhendong Li, Seunghoon Lee, and Garnet Kin-Lic Chan, arXiv, January 2026 (89 pages).
- [234] [Determining eigenstates and thermal states on a quantum computer using quantum imaginary time evolution](#), by Mario Motta, Chong Sun, Adrian T. K. Tan, Matthew J. O'Rourke, Erika Ye, Austin J. Minnich, Fernando G. S. L. Brandão, and Garnet Kin-Lic Chan, Nature Physics, February 2020 (6 pages)  (preprint on [arXiv](#)).
- [235] [Adaptive time Compressed QITE \(ACQ\) and its geometrical interpretation](#), by Alberto Acevedo Meléndez, Andreu Anglés-Castillo, et al., arXiv, October 2025 (24 pages).
- [236] [An Analytic Theory of Quantum Imaginary Time Evolution](#), by Min Chen, Bingzhi Zhang, Quntao Zhuang, and Junyu Liu, arXiv, October 2025 (35 pages).
- [237] [An Adaptive Weighted QITE-VQE Algorithm for Combinatorial Optimization Problems](#), by Ningyi Xie, Xinwei Lee, Tiejun Chen, Yoshiyuki Saito, Nobuyoshi Asai, and Dongsheng Cai, arXiv, April 2025 (10 pages).
- [238] [Post-Variational Ground State Estimation via QPE-Based Quantum Imaginary Time Evolution](#), by Nora Bauer and George Siopsis, arXiv, April 2025 (6 pages).
- [239] [Understanding Quantum Imaginary Time Evolution and its Variational form](#), by Anglés-Castillo Andreu, Garcia-March Miguel Angel, et al., arXiv, October 2025 (10 pages).
- [240] [Multiple-time Quantum Imaginary Time Evolution](#), by Julio Del Castillo, Mats Granath, and Evert van Nieuwenburg, arXiv, June 2026 (13 pages).
- [241] [Optimization by VarQITE on Adaptive Variational Quantum Kolmogorov-Arnold Network](#), by Hikaru Wakaura, Rahmat Mulyawan, and Andriyan B. Suksmono, arXiv, June 2025 (9 pages).
- [242] [Double-bracket quantum algorithms for quantum imaginary-time evolution](#), by Marek Gluza, Zoë Holmes, Nelly H. Y. Ng, et al., arXiv, July 2025 (37 pages).
- [243] [Dynamical systems that sort lists, diagonalize matrices, and solve linear programming problems](#), by R.W. Brockett, Linear Algebra and its Applications, February 1991 (13 pages).
- [244] [Bowtie VarQTE: A Resource-Efficient Quantum State Preparation Primitive](#), by Marc Drudis, Alberto Baiardi, Mattia Chiurco, Francesco Tacchino, Stefan Woerner, and Christa Zoufal, arXiv, May 2026 (19 pages).
- [245] [On the generalized eigenvalue problem in subspace-based excited state methods for quantum computers](#), by Prince Frederick Kwao, Ayush Asthana, et al., arXiv, March 2025 (9 pages).
- [246] [Amplitude estimation without phase estimation](#), by Yohichi Suzuki, Naoki Yamamoto, et al., arXiv, January 2020 (13 pages).
- [247] [Accelerated Quantum Amplitude Estimation without QFT](#), by Alet Roux and Tomasz Zastawniak, arXiv, July 2024 (25 pages).
- [248] [Bayesian Quantum Amplitude Estimation](#), by Alexandra Ramôa and Luis Paulo Santos, arXiv, July 2025 (40 pages).
- [249] [Quantum amplitude estimation from classical signal processing](#), by Farrokh Labib, William J. Zeng, et al., arXiv, May 2025 (11 pages).
- [250] [Optimal Hamiltonian Simulation by Quantum Signal Processing](#), by Guang Hao Low and Isaac L. Chuang, Physical Review Letters, January 2017.
- [251] [Realization of quantum signal processing on a noisy quantum computer](#), by Yuta Kikuchi, Conor Mc Keever, Luuk Coopmans, Michael Lubasch, and Marcello Benedetti, arXiv, September 2023 (16 pages).
- [252] [Quantum Signal Processing for Linear PDEs: Circuit Design and Experimental Validation](#), by Hyeonjin Kim, Revathi Jambunathan, Jan Balewski, and Daan Camps, arXiv, May 2026 (20 pages).
- [253] [Complementary polynomials in quantum signal processing](#), by Bjorn K. Berntson and Christoph Sünderhauf, arXiv, June 2025 (29 pages).
- [254] [Toward Mixed Analog-Digital Quantum Signal Processing: Quantum AD/DA Conversion and the Fourier Transform](#), by Yuan Liu, John M. Martyn, Jasmine Sinanan-Singh, Kevin C. Smith, Steven M. Girvin, and Isaac L. Chuang, arXiv, May 2025 (16 pages).
- [255] [Quantum computational sensing using quantum signal processing, quantum neural networks, and Hamiltonian engineering](#), by Saeed A. Khan, Sridhar Prabhu, Logan G. Wright, and Peter L. McMahon, arXiv, July 2025 (39 pages).
- [256] [Generalized Quantum Signal Processing](#), by Danial Motlagh and Nathan Wiebe, arXiv, January 2024 (21 pages).
- [257] [From Block-encoding to Generalized Quantum Signal Processing: Principles, Algorithms and Applications](#), by Tal Gurfinkel, Kaushika De Silva, Anuradha Mahasinghe, Jens Renders, Jack Blyth, James Greenwell, Archie Butterworth, Yusen Wu, Lyle Noakes, Miloud Bessafi, Frederic Cadet, and Jingbo Wang, arXiv, September 2026 (45 pages).
- [258] [A quantum approach for digital signal processing](#), by Alok Shukla and Prakash Vedula, arXiv, September 2023 (29 pages).
- [259] [Quantum Signal Processing and Quantum Singular Value Transformation on \$U\(N\)\$](#) , by Xi Lu, Yuan Liu, and Hongwei Lin, arXiv, March 2026 (19 pages).
- [260] [Multivariable QSP and Bosonic Quantum Simulation using Iterated Quantum Signal Processing](#), by Niladri Gomes, Hokiat Lim, and Nathan Wiebe, arXiv, August 2024 (26 pages).
- [261] [Analytical Angle-Finding and Series Expansions for Quantum Signal Processing via Orthogonal Polynomial Theory](#), by Pierre-Antoine Bernard and Nathan Wiebe, arXiv, May 2026 (49 pages).
- [262] [Analytic Approach to Quantum Control Using Quantum Signal Processing](#), by Aishwarya Majumdar, John M. Martyn, Yuan Liu, and Nathan Wiebe, arXiv, June 2026 (15 pages).
- [263] [Forrelation: A Problem that Optimally Separates Quantum from Classical Computing](#), by Scott Aaronson and Andris Ambainis, arXiv, November 2014 (60 pages).
- [264] [Extending Forrelation: Quantum Algorithms Related to Generalized Fourier-Correlation](#), by Suman Dutta, Subhamoy Maitra, and Pantelimon Stanica, arXiv, July 2025 (21 pages).
- [265] [IQP circuits for 2-Forrelation](#), by Quentin Buzet and André Chailloux, arXiv, April 2026 (27 pages).
- [266] [Quantum Linear System Solvers: A Survey of Algorithms and Applications](#), by Mauro E. S. Morales, Alán Aspuru-Guzik, Lin Lin, Patrick Rebentrost, Dominic W. Berry, et al., arXiv, January 2025 (42 pages).
- [267] [An Enhanced Hybrid HHL Algorithm](#), by Jack Morgan, Eric Ghysels, and Hamed Mohammadbagherpoor, arXiv, November 2024 (27 pages).

- [268] [Matrix Inversion by Quantum Walk](#), by Alastair Kay and Christino Tamon, arXiv, August 2025 (7 pages).
- [269] [Advanced Quantum Poisson Solver in the NISQ era](#), by Walter Robson, Jaroslaw Nabrzyski, et al., arXiv, September 2022 (4 pages).
- [270] [Identifying Bottlenecks of NISQ-friendly HHL algorithms](#), by Marc Andreu Marfany, Jeanette Miriam Lorenz, et al., arXiv, August 2024 (10 pages).
- [271] [High-Entanglement Capabilities for Variational Quantum Algorithms: The Poisson Equation Case](#), by Fouad Ayoub and James D. Baeder, arXiv, October 2024 (16 pages).
- [272] [Solving a Linear System of Equations on a Quantum Computer by Measurement](#), by Alain Giresse Tene and Thomas Konrad, arXiv, April 2026 (8 pages).
- [273] [Matrix inversion algorithms by means of automatic differentiation](#), by Kenneth Kubota, Applied Mathematics Letters, July 1994 (4 pages).
- [274] [Quantum algorithm for the gradient of a logarithm-determinant](#), by Thomas E. Baker and Jaimie Greasley, arXiv, April 2025 (20 pages).
- [275] [Variable time amplitude amplification and a faster quantum algorithm for solving systems of linear equations](#), by Andris Ambainis, arXiv, November 2010 (17 pages).
- [276] [Quantum Algorithm for Systems of Linear Equations with Exponentially Improved Dependence on Precision](#), by Andrew M. Childs, Robin Kothari, and Rolando D. Somma, SIAM Journal on Computing, December 2017 (31 pages).
- [277] [A shortcut to an optimal quantum linear system solver](#), by Alexander M. Dalzell, arXiv, April 2026 (13 pages).
- [278] [Constant Factor Analysis of Optimal Quantum Linear Solvers in Practice](#), by Pedro C. S. Costa, Alexander M. Dalzell, Dong An, and Dominic W. Berry, arXiv, April 2026 (20 pages).
- [279] [Constrained Optimal Polynomials for Quantum Linear System Solvers](#), by Matthias Deiml and Daniel Peterseim, arXiv, April 2026 (21 pages).
- [280] [Survey of a Class of Iterative Row-Action Methods: The Kaczmarz Method](#), by Inês A. Ferreira, Juan A. Acebrón, and José Monteiro, arXiv, April 2024 (35 pages).
- [281] [Quantum Kaczmarz Algorithm for Solving Linear Algebraic Equations](#), by Nhat A. Nghiem, Tuan K. Do, and Trung V. Phan, arXiv, January 2026 (10 pages).
- [282] [End-to-end performance of quantum-accelerated large-scale linear algebra workflows](#), by Daiwei Zhu, Miguel Angel Lopez-Ruiz, François-Henry Rouet, Claudio Grotto, Willie Aboumrar, Robert Lucas, Ananth Kaushik, and Martin Roetteler, arXiv, March 2026 (14 pages).
- [283] [Quantum algorithm for linear matrix equations](#), by Rolando D. Somma, Guang Hao Low, Dominic W. Berry, and Ryan Babbush, arXiv, August 2025 (24 pages).
- [284] [Efficient quantum algorithm for linear matrix differential equations and applications to open quantum systems](#), by Sophia Simon, Dominic W. Berry, and Rolando D. Somma, arXiv, May 2026 (56 pages).
- [285] [Analog quantum algorithms for the mixing of Markov chains](#), by Shantanav Chakraborty, Kyle Luh, and Jérémie Roland, Physical Review A, August 2020.
- [286] [The Fine Print of Solving Markov Chains with Analog Quantum Computing](#), by Ward van der Schoot and Niels M. P. Neumann, Intelligence Computing, September 2025 (12 pages) 📄 (preprint on [arXiv](#)).
- [287] [Addressing the Readout Problem in Quantum Differential Equation Algorithms with Quantum Scientific Machine Learning](#), by Chelsea A. Williams, Oleksandr Kyriienko, et al., arXiv, November 2024 (5 pages).
- [288] [Qubit-Efficient Quantum Algorithm for Linear Differential Equations](#), by Di Fang, David Lloyd George, and Yu Tong, arXiv, July 2025 (18 pages).
- [289] [A quantum algorithm for linear autonomous differential equations via Padé approximation](#), by Dekuan Dong, Yingzhou Li, and Jungong Xue, arXiv, June 2025 (51 pages).
- [290] [Investigation on a quantum algorithm for linear differential equations](#), by Xiaojing Dong, Yizhe Peng, Qili Tang, Yin Yang, and Yue Yu, arXiv, August 2024 (22 pages).
- [291] [Constant-Factor Improvements in Quantum Algorithms for Linear Differential Equations](#), by Matthew Pocrnic, Nathan Wiebe, et al., arXiv, June 2025 (42 pages).
- [292] [Efficient Quantum Algorithm for Solving Linear Distributed Delay Differential Equations](#), by Wataru Setoyama and Keisuke Fujii, arXiv, March 2026 (10 pages).
- [293] [Quantum Koopman Algorithms](#), by David Jennings, Kamil Korzekwa, Matteo Lostaglio, and Guoming Wang, arXiv, May 2026 (33 pages).
- [294] [Arbitrary Boundary Conditions and Constraints in Quantum Algorithms for Differential Equations via Penalty Projections](#), by Philipp Schleich, Alán Aspuru-Guzik, Nathan Wiebe, et al., arXiv, June 2025 (59 pages).
- [295] [Tensor-Programmable Quantum Circuits for Solving Differential Equations](#), by Pia Siegl, Dieter Jaksch, et al., arXiv, February 2025 (5 pages).
- [296] [PDEs: what a nice hypercube you have here!](#), by Adrien Suau, Adrien Suau blog, September 2023.
- [297] [Unsupervised Random Quantum Networks for PDEs](#), by Josh Dees, Antoine Jacquier, and Sylvain Laizet, arXiv, December 2023 (33 pages).
- [298] [Quantum algorithm for time-dependent differential equations using Dyson series](#), by Dominic W. Berry and Pedro C. S. Costa, arXiv, June 2024 (19 pages).
- [299] [Fast quantum algorithm for differential equations](#), by Mohsen Bagherimehrab, Kouhei Nakaji, Nathan Wiebe, and Alán Aspuru-Guzik, arXiv, September 2023 (5 pages).
- [300] [Quantum simulation of partial differential equations via Schrödingerisation](#), by Shi Jin, Nana Liu, and Yue Yu, arXiv, December 2022 (9 pages).
- [301] [Quantum Algorithms for Stochastic Differential Equations: A Schrödingerisation Approach](#), by Shi Jin, Nana Liu, and Wei Wei, arXiv, December 2024 (27 pages).
- [302] [Quantum Circuits for partial differential equations via Schrödingerisation](#), by Junpeng Hu, Shi Jin, Nana Liu, and Lei Zhang, arXiv, December 2024 (28 pages).
- [303] [Quantum algorithm for partial differential equations of non-conservative systems with spatially varying parameters](#), by Yuki Sato, Naoki Yamamoto, et al., arXiv, January 2025 (21 pages).
- [304] [Quantum algorithm for solving differential equations using SLAC derivatives](#), by Rakshit M. Gharat, Gopikrishnan Muraleedharan, Dominic W. Berry, and Gavin K. Brennen, arXiv, May 2026 (37 pages).
- [305] [A Quantum Spectral Framework for Solving PDEs](#), by Chih-Kang Huang, Giacomo Antonioli, and Frédéric Barbaresco, arXiv, April 2026 (19 pages).
- [306] [Quantum algorithm for anisotropic diffusion and convection equations with vector norm scaling](#), by Julien Zylberman, Thibault Fredon, Nuno F. Loureiro, and Fabrice Debbasch, Quantum Engineering Sciences and Technologies for Industry and Services, QUEST-IS 2025, March 2026 (9 pages).
- [307] [A Residual-Based Quantum Linear System Algorithm with Dynamic Stopping and Applications to Elliptic PDEs](#), by Xiantao Li, arXiv, May 2026 (26 pages).
- [308] [QCPINN: Quantum-Classical Physics-Informed Neural Networks for Solving PDEs](#), by Afrah Farea, Saiful Khan, and Mustafa Serdar Celebi, arXiv, April 2025 (24 pages).
- [309] [Quantum Physics-Informed Neural Networks for Maxwell's Equations: Circuit Design, "Black Hole" Barren Plateaus Mitigation, and GPU Acceleration](#), by Ziv Chen, Steven H. Frankel, Helen Diller, et al., arXiv, June 2025 (18 pages).
- [310] [Quantum Random Feature Method for Solving Partial Differential Equations](#), by Junpeng Hu, Shi Jin, Nana Liu, and Lei Zhang, arXiv, October 2025 (27 pages).

- [311] [A Quantum Algorithm for the Finite Element Method](#), by Ahmad M. Alkadri, Kranthi K. Mandadapu, et al., arXiv, October 2025 (63 pages).
- [312] [Quantum Algorithms for Solving Partial Differential Equations](#), by Arthur Pesah, March 2020 (8 pages).
- [313] [High-precision quantum algorithms for partial differential equations](#), by Andrew M. Childs, Jin-Peng Liu, and Aaron Ostrander, arXiv, November 2021 (40 pages).
- [314] [Solving partial differential equations on near-term quantum computers](#), by Anton Simen Albino, Erick Giovanni Sperandio Nascimento, et al., arXiv, August 2022 (9 pages).
- [315] [Solving Fractional Differential Equations on a Quantum Computer: A Variational Approach](#), by Fong Yew Leong, Dario Poletti, et al., arXiv, June 2024 (26 pages).
- [316] [Variational quantum algorithm based on Lagrange polynomial encoding to solve differential equations](#), by Josephine Hunout, Sylvain Laizet, and Lorenzo Iannucci, arXiv, June 2025 (21 pages).
- [317] [H-DES: a Quantum-Classical Hybrid Differential Equation Solver](#), by Hamza Jaffali, Frédéric Holweck, et al., arXiv, October 2024 (40 pages).
- [318] [Variational quantum algorithms for nonlinear problems](#), by Michael Lubasch, Jaewoo Joo, Pierre Moinier, Martin Kiffner, and Dieter Jaksch, arXiv, December 2019 (8 pages).
- [319] [Quantum Variational Solving of Nonlinear and Multi-Dimensional Partial Differential Equations](#), by Abhijat Sarma, Peter L. McMahon, et al., arXiv, June 2024 (15 pages).
- [320] [Analog quantum simulation of partial differential equations](#), by Shi Jin and Nana Liu, arXiv, December 2023 (31 pages).
- [321] [Annealing-based approach to solving partial differential equations](#), by Kazue Kudo, arXiv, November 2024 (12 pages).
- [322] [Solving Differential Equations Using Continuous-Variable Quantum Annealing](#), by Kazuki Miyanishi, Soshun Naito, Asuka Koura, Kazue Kudo, Toshiaki Yamaji, and Yuichiro Matsuzaki, arXiv, August 2026 (7 pages).
- [323] [Assessing Quantum Advantage for Gaussian Process Regression](#), by Dominic Lowe, M.S. Kim, and Roberto Bondesan, arXiv, May 2025 (18 pages).

This **Basic algorithms toolbox** subsection contains 287 bibliographical references and notes containing at least 5,782 pages.

1.3 Higher level algorithms

We'll now cover higher level algorithms which are based on the algorithm's toolbox described in the previous part [324–326]. Quantum software engineering requires three main sets of skills: understanding both low and high-level algorithms, then some know-how about the way these algorithms can be assembled and coupled with classical algorithms, and then, above all, how to find the ways to translate “business problems” into these algorithms [327].

1.3.1 Oracle-based algorithms

Deutsch-Jozsa

One of the first quantum algorithms invented comes from David Deutsch, with its derivative called Deutsch-Jozsa, co-invented with Richard Jozsa in 1992 [33]. This algorithm characterizes a function $f()$ called an “oracle” for which we know in advance that it will return for all its inputs, either always with the same value, 0 or 1, or with values 0 and 1 in equal parts.

The Deutsch-Jozsa algorithm solves a promise problem for a Boolean function $f : \{0, 1\}^n \rightarrow \{0, 1\}$ known in advance to be either constant or balanced. The standard circuit uses an n -qubit query register and one ancilla qubit. After the Hadamard gates, the query register is in a uniform superposition over its 2^n computational-basis strings and the ancilla is in $|- \rangle$.

The exact quantum algorithm needs a single call to the quantum oracle. A deterministic classical algorithm needs up to $2^{n-1} + 1$ calls in the worst case, although randomized classical sampling removes the exponential practical gap if a small error probability is accepted [328]. The quantum advantage is therefore an oracle-query separation, not an “instantaneous” evaluation of all inputs. The cost of implementing the oracle itself must also be included when assessing a practical circuit.

Initializing the ancilla in $|1\rangle$ before its Hadamard gate prepares $|- \rangle$, so the oracle action $|x\rangle|y\rangle \mapsto |x\rangle|y \oplus f(x)\rangle$ becomes a phase factor $(-1)^{f(x)}$ on the query register, the usual phase-kickback mechanism. After the final Hadamards, f is constant if the query-register measurement is $|00 \dots 0\rangle$ and balanced otherwise [329].

What is the practical interest of such an algorithm given there are rather few functions $f()$ of this kind? This is an example of a powerful oracle algorithm with no established practical application to date. It has nevertheless been used as a small benchmark and noise-characterization circuit [330]. An extension proposed in 2025 determines explicit output values of the function $f()$ with a single oracle query under an enlarged promise structure [331].

A randomized classical algorithm can query only k randomly selected inputs. Under the Deutsch-Jozsa promise, its error probability is bounded by approxi-

mately 2^{1-k} , so only a few oracle calls are needed for high confidence [328]. This is why the exponential deterministic query separation does not translate into a comparable practical speedup against bounded-error classical computation.

So, is quantum computing useless? Of course not. Other algorithms, less powerful but more useful, have emerged since this “patient zero” of quantum algorithms!

Bernstein-Vazirani

Bernstein-Vazirani's algorithm is less talked-about in textbooks. This algorithm created by Ethan Bernstein and Umesh Vazirani in 1992 is a variant of the Deutsch-Jozsa algorithm. Instead of using two different classes of functions, it tries to learn a secret string encoded in an oracle function (Figure 33). The algorithm was designed to prove an oracle separation between complexity classes BQP and BPP.

For the standard Bernstein-Vazirani promise problem, the oracle is a Boolean function $f_s(x) = s \cdot x \pmod{2}$ hiding an n -bit string s . Although the oracle has 2^n possible inputs, its promised linear structure means that a classical algorithm does not need to scan them. Querying the n unit vectors reveals the n bits of s , requiring $\Theta(n)$ oracle queries. The quantum algorithm recovers the complete string with a single oracle query, giving a linear query-complexity advantage, although its gate complexity remains $O(n)$. Bernstein and Vazirani's recursive Fourier-sampling construction amplifies this into a superpolynomial oracle separation between quantum and classical probabilistic computation, but not a full exponential separation [332, 333].

In the previous edition of this book and other sources, the Bernstein-Vazirani algorithm is presented as bringing an exponential computing time speedup versus classical methods. Actually, it gives only a linear query-complexity speedup over the best classical algorithms (1 query versus n queries), and its overall computing time speedup is similarly linear, not exponential.

Simon

Simon's algorithm solves the promise problem $f(x) = f(y)$ if and only if $y = x \oplus s$ for an unknown nonzero bit string s [32]. It is an Abelian hidden-subgroup problem over $\mathbb{Z}_2^n$ and was historically important because it directly inspired Shor's period-finding ideas. The quantum algorithm uses $O(n)$ oracle queries and classical linear-algebra post-processing, while a classical randomized black-box algorithm needs $\Omega(2^{n/2})$ queries to obtain a collision with constant probability.

In 2025, a US academic team demonstrated a quantum speedup for solving a variant of Simon's problem with 58 qubits of an IBM quantum computer[334]. It was a tailored version of Simon's problem that reduces the problem's complexity and was designed to be tractable on a NISQ system. The speedup was demonstrated with an oracle model, where the function is accessed

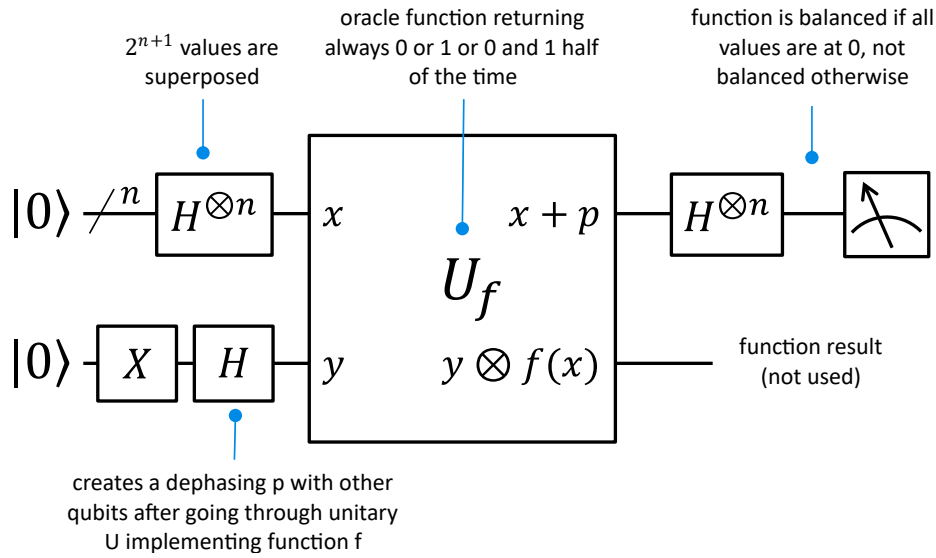


Figure 32: The Deutsch-Jozsa algorithm which determines whether a function is balanced or not. It has no known practical application beyond illustrating a query-model separation. In the usual textbook instances, the oracle is built from CNOT and X gates, so the whole circuit uses only Clifford gates and is efficiently classically simulable. © Olivier Ezratty with various sources. 2022-2024.

as a black box, and is using Clifford gates only, which are easy to classically emulate. It helps run theoretical analysis but doesn't correspond to real-world optimization problems, where the structure of the function is often known or exploited classically. This record was extended in 2026 with supporting 129 qubits for solving a restricted Hamming-weight version of Simon's problem [335].

Grover

The other best-known algorithm in this category is Grover's search algorithm (GSA), introduced in 1996 by Lov Grover [39]. It searches an unstructured space of N candidates for one or more items marked by an oracle. Many combinatorial algorithms use Grover search or amplitude amplification as a subroutine, including minimum finding, SAT variants, graph problems and structured optimization schemes. These applications require problem-specific oracles and often additional algorithmic structure, so their end-to-end complexity is not given by Grover's $\sqrt{N}$ query count alone [336–338].

The oracle may mark $M \geq 1$ candidate states among a search space of size $N = 2^n$, where n is the number of index qubits. In the standard phase-oracle formulation, marked states acquire a minus sign. The diffusion operator then rotates amplitude from the unmarked subspace toward the marked subspace. Grover search is therefore a statement about unstructured oracle access. It is not a replacement for an indexed database, where classical lookup can already be logarithmic or nearly constant time.

For known $M \ll N$, the success probability is near its first maximum after about $\frac{\pi}{4} \sqrt{N/M}$ Grover iterations [339]. A measurement then returns a marked item with high, but generally not unit, probability. Repeating the

complete run can increase confidence. The outcomes are not averaged to identify the marked string.

In the black-box model, exhaustive classical search requires $O(N)$ oracle queries while Grover search requires $O(\sqrt{N})$. For an n -bit search space with $N = 2^n$, this is $O(2^n)$ versus $O(2^{n/2})$. The quadratic gain therefore does not by itself turn an exponentially large search into a polynomial-time algorithm. Structured problems can admit much better classical algorithms, and practical advantage depends critically on the oracle construction, state preparation, logical gate cost and readout and other quantum computing prefactors. Grover-based minimum finding and other optimization methods use additional outer logic rather than a single plain Grover call.

The presentation above is the usual high-success-probability form. Exact variants can reach unit success probability when the number of marked states, or equivalent amplitude information, is known and the final reflection phases are adjusted appropriately. Both standard and exact forms retain $O(\sqrt{N/M})$ query complexity [339]. This exactness is an algorithmic phase-matching property. It is not a switch from “amplitude encoding” to “phase encoding”. It is also distinct from the number of circuit shots used to obtain a desired confidence level.

Attempts to beat the $\sqrt{N}$ bound by relaxing unitarity do not deliver either. Changing the metric tensor hidden in the diffusion operator yields a non-unitary generalization that reaches the solution in a single large rotation instead of many small ones. But normalization has to be paid for somewhere. Implemented with Kraus operators and post-selection, it requires $O(N)$ repetitions, which is no better than a classical brute force scan. Implemented with block encoding and a QSVT

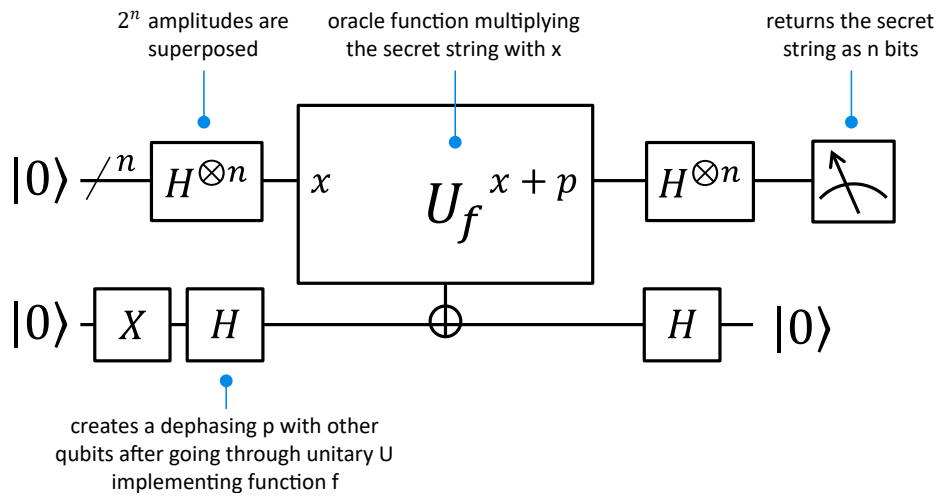


Figure 33: Bernstein-Vazirani algorithm, which has only a linear speedup. © Olivier Ezratty with various sources. 2022-September 2026.

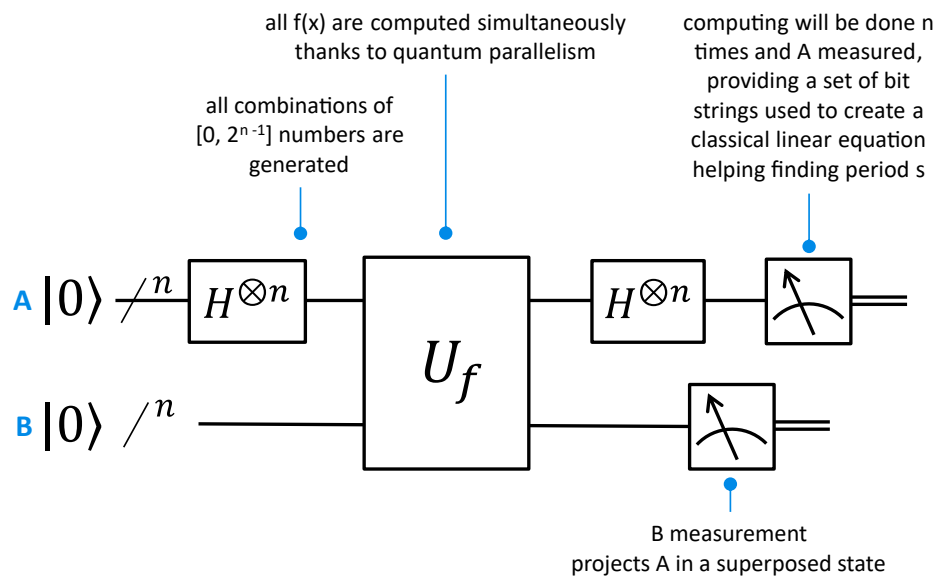


Figure 34: Simon algorithm. © Olivier Ezratty with various sources. 2022-2024.

using Chebyshev polynomials (sequence of orthogonal polynomials that are deeply connected to trigonometric functions), the cost falls back to $O(\sqrt{N})$ with one extra ancilla qubit, exactly Grover's bound [340].

In a 2002 lesson, Serge Haroche pointed out the known fact that these search algorithms have quantum optical interference equivalent implementations, as described in Figure 36 with 4 qubits. This has been described for a while, even trying to use only classical optical elements [341–343].

Variants of Grover's algorithm can solve traveling salesperson problems (TSP) [344, 345] (although not with real speedups vs classical solutions) and various other optimization problems like the shortest vector problem [346–348].

Grover adaptive search (GAS) is the mainstream way to turn a search into an optimization. It was formalized in 2019 by **Austin Gilliam** and **Constantin**

Gonciulea (JPMorgan Chase) with **Stefan Woerner** (IBM Quantum) and published in Quantum in 2021 [349]. Instead of marking the optimum, which is unknown, the oracle marks all states whose cost is below the current best value. Each measured improvement becomes the new threshold and the search restarts. Their contribution was an oracle that encodes polynomial cost functions in the Fourier basis with singly and doubly controlled phase gates, sidestepping the Toffoli-hungry quantum arithmetic that made earlier constructions unusable, demonstrated on small portfolio optimization QUBOs. GAS is now the baseline for Grover-based combinatorial optimization, but it inherits the quadratic query scaling and therefore does not escape the exponential wall on its own.

The 2026 literature accepts this and tries to reduce the prefactors. A 2026 paper from the German QUTAC consortium stated that a quadratic speedup is not enough to naively search an exponentially large space, so the

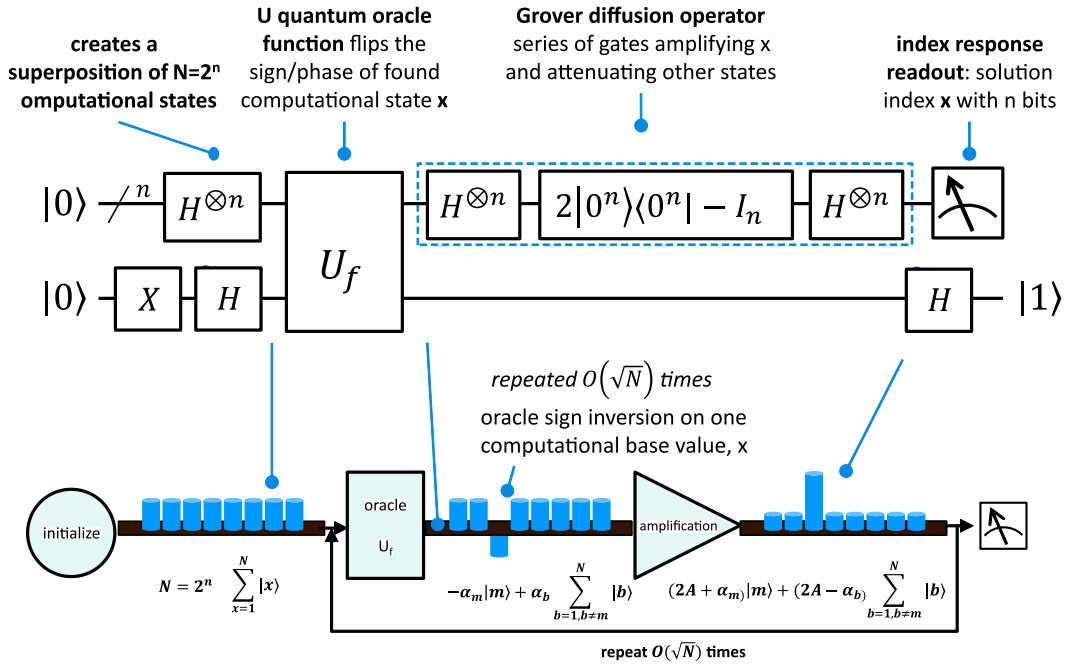


Figure 35: Grover algorithm. © Olivier Ezratty with various sources, 2020-December 2024.

feature	standard Grover search	exact amplitude-amplification variants
Grover iterations	About $\frac{\pi}{4} \sqrt{N/M}$ near the first success maximum for $M \ll N$.	$O(\sqrt{N/M})$, with phases chosen from known information about M or the initial success amplitude.
Purpose	Rotate amplitude toward the marked subspace.	Adjust the final rotation so that it lands exactly in the marked subspace in the ideal model.
Shots	Additional runs may be used to boost confidence.	One ideal run can be sufficient after an exact construction, although physical noise can still require repetition.
Success per ideal run	High, generally below 100%.	100% for the exact construction under its assumptions.

Table 4: Comparison of standard and exact Grover-style search. $N = 2^n$ is the search-space size and M the number of marked states. Exact success does not improve the asymptotic $O(\sqrt{N/M})$ query complexity [339].

search must be preceded by a problem-specific state preparation that skews amplitudes toward promising states [350]. They built one for the TSP, preparing Dicke state superpositions of feasible tours that differ from the incumbent solution by a bounded exchange chain, mimicking the classical Lin-Kernighan heuristic, and compared three strategies for choosing the number of Grover iterations when the number of marked states is unknown. Their benchmark covered 15 instances with 8, 10 and 12 cities, or 64 to 144 binary variables, well beyond what QAOA benchmarking achieves, but way below classical TSP solutions capabilities. The runs were classical emulations using a precomputed set of improving tours rather than real circuit executions. The local exchange chains had to reassign nearly every city on such small instances, and the authors stated they couldn't tell whether the scheme competed with classical heuristics.

GAS has also been extended to noisy objective functions. Georgia Tech researchers proposed in 2026 SOGAS for discrete simulation optimization in the fixed-confidence setting, where the cost of a candidate can only be estimated through stochastic simulation [351]. With it, a binary search narrows an interval containing the optimal mean performance while controlling the error probability, this interval defines the flagging oracle, and amplitude amplification then returns an ϵ -optimal solution. They proved a quadratic speedup in the dependence on the number of candidate solutions. The whole construction relied on a “quantum simulation oracle” preparing a coherent superposition over all candidates together with their noise distributions, obtained by compiling the classical emulator source code into a reversible circuit. That is precisely the assumption that makes oracle-based speedups so hard to work on real problems.

Generalized Grover Search Algorithm (GGSA) replaces the standard Hadamard layer with an arbitrary

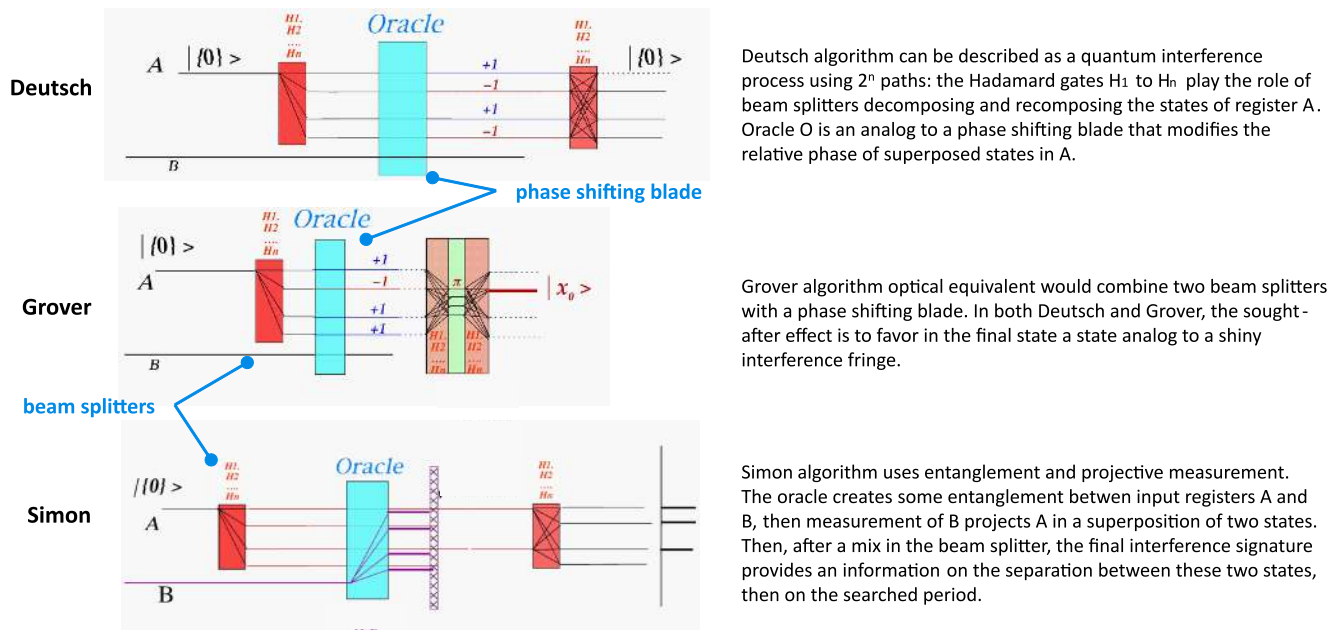


Figure 36: Quantum algorithms explained with interferences when implemented with quantum optics. Source: Serge Haroche, Chaire de Physique quantique. 2001-2002. 8th lesson, February 26th, 2002.

unitary transformation U to prepare non-uniform initial states, providing a theoretical framework for heuristic-driven search, quantum machine learning subroutines, and modeling state preparation noise in an $N = 2^n$ unstructured search space. Theoretical papers demonstrate that the optimal average success probability of a GGSA, which maintains an $O(\sqrt{N})$ quantum advantage over classical $O(N)$ searches, is strictly bounded by the initial state's coherence fraction rather than standard coherence or entanglement [352, 353]. When extended to quantum random-walk search (SKW) algorithms, this resource dependence is highly context-specific. While the coherence fraction positively drives success in standard walks, the presence of initial entanglement or standard coherence actively suppresses the success probability in structurally modified variants. This reveals that the computational utility of intrinsic quantum properties is not absolute, but is fundamentally conditioned by the specific gate architecture applied prior to the $O(\sqrt{N})$ search evolution.

Preparing some non-uniform initial states was studied in 2026 by Korean researchers who proposed a constraint-aware initialization that encodes cardinality constraints with Dicke states and parity constraints with GHZ-type states in the X basis, shrinking the searched subspace before the first oracle call [354]. On a 10-variable exact-cover instance, the optimal number of queries dropped from 25 with a uniform superposition to 5, and the noisy simulation degraded less because fewer queries accumulated less depolarizing noise. The state preparation and the modified diffusion operator cost more per iteration, but their resource analysis did show that the trade is favorable when the oracle and diffusion costs exceed a threshold of about 67 gates.

FTQC requirement. A scalable Grover advantage is generally expected to require fault tolerance because the number of oracle/diffusion rounds grows as $O(2^{n/2})$ for an n -qubit unstructured search, while each round contains a problem-dependent oracle and multi-qubit reflections. NISQ demonstrations therefore remain small and noise rapidly erases the quadratic advantage [355–357]. A March 2026 IBM Heron experiment comparing dynamical-decoupling sequences found that its six-qubit instance fell to the 1/64 random-guess baseline under the tested conditions [358].

Noise does not merely reduce the success probability, it removes the speedup. Researchers from Qruise and academic partners simulated Grover from 3 to 13 qubits under isotropic errors, a rotationally symmetric error class that earlier work argues quantum codes cannot correct [359]. They count the number of repetitions $k(n)$ needed to recover the noiseless success probability and found that it grows exponentially with the qubit count, as $k \sim b^n$ where b depends on the per-gate fidelity. With a per-gate infidelity of 10^{-4} or below, k stays under 10 across all sizes. At 10^{-3} it takes off beyond 10 qubits. At 10^{-2} , which is still within reach of some current platforms, k hits 10^4 at 13 qubits and the success probability at the optimal iteration count collapses to the 1/ N random-guessing baseline. An exponential repetition overhead cancels a quadratic speedup outright.

qRAM dependence. When polling classical data, it also needs some quantum memory (qRAM) to “load” the related database in memory, in the oracle function [360]! There are, however, some available optimization techniques for quantum state preparation in Grover's oracle that could remove this qRAM requirement [361]. Grover's algorithm can also be potentially distributed to several non-interconnected QPUs [362]. An oracle-based

algorithm is efficient if the oracle itself is efficient, which depends on its implementation. If it is accessing some classical data or function, the algorithm’s efficiency may be questionable in the end. One way around a costly monolithic oracle is to avoid building it. Korean researchers studied the case where the target set is the intersection of k constraint sets, each with a cheap oracle, while the global oracle marking the intersection is expensive to assemble [363].

Grover’s absent practical advantage. In March 2023, E.M. Stoudenmire (Flatiron Institute) and Xavier Waintal (CEA IRIG) questioned the very existence of any quantum advantage for Grover’s algorithm, even with an entire quantum oracle. Their rationale was that the Grover diffusion operator is not using much entanglement [364]. Grover’s algorithm quantumness is implemented in the oracle which is usually unspecified. The authors argue that the oracle could be efficiently classically emulated. The authors built a QiGA, a quantum inspired version of Grover’s algorithm, to prove their point and run it using three scenarios: for worst case unstructured problems, for common problems and for easy problems (Figure 37). You’d have to wait thousands of years to get some quantum advantage. The paper drove Scott Aaronson to write a scathing response, and various push-back from the authors [365, 366]. This debate did show a typical discrepancy between theory and practice.

There can be huge differences between theoretical and practical speedups. The paper was then revised, peer reviewed and published in PRX in November 2024 with a different title. It is corroborated by Matthias Troyer from Microsoft who also explained that Grover’s algorithm has prefactors creating a threshold preventing it from bringing a quantum advantage in reasonable timescales, even considering future FTQC QPUs [367]. And the situation can be even worse with “slow” gates QPUs, like with cold atoms and trapped ions, which quickly invalidates theoretical papers like [368] which is about solving NP problems with Grover algorithm running on cold atoms QPUs.

In late 2024, another paper from researchers in the United States and China was even more pessimistic, casting doubts that quantum combinatorial optimizations could one day bring a practical quantum advantage. It focused on the Sherrington-Kirkpatrick problem, which is about solving spin models, with applications in optimizations [369].

And to make things even harder, Ewin Tang and John Wright published a preprint in July 2025 demonstrating that a condition to obtain a quadratic speedup with amplitude amplification and estimation based on Grover’s algorithm, used for solving brute-force search and counting problems, depended on the ability to prepare the oracle unitary U and its inverse $U^\dagger$. They explain that this handicaps the usage of these algorithms when using data coming out of sensors, for which it is hard if not impossible to create $U^\dagger$ [370].

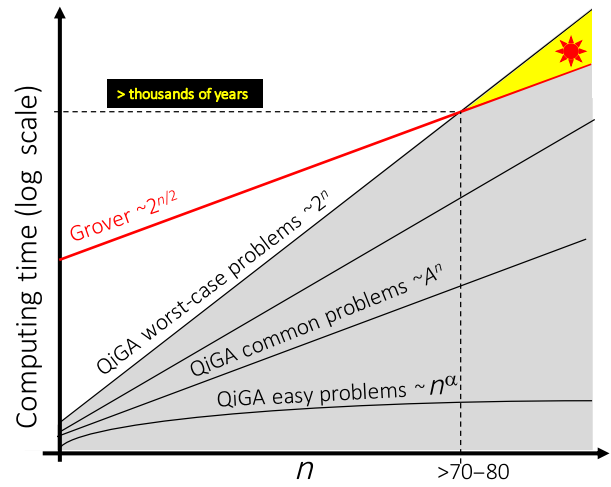


Figure 37: Even in the worst case, the quantum-inspired classical version of Grover’s algorithm fares better than Grover with over 70 FTQC qubits and thousands of years of classical computing. Source: [364].

One need is to generate a circuit implementing $U^\dagger$ from repeated calls to the unknown unitary U . It is possible but expensive. Chen et al. proved in 2026 that implementing such “time-reverse” of an unknown d -dimensional unitary U within a diamond-norm error ϵ (topic covered in the quantum error correction part of this book) requires $\Omega((1 - \epsilon)d^2)$ queries to U , a bound which is tight given the known deterministic and exact inversion protocols using $O(d^2)$ calls [371]. Approximation doesn’t help since allowing a constant error does not reduce the query count on U , and the bound holds for adaptive and coherent algorithms with unbounded ancilla qubits and even for an average-case error metric. With $d = 2^n$, the cost scales as 4^n , which is about 10^6 calls to U for a 10-qubit register and 10^{12} calls for 20 qubits. Reversing an unknown process is therefore practical only in low dimension. Beyond a couple qubits, $U^\dagger$ must come from a known gate-level description of U that can be reversed gate by gate. It is unfortunately missing when U is the evolution of an experimental system.

Bumping on a paper proposing some image detection algorithm using Grover’s algorithm with 2 qubits and no scaling laws should be treated cautiously [372].

1.3.2 Shor integer factoring

Shor’s factoring algorithm factors an n -bit integer in quantum polynomial time [31]. Relative to the best known general-purpose classical factoring algorithms, whose number-field-sieve complexity is subexponential in the integer value but superpolynomial in its bit length, this is more precisely described as a superpolynomial quantum speedup rather than a proven exponential speedup.

The algorithm is based on modular arithmetic and order finding, where a period is searched through controlled modular multiplications implemented in a quantum circuit. At cryptographically relevant key sizes, Shor’s

algorithm requires very deep modular-arithmetic circuits. It therefore needs FTQC systems [373]. The FTQC requirement comes from the accumulated circuit depth and error budget.

It works in three main stages as described in Figure 38.

- A **classical part** chooses an integer a with $1 < a < N$ and $\gcd(a, N) = 1$, then reduces factoring to finding the multiplicative order r of a modulo N . The function is $f(x) = a^x \bmod N$, and r is the smallest positive integer satisfying $a^r \equiv 1 \pmod{N}$. The repeated sequence is therefore $a^0 \bmod N, a^1 \bmod N, a^2 \bmod N, \dots$, not powers of x . Trivial cases are eliminated classically before the quantum order-finding stage.
- A **quantum order-finding part** prepares a phase-estimation-type circuit using controlled modular multiplication by powers of a , followed by an inverse QFT. Resource counts depend strongly on the arithmetic design. A textbook implementation commonly uses roughly $2n$ phase-estimation bits plus an n -bit work register, while optimized implementations trade logical qubits against depth. The Griffiths–Niu semiclassical inverse QFT replaces the multi-qubit QFT readout by sequential single-qubit measurements and classically controlled rotations [375]. It can reduce the phase register to one recycled control qubit, but it still requires a sequence of controlled modular-exponentiation operations and measurements.
- The period finding / modular exponentiation part of the algorithm is the most expensive one and several proposals aim to optimize it or reduce its logical-qubit requirement [376]. This is where the “magic” happens, due to the heavy use of arbitrary phase rotation gates [377].
- Another **classical part** converts the measured phase information into a candidate order r , typically using continued-fraction expansion, then evaluates $\gcd(a^{r/2} \pm 1, N)$ when the required parity conditions hold. Continued fractions recover a rational approximation to the measured phase because phase estimation has finite binary precision. A semiclassical QFT does not remove this classical step. If a chosen a or the measured phase does not produce a nontrivial factor, the procedure is repeated. The overall success probability is a non-negligible constant under standard choices and is amplified by repetition, rather than being a universal 50%.

The speedup should be assessed for the complete factoring problem, not by comparing a classical FFT with the QFT subroutine. For an $n = \lceil \log_2 N \rceil$ -bit integer, Shor’s algorithm has polynomial gate complexity, with the modular arithmetic dominating the cost. The general number field sieve has heuristic asymptotic complexity

$$\exp\left(\left((64/9)^{1/3} + o(1)\right) (\ln N)^{1/3} (\ln \ln N)^{2/3}\right),$$

where $(64/9)^{1/3} \simeq 1.923$. This is subexponential in N but superpolynomial in its bit length.

One of the first experimental demonstrations of Shor’s algorithm was the 2001 seven-qubit NMR experiment factoring 15. These early demonstrations used substantial compilation tailored to the small integer. A 2012 photonic experiment using qubit recycling factored $N = 21$ with a compiled implementation of Shor’s period-finding core [378]. As of 2026, 21 remains the largest well-established integer factored on genuine quantum hardware using this Shor-type period-finding route. Reported attempts at 35 on conventional quantum hardware did not establish a robust larger Shor record. Larger numbers have been factored with other quantum or hybrid optimization methods, but those are not Shor’s algorithm.

On a classical emulator using 2,048 GPUs and 40 simulated qubits, the iterative Shor algorithm factored $549,755,813,701 = 712,321 \times 771,781$, a 39-bit semiprime, in about three minutes [379, 380]. This is a useful circuit-simulation record, not a quantum-computing speedup. The same integers are trivial for modern classical factoring software at these bit lengths.

On a 4-qubit QPU, a 5-digit number, 56,153 [383], was factored, but using a different algorithm than Shor’s. With an optimization algorithm running on a D-Wave annealer, a record was reached in 2016 with the factorization of 200,099 using 897 qubits. In 2019, engineers from Zapata Computing and IBM reported a larger factoring, of 1,099,551,473,989 into $1,048,589 \times 1,048,601$, but using a variational hybrid algorithm on a few qubits, and with an undocumented speedup [384].

Shor’s algorithm threatens RSA because an RSA public key contains a modulus $N = pq$ and a public exponent e . The private exponent d is computed from number-theoretic information derived from the secret primes p and q . Efficiently factoring the public modulus therefore reveals enough information to reconstruct the private key. RSA is not based on a recipient already knowing and dividing by one factor of a transmitted ciphertext.

Large RSA-style integers have been factored publicly by academic teams. RSA-250, a 250-decimal-digit or 829-bit semiprime, was factored in 2020 using the number field sieve [385]. An RSA-260 modulus, a 260-digit, 862-bit integer, was announced as factored by Eric Lu on September 3, 2026 in a post on X. The factors can be checked directly, but no paper or independently audited computation. No general-purpose factorization of a well-generated RSA-2048 key modulus has been achieved classically.

Fault-tolerant resource estimates vary by orders of magnitude with the circuit architecture, error-correction code, physical error rate, cycle time and allowed run-

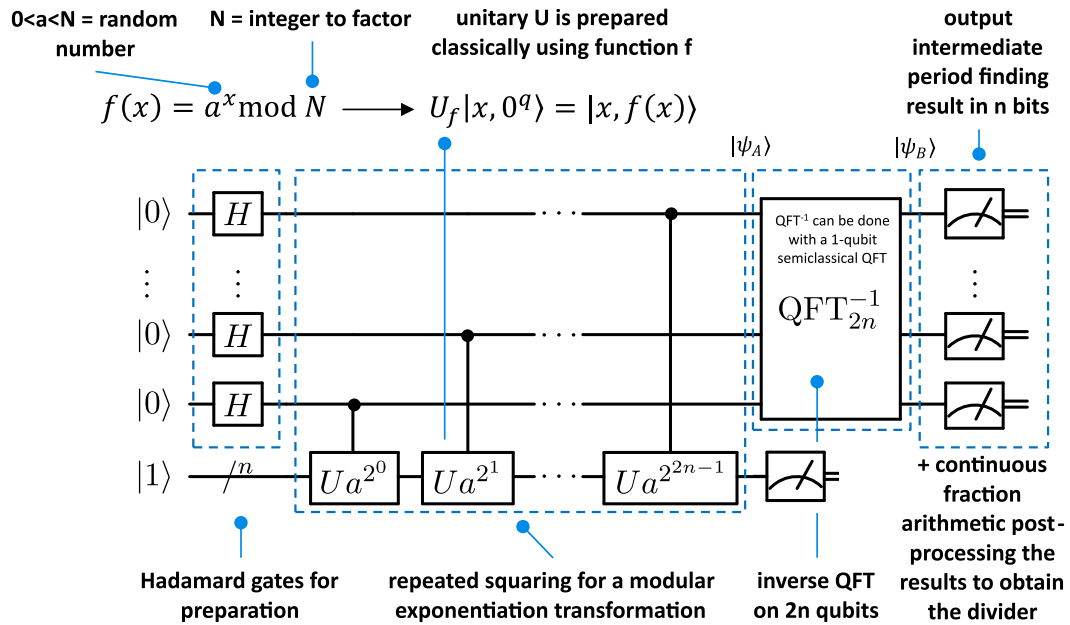


Figure 38: Shor's algorithm with all its qubits. Logs are in base 2. Source: [Wikipedia description](#) of Shor's factoring algorithm and Olivier Ezratty additions. 2021-2024.

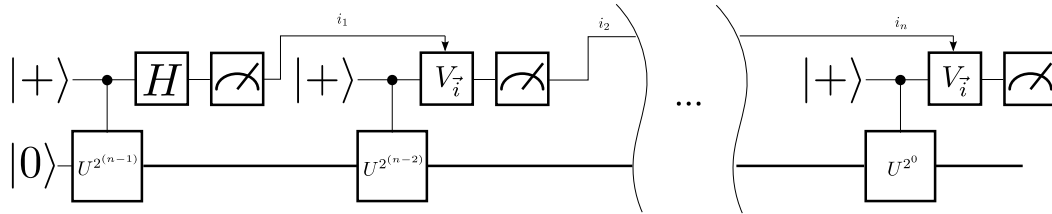


Figure 39: Shor's integer factoring algorithm can use a semi-classical quantum Fourier transform which relies on a single qubit, at the top, using the technique of qubit recycling. It uses a recycled control qubit, sequential measurements and classically controlled rotations between the controlled modular-multiplication operations. Source: [374]. Added in 2024.

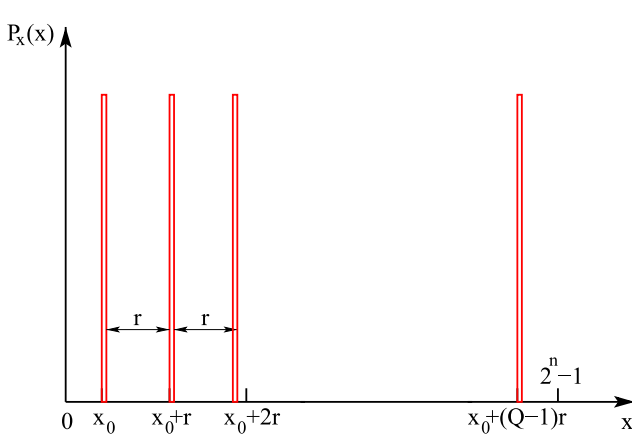


Figure 40: These two figures above illustrate the phase register before and after the inverse QFT in a textbook Shor order-finding circuit. Before the inverse QFT, the amplitudes are supported on basis states separated by the period r , with an offset x_0 . The inverse QFT maps this periodic structure to peaks near integer multiples of $2^m/r$, where m is the number of phase-register qubits.

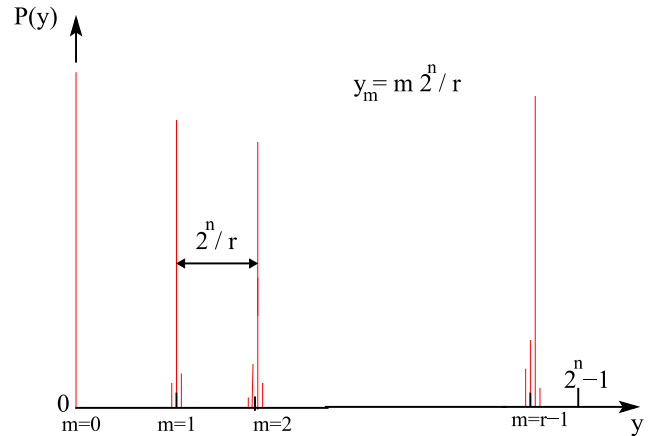


Figure 41: Repeated measurements provide rational approximations from which continued fractions can recover a candidate r . If r is even and $a^{r/2} \not\equiv -1 \pmod{N}$, nontrivial factors can then be obtained from $\gcd(a^{r/2} \pm 1, N)$. Source: [381].

time. The often-quoted $2n + 2$ logical-qubit count is not a theoretical minimum. It is the space requirement

of one specific modular-arithmetic construction, which gives 4,098 logical qubits for $n = 2048$ [386].

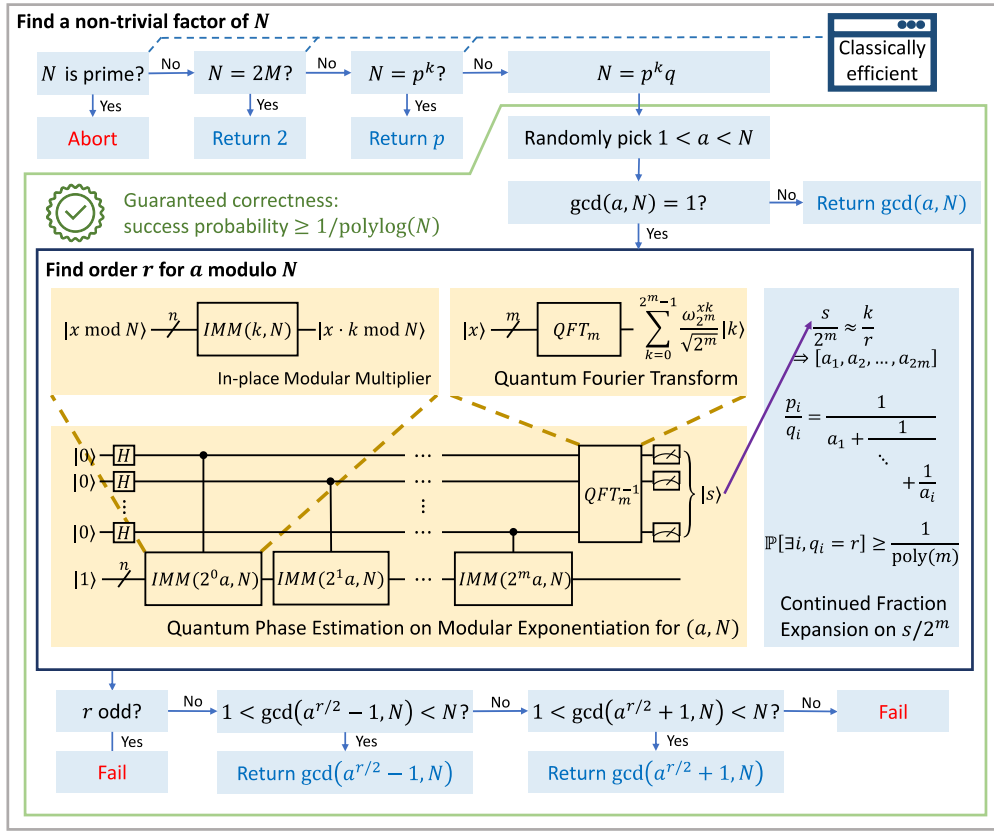


Figure 42: Detailed process of Shor’s integer factoring algorithm, the blue part being the classical part and the yellow part being the quantum part. Source: [382]. Added in 2024.

The surface-code based 2019 estimate by Craig Gidney and Martin Ekerå found that factoring RSA-2048 in about eight hours would require about 20 million noisy physical qubits under assumptions including a 10^{-3} physical gate error rate and a $1 \mu\text{s}$ surface-code cycle [373]. Gidney’s 2025 update reduced the estimate to about a million physical qubits at the cost of a longer runtime of about a week and under comparable physical assumptions [387].

A different architecture proposed by Elie Gouzien and Nicolas Sangouard in 2021 traded runtime and a very large multimode quantum memory for a smaller logical processor. Their estimate uses 13,436 physical processor qubits representing two logical qubits, a memory storing 28 million spatial modes and 45 temporal modes, and up to two hours of storage to factor RSA-2048 in 177 days at a 10^{-3} physical gate error rate [388]. The proposed memory was based on photon echoes in rare-earth-ion-doped solids.

Shor’s algorithm is one motivation for post-quantum cryptography (PQC) and for quantum key distribution (QKD), but neither should be described as “tamper-proof”. PQC uses classical algorithms designed to resist known quantum attacks. QKD can provide information-theoretic key-distribution security under explicit assumptions and with an authenticated classical channel, while practical devices can still have implementation side channels.

But Shor’s factoring is not the only quantum factoring algorithm created so far. Several other options are investigated like the **Variational Quantum Factoring** algorithm alternative that maps the factoring problem to the ground state of an Ising Hamiltonian that could be solved in a hybrid manner using the quantum approximate optimization algorithm (QAOA), running on a gate-based NISQ processor [389]. So far, this algorithm has not been shown to scale to large integers or to cryptographically relevant sizes. In the same way, a researcher in Taiwan experimented an implementation of Shor’s algorithm using the encoding of information on a temporally long single photon across 32 time bins. It could factorize only the integer 15, but it could potentially support a larger number of time bins and larger integers [390].

1.3.3 Shor dlog

Peter Shor introduced his quantum discrete-logarithm algorithm alongside factoring. The discrete logarithm is defined in a finite cyclic group: given a generator-like element b and an element a in the subgroup it generates, find an integer k such that $b^k = a$ under the group operation, for example $b^k \equiv a \pmod{p}$ in a finite field. It is therefore not the ordinary real-number logarithm of two integers [31].

Like the regular Shor integer factoring algorithm, the Shor dlog algorithm yields a vector state that requires some sampling and post-processing to find the solution. One solution exists that yields an exact result.

Mosca and Zalka make the QFT exact for arbitrary orders using amplitude amplification, which turns the discrete-logarithm algorithm into an exact one at the cost of extra gates and ancillas, while preserving its polynomial-time scaling [391]. Another one uses a semi-classical QFT, which can yield a good result between 60% and 82% of the time [151].

Table 5 summarizes the differences between Shor factoring, Shor discrete logarithm and generic QPE. They share Fourier/phase-estimation ideas, but their controlled unitaries, register structure and classical post-processing differ. Factoring reduces to multiplicative order finding, discrete logarithm uses two-dimensional Fourier sampling over a finite group, and generic QPE estimates eigenphases of an arbitrary specified unitary.

1.3.4 Hidden Subgroup Problems (HSP)

The hidden subgroup problem is a generic oracle problem encompassing Simon’s problem and the Abelian structures behind Shor’s factoring and discrete-log algorithms. Graph isomorphism can also be reduced to a non-Abelian HSP, but no efficient quantum algorithm is known from that formulation [392]. Let G be a group, $H \leq G$ a subgroup, and $f : G \rightarrow S$ a function that is constant and distinct on left cosets of H . The HSP is used to determine H through oracle access to f .

The definition is easier to follow after introducing groups, subgroups, sets and cosets:

- **Set:** arbitrary ensemble of elements.
- **Subset:** ensemble of some elements from a set.
- **Group:** a set equipped with a binary operation that is closed and associative, has an identity element, and gives every element an inverse. For example, $\mathbb{Z}$ under addition is a group: the identity is 0, the inverse of a is $-a$, and associativity means $a + (b + c) = (a + b) + c$. Associativity concerns parenthesization, not commutativity.
- **Subgroup:** a subset of a group that is itself a group under the same operation. The even integers form a subgroup of $\mathbb{Z}$ under addition. The odd integers do not, because the sum of two odd integers is even.
- **Coset:** for a subgroup H of a group G and an element g of G , the left coset is $gH = \{gh : h \in H\}$ and the right coset is $Hg = \{hg : h \in H\}$ (these operations may be non-commutative with some non-integer elements like matrices). A subgroup H of a group G may be used to decompose G into disjoint equal-size cosets. H cosets have the same number of elements as H .

Another definition of the hidden subgroup problem is: given a function f that is constant on each left coset of some subgroup H and distinct on different cosets, find H .

The Abelian HSP can be solved efficiently on a quantum computer under standard black-box access assumptions, and this framework contains Simon’s problem and the period-finding structure used by Shor. The general non-Abelian HSP remains open. In particular, the straightforward Fourier-sampling extension does not efficiently solve graph isomorphism [392].

A Simon-type Abelian HSP speedup demonstration was achieved on IBM hardware with an explicitly implemented quantum oracle [393]. The QFT or Fourier sampling is a core processing step after querying the hiding function. It does not, in the standard HSP formulation, eliminate the need for oracle access to f .

Hallgren’s polynomial-time quantum algorithm for Pell’s equation, $x^2 - ny^2 = 1$ for nonsquare positive integer n , is closely related to hidden-period and hidden-subgroup techniques over infinite or continuous structures rather than being an ordinary finite-group HSP [38]. It uses quantum Fourier sampling together with number-theoretic classical post-processing. The result gives a polynomial-time quantum algorithm in the input size for a problem whose best known classical algorithms are substantially more expensive. Calling this a generic “exponential speedup” should be qualified because the precise comparison depends on the classical complexity model.

1.3.5 Computational Fluid Dynamics (CFD)

Fluid mechanics simulations *aka* Computational Fluid Dynamics (CFD) are mostly based on solving Euler (non-viscous fluid) and Navier-Stokes equations (viscous fluids, Figure 44).

These are nonlinear partial differential equations, whose solution is essential to the aerospace industry [394, 395], weather forecasting, plasma magneto-hydrodynamics, astrophysics and many other fields. The problem with Navier-Stokes is twofold: the equations are nonlinear while quantum computing relies on linear algebra, and fluid dynamics are largely dissipative (non-unitary) while quantum time evolution is strictly unitary. Various methods are available to turn nonlinear and non-unitary equations into linear, unitary ones [396–401].

Many quantum algorithms have been designed to solve Navier-Stokes equations, mostly operating in the FTQC regime:

- **Converting a nonlinear system into linear ordinary differential equations (ODEs)** with transforming nonlinear problems into an array of linear equations through the Carleman linearization process (Figure 43) [403–408], which however seems to have strong limitations [409]. It can also rely on

Algorithm Step	Shor Factoring	Shor Dlog	QPE
Problem	Factorize $N = pq$ by quantum order finding	Given group elements a and b , find k such that $b^k = a$	Given a unitary U and an input state with eigenstate overlap, estimate an eigenphase
Input state	Uniform phase register plus modular-arithmetic work register	Two phase registers plus a group-element work register	An eigenstate or a state having nonzero overlap with eigenstates of U
Controlled operation	Modular multiplication by powers $a^{2^j} \bmod N$	Controlled group operations involving powers of the known group elements	Controlled powers U^{2^j}
Quantum output	Phase samples carrying information about the order r	Phase samples giving modular relations from which k is recovered	A binary estimate of an eigenphase, with probabilities weighted by the input-state overlaps
Precision / repetitions	Polynomial resources in the bit length, with repetitions when a sample or chosen a is unusable	Polynomial resources in the group-description size, with repetitions as required	Coherent evolution or query cost scales as $O(1/\varepsilon)$ for additive phase precision ε , while confidence can be boosted by repetition
Classical Post-Processing	Continued fractions or equivalent rational reconstruction, then GCD arithmetic	Modular linear algebra and arithmetic in the underlying group	Convert the measured phase bits into the desired eigenvalue or physical quantity

Table 5: Comparison between Shor integer factoring, Shor discrete logarithm and generic quantum phase estimation.

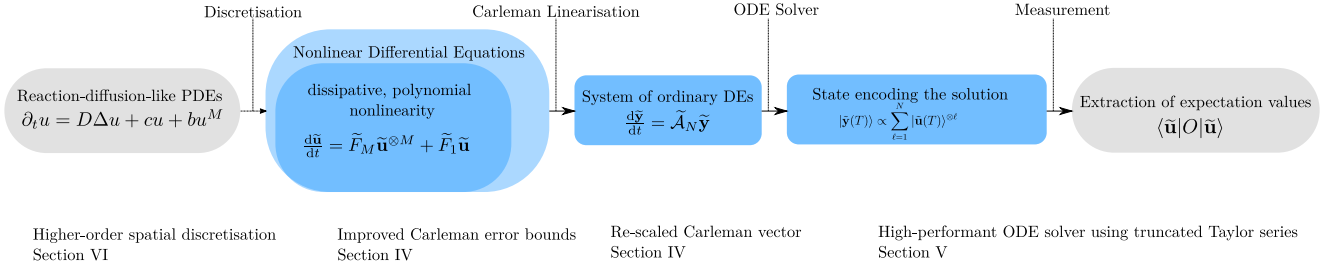


Figure 43: How nonlinear equations can be converted into ordinary linear differential equations using Carleman linearization. Source: [402]. Added in 2024.

high order discretization schemes [410]. A different embedding relies on quantum homotopy analysis rather than Carleman, with better scaling in operator norm, condition number, simulation time and accuracy, an integration window bounded by the inverse of a Reynolds-like parameter, and a 1D Burgers demonstration. The authors present it as usable in both NISQ and FTQC regimes, although its end-to-end complexity is FTQC-grade [411]. One important variant are nonlinear stochastic differential equations (NSDEs) which also have quantum versions [412, 413].

- **Schrödingerisation** techniques which map non-unitary, dissipative classical PDEs onto a higher-dimensional unitary Schrödinger equation. By introducing a continuous auxiliary variable which is a fictitious extra dimension, classical decay is mathematically translated into unitary quantum oscillations.

This allows standard FTQC algorithms like Hamiltonian simulation or Quantum Signal Processing (QSP) to process fluid dynamics. It often leverages Green’s functions to map complex classical initial states and boundary constraints into the quantum circuit [300, 414]. A 2026 reassessment of the Schrödinger-Navier-Stokes (SNS) formulation from Dietrich and Vautherin (1985) shows that its dissipator is very costly to implement on a quantum computer. It proposes instead a Carleman embedding of the Hamilton-Jacobi form of the flow, with a tensor network representation cutting the emulation memory footprint from about 1 GB down to 0.01 GB on Kolmogorov-like flows. It stays an FTQC-regime proposal which is so far validated only in classical emulation [415].

- **Hybrid methods** using a so-called quantum nonlinear processing unit (QNPU) [318, 416]. It is about a

quantum subroutine embedded in an algorithm for a nonlinear equation. The quantum gates themselves remain linear unitary maps. Effective nonlinear dependence is obtained through encodings, multiple state copies, measurements or classical feedback, not through a physically nonlinear unitary gate.

- **Continuous-variable quantum modes** can encode nonlinear differential-equation constructions using multiple copies of a state [417]. These are qumodes, not “continuous-variable qubits”. Polynomial nonlinearities can be represented through tensor products of multiple copies and then processed by quantum linear differential-equation solvers.
- **Differentiable quantum circuits** to solve differential equations [418].
- **Lattice Boltzmann fluid flow** algorithms map lattice-Boltzmann updates into quantum circuits [419, 420]. The Quantum Lattice Boltzmann Method (QLBM) has several formulations [421–423]. State preparation, arithmetic and repeated readout can be expensive, and most proposals target fault-tolerant use. A phase-change heat-transfer formulation published in 2026 used 53 qubits for 17 lattice nodes, but the reported computation ran on a Qiskit Aer matrix-product-state simulator, not on a 53-qubit QPU [424]. PsiQuantum and Airbus extended incompressible QLBM boundary handling with classical validation on Taylor–Green, lid-driven-cavity and cylinder-flow cases [425]. Another approach approximates the nonlinear collision operator with a trained unitary in a restricted low-velocity regime [426].
- **Linear combination of Hamiltonian simulations** (LCHS) applied end-to-end to rapidly distorted turbulence, covering the preparation of an initial turbulent state with a prescribed energy spectrum, its time evolution and the direct measurement of Reynolds stresses and fluctuating velocity spectra. The claimed practical speedup only shows up on large grids and the treated case stays linear, making it an FTQC stepping stone rather than a Navier-Stokes solver [427].
- **Cloud microphysics**, with the collision-coalescence of cloud droplets described by a master equation, the droplet mass distribution amplitude-encoded and the expected droplet number extracted with quantum amplitude estimation (QAE), borrowing from quantum finance algorithms. Due to the use of QAE, it is an FTQC application and it covers only one sub-process of atmospheric modeling [428].
- Solving the **Gross-Pitaevskii equation** (which describes the behavior of a Bose-Einstein condensate) and the **Lotka-Volterra equation** (used to model the dynamics of biological systems in which two species interact, like a predator and a prey) with applications in physics and biology [429].

- **QUBO quantum annealing methods** to solve the laminar plane channel flow problem [430], the **turbine balancing problem** (TBP), for adjusting the mass distribution of turbine components to minimize vibration and ensure efficient operation [431] and **lattice-based automata** computational models which represent the fluid as particles moving on a discrete lattice [432].
- **NISQ variational algorithms** are also proposed although not being in a quantum advantage regime [433, 434]. One study simulated a fluid vortex on 8 superconducting qubits [435] and another used a 15-qubit emulator and 8 IBM hardware qubits [436]. Neither provided resource estimates for reaching a quantum advantage. Quantum machine learning could also be used there [437]. One 2025 example couples a quantum proper orthogonal decomposition with a quantum-enhanced deep kernel learning for reduced order modeling of turbulent flows, claiming $10\times$ faster training with fewer parameters on three benchmarks, but experimented only in emulation and explicitly limited by NISQ-system capabilities [438]. A study of one-dimensional scalar convection with a spectral quantum algorithm also showed that NISQ gate noise does not act as a purely stochastic perturbation but as deterministic physical terms, which makes it modelizable and partially correctable [439].
- **Hybrid quantum physics-informed neural network** (HQPINN) architectures which are proposed using variational circuits in NISQ mode but are so far tested with too few qubits (8) to bring any computational advantage [440–442].

Other ad-hoc algorithms are proposed with up to some exponential speedups [420, 443–445].

According to a 2024 resource estimation study, CFD FTQC resource requirements are way more demanding than with quantum chemical simulations, at least in the compressible and weakly compressible regimes that are in maritime applications [446]. In 2025, a team in China determined that a practical quantum advantage in solving Navier-Stokes equations on a 2^{80} grid could be obtained with 8.71 million physical qubits at an error rate of 5×10^{-4} in 42.6 days [447]. One can wonder why they picked this fidelity level which, particularly at scale, would be quite challenging to obtain. Other resource estimates of CFD based on QSVT are more optimistic [448].

Encoding and readout drive a large share of these budgets. A 2026 analysis quantifies the circuit depth of amplitude encoding initialization and bounds the number of circuit executions needed to extract a whole field, concluding that amplitude-encoded CFD only pays off when the quantum solver feeds another quantum routine instead of being fully read out [449]. Dedicated real space and Fourier space readout methods are more practical than plain tomography on near-term devices, the QAE-based variants being reserved for FTQC [450].

In the same vein, an HHL-based pressure Poisson solver coupled to an incompressible Navier-Stokes discretization was validated in Qiskit emulation on the lid-driven cavity and the Taylor-Green vortex, with an approximate Chebyshev-based state tomography avoiding full state reconstruction, but HHL keeps it firmly in the FTQC regime [451].

On the pessimistic side, a 2026 lower bound analysis shows that any quantum algorithm preparing the final state of the Korteweg-de Vries equation over a time T requires $\Omega(T^2)$ copies of the initial state, and $e^{\Omega(T)}$ copies for the incompressible Euler equations, which rules out a generic exponential quantum speedup for fluid dynamics and echoes the Carleman limitations mentioned above [409, 452]. The classical side is not standing still either, with entanglement and non-stabilizerness (magic) measured on matrix product state encodings of two-dimensional incompressible shear flow data staying low in several flow regimes, meaning tensor network and stabilizer solvers remain competitive on the very data that quantum CFD targets [453].

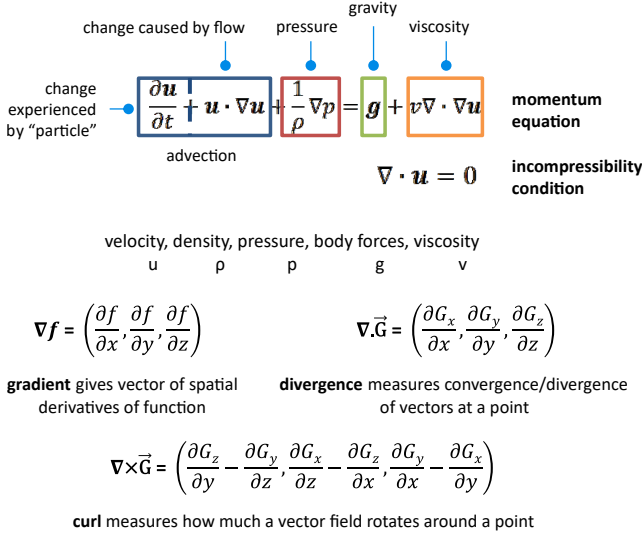


Figure 44: Navier-Stokes equations explained. © Olivier Ezratty with various sources. 2021-2025.

Worth mentioning, **Vorticity** (2001, USA) is developing custom (classical) DSA (domain specific accelerators) to solve Navier-Stokes equations, with a claimed 10^5 speed gain over classical methods. This is an unverified vendor claim with no named metric or baseline, and they do not provide any technical information on their technology (FPGA, ASIC?).

1.3.6 Quantum Artificial Intelligence (QAI)

Quantum Artificial Intelligence covers the broad field of quantum algorithms implementing various forms of artificial intelligence features. There are many potential use cases [454] but their quantum advantage is still in question for a mix of practical and theoretical reasons. Two key challenges arise that are related to the time it would take to load training data and defining

whether quantum machine learning is used for training or inference, or both.

We must differentiate quantum machine learning (QML) from the more generic concept of quantum artificial intelligence (QAI) as was suggested in 2024 by a team of German researchers [455]. It was complemented by a white paper on a research agenda published in 2025 by a group of European scientists, positioning the broad scope of relations between AI and quantum technologies, including QAI [456]. Another similar survey was published in 2026 by Google AI [457], in Italy [458], and in China [459]. Several quantum computing hardware vendors are promoting the use of quantum AI: IonQ (with LLM fine-tuning [460]), Quantinuum (with generative AI [461]), Pasqal (with quantum kernels [462]) and Quandela (with its Merlin emulator running on Nvidia GPUs launched in 2025) with uncertain outcomes at scale.

Quantum AI applications cover potentially all industry sectors, with some documented cases in imaging and in drug discovery in general [463, 464].

Definitions

QAI is a broad computational field that contains quantum machine learning, quantum reasoning, quantum automated planning and scheduling (QPS), quantum natural language processing (QNLP), quantum computer vision (QCV), and quantum agents and multi-agent systems (QMAS). However, QCV and QNLP are mostly always implemented with machine learning techniques, so we have a chicken and egg problem in the terminology. Solving decision problems indeed belongs to the field of AI but is not necessarily implemented with machine learning techniques.

One way to better segment these algorithms is to use the old distinction between connectionist AI which relies on machine learning and symbolic AI which relies on decision trees and symbolic languages (historically, like LISP) and agent networks. Sometimes, symbolic AI is handled with deep learning techniques like with graph neural networks. In classical computing, deep learning is a subset of machine learning that deals with solving problems with (software-based) neural networks, usually to handle larger dimension data than simple machine learning. Quantum symbolic AI is not preminent in the algorithm landscape. It explains why QML dominates in the terminology of QAI.

Could quantum computing accelerate machine-learning training or inference? This remains application dependent. Quantum computing does not automatically create ML tasks that are impossible classically, and end-to-end advantage must include data encoding, training, sampling and readout. Many NISQ QML methods are variational algorithms, but QML as a field also includes other quantum algorithms, particularly targeting FTQC systems [465]. Benchmarks often highlight model quality

or parameter count rather than a demonstrated wall-clock quantum speedup [466, 467].

Various quantum algorithms have been created in the last decades that cover the field of classical machine learning and with a lot of variations in neural networks and deep learning [468].

QML algorithms range from NISQ variational circuits to fault-tolerant linear-algebra, amplitude-estimation and Hamiltonian-simulation methods [469, 470]. There is no general rule that FTQC QML merges training and inference into one pass. Some algorithms learn parameters or model states and then run a separate inference stage, while others estimate a single property directly. For classical data, state-preparation or qRAM assumptions can dominate the claimed asymptotic gain. This is particularly severe for very large language models, whose parameter and activation data are overwhelmingly classical.

Four QML models

The QML literature defines four models that connect how the data is fed into the model (classically, quantumly) and how the process is handled (classically or at least partially quantumly) as shown in Figure 45 [471].

CC with classical data that is processed by classical algorithms. This is classical machine learning.

CQ with classical data that is encoded in quantum states and processed by quantum algorithms, which may need the use of a quantum RAM (qRAM). This is the most common method in NISQ systems.

QC with quantum data that is converted in a classical form and processed by classical algorithms, a solution implemented to analyze quantum physics and sensors measurement statistics, like for doing a qubit tomography.

QQ with quantum data that is processed by quantum algorithms which could be implemented by feeding a QML algorithm directly with quantum data coming from a quantum sensor or generated by some quantum circuit. This seems like the most promising usage of quantum machine learning since it avoids the pitfalls of classical data encoding [46].

Variational circuits

Generically, QML algorithms from the CQ category rely on variational circuits, a family of hybrid algorithms that combine a quantum algorithm and a traditional algorithm that drives the former [472]. VQE and VQA are some of them [473–475]. These algorithms are adapted to NISQ QPUs but their true quantum acceleration is not proven yet [476, 477].

These algorithms are characterized by a cost function defining the problem, the quantum ansatz, how the cost function is optimized, and the data encoded as the input to the VQA and the desired output.

Variational circuits use parameterized quantum circuits (PQC), or parametric quantum circuits (not to be confused with post-quantum cryptography) [478] or even a differentiable quantum circuit (DQC). These are also labelled “ansatz”. These circuits usually contain a sequence of parameterized single-qubit rotations and two-qubit entangling gates that carry the parameters of the problem to encode. A PQC circuit and its rotation angles are refined by a classical optimizer.

There are several types of data preparation ansatzes: with only single-qubit gates (“mean-field ansatz”), which requires only N gates for N qubits and is simple and fast, then the “hardware-efficient ansatz”, which also uses a fixed number of entangling gates and then parametrized 2-qubit gates that are entirely tailored for the problem. These three models are presented in Figure 46 [479]. Various optimization techniques exist to minimize the number of physical gates required to build the ansatz [480] or to improve the ansatz expressivity at the usual expense of the model trainability, particularly with quantum machine learning models [481, 482].

The science behind building a PQC or ansatz is called **Quantum Architecture Search** [483]. It can involve classical machine learning and even tensor networks [484, 485], including for the design of QML variational circuits [486].

The breadth of QML covers analogs of major classical learning paradigms: **supervised learning** uses labeled examples, **unsupervised learning** extracts structure without labels, and **reinforcement learning** learns a policy through interaction with an environment and a reward signal, human feedback being one special reinforcement-learning technique. Quantum versions have been proposed for many neural-network families [487].

One can also wonder about how quantum machine learning implements model training and inferences. In the case of variational QML algorithms, the model is the ansatz from the variational algorithm. Its training happens while optimizing the ansatz for minimizing the cost function.

When the ansatz converges to a satisfactory error rate that can be tested with a test data set, the ansatz can be used for inferences, like for classification tasks. In a way, it is like a classical deep learning circuit that is trained with backpropagation and a classical optimizer. When the model is trained, it can be used for inferences with the trained circuit and some input data.

In classical deep learning, trainable parameters are typically weights and biases, while architectural choices such as layer count, width and activation functions are hyperparameters. In parameterized quantum circuits, trainable parameters can appear in single-qubit rotations, two-qubit gates or more general parameterized blocks. Relevant hyperparameters include the circuit architecture, depth, encoding strategy, number of qubits,

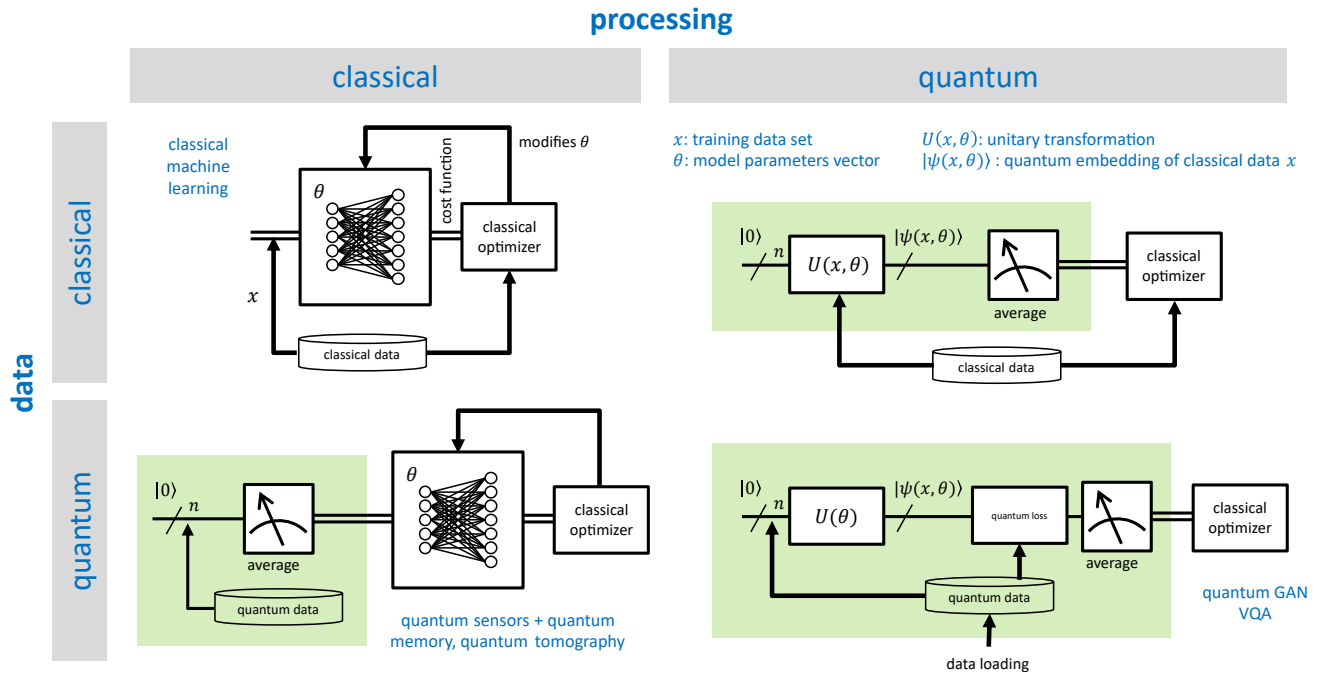


Figure 45: Four main types of QML depending on whether data loading is classical or quantum and part of the processing is classical or quantum. Source: inspired by [471].

observable and optimizer, not only the qubit and layer counts.

For a variational QML model, inference usually encodes one input, executes the trained circuit and estimates one or more observables from repeated shots. No generic end-to-end inference speedup has been demonstrated for such NISQ models. More broadly, however, some fault-tolerant quantum learning and linear-algebra algorithms have conditional asymptotic inference advantages under strong data-access, sparsity or conditioning assumptions [465]. Sampling therefore does not imply that quantum inference can never accelerate. Higher expressivity or lower prediction error is a quality advantage, not by itself a computational speedup.

VQC-based QML can be hybridized in two ways on top of using a classical optimizer to tune the ansatz:

Classical inferences which are run with some quantum code emulation or tensor network libraries or turn the VQC into a classical neural network [488, 489]. This may be interesting, for example, when you need to run these inferences in embedded systems like for machine vision. In that case, QML is used only for training a neural network more efficiently than it could be done classically and with the aim of reducing detection errors.

Classical training when the circuit is entirely trained classically and then run in a quantum computer only for inferences [490]. How this can scale to large machine learning models is an open question given the classical model that would need to be loaded in a quantum circuit.

Quantum machine learning algorithms

We can start with the quantum equivalent of traditional classical machine learning methods:

- **Quantum kernels** are mathematical objects that measure the similarity between data points using quantum states [491–493]. They leverage the high-dimensional Hilbert space of a quantum register to compute inner products between feature vectors mapped into quantum states. This belongs to the quantum kernel methods class which enhances classical kernel methods by using quantum computation to evaluate or approximate kernel functions that would be classically hard to compute [494]. It uses Quantum Feature Maps which encode classical data into quantum states, with angle encoding (which requires a lot of qubits), amplitude encoding (requiring fewer qubits but a lot of gates), block-encoding constructions [164] or even analog computing paradigms [495, 496]. These techniques are typically used in quantum versions of SVMs [497–499].
- **QSVM** (Quantum Support Vector Machine) is a traditional method of segmentation that often relies on matrix inversions, based on its use of HHL [500, 501] or which can be reformulated as a QUBO problem for analog quantum computers [502, 503]. It can be used for example in text sentiment analysis. It is a typical supervised learning method.
- **Feature extraction** [504–506], which includes **PCA** (Principal Component Analysis) is used to determine the key variables in a data set [507, 508]. This is like searching for eigenvectors of a data set. Again, HHL

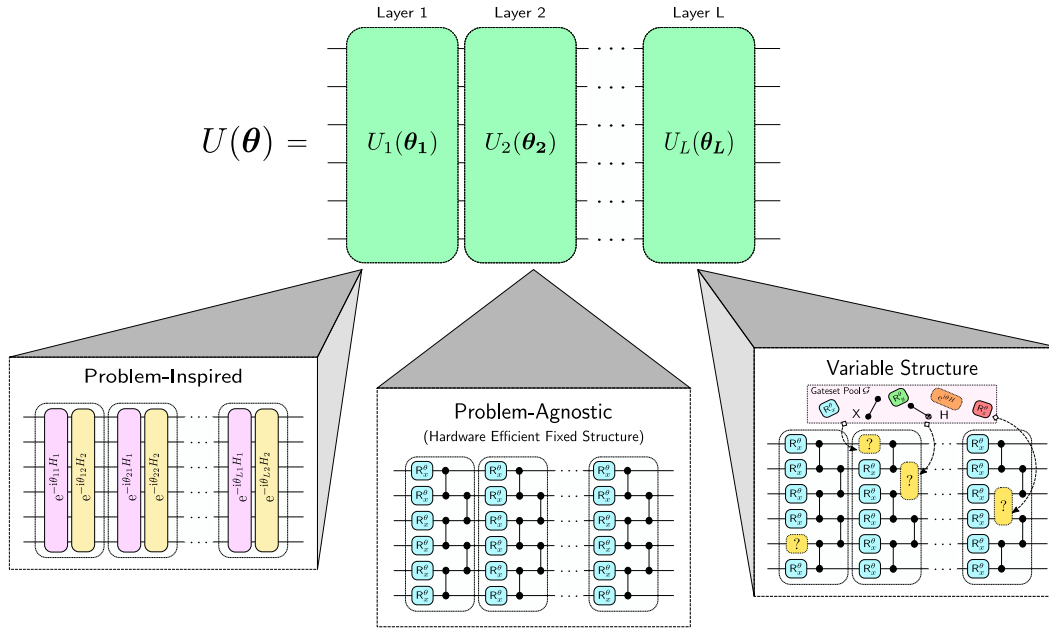


Figure 46: Various ways of preparing a QML ansatz or model, problem inspired, problem agnostic and with a variable structure. Source: [479].

is behind it. It is a typical unsupervised machine learning method. It has potential use cases in cybersecurity for the detection of new threats [509] or for the analysis of data generated by chemiresistive sensor arrays [510].

- **Automatic data clustering** using unsupervised machine learning algorithms [511]. It includes quantum equivalents of k-means clustering, that is usually done after a PCA [512] and spectral clustering, a method that builds a similarity graph using distances between data vectors, extracting the eigenvectors from a matrix built with this graph and projecting the data onto this new orthogonal space and applying a classical k-means clustering method. But as seen frequently with QML algorithms, the best acceleration requires using some qRAM [513]. It can for example help identify cybersecurity threats [514, 515] or detect anomalies in various industry use cases [516].
- **Quantum nonlinear regression** algorithm, one of the basic quantitative value prediction methods of machine learning [517].
- **Decision tree classification** algorithms, enabling regression and binary classification tree construction and retraining [518], including Random Forest supervised learning [519]. Des-q is a quantum algorithm designed for constructing and efficiently retraining decision trees used for regression and binary classification. Under its data-access assumptions, it obtains polylogarithmic dependence on the stored data size when retraining with small update batches. The advantage is therefore conditional on large existing training sets, small updates and efficient qRAM-style access [518].

- **Recommendation systems** that are particularly useful in marketing or content publishing including the famous algorithm from Kerenidis et al. [520] which was dequantized by Ewin Tang in 2018, meaning that she found a classical efficient equivalent. Kerenidis then pointed out that with certain recommendation parameters, the quantum algorithm was still clearly superior [521].
- **Ensemble learning methods** combine various machine learning techniques to improve predictions or regressions [522].

Now onto the various breeds of neural networks based deep learning models which nearly all have quantum equivalents, so, are labeled quantum neural networks [523]:

- **Perceptrons** implementation, related to the original neural network architecture designed by Franck Rosenblatt in 1957 [524, 525].
- **Recurrent Neural Networks** (RNNs) process sequential or time-series data [526]. LSTM (Long Short-Term Memory) networks are gated RNNs specifically designed to mitigate vanishing gradients and retain dependencies over longer time intervals than vanilla RNNs [527]. Quantum LSTM proposals currently use small variational circuits, including time-series forecasting applications [528], and can be partitioned across QPUs with classical communication [529].
- **Gradient descent and backpropagation** used during training phase of neural network [530, 531]. So far, it has been tested at a very limited scale (6 qubits in [532]) and doesn't scale well unless some conditions are met [533]. For example, Melnikov et

al. found that gradient-based optimization will not scale as efficiently as deep neural networks equipped with backpropagation [534]. The researchers propose a way forward by reducing the task of estimating gradients from shadow tomography with a “quantum-efficient” protocol for computing gradients, but highlight the unavoidable classical costs involved.

- **Quantum Convolutional Neural Networks (QCNNs)** are hierarchical quantum circuits with local convolution-like unitaries and pooling operations, originally proposed for tasks such as quantum phase recognition and quantum error correction [535]. They are not synonymous with **quantum convolutional neural networks**, which apply small quantum circuits as feature-generating filters to patches of classical data before classical processing [536]. Certain QCNN architectures can have favorable trainability properties [537], while classical simulability depends on circuit structure and task rather than on the QCNN label itself [538].
- **Reinforcement Learning** networks where a computing agent learns how to make decisions and take actions in an environment to maximize a cumulative reward [539]. It is inspired by the way humans and animals learn from interaction with their surroundings [540, 541]. There are many developments in this field [542–548], for robots navigation [549] (with 8 qubits), process synthesis optimization problems [550], and new benchmarking techniques [551].
- **Equivariant Neural Networks (ENN)** with better geometrical robustness and better resistance to adversarial attacks [552].
- **Quantum Residual Neural Networks** are the quantum equivalents of residual neural networks. It is a deep learning architecture introduced in 2015 by Kaiming He et al. [553] to solve the vanishing gradients problem which is not far conceptually from the barren plateaus that limit the training capability of quantum neural networks. The technique bypasses some layers in the network and adds input directly to the output of the remaining layers. A quantum version framework proposal would use Rydberg atom-based analog quantum simulators to implement classification tasks [554].
- **Quantum Graph Neural Networks (QGNN)** have many applications [555–557], particularly in chemistry and biology [558], high-energy physics, complex systems optimization and finance. But given they are usually variational algorithms subject to the barren plateau phenomenon, they have some limitations to address, one of them being the use of logical qubits and FTQC platforms [559, 560], the other being limited quadratic speedup [561]. There’s also a quantum graph neural network (QGNN) framework variant [562].
- **Quantum transfer learning** including hybrid transfer learning which trains a quantum neural network using an already trained classical network [563–565].
- **Capsule Networks**, which recognize images features with considering their relative position [566].
- **Reservoir Computing Networks** which map input signals to a high-dimensional space having complex quantum superposition. Its output is then processed through a classical linear regression model or a relatively simple neural network [567, 568]. Only the classical readout is trained. So, there is no quantum gradient and no barren plateau to worry about, which makes QRC (Quantum Reservoir Computing) the cheapest QML family to run on current hardware and explains its 2025-2026 publication surge [569]. It has various use cases like time-series predictions [570–572], binary classifications for credit card default prediction, run on the 256-qubit neutral atom QuEra’s Aquila simulator [573], and in foreign exchange market applications, including, potentially, in NISQ platforms [574–576]. LINKS Foundation and Intesa Sanpaolo ran ATM cash demand forecasting, with a 4-qubit QRC running on an IQM Spark but it lost to a classical baseline [577]. The largest demonstrations are analog, with 108 neutral atoms [578], with over 400 optical modes in a Gaussian boson sampler [579] and 4,400 qubit D-Wave quantum annealer [580], and digital, with 124 qubits and more than 5,000 two-qubit gates on IBM processors [581], all of them reaching parity with well-tuned classical reservoirs rather than beating them [582, 583]. Theory now explains why. Memory and nonlinear response trade off against each other across every QRC protocol [584]. A linear reservoir with linear readout cannot create fixed-delay expressive power that the preprocessed input does not already contain [585], and in the common qubit-resetting encoding, all nonlinearity comes from the classical encoding map unless the dynamics generate some quantum magic [586]. Also, QRC seems to not scale well [587]. While it can potentially be implemented with all qubit types, there are custom hardware proposals for its realization, like with photonic platforms [588–591] and analog cold atom simulators [592–594]. No quantum advantage has been demonstrated yet with QRC.
- **Active Learning** algorithms which select the training data set to make learning more efficient and save up to 85% of training time. It labels unlabeled data using an iterative supervised learning [595].
- **Binary Neural Networks** with parameters and activation functions constrained by only two possible values [596].
- **Spiking neurons** emulation although its speedup is not obvious [597].

- **Image classification and analysis** [598, 599] with different use cases like improving satellite imaging interpretations [600, 601] although with quite a distant quantum advantage [602], image segmentation [603], edge detection improvements [604, 605], generic image classification [606, 607], medical image classification [608–610], similarity detection [611], image denoising [612], and defects detection in manufacturing [613]. One critical aspect is how images are encoded, using dimensionality reduction, patching like with Another Block Amplitude Encoding (BAE) which slices images into smaller parts [614] or uncompressed (still small) images [615]. One technique is the Amplitude Encoding [616], with an Approximate Amplitude Encoding (AAE) variant which can encode images with up to 2 million pixels. It can also use a QFT to reduce the number of gates in the encoding process [617]. On NISQ QPUs, quantum inferences have a lower quality than with classical CNNs [618], even with handwritten digits [619]. This may be overcome by reducing the dimensionality of the input data [620]. Image classification can also be based on inverse Quantum Fourier Transforms [621]. There are also quantum versions of video analysis algorithms although they are usually tested with small resolution videos without addressing the data loading question [622]. Compression techniques can be used to lighten training data-loading operations [623]. Some variations of QCNN exist for quantum annealers [624, 625]. Image classification can also benefit from using quantum equivalent of fuzzy networks [626].
- **Quantum Natural Language Processing** (QNLP) is making some inroads as well, in various limited small-scale cases [627, 628]. Using the technique of word embedding [629], it can help with classification tasks [630, 631] and simple text summaries. One algorithm was proposed to use Grover’s search algorithm to invert a sparse attention computation matrix efficiently to accelerate a hybrid LLM, but no resource estimates are provided and the potential polynomial speedup from Grover algorithm is suspicious and highly dependent on how the matrix, sparse or not, is loaded [632].
- **Generative Adversarial Networks** (GAN), which can create various content extrapolated from existing content. It includes Quantum Generative Learning Models (QGLMs) [633], the models based on so-called Quantum Circuit Born machines (QCBM) [634–636], Quantum Variational Autoencoders (QVAE) [637], Quantum Generative Adversarial Networks (QGAN) which generate synthetic content from existing content by checking its plausibility via a network of recognition neurons [638–643], Quantum Invertible Neural Networks (QINN), which are designed to be reversible and bijective and Diffusion Probabilistic Models (DPM) to yield high-quality images and audio [644]. They can perform both forward computation

(mapping inputs to outputs) and inverse computation (mapping outputs back to inputs). They can transform data so that the original input can be recovered from the transformed output. Their applications include image and signal processing, including image denoising, image super-resolution and image synthesis [645], although not yet with a demonstrable quantum advantage [646]. There are also proposals for qGANs generating tabular data [647]. We can also mention the Quantum Flow Matching proposal, a generative model extrapolating intermediate steps between two density matrices [648].

QGLMs can be used to create (quantumly generated) synthetic training data sets used in classical machine learning models [649–651] as well as for option derivatives pricing in financial services [652]. Their quantum advantage may however be limited in scope compared to classical generative models. Generative models could demonstrate a quantum advantage only with limited training data [653]. GANs can be protected with some quantum watermarking techniques useful for GAN QaaS (quantum as a service) [654].

- **Transformer neural networks** which put in place self-attention mechanisms. These are the basis of all current large language models, the LLMs like GPT, Mistral and Llama. They detect dependencies between different portions of sequences of inputs, like in text and enable models to assign different weights to different parts of sequences, making it possible to focus on relevant information and capture long-range dependencies and predict the next item like with texts [655, 656]. Some are wondering whether LLMs could be potentially implemented on a quantum computer [657]. It may be a very long-term dream, at least given the challenges with data loading and overall resource estimations [658], particularly given it seems difficult to obtain some exponential speedup either on training or inferences [659]. This concern has motivated various proposals to compress models [660]. Transformer models are tested at small scale like at Quantinuum [661] or hybrid techniques associating classical tensor networks and quantum computing [662]. Recent work focuses quantum computing on fine tuning LLM models, but sometimes, on dubious claims on energy savings [460] or on the creation of AGI, thanks to a simple reduction of the number of parameters by 76% over one billion parameters [663]. Some researchers are proposing to use FTQC systems and to block-encode the matrices used in quantum transformers, focusing on inferences and considering it can use an already trained model as input. But the query complexity gain seems only polynomial, and it presupposes that some qRAM is available [657]. Another proposal, from Multiverse, combining a VQC and a tensor network classical component to improve LLM accuracy, but with no resource estimates [662].
- **Geometric Quantum Machine Learning** (GQML) which takes into account group symmet-

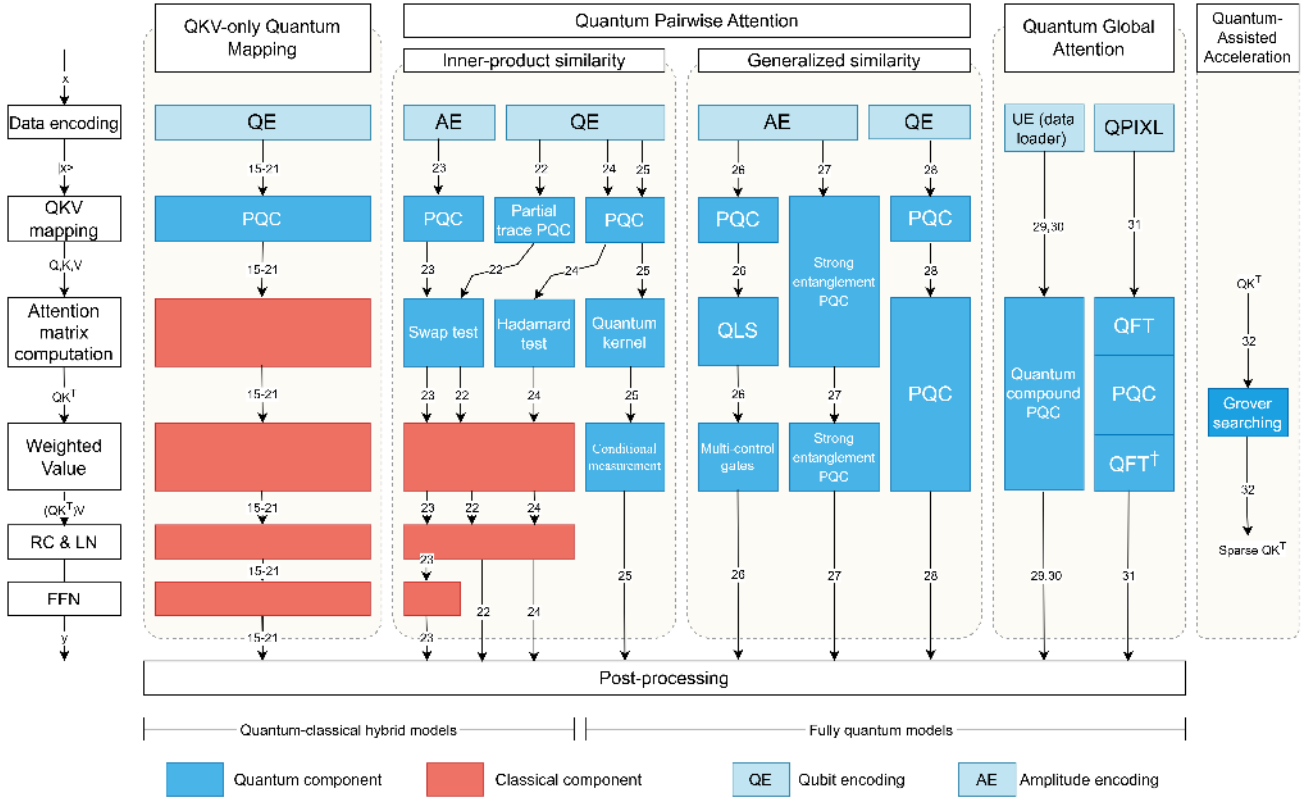


Figure 47: Various scenarios of quantum transformers with different balances between quantum components (blue) and classical components (red). Source: [664].

ric abstract representation of training data [665]. For various reasons like the need to avoid barren plateaus or reducing the number of dimensions and parameters, some strong inductive biases must be embedded in QML models. Imposing an a priori or an inductive bias restricts models to exploring only a subset of the function space due to its definition hypotheses, keeping the optimal functions in the target space. It can sometimes considerably reduce the parameter space without altering the search for solutions. Geometric QML is based on this desire to build QML models that respect the underlying invariances in the data, called symmetries. The most successful neural networks are models with inductive biases respecting the underlying structure and symmetries of the domain on which they act (translational invariance for a CNN, invariance by permutation for a transformer, invariance by time deformation for an LSTM, *aka* long-short term memory). Geometric QML implements systematic methods for creating such models having better generalization performance, smaller training data requirements, as well as favorable optimization landscapes. The underlying concepts are quite complicated and involve Lie groups, group invariant QML and the like [666–668].

- **Quantum Federated Learning (QFL)** algorithms, which distribute training on several quantum computers to improve the training time while preserving

privacy, with distributing and sharing the learned model instead of the training data [669–677], including ways to create secured communication with Quantum Secure Aggregation (QSA) [678], homomorphic encryption [679–681], to optimize data encoding [682] and to distribute part of the processing to classical clients [683].

- **QPINN**, *aka* Quantum Physics-Informed Neural Networks, incorporate differential-equation residuals, boundary conditions or conservation constraints into a quantum or hybrid learning objective [308]. It can for example support Maxwell’s equations [309] and Schrödinger’s wave function [684]. As with classical PINNs, these constraints are usually enforced softly through loss terms. Exact or “hard” enforcement requires a specific constrained ansatz or output transformation. Most case studies use NISQ parameterized circuits [685].
- **Quantum theorem provers** are also investigated, with a proposal using combinatorial logic and symbolic AI generating some quadratic speedup, which is probably insufficient to deliver a practical quantum advantage [686].
- **Artificial General Intelligence (AGI)** denotes a loosely defined goal of systems with broad, human-level or beyond-human competence across domains. Modern language models can appear human-like in

some conversational Turing-test-style protocols, but passing a particular imitation test is neither a consensus definition nor evidence of AGI. Whether quantum computing offers a useful primitive for AGI remains speculative, particularly with regards to the understanding of the physical world in general. Proposed Hamiltonian formalisms for agents are currently theoretical [687, 688]. However, as of September 2026, advanced LLM-based models perform strongly on selected knowledge and reasoning benchmarks, which does not establish human-level general intelligence. And this is obtained without any need for a quantum computer.

In all these cases, the proposed algorithms are not specific to some qubit hardware architecture beyond the nuances between gate-based and analog quantum computers. There are still some proposals that are very hardware specific like some which are using continuous variable photonic QPUs [689].

The table in Figure 48, created in 2018, summarizes asymptotic speedups claimed for several early QML algorithms [690]. Such tables must be read cautiously because they often show scaling only in the training-set size N , while suppressing feature dimension, condition number, precision, state-preparation and output costs. Large quantum processors with far more than fifty physical qubits now exist, but this has not produced a large-scale, end-to-end QML quantum advantage. The limiting issues are circuit quality, barren plateau effects, data access, trainability, readout and fair classical baselines rather than qubit count alone.

QML is also one of the fields of application of D-Wave’s quantum annealers. Annealers work well to find minimum energy of complex systems which is equivalent to searching for a minimum level of errors in the adjustment of the weight of neurons in a neural network [691]. So far, D-Wave has tested an RBM (Restricted Boltzmann Machine) model [692]. They also did it with a hybrid algorithm for image recognition in a neural network, based on a variational circuit and a hybrid algorithm. But with very low-resolution images [693, 694]! D-Wave offers machine learning services in its Leap quantum cloud computing offering. But they are not the only ones.

QAI challenges

Many challenges remain to be addressed to operationalize QAI and QML beyond the emergence of sufficiently powerful and reliable quantum processors [467, 696]. These challenges relate to a mix of hardware support issues, data loading costs, and algorithmic shortcomings [46].

This list however shows, when available, some of the potential solutions being studied to address these challenges.

- **No demonstrated NISQ QML advantage at scale.** A great majority of practical QML papers test small circuits, often between 3 and 16 qubits, and many use classical emulators [716–719]. Small parameter counts or occasional accuracy gains do not by themselves establish quantum advantage if comparable classical models, inference time, data-loading cost and hardware cost are omitted. Examples in image analysis, anomaly detection and cybersecurity remain proof-of-concept scale [720–723]. The QML field has a substantial theoretical literature, including results on trainability, generalization, dequantization and data-access lower bounds. The unresolved problem is connecting these theoretical conditions to an end-to-end practical advantage on useful data.

- **Data encoding** may take time and have a negative impact on the acceleration provided by QML. It is rarely accounted for in most QML related papers [724]. Its performance depends on the used method [725] and the training data structure and size [726, 727]. For FTQC setups, it also frequently requires quantum random access memory (qRAM), and no large-scale fault-tolerant qRAM meeting these access assumptions has been demonstrated, even if some Quantum Data Loaders are proposed to circumvent this need [489, 728–733]. Some specific methods must be created to load images efficiently as we’ve already seen [734], including methods based on a search-based classical genetic algorithm [735, 736] and using classical machine learning [737]. Another method is data re-uploading that interleaves trainable parameterized gates between encoding gates, which enhances model expressivity, but it doesn’t scale well with deep circuits [738]. Other methods are developed that could require less training data than with classical machine learning [739]. Various papers tend to show that a quantum acceleration would be warranted only with having training data already provided as a quantum state, which is impractical in most cases [740, 741]. Others propose standardized machine learning training datasets that are encoded in quantum gates and usable classically, to benchmark QML and classical solutions [742]. Having a preliminary understanding of the training data structure can help optimize data loading in a variational circuit [743]. There are also techniques for compressing quantum neural networks to save quantum resources [744, 745], some using the knowledge distillation technique [746].

- **Nonlinear activation functions** such as sigmoids used in classical neural networks are difficult to implement in quantum algorithms since quantum gates all apply linear transformations [748]. ReLU, which stands for rectified linear unit, is the function $f(x) = \max(0, x)$. It is nonlinear and nonunitary, so it cannot be applied by a gate directly and also requires an indirect construction based on measurement, ancillas, approximation or a continuous-variable nonlinear element [749]. There are workarounds, on

TABLE 1 Overview of the main quantum machine learning algorithms that have been reported in the literature, and complexities

Algorithm	Classical	Quantum	QRAM
Linear regression	$\mathcal{O}(N)$	$\mathcal{O}(\log N)^*$	Yes
Gaussian process regression	$\mathcal{O}(N^3)$	$\mathcal{O}(\log N)^\dagger$	Yes
Decision trees	$\mathcal{O}(N \log N)$	Unclear	No
Ensemble methods	$\mathcal{O}(N)$	$\mathcal{O}(\sqrt{N})$	No
Support vector machines	$\approx \mathcal{O}(N^2) - \mathcal{O}(N^3)$	$\mathcal{O}(\log N)$	Yes
Hidden Markov models	$\mathcal{O}(N)$	Unclear	No
Bayesian networks	$\mathcal{O}(N)$	$\mathcal{O}(\sqrt{N})$	No
Graphical models	$\mathcal{O}(N)$	Unclear	No
k -Means clustering	$\mathcal{O}(kN)$	$\mathcal{O}(\log kN)$	Yes
Principal component analysys	$\mathcal{O}(N)$	$\mathcal{O}(\log N)$	No
Persistent homology	$\mathcal{O}(\exp N)$	$\mathcal{O}(N^5)$	No
Gaussian mixture models	$\mathcal{O}(\log N)$	$\mathcal{O}(\text{polylog } N)$	Yes
Variational autoencoder	$\mathcal{O}(\exp N)$	Unclear	No
Multilayer perceptrons	$\mathcal{O}(N)$	Unclear	No
Convolutional neural networks	$\mathcal{O}(N)$	$\mathcal{O}(\log N)$	No
Bayesian deep learning	$\mathcal{O}(N)$	$\mathcal{O}(\sqrt{N})$	No
Generative adversarial networks	$\mathcal{O}(N)$	$\mathcal{O}(\text{polylog } N)$	No
Boltzmann machines	$\mathcal{O}(N)$	$\mathcal{O}(\sqrt{N})$	No
Reinforcement learning	$\mathcal{O}(N)$	$\mathcal{O}(\sqrt{N})$	No

Figure 48: Main QML algorithms. This table provides a simplified comparison based primarily on the training set size N and does not account for other important parameters such as the number of features, data dimensionality, condition number, target precision, data-loading cost, number of optimization iterations, or assumptions on data access such as QRAM. The mentioned quantum complexities should therefore be interpreted as indicative asymptotic scaling under the assumptions of the corresponding algorithms, rather than as end-to-end computational complexities. Source from 2020: [695].

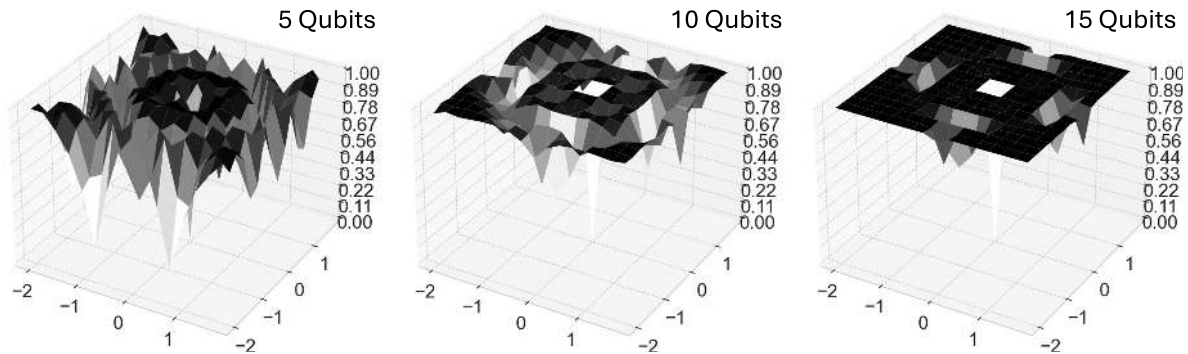


Figure 49: The effect of barren plateaus can appear at moderate circuit sizes. The X and Y axes are two variational parameters θ_1 and θ_2 spanning an angular domain, while the plotted landscape becomes progressively flatter as the number of qubits increases. A barren plateau corresponds to gradients and gradient variance becoming exponentially small, not large. The original caption's $-\pi$ to $-\pi$ range and its interpretation of zero versus unit variance were reversed. Sources: chart in [697] and explanation in [698].

which Iordanis Kerenidis has worked [750, 751] and some others [15, 752, 753]. For example, mid-circuit quantum measurements can create the sought-after activation function nonlinearity in neural networks [754]. There are also suggestions to use continuous-variable photonic architectures and modes to handle neural networks with nonlinearity provided by non-

Gaussian qubit gates [15], photonic quantum neural networks using Kerr nonlinearities [755] and even other techniques not requiring measurement [756–758].

- **Training or inference?** In many cases, the ambiguity is strong between these two aspects. Is a quantum advantage searched for both of them or for

QML dequantization conditions

Trigger	Threshold	Why dequantization works	References
Non-Clifford count	$t \in \mathcal{O}(\log n)$ relevant T -gates	Clifford+ T simulation costs $\text{poly}(n) 2^{\mathcal{O}(t)}$, through stabilizer decompositions or related techniques. Logarithmic T -count drives polynomial classical complexity.	[699, 700]
Circuit depth	$\mathcal{O}(\log n)$ total depth of local gates in 1D	limited entanglement growth limits the MPS/MPO bond dimension by $\text{poly}(n)$, enabling efficient tensor-network contraction (MPS and MPO are defined in 1.7.2).	[701, 702]
Dynamical Lie algebra	$\dim \mathfrak{g} \in \text{poly}(n)$	relevant observables evolve inside a polynomial-dimensional invariant operator space, allowing Heisenberg evolution to be computed there instead of in the full 4^n operator space.	[703–705]
Circuit noise	constant local noise, under the assumptions of the simulation method	noise suppresses high-weight Pauli contributions, allowing approximate truncation to a polynomially tractable set of Pauli paths or operators.	[706–708]
Architecture	separated encoding and trainable blocks with a $\text{poly}(n)$ -size induced function basis	separation of data and parameter dependence can expose an efficiently representable classical function space containing the quantum model.	[702, 709]
Data encoding	$\text{poly}(n)$ distinct joint Fourier frequencies	if $f_{\vartheta}(x) = \sum_i C_i(\vartheta) T_i(x)$ has only $\text{poly}(n)$ efficiently computable basis functions, it admits a polynomial-size classical surrogate.	[710–713]

Table 6: Structural conditions that can make a variational QML model classically simulable or approximated by a classical model, following [702]. They reflect different forms of low effective classical description complexity, including stabilizer, tensor-network, Lie-algebraic, Pauli and function-space representations. Full simulation classically computes $(x, \vartheta) \mapsto f_{\vartheta}(x)$, whereas surrogation reproduces $x \mapsto f_{\vartheta}(x)$ for fixed trained parameters, removing the need for quantum inference but not necessarily excluding advantages in training or generalization [711]. These are sufficient regimes under the assumptions of the mentioned methods, but not universal dequantization criteria. Avoiding them does not generate some quantum advantage. Increasing circuit depth or expressivity can also degrade trainability through barren plateaus [714, 715]. Some circuits outside known worst-case regimes may also remain efficiently simulable on average [708]. © Olivier Ezratty, September 2026.

Q Is oracle sketching solving the data-loading problem?

An April 2026 preprint from Google AI, Oratomic, Caltech and MIT indicated a possible quantum advantage with QML applications by bypassing the qRAM data-loading bottleneck by using a sampling method and an optimized sequential loading using incremental quantum state rotations called “oracle sketching”, which is an approximate oracle based on the exploitation of data samples and the use of classical shadows for measurement [733].

The theoretical paper exhibited a quantum advantage exponential in memory, when using fewer than 60 (logical) qubits, but only polynomial in (training) computation time, mainly at the expense of deeper circuits [747]. There is no free lunch! Three use cases were presented: RNA sequencing by PCA (principal component analysis), movie sentiment analysis, and

the resolution of dynamic linear systems ($Ax=b$) for the detection of anomalies in networks, with simulated training accuracies improving without adding more logical qubits (Fig 3. in the paper). These were not implemented on real quantum hardware since it is FTQC-grade.

The approach requires sparse, well-conditioned data and specific initial estimates, making it inapplicable to generalized algorithms like deep neural networks. The proposal sits in the FTQC regime since the accumulation of data in memory presupposes effective quantum error correction. The model also applies mainly to training, using “classical shadows” to generate a vector that is then used for classical inference. There is not yet any practical full-stack computing time analysis of this hybrid FTQC-classical scheme.

only one? For NISQ variational circuits, inference is just about running a trained circuit with a settled ansatz. But an FTQC circuit is more complex, with all the model loading. Can this run real time for inferences? As a result came the notion of classical surrogates which are classical implementations

of QML models for classical inferences. It can save QPU resources and improve response times, which can make sense if it must run in some embedded system like in a car [759]. But in that case, where is the quantum advantage? In training quality? In

training cost reduction? They indirectly question the nature of the QML quantum advantage [760].

- **Dequantization** is a major challenge for quantum machine learning. It means constructing an efficient classical algorithm, simulator or surrogate for a quantum method under specified input-access and structural assumptions. Ewin Tang’s 2018 recommendation-system result is a prominent example [761]. A polynomial classical counterpart does not automatically imply identical practical runtime because polynomial degrees and prefactors matter, but it removes claims of an asymptotic superpolynomial separation under the same access model.
- **Avoiding the barren plateau** phenomenon which limits the efficiency of QML algorithms as the problem size scales [762] and starting with relatively few qubits (see Figure 49). For example, [763] circumvents this problem with a quantum model and quantum optimizer. In a July 2026 theoretical paper, Iordanis Kerenidis proposed the unitary brick-wall and butterfly architectures to overcome barren plateaus and scaling costs in quantum deep learning [764, 765]. It relies on some photonic system generating up to thirty synchronized 4-qubit or 6-qubit blocks of non-Gaussian magic-states. While targeting early FTQC, the resource sizing doesn’t account for fault-tolerance costs in such photonic system. The butterfly architecture’s hardness also relies on multiple unproven mathematical conjectures.
- **Avoiding spectral bias** of parameterized quantum circuits used in NISQ QML. It can be avoided with data encoding strategies that distribute redundancy uniformly [766].
- **Exploiting or fighting noise?** As shown in Table 6, in a NISQ regime, noise can influence dequantization potential for QML algorithms. It could however in some circumstances help improve a system trainability and its inferences. However, this depends a lot on multiple parameters [767].
- **Benchmarking** QAI/QML vs classical solutions should be more rigorous to demonstrate a clear quantum advantage, even with papers providing resource estimates for future FTQC QPUs. Most of the papers mentioned in this part on QAI present tests implemented in a classical emulable regime, with fewer than 20 qubits, and with even no clear qualitative quantum advantage, like with MNIST image recognition. This happens even on QPU hardware when >100 qubits are available like in this time series forecasting case published in 2025 by IBM researchers that limits itself to 10 qubits with no resource estimates for a quantum advantage [768]. This doesn’t demonstrate any QAI utility. Benchmarking should ideally focus on AI applications where classical solutions fail to deliver good solutions. It can also help compare different QML solutions solving the same problem [769, 770].
- **Resource estimates** are missing in most papers for delivering some quantum advantage, or when they are provided, they are worrying. In [771], an Australian academic team proposed in 2025 to reduce the cost of QEC without implementing magic state distillation. The cost of this QML solution is still high due to the usage of a variational circuit with between 50 and 100 layers. They found that their method could mitigate the barren plateau effect and drastically reduce hardware resources. The result? You’d still need 30K physical qubits with the usual 0.1% physical error rate to support 10 logical qubits, which can be efficiently emulated on a \$50 Raspberry Pi! The test application is a MNIST image recognition task. All tests were done with classical emulators.
- **Distributing Quantum Machine Learning** processes across multiple quantum processors could be interesting for handling large problems [772], provided it is implemented in a secured manner [773] and doesn’t annihilate any quantum advantage if each processor runs in a dequantizable regime due to the absence of quantum interconnect between them.
- **Speedup** must be brought by QML compared to today’s most advanced processors. We still lack resource estimates to assess these speedups which are very theoretical at this point. IBM published in 2021 a mathematical proof of a potential quantum advantage for a quantum machine learning classification task done with a quantum kernel method based on the Shor dlog algorithm. There was no actual experiment done due to the absence of sufficiently powerful quantum computers [774]. We’ll probably be stuck in this situation for several years from now. A recent proof of exponential separation between an HHL-based task and its quantum inspired equivalent is conditioned by the sparsity of the data and thus cannot be generalized [775]. In the case of polynomial quantum speedups, one approach is to optimize these speedup coefficients and finding hardware approaches with short circuit shot times [776].
- **Overselling LLM support.** Since the advent of ChatGPT and its growing energetic needs to feed hundreds of millions of users with costly inferences, quantum-based LLMs have been put forward as a potential solution to mitigate their huge energetic drain. This doesn’t hold given our existing knowledge of the matter. LLMs require loads of training data that would require centuries of data-loading, just for training or a single inference. And hefty FTQC platforms supporting deep circuits and at least a couple hundred logical qubits. Will quantum LLMs be used for training or inferences? If it’ll be only for training, the energetic gain will be relatively moderate. If it will be for inferences, how can we make them real-time as they are today? These are open questions without clear answers, or we know the answer but it’s bad news. The scientific literature in the quantum

LLM domain that we covered in this part is showing some potential benefit of QAI for relatively narrow tasks, like for using Gen AI to create synthetic data that could improve training, but not for LLMs, or for LLM fine tuning [460, 777]. These tasks won't probably address the LLM energetic cost issue nor improve response times despite various claims [778, 779].

- **Algorithms explainability** must also be addressed [780]. The decomposition of the training and inference processes of these quantum neural networks will probably be different from their implementation in more traditional processors [781, 782]. Some proposals for explainable quantum AI are already made [783–785]. QML may be challenged by future memristor-based neuromorphic processors that will allow networks to converge more rapidly with back-propagation. Memristors will make it possible to place the neuron's computational functions and the associated memory in the same location in a semiconductor circuit, accelerating access to memory by several orders of magnitude during computations.
- **Privacy handling** is a usual concern with classical machine learning [786]. It can be addressed in many ways with quantum machine learning. One way relates to the use of differential privacy, a mathematical framework and technique ensuring that the privacy of individual data entries is preserved when aggregate data or models are deployed and queried. It protects the privacy of individuals whose data was used to train machine learning models, and to make sure that no single individual's data can be inferred from some machine learning inference result. It is mainly based on adding some random noise to the training data or network gradients during training, drawn from a Laplace or Gaussian distribution. Variations of differential privacy are proposed for quantum machine learning [787–790], including an interesting technique leveraging the effects of quantum error correction [791], one that is leveraging noise [792], with shuffling data [793], and one that works over distributed quantum computing [794].
- **Resilience to various attacks** to resist the stealing of a QML model exposed on the cloud [795], adversarial attacks [796–799], Trojan horses [800] and more generic holistic attacks [801].

On the other hand, QML's algorithm developments have served as a source of inspiration to improve classical algorithms.

In the AI domain, the European project H2020 **Quromorphic** launched in July 2019 aimed at creating a quantum processor dedicated to the execution of neural networks inspired by the brain. Quromorphic involves IBM Zurich, ETH Zurich, TU Delft, Volkswagen and Spanish and German Universities. Given the participants, we can guess that this will be based on superconducting qubits. The project got a funding of €2.9M and

was scheduled to end in 2022 [803]. It reminds us of the very controversial European flagship Human Brain Project led by Henri Markram from Switzerland that ran from 2013 to 2023.

Let us note finally the existence of an association promoting the field of AI and quantum computing, the **IAIQT** foundation based in Switzerland.

Classical machine learning can be useful for quantum physics and quantum computing. We saw that Google used a deep learning algorithm to optimize the microwave frequency plan of the Sycamore processor's qubit control. Machine learning can also be used to model and simulate condensed matter, with an impact on the development of various qubits, especially superconducting qubits [804–807].

We will probably be exposed to various fancy claims combining artificial intelligence and quantum computing and the devil will always be in complicated details [808]. For instance, how about using these QML algorithms in robotics or for creating some sort of artificial general intelligence? Not so fast [809–812]! It is still science fiction operating in the *click-bait* regime.

Will there be some quantum LLMs?

Running a large language model on a quantum computer is now a field of its own, with two paradigms surveyed in [813]: variational circuits replacing pieces of the attention block, and FTQC quantum linear algebra reimplementing the whole transformer. One complete construction in the second family assembles block encodings, QSVT and a row-wise softmax subroutine into an inference pipeline whose cost is polylogarithmic in sequence length and embedding dimension [814]. A quartic speedup can be obtained with removing the tomographic readout between layers, obtaining the first fully coherent multilayer network provided the weights sit in an elusive QRAM [758].

Sizing remains a major obstacle for quantum LLM proposals. A Llama 3.1 405B model has about 4.05×10^{11} parameters. Mathematically, an amplitude vector of that length can be encoded on 39 qubits since $2^{39} \simeq 5.5 \times 10^{11}$. This does not mean that 405 billion classical weights become an operationally accessible 39-qubit memory. Preparing a generic amplitude-encoded state has $\Omega(N)$ complexity [55], so an illustrative direct loading circuit would require of order 10^{11} controlled rotations or entangling operations before fault-tolerant synthesis overhead. Under the optimistic throughput assumptions used here, the stated 2×10^{13} T-gate estimate corresponds to roughly one year of initial state preparation. A qRAM would move rather than erase the storage problem. At 16 bits per weight, 4.05×10^{11} parameters already represent about 6.5×10^{12} classical bits, before activations and metadata [815].

How about demonstrations? In 2026, Multiverse researchers inserted 4×4 two-qubit Cayley-parameterized unitary adapters in frozen projection layers of a Llama

Q What learning means in the context of quantum computing?

Learning refers in general to algorithms that adapt or optimize their output based on data, usually so-called “training data” functioning similarly to classical machine learning but utilizing quantum circuits.

The most common approach involves training Parameterized Quantum Circuits (PQC like VQE or QAOA), where a classical optimizer iteratively adjusts the variable angles of single qubit quantum gates based on sampled outputs to find an optimal solution.

This concept is used in Quantum Machine Learning (QML), where quantum algorithms are used to classify or generate data. These models can process traditional classical data mapped into a quantum

state, or they can directly ingest native quantum data coming from quantum sensors.

Learning also applies to the physical systems themselves, such as using algorithms to deduce the properties of an unknown quantum state through efficient measurement techniques like classical shadows or with a state tomography, or with learning an Hamiltonian when the algorithm’s goal is to compute unknown internal dynamics, interactions, and coupling strengths of a physical quantum system by observing its behavior [802]. Quantum learning theory studies the mathematical limits of these processes, exploring whether quantum entanglement and superposition can fundamentally reduce the amount of data required to train a model.

3.1 8B model on a 156-qubit IBM quantum processor and improved perplexity by 1.4%, from 8.877 to 8.752, with lower being better, using 6,144 extra parameters [816]. They actually used 64 independent two-qubit blocks, a regime that is straightforward to simulate classically, so this does not demonstrate quantum advantage. The Llama model remained classical and was not loaded into the quantum processor. The Cayley unitaries were used only as small adapter components. Arunachalam et al. proved unconditional separations between constant-depth quantum circuits and bounded-resource transformer and diffusion language models [817], but for engineered sampling and relation problems, not next-token prediction on natural language. Scaling Cayley adapters to a classically hard regime would require much larger coherent blocks and a full accounting of data transfer, circuit depth, error correction and sampling cost.

1.3.7 Quantum physics simulation

Quantum simulation algorithms are used to reproduce matter at the quantum level in a computer. It can be used to simulate the interaction between atoms in molecules for new materials or drugs design.

They can also simulate physical phenomena related to magnetism or the interaction between photons and matter. This amounts to solving “N-body problems”, i.e., calculating the interaction between several particles according to the physical laws governing their interaction. Quantum simulation also helps in studying how superconducting materials behave, particularly at (relatively) high temperature, superfluids at low temperature, the temperature-dependent magnetism of certain materials and the interactions between graphene and light.

These solutions run in gate-based universal quantum computers as well as on quantum simulators and quan-

tum annealers although we still lack data to compare their respective performance.

Exact full-configuration-interaction treatment of an electronic structure grows combinatorially with the number of spin orbitals and electrons, so generic real-time dynamics rapidly becomes intractable. Classical approximate methods such as coupled cluster, tensor networks, density-matrix renormalization, embedding and quantum Monte Carlo can treat far larger orbital spaces when the physical structure is favorable. The relevant quantum opportunity is therefore strongly correlated electronic structure and dynamics in regimes where controlled classical approximations fail [818].

Quantum-chemistry resource estimates span a very broad range. Some active-space ground-state problems may fit in tens to hundreds of logical qubits, while chemically relevant fault-tolerant calculations can require thousands or more logical qubits and 10^9 to 10^{15} non-Clifford operations depending on the molecule, basis, precision and algorithm. A number such as 10^{-14} is therefore meaningful only as an approximate *logical error per operation* for an extremely long circuit, not as a generic physical-qubit error-rate requirement. Physical error rates can be many orders of magnitude larger when quantum error correction is used.

Figure 51 positions the number of logical qubits needed to simulate the functioning of a mitochondrial protein, MRC2, in an optimistic way. On a practical level, most quantum chemistry proposed algorithms are about computing the ground state of the electronic Hamiltonian of a molecule. It deals with finding the smallest eigenvalue of this Hamiltonian. With this starting point, quantum chemists can determine chemical reaction energies, potential energy surfaces in molecules to better understand reaction dynamics and find “molecular docking” sites and at last, undertake virtual spectroscopy of the studied molecules.

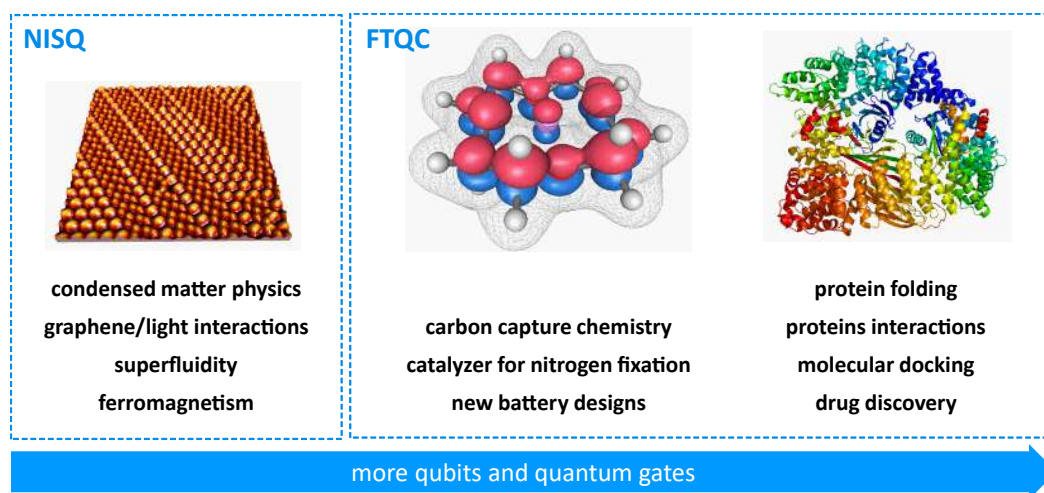


Figure 50: Quantum physics simulation applications and a fuzzy grade of complexity given that very quickly these simulations require thousands of logical qubits in the FTQC regime, particularly for organic chemistry. © Olivier Ezratty, 2020-2024.

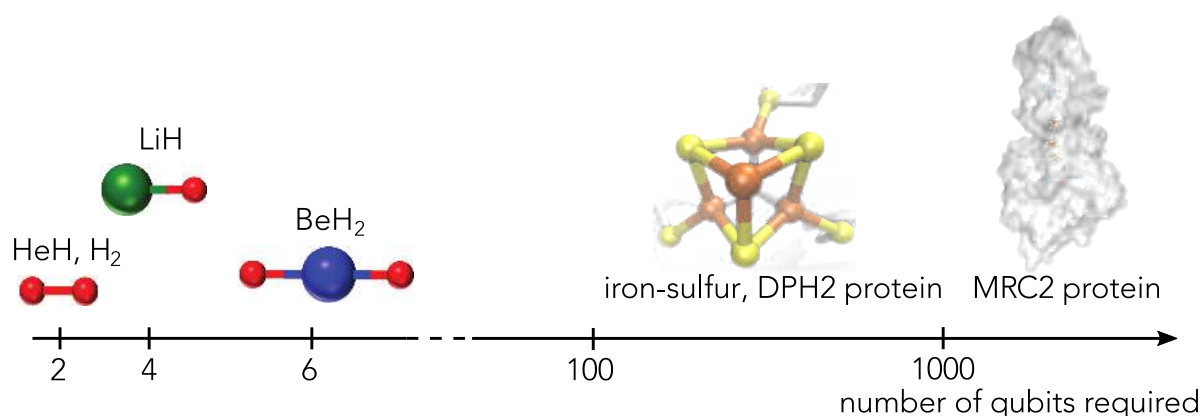


Figure 51: Another grade of complexity, for molecular ground state Hamiltonian simulations, in logical qubits. It seems quite optimistic since many cases studies mention logical qubit counts in the thousands and tens of thousands even for rather simple molecules. The number of logical qubits is dependent on the number of relevant electron orbitals in the molecule and there are many such orbitals per atom. And these are just resource estimates for ground state Hamiltonian estimation, not for molecular dynamics, molecular docking or chemical reaction simulations. The “today” mark has not changed since the publication of the paper. Source: [819].

Here are some examples of quantum simulation algorithms tested at very small scales to compute energy ground states of small molecules or materials, that you could run in a second on your own smartphone with Quirk:

- Simulation of **quantum field theory** as reviewed by John Preskill [820].
- Simulation of **hydrogen atoms** using two superconducting qubits computers [821].
- Hybrid simulation of a **simple CaH⁺ ion** also using two superconducting qubits [822].
- Simulation of **simple molecules** by Alán Aspuru-Guzik, one of the world’s leading authorities in the field [823]. It is centered around the H₂ molecule. It was tested with 5 qubits.
- Simulation of a **three atoms molecule**, beryllium hydride (BeH₂) with 6 qubits by IBM [824] and 8 qubits by DTU [825].
- Determination of the **equilibrium state of simple molecules**, also with 5 qubits [826].
- Hybrid **molecular simulations** combining classical and quantum algorithms, with 6 qubits [827].
- Simulation of **semiconductor dynamics** using 6 qubits [828].

And then, some resource estimates for more complicated quantum chemistry simulations:

- Simulation of the **cytochrome molecule**. With error rates of 0.1%, the run time would be 73 hours and require 4.6M physical qubits. With an error rate of 0.001% which is way out of reach, computing time could be down to 25 hours with 500K physical qubits.

The equivalent classical computing time is 4 years and requires 348 GB RAM and 2TB of storage [829].

- Simulation of **catalytic chemical processes** proposed by researchers from Microsoft and ETH Zurich with a requirement of about 4,000 logical qubits [830].
- Simulating organic molecules of medium complexity such as **cholesterol** would require about 1,500 logical qubits and, above all, the ability to use billions of quantum gates [831] (Figure 53). VQE algorithms could also be used there with a universal gate-based quantum computer using a reasonable depth of quantum gates (number of steps in the algorithm) [832, 833].
- Simulating **periodic solids**, which would require tens to hundred million of qubits [834]!

Iron-sulfur clusters such as ferredoxins and other cofactors involved in photosynthetic electron transfer are plausible targets for high-accuracy electronic-structure calculations. Quantum computing could eventually contribute to such calculations, but this should be separated from simulating the complete photosynthetic process.

At a higher abstraction level sits the simulation of atomic interactions in organic chemistry and molecular biology, going progressively from the smallest to the largest molecules: amino acids, peptides, polypeptides, proteins and perhaps much later, ultra-complex molecules such as ribosomes that produce proteins with amino acids using messenger RNA code.

Protein-structure prediction and protein design should also be separated from electronic-structure simulation. AlphaFold 3 predicts structures and biomolecular interactions with a classical deep-learning model [835], while RFDiffusion has demonstrated de novo generation of experimentally validated protein structures using classical diffusion models [836]. Quantum simulation is therefore not required for “entirely new” proteins. A future quantum computer may instead help with specific electronic-structure, reaction-energy or strongly correlated subproblems that remain difficult for classical chemistry.

An early quantum-annealing protein-folding experiment was reported in 2012 by Alejandro Perdomo-Ortiz, Neil Dickson, Marshall Drew-Brook, Geordie Rose and Alán Aspuru-Guzik. It mapped coarse-grained lattice-protein energy minimization to a D-Wave quantum annealer and used up to 81 superconducting qubits [838]. It was a proof-of-principle optimization experiment on lattice models, not a general ab initio protein-folding algorithm.

In 2021, Anton Robert, Panagiotis Barkoutsos, Stefan Woerner and Ivano Tavernelli published a resource-efficient variational lattice-protein model. They simulated the 10-amino-acid Angiotensin peptide with a 22-qubit model, while a 7-amino-acid neuropeptide instance used 9 qubits on an IBM 20-qubit processor [839]. The 22-qubit Angiotensin result was a coarse-grained

lattice-model simulation, not a full atomistic prediction of an arbitrary protein’s 3D structure. Algorithms have also been proposed for RNA folding [840].

The orders of magnitude of the quantum computers needed to solve these organic chemistry problems for large proteins have yet to be evaluated. Computing times can be extremely long even with the various optimizations that are proposed [841–843].

Quantum computing can also help simulate classical physics more efficiently than classical computing. For example, a Google AI team created a complex classical oscillators quantum algorithm providing an exponential speedup compared with the best possible classical algorithms [844].

1.3.8 Optimization algorithms

Optimization, search and decision problems are related but distinct. A decision problem asks whether a feasible solution satisfying a condition exists. A search problem asks for such a solution. An optimization problem asks for a feasible solution minimizing or maximizing an objective. TSP, knapsack, graph coloring and SAT have decision and optimization variants, and many standard decision versions are NP-complete.

They are a key domain for quantum computing, whatever the paradigm that is being used or envisioned (analog, NISQ, FTQC) [845, 846]. This is however a challenging one given it is hard to find algorithms with potential exponential speedups that could compete in a practical manner with the best classical solvers around. For example, current implementations of quantum TSP solvers work with only a few qubits and are far from exceeding the capabilities of classical solvers [847, 848].

Classical solvers

Quantum optimization algorithms are frequently compared to classical solutions like IBM CPLEX, Gurobi and Tabu Search as shown in Figure 55.

- **CPLEX** (IBM ILOG CPLEX Optimization Studio) is an optimization software using mathematical programming techniques to solve linear programming (LP), mixed-integer programming (MIP), and other types of optimization problems as a solver which provides different kinds of solutions (exact, bounded, time-limited or approximate). It employs algorithms like simplex and barrier methods for LP, and branch-and-bound and branch-and-cut for MIP. It can handle very large-scale optimization problems and is used in logistics, finance, and manufacturing for tasks like resource allocation, scheduling, and supply chain optimization.
- **Gurobi** is another commercial optimization exact solver designed for solving LP, MIP, and other types of mathematical optimization problems. It utilizes advanced algorithms, including simplex, dual simplex,

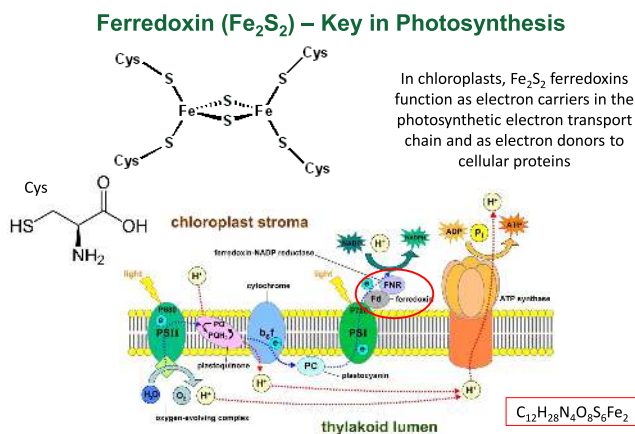


Figure 52: The role of ferredoxin in photosynthesis in plants chloroplasts. Understanding this chemical process could potentially use quantum computer-based simulations. Source: [837].

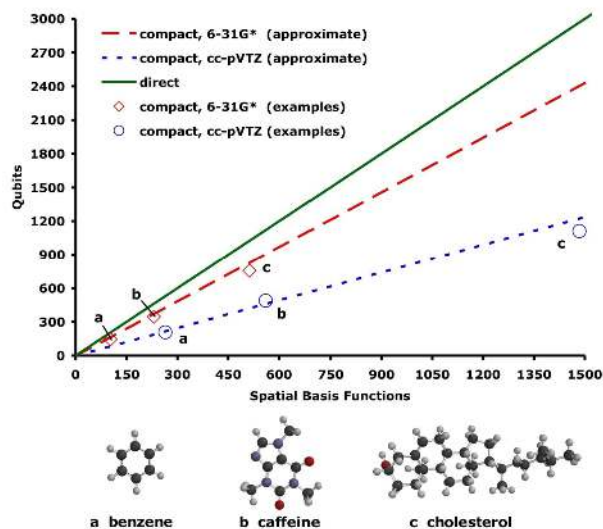


Figure 53: How many logical qubits are actually necessary to simulate relatively simple molecules. Source: [826].

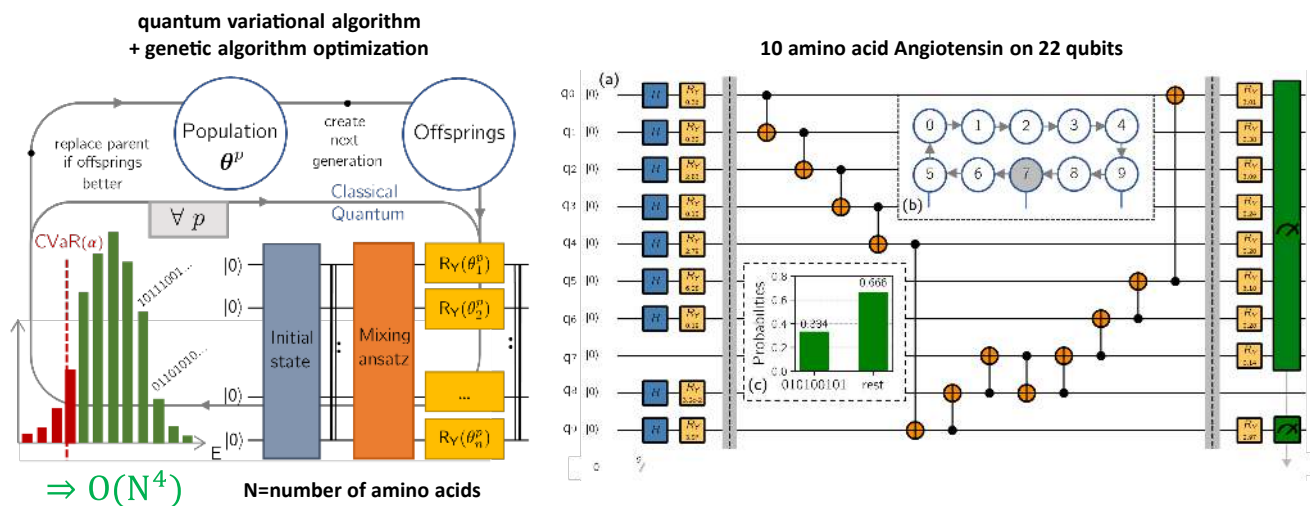


Figure 54: Hybrid classical and quantum lattice-model approach to protein folding. The 10-amino-acid Angiotensin instance was represented with 22 qubits in simulation, while the paper's smaller 7-amino-acid instance was run on IBM hardware. This is not directly comparable to AlphaFold 3, which solves a different classical structure-prediction task. Source: [839].

barrier methods, and various MIP techniques and is optimized for parallel processing, using multi-core processors.

- **Tabu Search** is a metaheuristic optimization algorithm and software designed for solving combinatorial optimization problems. It uses an approximate method to find good, if not optimal, solutions by exploring the solution space more broadly. It uses local search techniques and incorporates memory structures to avoid cycling back to previously visited solutions. It is effective for problems where the solution space is large and complex, and where exact methods might be too time-consuming or infeasible. It does not guarantee an optimal solution.

- **Concorde TSP Solver** is specialized in solving the Traveling Salesperson Problem (TSP). This exact solver was created by David Applegate, Robert E. Bixby, Vašek Chvátal, and William J. Cook. It is based on a mix of branch-and-cut, linear programming (LP) relaxation, cutting planes (or “cuts”) and branch-and-bound algorithms. It can solve TSP problems with up to tens of thousands of points [850], the record being an 85,900 cities TSP problem. Concorde relies on external linear programming (LP) solvers like CPLEX and QSOpt.
- **VeloxQ** is a physics-inspired classical solver developed in Poland by Quantumz.io. It solves QUBO and HUBO problems (defined in 1.4.5) and competes with

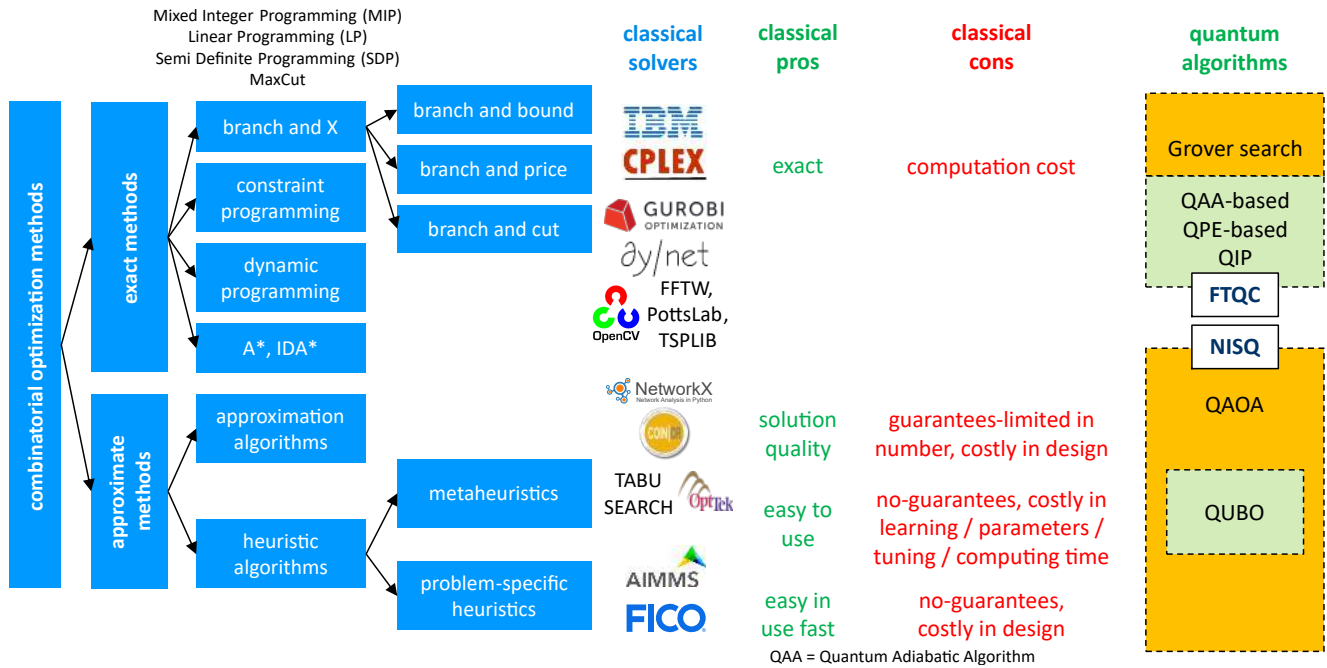


Figure 55: Comparison between different combinatorial optimization methods, their advantages and shortcomings, and where typical quantum algorithms sit and their related paradigm (NISQ, FTQC). QAA = Quantum Adiabatic Algorithm. © Olivier Ezratty, inspired from [849]. Added in August 2024, updated in November 2024.

quantum solvers running on D-Wave and gate-based QPUs, supporting up to 10^8 variables [851].

The classical baseline itself keeps moving, which is easy to forget when quantum speedups are quoted against textbook algorithms. In 2025, Ran Duan et al. broke the $\mathcal{O}(m + n \log n)$ bound of Dijkstra's algorithm on sparse graphs with a deterministic $\mathcal{O}(m \log^{2/3} n)$ single-source shortest path algorithm, showing that Dijkstra is not optimal [852]. Any quantum graph algorithm compared against Dijkstra rather than against the current record is therefore measuring itself against a target that has already moved.

NISQ solvers

We will describe in the NISQ section the QAOA algorithms variants that are proposed for gate-based QPUs and QUBO-based models that are mostly suitable for analog QPUs, both quantum annealers and analog quantum simulators, although QUBO problems can be reformulated into QAOA problems and be run on gate-based QPUs (1.4).

FTQC solvers

Several optimization problem formulations and algorithms are bound to be used on FTQC QPUs which we describe here:

- **Oracle based-search algorithms** that we already covered, including Grover's search algorithm with many variants to solve combinatorial optimization problems (MaxCut, TSP, etc.). Recent variants restrict the search to a structured feasible subspace: a

Grover-based algorithm for cardinality-constrained binary optimization needs $\mathcal{O}(\sqrt{\binom{n}{k}}/M)$ rotations instead of $\mathcal{O}(\sqrt{2^n})$, which is an exponential reduction against unstructured search but not against a good classical solver [853], a nested amplitude amplification protocol splits the binary knapsack decision tree at a tunable depth before running a global Grover adaptive search [854], and a constraint-oriented biased search covers general linear constraints but assumes parallel logical gates running under 10 ns which is currently unrealistic [855]. On the query complexity side, the relative difficulty of path and cycle containment problems has been characterized through randomized reductions in the adjacency matrix model [856].

- **Quantum Adiabatic Algorithm (QAA)** encodes the objective in a problem Hamiltonian and starts from an easier ground state. In the ideal adiabatic limit the system remains in the instantaneous ground-state manifold, but a finite-time computation is not automatically exact. Runtime depends strongly on the minimum spectral gap, schedule and matrix elements, and exponential quantum speedups occur only for particular problem families [857]. Degeneracy and non-stoquastic drivers can alter this behavior [858, 859], but these results can't be generalized to arbitrary optimization instances.
- **QAOA** can be run at shallow depth on NISQ hardware or at much larger depth on future FTQC hardware. Increasing the depth usually raises fidelity requirements and may eventually make error correction necessary. But it won't necessarily solve barren plateau problems.

- **Quantum Monte Carlo Methods** which are used in operational research for solving complex problems involving large sampling using random processes, such as in financial risk and modeling assessments [860]. They require FTQC QPUs to solve large scale problems in a quantum advantage regime. Two 2026 proposals document approach qubit cost. A Szegedy walk following the classical proposal-acceptance logic avoids coherently computing the transition probabilities and needs only a constant-size ancilla register [861]. A fully quantum Metropolis-Hastings for integer linear programming uses $\mathcal{O}(n \log_2 N)$ qubits for n integer variables over $[-N, N - 1]$ with a Toffoli cost per step growing linearly with the logical qubit count, with no qRAM and no classical pre or post-processing [862]. But neither compares runtime costs against an actual ILP solver (Integer linear programming).
- **Quadratically Constrained Quadratic Programs** (QCQPs) is a generic class of problems used to optimize a quadratic objective function subject to quadratic constraints [863]. It contains the Max-Cut problems and various Boolean optimizations, optimal power flow (OPF) problems used in electrical engineering [864] and various financial portfolio optimizations. These can be solved with NISQ (QAOA, VQE) and FTQC (Grover, Quantum Semidefinite Programming solvers) algorithms.
- **Quantum Integer Programming** is an optimization problem formulation to find optimal solutions within a discrete set of possible values. The applications are various, like in supply chain optimization and resource allocation. These problems can be converted into QUBO and QAOA problems running on NISQ QPUs or into Grover search and QPE FTQC versions [865]. Super-quadratic speedups have been hard to obtain here because classical algorithms for many-constraint ILPs are global and exhaustive while the quantum frameworks offering more than quadratic gains exploit local structure. JPMorgan-Chase researchers addressed this in 2026 by targeting Gomory's group relaxation, which drops nonnegativity on the variables that are positive in the LP relaxation optimum while keeping integrality, exposing the local structure the quantum side needs [866]. The result yields the ILP optimum only under a nondegeneracy condition stronger than LP nondegeneracy, otherwise it merely tightens bounds.
- **Quantum least-squares fitting** can be reduced to a linear-system problem in some formulations [867]. HHL-type algorithms can then prepare a quantum state proportional to the solution under assumptions such as sparse or efficiently block-encoded matrices, favorable condition number and efficient state preparation [158]. They do not generically output every fitted coefficient exponentially faster: full classical readout of an N -component solution requires at least $\mathcal{O}(N)$ information extraction. Quantum speedups are therefore most relevant when only a small number

of observables or global properties of the solution are needed.

- **Floquet adiabatic algorithm** to solve Max-Cut problems, which could bring a quantum speedup provided 2,000 logical qubits are available with 2-qubit gate error rates at $\approx 10^{-5}$ [868].
- **Quantum semidefinite programming** (SDP) is a problem formulation to optimize a linear objective function [869]. It can be useful to solve some problems in quantum information science [870]. Various quantum algorithms exist to solve SDP problems: Quantum SDP Solver, Quantum Interior-Point Methods, Quantum Alternating Minimization and Quantum Semidefinite Relaxation Solvers. They all require efficient matrix encoding, are effective for sparse data, and require specific steps to extract classical values out of the resulting vector state for decision-making. Their speedup is polynomial or exponential depending on various conditions.
- **Quantum branch-and-bound algorithms** split a problem into smaller subproblems and compute upper and lower bounds on the optimal solution within a branch to decide whether it can be pruned (i.e. avoided). It is solved with Grover's algorithm, quantum semi-definite programming and QAOA.

Resource estimates. A recent paper by Binkowski et al. proposes an optimized way to encode uniform superpositions of permutations, one use case being solving a TSP problem with n steps [871]. A 100-node TSP can currently be solved classically with the Concorde solvers in a couple seconds on a current laptop. With the quantum algorithms proposed in the paper, it would require 664 logical qubits ($\mathcal{O}(n \log n)$) and a computing depth of 441K ($\mathcal{O}(n^2(\log n)^2)$) (algorithms A and $\tilde{A}$). With a different trade-off between qubits and computing depth, algorithm $\tilde{A}_{oh}$ requires 10K logical qubits ($\mathcal{O}(n^2)$) and 66K computing depth ($\mathcal{O}(n^2 \log n)$). It would correspond respectively to 293 MQops and 664 MQops. The number of circuit shots not being discussed in the paper, it is hard to compare computing times. But at least, if we look at the potential hardware cost, the laptop would probably be a winner. If we extend the challenge further, a 1000-step TSP that can be solved with Concorde in a couple days would require about 10K logical qubits and 1 TQops. This bleak situation doesn't prevent the creation of software tools to help select the right quantum and classical optimizers, like the ProvideQ toolkit from KIT (Germany) and the University of Oxford that was disclosed in 2025 [872]. It was extended in 2026 by a Domain-Specific Language (DSL) to describe the workflow to optimize [873].

In July 2026, Gabbasso et al. from the University of Waterloo (Canada) proposed a way to optimize the quantum formulation of constrained combinatorial problems across all paradigms (analog, NISQ, FTQC) [874]. They demonstrate that the primary driver of quantum

computational slowdown is entanglement restructuring, which involves the rapid creation, re-distribution, and destruction of entanglement during state evolution. Leveraging the algebraic properties of problem constraints allows constraint-aware quantum dynamics to bypass unnecessary entanglement changes, achieving greater efficiency than generic penalty-based methods. One technique is about leveraging symmetry-preserving gates to confine execution to much smaller invariant subspaces. Another is a diabatic fast-jump regime that lets the circuit dynamically bypass narrow spectral gaps without the extensive gate depth otherwise required to restructure entanglement. Algorithm designers are advised to abandon standard penalty-based formulations where possible, explicitly construct feasibility-preserving mixers that map problem symmetries into the circuit architecture. This is however a purely theoretical paper with no indications on practical savings.

Decoded Quantum Interferometry

Decoded Quantum Interferometry (DQI) is an algorithm introduced by Google in 2025 which replaces traditional energy minimum search methods with interferometry-based techniques. It kind of emulates nature. DQI uses a QFT to turn discrete optimization problems like the NP-hard max-LINSAT and max-k-XORSAT optimization problems into classical decoding problems [875] akin to qLDPC error syndrome decoding. These are optimization problems where the goal is to satisfy the largest possible number of linear equations over a finite field (with a set of numbers and operations on these numbers producing results that stay in the set). It brings a potential exponential speedup for finding an approximate optimum of some combinatorial problem [876].

Q Will DQI solve complex optimization problems?

The resource estimates in the DQI Google paper are quite significant [876], sitting between 907 and 41,132 logical qubits and up to 12 TeraQops. Another Google paper from October 2025 estimated that solving an intractable Optimal Polynomial Intersection (OPI) problem [878, 879] requiring 10^{23} classical trials would require 5.72 million Toffoli gates, 1885 logical qubits, 800K physical superconducting qubits and a reasonable runtime of 1 hour [880].

BCG, BMW and the Technical University of Munich confirmed in 2025 the huge FTQC resources needed to obtain some quantum advantage when being compared with the classical solver Gurobi [881]. And another researcher, from Sandia Labs, did demonstrate in 2025 that DQI can't bring an exponential speedup to solve MaxCut problems [882].

In 2026, Infineon researchers benchmarked a blossom and greedy version of DQI against Gurobi and found that Gurobi always beat these to solve a Binary Paint Shop Problem [883]. And DQI scales badly both in space and time.

Other researchers from Berlin and IBM found in 2026 that using classical Markov chain Monte Carlo (MCMC) sampling techniques to solve optimization is competitive with DQI for practical instance sizes [884].

In DQI, the QFT generates amplitudes that interfere constructively on symbol strings for which the objective value is large, enhancing the probability of obtaining good solutions upon measurement. Solving max-LINSAT optimizations can be helpful in various fields: scheduling, production planning, network flow optimization, paint shop problems, portfolio allocation with soft constraints and modeling financial risks. This requires some conversion from the given problem into max-LINSAT and max-k-XORSAT or formulation (Figure 56).

A September 2025 IBM preprint proved that DQI was hard to classically simulate [885]. Another one explained that solving max-k-XORSAT problems with DQI requires the problem to have some structure [886]. Google researchers then extended the reach of DQI to syndrome decoding of Hermitian algebraic geometry error correction codes [887]. Introduced in October 2025 by Google, MIT and EPFL, HDQI (Hamiltonian Decoded Quantum Interferometry) is a generalization of DQI. While DQI focuses on preparing stabilizer states using coherent Bell measurements and classical decoding, HDQI extends it to arbitrary signed Pauli Hamiltonians, enabling the

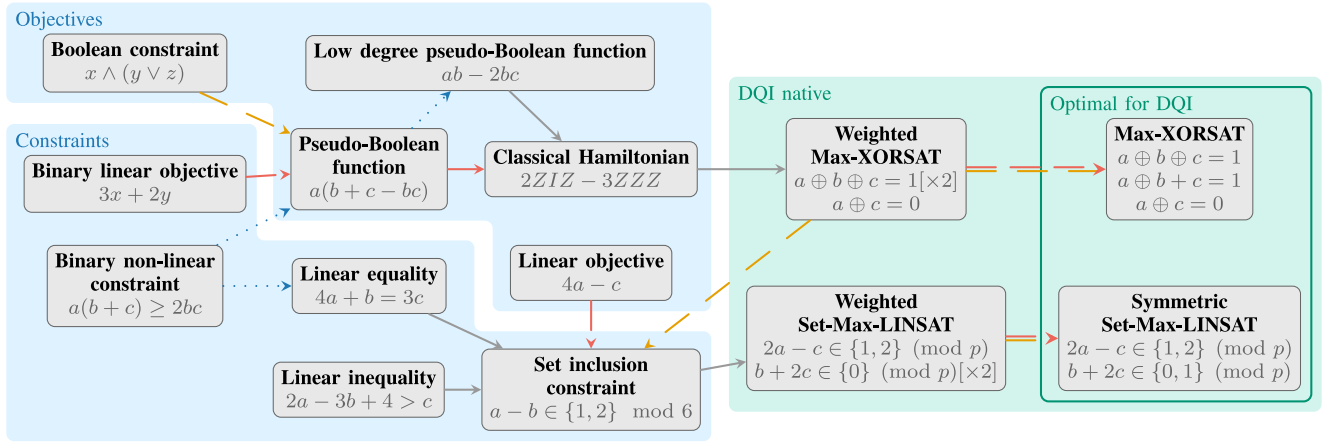


Figure 56: The DQI-Kit introduced in 2026 by Simon Thelen and Wolfgang Mauerer from Germany helps automatically encode constrained optimization problems into Max-LINSAT problems. Users can describe the types of objectives and constraints that are commonplace in industrial optimization problems. Transformations between formulations implemented by the framework: red dashed lines represent transformations that increase the number of constraints, blue dotted lines represent transformations that add additional auxiliary variables, yellow dash-dotted lines represent transformations that lead to many linearly dependent constraints often negatively impacting DQI performance and grey solid lines represent (favorable) transformations where none of the above apply. Source including caption: Figure 1 in [877].

preparation of Gibbs states, ground states, and spectral filters with use cases in quantum simulation for preparing thermal or ground states for benchmarking or dynamics and quantum machine learning for spectral filtering for kernel methods or feature extraction.

Quantum walks

Quantum walks are quantum analogs of classical random walks and form a computational primitive rather than only a class of graph-search algorithms [888, 889]. They support search, graph-property testing, sampling and Hamiltonian-simulation algorithms [890]. Childs showed that continuous-time quantum walk is universal for quantum computation [891]. Hardware demonstrations include superconducting and photonic platforms [892–894].

Classical random walks and Markov chains are tools for search, sampling and mixing. Quantum walks evolve coherently on related graph structures and can reduce hitting, search or mixing complexity for specific problems. The size of the speedup is problem and oracle-model dependent. Some constructed oracle problems exhibit exponential separations, while many important search algorithms obtain quadratic or other polynomial improvements. Quantum-walk techniques can also appear in Boolean-formula and graph algorithms [325].

Theorem [Childs et al '02]

- A continuous-time quantum walk which starts at the entrance (on the LHS) and runs for time $O(\log N)$ finds the exit (on the RHS) with probability at least $1/\text{poly}(\log N)$.
- Any classical algorithm given black-box access to the graph requires $O(N^{1/6})$ queries to find the exit.

Other applications of continuous-time quantum walks:

- Spatial search [Childs and Goldstone '03]
- Evaluation of boolean formulae [Farhi et al '07] [Childs et al '07]

Quantum walks can be used to solve many different search problems, such as:

- Finding a triangle in a graph: $O(n^{1.25})$ queries, vs. classical $O(n^2)$ [Le Gall '14] [Jeffery et al '12] [Magniez et al '03]



- Matrix product verification: $O(n^{5/3})$ queries, vs. classical $O(n^2)$ [Buhrman and Špalek '04]

$$\begin{pmatrix} 1 & 0 & -1 \\ 0 & 2 & 3 \\ -2 & 0 & 1 \end{pmatrix} \times \begin{pmatrix} 0 & 5 & -2 \\ -1 & 1 & 0 \\ 1 & 1 & 1 \end{pmatrix} \stackrel{?}{=} \begin{pmatrix} -1 & 4 & -3 \\ 1 & 5 & 4 \\ 1 & -9 & 5 \end{pmatrix}$$

Figure 57: Quantum walks and their applications.

In gate-based systems, Grover and quantum-walk subroutines can be used in algorithms for shortest paths, graph properties, subgraph finding and clique-related problems, but practical speedup depends on the input model and on oracle construction [895]. Superpolynomial or exponential oracle separations are known for specially structured problems, while broad practical combinatorial optimization has no generic exponential advantage established. A canonical example is the welded-

tree problem, for which a continuous-time quantum walk gives an exponential oracle separation from classical algorithms [896]. The welded tree is a deliberately constructed black-box graph problem. Descriptions linking it directly to logistics, supply-chain routing, drug discovery or materials design should be presented only as speculative analogies, not as demonstrated applications.

Then, you have quantum random walks that help reduce query complexity to search and find graph properties, with the discrete time and continuous time variations [326, 897] (see Figure 58). These are equivalents of the famous Galton’s board.

Q What is the dimension of TSP classical benchmark sets?

The current data set benchmark for solving a TSP approximately is the World TSP from the University of Waterloo which contains 1,904,711 cities [898]. The current best lower bound on the length of a tour for the World TSP is 7,512,218,268 km. It was established with the classical solver CPLEX. But since it seems it is not enough to play with classical solvers, there is also a Galaxy TSP which contains 1.69 billion stars [899].

This gives an idea of the challenge for quantum computers to beat classical solvers in the TSP game, whether to find an exact or approximate solution.

You must also build in the subtleties about introducing constraints in TSP problems, which make it difficult for both classical and quantum solvers, particularly NISQ-based solved using a QUBO formulation, where the U means “unconstrained”.

The Quantum Walk Optimization Algorithm (QWOA) with its non-variational variant seems to potentially bring some useful speedup and with the ability to theoretically scale [901]. The same group benchmarked

it in late 2025 against Lie-algebraic pretraining on unweighted MaxCut at depth $p = 256$: the non-variational variant reaches a mean approximation ratio of 98.9% in 60 iterations with only three hyperparameters, against 77.71% after 500 iterations for the pretrained variant [902]. This is remarkable for so few tunable parameters, but the graphs have 16 vertices and the authors state that scaling remains to be confirmed.

Quantum walks can also handle topological data analysis with some superpolynomial speedup [903]. IonQ researchers argue in 2026 that the narrow advantage regime usually attributed to quantum topological data analysis (TDA) (defined in 1.4.6) comes from targeting exact or high-precision Betti number estimation, and that framing it instead as low-order spectral feature extraction from the combinatorial Laplacian makes the quantum step useful as an input to downstream classical analysis, demonstrated on functional MRI classification and financial time series [904].

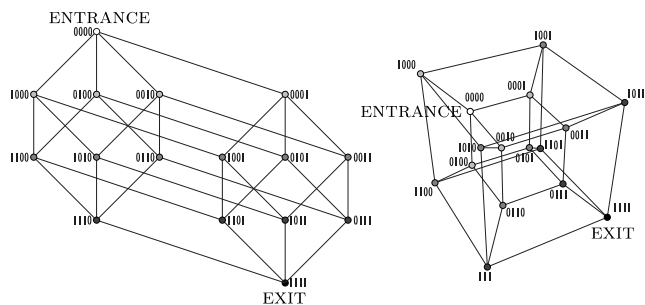


Figure 59: Quantum walks can help solve a shortest path problem. Source: [905], page 13.

Quantum walks can also be implemented in analog mode like with cold atoms, as proposed by Pasqal [906] and researchers from Portugal, the UK, and the Netherlands [907], quantum walks being relatively easy to map to atom graphs in lattices.

On top of Andrew M. Childs, let’s mention several great contributors to the quantum walk domain: **Stacey Jeffery** [908], **Alain Joye** [909], **Julia Kempe** and **Frédéric Magniez**.

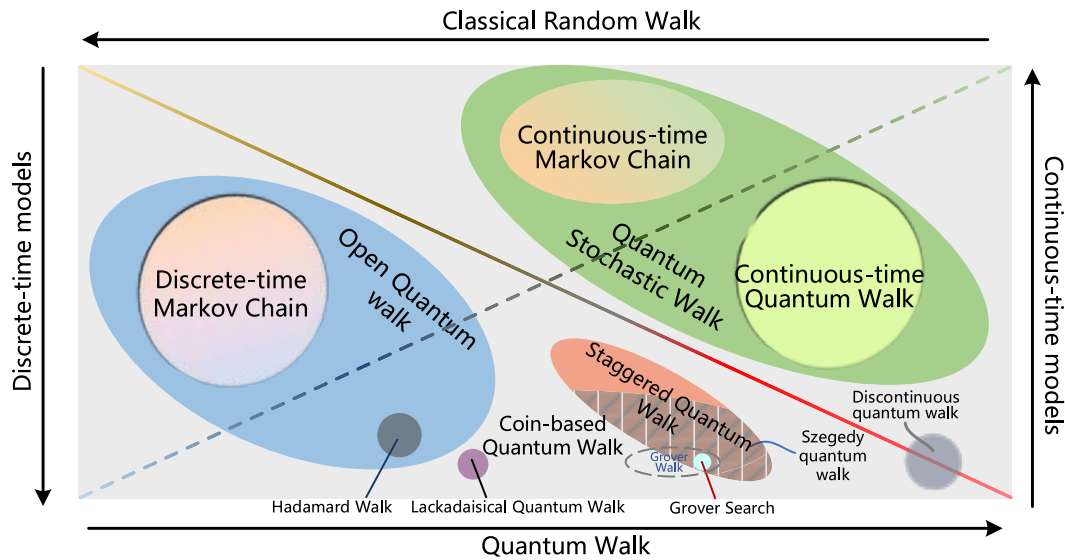
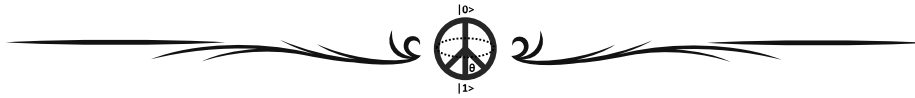


Figure 58: Segmentation of variations of quantum walks. Source: [900].



1.3.9 Bibliography and notes

These are the bibliographical references and notes for the **Higher level algorithms** subsection.

- [15] [Continuous-variable quantum neural networks](#), by Nathan Killoran, Thomas R. Bromley, Juan Miguel Arrazola, Maria Schuld, Nicolás Quesada, and Seth Lloyd, arXiv, June 2018 (21 pages).
- [31] [Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer](#), by Peter W. Shor, SIAM Journal on Computing, October 1997 (26 pages).
- [32] [On the Power of Quantum Computation](#), by Daniel R. Simon, SIAM Journal on Computing, October 1997 (10 pages) 📄.
- [33] [Rapid solution of problems by quantum computation](#), by David Deutsch and Richard Jozsa, Royal Society Proceedings A, December 1992 📄.
- [38] [Polynomial-Time Quantum Algorithms for Pell's Equation and the Principal Ideal Problem](#), by Sean Hallgren, 2006 (21 pages).
- [39] [A Fast Quantum Mechanical Algorithm for Database Search](#), by Lov K. Grover, Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing (STOC '96), May 1996 (8 pages).
- [46] [A Primer on Quantum Machine Learning](#), by Su Yeon Chang and M. Cerezo, arXiv, November 2025 (45 pages).
- [55] [Quantum-state preparation with universal gate decompositions](#), by Martin Plesch and Časlav Brukner, Physical Review A, March 2011 📄 (preprint on [arXiv](#)).
- [151] [Revisiting Shor's quantum algorithm for computing general discrete logarithms](#), by Martin Ekerå, arXiv, May 2019 (32 pages).
- [158] [Quantum algorithm for linear systems of equations](#), by Aram W. Harrow, Avinandan Hassidim, and Seth Lloyd, Physical Review Letters, October 2009 (4 pages).
- [164] [Supervised learning with quantum enhanced feature spaces](#), by Vojtech Havlicek, Jerry M. Chow, Jay M. Gambetta, et al., arXiv, June 2018 (22 pages).
- [300] [Quantum simulation of partial differential equations via Schrodingerisation](#), by Shi Jin, Nana Liu, and Yue Yu, arXiv, December 2022 (9 pages).
- [308] [QCPINN: Quantum-Classical Physics-Informed Neural Networks for Solving PDEs](#), by Afrah Farea, Saiful Khan, and Mustafa Serdar Celebi, arXiv, April 2025 (24 pages).
- [309] [Quantum Physics-Informed Neural Networks for Maxwell's Equations: Circuit Design, "Black Hole" Barren Plateaus Mitigation, and GPU Acceleration](#), by Ziv Chen, Steven H. Frankel, Helen Diller, et al., arXiv, June 2025 (18 pages).
- [318] [Variational quantum algorithms for nonlinear problems](#), by Michael Lubasch, Jaewoo Joo, Pierre Moinier, Martin Kiffner, and Dieter Jaksch, arXiv, December 2019 (8 pages).
- [324] [Quantum Algorithms](#), by Ashley Montanaro, July 2016 (62 slides).
- [325] [Quantum algorithms: an overview](#), by Ashley Montanaro, arXiv, December 2015 (17 pages).
- [326] [Quantum Algorithm Implementations for Beginners](#), by Abhijith J., Andrey Y. Lokhov, et al., arXiv, June 2022 (96 pages).
- [327] [Quantum algorithms: A survey of applications and end-to-end complexities](#), by Alexander M. Dalzell, Fernando G. S. L. Brandão, et al., arXiv, August 2025 (416 pages).
- [328] [Efficient classical simulation of the Deutsch-Jozsa and Simon's algorithms](#), by Niklas Johansson and Jan-Åke Larsson, Quantum Information Processing, August 2017 (14 pages).
- [329] [Associated mathematical formulas](#), by John Watrous, 2006.
- [330] [Signatures of quantum noise in the operation of Deutsch's algorithm](#), by Małgorzata Strzałka and Katarzyna Roszak, arXiv, May 2026 (14 pages).
- [331] [Generalized Deutsch-Jozsa Algorithm for Applications in Data Classification, Logistic Regression, and Quantum Key Distribution](#), by M. Ghadimi, V. Salari, S. Bakrani, M. Zomorodi, N. Gohari-Kamel, S. Moradi, and D. Oblak, arXiv, November 2025 (18 pages).
- [332] [Using Bernstein-Vazirani algorithm to attack block ciphers](#), by Huiqin Xie, 2019 (22 pages).

- [333] [Effects of noise on performance of Bernstein-Vazirani algorithm](#), by Archi Gupta, Priya Ghosh, Kornikar Sen, and Ujjwal Sen, arXiv, August 2024 (15 pages).
- [334] [Demonstration of Algorithmic Quantum Speedup for an Abelian Hidden Subgroup Problem](#), by Phattharaporn Singkanipa, Physical Review X, June 2025 (42 pages).
- [335] [Demonstration of Exponential Quantum Speedup with Constant-Depth Compiled Circuits for Simon's Problem](#), by Phattharaporn Singkanipa, Victor Kasatkin, and Daniel A. Lidar, arXiv, June 2026 (22 pages).
- [336] [Utilizing Novel Quantum Counters for Grover's Algorithm to Solve the Dominating Set Problem](#), by Jehn-Ruey Jiang and Qiao-Yi Lin, arXiv, December 2023 (9 pages).
- [337] [Grover-QAOA for 3-SAT: Quadratic Speedup, Fair-Sampling, and Parameter Clustering](#), by Zewen Zhang, Kaden R. A. Hazzard, et al., arXiv, February 2024 (12 pages).
- [338] [Grover's search meets Ising models: a quantum algorithm for finding low-energy states](#), by Andrey Zhukov, Andrey Lebedev, and Walter Pogosov, arXiv, February 2025 (30 pages).
- [339] [Tight Bounds on Quantum Searching](#), by Michel Boyer, Gilles Brassard, Peter Høyer, and Alain Tapp, Fortschritte der Physik, June 1998 (13 pages).
- [340] [Non-unitary extension of Grover's search algorithm](#), by V. N. A. Lula-Rocha and M. A. S. Trindade, arXiv, April 2026 (35 pages).
- [341] [Grover's search algorithm: An optical approach](#), by P. G. Kwiat, J. R. Mitchell, P. D. D. Schwindt, and A. G. White, arXiv, May 1999 (6 pages).
- [342] [Implementation of quantum search algorithm using classical Fourier optics](#), by N. Bhattacharya, H. B. van Linden van den Heuvell, and R. J. C. Spreeuw, arXiv, February 2002 (4 pages).
- [343] [Classical wave-optics analogy of quantum information processing](#), by Robert J. C. Spreeuw, arXiv, January 2001 (9 pages).
- [344] [A Realizable GAS-based Quantum Algorithm for Traveling Salesman Problem](#), by Jieao Zhu, Yihuai Gao, Hansen Wang, Tiefu Li, and Hao Wu, arXiv, December 2022 (16 pages).
- [345] [Two-Step Quantum Search Algorithm for Solving Traveling Salesman Problems](#), by Rei Sato, Gordon Cui, Kazuhiro Saito, Hideyuki Kawashima, Tetsuro Nikuni, and Shohei Watabe, arXiv, March 2025 (18 pages).
- [346] [Grover's oracle for the Shortest Vector Problem and its application in hybrid classical-quantum solvers](#), by Milos Prokop, Petros Wallden, and David Joseph, arXiv, February 2024 (29 pages).
- [347] [Quantum algorithm for robust optimization via stochastic-gradient online learning](#), by Debbie Lim, João F. Doriguello, and Patrick Rebentrost, arXiv, April 2023 (21 pages).
- [348] [Oracle-Based Robust Optimization via Online Learning](#), by Aharon Ben-Tal, Elad Hazan, Tomer Koren, and Shie Mannor, arXiv, February 2014 (21 pages).
- [349] [Grover Adaptive Search for Constrained Polynomial Binary Optimization](#), by Austin Gilliam, Stefan Woerner, and Constantin Gonciulea, Quantum, April 2021 (11 pages).
- [350] [Grover Adaptive Search with Problem-Specific State Preparation](#), by Maximilian Hess, Lilly Palackal, Abhishek Awasthi, Peter J. Eder, Manuel Schnaus, Laurin Demmler, Karen Wintersperger, and Joseph Doetsch, arXiv, February 2026 (10 pages).
- [351] [Quantum Grover Adaptive Search for Discrete Simulation Optimization](#), by Mingjie Hu, Jian-qiang Hu, and Enlu Zhou, arXiv, April 2026 (37 pages).
- [352] [Coherence Fraction in Grover Search Algorithm](#), by Si-Qi Zhou, Hai Jin, Jin-Min Liang, Shao-Ming Fei, Yunlong Xiao, and Zhihao Ma, arXiv, November 2025 (9 pages).
- [353] [Decoding Quantum Search Advantage: The Critical Role of State Properties in Random Walks](#), by Si-Qi Zhou, Jin-Min Liang, Ziheng Ding, Zhihua Chen, Shao-Ming Fei, and Zhihao Ma, arXiv, November 2025 (7 pages).
- [354] [Reducing Circuit Resources in Grover's Algorithm via Constraint-Aware Initialization](#), by Eunok Bae, Jeonghyeon Shin, and Minjin Choi, arXiv, January 2026 (11 pages).
- [355] [Performance of Uncoded Implementation of Grover's Algorithm on Today's Quantum Processors](#), by Yunos El Kaderi, Andreas Honecker, and Iryna Andriyanova, arXiv, December 2022 (5 pages).
- [356] [Analyses of the viability of automating the quantum circuit construction of Grover Oracle for executing wildcard searches on NISQ processors](#), by Willie Huang, arXiv, March 2023 (11 pages).
- [357] [Better-than-classical Grover search via quantum error detection and suppression](#), by Bibek Pokharel and Daniel Lidar, arXiv, November 2022 (21 pages).
- [358] [Improvement of performance of Grover's algorithm on three generations of Heron family IBM QPUs without and with topological dynamical decoupling](#), by Tihomir G. Tenev, Nayden P. Nedev, and Nikolay V. Vitanov, arXiv, April 2026 (15 pages).
- [359] [Effect of isotropic errors on the complexity of Grover's algorithm](#), by Anurag Saha Roy and Jesus Lacalle, arXiv, June 2026 (19 pages).
- [360] [Quantum algorithms for linear algebra](#), by Anupam Prakash, 2015 (92 slides).
- [361] [Black-box quantum state preparation without arithmetic](#), by Yuval R. Sanders, Dominic W. Berry, et al., arXiv, January 2019 (5 pages).
- [362] [Distributed Exact Generalized Grover's Algorithm](#), by Xu Zhou, Xusheng Xu, Shenggen Zheng, and Le Luo, arXiv, July 2024 (37 pages).
- [363] [Random Grover Search](#), by Dekuan Dong, Yingzhou Li, and Jiaxin Ma, arXiv, June 2026 (23 pages).
- [364] [Opening the Black Box inside Grover's Algorithm](#), by E. Miles Stoudenmire and Xavier Waintal, PRX, November 2024.
- [365] [Of course Grover's algorithm offers a quantum advantage!](#), by Scott Aaronson, March 2023.
- [366] [Xavier Waintal responds \(tl;dr Grover is still quadratically faster\)](#), by Scott Aaronson, March 2023.
- [367] [Disentangling Hype from Practicality: On Realistically Achieving Quantum Advantage](#), by Torsten Hoeffler et al., Communications of the ACM, May 2023.
- [368] [Hardware-Efficient Rydberg Atomic Quantum Solvers for NP Problems](#), by Shuaifan Cao and Xiaopeng Li, arXiv, July 2025 (15 pages).
- [369] [Assessing Quantum and Classical Approaches to Combinatorial Optimization: Testing Quadratic Speed-ups for Heuristic Algorithms](#), by Pedro C. S. Costa, Mauro E.S. Morales, Dong An, and Yuval R. Sanders, arXiv, December 2024 (11 pages).
- [370] [Amplitude amplification and estimation require inverses](#), by Ewin Tang and John Wright, arXiv, July 2025 (20 pages).
- [371] [Approximation Does Not Help in Quantum Unitary Time-Reversal](#), by Kean Chen, Nengkun Yu, and Zhicheng Zhang, STOC '26: Proceedings of the 58th Annual ACM Symposium on Theory of Computing, June 2026 (29 pages)  (preprint on [arXiv](#)).
- [372] [Grover's algorithm for image edge detection](#), by Pulak Ranjan Giri and Kazuhiro Saito, arXiv, July 2026 (2 pages).
- [373] [How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits](#), by Craig Gidney and Martin Ekerå, Quantum, April 2021.
- [374] [Oversimplifying quantum factoring](#), by John A. Smolin, Graeme Smith, and Alexander Vargo, Nature, July 2013  (preprint on [arXiv](#)).

- [375] [Semiclassical Fourier Transform for Quantum Computation](#), by Robert B. Griffiths and Chi-Sheng Niu, Physical Review Letters, April 1996 (4 pages)  (preprint on [arXiv](#)).
- [376] [Truncated Modular Exponentiation Operators: A Strategy for Quantum Factoring](#), by Robert L. Singleton Jr, arXiv, June 2024 (45 pages).
- [377] [The true cost of factoring: Linking magic and number-theoretic complexity in Shor's algorithm](#), by Alessio Paviglianiti, Matteo Seclì, Emanuele Tirrito, and Vincenzo Savona, arXiv, May 2026 (7 pages).
- [378] [Experimental realization of Shor's quantum factoring algorithm using qubit recycling](#), by Enrique Martín-López, Anthony Laing, Thomas Lawson, Roberto Alvarez, Xiao-Qi Zhou, and Jeremy L. O'Brien, Nature Photonics, October 2012 (4 pages)  (preprint on [arXiv](#)).
- [379] [Large-Scale Simulation of Shor's Quantum Factoring Algorithm](#), by Dennis Willsch et al., October 2023 (38 pages).
- [380] [The State of Factoring on Quantum Computers](#), by Dennis Willsch, Philipp Hanussek, Georg Hoever, Madita Willsch, Fengping Jin, Hans De Raedt, and Kristel Michiels, arXiv, May 2025 (14 pages).
- [381] [Shor's Algorithm for Period Finding on a Quantum Computer](#), by Peter Young, December 2019 (23 pages).
- [382] [A Formally Certified End-to-End Implementation of Shor's Factorization Algorithm](#), by Yuxiang Peng, Kesha Hietala, Runzhou Tao, Liyi Li, Robert Rand, Michael Hicks, and Xiaodi Wu, arXiv, April 2022 (18 pages).
- [383] [Quantum factorization of 56153 with only 4 qubits](#), by Nikesh S. Dattani and Nathaniel Bryans, arXiv, November 2014 (6 pages).
- [384] [Analyzing the Performance of Variational Quantum Factoring on a Superconducting Quantum Processor](#), by Amir H. Karamlou, William A. Simon, Amara Katabarwa, Travis L. Scholten, Borja Peropadre, and Yudong Cao, arXiv, March 2021 (14 pages).
- [385] [Comparing the Difficulty of Factorization and Discrete Logarithm: A 240-Digit Experiment](#), by Fabrice Boudot, Pierick Gaudry, Aurore Guillevic, Nadia Heninger, Emmanuel Thomé, and Paul Zimmermann, Advances in Cryptology – CRYPTO 2020, Lecture Notes in Computer Science 12171, August 2020 (30 pages) .
- [386] [Factoring using \$2n+2\$ qubits with Toffoli based modular multiplication](#), by Thomas Häner, Martin Roetteler, and Krysta M. Svore, arXiv, June 2017 (12 pages).
- [387] [How to factor 2048 bit RSA integers with less than a million noisy qubits](#), by Craig Gidney, arXiv, May 2025 (40 pages).
- [388] [Factoring 2048-bit RSA Integers in 177 Days with 13 436 Qubits and a Multimode Memory](#), by Élie Gouzien and Nicolas Sangouard, Physical Review Letters, September 2021  (preprint on [arXiv](#)).
- [389] [Variational Quantum Factoring](#), by Eric R. Anschuetz, Jonathan P. Olson, Alán Aspuru-Guzik, and Yudong Cao, arXiv, August 2018 (18 pages).
- [390] [Implementation of Shor's Algorithm with a Single Photon in 32 Dimensions](#), by Hao-Cheng Weng and Chih-Sung Chu, arXiv, August 2024 (7 pages).
- [391] [Exact quantum Fourier transforms and discrete logarithm algorithms](#), by Michele Mosca and Christof Zalka, arXiv, January 2003 (10 pages).
- [392] [The Hidden Subgroup Problem and Quantum Computation Using Group Representations](#), by Sean Hallgren, Alexander Russell, and Amnon Ta-Shma, SIAM Journal on Computing, January 2003 (19 pages) .
- [393] [Demonstration of Algorithmic Quantum Speedup for an Abelian Hidden Subgroup Problem](#), by P. Singkanipa, V. Kasatkin, Z. Zhou, G. Quiroz, and D.A. Lidar, arXiv, January 2025 (39 pages).
- [394] [Faster by Design: Interactive Aerodynamics via Neural Surrogates Trained on Expert-Validated CFD](#), by Nicholas Thumiger, Andrea Bartezzaghi, Mattia Rigotti, Cezary Skura, Thomas Frick, Elisa Seriola, Fabrizio Arbucci, and A. Cristiano I. Malossi, arXiv, April 2026 (7 pages).
- [395] [Quantum-Inspired Simulation of 2D Turbulent Rayleigh-Bénard Convection](#), by Nis-Luca van Hülst, Mario Guillaume Cecile, Hai-Yen Van, Tomohiro Hashizume, Eugene de Villiers, and Dieter Jaksch, arXiv, April 2026 (14 pages).
- [396] [Quantum computing for simulation of fluid dynamics](#), by Claudio Sanavio and Sauro Succi, arXiv, January 2024 (11 pages).
- [397] [Simulating fluid flows with quantum computing](#), by Sachin S. Bharadwaj and Katepalli R. Sreenivasan, arXiv, September 2024 (17 pages).
- [398] [Nonlinear quantum computing by amplified encodings](#), by Matthias Deiml and Daniel Peterseim, arXiv, November 2024 (21 pages).
- [399] [A robust quantum nonlinear solver based on the asymptotic numerical method](#), by Yongchun Xu, Heng Hu, et al., arXiv, December 2024 (35 pages).
- [400] [An Efficient Explicit Implementation of a Quantum Algorithm with Quantum Advantage for Nonlinear Scalar Conservation Laws](#), by Kezhen Wang, Junpeng Hu, and Lei Zhang, arXiv, August 2026 (29 pages).
- [401] [Efficient Treatment of Non-Linearity in Quantum Computational Fluid Dynamics Using Hybrid Tensor Networks](#), by Pia Siegl, Nis-Luca van Hülst, Maximilian Mandelt Buxadé, Tomohiro Hashizume, and Dieter Jaksch, arXiv, August 2026 (16 pages).
- [402] [Further improving quantum algorithms for nonlinear differential equations via higher-order methods and rescaling](#), by Pedro C. S. Costa, Philipp Schleich, Mauro E. S. Morales, and Dominic W. Berry, arXiv, December 2023 (37 pages).
- [403] [Efficient quantum algorithm for dissipative nonlinear differential equations](#), by Jin-Peng Liu, Herman Øie Kolden, Hari K. Krovi, Nuno F. Loureiro, Konstantina Trivisa, and Andrew M. Childs, arXiv, October 2021 (36 pages).
- [404] [Quantum Carleman linearisation efficiency in nonlinear fluid dynamics](#), by Javier Gonzalez-Conde, Dylan Lewis, Sachin S. Bharadwaj, and Mikel Sanz, arXiv, October 2024 (17 pages).
- [405] [Potential quantum advantage for simulation of fluid dynamics](#), by Xiangyu Li, Physical Review Research, January 2025.
- [406] [Quantum algorithms for general nonlinear dynamics based on the Carleman embedding](#), by David Jennings, Guoming Wang, et al., arXiv, September 2025 (148 pages).
- [407] [Efficient quantum algorithm for solving differential equations with Fourier nonlinearity via Koopman linearization](#), by Judd Katz, Gopikrishnan Muraleedharan, and Abhijeet Alase, arXiv, December 2025 (46 pages).
- [408] [Quantum Algorithms for Nonlinear Differential Equations via Pivot-Shifted Carleman Linearization](#), by Ke Wang, Zikang Jia, Shravan Veerapaneni, and Zhiyan Ding, arXiv, May 2026.
- [409] [Limitations for Quantum Algorithms to Solve Turbulent and Chaotic Systems](#), by Dylan Lewis, Stephan Eidenbenz, Balasubramanya Nadiga, and Yiğit Subaşı, arXiv, October 2024 (21 pages).
- [410] [High order schemes for solving partial differential equations on a quantum computer](#), by Boris Arseniev, Dmitry Guskov, Richik Sengupta, and Igor Zacharov, arXiv, December 2024 (20 pages).
- [411] [Quantum Homotopy Algorithm for Solving Nonlinear PDEs and Flow Problems](#), by Sachin S. Bharadwaj, Balasubramanya Nadiga, Stephan Eidenbenz, and Katepalli R. Sreenivasan, arXiv, December 2025 (24 pages).
- [412] [Quantum algorithms for stochastic nonlinear differential equations](#), by Sergey Bravyi, Adam Byrne, Mykhaylo Zayats, and Sergiy Zhuk, arXiv, June 2026 (95 pages).

- [413] [Efficient Quantum Simulation for Nonlinear Stochastic Differential Equations](#), by Xiangyu Li, Ahmet Burak Catli, Ho Kiat Lim, Matthew Pocrnic, Dong An, Jin-Peng Liu, and Nathan Wiebe, arXiv, March 2026 (70 pages).
- [414] [Quantum simulation of partial differential equations via Schrödingerisation: technical details](#), by Shi Jin, Nana Liu, and Yue Yu, arXiv, December 2022 (37 pages).
- [415] [Schrödinger-Navier-Stokes Equation for the Quantum Simulation of Navier-Stokes Flows](#), by Luca Cappelli, Sauro Succi, Monica Lacatus, Alessandro Zecchi, and Alessandro Roggero, arXiv, April 2026 (15 pages).
- [416] [Hybrid quantum algorithms for flow problems](#), by Sachin S. Bharadwaj and Katepalli R. Sreenivasan, arXiv, November 2023 (21 pages).
- [417] [Quantum algorithm for nonlinear differential equations](#), by Seth Lloyd, Giacomo De Palma, Can Gokler, Bobak Kiani, Zi-Wen Liu, Milad Marvian, Felix Tennie, and Tim Palmer, arXiv, December 2020 (13 pages).
- [418] [Solving nonlinear differential equations with differentiable quantum circuits](#), by Oleksandr Kyriienko, Annie E. Paine, and Vincent E. Elfving, arXiv, May 2021 (22 pages).
- [419] [Quantum Unitary Matrix Representation of Lattice Boltzmann Model for Low Reynolds Fluid Flow Simulation](#), by E. Dinesh Kumar and Steven H. Frankel, arXiv, February 2025 (16 pages).
- [420] [Potential quantum advantage for simulation of fluid dynamics](#), by Xiangyu Li, Xiaolong Yin, Nathan Wiebe, Jaehun Chun, Gregory K. Schenter, Margaret S. Cheung, and Johannes Mülmenstädt, arXiv, March 2024 (30 pages).
- [421] [Fully quantum algorithm for lattice Boltzmann methods with application to partial differential equations](#), by Sriharsha Kocherla, Zhixin Song, Fatima Ezahra Chrit, Bryan Gard, Eugene F. Dumitrescu, Alexander Alexeev, and Spencer H. Bryngelson, arXiv, August 2024 (12 pages).
- [422] [Decomposition of Nonlinear Collision Operator in Quantum Lattice Boltzmann Algorithm](#), by Dinesh Kumar E and Steven H. Frankel, arXiv, August 2024 (5 pages).
- [423] [A quantum algorithm for the lattice-Boltzmann method advection-diffusion equation](#), by David Wawrzyniak et al., January 2025 (16 pages).
- [424] [Quantum lattice Boltzmann algorithm for heat transfer with phase change](#), by Christopher L. Jawetz, Zhixin Song, Spencer H. Bryngelson, and Alexander Alexeev, Quantum Science and Technology, June 2026.
- [425] [Simulating non-trivial incompressible flows with a quantum lattice Boltzmann algorithm](#), by David Jennings, Stephen Rolston, et al., arXiv, December 2025 (25 pages).
- [426] [Towards Quantum Machine Learning of Lattice Boltzmann Collision Operators for Fluid Dynamic Simulations](#), by Wael Itani and Katepalli R. Sreenivasan, arXiv, December 2025 (33 pages).
- [427] [Toward end-to-end quantum simulation of rapidly distorted turbulence](#), by Zhaoyuan Meng, Leyu Chen, Jin-Peng Liu, and Guowei He, arXiv, November 2025 (30 pages).
- [428] [Quantum algorithm for the collision-coalescence of cloud droplets](#), by Kazumasa Ueno and Hiroaki Miura, arXiv, March 2026 (19 pages).
- [429] [Quantum Algorithm For Solving Nonlinear Algebraic Equations](#), by Nhat A. Nghiem and Tzu-Chieh Wei, arXiv, August 2024 (19 pages).
- [430] [Towards Solving the Navier-Stokes Equation on Quantum Computers](#), by Navamita Ray, Tirtha Banerjee, Balasubramanya Nadiga, and Satish Karra, arXiv, April 2019 (16 pages).
- [431] [Solving the Turbine Balancing Problem using Quantum Annealing](#), by Arnold Unterauer, David Bucher, Matthias Knoll, Constantin Economides, Michael Lachner, Thomas Germain, Moritz Kessel, Smajo Hajdinovic, and Jonas Stein, arXiv, May 2024 (6 pages).
- [432] [Quantum annealing-based algorithm for lattice gas automata](#), by Yuichi Kuya, April 2024 (7 pages).
- [433] [Nonlinear dynamics as a ground-state solution on quantum computers](#), by Albert J. Pool, Alejandro D. Somoza, Conor Mc Keever, Michael Lubasch, and Birger Horstmann, arXiv, September 2024 (20 pages).
- [434] [Probing the limits of variational quantum algorithms for nonlinear ground states on real quantum hardware: The effects of noise](#), by Muhammad Umer, Eleftherios Mas-torakis, Sofia Evangelou, and Dimitris G. Angelakis, arXiv, September 2024 (11 pages).
- [435] [Simulating fluid vortex interactions on a superconducting quantum processor](#), by Ziteng Wang, Jiarun Zhong, Ke Wang, Zitian Zhu, Zehang Bao, Chenjia Zhu, Wenwen Zhao, Yaomin Zhao, Yue Yang, Chao Song, and Shiyong Xiong, arXiv, June 2025 (19 pages).
- [436] [Solving the Hele-Shaw flow using the Harrow-Has-sidim-Lloyd algorithm on superconducting devices: A study of efficiency and challenges](#), by Muralikrishnan Gopalakrishnan Meena, Eduardo Antonio Coello Pérez, et al., Physics of FLuids, October 2024 (preprint on arXiv).
- [437] [Quantum machine learning and quantum-inspired methods applied to computational fluid dynamics: a short review](#), by Cesar A. Amaral, Eduardo I. Duzzioni, et al., arXiv, October 2025 (48 pages).
- [438] [Quantum machine learning for efficient reduced order modelling of turbulent flows](#), by Han Li, Yutong Lou, and Dunhui Xiao, arXiv, November 2025 (19 pages).
- [439] [Modeling Noise in Quantum Computing of Scalar Convection](#), by Jiahua Yang, Zhen Lu, and Yue Yang, arXiv, December 2025 (26 pages).
- [440] [Hybrid quantum-classical physics-informed neural networks for solving nonlinear PDEs: when and where hybridization is effective?](#), by Kaveh Zabihi, Hamid Montazeri, and Akke S.J. Suiker, arXiv, June 2026 (18 pages).
- [441] [Geometric Quantum Physics Informed Neural Network](#), by Wai-Hong Tam, Reza Safari, and Hiromichi Matsuyama, arXiv, May 2026 (16 pages).
- [442] [Quantum-Enhanced Convergence of Physics-Informed Neural Networks](#), by Nils Klement, Veronika Eyring, and Mierk Schwabe, arXiv, April 2026 (13 pages).
- [443] [Quantum computing of fluid dynamics using the hydro-dynamic Schrödinger equation](#), by Zhaoyuan Meng et al., PRR, September 2023 (17 pages).
- [444] [Quantum Algorithm for Lattice Boltzmann \(QALB\) Simulation of Incompressible Fluids with a Nonlinear Collision Term](#), by Wael Itani, Katepalli R. Sreenivasan, and Sauro Succi, arXiv, April 2023 (62 pages).
- [445] [Ensemble Fluid Simulations on Quantum Computers](#), by Sauro Succi, Wael Itani, Katepalli R. Sreenivasan, and Rene Steijl, arXiv, April 2023 (8 pages).
- [446] [Feasibility of accelerating incompressible computational fluid dynamics simulations with fault-tolerant quantum computers](#), by John Penuel, Amara Katabarwa, Peter D. Johnson, Collin Farquhar, Yudong Cao, and Michael C. Garrett, arXiv, September 2024 (70 pages).
- [447] [A Pathway to Practical Quantum Advantage in Solving Navier-Stokes Equations](#), by Xi-Ning Zhuang, Guo-Ping Guo, et al., arXiv, September 2025 (57 pages).
- [448] [Measurement Schemes for Quantum Linear Equation Solvers](#), by Andrew Patterson and Leigh Lapworth, arXiv, March 2025 (23 pages).
- [449] [Resource Implications of Different Encodings for Quantum Computational Fluid Dynamics](#), by Hans A. Kösel, Roland Ewert, and Jan W. Delfs, arXiv, April 2026 (28 pages).
- [450] [Real and Fourier space readout methods: Comparison of complexity and applications to CFD problems](#), by Xinchu Huang, Hirofumi Nishi, Yoshifumi Kawada, Tomofumi Zushi, and Yu-ichiro Matsushita, arXiv, November 2025 (41 pages).

- [451] [An HHL-Based Quantum-Classical Solver for the Incompressible Navier-Stokes Equations with Approximate QST](#), by Moshe Inger and Steven Frankel, arXiv, March 2026 (16 pages).
- [452] [Quantum lower bounds for simulating fluid dynamics](#), by Abtin Ameri, Joseph Carolan, Andrew M. Childs, and Hari Krovi, arXiv, March 2026 (52 pages).
- [453] [Magic of the Well: assessing quantum resources of fluid dynamics data](#), by Antonio Francesco Mello, Mario Colura, E. Miles Stoudenmire, and Ryan Levy, arXiv, December 2025 (5 pages).
- [454] [Quantum Computing and Artificial Intelligence Use Cases](#), by (QED-C), QED-C, March 2025 (33 pages).
- [455] [Quantum Artificial Intelligence: A Brief Survey](#), by Matthias Klusch, Jörg Lässig, Daniel Müssig, Antonio Macaluso, and Frank K. Wilhelm, arXiv, August 2024 (21 pages).
- [456] [Quantum computing and artificial intelligence: status and perspectives](#), by Giovanni Acampora, Andris Ambainis, Natalia Ares, Leonardo Banchi, Pallavi Bhardwaj, Daniele Binosi, G. Andrew D. Briggs, Tommaso Calarco, Vedran Dunjko, Jens Eisert, Olivier Ezratty, Paul Erker, Federico Fedele, Elies Gil-Fuster, Martin Gärtner, Mats Granath, Markus Heyl, Iordanis Kerenidis, Matthias Klusch, Anton Frisk Kockum, Richard Kueng, Mario Krenn, Jörg Lässig, Antonio Macaluso, Sabrina Maniscalco, Florian Marquardt, Kristel Michielsen, Gorka Muñoz-Gil, Daniel Müssig, Hendrik Poulsen Nautrup, Evert van Nieuwenburg, Roman Orus, Jörg Schmiedmayer, Markus Schmitt, Philipp Slusalek, Filippo Vicentini, Christof Weitenberg, and Frank K. Wilhelm, arXiv, May 2025 (32 pages).
- [457] [The Virtuous Cycle of Quantum-Classical Machine Learning](#), by Nenad Tomašev, Jarrod R. McClean, and Johannes Bausch, arXiv, July 2026 (40 pages).
- [458] [Quantum artificial intelligence: A survey](#), by Giovanni Acampora, Angela Chiatto, Roberto Schiattarella, and Autilia Vitiello, Computer Science Review, February 2026 (35 pages).
- [459] [Research progress on quantum neural networks and quantum machine learning](#), by Yifan Sun, Boyuan Sun, Jiameng Tian, and Xiangdong Zhang, arXiv, May 2026 (47 pages).
- [460] [Quantum Large Language Model Fine-Tuning](#), by Sang Hyub Kim, Martin Roetteler, et al., arXiv, April 2025 (11 pages).
- [461] [The generative quantum eigensolver \(GQE\) and its application for ground state search](#), by Kouhei Nakaji, Alan Aspuru-Guzik, et al., arXiv, January 2024 (16 pages).
- [462] [Quantum Feature Maps for Graph Machine Learning on a Neutral Atom Quantum Processor](#), by Boris Albrecht, Constantin Dalyac, Louis-Paul Henry, Adrien Signoles, Loïc Henriët, et al., arXiv, November 2022 (19 pages).
- [463] [Quantum Machine Learning for Complex Systems](#), by Vinit Singh, Amandeep Singh Bhatia, Mandeep Kaur Saggi, Manas Sajjan, and Sabre Kais, arXiv, February 2026.
- [464] [Molecular Design beyond Training Data with Novel Extended Objective Functionals of Generative AI Models Driven by Quantum Annealing Computer](#), by Hayato Kunugi, Mohsen Rahmani, Yosuke Iyama, Yutaro Hirano, Akira Suma, Matthew Woolway, Vladimir Vargas-Calderón, William Kim, Kevin Chern, Mohammad Amin, and Masaru Tateno, arXiv, May 2026 (28 pages).
- [465] [Quantum machine learning](#), by Jacob Biamonte, Peter Wittek, Nicola Pancotti, Patrick Rebentrost, Nathan Wiebe, and Seth Lloyd, Nature, September 2017 (8 pages)  (preprint on arXiv).
- [466] [Is quantum advantage the right goal for quantum machine learning?](#), by Maria Schuld and Nathan Killoran, arXiv, February 2023 (13 pages).
- [467] [Quantum machine learning: a classical perspective](#), by Ciliberto, 2020 (26 pages).
- [468] [A comprehensive review of Quantum Machine Learning: from NISQ to Fault Tolerance](#), by Yunfei Wang and Junyu Liu, arXiv, March 2024 (28 pages).
- [469] [Quantum Machine Learning on Near-Term Quantum Devices: Current State of Supervised and Unsupervised Techniques for Real-World Applications](#), by Yaswitha Gujju, Atsushi Matsuo, and Rudy Raymond, arXiv, June 2024 (40 pages).
- [470] [Towards provably efficient quantum algorithms for large-scale machine-learning models](#), by Junyu Liu, Minzhao Liu, Jin-Peng Liu, Ziyu Ye, Yunfei Wang, Yuri Alexeev, Jens Eisert, and Liang Jiang, Nature Communications, January 2024.
- [471] [An Introduction to Quantum Machine Learning for Engineers](#), by Osvaldo Simeone, arXiv, June 2022 (229 pages).
- [472] [Universal Variational Quantum Computation](#), by Jacob Biamonte, arXiv, December 2020 (12 pages).
- [473] [Accelerated Variational Quantum Eigensolver](#), by Daochen Wang, Oscar Higgott, and Stephen Brierley, arXiv, March 2019 (11 pages).
- [474] [Quantum Machine Learning What Quantum Computing Means to Data Mining](#), by Peter Wittek, 2014 (178 pages).
- [475] [Quantum advantage with shallow circuits](#), by Robert König, 2018 (97 slides).
- [476] [Classically Approximating Variational Quantum Machine Learning with Random Fourier Features](#), by Jonas Landman, Slimane Thabet, Constantin Dalyac, Hela Mhiri, and Elham Kashefi, arXiv, October 2022 (17 pages).
- [477] [Comprehensive Survey of QML: From Data Analysis to Algorithmic Advancements](#), by Sahil Tomar, Rajeshwar Tripathi, and Sandeep Kumar, arXiv, January 2025 (88 pages).
- [478] [Parameterized quantum circuits as machine learning models](#), by Marcello Benedetti, Erika Lloyd, Stefan Sack, and Mattia Fiorentini, arXiv, October 2019 (18 pages).
- [479] [Machine learning applications for noisy intermediate-scale quantum computers](#), by Brian Coyle, arXiv, May 2022 (263 pages).
- [480] [Optimal number of parametrized rotations and Hadamard gates in parametrized Clifford circuits with non-repeated parameters](#), by Vivien Vandaele, Simon Perdrix, and Christophe Vuillot, arXiv, July 2024 (26 pages).
- [481] [QNAS: A Neural Architecture Search Framework for Accurate and Efficient Quantum Neural Networks](#), by Kooshan Maleki, Alberto Marchisio, and Muhammad Shafique, arXiv, April 2026 (8 pages).
- [482] [Learnable yet not simulable: a quantum resource theory of learning models](#), by Xinbiao Wang, Yuxuan Du, and Dacheng Tao, arXiv, August 2026 (61 pages).
- [483] [Quantum Architecture Search: A Survey](#), by Darya Martyniuk, Johannes Jung, and Adrian Paschke, arXiv, June 2024 (12 pages).
- [484] [TensorRL-QAS: Reinforcement learning with tensor networks for scalable quantum architecture search](#), by Akash Kundu and Stefano Mangini, arXiv, May 2025 (27 pages).
- [485] [Introduction to Quantum Machine Learning and Quantum Architecture Search](#), by Samuel Yen-Chi Chen and Zhiding Liang, arXiv, April 2025 (6 pages).
- [486] [Regression of Functions by Quantum Neural Networks Circuits](#), by Fernando M. de Paula Neto, Lucas dos Reis Silva, Paulo S. G. de Mattos Neto, and Felipe F. Fanchini, arXiv, December 2025 (79 pages).
- [487] [Quantum Machine Learning - State of the Art and Future Directions](#), by Christian Bauckhage et al., 2023 (122 pages).
- [488] [Differentiable Quantum Architecture Search in Quantum-Enhanced Neural Network Parameter Generation](#), by Samuel Yen-Chi Chen, Chen-Yu Liu, Kuan-Cheng Chen, Wei-Jia Huang, Yen-Jui Chang, and Wei-Hao Huang, arXiv, May 2025 (11 pages).

- [489] [Quantum-Train: Rethinking Hybrid Quantum-Classical Machine Learning in the Model Compression Perspective](#), by Chen-Yu Liu, En-Jui Kuo, Chu-Hsuan Abraham Lin, Jason Gemsun Young, Yeong-Jar Chang, Min-Hsiu Hsieh, and Hsi-Sheng Goan, arXiv, June 2024 (12 pages).
- [490] [Train on classical, deploy on quantum: scaling generative quantum machine learning to a thousand qubits](#), by Erik Recio-Armengol, Shah Nawaz Ahmed, and Joseph Bowles, arXiv, March 2025 (50 pages).
- [491] [Quantum Machine Learning: A Hands-on Tutorial for Machine Learning Practitioners and Researchers](#), by Yuxuan Du, Xinbiao Wang, Naixu Guo, Zhan Yu, Yang Qian, Kaining Zhang, Min-Hsiu Hsieh, Patrick Rebentrost, and Dacheng Tao, arXiv, February 2025 (260 pages).
- [492] [New perspectives on quantum kernels through the lens of entangled tensor kernels](#), by Seongwook Shin, Ryan Sweke, and Hyunseok Jeong, arXiv, March 2025 (21 pages).
- [493] [Towards Practical Quantum Neural Network Diagnostics with Neural Tangent Kernels](#), by Francesco Scala, Francesco Tacchino, et al., arXiv, March 2025 (8 pages).
- [494] [Quantum Kernel Methods: Convergence Theory, Separation Bounds and Applications to Marketing Analytics](#), by Laura Sáez-Ortuño, Santiago Forgas-Coll, and Massimiliano Ferrara, arXiv, October 2025 (15 pages).
- [495] [Noise-enhanced quantum kernels on analog quantum computers](#), by Hsiang-Wei Huang, Shen-Liang Yang, Chuan-Chi Huang, Yueh-Nan Chen, and Hong-Bin Chen, arXiv, April 2026 (15 pages).
- [496] [Rydberg Vision via frugal Quantum Image Fingerprinting](#), by Vikrant Sharma and Neel Kanth Kundu, arXiv, March 2026 (13 pages).
- [497] [Large-Scale Quantum Kernels for Hyperspectral Data Classification](#), by A. Delilbasic, A. Miroszewski, A. Wijata, J. Nalepa, J. Mielczarek, M. Riedel, and G. Cavallaro, arXiv, May 2026 (21 pages).
- [498] [Benchmarking Quantum Kernel Support Vector Machines Against Classical Baselines on Tabular Data: A Rigorous Empirical Study with Hardware Validation](#), by Siavash Kakavand, Christoph Strohmeier, and Michael Schlotter, arXiv, April 2026 (45 pages).
- [499] [Quantum Kernel Advantage over Classical Collapse in Medical Foundation Model Embeddings](#), by Sebastian Cajas Ordóñez, Felipe Ocampo Osorio, Dax Enshan Koh, Rafi Al Attrach, Aldo Marzullo, Ariel Guerra-Adames, J. Alejandro Andrade, Siong Thye Goh, Chi-Yu Chen, Rahul Gorijavolu, Xue Yang, Noah Dane Hebdon, and Leo Anthony Celi, arXiv, April 2026 (19 pages).
- [500] [Support Vector Machines on Noisy Intermediate Scale Quantum Computers](#), by Jiaying Yang, Ahsan Javed Awan, and Gemma Vall-Lllosera, arXiv, September 2019 (12 pages).
- [501] [Quantum support vector machines for aerodynamic classification](#), by Xi-Jun Yuan, August 2022 (12 pages).
- [502] [Investigating the Relation Between Problem Hardness and QUBO Properties](#), by Thore Gerlach and Sascha Mücke, arXiv, April 2024 (12 pages).
- [503] [Local Binary and Multiclass SVMs Trained on a Quantum Annealer](#), by Enrico Zardini, Amer Delilbasic, Enrico Blanzieri, Gabriele Cavallaro, and Davide Pastorello, arXiv, March 2024 (12 pages).
- [504] [Unsupervised Feature Extraction and Reconstruction Using Parameterized Quantum Circuits](#), by Li-An Lo, Li-Yi Hsu, and En-Jui Kuo, arXiv, February 2025 (25 pages).
- [505] [QuFeX: Quantum feature extraction module for hybrid quantum-classical deep neural networks](#), by Naman Jain and Amir Kalev, arXiv, January 2025 (12 pages).
- [506] [Quantum Phases Classification Using Quantum Machine Learning with SHAP-Driven Feature Selection](#), by Giovanni S. Franco, Felipe F. Fanchini, et al., arXiv, April 2025 (10 pages).
- [507] [Quantum principal component analysis](#), by Seth Lloyd, Masoud Mohseni, and Patrick Rebentrost, arXiv, September 2013 (9 pages).
- [508] [HarmoniQ: Efficient Data Augmentation on a Quantum Computer Inspired by Harmonic Analysis](#), by Kristina Kirova, Monika Doerfler, Franz Luef, and Richard Kueng, arXiv, July 2026.
- [509] [Evaluating the Potential of Quantum Machine Learning in Cybersecurity: A Case-Study on PCA-based Intrusion Detection Systems](#), by Armando Bellante, Tommaso Fioravanti, Michele Carminati, Stefano Zanero, and Alessandro Luongo, arXiv, February 2025 (29 pages).
- [510] [Self-Adaptive Quantum Kernel Principal Component Analysis for Compact Readout of Chemiresistive Sensor Arrays](#), by Zeheng Wang et al., January 2025 (10 pages).
- [511] [qLUE: A Quantum Clustering Algorithm for Multi-Dimensional Datasets](#), by Dhruv Gopalakrishnan, Luca Dellantonio, Antonio Di Pilato, Wahid Redjeb, Felice Pantaleo, and Michele Mosca, arXiv, July 2024 (17 pages).
- [512] [Provably faster randomized and quantum algorithms for k-means clustering via uniform sampling](#), by Tyler Chen, Marco Pistoia, et al., arXiv, May 2025 (30 pages).
- [513] [Quantum Spectral Clustering](#), by Iordanis Kerenidis and Jonas Landman, arXiv, June 2021 (20 pages).
- [514] [Quantum Clustering for Cybersecurity](#), by Walid El Maouaki, Nouhaila Innan, Alberto Marchisio, Taoufik Said, Mohamed Bennai, and Muhammad Shafique, arXiv, August 2024 (6 pages).
- [515] [Quantum Machine Learning in Log-based Anomaly Detection: Challenges and Opportunities](#), by Jiaying Qi, Chang Zeng, Zhongzhi Luan, Shaohan Huang, Shu Yang, Yao Lu, Bin Han, Hailong Yang, and Depei Qian, arXiv, December 2024 (13 pages).
- [516] [Quantum machine learning algorithms for anomaly detection: A review](#), by Sebastiano Corli, Lorenzo Moro, Daniele Dragoni, Massimiliano Dispenza, and Enrico Prati, arXiv, March 2025 (60 pages).
- [517] [Nonlinear regression based on a hybrid quantum computer](#), by Dan-Bo Zhang, Shi-Liang Zhu, and Z. D. Wang, arXiv, August 2018 (6 pages).
- [518] [Des-q: a quantum algorithm to provably speedup retraining of decision trees](#), by Niraj Kumar, Romina Yalovetzky, Changhao Li, Pierre Minssen, and Marco Pistoia, arXiv, January 2025 (45 pages).
- [519] [QC-Forest: a Classical-Quantum Algorithm to Provably Speedup Retraining of Random Forest](#), by Romina Yalovetzky, Niraj Kumar, Changhao Li, and Marco Pistoia, arXiv, July 2024 (24 pages).
- [520] [Quantum Recommendation Systems](#), by Iordanis Kerenidis and Anupam Prakash, arXiv, September 2016 (22 pages).
- [521] [Exponential Advantages in Quantum Machine Learning through Feature Mapping](#), by Andrew Nader, December 2020 (16 pages).
- [522] [Quantum ensemble learning with a programmable superconducting processor](#), by Jiachen Chen, H. Wang, et al., arXiv, March 2025 (9 pages).
- [523] [Quantum Deep Learning: A Comprehensive Review](#), by Yanjun Ji, Zhao-Yun Chen, Marco Roth, David A. Kreplin, Christian Schiffer, Martin King, Oliver Anton, M. Sahnawaz Alam, Markus Krutzik, Dennis Willsch, Ludwig Mathey, Frank K. Wilhelm, and Guo-Ping Guo, arXiv, February 2026 (69 pages).
- [524] [An Artificial Neuron Implemented on an Actual Quantum Processor](#), by Francesco Tacchino, Chiara Macchiavella, Dario Gerace, and Daniele Bajoni, arXiv, November 2018 (8 pages).
- [525] [Training Multilayer Perceptrons by Sampling with Quantum Annealers](#), by Frances Fengyi Yang, Michele Sasdelli, and Tat-Jun Chin, arXiv, March 2023 (22 pages).

- [526] [Learning Quantum Processes with Memory – Quantum Recurrent Neural Networks](#), by Dmytro Bondarenko, Robert Salzmann, and Viktoria-S. Schmiesing, arXiv, January 2023 (5 pages).
- [527] [Long Short-Term Memory](#), by Sepp Hochreiter and Jürgen Schmidhuber, Neural Computation, November 1997 (46 pages).
- [528] [Quantum Kernel-Based Long Short-term Memory for Climate Time-Series Forecasting](#), by Yu-Chao Hsu, Nan-Yow Chen, Tai-Yu Li, Po-Heng Lee, and Kuan-Cheng Chen, arXiv, December 2024 (8 pages).
- [529] [Toward Large-Scale Distributed Quantum Long Short-Term Memory with Modular Quantum Computers](#), by Kuan-Cheng Chen, Samuel Yen-Chi Chen, Chen-Yu Liu, and Kin K. Leung, arXiv, March 2025 (6 pages).
- [530] [Quantum algorithms for feedforward neural networks](#), by Jonathan Allcock, Chang-Yu Hsieh, Iordanis Kerenidis, and Shengyu Zhang, arXiv, September 2019 (24 pages).
- [531] [Quantum Circuit Parameters Learning with Gradient Descent Using Backpropagation](#), by M Watabe, 2020 (15 pages).
- [532] [Deep quantum neural networks equipped with backpropagation on a superconducting processor](#), by Xiaoxuan Pan, Luyan Sun, et al., arXiv, December 2022 (7 pages).
- [533] [Quantum Machine Learning: from physics to software engineering](#), by Alexey Melnikov, Mohammad Kordzanganeh, Alexander Alodjants, and Ray-Kuang Lee, arXiv, March 2023 (46 pages).
- [534] [On quantum backpropagation, information reuse, and cheating measurement collapse](#), by Amira Abbas, Jarrod R. McClean, et al., arXiv, May 2023 (29 pages).
- [535] [Quantum Convolutional Neural Networks](#), by Iris Cong, Soonwon Choi, and Mikhail D. Lukin, arXiv, May 2019 (12 pages).
- [536] [Quantum Convolutional Neural Networks: Powering Image Recognition with Quantum Circuits](#), by Maxwell Henderson, Samridhi Shakya, Shashindra Pradhan, and Tristan Cook, Quantum Machine Intelligence, June 2020 (9 pages)  (preprint on [arXiv](#)).
- [537] [Absence of Barren Plateaus in Quantum Convolutional Neural Networks](#), by Arthur Pesah et al., PRX, November 2021 (26 pages).
- [538] [Quantum Convolutional Neural Networks are Effectively Classically Simulable](#), by Pablo Bermejo, Paolo Braccia, Manuel S. Rudolph, Zoë Holmes, Lukasz Cincio, and M. Cerezo, PRX Quantum, June 2026 (13 pages).
- [539] [Challenges for Reinforcement Learning in Quantum Circuit Design](#), by Philipp Altmann, Jonas Stein, Michael Kölle, Adelina Bärligea, Thomas Gabor, Thomy Phan, Sebastian Feld, and Claudia Linnhoff-Popien, arXiv, August 2024 (11 pages).
- [540] [A Survey on Quantum Reinforcement Learning](#), by Nico Meyer, Christian Ufrecht, Maniraman Periyasamy, Daniel D. Scherer, Axel Plinge, and Christopher Mutschler, arXiv, March 2024 (83 pages).
- [541] [Experimental quantum speed-up in reinforcement learning agents](#), by V. Saggio, Dirk Englund, Hans J. Briegel, P. Walther, et al., Nature, March 2021  (preprint on [arXiv](#)).
- [542] [VQC-Based Reinforcement Learning with Data Re-uploading: Performance and Trainability](#), by Rodrigo Coelho, André Sequeira, and Luís Paulo Santos, arXiv, November 2024 (26 pages).
- [543] [Quantum Advantage Actor-Critic for Reinforcement Learning](#), by Michael Kölle, Mohamad Hgog, Fabian Ritz, Philipp Altmann, Maximilian Zorn, Jonas Stein, and Claudia Linnhoff-Popien, arXiv, January 2024 (12 pages).
- [544] [Model-based Offline Quantum Reinforcement Learning](#), by Simon Eisenmann, Daniel Hein, Steffen Udluft, and Thomas A. Runkler, arXiv, April 2024 (16 pages).
- [545] [A Study on Optimization Techniques for Variational Quantum Circuits in Reinforcement Learning](#), by Michael Kölle, Timo Witter, Tobias Rohe, Gerhard Stenzel, Philipp Altmann, and Thomas Gabor, arXiv, May 2024 (11 pages).
- [546] [Training Hybrid Deep Quantum Neural Network for Efficient Reinforcement Learning](#), by Jie Luo, Xueyin Chen, Jeremy Kulcsar, Georgios Korpas, and Giulio Giacconi, arXiv, June 2025 (26 pages).
- [547] [Quantum-Train-Based Distributed Multi-Agent Reinforcement Learning](#), by Kuan-Cheng Chen, Samuel Yen-Chi Chen, Chen-Yu Liu, and Kin K. Leung, arXiv, December 2024 (5 pages).
- [548] [Quantum Reinforcement Learning: Recent Advances and Future Directions](#), by Jawaher Kaldari, Hyundong Shin, et al., arXiv, October 2025 (23 pages).
- [549] [Q-SpiRL: Quantum Spiking Reinforcement Learning for Adaptive Robot Navigation](#), by Mohamed Khair Altrabulsi, Nouhaila Innan, Alberto Marchisio, Muhammad Kashif, and Muhammad Shafique, arXiv, July 2026 (11 pages).
- [550] [Enhanced Reinforcement Learning-based Process Synthesis via Quantum Computing](#), by Austin Braniff, Fengqi You, Yuhe Tian Department of Chemical, Biomedical Engineering, West Virginia University, R.F. Smith School of Chemical, Biomolecular Engineering, and Cornell University, arXiv, May 2026 (41 pages).
- [551] [Benchmarking Quantum Reinforcement Learning](#), by Nico Meyer, Christian Ufrecht, George Yammine, Georgios Kontes, Christopher Mutschler, and Daniel D. Scherer, arXiv, May 2025 (31 pages).
- [552] [Introduction to Robust Machine Learning with Geometric Methods for Defense Applications](#), by Pierre-Yves Lagrave et al., July 2021 (9 pages).
- [553] [Deep Residual Learning for Image Recognition](#), by Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, arXiv, December 2015 (12 pages).
- [554] [ResQ: A Novel Framework to Implement Residual Neural Networks on Analog Rydberg Atom Quantum Computers](#), by Nicholas S. DiBrita, Jason Han, and Tirthak Patel, arXiv, June 2025 (11 pages).
- [555] [Graph Neural Networks on Quantum Computers](#), by Yidong Liao, Xiao-Ming Zhang, and Chris Ferrie, arXiv, May 2024 (50 pages).
- [556] [Quantum Graph Neural Networks](#), by Guillaume Verdon, Trevor McCourt, Enxhell Luzhnica, Vikash Singh, Stefan Leichenauer, and Jack Hidary, arXiv, September 2019 (8 pages).
- [557] [Quantum neural networks](#), by Kerstin Beer, arXiv, May 2022 (189 pages).
- [558] [Scalable Message-Passing Quantum Graph Neural Networks in the Weisfeiler-Leman Hierarchy](#), by Snehal Raj, Brian Coyle, Léo Monbroussou, André J. Ferreira-Martins, Renato M. S. Farias, and Elham Kashefi, arXiv, June 2026 (10 pages).
- [559] [From Graphs to Qubits: A Critical Review of Quantum Graph Neural Networks](#), by Andrea Ceschini, Francesco Mauro, Francesca De Falco, Alessandro Sebastianelli, Alessio Verdone, Antonello Rosato, Bertrand Le Saux, Massimo Panella, Paolo Gamba, and Silvia L. Ullo, arXiv, August 2024 (21 pages).
- [560] [Quantum Graph Convolutional Networks Based on Spectral Methods](#), by Zi Ye, Kai Yu, and Song Lin, arXiv, March 2025 (7 pages).
- [561] [QGShap: Quantum Acceleration for Faithful GNN Explanations](#), by Haribandhu Jena, Jyotirmaya Shivottam, and Subhankar Mishra, arXiv, December 2025.
- [562] [Quantum Graph Convolutional Networks: Implementation and Trainability Analysis](#), by Paul San Sebastian Sein, Theodor Iosif, Tilen G. Limbäck-Stokin, Kin Ian Lo, and Yidong Liao, arXiv, September 2026 (69 pages).

- [563] [Classical-to-quantum convolutional neural network transfer learning](#), by Juhyeon Kim, Joonsuk Huh, and Daniel K. Park, arXiv, September 2023 (16 pages).
- [564] [Transfer Learning Based Hybrid Quantum Neural Network Model for Surface Anomaly Detection](#), by Sounak Bhowmik and Himanshu Thapliyal, arXiv, August 2024 (6 pages).
- [565] [Object Detection Using Quantum Transfer Learning](#), by Mohammad Bahrami, Morteza Rafiee, and Mohsen Norouzi, arXiv, September 2026 (17 pages).
- [566] [Quantum Capsule Networks](#), by Zidu Liu, Pei-Xin Shen, Weikang Li, L.-M. Duan, and Dong-Ling Deng, arXiv, December 2022 (7 pages).
- [567] [Non-Markovianity and memory enhancement in Quantum Reservoir Computing](#), by Antonio Sannia, Ricard Ravell Rodríguez, Gian Luca Giorgi, and Roberta Zambrini, arXiv, May 2025 (12 pages).
- [568] [Expressivity Limits of Quantum Reservoir Computing](#), by Nils-Erik Schütte, Niclas Götting, et al., arXiv, May 2025 (11 pages).
- [569] [Quantum Reservoir Computing: Recent Advances and Future Directions](#), by Shehbaz Tariq, Muhammad Talha, Arshid Ali, Muhammad Diyan, and Symeon Chatzinotas, arXiv, July 2026 (22 pages).
- [570] [The Role of Entanglement in Quantum Reservoir Computing with Coupled Kerr Nonlinear Oscillators](#), by Ali Karimi, Hadi Zadeh-Haghighi, Youssef Kora, and Christoph Simon, arXiv, August 2025 (12 pages).
- [571] [Robust quantum reservoir computers for forecasting chaotic dynamics: generalized synchronization and stability](#), by Osama Ahmed, Felix Tennie, and Luca Magri, arXiv, June 2025 (28 pages).
- [572] [Quantum Reservoir Computing and Risk Bounds](#), by Naomi Mona Chmielewski, Nina Amini, and Joseph Mikael, arXiv, January 2025 (33 pages).
- [573] [Quantum Reservoir Computing for Credit Card Default Prediction on a Neutral Atom Platform](#), by Giacomo Vitali, Davide Corbelleto, et al., arXiv, October 2025 (21 pages).
- [574] [Opportunities in Quantum Reservoir Computing and Extreme Learning Machines](#), by Pere Mujal, Rodrigo Martínez-Peña, Johannes Nokkala, Jorge García-Beni, Gian Luca Giorgi, Miguel C. Soriano, and Roberta Zambrini, arXiv, July 2021 (14 pages).
- [575] [Time Series Quantum Reservoir Computing with Weak and Projective Measurements](#), by Pere Mujal, Rodrigo Martínez-Peña, Gian Luca Giorgi, Miguel C. Soriano, and Roberta Zambrini, arXiv, May 2022 (19 pages).
- [576] [Configured Quantum Reservoir Computing for Multi-Task Machine Learning](#), by Wei Xia, Xiaopeng Li, et al., arXiv, March 2023 (15 pages).
- [577] [Digital Quantum Reservoir Computing for ATM Time Series Prediction](#), by Chiara Vercellino, Giacomo Vitali, Valeria Zaffaroni, Francesca Cibrario, Emanuele Dri, Paolo Viviani, Olivier Terzo, and Davide Corbelleto, arXiv, June 2026 (11 pages).
- [578] [Large-scale quantum reservoir learning with an analog quantum computer](#), by Milan Kornjaca, Sheng-Tao Wang, et al., arXiv, July 2024 (24 pages).
- [579] [Large-scale quantum reservoir computing using a Gaussian Boson Sampler](#), by Valeria Cimini, Mandar M. Sohoni, Federico Presutti, Benjamin K. Malia, Shi-Yuan Ma, Ryotatsu Yanagimoto, Tianyu Wang, Tatsuhiro Onodera, Logan G. Wright, and Peter L. McMahon, npj Quantum Information, May 2026 (30 pages).
- [580] [Temporal information processing on a 4,500-qubit quantum annealer](#), by Antonio Sannia, Roberto Menta, Pratik Sathe, Dario De Santis, Vittorio Giovannetti, Luis Pedro García-Pintos, Davide Venturelli, Gian Luca Giorgi, Roberta Zambrini, and Francesco Caravelli, arXiv, September 2026 (21 pages).
- [581] [Breaking concentration barriers for quantum extreme learning on digital quantum processors](#), by Timothee Dao, Ege Yilmaz, Ibrahim Shehzad, Christophe Pere, Kumar Ghosh, Isabelle Wittmann, Thomas Brunswiler, Giorgio Cortiana, Corey O'Meara, Stefan Woerner, and Francesco Tacchino, arXiv, March 2026 (17 pages).
- [582] [When Classical Baselines Are Tuned as Carefully as the Quantum Model, Does Quantum Reservoir Computing Still Win?](#), by Tushar Pandey, arXiv, July 2026 (9 pages).
- [583] [A Quantum Reservoir Architecture for Chaotic Forecasting and a Test of Whether Its High Dimension Helps](#), by Tushar Pandey, arXiv, July 2026 (11 pages).
- [584] [Memory-Nonlinearity Trade-off across Quantum Reservoir Computing Frameworks](#), by Saud Cindrak, Lara Giebler, Niclas Gotting, Christopher Gies, and Kathy Ludge, arXiv, March 2026 (12 pages).
- [585] [A hidden bottleneck in classical and quantum linear reservoir computing](#), by Johannes Nokkala, Federico Centrone, and Francesco Arzani, arXiv, May 2026 (8 pages).
- [586] [Quantum magic and non-commutativity as computational resources in quantum reservoir computing](#), by Wei Xia, Shuaifan Cao, Xingze Qiu, and Xiaopeng Li, arXiv, July 2026 (28 pages).
- [587] [Role of scrambling and noise in temporal information processing with quantum systems](#), by Weijie Xiong, Zoë Holmes, Armando Angrisani, Yudai Suzuki, Thiparat Chotibut, and Supanut Thanasilp, arXiv, May 2025 (35 pages).
- [588] [Experimental memory control in continuous variable optical quantum reservoir computing](#), by Iris Paparelle, Valentina Parigi, et al., arXiv, June 2025 (17 pages).
- [589] [Quantum machine learning via continuous-variable cluster states and teleportation](#), by Jorge García-Beni, Iris Paparelle, Valentina Parigi, Gian Luca Giorgi, Miguel C. Soriano, and Roberta Zambrini, arXiv, November 2024 (15 pages).
- [590] [Time-series forecasting with multiphoton quantum states and integrated photonics](#), by Rosario Di Bartolo, Simone Piacentini, Francesco Ceccarelli, Giacomo Corrielli, Roberto Osellame, Valeria Cimini, and Fabio Sciarino, npj Quantum Information, April 2026 (28 pages).
- [591] [Quantum reservoir computing with classical and nonclassical states in an integrated optical circuit](#), by S. Swierczewski, W. Verstraelen, P. Deuar, T. C. H. Liew, A. Opala, and M. Matuszewski, arXiv, March 2026 (12 pages).
- [592] [Practical Quantum Reservoir Computing in Rydberg Atom Arrays](#), by Dong-Sheng Liu, Qing-Xuan Jie, Chang-Ling Zou, Xi-Feng Ren, and Guang-Can Guo, Physical Review A, April 2026 (9 pages).
- [593] [Quantum Reservoir Computing with Neutral Atoms on a Small, Complex, Medical Dataset](#), by Luke Antoncich, Yuben Moodley, Ugo Varetto, Jingbo Wang, Jonathan Wurtz, Jing Chen, Pascal Jahan Elahi, and Casey R. Myers, arXiv, February 2026 (16 pages).
- [594] [Analog neutral-atom for in-memory processing in quantum reservoir computing](#), by Luca Nigro, Gian Luca Giorgi, Enrico Prati, and Roberta Zambrini, arXiv, September 2026 (17 pages).
- [595] [Active Learning on a Programmable Photonic Quantum Processor](#), by Chen Ding, He-Liang Huang, et al., arXiv, August 2022 (17 pages).
- [596] [Quantum HyperNetworks: Training Binary Neural Networks in Quantum Superposition](#), by Juan Carrasquilla, Giacomo Torlai, et al., arXiv, July 2025 (15 pages).
- [597] [An artificial spiking quantum neuron](#), by Lasse Bjørn Kristensen, Alán Aspuru-Guzik, Nikolaj T. Zinner, et al., npj Quantum Information, April 2021.
- [598] [Quantum Image Processing](#), by Alok Anand, Meizhong Lyu, Prabh Simran Baweja, and Vinay Patil, arXiv, March 2022 (10 pages).

- [599] [Processing Images in Entangled Quantum Systems](#), by S.E. Venegas-Andraca and J.L. Ball, 2010 (11 pages)  (preprint on [arXiv](#)).
- [600] [Towards Bundle Adjustment for Satellite Imaging via Quantum Machine Learning](#), by Nico Piatkowski, Thore Gerlach, Romain Hugues, Rafet Sifa, Christian Bauckhage, and Frederic Barbaresco, arXiv, April 2022 (8 pages).
- [601] [Hyperbolic Equivariant Convolutional Neural Networks for Fish-Eye Image Processing](#), by Pierre-Yves Lagrave et al., February 2022 (9 pages).
- [602] [Exploiting the Quantum Advantage for Satellite Image Processing: Review and Assessment](#), by Soronzonbold Otgonbaatar and Dieter Kranzlmüller, arXiv, November 2023 (9 pages).
- [603] [Qubit-efficient Variational Quantum Algorithms for Image Segmentation](#), by Supreeth Mysore Venkatesh, Antonio Macaluso, Marlon Nuske, Matthias Klusch, and Andreas Dengel, arXiv, May 2024 (7 pages).
- [604] [A hybrid quantum image edge detector for the NISQ era](#), by Alexander Geng, Ali Moghiseh, Claudia Redenbach, and Katja Schladitz, arXiv, March 2022 (19 pages).
- [605] [Edge Detection Quantumized: A Novel Quantum Algorithm For Image Processing](#), by Syed Emad Uddin Shubha, Mir Muzahedul Islam, Tanvir Ahahmed Sadi, Md. Hasibul Hasan Miraz, and M.R.C. Mahdy, arXiv, April 2024 (10 pages).
- [606] [Quantum machine learning for image classification](#), by Arsenii Senokosov, Alexandr Sedykh, Asel Sagingalieva, Basil Kyriacou, and Alexey Melnikov, arXiv, March 2024 (13 pages).
- [607] [A practical overview of image classification with variational tensor-network quantum circuits](#), by Diego Guala, Shaoming Zhang, Esther Cruz, Carlos A. Riofrío, Johannes Klepsch, and Juan Miguel Arrazola, arXiv, September 2022 (11 pages).
- [608] [Quantum Methods for Neural Networks and Application to Medical Image Classification](#), by Jonas Landman, Natansh Mathur, Yun Yvonna Li, Martin Strahm, Skander Kazdaghi, Anupam Prakash, and Iordanis Kerenidis, arXiv, December 2022 (30 pages).
- [609] [Quantum Vision Transformers](#), by El Amine Cherrat, Iordanis Kerenidis, Natansh Mathur, Jonas Landman, Martin Strahm, and Yun Yvonna Li, arXiv, February 2024 (20 pages).
- [610] [Hybrid quantum image classification and federated learning for hepatic steatosis diagnosis](#), by Luca Lusnig, Asel Sagingalieva, Mikhail Surmach, Tatjana Protasevich, Ovidiu Michiu, Joseph McLoughlin, Christopher Mansell, Graziano de'Petris, Deborah Bonazza, Fabrizio Zanconati, Alexey Melnikov, and Fabio Cavalli, arXiv, March 2024 (13 pages).
- [611] [SLIQ: Quantum Image Similarity Networks on Noisy Quantum Computers](#), by Daniel Silver, Tirthak Patel, Aditya Ranjan, Harshitta Gandhi, William Cutler, and Devesh Tiwari, arXiv, September 2023 (9 pages).
- [612] [Image Denoising with Machine Learning: A Novel Approach to Improve Quantum Image Processing Quality and Reliability](#), by Yifan Zhou and Yan Shing Liang, arXiv, September 2024 (9 pages).
- [613] [Quantum Kernel for Image Classification of Real World Manufacturing Defects](#), by Daniel Beaulieu, Dylan Miracle, Anh Pham, and William Scherr, arXiv, December 2022 (8 pages).
- [614] [Quantum Image Loading: Hierarchical Learning and Block-Amplitude Encoding](#), by Hrant Gharibyan, Hovnatan Karapetyan, Tigran Sedrakyan, Pero Subasic, Vincent P. Su, Rudy H. Tanin, and Hayk Tepanyan, arXiv, April 2025 (18 pages).
- [615] [Scaling Quantum Machine Learning without Tricks: Full-Resolution and Diverse Image Generation](#), by Jonas Jäger, Florian J. Kiwit, and Carlos A. Riofrío, arXiv, July 2026 (29 pages).
- [616] [An Amplitude-Encoding-Based Classical-Quantum Transfer Learning framework: Outperforming Classical Methods in Image Recognition](#), by Shouwei Hu, Xi Li, Banyao Ruan, and Zhihao Liu, arXiv, February 2025 (35 pages).
- [617] [Quantum medical image encoding and compression using Fourier-based methods](#), by Taehee Ko, Inho Lee, and Hyeon Won Yu, arXiv, May 2025 (12 pages).
- [618] [Quantum Image Classification: Experiments on Utility-Scale Quantum Computers](#), by Hrant Gharibyan, Hovnatan Karapetyan, Tigran Sedrakyan, Pero Subasic, Vincent P. Su, Rudy H. Tanin, and Hayk Tepanyan, arXiv, April 2025 (18 pages).
- [619] [Quantum Neural Networks in Practice: A Comparative Study with Classical Models from Standard Data Sets to Industrial Images](#), by Daniel Basilewitsch, João F. Bravo, Christian Tutschku, and Frederick Struckmeier, arXiv, May 2025 (24 pages).
- [620] [Efficient Quantum Convolutional Neural Networks for Image Classification: Overcoming Hardware Constraints](#), by Peter Röseler, Oliver Schaudt, Helmut Berg, Christian Bauckhage, and Matthias Koch, arXiv, May 2025 (20 pages).
- [621] [Inverse Quantum Fourier Transform Inspired Algorithm for Unsupervised Image Segmentation](#), by Taoreed Akinola, Lijun Qian, et al., arXiv, January 2023 (8 pages).
- [622] [A Quantum Moving Target Segmentation Algorithm for Grayscale Video](#), by Wenjie Liu et al., October 2023 (15 pages)  (preprint on [arXiv](#)).
- [623] [Efficient MPS representations and quantum circuits from the Fourier modes of classical image data](#), by Bernhard Jobst, Kevin Shen, Carlos A. Riofrío, Elvira Shishenina, and Frank Pollmann, arXiv, November 2024 (17 pages).
- [624] [Quantum Annealing for Computer Vision Minimization Problems](#), by Shahrokh Heidari, Michael J. Dinneen, and Patrice Delmas, arXiv, December 2023 (23 pages).
- [625] [Quantum Vision Clustering](#), by Xuan Bac Nguyen, Hugh Churchill, Khoa Luu, and Samee U. Khan, arXiv, February 2025 (16 pages).
- [626] [HQFNN: A Compact Quantum-Fuzzy Neural Network for Accurate Image Classification](#), by Jianhong Yao and Yangming Guo, arXiv, June 2025 (14 pages).
- [627] [Near-term advances in quantum natural language processing](#), by Dominic Widdows, Aaranya Alexander, Daiwei Zhu, Chase Zimmerman, and Arunava Majumder, Annals of Mathematics and Artificial Intelligence, April 2024 (27 pages)  (preprint on [arXiv](#)).
- [628] [Quantum Natural Language Processing](#), by Dominic Widdows, Willie Aboumrar, Dohun Kim, Sayonee Ray, and Jonathan Mei, arXiv, April 2024 (19 pages).
- [629] [QuCoWE Quantum Contrastive Word Embeddings with Variational Circuits for NearTerm Quantum Devices](#), by Rabimba Karanjai, Hemanth Hegadehalli Madhavarao, Lei Xu, and Weidong Shi, arXiv, November 2025 (15 pages).
- [630] [Quantum Text Encoding for Classification Tasks](#), by Aaranya Alexander and Dominic Widdows, arXiv, January 2023 (7 pages).
- [631] [Quantum Text Classifier – A Synchronistic Approach Towards Classical and Quantum Machine Learning](#), by Prabhath Santi, Kamakhya Mishra, and Sibabrata Mohanty, arXiv, May 2023 (7 pages).
- [632] [Fast Quantum Algorithm for Attention Computation](#), by Yeqi Gao, Zhao Song, Xin Yang, and Ruizhe Zhang, arXiv, July 2023 (21 pages).
- [633] [Recent Advances for Quantum Neural Networks in Generative Learning](#), by Jinkai Tian, Dacheng Tao, et al., arXiv, June 2022 (30 pages).
- [634] [Hierarchical Learning for Quantum ML: Novel Training Technique for Large-Scale Variational Quantum Circuits](#), by Hrant Gharibyan, Vincent Su, and Hayk Tepanyan, arXiv, November 2023 (22 pages).

- [635] [Universality of Classically Trainable, Quantum-Deployed Boson-Sampling Generative Models](#), by Andrii Kurkin, Ulysse Chabaud, Zoltán Kolarovszki, Bence Bakó, Zoltán Zimborás, and Vedran Dunjko, arXiv, March 2026 (20 pages).
- [636] [Quantum-Enhanced Synthetic Data Generation Using Quantum Circuit Born Machines for Imbalanced Tabular Learning](#), by Tanapol Nuatho, Narisorn Sangnakara, Prapong Prechaprapranwong, and Rajchawit Sarochawikasit, arXiv, July 2026.
- [637] [Tailor Made Embeddings for Quantum Machine Learning](#), by Aldo Lamarre and Dominik Šafránek, arXiv, June 2026 (17 pages).
- [638] [Synthetic weather radar using hybrid quantum-classical machine learning](#), by Graham R. Enos, Chad Rigetti, et al., arXiv, November 2021 (8 pages).
- [639] [A Hybrid Quantum-Classical Generative Adversarial Network for Near-Term Quantum Processors](#), by Albha O'Dwyer Boyle and Reza Nikandish, arXiv, July 2023 (13 pages).
- [640] [MosaiQ: Quantum Generative Adversarial Networks for Image Generation on NISQ Computers](#), by Daniel Silver, Devesh Tiwari, et al., arXiv, August 2023 (10 pages).
- [641] [Quantum Generative Adversarial Networks: Bridging Classical and Quantum Realms](#), by Sahil Nokhwal, Suman Nokhwal, Saurabh Pahune, and Ankit Chaudhary, arXiv, December 2023 (8 pages).
- [642] [A Survey of Quantum Generative Adversarial Networks: Architectures, Use Cases, and Real-World Implementations](#), by Mujahidul Islam, Serkan Turkeli, and Fatih Ozyaydin, arXiv, June 2025 (30 pages).
- [643] [Exponential capacity scaling of classical GANs compared to hybrid latent style-based quantum GANs](#), by Milan Liepelt and Julien Baglio, arXiv, January 2026 (34 pages).
- [644] [Towards efficient quantum algorithms for diffusion probability models](#), by Yunfei Wang, Ruoxi Jiang, Yingda Fan, Xiaowei Jia, Jens Eisert, Junyu Liu, and Jin-Peng Liu, arXiv, February 2025 (20 pages).
- [645] [Generative Invertible Quantum Neural Networks](#), by Armand Rousselot and Michael Spannowsky, arXiv, February 2024 (18 pages).
- [646] [End-to-End Demonstration of Quantum Generative Adversarial Networks for Steel Microstructure Image Augmentation on a Trapped-Ion Quantum Computer](#), by Samwel Sekwao, Kyujin Shin, et al., arXiv, April 2025 (15 pages).
- [647] [TabularQGAN: A Quantum Generative Model for Tabular Data](#), by Pallavi Bhardwaj, Caitlin Jones, Lasse Dierich, and Aleksandar Vučković, arXiv, May 2025 (18 pages).
- [648] [Quantum Flow Matching](#), by Zidong Cui, Pan Zhang, and Ying Tang, arXiv, August 2025 (15 pages).
- [649] [Quantum versus Classical Generative Modelling in Finance](#), by Brian Coyle, Maxwell Henderson, Justin Chan Jin Le, Niraj Kumar, Marco Paini, and Elham Kashefi, arXiv, August 2020 (17 pages).
- [650] [The Born Supremacy: Quantum Advantage and Training of an Ising Born Machine](#), by Brian Coyle, Daniel Mills, Vincent Danos, and Elham Kashefi, arXiv, April 2021 (18 pages).
- [651] [The Born Ultimatum: Conditions for Classical Surrogation of Quantum Generative Models with Correlators](#), by Mario Herrero-Gonzalez, Brian Coyle, Kieran McDowall, Ross Grassie, Sjoerd Beentjes, Ava Khamseh, and Elham Kashefi, arXiv, November 2025 (45 pages).
- [652] [Conditional Generative Models for Learning Stochastic Processes](#), by Salvatore Certo, Anh Pham, Nicolas Robles, and Andrew Vlasic, arXiv, August 2023 (11 pages).
- [653] [A Framework for Demonstrating Practical Quantum Advantage: Racing Quantum against Classical Generative Models](#), by Mohamed Hibat-Allah, Marta Mauri, Juan Carlosquilla, and Alejandro Perdomo-Ortiz, arXiv, March 2023 (17 pages).
- [654] [Guardians of the Quantum GAN](#), by Archisman Ghosh, Debarshi Kundu, Avimita Chatterjee, and Swaroop Ghosh, arXiv, April 2024 (11 pages).
- [655] [A natural NISQ model of quantum self-attention mechanism](#), by Shangshang Shi, Zhimin Wang, Jiaxin Li, Yanan Li, Ruimin Shang, Haiyong Zheng, Guoqiang Zhong, and Yongjian Gu, arXiv, May 2023 (13 pages).
- [656] [Quantum Self-Attention Neural Networks for Text Classification](#), by Guangxi Li, Xuanqiang Zhao, and Xin Wang, arXiv, September 2023 (13 pages).
- [657] [Quantum linear algebra is all you need for Transformer architectures](#), by Naixu Guo, Alán Aspuru-Guzik, Patrick Rebentrost, et al., arXiv, May 2024 (31 pages).
- [658] [GPT on a Quantum Computer](#), by Yidong Liao and Chris Ferrie, arXiv, March 2024 (35 pages).
- [659] [A Survey of Quantum Transformers: Architectures, Challenges and Outlooks](#), by Hui Zhang, Qinglin Zhao, Mengchu Zhou, Li Feng, Dusit Niyato, Shenggen Zheng, and Lin Chen, arXiv, June 2025 (25 pages).
- [660] [Quantum Knowledge Distillation for Large Language Models](#), by Lingxiao Li, Fei Gao, et al., arXiv, May 2025 (20 pages).
- [661] [Quixer: A Quantum Transformer Model](#), by Nikhil Khatri, Gabriel Matos, Luuk Coopmans, and Stephen Clark, arXiv, June 2024 (17 pages).
- [662] [Quantum Large Language Models via Tensor Network Disentangled](#), by Borja Aizpurua, Saeed S. Jahromi, Sukhbinder Singh, and Roman Orus, arXiv, October 2024 (4 pages).
- [663] [Quantum-Enhanced LLM Efficient Fine Tuning](#), by Xiaofei Kong, Guoping Guo, et al., arXiv, April 2025 (7 pages).
- [664] [A Survey of Quantum Transformers: Technical Approaches, Challenges and Outlooks](#), by Hui Zhang and Qinglin Zhao, arXiv, 2025 (25 pages).
- [665] [Quantum machine learning models for graphs](#), by Frédéric Sauvage, Pranav Kalidindi, Frederic Rapp, and Martin Larocca, arXiv, July 2026.
- [666] [Representation Theory for Geometric Quantum Machine Learning](#), by Michael Ragone, Marco Cerezo, et al., arXiv, February 2023 (43 pages).
- [667] [Theoretical Guarantees for Permutation-Equivariant Quantum Neural Networks](#), by Louis Schatzki, Marco Cerezo, et al., arXiv, February 2024 (21 pages).
- [668] [Group-Invariant Quantum Machine Learning](#), by Martin Larocca, Frederic Sauvage, Faris M. Sbahi, Guillaume Verdon, Patrick J. Coles, and M. Cerezo, arXiv, September 2022 (10 pages).
- [669] [Towards Quantum Federated Learning](#), by Chao Ren, Rudai Yan, Huihui Zhu, Han Yu, Minrui Xu, Yuan Shen, Yan Xu, Ming Xiao, Zhao Yang Dong, Mikael Skoglund, Dusit Niyato, and Leong Chuan Kwek, arXiv, August 2024 (21 pages).
- [670] [Federated Quantum Machine Learning](#), by Samuel Yen-Chi Chen and Shinjae Yoo, arXiv, March 2021 (25 pages).
- [671] [Federated Quantum Natural Gradient Descent for Quantum Federated Learning](#), by Jun Qi, arXiv, August 2022 (9 pages).
- [672] [Quantum Federated Learning for Distributed Quantum Networks](#), by Kai Yu, Fei Gao, and Song Lin, arXiv, April 2024 (12 pages).
- [673] [Quantum Federated Learning: Analysis, Design and Implementation Challenges](#), by Dev Gurung, Shiva Raj Pokhrel, and Gang Li, arXiv, June 2023 (7 pages).
- [674] [FL-QDSNNs: Federated Learning with Quantum Dynamic Spiking Neural Networks](#), by Nouhaila Innan, Alberto Marchisio, and Muhammad Shafique, arXiv, December 2024 (9 pages).

- [675] [Practical quantum federated learning and its experimental demonstration](#), by Zhi-Ping Liu, Xiao-Yu Cao, Hao-Wen Liu, Xiao-Ran Sun, Yu Bao, Yu-Shuo Lu, Hua-Lei Yin, and Zeng-Bing Chen, arXiv, January 2025 (21 pages).
- [676] [Towards Heterogeneous Quantum Federated Learning: Challenges and Solutions](#), by Ratun Rahman, Dinh C. Nguyen, Christo Kurisummoottil Thomas, and Walid Saad, arXiv, December 2025.
- [677] [Federated Quantum Kernel Learning for Anomaly Detection in Multivariate IoT Time-Series](#), by Kuan-Cheng Chen, Samuel Yen-Chi Chen, Chen-Yu Liu, and Kin K. Leung, arXiv, November 2025 (8 pages).
- [678] [Federated Learning with Quantum Secure Aggregation](#), by Yichi Zhang, Chao Zhang, Cai Zhang, Lixin Fan, Bei Zeng, and Qiang Yang, arXiv, September 2023 (31 pages).
- [679] [MQFL-FHE: Multimodal Quantum Federated Learning Framework with Fully Homomorphic Encryption](#), by Siddhant Dutta, Nouhaila Innan, Sadok Ben Yahia, Muhammad Shafique, and David Esteban Bernal Neira, arXiv, April 2025 (10 pages).
- [680] [Efficient Quantum Fully Homomorphic Encryption](#), by Fengxia Liu, Zixian Gong, Kun Tian, Yi Zhang, Zhiming Zheng, and Maozhi Xu, arXiv, June 2026 (32 pages).
- [681] [Quantum Homomorphic Encryption: Towards Practical and Private Computation on Untrusted Quantum Hardware](#), by Jon Hernández-Bueno, Oscar Lage, Marivi Higuero, and Jasone Astorga, arXiv, April 2026 (14 pages).
- [682] [Quantum Federated Learning Experiments in the Cloud with Data Encoding](#), by Shiva Raj Pokhrel, Lei Pan, et al., arXiv, May 2024 (7 pages).
- [683] [A Quantum Federated Learning Framework for Classical Clients](#), by Yanqi Song, Yusen Wu, Shengyao Wu, Dandan Li, Qiaoyan Wen, Sujuan Qin, and Fei Gao, arXiv, December 2023 (9 pages).
- [684] [Physics-Informed Classical and Quantum Neural Networks for One-Dimensional Schrodinger Eigenvalue Problems](#), by Tariq Mahmood, Waqas Arshad, Bilal Naseer, and Alfredo Raya, arXiv, August 2026 (20 pages).
- [685] [PINNACLE: An Open-Source Computational Framework for Classical and Quantum PINNs](#), by Shimon Pissnoy, Hemant Chandravamsi, Ziv Chen, Aaron Goldgewert, Gal Shaviner, Boris Shragner, and Steven H. Frankel, arXiv, April 2026 (48 pages).
- [686] [Quantum automated theorem proving](#), by Zheng-Zhi Sun, Qi Ye, and Dong-Ling Deng, arXiv, January 2026 (28 pages).
- [687] [Quantum Agents](#), by Eldar Sultanow, Madjid Tehrani, Siddhant Dutta, William J Buchanan, and Muhammad Shahbaz Khan, arXiv, June 2025 (45 pages).
- [688] [Hamiltonian Formalism for Comparing Quantum and Classical Intelligence](#), by Elija Perrier, arXiv, June 2025 (10 pages).
- [689] [Experimentally Realizable Continuous-variable Quantum Neural Networks](#), by Shikha Bangar, Leanto Sunny, Kubra Yeter-Aydeniz, and George Siopsis, arXiv, June 2023 (13 pages).
- [690] [Quantum Machine Learning](#), by Jacob Biamonte, Peter Wittek, Nicola Pancotti, Patrick Rebentrost, Nathan Wiebe, and Seth Lloyd, arXiv, May 2018 (24 pages).
- [691] [D-Wave Quantum Computing - Access & application via cloud deployment](#), by Colin Williams, 2017 (43 slides).
- [692] [Benchmarking Quantum Hardware for Training of Fully Visible Boltzmann Machines](#), by Dmytro Korenkevych, Yanbo Xue, Zhengbing Bian, Fabian Chudak, William G. Macready, Jason Rolfe, and Evgeny Andriyash, arXiv, November 2016 (22 pages).
- [693] [Analog quantum variational embedding classifier](#), by Rui Yang, Samuel Bosch, Bobak Kiani, Seth Lloyd, and Adrian Lupascu, arXiv, May 2023 (10 pages).
- [694] [Beyond Optimization: Harnessing Quantum Annealer Dynamics for Machine Learning](#), by Akitada Sakurai, Aoi Hayashi, Tadayoshi Matsumori, Daisuke Kaji, Tadashi Kadowaki, and Kae Nemoto, arXiv, February 2026.
- [695] [The prospects of quantum computing in computational molecular biology](#), by Carlos Outeiral, April 2020 (23 pages).
- [696] [Challenges and Opportunities in Quantum Machine Learning](#), by M. Cerezo, Guillaume Verdon, Hsin-Yuan Huang, Lukasz Cincio, and Patrick J. Coles, arXiv, March 2023 (14 pages).
- [697] [Next-Generation Quantum Neural Networks: Enhancing Efficiency, Security, and Privacy](#), by Nouhaila Innan, Muhammad Kashif, Alberto Marchisio, Mohamed Bennai, and Muhammad Shafique, arXiv, July 2025 (4 pages).
- [698] [A Survey of Methods for Mitigating Barren Plateaus for Parameterized Quantum Circuits](#), by Michelle Gelman, arXiv, June 2024 (18 pages).
- [699] [Improved simulation of stabilizer circuits](#), by Scott Aaronson and Daniel Gottesman, Physical Review A, November 2004 (15 pages).
- [700] [Simulation of quantum circuits by low-rank stabilizer decompositions](#), by Sergey Bravyi, Dan Browne, Padraic Calpin, Earl Campbell, David Gosset, and Mark Howard, Quantum, September 2019 (48 pages).
- [701] [Efficient classical simulation of slightly entangled quantum computations](#), by Guifre Vidal, Physical Review Letters, October 2003 (39 pages).
- [702] [Prospects for quantum advantage in machine learning from the representability of functions](#), by Sergi Masot-Llima, Elies Gil-Fuster, Carlos Bravo-Prieto, Jens Eisert, and Tommaso Guaita, arXiv, December 2025 (21 pages).
- [703] [Lie-algebraic classical simulations for variational quantum computing](#), by Matthew L. Goh, Martin Larocca, Lukasz Cincio, M. Cerezo, and Frederic Sauvage, arXiv, August 2023.
- [704] [Full classification of Pauli Lie algebras](#), by Gerard Aguilar, Simon Cichy, Jens Eisert, and Lennart Bittel, arXiv, August 2024 (20 pages).
- [705] [Classical simulation of noninteracting-fermion quantum circuits](#), by Barbara M. Terhal and David P. DiVincenzo, Physical Review A, February 2002 (13 pages).
- [706] [A polynomial-time classical algorithm for noisy quantum circuits](#), by Thomas Schuster, Chao Yin, Xun Gao, and Norman Y. Yao, arXiv, July 2024.
- [707] [Classical surrogate simulation of quantum systems with LOWESA](#), by Manuel S. Rudolph, Enrico Fontana, Zoe Holmes, and Lukasz Cincio, arXiv, August 2023.
- [708] [Classically estimating observables of noiseless quantum circuits](#), by Armando Angrisani, Alexander Schmidhuber, Manuel S. Rudolph, M. Cerezo, Zoe Holmes, and Hsin-Yuan Huang, arXiv, September 2024 (38 pages).
- [709] [Shadows of quantum machine learning](#), by Sofiene Jerbi, Vedran Dunjko, et al., Nature Communications, July 2024 (7 pages).
- [710] [Effect of data encoding on the expressive power of variational quantum-machine-learning models](#), by Maria Schuld, Ryan Sweke, and Johannes Jakob Meyer, Physical Review A, March 2021 (16 pages).
- [711] [Classical surrogates for quantum learning models](#), by Franz J. Schreiber, Jens Eisert, and Johannes Jakob Meyer, Physical Review Letters, September 2023.
- [712] [Potential and limitations of random Fourier features for dequantizing quantum machine learning](#), by Ryan Sweke, Jens Eisert, Johannes Jakob Meyer, et al., Quantum, February 2025 (44 pages).
- [713] [Classically approximating variational quantum machine learning with random Fourier features](#), by Jonas Landman, Slimane Thabet, Constantin Dalyac, Hela Mhiri, and Elham Kashefi, arXiv, October 2022 (17 pages).

- [714] [Barren plateaus in quantum neural network training landscapes](#), by Jarrod R. McClean, Sergio Boixo, Vadim N. Smelyanskiy, Ryan Babbush, and Hartmut Neven, *Nature Communications*, November 2018 (6 pages).
- [715] [On the relation between trainability and dequantization of variational quantum learning models](#), by Elies Gil-Fuster, Casper Gyurik, Adrian Perez-Salinas, and Vedran Dunjko, *arXiv*, June 2024 (25 pages).
- [716] [QLook:Quantum-Driven Viewport Prediction for Virtual Reality](#), by Niusha Sabri Kadijani, Yoga Suhas Kuruba Manjunath, Xiaodan Bi, and Lian Zhao, *arXiv*, September 2025 (5 pages).
- [717] [Maritime object classification with SAR imagery using quantum kernel methods](#), by John Tanner, Jingbo Wang, et al., *arXiv*, December 2025 (5 pages).
- [718] [Non-Unitary Quantum Machine Learning: Fisher Efficiency Transitions from Distributed Quantum Expressivity](#), by Apoorv Kumar Masta, Srinjoy Ganguly, Shalini Deven-drababu, Farina Riaz, Rajib Rana, and Björn Schuller, *arXiv*, March 2026.
- [719] [Robust evaluation of classical and quantum machine learning under noise, imbalance, feature reduction and explainability](#), by Savita Kumari Sheoran, Vikesh Yadav, and Rakesh Kumar Sheoran, *Scientific Reports*, December 2025.
- [720] [Quantum Machine Learning for UAV Swarm Intrusion Detection](#), by Kuan-Cheng Chen, Samuel Yen-Chi Chen, Tai-Yue Li, Chen-Yu Liu, and Kin K. Leung, *arXiv*, September 2025 (6 pages).
- [721] [Toward Near-Real-Time Marine Oil Spill Detection in SAR Imagery using Quantum-Assisted SVM](#), by Joseph Strauss and Jyotsna Sharma, *arXiv*, May 2026 (5 pages).
- [722] [HQCNN: A Hybrid Quantum-Classical Neural Network for Medical Image Classification](#), by Shahjalal, Jahid Karim Fahim, Pintu Chandra Paul, Md Robin Hossain, Md. Tofael Ahmed, and Dulal Chakraborty, *arXiv*, September 2025 (21 pages).
- [723] [Detecting Phishing in Ethereum Networks using Quantum Machine Learning](#), by Sai Sakunthala Guddanti, Anupama Ray, Mrunal Arun Kumavat, and Anil Prabhakar, *arXiv*, July 2026 (12 pages).
- [724] [The Input Problem: A Permanent Bottleneck for Quantum Machine Learning](#), by Muhammad Faryad, *arXiv*, August 2026 (5 pages).
- [725] [Evaluating Angle and Amplitude Encoding Strategies for Variational Quantum Machine Learning: their impact on model's accuracy](#), by Antonio Tudisco, Giovanna Turvani, et al., *arXiv*, August 2025 (23 pages).
- [726] [Benchmarking data encoding methods in Quantum Machine Learning](#), by Orlane Zang, Grégoire Barrué, and Tony Quertier, *arXiv*, May 2025 (30 pages).
- [727] [Typical Machine Learning Datasets as Low-Depth Quantum Circuits](#), by Florian J. Kiwit, Carlos A. Riofrio, et al., *arXiv*, May 2025 (33 pages).
- [728] [Quantum embeddings for machine learning](#), by Seth Lloyd, Maria Schuld, Aroosa Ijaz, Josh Izaac, and Nathan Killoran, *arXiv*, February 2020 (11 pages).
- [729] [Nearest Centroid Classification on a Trapped Ion Quantum Computer](#), by Sonika Johri, Shantanu Debnath, Avinash Mocherla, Alexandros Singh, Anupam Prakash, Jungsang Kim, and Iordanis Kerenidis, *arXiv*, December 2020 (15 pages).
- [730] [Classification of the Fashion-MNIST Dataset on a Quantum Computer](#), by Kevin Shen, Bernhard Jobst, Elvira Shishenina, and Frank Pollmann, *arXiv*, March 2024 (15 pages).
- [731] [Quantum Algorithms for Unsupervised Machine Learning and Neural Networks](#), by Jonas Landman, *arXiv*, November 2021 (192 pages).
- [732] [Practicality of training a quantum-classical machine in the NISQ era](#), by Tarun Dutta, Alex Jin, Clarence Liu Huihong, J I Latorre, and Manas Mukherjee, *arXiv*, February 2025 (30 pages).
- [733] [Exponential quantum advantage in processing massive classical data](#), by Haimeng Zhao, Alexander Zlokapa, Hartmut Neven, Ryan Babbush, John Preskill, Jarrod R. McClean, and Hsin-Yuan Huang, *arXiv*, April 2026 (144 pages).
- [734] [A Novel Quantum Image Compression Method Based on JPEG](#), by Jian Wang, Nan Jiang, and Na Zhao, *arXiv*, January 2017 (30 pages).
- [735] [Optimizing Quantum Embedding using Genetic Algorithm for QML Applications](#), by Koustubh Phalak, Archisman Ghosh, and Swaroop Ghosh, *arXiv*, November 2024 (9 pages).
- [736] [Drastic Circuit Depth Reductions with Preserved Adversarial Robustness by Approximate Encoding for Quantum Machine Learning | Intelligent Computing](#), by Maxwell T. West et al., *Intelligent Computing*, June 2025 (11 pages).
- [737] [Quantum State Preparation via Neural Network Encoding in Quantum Machine Learning](#), by Kevin W. Aoun, Florian J. Kiwit, Carlos A. Riofrio, Samer Saab Jr., Charbel Al Bateh, Joe Tekli, and Andre Luckow, *arXiv*, May 2026 (8 pages).
- [738] [Predictive Performance of Deep Quantum Data Re-uploading Models](#), by Xin Wang, Han-Xiao Tao, and Re-Bing Wu, *arXiv*, May 2025 (34 pages).
- [739] [Generalization in quantum machine learning from few training data](#), by Matthias C. Caro, Hsin-Yuan Huang, M. Cerezo, Kunal Sharma, Andrew Sornborger, Lukasz Cincio, and Patrick J. Coles, *Nature Communications*, August 2022.
- [740] [Quantum algorithms for group convolution, cross-correlation, and equivariant transformations](#), by Grecia Castelazo, Dirk Englund, Seth Lloyd, Bobak T. Kiani, et al., *arXiv*, September 2022 (27 pages).
- [741] [Quantum advantage in learning from experiments](#), by Hsin-Yuan Huang, Michael Broughton, Jordan Cotler, Sitan Chen, Jerry Li, Masoud Mohseni, Hartmut Neven, Ryan Babbush, Richard Kueng, John Preskill, and Jarrod R. McClean, *arXiv*, December 2021 (6 pages).
- [742] [MNISQ: A Large-Scale Quantum Circuit Dataset for Machine Learning on/for Quantum Computers in the NISQ era](#), by Leonardo Placidi, Keisuke Fujii, et al., *arXiv*, June 2023 (23 pages).
- [743] [Towards structure-preserving quantum encodings](#), by Arthur J. Parzygnat, Tai-Danae Bradley, Andrew Vlasic, and Anh Pham, *arXiv*, December 2024 (17 pages).
- [744] [Quantum Neural Network Compression](#), by Zhirui Hu, Peiyan Dong, Zhepeng Wang, Youzuo Lin, Yanzhi Wang, and Weiwen Jiang, *arXiv*, July 2022 (11 pages).
- [745] [Experimental Quantum End-to-End Learning on a Superconducting Processor](#), by Xiaoxuan Pan, Luyan Sun, et al., *arXiv*, March 2022 (6 pages).
- [746] [Distilling the knowledge with quantum neural networks](#), by Yuxuan Yan, Sitian Qian, Qi Zhao, and Xingjian Zhang, *arXiv*, March 2026 (8 pages).
- [747] [Practical Considerations for Processing Massive Classical Data via Quantum Oracle Sketching](#), by Stefano Markidis, *arXiv*, September 2026 (15 pages).
- [748] [Quantum Neuron: an elementary building block for machine learning on quantum computers](#), by Yudong Cao, Gian Giacomo Guerreschi, and Alán Aspuru-Guzik, *arXiv*, November 2017 (27 pages).
- [749] [Efficient Quantum Circuits for Machine Learning Activation Functions including Constant T-depth ReLU](#), by Wei Zi, Patrick Rebentrost, et al., *arXiv*, April 2024 (13 pages).
- [750] [Quantum Algorithms for Deep Convolutional Neural Network](#), by Iordanis Kerenidis, 2020 (36 pages).
- [751] [Deep Convolutional Neural Networks for Quantum Computers](#), by Jonas Landman, 2020.

- [752] [A Novel Spatial-Temporal Variational Quantum Circuit to Enable Deep Learning on NISQ Devices](#), by Jinyang Li, Weiwen Jiang, et al., arXiv, July 2023 (11 pages).
- [753] [Realization of a quantum neural network using repeat-until-success circuits in a superconducting quantum processor](#), by M. S. Moreira, Anna Y. Matsuura, Leo DiCarlo, et al., arXiv, December 2022 (24 pages).
- [754] [Perspectives on Utilization of Measurements in Quantum Algorithms](#), by Valter Uotila, Ilmo Salmenperä, Leo Becker, Arianne Meijer - van de Griend, Aakash Ravindra Shinde, and Jukka K. Nurminen, arXiv, July 2025 (12 pages).
- [755] [Realistic quantum photonic neural networks](#), by Jacob Ewaniuk, Jacques Carolan, Bhavin J. Shastri, and Nir Rotenberg, arXiv, August 2022 (20 pages).
- [756] [Quantum activation functions for quantum neural networks](#), by Marco Maronese, Claudio Destri, and Enrico Prati, arXiv, January 2022 (28 pages).
- [757] [Non-Linear Transformations of Quantum Amplitudes: Exponential Improvement, Generalization, and Applications](#), by Arthur G. Rattew and Patrick Rebentrost, arXiv, September 2023 (25 pages).
- [758] [Accelerating Inference for Multilayer Neural Networks with Quantum Computers](#), by Arthur G. Rattew, Po-Wei Huang, Naixu Guo, Lirandé Pira, and Patrick Rebentrost, arXiv, October 2025 (44 pages).
- [759] [Shadows of quantum machine learning](#), by Sofiene Jerbi, Casper Gyurik, Simon C. Marshall, Riccardo Molteni, and Vedran Dunjko, arXiv, July 2024 (16 pages).
- [760] [Classical surrogates for quantum learning models](#), by Franz J. Schreiber, Jens Eisert, and Johannes Jakob Meyer, arXiv, June 2022 (4 pages).
- [761] [A quantum-inspired classical algorithm for recommendation systems](#), by Ewin Tang, arXiv, May 2019 (32 pages).
- [762] [Equivalence between exponential concentration in quantum machine learning kernels and barren plateaus in variational algorithms](#), by Pranav Kairon, Jonas Jäger, and Roman V. Krems, arXiv, January 2025 (18 pages).
- [763] [Quark: A Gradient-Free Quantum Learning Framework for Classification Tasks](#), by Zhihao Zhang, Zhuoming Chen, Heyang Huang, and Zhihao Jia, arXiv, October 2022 (19 pages).
- [764] [Scalable Quantum Machine Learning: Trainability, Expressivity and Efficiency](#), by Iordanis Kerenidis, arXiv, July 2026 (38 pages).
- [765] [Comment on “Scalable Quantum Machine Learning: Trainability, Expressivity and Efficiency”: Polynomial Evaluation of the Triplet-Block Readout](#), by Erfan Amidi, arXiv, August 2026 (3 pages).
- [766] [Spectral Bias in Variational Quantum Machine Learning](#), by Callum Duffy and Marcin Jastrzebski, arXiv, June 2025 (12 pages).
- [767] [Rethinking Quantum Noise in Quantum Machine Learning: When Noise Improves Learning](#), by Linghua Zhu, Yulong Dong, Ziyu Zhang, and Xiaosong Li, arXiv, January 2026 (7 pages).
- [768] [Entanglement for Pattern Learning in Temporal Data with Logarithmic Complexity: Benchmarking on IBM Quantum Hardware](#), by Mostafizur Rahaman Laskar and Richa Goel, arXiv, May 2025 (11 pages).
- [769] [A Comprehensive Analysis of Accuracy and Robustness in Quantum Neural Networks](#), by Ban Q. Tran, Duong M. Chu, Hai T.D. Pham, Viet Q. Nguyen, Quan A. Pham, and Susan Mengel, arXiv, April 2026 (25 pages).
- [770] [The Role of Quantum in Hybrid Quantum-Classical Neural Networks: A Realistic Assessment](#), by Dominik Freinberger and Philipp Moser, arXiv, January 2026 (16 pages).
- [771] [Almost fault-tolerant quantum machine learning with drastic overhead reduction](#), by Haiyue Kang, Younghun Kim, Eromanga Adermann, Martin Sevier, and Muhammad Usman, arXiv, July 2025 (18 pages).
- [772] [An Invitation to Distributed Quantum Neural Networks](#), by Lirandé Pira and Chris Ferrie, arXiv, November 2022 (34 pages).
- [773] [Distributed and Secure Kernel-Based Quantum Machine Learning](#), by Arjhun Swaminathan and Mete Akgün, arXiv, April 2025 (23 pages).
- [774] [A rigorous and robust quantum speed-up in supervised machine learning](#), by Yunchao Liu, Srinivasan Arunachalam, and Kristan Temme, arXiv, November 2020 (27 pages).
- [775] [An Exponential Separation Between Quantum and Quantum-Inspired Classical Algorithms for Linear Systems](#), by Allan Grønlund and Kasper Green Larsen, arXiv, April 2025 (23 pages).
- [776] [Quantum Machine Learning for Industrial Applications](#), by Léo Monbroussou, arXiv, June 2026 (203 pages).
- [777] [Pushing the Accuracy Limit of Foundation Neural Network Models with Quantum Monte Carlo Forces and Path Integrals](#), by Anouar Benali, Jean-Philip Piquemal, et al., arXiv, April 2025 (16 pages).
- [778] [Incorporating Quantum Computing Into Genesis Mission Could Help Address the AI Energy Crunch](#), by Allison Schwartz and Senior Vice President, Global Government Relations, Public Affairs, and D-Wave, The Well News, May 2026.
- [779] [Quantum Combinatorial Reasoning for Large Language Models](#), by Carlos Flores-Garrigos, Enrique Solano, and Narendra N. Hegade, arXiv, October 2025 (9 pages).
- [780] [Opportunities and limitations of explaining quantum machine learning](#), by Elies Gil-Fuster, Jens Eisert, et al., arXiv, December 2024 (16 pages).
- [781] [eXplainable AI for Quantum Machine Learning](#), by Patrick Steinmüller, Tobias Schulz, Ferdinand Graf, and Daniel Herr, arXiv, November 2022 (14 pages).
- [782] [Explaining Quantum Circuits with Shapley Values: Towards Explainable Quantum Machine Learning](#), by Raoul Heese, Nico Piatkowski, et al., arXiv, February 2025 (41 pages).
- [783] [Quantum automated learning with provable and explainable trainability](#), by Qi Ye, Dong-Ling Deng, et al., arXiv, February 2025 (21 pages).
- [784] [QuXAI: Explainers for Hybrid Quantum Machine Learning Models](#), by Saikat Barua, Shahnewaz Siddique, et al., arXiv, June 2025 (16 pages).
- [785] [Towards Explainable Quantum AI: Informing the Encoder Selection of Quantum Neural Networks via Visualization](#), by Shaolun Ruan, Feng Liang, Rohan Ramakrishna, Chao Ren, Rudai Yan, Qiang Guan, Jiannan Li, and Yong Wang, arXiv, December 2025 (9 pages).
- [786] [Characterizing Privacy Risks of Quantum Machine Learning with Emergent Quantum-Native Access](#), by Liou Tang, James Joshi, and Ashish Kundu, arXiv, September 2026 (10 pages).
- [787] [Prospects of Privacy Advantage in Quantum Machine Learning](#), by Jamie Heredge, Marco Pistoia, et al., arXiv, May 2024 (28 pages).
- [788] [Bridging Quantum Computing and Differential Privacy: Insights into Quantum Computing Privacy](#), by Yusheng Zhao, Miao Pan, et al., arXiv, August 2024 (12 pages).
- [789] [Differential Privacy Preserving Quantum Computing via Projection Operator Measurements](#), by Yuqing Li, Yusheng Zhao, Xinyue Zhang, Hui Zhong, Miao Pan, and Chi Zhang, arXiv, April 2024 (6 pages).
- [790] [Quantum Information Ordering and Differential Privacy](#), by Naqeeb Ahmad Warsi, Ayanava Dasgupta, and Masahito Hayashi, arXiv, February 2026 (36 pages).
- [791] [Tuning Quantum Computing Privacy through Quantum Error Correction](#), by Hui Zhong, Keyi Ju, Manojna Sistla, Xinyue Zhang, Xiaoqi Qin, Xin Fu, and Miao Pan, arXiv, December 2023 (6 pages).

- [792] [Private training in quantum machine learning](#), by Tigran Sedrakyán, Frédéric Grosshans, and Elham Kashefi, arXiv, July 2026.
- [793] [Efficient Fault-Tolerant Quantum Protocol for Differential Privacy in the Shuffle Model](#), by Hassan Jameel Asghar, Arghya Mukherjee, and Gavin K. Brennen, arXiv, September 2024 (31 pages).
- [794] [Differential Privacy Preserving Distributed Quantum Computing](#), by Hui Zhong, Zhu Han, et al., arXiv, January 2025 (13 pages).
- [795] [Quantum Quandaries: Unraveling Encoding Vulnerabilities in Quantum Neural Networks](#), by Suryansh Upadhyay and Swaroop Ghosh, arXiv, February 2025 (7 pages).
- [796] [Adversarial Threats in Quantum Machine Learning: A Survey of Attacks and Defenses](#), by Archisman Ghosh, Satwik Kundu, and Swaroop Ghosh, arXiv, June 2025 (23 pages).
- [797] [Controlled Steering-Based State Preparation for Adversarial-Robust Quantum Machine Learning](#), by Sahan Sanjaya, Hari Krishna Parvatham, Emma Andrews, and Prabhat Mishra, arXiv, April 2026.
- [798] [Quantum Patches: Enhancing Robustness of Quantum Machine Learning Models](#), by Ban Q. Tran, Chuong K. Luong, Viet Q. Nguyen, Duong M. Chu, and Susan Mengel, arXiv, April 2026 (12 pages).
- [799] [Efficient Quantum Algorithm for Robust Training](#), by Yue Wang, Guangyi He, Liepeng Zhang, Lukas Gonon, and Qi Zhao, arXiv, March 2026 (33 pages).
- [800] [Quantum Properties Trojans \(QuPTs\) for Attacking Quantum Neural Networks](#), by Sounak Bhowmik, Travis S. Humble, and Himanshu Thapliyal, arXiv, July 2025 (6 pages).
- [801] [Entangled Threats: A Unified Kill Chain Model for Quantum Machine Learning Security](#), by Pascal Debus, Fabian Petsch, et al., arXiv, July 2025 (12 pages).
- [802] [Certifying and learning quantum Ising Hamiltonians](#), by Andreas Bluhm, Cambyse Rouzé, et al., arXiv, September 2025 (20 pages).
- [803] [Quantum computer: we're planning to create one that acts like a brain](#), by Michael Hartmann, January 2019.
- [804] [Machine learning & artificial intelligence in the quantum domain](#), by Vedran Dunjko and Hans J. Briegel, arXiv, September 2017 (106 pages).
- [805] [Modern applications of machine learning in quantum sciences](#), by Anna Dawid, Alexandre Dauphin, et al., arXiv, June 2025 (287 pages).
- [806] [Artificial Intelligence and Machine Learning for Quantum Technologies](#), by Mario Krenn, Jonas Landgraf, Thomas Foesel, and Florian Marquardt, arXiv, August 2022 (23 pages).
- [807] [Synergy of machine learning with quantum computing and communication](#), by Debasmita Bhoulmik, Susmita Surkolay, Latesh Kumar K. J., and Sundaraja Sitharama Iyengar, arXiv, October 2023 (34 pages).
- [808] [Variational inference with a quantum computer](#), by Marcello Benedetti, Matthias Rosenkranz, et al., arXiv, November 2021 (17 pages).
- [809] [The Rise of Quantum Robots](#), by Daniel Manzano, April 2018.
- [810] [Qubit or Qubot? Quantum Technology May Help Robots Learn Faster](#), by Matt Wayne, The Quantum Daily, March 2021.
- [811] [Experimental quantum speed-up in reinforcement learning agents](#), by Valeria Saggio, Hans J. Briegel, Philip Walther, et al., arXiv, March 2021 (10 pages).
- [812] [On quantum computing for artificial superintelligence](#), by Anna Grabowska et al., Springer, June 2024 (30 pages).
- [813] [A Survey of Quantum Transformers: Architectures, Challenges and Outlooks](#), by Hui Zhang, Qinglin Zhao, Mengchu Zhou, Li Feng, Dusit Niyato, Shenggen Zheng, and Lin Chen, arXiv, April 2025 (25 pages).
- [814] [Quantum Transformer: Accelerating model inference via quantum linear algebra](#), by Naixu Guo, Zhan Yu, Matthew Choi, Yizhan Han, Aman Agrawal, Kouhei Nakaji, Alan Aspuru-Guzik, and Patrick Rebentrost, arXiv, February 2024 (45 pages).
- [815] [QRAM: A Survey and Critique](#), by Samuel Jaques and Arthur G. Rattew, Quantum, December 2025 (51 pages).
- [816] [Quantum-enhanced Large Language Models on Quantum Hardware via Cayley Unitary Adapters](#), by Borja Aizpurua, Sukhbinder Singh, Augustine Kshetrimayum, Saeed S. Jahromi, and Roman Orus, arXiv, May 2026 (31 pages).
- [817] [Separating quantum circuits from classical LLMs](#), by Srinivasan Arunachalam, Arkopal Dutt, Hari Krovi, and Rik Sengupta, arXiv, August 2026 (60 pages).
- [818] [Walking through Hilbert Space with Quantum Computers](#), by Tong Jiang, Jinghong Zhang, Moritz K. A. Baumgarten, Meng-Fu Chen, Hieu Q. Dinh, Aadithya Ganeshram, Nishad Maskara, Anton Ni, and Joonho Lee, Chemical Reviews, May 2025 (34 pages) .
- [819] [Quantum optimization using variational algorithms on near-term quantum devices](#), by Nikolaj Moll, Daniel J. Egger, Stefan Filipp, Jay M. Gambetta, Ivano Tavernelli, Kristan Temme, et al., arXiv, October 2017 (30 pages).
- [820] [Simulating quantum field theory with a quantum computer](#), by John Preskill, arXiv, November 2018 (22 pages).
- [821] [Computation of Molecular Spectra on Quantum Processor with an Error-Resilient Algorithm](#), by J. I. Colless et al., Physical Review X, February 2018 (7 pages).
- [822] [Preparation and coherent manipulation of pure quantum states of a single molecular ion](#), by Chin-wen Chou, Christoph Kurz, David B. Hume, Philipp N. Plessow, David R. Leibrandt, and Dietrich Leibfried, arXiv, February 2017 (38 pages).
- [823] [Simulation of Electronic Structure Hamiltonians Using Quantum Computers](#), by James D. Whitfield, Jacob Biamonte, and Alán Aspuru-Guzik, arXiv, December 2010 (22 pages).
- [824] [Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets](#), by Abhinav Kandala, Antonio Mezzacapo, Kristan Temme, Maika Takita, Markus Brink, Jerry M. Chow, and Jay M. Gambetta, Nature, September 2017  (preprint on [arXiv](#)).
- [825] [Quantum Equation of Motion with Orbital Optimization for Computing Molecular Properties in Near-Term Quantum Computing](#), by Phillip W. K. Jensen, Erik Rosendahl Kjellgren, Peter Reinholdt, Karl Michael Ziem, Sonia Coriani, Jacob Kongsted, and Stephan P. A. Sauer, arXiv, May 2024 (14 pages).
- [826] [Simulated Quantum Computation of Molecular Energies](#), by Alán Aspuru-Guzik, Anthony D. Dutoi, Peter J. Love, and Martin Head-Gordon, arXiv, April 2006 (20 pages).
- [827] [Quantum Machine Learning for Electronic Structure Calculations](#), by Rongxin Xia and Sabre Kais, arXiv, October 2018 (16 pages).
- [828] [Solving strongly correlated electron models on a quantum computer](#), by Dave Wecker, Matthew B. Hastings, Nathan Wiebe, Bryan K. Clark, Chetan Nayak, and Matthias Troyer, arXiv, August 2015 (25 pages).
- [829] [Reliably assessing the electronic structure of cytochrome P450 on today's classical computers and tomorrow's quantum computers](#), by Joshua J. Goings et al., PNAS, February 2022 (24 pages).
- [830] [Quantum computing enhanced computational catalysis](#), by Vera von Burg, Guang Hao Low, Thomas Häner, Damian S. Steiger, Markus Reiher, Martin Roetteler, and Matthias Troyer, arXiv, March 2021 (19 pages).
- [831] [Quantum Computation for Chemistry and Materials](#), by Jarrod McClean, 2018 (36 slides).

- [832] [An adaptive variational algorithm for exact molecular simulations on a quantum computer](#), by Sophia Economou et al., *Nature Communications*, July 2019 (9 pages).
- [833] [An adaptive variational algorithm for exact molecular simulations on a quantum computer](#), by Harper R. Grimsley, Sophia E. Economou, Edwin Barnes, and Nicholas J. Mayhall, *Nature Communications*, July 2019.
- [834] [Quantum Computation for Periodic Solids in Second Quantization](#), by Aleksei V. Ivanov et al., *PRR*, 2022 (29 pages).
- [835] [Accurate structure prediction of biomolecular interactions with AlphaFold 3](#), by Josh Abramson, Demis Hassabis, John M. Jumper, et al., *Nature*, May 2024 (8 pages).
- [836] [De novo design of protein structure and function with RFdiffusion](#), by Joseph L. Watson, David Baker, et al., *Nature*, July 2023 (12 pages).
- [837] [Quantum Computing \(and Quantum Information Science\)](#), by Steve Binkley, US Department of Energy, 2016 (23 slides).
- [838] [Finding low-energy conformations of lattice protein models by quantum annealing](#), by Alejandro Perdomo-Ortiz, Neil Dickson, Marshall Drew-Brook, Geordie Rose, and Alán Aspuru-Guzik, *Scientific Reports*, August 2012.
- [839] [Resource-efficient quantum algorithm for protein folding](#), by Anton Robert, Panagiotis Kl. Barkoutsos, Stefan Woerner, and Ivano Tavernelli, *npj Quantum Information*, February 2021 (5 pages).
- [840] [A QUBO model of the RNA folding problem optimized by variational hybrid quantum annealing](#), by Tristan Zaborniak, Juan Giraldo, Hausi Müller, Hosna Jabbari, and Ulrike Stege, *arXiv*, August 2022 (12 pages).
- [841] [Quantum Information and Computation for Chemistry](#), by Jonathan Olson et al., *arXiv*, 2017 (60 pages).
- [842] [A Comparison of the Bravyi–Kitaev and Jordan–Wigner Transformations for the Quantum Simulation of Quantum Chemistry](#), by Andrew Tranter, 2018 (14 pages).
- [843] [Creating and manipulating a Laughlin-type \$\mu = 1/3\$ fractional quantum Hall state on a quantum computer with linear depth circuits](#), by Armin Rahmani, Kevin J. Sung, Harald Putterman, Pedram Roushan, Pouyan Ghaemi, and Zhang Jiang, *arXiv*, November 2020 (7 pages).
- [844] [Exponential quantum speedup in simulating coupled classical oscillators](#), by Ryan Babbush, Dominic W. Berry, Robin Kothari, Rolando D. Somma, and Nathan Wiebe, *arXiv*, September 2023 (43 pages).
- [845] [Challenges and opportunities in quantum optimization](#), by Amira Abbas, Christa Zoufal, et al., *Nature Reviews Physics*, October 2024 (preprint on research.tilburguniversity.edu).
- [846] [Variational and Annealing-Based Approaches to Quantum Combinatorial Optimization](#), by Hala Hawashin, Deep Nath, and Marco Alberto Javarone, *arXiv*, March 2026 (23 pages).
- [847] [Hybrid Quantum-Classical Optimisation of Traveling Salesperson Problem](#), by Christos Lytrosyngounis and Ioannis Lytrosyngounis, *arXiv*, February 2025 (14 pages).
- [848] [A quantum speedup algorithm for TSP based on quantum dynamic programming with very few qubits](#), by Bai Xujun and Shang Yun, *arXiv*, April 2025 (15 pages).
- [849] [Rich Vehicle Routing Problem: Survey](#), by Jose Caceres Cruz et al., *ACM Computing Surveys*, December 2014 (30 pages).
- [850] [The Traveling Salesman Problem: A Computational Study](#), by William J. Cook et al, *Princeton Series in Applied Mathematics*, 2007 (608 pages).
- [851] [VeloxQ: A Fast and Efficient QUBO Solver](#), by J. Pawlowski, J. Tuziemski, P. Tarasiuk, A. Przybysz, R. Adamski, K. Hendzel, Ł. Pawela, and B. Gardas, *arXiv*, January 2025 (20 pages).
- [852] [Breaking the Sorting Barrier for Directed Single-Source Shortest Paths](#), by Ran Duan, Longhui Yin, et al., *Proceedings of the 57th Annual ACM Symposium on Theory of Computing, STOC 2025, April 2025* (17 pages).
- [853] [Towards Exponential Quantum Improvements in Solving Cardinality-Constrained Binary Optimization](#), by Haomu Yuan, Hanqing Wu, Kuan-Cheng Chen, Bin Cheng, and Crispin H. W. Barnes, *arXiv*, March 2026 (19 pages).
- [854] [A Nested Amplitude Amplification Protocol for the Binary Knapsack Problem](#), by Laurin Demmler and Maximilian Hess, *arXiv*, April 2026 (11 pages).
- [855] [Constraint-oriented biased quantum search for linear constrained combinatorial optimization problems](#), by Sören Wilkening, Timo Ziegler, and Maximilian Hess, *arXiv*, December 2025 (13 pages).
- [856] [Quantum algorithms for path and cycle containment problems](#), by Arjan Cornelissen, Amin Shiraz Gilani, and Subhasree Patro, *arXiv*, May 2026 (37 pages).
- [857] [\(Sub\)Exponential Quantum Speedup for Optimization](#), by Jiaqi Leng, Kewen Wu, Xiaodi Wu, and Yufan Zheng, *arXiv*, April 2025 (69 pages).
- [858] [Topological Obstructions in Quantum Adiabatic Algorithms](#), by Prathamesh S. Joshi and Emil Prodan, *arXiv*, March 2026 (9 pages).
- [859] [Exponential Quantum Speedup on Structured Hard Instances of Maximum Independent Set](#), by Vicky Choi, *arXiv*, January 2026 (18 pages).
- [860] [Quantum Algorithms for the Monte Carlo Method](#), by Yassine Hamoudi, 2021 (175 pages).
- [861] [Quantum circuits for the Metropolis-Hastings algorithm](#), by Baptiste Claudon, Pablo Rodenas-Ruiz, Jean-Philip Piquemal, and Pierre Monmarché, *Journal of Physics A: Mathematical and Theoretical*, volume 59, number 30, 305304, July 2026 (9 pages).
- [862] [Resource-Scalable Fully Quantum Metropolis-Hastings for Integer Linear Programming](#), by Gabriel Escrig, Roberto Campos, and M. A. Martin-Delgado, *arXiv*, February 2026 (22 pages).
- [863] [A hybrid algorithm for quadratically constrained quadratic optimization problems](#), by Hongyi Zhou, Sirui Peng, Qian Li, and Xiaoming Sun, *arXiv*, September 2023 (8 pages).
- [864] [Solving Optimal Power Flow using a Variational Quantum Approach](#), by Thinh Viet Le, Mark M. Wilde, and Vassilis Kerkatos, *arXiv*, August 2025 (22 pages).
- [865] [Quantum Integer Programming \(QuIP\) 47-779: Lecture Notes](#), by David E. Bernal, Sridhar Tayur, and Davide Venturelli, *arXiv*, January 2021 (71 pages).
- [866] [Quantum Speedups for Group Relaxations of Integer Linear Programs](#), by Brandon Augustino, Shouvanik Chakrabarti, et al., *arXiv*, February 2026 (46 pages).
- [867] [Quantum algorithm for total least squares data fitting](#), by Hefeng Wang and Hua Xiang, *arXiv*, June 2019 (7 pages).
- [868] [Benchmarking a heuristic Floquet adiabatic algorithm for the Max-Cut problem](#), by Etienne Granet and Henrik Dreyer, *arXiv*, April 2024 (6 pages).
- [869] [Variational Quantum Algorithms for Semidefinite Programming](#), by Dhruvil Patel, Patrick J. Coles, and Mark M. Wilde, *arXiv*, June 2024 (40 pages).
- [870] [Semidefinite Programming in Quantum Information Science](#), by Paul Skrzypczyk and Daniel Cavalcanti, *arXiv*, June 2023 (31 pages).
- [871] [Quantum Fisher-Yates shuffle: Unifying methods for generating uniform superpositions of permutations](#), by Lennart Binkowski and Marvin Schwiering, *arXiv*, April 2025 (12 pages).
- [872] [ProvideQ: A Quantum Optimization Toolbox](#), by Domenik Eichhorn, Nick Poser, Maximilian Schweikart, and Ina Schaefer, *arXiv*, July 2025 (9 pages).
- [873] [A Domain-Specific Language for Formulating Hybrid Quantum-Classical Meta-Solver Strategies](#), by Nick Poser, Domenik Eichhorn, and Ina Schaefer, *arXiv*, August 2026 (13 pages).

- [874] [Principles of Quantum Optimization for Constrained Problems](#), by Einar Gabbassov, Gurpahul Singh, and Achim Kempf, arXiv, July 2026 (25 pages).
- [875] [Quantum advantage from soft decoders](#), by André Chailoux and Jean-Pierre Tillich, arXiv, November 2024 (33 pages).
- [876] [Optimization by Decoded Quantum Interferometry](#), by Stephen P. Jordan, Ryan Babbush, et al., arXiv, May 2025 (83 pages).
- [877] [From Constraint to Code: DQI-Kit – A Software Framework for Decoded Quantum Interferometry](#), by Simon Thelen and Wolfgang Mauere, arXiv, May 2026 (12 pages).
- [878] [A nearly linear-time Decoded Quantum Interferometry algorithm for the Optimal Polynomial Intersection problem](#), by Ansis Rosmanis, arXiv, January 2026 (52 pages).
- [879] [Quantum Decoding Algorithms: Quantum Speedups in Optimization](#), by Jan Ljubaš and Tim Byrnes, arXiv, May 2026 (24 pages).
- [880] [Verifiable Quantum Advantage via Optimized DQI Circuits](#), by Tanuj Khattar, Ryan Babbush, Stephen P. Jordan, et al., arXiv, October 2025 (52 pages).
- [881] [Towards solving industrial integer linear programs with Decoded Quantum Interferometry](#), by Francesc Sabater, Jean-Francois Bobier, Carlos A. Riofrio, et al., arXiv, September 2025 (47 pages).
- [882] [No Quantum Advantage in Decoded Quantum Interferometry for MaxCut](#), by Ojas Parekh, arXiv, September 2025 (12 pages).
- [883] [Benchmarking Techniques for Decoded Quantum Interferometry](#), by Leon Bollmann and Maximilian Hess, arXiv, March 2026.
- [884] [Approximate sampling from decoded quantum interferometry via Markov chain Monte Carlo methods](#), by Elies Gil-Fuster, Matan Ninio, Lennart Bittel, Yishai Shimoni, Jens Eisert, Stefan Woerner, and Almudena Carrera Vázquez, arXiv, July 2026 (24 pages).
- [885] [On the Complexity of Decoded Quantum Interferometry](#), by Kunal Marwaha, Bill Fefferman, Alexandru Gheorghiu, and Vojtech Havlicek, arXiv, September 2025 (35 pages).
- [886] [Decoded Quantum Interferometry Requires Structure](#), by Eric R. Anschuetz, David Gamarnik, and Jonathan Z. Lu, arXiv, September 2025 (51 pages).
- [887] [Algebraic Geometry Codes and Decoded Quantum Interferometry](#), by Andi Gu and Stephen P. Jordan, arXiv, October 2025 (23 pages).
- [888] [Quantum random walks](#), by L. Davidovich Yakir Aharonov and N. Zagury, Physical Review A, 1993.
- [889] [Exponential speedups for quantum walks in random hierarchical graphs](#), by Shankar Balasubramanian, Tongyang Li, and Aram Harrow, arXiv, August 2024 (69 pages).
- [890] [On the relationship between continuous- and discrete-time quantum walk](#), by Andrew M. Childs, arXiv, October 2009 (22 pages).
- [891] [Universal computation by quantum walk](#), by Andrew M. Childs, arXiv, June 2008 (9 pages).
- [892] [Quantum walks on a programmable two-dimensional 62-qubit superconducting processor](#), by Ming Gong, Jian-Wei Pan, et al., arXiv, July 2021 (13 pages).
- [893] [Two-Dimensional Quantum Walk of Correlated Photons](#), by Zhi-Qiang Jiao, Jun Gao, Wen-Hao Zhou, Xiao-Wei Wang, Ruo-Jing Ren, Xiao-Yun Xu, Lu-Feng Qiao, and Xian-Min Jin, arXiv, July 2020 (22 pages).
- [894] [Resource-efficient universal photonic processor based on time-multiplexed hybrid architectures](#), by Jonas Lammers, Christine Silberhorn, et al., arXiv, September 2025 (11 pages).
- [895] [A Short Review on Novel Approaches for Maximum Clique Problem: from Classical algorithms to Graph Neural Networks and Quantum algorithms](#), by Raffaele Marino, Lorenzo Buffoni, and Bogdan Zavalnij, arXiv, March 2024 (24 pages).
- [896] [Exponential algorithmic speedup by quantum walk](#), by Andrew M. Childs, Richard Cleve, Enrico Deotto, Edward Farhi, Sam Gutmann, and Daniel A. Spielman, arXiv, October 2002 (24 pages).
- [897] [Selected Open Problems in Continuous-Time Quantum Walks](#), by Gabriel Coutinho and Krystal Guo, arXiv, April 2024 (16 pages).
- [898] [World TSP](#), by University of Waterloo.
- [899] [Galaxy TSP](#), by Iddo Drori, Brandon Kates, William R. Sickinger, Anant Kharkar, Brenda Dietrich, Avi Shporer, and Madeleine Udell, MIT, Cornell University, Columbia University, 2020.
- [900] [Review on Quantum Walk Computing: Theory, Implementation, and Application](#), by Xiaogang Qiang, Shixin Ma, and Haijing Song, arXiv, April 2024 (61 pages).
- [901] [Benchmarking Quantum Heuristics: Non-Variational QWOA for Weighted Maxcut](#), by Tavis Bennett, Jingbo Wang, et al., arXiv, May 2025 (14 pages).
- [902] [Benchmarking Lie-Algebraic Pretraining and Non-Variational QWOA for the MaxCut Problem](#), by Matthias Zering, Jolyon Joyce, Tal Gurfinkel, and Jingbo Wang, arXiv, December 2025 (18 pages).
- [903] [Quantum Walks on Simplicial Complexes and Harmonic Homology: Application to Topological Data Analysis with Superpolynomial Speedups](#), by Ryu Hayakawa, Kuo-Chin Chen, and Min-Hsiu Hsieh, arXiv, April 2024 (40 pages).
- [904] [Practical Quantum Topological Data Analysis with Applications to High-Dimensional Feature Extraction and Time Series Analysis](#), by Jason Iaconis, Sayonee Ray, Samwel Sekwao, Claudio Girotto, and Martin Roetteler, arXiv, July 2026 (38 pages).
- [905] [Quantum Walks](#), by Daniel Reitzner, Daniel Nagaj, and Vladimir Buzek, arXiv, May 2013 (124 pages).
- [906] [Multiparticle quantum walks for distinguishing hard graphs](#), by Sachin Kasture, Shaheen Acheche, Loic Henriot, and Louis-Paul Henry, arXiv, March 2025 (11 pages).
- [907] [Quantum Walks on Arbitrary Spatial Networks with Rydberg Atoms](#), by Gabriel Almeida, Raul Santos, Lara Janiurek, and Yasser Omar, arXiv, July 2025 (7 pages).
- [908] [Frameworks for Quantum Algorithms](#), by Stacey Jeffery, 2014 (166 pages).
- [909] [Unitary and Open Scattering Quantum Walks on Graphs](#), by Alain Joye, arXiv, September 2024 (54 pages).

This **Higher level algorithms** subsection contains 601 bibliographical references and notes containing at least 15,142 pages.

1.4 NISQ algorithms

The Noisy Intermediate Scale Quantum computing era was defined by John Preskill in his keynote address at the first Q2B conference from QC Ware in California in December 2017 and laid out in a paper published in *Quantum* in 2018 [910]. He conjectured that it may be possible to use quantum computers with at least 50 qubits that could perform tasks which surpass the capabilities of today’s classical digital computers but would be limited to about 1,000 quantum gates.

NISQ proposals therefore usually favor shallow circuits and algorithms that tolerate imperfect initialization, gates and readout. In some settings, noise can even be exploited as a source of stochasticity, for example in quantum machine learning tasks [911]. John Preskill also included analog quantum computing in the NISQ era [912]. As of 2026, error suppression and quantum error mitigation have enabled experiments with up to 13,800 two-qubit gates [913]. This does not by itself prove that NISQ computing can scale. The sampling overhead of quantum error-mitigation (QEM) approaches can grow exponentially with circuit depth, and worst-case bounds become superpolynomial even at shallow depths as system size and noise accumulate [914, 915]. Variational optimization adds a second scaling constraint because the precision required to resolve the cost landscape can become increasingly demanding with problem size [916].

1.4.1 Algorithms classes

The best-known quantum algorithms proposed for NISQ systems belong to the broad variational quantum algorithm (VQA) class. They use variational quantum circuits (VQCs), also called parameterized quantum circuits [917] (Figure 61). On noisy hardware, these circuits should remain as shallow as possible. There is no universal depth threshold because the tolerable depth depends on the noise model, connectivity, observable, mitigation strategy and required output precision.

This class includes VQE (variational quantum eigensolvers [162, 918]) for quantum physics simulations, QAOA (quantum approximate optimization algorithm [161]) for various optimization tasks, VQLS (variational quantum linear solvers [919]) to solve linear equations and QML (quantum machine learning) for various machine learning and deep learning use cases.

A comprehensive 2026 review from Peking University covers the whole VQA stack, from ansatz design and cost-function formulation to gradient computation, optimizer selection and barren plateaus, then asks whether VQAs survive the transition to eFTQC and FTQC [920]. VQAs are not incompatible with fault tolerance, but their original NISQ rationale weakens once logical error correction is available. Error correction suppresses hardware noise, not the intrinsic costs of training, repeated expectation-value estimation, barren plateaus or the classical optimizer. Parameterized rotations must also be

synthesized into the fault-tolerant gate set, introducing non-Clifford gate and magic-state costs. Small fault-tolerant VQE constructions have been demonstrated in simulation up to 16 qubits [921], so viability is established in principle but not scalability. For large-scale eigenvalue estimation, coherent algorithms such as QPE become stronger competitors, although their own state-preparation and overlap costs can also scale badly [922]. VQAs may therefore survive in eFTQC and FTQC as state-preparation, warm-start, variational compilation or limited-T-depth subroutines rather than as the dominant large-scale algorithmic paradigm. The design of the ansatz itself is now frequently delegated to automated search, an area reviewed under the label of quantum architecture search [923], and even to LLM-based agents that generate, evaluate and refine circuit candidates [924]. These automated approaches add a classical cost that is rarely accounted for in the overall resource budget. Selecting ansatz families that are simultaneously expressive and trainable remains an unsolved trade-off [925].

Many other species of NISQ VQA algorithms are also proposed, particularly in chemical simulations [926–928] and for search [929]. These are most of the time heuristic algorithms that determine near-optimal solutions to various forms of optimization problems, VQE, QAOA and QML being all various breeds of optimization problems to find energy or cost function minima.

Variational algorithms are hybrid by design, with a substantial fraction of the work performed classically. The training problem is NP-hard in the worst case for broad VQA families [930]. This does not mean that every instance or optimizer cost scales exponentially. A separate 2025 preprint identifies undecidable questions associated with some variational formulations [931]. The classical part can itself use tensor networks [932, 933], and there are even proposals to use a quantum annealer to optimize variational-circuit parameters [934]. The resulting resource comparison must therefore include the complete quantum-classical loop.

There is no universal qubit-count threshold for quantum advantage. Preskill’s original discussion used roughly 50 to 100 qubits as the regime in which brute-force statevector simulation starts to become difficult, but practical advantage depends much more strongly on circuit structure, depth, noise, observable and the best classical algorithm used as a baseline. An exact complex-double statevector requires about 16 GB for 30 qubits, 16 TB for 40 qubits and 16 PB for 50 qubits. This memory argument alone is insufficient because shallow, low-entanglement or otherwise structured circuits can often be simulated much farther with tensor networks and neural quantum states (NQS). For example, a classical variational simulation reproduced $p = 4$ QAOA circuits with 54 qubits on modest classical resources [935]. Also, a variational algorithm can become untrainable well below 30 qubits. The relevant threshold is therefore not a qubit count but the point where the full

quantum workflow beats a best-in-class classical method at matched solution quality and total wall-clock cost.

A laptop, a single cloud server or server cluster is always cheaper than a quantum computer in that case. As shown in Figure 60, a majority of published use cases in arXiv as of 2026 were still in the “laptop equivalent” realm with below 16 physical qubits.

Most published variational demonstrations still use small problem instances, often fewer than 30 qubits, especially in QML and VQE. Most VQAs have not demonstrated end-to-end scaling advantage as the problem size grows through roughly the 30 to 50 qubit range. The main problems are circuit noise, increasingly costly expectation-value estimation, optimizer sensitivity and barren plateaus, while the strongest classical comparables improve with tensor networks, neural quantum states, symmetry reduction and problem-specific heuristics [916, 936]. Still, experiments above 100 physical qubits are now commonplace, particularly for many-body physics workloads, but a large qubit count by itself does not demonstrate algorithmic scaling or quantum advantage.

Another concern is that identifying the best-in-class classical solution is difficult, which complicates the creation of fair quantum speedup assessments. Classical and quantum algorithms may also yield different results, for example a complete classical solution versus an expectation value or samples from a quantum workflow, or an exact solution vs an approximate solution for an optimization task. Comparisons should therefore match output quality, success probability, precision, preprocessing, post-processing and total wall-clock cost.

When trying to obtain a quantum speedup, the breadth and depth required by many variational algorithms remain difficult to reconcile with current NISQ noise levels and connectivity [938]. Quantum error suppression and mitigation can extend the reachable circuit regime, but they do not provide a generic path to arbitrary scale. Rigorous bounds show exponential sampling overhead with circuit depth for broad noise and mitigation models [914], and worst-case superpolynomial overheads can arise as circuit width and depth grow [915]. The shallow or weakly entangled circuits favored by NISQ hardware are often precisely those that tensor-network or neural-state methods can simulate efficiently [935].

It can be done efficiently, which means “at most in polynomial time”, but not necessarily faster than a quantum computer. And there are only a few benchmarks yet done in that regime [939].

Hardware resource and time estimation is a key quantum computing discipline. It creates a bridge between practical use cases, their related algorithms and their required physical resources and computing time. Late 2022, Microsoft released its resource estimator for fault-tolerant quantum computing algorithms [940]. PsiQuantum created such a tool, first dedicated to linear differential

equation solving [941] and then, was extended as the Bartiq tool for generic use cases.

No comparably generic estimator is widely used for the complete NISQ or VQA workflow, including the number of circuit evaluations needed for observables, the number of optimizer iterations, mitigation overhead and classical preprocessing and post-processing, as summarized in Figure 63 [942]. Early fault-tolerant QPUs may make many present-day NISQ circuits executable at the logical level, but a logical error rate in the 10^{-4} to 10^{-6} range is not by itself sufficient to establish usefulness. The dominant cost can move from physical noise to repeated logical circuit execution, non-Clifford synthesis, measurements and optimization. This is particularly important for VQE and other expectation-value-based VQAs.

There are, however, some very interesting review papers documenting well this aspect, at least for VQE algorithms [943]. At the same time, any estimation of NISQ resources should be compared to an estimation of the classical computing resources required to solve the same problem.

At present, there is a lack of estimators for such best-in-class classical algorithms computing resources. This is always done on a case-by-case analysis, and comparing things with a moving classical target, often in different circumstances, with or without heuristic approaches. And the results are usually disappointing [944].

1.4.2 Qubit quality

A rough unmitigated error-budget estimate links the physical error probability to the number of fault locations in a circuit [54, 945]. It is an approximate rule of thumb. The error rate considered corresponds to the gates with the lowest fidelity, which corresponds with most qubit technologies to two-qubit gates like a CNOT [946]:

$$\text{qubit gates error rate} \ll \frac{1}{\text{algo breadth} * \text{algo depth}}$$

The breadth corresponds to the number of qubits used in the algorithms and its depth, to the number of quantum gate cycles. It is a sort of quantum algorithm quantum volume. You could make some trade-offs here between these two dimensions and run either a very shallow algorithm with more qubits or a deeper algorithm with fewer qubits.

If one assumes independent stochastic errors, comparable error probabilities at each relevant fault location and no mitigation, the inverse breadth-times-depth expression gives only an order-of-magnitude condition for having substantially fewer than one error per circuit. Real devices have nonuniform gates, crosstalk, coherent errors, idle errors, routing overhead and correlated noise, so the number of actual fault locations is more meaningful than qubit count times logical depth [947].

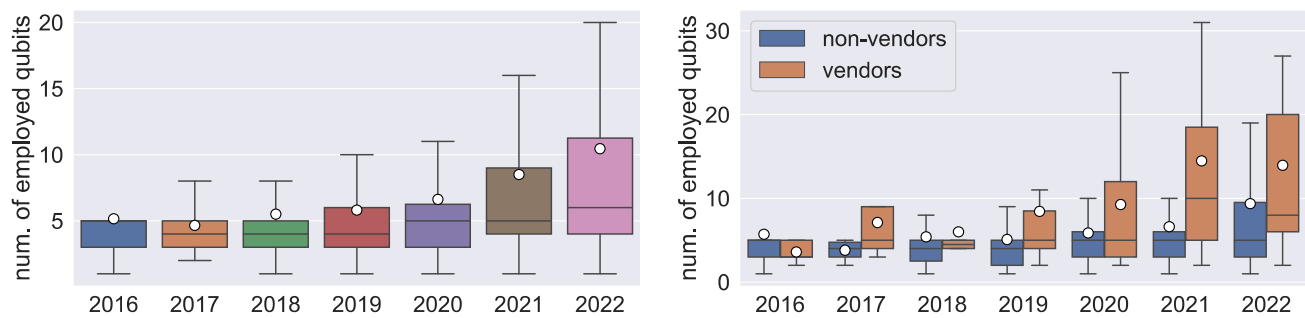


Figure 60: How many qubits were used in the algorithms and case studies preprints on arXiv? Source: [937] which analyzed 750 preprints on arXiv between 2016 and 2022. Added in 2023.

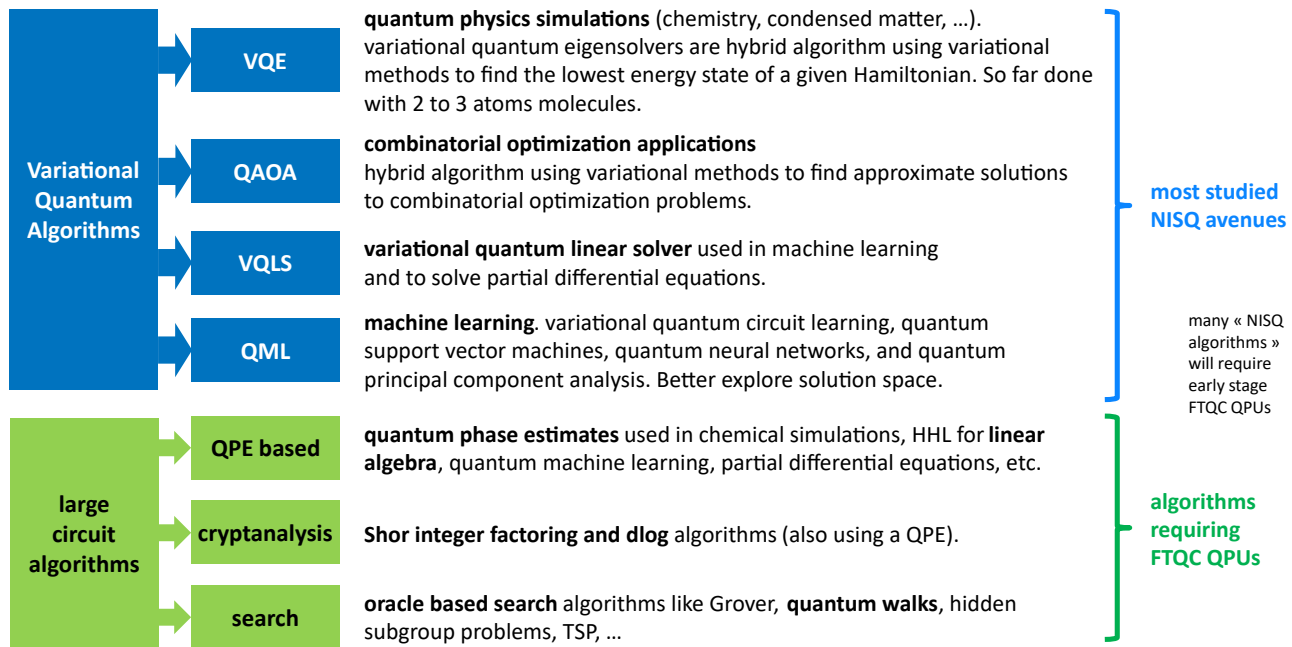


Figure 61: The main NISQ algorithm classes proposed by researchers and industry vendors are shown in blue, while the green classes generally require fault-tolerant quantum computers. Variational algorithms were designed to keep circuits relatively shallow, but their end-to-end scalability is constrained by a three-way tension between circuit expressivity, trainability and classical simulability. VQE scaling depends on the Hamiltonian, mapping, ansatz, connectivity, compilation and target precision. Some non-variational or partially variational methods may become preferable as hardware moves toward FTQC. © Olivier Ezratty, 2023-September 2026.

Applying that simplified rule illustrates the tension shown in Table 7. For example, 50 qubits times eight error-sensitive layers gives about 400 nominal fault locations, corresponding to an error probability well below $1/400$ if one wants much less than one error on average. This should be read as an illustrative estimate, not as a universal 99.7% fidelity requirement, because the compiled two-qubit gate count and the error model determine the actual budget [948].

As of 2026, some QPUs could reach such two-qubit gate fidelities with over 50 qubits, namely IBM's Heron r3 with 156 qubits and 99.85% median two-qubit CZ gate fidelity, and Quantinuum Helios with 98 qubits and 99.92% two-qubit gate fidelity. Other vendors are still off the mark, like Rigetti with its 108-qubit Cepheus system released in 2026 with only 99.1% two-qubit gate

fidelities and, later, a 336 qubits version which would barely reaching 99.5% fidelities, which is clearly insufficient for running any NISQ algorithm with that qubit number.

Most two-qubit gate fidelities provided by industry vendors are median or average fidelities. What is important is their standard deviation and minimum values [949]. Good median fidelities with high standard deviation are not at all practical, particularly for the first gates of a given algorithm. High error rates can irreversibly damage early on most running algorithm [950].

One solution is to deactivate, after calibration, the adjacent qubits for which hardware defects create "stable" faulty two-qubit gates. Still, even with these precautions, the publicized median two-qubit gate fidelities are still not good enough to run NISQ algorithms suc-

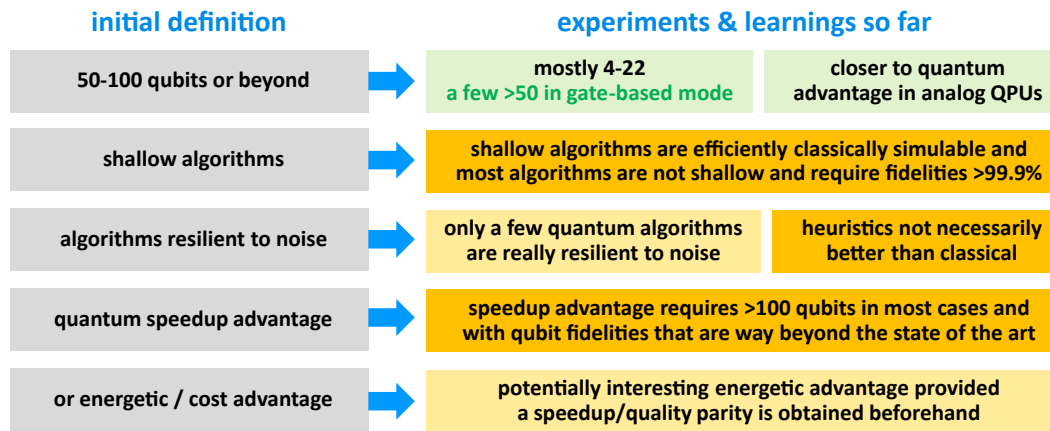


Figure 62: From John Preskill’s NISQ definition to actual experiments and learnings. Analog computing refers to digital annealing (D-Wave) and quantum simulators (Pasqal, QuEra, ...). Shallow algorithms have only a few gate cycles, preferably under 10. Most variational algorithms have a much larger number of gate cycles in a potential quantum advantage speedup regime. Source: © Olivier Ezratty, 2023-August 2024.

cessfully. It is also the case with ion-trap qubits which have very good fidelities but are seemingly hard to scale beyond a couple dozen qubits preventing developers from obtaining a space-related computing advantage. These qubits are also too slow to drive, damaging their potential to generate a speedup in a quantum advantage regime. This doesn’t show up with experiments implemented with fewer than 25 qubits.

In their current plans and roadmaps, most QPUs are not expected to scale-in beyond a couple of hundred qubits with supporting the high two-qubit gate fidelities required for either NISQ or FTQC, above 99.9%.

The most frequently retained option is a scale-out approach with connecting several QPUs together, a bit like with distributed and parallel computing used in high-performance computing (HPC) [951]. These connections must preserve qubits overall entanglement and fidelities.

Only a few quantum computing companies have started to work on this next stage which could be explored in a parallel way to the development of their QPU. Scale-out architectures can use multiple techniques like microwave guides between qubits or entangled photons-based connections and qubit transduction.

Specialized quantum information network startups like Welinq (France) have started to build quantum links based on entangled photon-based connections, also providing quantum memories capabilities for computing and intermediate communication buffers. QPhoX (The Netherlands) focuses on qubit transduction from microwave to optical photons.

With several hundred or a thousand qubits, an uncorrected circuit containing many two-qubit gates can require error probabilities far below present hardware values if one demands a low probability of any fault. The exact fidelity requirement cannot be inferred from qubit count alone. It depends on the compiled gate

count, noise correlations, circuit observable and mitigation method. This is especially restrictive because useful variational circuits generally become deeper as the problem size and ansatz expressivity increase.

There is another notable difference between NISQ and FTQC implementations. NISQ variational algorithms make extensive use of continuously parameterized rotations such as $R_x(\theta)$, $R_y(\theta)$ and $R_z(\theta)$. Their angles are updated by the classical optimizer, while the corresponding quantum state is prepared natively on the quantum processor.

These R arbitrary rotations gates must be implemented with very high precision, which is constrained, among other aspects, by the quality of their electronic drive [956]. It leads to the notion of parameterized gates, in the case for example of superconducting qubits, where some optimization is implemented in the ansatz preparation to bypass the very notion of gates and drive the qubits directly at the pulse level controlling the rotations within the Bloch sphere. There are various implementation proposals [957–962]. A 2026 Hitachi Cambridge Laboratory proposal builds pulse-optimised circuit elements specifically for quantum chemistry ansatzes, trading the gate abstraction for shorter and more noise-resilient pulse sequences [963]. The gain is real but bounded, since it compresses a constant factor and does not change the exponential iteration scaling documented above.

In FTQC, logical operations must be implemented through a gate set supported fault-tolerantly by the chosen code. A common abstract choice is Clifford+ T . The Eastin-Knill theorem prevents any quantum error-correcting code from having a universal set of transversal gates, so it is incorrect to describe T and Toffoli as generically transversal. In surface-code-like schemes, non-Clifford operations are commonly supplied through magic-state injection and distillation or related code-specific constructions [954].

NISQ gate-based hardware resource requirements

resources	initial NISQ assumptions	scaling reality and constraints												
qubit number	around 50-100 physical qubits were initially viewed as a regime where brute-force classical statevector simulation becomes difficult [952].	practical VQA advantage is problem-dependent and may require hundreds or more qubits [314, 953]. Shallow and structured circuits with substantially more than 30 qubits may remain classically simulable with tensor networks or related methods.												
circuit depth	NISQ algorithms should remain shallow enough that accumulated hardware errors do not destroy the useful signal.	useful VQA instances often require a circuit depth that grows with problem size. After compilation and connectivity-induced routing, hundreds or more two-qubit gate layers may be required, while shallow circuits are accessible to efficient classical simulations.												
available fidelities	NISQ computation relies directly on physical gate fidelities without QEC.	two-qubit gate fidelities are the usual limitations. Average or median fidelity is insufficient since variability, coherent errors, crosstalk, connectivity and correlated noise also determine the effective circuit error. Quantum error detection and mitigation can help extend the reach of NISQ [954].												
required fidelities	order-of-magnitude condition for a circuit without QEM: $p_{\text{err}} \ll 1/N_{\text{fault}}$ with p_{err} being a gate/idle error rate and N_{fault} the number of faulty operations in the circuit [54], considering that a good approximation is $N_{\text{fault}} = \text{qubits} * \text{algo depth}$ due to noisy idle operations.	the simple condition $p_{\text{err}} \sim 1/N_{\text{fault}}$ corresponds only to about one expected stochastic fault: <table style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>N_{fault}</th> <th>$p_{\text{err}} \sim 1/N_{\text{fault}}$</th> <th>corresponding fidelity</th> </tr> </thead> <tbody> <tr> <td>10^3</td> <td>10^{-3}</td> <td>99.9%</td> </tr> <tr> <td>10^4</td> <td>10^{-4}</td> <td>99.99%</td> </tr> <tr> <td>10^5</td> <td>10^{-5}</td> <td>99.999%</td> </tr> </tbody> </table> useful computation generally requires a substantially smaller total effective error, unless error mitigation or algorithm-specific noise resilience can recover the observable.	N_{fault}	$p_{\text{err}} \sim 1/N_{\text{fault}}$	corresponding fidelity	10^3	10^{-3}	99.9%	10^4	10^{-4}	99.99%	10^5	10^{-5}	99.999%
N_{fault}	$p_{\text{err}} \sim 1/N_{\text{fault}}$	corresponding fidelity												
10^3	10^{-3}	99.9%												
10^4	10^{-4}	99.99%												
10^5	10^{-5}	99.999%												
qubit gate set	variational algorithms use arbitrary single-qubit rotations gates such as R_x , R_y and R_z together with entangling gates (CNOT, CZ).	arbitrary rotations gates require very high precision and finely tuned calibration. In FTQC, arbitrary logical rotations are synthesized from a Clifford+ T gate set with magic-state preparation and distillation [955].												

Table 7: Illustrative NISQ gate-based hardware resource requirements. Neither a qubit count nor the inverse breadth-times-depth expression defines a universal quantum-advantage or fidelity threshold. A more relevant quantity is the number and distribution of error-sensitive fault locations in the compiled circuit, together with the physical noise model. The table also highlights a central difficulty of variational NISQ algorithms: circuits shallow and structured enough to remain trainable and resilient to noise are often also easier to approximate or simulate classically. In FTQC, physical noise can be suppressed by quantum error correction, but variational algorithms retain measurement, optimization and trainability costs, while arbitrary rotations acquire a fault-tolerant synthesis overhead. Source: © Olivier Ezratty, 2023–September 2026.

Continuously parameterized rotations can be synthesized to a target precision from Clifford+ T sequences. Solovay-Kitaev establishes efficient universal compilation in principle, but modern synthesis is much more efficient. For a typical ancilla-free R_z rotation, Ross-Selinger synthesis gives a T count of approximately $3 \log_2(1/\epsilon) + O(\log \log(1/\epsilon))$ for target error ϵ [955]. This non-Clifford synthesis cost is one of the additional resource terms that a fault-tolerant VQA must pay.

1.4.3 Computing time

Another resource to estimate is the total NISQ algorithm computing time, including its classical portion. After all, we’re looking for some computing speedup, but with reasonable computing times related to our patience. Its scaling must be carefully estimated in the quantum advantage regime due to various costs: the

number of Pauli strings, the sought precision and the exponential cost of quantum error mitigation, as shown in Figure 63. Whatever the use cases and speedup, NISQ computing times should be reasonable. We will see that it is not necessarily the case in a quantum advantage regime, when it fares better than classical computing. Most NISQ variational algorithms have a computing time with a lot of variables, equal to $N_i * I_t$ with $I_t = (C_t + S * Q_t)$ with:

N_i = **number of iterations** of the variational algorithm to converge on an acceptable value. It is case dependent and depends on the way the variational algorithm converges to the expected solution. Various optimization techniques are proposed here to reduce the number of iterations [964, 965].

I_t = **iteration time** to classically prepare an ansatz (C_t) and to run it on the quantum computer ($S * Q_t$),

representing one iteration, Q_t being the time to run a single shot. C_t also contains the time it takes to handle the classical post-processing of the data coming from the quantum computing shots to generate the expectation value of the Hamiltonian observable from the ansatz. It is highly dependent on the number of shots described below. The ansatz preparation usually converts the problem specific ansatz to a hardware efficient ansatz with these series of rotation and entanglement gates. It minimizes the circuit depth and maximizes the circuit accuracy.

$S = \text{number of quantum circuit shots}$ required to estimate the observables used by the cost function to a target statistical precision. For a single bounded Pauli observable, the elementary Monte Carlo scaling is $O(1/\epsilon^2)$. The total VQE measurement cost multiplies this by the number of measured Hamiltonian terms or groups, parameter-gradient evaluations and optimizer iterations. In electronic-structure Hamiltonians the number of Pauli terms can grow polynomially, often quoted as high as $O(n^4)$ before exploiting structure, but this is not a universal VQE law [966].

The often-quoted $O(n^4)$ scaling refers to the number of fermionic two-body Hamiltonian terms before problem-specific reductions and measurement grouping, not to a fundamental number of VQE shots. After fermion-to-qubit mapping these terms become Pauli strings, whose expectation values are estimated after appropriate basis rotations. The resulting task resembles partial observable tomography rather than full state tomography [966]. For example, determining the ground state of a molecule such as benzene with its 12 atoms (C_6H_6), would require 72 qubits and running 330,816 Pauli strings in a VQE algorithm [967, 968], although this could be optimized to use only 8 qubits [969]. Some Pauli strings can however be regrouped with some preprocessing, reducing the number of Pauli strings to $O(n^2)$ [970] or even $O(n)$, but at the cost of longer circuits that may be prohibitive in NISQ regime.

For QAOA, there is no generic $O(2L)$ law for the number of Pauli strings, with L the number of layers. For standard MaxCut, the cost Hamiltonian contains one $Z_i Z_j$ term per graph edge, while circuit depth and gate count scale with the number of QAOA layers p , graph degree, connectivity and compilation. Measurement grouping and sampling cost must therefore be derived from the specific problem Hamiltonian. In 2025, a French academic team found out that QAOA has trouble finding the global optimal solution for linear functions, requiring exponential time if the number of layers is fixed [971]. Another one found that QAOA might not show an advantage over classical algorithms to solve random MaxXOR problems [972].

A 2026 statistical analysis from Bar-Ilan University derives sufficient conditions on the number of shots per cost operator evaluation for QAOA on MaxCut, both to estimate the expected cost within a relative error δ at confidence $1 - \epsilon$, and to guarantee that stochastic gradi-

ent descent converges to a target relative suboptimality, with an explicit bound on the number of SGD iterations [973]. Counter-intuitively, for the graph classes they characterize, the total shot budget needed to reach a fixed relative performance target decreases as the instance grows. This is a consequence of cost function concentration, the same phenomenon that makes parameter optimization redundant and that underpins barren plateaus, rather than a genuine efficiency gain. The analysis is numerical and involves no hardware run.

With VQE, a bounded observable estimated by independent sampling has a standard error $\epsilon = \sigma/\sqrt{S}$, hence $S = O(\sigma^2/\epsilon^2)$. For a Pauli observable with outcomes ± 1 , $\sigma \leq 1$. For a Bernoulli 0/1 measurement, $\sigma \leq 1/2$, not 1. The statistical precision of each expectation value must then be propagated through the Hamiltonian coefficients and the optimizer. Chemical accuracy is conventionally about 1 kcal/mol, or roughly 1.6 mHa and 4 kJ/mol, so measurement cost can become extremely large even before hardware noise is included [966, 974].

Then, some classical postprocessing generates the ansatz objective function result. Some other tactics can optimize the number of required shots [975]. The target error rate ϵ can be very low for VQE algorithms used in quantum chemical simulations, increasing as a result the number of required shots to astronomical levels.

In 2015, it was estimated that finding the ground energy state of ferredoxin (Fe_2S_2) with 112 spin-orbitals with VQE would require 10^{19} circuit shots and 10^{26} gate operations [976].

Various optimizations reduce measurement grouping, shot allocation or gradient cost, and their benefit is problem dependent [977, 978]. The measurement bottleneck can become prohibitive below 40 qubits for demanding chemistry precision, while structured problems may remain tractable above 40 qubits. The relevant scaling variable is the complete number of circuit evaluations needed to reach a fixed solution accuracy.

Quantum error mitigation adds further circuit executions and classical post-processing. The overhead is method and noise-model dependent. General lower bounds show exponential sampling overhead with circuit depth for local depolarizing noise, while stronger worst-case results show superpolynomial growth with width and depth for broad mitigation classes [914, 915].

For chemistry, full-stack VQE runtime estimates can become extremely large because the circuit must be repeated for many Hamiltonian terms, shots and optimizer iterations [966]. Faster gate times help but do not by themselves remove this multiplicative measurement cost. Trapped-ion entangling gates are typically much slower than superconducting two-qubit gates, often by two to three orders of magnitude depending on the platform, although they can offer higher fidelity and connectivity. The relevant comparison is therefore total time to solution at matched accuracy, not gate speed alone.

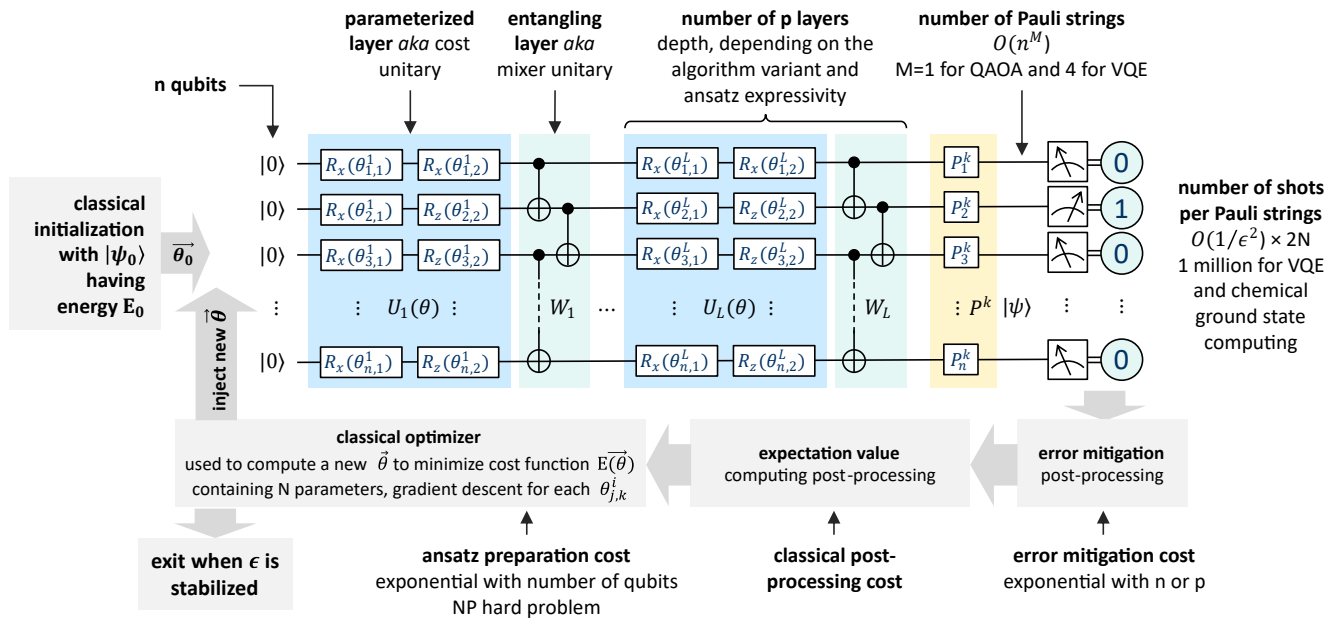


Figure 63: Chart describing how variational quantum algorithms (VQAs) operate and the main terms in their end-to-end resource cost. The grey blocks are classical. A parameterized quantum circuit prepares a trial state, the QPU estimates one or more observables, and a classical optimizer updates the parameter vector $\vec{\theta}$. The total cost is governed by the number of qubits, compiled circuit depth and two-qubit gate count, number of Hamiltonian terms or measurement groups, shots per expectation value, gradient evaluations, optimizer iterations, error-mitigation overhead and classical preprocessing and post-processing. None of these terms has a universal scaling law across VQE, QAOA and QML. In particular, the classical training problem is NP-hard in the worst case rather than “usually NP-complete”, circuit depth is not generically at least the qubit count, and gradient estimation need not simply multiply the full shot budget by every parameter because analytic gradients, simultaneous perturbation, parameter sharing, adaptive ansatzes and other methods change that cost [930, 966]. The central scalability issue is the product of all these factors at a fixed target solution quality. Source: © Olivier Ezratty, 2023-September 2026 and Jules Tilly et al. in [979].

A practical full-stack evaluation of these costs is essential when discussing NISQ quantum advantage. Many published VQA studies remain in the small-instance regime, frequently below 30 qubits, where neither classical state-space growth nor the difficult trainability regime is fully exposed. Parallel execution can reduce wall-clock time by distributing independent shots across QPUs or across disjoint regions of one processor [980], but it does not reduce the total sampling work and therefore does not change the asymptotic sample complexity.

1.4.4 Variational Quantum Eigensolvers

One of the most prominent NISQ algorithm classes is the VQE (Variational Quantum Eigensolver), introduced experimentally by Alberto Peruzzo et al. in 2014 [162]. Its main use is to estimate low-energy states of many-body Hamiltonians [981, 982]. VQE belongs to the broader class of hybrid Variational Quantum Algorithms (VQAs).

VQE is widely studied for electronic-structure and other many-body problems. The classical computer chooses and updates the parameters of an ansatz, while the QPU prepares the corresponding trial state and measures the Hamiltonian expectation value. The optimizer then uses these measurements to propose the next parameter vector [162].

The quantum side of the algorithm is used to compute a cost function outputting a real number that we seek to minimize by varying the parameters of the ansatz in the classical part thanks to a great variety of VQE optimizer [832, 983, 984].

VQE can also help compute the excited states of molecules [985, 986] and be used in quantum machine learning [987].

Most experimental VQE demonstrations still use small active spaces and qubit counts, despite the availability of much larger physical QPUs. The limiting factors are not simply the number of physical qubits. They include the compiled entangling-gate count, measurement precision, optimizer stability and the growth of the ansatz needed for correlated systems. Quantum Volume is useful as a generic hardware benchmark but is not a direct predictor of VQE time to solution or chemical accuracy.

These experiments are useful to test algorithms whereabouts before QPUs can scale and accommodate a larger number of qubits [988]. They most often deal with condensed matter physics, nuclear physics, high-energy particles physics, vibrational and vibronic spectroscopy, photochemical reaction property predictions, to name a few, as described in the excellent Tilly et al. VQE review paper [979].

In the chemical-simulation realm, hardware VQE experiments are usually limited to small molecules or aggressively reduced active spaces, such as LiH, BeH₂ or H₂O [989, 990]. Scaling to chemically interesting correlated systems quickly increases the number of qubits, entangling gates and measurements. Dalton et al. estimate gate-error probabilities of roughly 10⁻⁶ to 10⁻⁴ for representative VQE chemistry calculations at chemical accuracy, with the tolerated error decreasing as the number of noisy gates grows [991]. A 2025 study likewise finds two-qubit fidelities around 99.99% necessary in its analyzed molecular-energy setting [992]. These are model-dependent resource estimates, but they consistently put useful uncorrected VQE chemistry beyond present NISQ hardware.

VQE has not yet demonstrated an advantage on the chemically demanding problems that motivate FTQC quantum chemistry. In mature FTQC, QPE and related coherent Hamiltonian-simulation methods are leading candidates for high-precision energy estimation because they avoid a classical variational outer loop and can improve the dependence on target precision. QPE is however not automatically scalable. Its success probability depends on the overlap between the prepared input state and the target eigenstate, and recent analysis finds that this overlap can decrease exponentially with system size in relevant regimes [922].

VQE was designed as a NISQ-friendly alternative to deeper coherent eigensolvers, but this does not guarantee scalability. Error analyses place stringent fidelity requirements on useful chemistry workloads [993, 994]. Fault-tolerant VQE is technically possible. Sayginel et al. compile continuously parameterized rotations into Clifford+*T* circuits and demonstrate convergence in statevector emulations up to 16 qubits [921]. This is evidence of feasibility, not large-scale advantage. Fault tolerance removes physical-noise accumulation but leaves measurement cost, optimizer complexity and trainability, while adding non-Clifford synthesis overhead. These factors are why QPE-like methods become increasingly competitive in the FTQC regime.

Since 2024, some progress came with the usage of various quantum error mitigation techniques at the scale of 100-150 qubits, particularly with IBM’s Heron 133 and 156 qubit QPUs.

A 2026 study by Thibaud Louvet, Thomas Ayrat and Xavier Waintal derives stringent accuracy criteria for VQE chemistry and concludes that useful calculations require fault-tolerant levels of performance in the regimes analyzed [922]. The paper also highlights a distinct limitation of QPE through the possible exponential suppression of input-state overlap. Related noisy-hardware limitations have been reported independently [995].

A 2026 study from Fraunhofer IPA and the University of Stuttgart pins down the scaling question with a purely classical statevector benchmark, MolVQE-21, covering 21 molecular systems with active spaces from 4 to 10

orbitals and including transition metal complexes such as ferrocene, FeO and FeO₄²⁻ [996]. Running CEO-ADAPT-VQE with the TETRIS extension, they find that the Rényi entropy computed from a purely classical CASSCF calculation predicts the number of adaptive iterations needed to reach the chemical accuracy of 1.6 mHa with a coefficient of determination up to $R^2 = 0.99$, and that this iteration count, and therefore the circuit depth, grows exponentially with system size on hydrogen chains ($R^2 = 0.98$). Extrapolating to Cr₂, [2Fe-2S] and [Cu₂O₂(NH₃)₆]²⁺, precisely the molecules that motivate running quantum chemistry on a QPU, they estimate hundreds to thousands of ADAPT iterations, corresponding to circuit depths far beyond anything current hardware can execute. Two secondary findings matter as much. First, the qubit count alone is not a predictor of cost, since ferrocene with seven orbitals converges in about 7 iterations while hexatriene with six orbitals needs roughly 12 iterations. Second, the usual H₄, LiH and BeH₂ test cases have low Rényi entropy and converge in a handful of iterations, so the entire published VQE benchmarking record says close to nothing about the hard cases. A 2026 survey of VQE developments and benchmarks reaches comparable conclusions on the gap between demonstrations and useful problems [997].

Various VQE optimization techniques are proposed which deal with reducing the number of qubits, improving ansatz encoding efficiency, reducing the number of Pauli strings or shots per Pauli string, and involving quantum error mitigation:

- The ADAPT-VQE derivative which uses fewer qubits and its variations thanks to building the ansatz iteratively [833, 998–1002]. The ADAPT-VQE-SCF variation proposed by Algorithmiq and Trinity College to optimize molecular simulations is using a “self-consistent field approach” within the Adaptive Derivative-Assembled Problem-Tailored Ansatz Variational Quantum Eigensolver (ADAPT-VQE) framework. They expected these techniques to yield useful quantum advantages in 2023. ADAPT-VQE can also be used to simulate atom nuclear shell models [1003]. Other variations of ADAPT-VQE make the ansatz circuit shallower [1004] or significantly reduce the number of measurement Paul strings and thus the number of circuit shots [1001, 1005]. Double Unitary Coupled Cluster (DUCC) improves ADAPT-VQE by providing more accurate Hamiltonians with fewer qubits in the ground state preparation and in the classical optimizer phases [1006]. And yet another variant of ADAPT-VQE reduces the number of iterations to converge toward the ground state [1007]. 2026 variants work on the operator selection step, which is where the measurement cost concentrates: an exact Hamiltonian transformation makes energy-based operator selection resource-efficient for fermionic pools [1008], counterdiabatic terms are added to the pool for molecular simulation [1009], and continuous-variable ADAPT-VQE extends the

framework to bosonic lattice models [1010]. ADAPT-VQE has also been applied to the ground state preparation of random all-to-all Hamiltonians, a harder connectivity regime than the usual molecular test cases [1011]. All of these remain classical emulations.

- Improve the efficiency of the ansatz, or its expressivity, using (classical) tensor networks for their preparation [932]. Eviden’s team use differentiable 2D tensor networks to pre-optimize the parameterized circuit preparing the ground state of the transverse field Ising model, showing that this warm start gives access to gradient zones that do not shrink exponentially with system size, so it mitigates barren plateaus [1012]. They also evaluate the classical cost of computing energies at these warm starts and identify the regime where quantum hardware would scale better than the tensor network itself, which is the right way to frame the question and is almost never done.
- Warm starting which makes the ansatz preparation more efficient using known or efficiently generated results instead of starting the variational optimization from scratch with parameters set to 0 for example [1013, 1014]. A 2026 proposal exploits adiabaticity itself to generate the warm start, preparing an approximate ground state along an adiabatic path before handing it to the variational optimizer [1015].
- Change the cost function rather than the ansatz. Instead of minimizing the expectation value of the Hamiltonian, the CVaR (Conditional Value at Risk) objective minimizes the mean of the worst tail of the measured energy distribution, which is better matched to optimization problems where only the best bitstrings matter and which improves convergence at equal circuit depth [1016].
- Optimize the block-encoding of the ansatz [1017] and use only Clifford gates, which may reduce its expressivity [1018].
- Use a pre-trained neural network to adjust the ansatz at each iteration, enabling faster convergence and reducing the cost of this preparation [1019]. It can even use generative models [1020], including transformer neural networks to tune the quantum circuit ansatz [461].
- Using transfer learning to improve the efficiency of the ansatz initialization for VQE used in combinatorial optimizations [1021].
- Encoding the ansatz as parametrized Dicke states to be used in a multiclass portfolio optimization based on VQE [1022]. Dicke states are symmetric, entangled quantum states defined as equal superpositions of all basis states with a fixed number of qubits in the $|1\rangle$ state. This encoding technique initializes the qubits in a superposition of only feasible states, satisfying the problem’s constraints, by ensuring a fixed number of assets are selected from each class.

It significantly reduces the search space of the VQE algorithm.

- Reducing the number of Pauli strings [1023] or shots using ancillary qubits [1024].
- Reducing the number of circuit shots per Pauli string to compute the energy gradient for each ansatz parameter [1025] (N in Figure 63).
- Embed measurement based elements in the VQE ansatz circuit to form a hybrid VQE, improving the ansatz tuning [1026, 1027].
- Implement VQE-specific error mitigation techniques using machine learning [1028].

VQE is, however, not the only variational algorithm for computational chemistry. QSCI (quantum-selected configuration interaction) can find low-lying eigenvalues and eigenstates of a many-electron Hamiltonian and is noise resilient. It can be used to compute excited states of many-body systems [1029]. But it seems that QSCI doesn’t scale well [1030].

VQE can also be used to solve constrained combinatorial optimizations problems [1031].

Several 2026 papers try to make VQE convergence provable rather than empirical, with robust convergence guarantees under a bounded noise model [1032], and to make the classical side scale, with JIT compilation and multi-GPU parallelization of the optimizer loop [1033]. This last point is worth mentioning, since the classical optimizer and post-processing dominate the wall-clock time of most published VQE runs, and speeding them up on GPUs simultaneously raises the bar for the QPU. On the application side, a benchmark of active space selection for drug discovery quantifies how much the chemistry result depends on a choice made entirely classically, upstream of the QPU [1034]. Beyond superconducting qubits, a Korean review surveys the implementation of VQE on photonic platforms, which run at room temperature, have low decoherence and offer multiple degrees of freedom per photon [1035]. And on the classical baseline front, large tensor network simulations of the Lipkin-Meshkov-Glick model provide a reference against which quantum simulations of that model can be checked, and these classical simulations generally outperform the quantum ones [1036]. Variational product formulas also offer an alternative to Trotterization for Hamiltonian simulation, with shallower circuits at the cost of a variational loop [1037], and the generative quantum eigensolver has been applied to Auger spectroscopy [1038].

An alternative to VQE is the **Projective Quantum Eigensolver** (PQE) for optimizing unitary coupled-cluster (UCC) simulations formulating ground state search as a root finding problem. PQE optimizes a trial state using residuals, aka projections of the Schrödinger equation, rather than energy gradients as in VQE. This is not a variational algorithm but it runs with so-called

micro-iterations. It could potentially work on NISQ QPUs [1039, 1040]. It could also be used to prepare a QPE ground state. Likewise, the NoVa-ADAPT proposal is a non-variational quantum algorithm which selects the operators and estimates the coefficients for updating the state based on the energy gradient information, without using a classical optimization loop. It seems more robust against gate errors [1041].

Two 2026 proposals extend this family of non-variational or partially variational eigensolvers. CANOE, a classically assisted non-orthogonal eigensolver, builds the eigenstate from a non-orthogonal basis whose overlaps are computed classically, reducing the burden on the quantum part [1042]. The Projector Quantum Variational Ansatz from CEA List constructs the ansatz around projectors rather than exponentials of excitation operators [1043]. Both target the same weakness, the number of iterations and measurements of the outer loop, and both are validated in emulation only.

1.4.5 Quantum Approximate Optimization Algorithms and other optimizations

The second key NISQ hybrid algorithm class deals with solving Combinatorial Optimization Problems (COP). In the classical realm, COPs are solved with exact and approximate methods, with heuristic and meta-heuristic sub methods as shown in Figure 55. The Grover algorithm that we have already seen is an exact method while QAOA is an approximate method with metaheuristics characteristics.

QAOA is mostly used to solve unconstrained optimization problems formulated as QUBO (quadratic unconstrained binary optimization, where the objective function is a quadratic polynomial of binary variables, and there are no constraints on these variables) and HUBO problems (high-order unconstrained binary problems, with an objective function including terms with products of more than two binary variables), given HUBO problems can be reduced to QUBO problems.

The Quantum Approximate Optimization Algorithm (QAOA), created by Edward Farhi, Jeffrey Goldstone and Sam Gutmann in 2014 is a combinatorial optimization algorithm implementing metaheuristics to find approximate solutions to solve many problems. Some of the related papers mention “exponential speedups” but are to be taken with caution. NISQ QPUs have hardware scaling limitations and when implementing QAOA on FTQC platform, error correction space and time overhead has to be accounted for.

- TSP (traveling salesperson problems) [1044, 1045].
- Graph and slice management problems (Max-Cut) [1046–1051], graph coloring problems [1052], although it seems to show no quantum advantage [1053].

- Various tasks and jobs scheduling problems like the Binary Paint Shop Problem (BPSP) [1054], where a 2026 result turns the absence of advantage into a classical gain: since logarithmic-depth QAOA is instance independent, the fact that the recursive star greedy heuristic beats QAOA at depth $p = 7$ with an expected paint swap ratio of 0.361 directly yields an improved classical upper bound [1055].
- Maximum Independent Set problems [1056–1059] which can help in complex task scheduling tasks and telecommunication networks optimization.
- Graph partitioning and cut problems, with encodings that avoid penalty terms altogether, either through a ring-structured XY mixer confining the evolution to feasible cut configurations [1060] or through a HUBO encoding using $\lceil \log_2 k \rceil$ qubits per k -valued vertex variable, the information-theoretic minimum, with a lexicographic penalty system that minimizes the partition count without indicator variables [1061]. The latter covers minimum graph coloring, minimum k -cut and community detection, and addresses the optimization versions rather than just the decision versions.
- 3SAT Boolean satisfiability problems which help verifying logical consistency in digital circuits [1062].
- Max-Sat Boolean problems [1063] which can for example help finding optimal plans under soft constraints.
- Closest vector problem (CVP) which has applications in cryptanalysis against lattice-based post-quantum cryptography [1064]. Hopefully, it probably doesn’t scale well.
- Approximate the ground-state of quantum Hamiltonians [1065].

QAOA also has a broader formulation called the Quantum Alternating Operator Ansatz, which generalizes the mixer and phase-separation operators to encode problem constraints [1066]. The scaling record is mixed. Many practically motivated QAOA studies remain at small size or require hundreds of qubits for a projected crossover, while large hardware runs above 100 qubits have not established end-to-end advantage. At the same time, there are problem-specific counterexamples to blanket pessimism, such as noiseless LABS simulations up to 40 qubits that show better empirical scaling than branch-and-bound for the tested regime [1067].

Because QAOA is normally used as a heuristic or approximation algorithm, benchmarking should compare matched solution quality and time to solution against the strongest relevant classical methods. Exact ILP or branch-and-bound solvers can still be valid baselines when the quantum workflow is also judged by its probability of finding the optimum. Approximation algorithms, semidefinite relaxations and heuristics are also essential baselines.

According to Anton Simen Albino et al., “*thousands of qubits will be needed before QAOA and its variants can be used to solve these problems, due to the linear relationship between the dimensionality of the problem and the number of qubits. However, the qubits used will not necessarily be error-corrected due to the characteristics of the heuristic itself, which requires low-depth circuits and few measurements of the final state*”, in a paper dealing with solving partial derivative equations (PDEs) in fluid mechanics [314]. Johannes Weidenfeller et al. provides a lot of clues on QAOA running on NISQ systems, highlighting some obstacles to overcome to “*improve to make QAOA competitive, such as gate fidelity, gate speed, and the large number of shots needed*” [1068]. Their paper covers transpiler optimization techniques and QAOA on IBM heavy-hex connectivity. It also derives a problem- and estimator-specific shot estimate with an $O(n^2/\epsilon)$ dependence under its assumptions [1068]. This is not a universal QAOA sample-complexity law. Shot requirements depend on the cost Hamiltonian, estimator, target confidence, parameter-setting strategy and optimization method.

In a 2019 MaxCut resource comparison, G. G. Guerreschi and A. Y. Matsuura estimated that a crossover would require several hundred qubits, with about 900 qubits in one analyzed setting [953]. The classical baseline was a single Intel Xeon Phi, so this number is not a current universal crossover threshold. Likewise, converting 900 qubits times eight layers directly into a 99.986% gate-fidelity requirement is only the simplified independent-error heuristic discussed above. A compiled QAOA circuit has a problem-dependent number of two-qubit gates and routing operations.

Meanwhile, most QAOA experiments were initially done with only a few qubits although since 2022, some are using IBM QPUs with over a 100 physical qubits [1069, 1070]. Some high-quality QAOA optimization targets can require deep circuits or very low effective error rates [1071]. Specific resource studies therefore place useful instances in an early FTQC regime [1072].

Guillermo González-García et al. landed on a similar conclusion in 2022 [1073]: “*We find that, even with a small noise rate, the quality of the obtained optima implies that a single-qubit error rate of $1/(nD)$ (where n is the number of qubits and D is the circuit depth) is needed for the possibility of a quantum advantage [...]. We estimate that this translates to an error rate lower than 10^{-6} using the QAOA for classical optimization problems with two-dimensional circuits*”. This analysis is strong evidence that uncorrected deep QAOA becomes difficult at large size, but it does not prove that every QAOA advantage requires FTQC. High-connectivity hardware, problem-specific shallow circuits, fixed or transferred parameters and early fault-tolerant execution may shift the crossover. No generic quantum speedup is currently established [1074].

A Quantinuum and JPMorganChase study published in Science Advances in 2024 provides an important addition

to QAOA scaling discussions. On the LABS problem, noiseless simulations up to 40 qubits show better empirical scaling than state-of-the-art exact branch-and-bound solvers, and QAOA combined with quantum minimum finding gives the best empirical scaling among the tested methods [1067]. This remains evidence for an idealized algorithmic scaling advantage, not a demonstrated hardware advantage.

The 2023 QAOA review paper by Kostas Blekos et al. is thorough and more nuanced. It provides guidelines on the way to select the right approach, ansatz variant, parameter optimizations and how to best embed error correction mitigation techniques [1075]. Various tricks are indeed proposed to reduce the circuit depth of QAOA ansatzes like ADAPT-QAOA or FALQON [1076]. They are slightly moving the needle in the direction of a potential quantum advantage [1075, 1077–1081], including some which are hardware dependent [1068, 1082, 1083]. Variations of this algorithm can potentially be used to compute excited states in quantum chemistry [1084]. These techniques, however, can have a significant classical overhead preventing their scaling to a large number of qubits [1085]. Other methods deal with improving global optimizations of the ansatz [1086]. There is also a more efficient QAOA variation that is based on Grover algorithm and uses twice as many qubits [1087].

Two 2026 results question the ansatz itself rather than its parameters. A LMU Munich team asked whether the entangling gates routinely inserted into hardware-efficient ansätze actually help on MaxCut, whose Hamiltonian is diagonal and whose ground states are classical product states requiring no entanglement at all [1088]. Tracking the Meyer-Wallach entanglement measure through training, they find that whenever the optimizer has indirect control over entanglement through the circuit parameters, it drives entanglement down, and that a fully separable ansatz outperforms every entangled hardware-efficient configuration tested, in a monotonic relationship. QAOA, whose entanglement is structurally derived from the problem Hamiltonian, maintains high entanglement and stays competitive, showing that the structure of entanglement and not its quantity determines its usefulness. The result is uncomfortable since a large share of published hardware-efficient VQA results are spending scarce two-qubit gate fidelity to produce entanglement that actively hurts. Second, a Volkswagen, RWTH Aachen and Jülich study proves an intrinsic feasibility bottleneck for generic transverse-field-mixer QAOA on permutation-constrained problems, where valid solutions occupy a low-dimensional manifold inside the Boolean hypercube [1089]. Even after angle optimization, circuits whose depth grows at most linearly with n cannot lift the probability mass on the feasible manifold much above the uniform baseline, itself suppressed by the size of the full Hilbert space. Embedding the constraint directly in a block-local XY Hamiltonian restores an exponential improvement in feasible mass, which confirms that penalty-based encod-

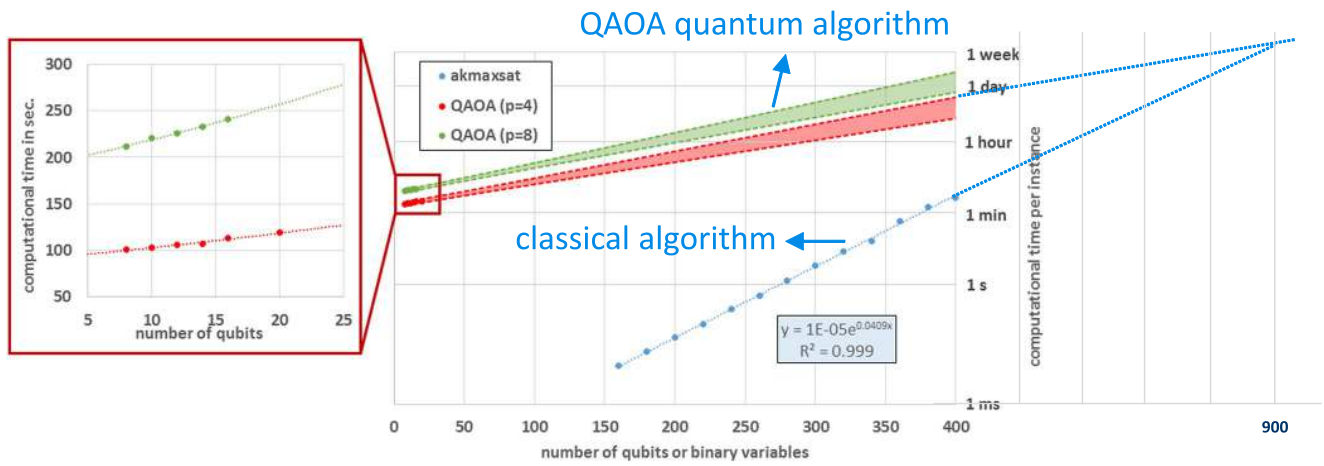


Figure 64: One 2019 QAOA resource study estimated a crossover near 900 qubits for its MaxCut setting and its classical hardware baseline [953]. This is a scenario-specific crossover, not a generic QAOA threshold. The required physical fidelity cannot be obtained reliably from $1/(np)$ because the relevant error budget depends on the compiled number of two-qubit gates, routing, noise correlations and acceptable output degradation. The figure should therefore be read as an illustration of the breadth-depth-noise tension rather than a precise hardware requirement. Source: [953] plus additions from Olivier Ezratty. Updated in September 2026.

ings are structurally the wrong route for constrained problems, and not merely a tuning nuisance.

A 2026 JPMorganChase proposal, RWS-QAOA, initializes the qubits by minimizing the expected energy with a regularizer that penalizes states too close to a single bitstring, which avoids the known failure mode where a warm start sitting on a good classical solution leaves QAOA stuck [1090]. A companion protocol fixes instance-independent angles, so the circuit becomes constant-depth and effectively non-variational, with only the classical warm start remaining instance-dependent. Benchmarked on Max-Cut over random regular graphs across three complexity classes, the authors conclude that constant-depth circuits combined with a classical warm start have a potential to surpass classical solvers, but on future quantum computers. The variational loop that defines QAOA and carries most of its cost is what recent proposals are removing.

A 2026 IBM Research approach exploits a property specific to optimization, stating that only high-quality bitstrings matter and not a faithful reproduction of the output distribution. An ancilla-free linear combination of unitaries based on Fourier decompositions then replaces highly connected qubit interactions with single-qubit gate layers, at the cost of a polynomial sampling overhead [1091]. This generates hardware-friendly decompositions of penalties while keeping performance guarantees against the fully coherent implementation, and it establishes a formal connection between Fourier-based quantum penalties and Lagrangian relaxation. It was validated on 12-qubit statevector simulations and on 106 superconducting qubits. The trade-off is about circuit complexity against shot count.

One looming question is whether QAOA (on gate-based QPUs) is more efficient than solving QUBO problems on quantum annealers. A 2024 paper from Q-CTRL

answered “yes” with solving a Max-Cut problem with 120 qubits and finding the ground state of a spin-glass model with a HOBQ algorithm with 127 qubits. In both cases, it fared better than a D-Wave annealer. In the Max-Cut problem, there was, however, no demonstrated quantum advantage over the CPLEX classical baseline, which provides the ground truth [1092]. Another 2024 paper, from Los Alamos National Laboratory researchers said “no” when comparing QAOA run on 127 qubits and quantum annealing to solve some Ising models matching both systems qubits topology, with 3,000 gates executed on an IBM QPU. Here, quantum annealing outperformed QAOA. Solutions qualities were compared with a classical CPLEX solver [1093].

We are still far from obtaining a potential quantum advantage in a NISQ regime for solving combinatorial optimization problems [1094]. Most of the recent papers deal with tests done with fewer than 20 qubits which can run efficiently on your own laptop. For example, a Max-Cut problem solving algorithm seems competitive with the Gurobi classical solver but not others such as the Burer-Monteiro approach for the maximum cut problem. It was only tested with 7 qubits, with results extrapolated for a larger number of qubits [1095].

A VQA with linear-depth circuits solving a Max-Cut problem was emulated classically with only 20 qubits with no estimates for larger settings [1096].

A so-called BENQO algorithm is said to be superior to QAOA to solve TSP problems but was tested only with up to 16 qubits, with no indication on computing times [1097]. A quantum graph optimization algorithm was tested on 4 to 12 qubits [1098]. Another MaxCut solving algorithm avoiding the exponential proliferation of local minima with increasing circuit depth was tested with only 8 qubits [1099]. A variational solver for solving MaxCut combinatorial optimizations over $m = O(n^k)$

binary variables using only n qubits, with tunable $k > 1$ is only tested with 18 trapped ions qubits [1100]. A 76-page single-author proposal using seed-conditioned circuits to generate interference patterns as finite-shot proposal distributions for QUBO and Ising sampling is benchmarked on instances of 18 to 29 bits with 100 shots per circuit, and against a matched classical control preserving the same search loop rather than against a real solver [1101]. On the industrial side, a rider-to-order assignment problem for food delivery, compared across classical, quantum-inspired and gate-based solvers, reported no quantum advantage [1102].

At the other end of the spectrum, several 2026 teams ran QAOA at what IBM calls the “utility scale”, above 100 qubits, and every one of them states explicitly that no quantum advantage was obtained. A University of Victoria and IBM Quantum team ran Minimum Vertex Cover, MIS and Maximum Clique instances, all reduced to unconstrained form, on a 156-qubit Heron r2 IBM processor, reaching graphs of up to 128 vertices with circuits using up to 128 qubits and 13,555 two-qubit gates [1103], not far from the record 13,800 gates published a month later by Q-CTRL in [913]. Their pipeline put classical preprocessing before the QPU to shrink the instance and classical post-processing after it to map noisy bitstrings back onto feasible solutions, which means the quantum part solves a reduced problem and never enforces the constraints itself. A Rensselaer and Marist team mapped honeypot traffic separation for DDoS mitigation onto weighted MaxCut, with benchmark graphs of 16, 32, 66 and 110 event nodes, and executed the largest 110-node, 181-edge instance on three IBM quantum processors from three generations, Eagle, Heron and Nighthawk, the latter compiling to the fewest two-qubit gates thanks to its 1-to-4 qubit connectivity [1104]. The authors note that simple classical heuristics solve all of their labeled benchmark graphs, and present the work as a hardware feasibility and architecture benchmark rather than an advantage claim. A collaboration led by IBM and involving NASA Ames, Los Alamos, Jülich and HSBC benchmarked angle setting strategies at utility scale, comparing matrix product state and Pauli propagation approximations against the transfer of angles trained on small instances, then validating on hardware [1105]. The best angle-setting methods are themselves classical simulations of the circuit, and the fact that angles transfer from small instances implies the instance-specific quantum work is small.

The noise cost of these runs has now been quantified. Running $p = 1$ QAOA on `ibm_fez`, a 156-qubit Heron r2, with up to 57,344 shots per grid point on three constrained QUBO portfolio instances, a landscape span compression metric shows that hardware noise flattens the energy landscape by 24 to 30% without displacing the global minimum, which is good news for transferring classically optimized angles onto hardware [1106]. The rest is less encouraging. IBM’s calibration-based noise model reproduces the landscape structure with a Pearson correlation of 0.959 but accounts for only about

42% of the observed approximation-ratio degradation, leaving crosstalk and coherent errors as the dominant unmodeled contributors. Zero-noise extrapolation yields inconsistent changes in the reported approximation ratio, of +7%, +9% and −4% across the three instances while inflating uncertainty by a factor of 3 to 5, and a residual noise cost of about 0.03 approximation-ratio units shows up in all instances. A landscape that flattens is a barren plateau in the making, which ties hardware noise directly to the trainability discussion further below.

In other recent papers, algorithmic improvements are presented but, usually, with no or minimal resource estimates to reach a quantum speedup advantage which is, most of the time, only quadratic [1107]:

- A recursive algorithm for solving combinatorial optimization problems like a Max-Cut instance, that is compared with QAOA, classical linear programming (LP) and semi-definite programming (SDP) [1108].
- A proposal to convert a pathfinding problem like a TSP into a QUBO formulation, with no indication of speedup, qubit counts and fidelity constraints [1109].
- Solving TSP with problem encoding methods using QUBO, HOBQ, QAOA and VQE but with no reasoning at scale [1110, 1111].
- A noniterative more resource-efficient landscape optimization procedure [1112].
- Improvements in the QAOA ansatz initialization enabling the support of a larger number of qubits to solve MaxCut problems [1113].
- Avoiding barren plateaus with parameter transfer from one model to the other [1114] and other machine learning-based denoising methods [1115]. Other ansatz like the Hamiltonian variational ansatz can also circumvent the barren plateau effect in some circumstances and with spin models [1116].
- Combining technique of warm start which incorporate classical approximate solutions to guide the quantum evolution (with arbitrary R_y rotations) and XY mixer which constrains the search to feasible subspaces aligned with the structure of the problem (with X and Y gates applied to pairs of qubits). It reformulates the combinatorial optimization problem into a Max-Cut problem. This was tested on a 5-cities TSP and 16 qubits! [1117–1119]
- Solving constrained optimization problems by mapping them to unconstrained problems at the cost of extending their graph [1120–1122].
- Using circuit overparametrization, with a number of parameters exceeding a critical threshold, a technique used in classical machine learning, where models with more parameters than training data can generalize well under certain conditions. It is used to solve Max-Cut and Max-2-Sat problems, but with validation

only in a sub-quantum advantage regime under 20 qubits [1123].

- Solve Local Hamiltonian Problems (LHPs) in which one must find the ground state energy of a Hamiltonian composed of terms acting only on a few sites of a many-body quantum state. It can compute 1-dimensional antiferromagnetic Heisenberg spin chains and even higher-order Ising spin glass models, which was tested on IBM 127-qubit QPUs [1124].
- Using a hybrid framework and an adaptive classical Tabu search for classical preprocessing to decompose large-scale quadratic unconstrained binary (QUBO) problems into NISQ-compatible subproblems. In the end, the result seems classically emulable. This was tested on China’s BAQIS Dongling superconducting qubits QPU with 104 qubits in a 169-qubit chip (13*13) [1125].
- Solving the NP-complete one-in-three SAT problem which is a type of 3-SAT problem with more constraints [1126], pretending to obtain some scaling advantage, but not accounting for the space and time cost of error correction for large variational circuits having several dozens of layers.
- Using QAOA to “warm start” a classical optimization task to optimize it with initial quantum-generated samples, for solving Max-Cut and Maximum Independent Set (MIS) problems [1127]. QAOA becomes in that case an accessory to improve the efficiency of a classical solver.
- Partitioning the problem graph into subgraphs small enough for the available QPU, keeping only the subgraph states whose local energies can belong to a global solution, then recombining with a binary encoding and iterating. This solves weighted random 3-regular graphs with 40 variables on about 10 qubits while retaining an approximation ratio around 99.9%, with the reduction improving at larger sizes [1128]. No error channel is modeled and 1,000 shots per circuit are allocated.
- Dropping the variational framework entirely in favor of Markov chain Monte Carlo heuristics, with Quantum-enhanced Simulated Annealing and Quantum-enhanced Parallel Tempering validated on hard Sherrington-Kirkpatrick instances [1129]. In the same vein, quantum-enhanced MCMC combined with warm starting and parallel tempering recovers global optima of MIS instances with up to 117 decision variables on 117 IBM qubits, the authors conceding that scales known to be hard for state-of-the-art classical solvers are still out of reach [1058].
- Using QAOA with an XY mixer on subgraphs only to train neural network surrogates that then supply the MCMC proposal distributions, which accelerates mixing by factors of about 20.3 and 7.6 on the autocorrelation decay rate against classical pair-flip

methods [1130]. Here the quantum step is a one-off training input and is no longer in the loop at all.

- Removing the SWAP gates that dominate the depth of QAOA circuits on nearest-neighbor topologies, through a routing-free compilation framework [1131]. On heavy-hex or linear connectivity, SWAP insertion typically multiplies the two-qubit gate count by a factor of several units, so this is one of the few optimizations acting directly on the dominant error source.
- Enforcing cardinality constraints through Hamming weight operators instead of penalty terms, which keeps the evolution inside the feasible subspace without inflating the operator norm of the cost Hamiltonian [1132].
- Predicting good angles rather than optimizing them, either by forecasting the success probability and minimal depth required for a given instance [1133], or with a graph-conditioned meta-optimizer generating parameters across several problem classes [1134]. These methods remove much of the outer loop cost, which also removes much of what makes the algorithm quantum.
- Explaining rather than improving, with an analysis of why QAOA works on random k-SAT, tracing its efficacy to an adiabatic manifold and deriving a sublinear parameter optimization scheme from it [1135].

Some papers are encouraging like a proposal for a generic QAOA implementation to simulate MIS, BPP, TSP, MaxCut, W-MaxCut, 3-MaxCut, KP, PO, Max-2-SAT, and Max3-SAT problems with up to 42 qubits and 400 layers on a classical supercomputer in an emulator. It also works well with up to 109 qubits, while being resilient to noise, seemingly in a quantum advantage regime [1136]. Another paper deals with two relevant industrial use cases: one in the realm of optimizing charging schedules for electric vehicles, the other with the optimization of truck routes. It runs on gate-based IBM QPUs using QAOA and VQE and on D-Wave annealers. Still, it is not yet in a quantum advantage regime although improving in the case of recent gate-based QPUs [1137].

A 2024 paper proposes a supervised machine learning based method to select the right solver among QAOA, VQE, Grover or quantum annealing [1138].

A 2025 paper from Zoë Holmes et al. shows that adaptive discrete optimizations where the variational approach is about optimizing the circuit structure itself and not just its parameters bring a potential exponential speedup to solve hard optimization problems [1139].

Divide-and-conquer keeps inflating the apparent scale of QAOA. ParaQAOA partitions large Max-Cut instances into subproblems solved in parallel, reporting up to a 1,600× reduction in classical simulated runtime

against other QAOA implementations on 400-vertex instances, which is not a quantum-over-classical or hardware speedup while staying within 2% of best-known solutions, and solving a 16,000-vertex instance in 19 minutes against more than 13.6 days for the best previous approach [1140]. The comparison baseline is other QAOA implementations running on classical parallel hardware, not a best-in-class classical Max-Cut solver such as Burer-Monteiro or Gurobi, so this is a simulation engineering result and not a quantum one, and the paper is honest about running on classical parallel architectures. On the hardware side, a team from RIKEN, the University of Tokyo and OptQC ran the first experimental continuous-variable QAOA on a measurement-based photonic platform using a quad-rail lattice cluster state, with a systematic mapping of arbitrary quadratic cost functions examined in settings involving up to 100 modes, and variational parameters tuned by Bayesian optimization [1141]. Going from depth 1 to depth 2 improved performance, further depth increases did not, whereas idealized simulations assuming infinite shots and gradient-based optimization keep improving with depth. This isolates noise accumulation and classical optimizer limitations as the causes, and it is the same wall that discrete-variable QAOA hits. A genuine platform first, and equally clearly nowhere near a quantum advantage.

Kipu Quantum with IBM Quantum, and several academic partners benchmarked in 2026 a hybrid sequential quantum computing workflow against highly optimized multi-core classical solvers using a single Nvidia GPU to solve a 156-variable HUBO optimization problems. It ran on IBM Heron r3 processors, the method combining a classical simulated annealing warm start with digitized counterdiabatic quantum optimization and tabu search post-processing. The experiment delivered high-quality solutions under a second but there was no quantum advantage. Advanced classical solvers, particularly an enhanced parallel tempering algorithm and a GPU-accelerated solver utilizing 8 NVIDIA A100s, still matched or exceeded the hybrid workflow’s performance and time-to-solution metrics [1142].

Other papers are very theoretical, like this solution to solve combinatorial problems by looking for the ground-state of an Ising model on a quantum annealer or a gate-based quantum device. It applies the method to two constrained NP-hard combinatorial optimization problems: a graph partitioning problem and the quadratic knapsack problem [1143]. Another one brings a theoretical super-polynomial quantum advantage for a formula coloring (FC) solving algorithm, which is not generic [1144]. A 2025 framework from the Chinese Academy of Sciences derives quantum dynamics with rigorous approximation ratio bounds from a time-dependent Lyapunov function, upper bounding the unknown optimum from the current quantum state, but it comes with no qubit count and no comparison against classical approximation algorithms [1145].

1.4.6 Quantum Machine Learning

Many NISQ quantum machine learning proposals use VQAs [1146]. They inherit the same trainability, measurement and noise-scaling problems as other parameterized quantum circuits, while adding a crucial data-access question: the classical cost of loading, querying or preprocessing the training data can erase an apparent quantum speedup [696].

In November 2022, Lucas Slattery et al. reported numerical evidence against an advantage for the quantum fidelity-kernel models they studied on classical data, rather than a no-go theorem for all NISQ QML [1147]. Mohammad Kordzanganeh et al. also benchmarked one shallow hybrid quantum neural-network workflow across several simulators and QPUs and observed rapidly degrading accuracy on some superconducting devices as the tested qubit count increased [1148].

Other advances in QML algorithms are tested on QPUs with a very low number of qubits, like in the work of Diego H. Useche et al. which “*presents a novel classical-quantum density estimation strategy for current noisy quantum computers, which combines quantum algorithms to compute the expectation values of density matrices with a new quantum variational representation of data called quantum adaptive Fourier features (QAFF)*”. It was tested on an IBM Oslo QPU with 7 qubits and the discussion about its scalability seems absent in regards of these systems qubit gate fidelities [1149].

Dequantization results sharpen the QML comparison by requiring classical and quantum algorithms to be compared under explicit data-access models. Jordan Cotler et al. show that classical algorithms with sample-and-query (SQ) access can in some settings match or even outperform learning procedures formulated with quantum-state inputs [1150]. It shows that claimed exponential speedups on classical data can disappear when the classical baseline has a data structure that provides access comparable to the quantum input oracle.

Quantum-algorithm dequantization is the construction of a classical algorithm that reproduces the relevant performance of a quantum algorithm with comparable input-access assumptions, up to polynomial overhead. It is broader than decomposing a circuit into tensor operations. Many QML dequantization results use sample-and-query or length-squared sampling access, the classical analogue of preparing a normalized vector as amplitudes and sampling it in the computational basis. Ewin Tang’s recommendation-system algorithm was the landmark example: under ℓ_2 -sampling access it reduced a previously claimed exponential quantum speedup to a polynomial separation [1151]. More recent work shows robust dequantization for broad QSVT and QML families under approximate length-squared sampling [1152]. This is an important bridge to the quantum-inspired algorithms section: whenever a VQA is made trainable by restricting it to a low-dimensional, low-entanglement or otherwise structured subspace, the

same structure may also make a stronger classical algorithm possible [936].

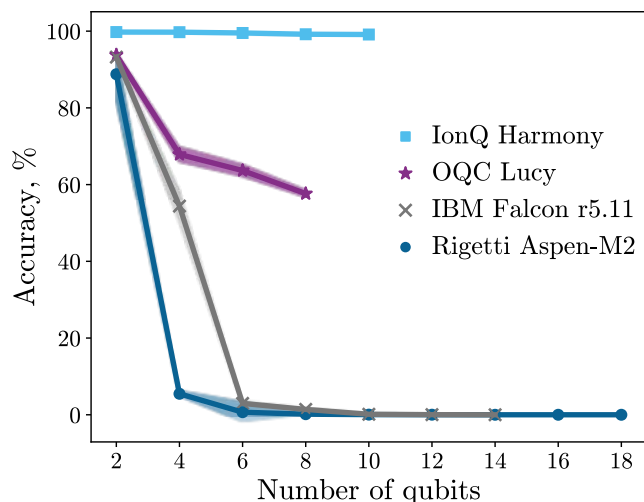


Figure 65: Hardware benchmark of one hybrid quantum neural-network workflow across simulators and several QPUs [1148]. The observed degradation on some superconducting platforms as the tested qubit count increases illustrates the breadth-depth-noise tension for this specific circuit family. Source: [1148]. Added in 2023, caption updated in September 2026.

Pradeep Niroula et al. created a deep learning algorithm enabling the creation of documents summaries [1153]. This hybrid algorithm had a classical part doing a lot of classical data preparation. It analyzed a dataset of 300,000 news articles from CNN and the Daily Mail and precomputed it with a BERT NLP (natural language processing) classical deep learning model that handled sentences extraction and their conversion into vectors. The quantum part managed the text summarizing from respectively 20 to 8 and 14 to 8 sentences, with Quantinuum QPUs H1-1 and H1-2 QPUs using respectively 20 and 14 qubits, and with a 100 qubit gates depth which is excellent. But we are not yet in the quantum advantage regime with this number of qubits which, again, is small enough for conventional classical emulation, with no matched wall-clock comparison reported. The paper does not provide resources requirements estimates for a larger summary set for, say, 100 or 1,000 sentences.

In another work, Robin Lorenz, Bob Coecke et al. implemented some natural language processing algorithm using more than 100 input sentences and only five qubits on an IBM QPU [1154, 1155]. Likewise, Wei Xia et al. presented in March 2023 an improved quantum reservoir computing algorithm that could run on up to 7 qubits with some precision improvement over classical reservoir computing methods for forex (foreign currencies exchange) forecasts. But these 7 qubits run faster on a classical software emulator than on any existing QPU and the paper doesn't mention any qubit fidelities and number requirements to reach some quantum advantage [576].

Late 2022, Ismail Yunus Akhalwaya et al. touted that NISQ systems would soon be able to solve Topological Data Analysis problems (TDA) [42]. TDA is used for extracting complex and shape-related summaries of high-dimensional data [1156, 1157]. NISQ-TDA was presented as the “*first fully implemented end-to-end quantum machine learning algorithm needing only a linear circuit-depth, that is applicable to non-handcrafted high-dimensional classical data, with potential speedup under stringent conditions*”. Practically speaking, TDA can identify clusters in high-dimensional data. It serves to estimate a “Betti number” which measures the connectivity of a topological space.

But we are far from being able to implement this algorithm in a NISQ regime [1158, 1159]. It is a narrow implementation of the TDA class of problems, and it imposes stringent data conditions to generate any computing advantage. On top of this, a NISQ computing advantage would require over 96 qubits with 99.99% two-qubit gate fidelities which are not in the radar yet. Again, with even a fidelity of 99.9%, we would need at least about 9,600 such physical qubits.

Alexander Schmidhuber and Seth Lloyd “*argue that quantum algorithms for TDA run in exponential time for almost all inputs by showing that (under widely believed complexity theoretic conjectures) the central problem of TDA - estimating Betti numbers - is intractable even for quantum computers [...] Our results imply that quantum algorithms for TDA offer only a polynomial advantage*” [1160], which, if implementable in a real NISQ regime would make sense. But given the overhead of FTQC that would be mandated to solve this class of problem, we'd have to look at the constants and other fixed costs to check that a quantum advantage would show up in a reasonable regime. Still, aside from estimating Betti numbers, under some circumstances, TDA variants could bring some theoretical exponential speedup [1161].

Another older paper is more optimistic on TDA resource requirements. It states that a quantum TDA algorithm can have a guaranteed superpolynomial quantum speedup vs classical computing [1162]. It says a quantum advantage would require at least 80 physical qubits but gives no precise indication on the algorithm depth. With the shallowest algorithm possible of 8 gate cycles, we still would need two-qubit gate fidelities in the 99.8% range.

On the other hand, quantum machine learning speedups are not the sole potential quantum advantage attribute but, as Maria Schuld and Nathan Killoran pinpointed in 2022, the comparisons are complicated between classical and quantum machine learning algorithms [466, 1163].

It deals with classifications quality, generalization capability on unseen training data, training data requirements and the like, with few benchmarking references. On top of this, training data ingestion is mostly done by the classical part to prepare the algorithm quantum

ansatz, and it scales linearly with the data size, so with no foreseeable quantum advantage. These doubts were amplified in a quantum AI software engineering review paper published in 2024 [1164].

Like VQE and QAOA, variational QML can exhibit barren plateaus. Noise can induce exponentially vanishing gradients when circuit depth grows with system size [1165], while even shallow circuits can suffer barren plateaus for global cost functions [1166]. Local cost functions and logarithmic-depth structured ansatzes can retain polynomially large gradients under specific assumptions. Barren plateaus are therefore architecture-, cost-function-, initialization- and noise-dependent rather than a simple shallow-versus-deep dichotomy.

1.4.7 Variational Quantum Linear Solver

VQLS (Variational Quantum Linear Solver) is a variational proposal for Quantum Linear Systems Problems (QLSPs) [1167]. It is studied for applications such as CFD [1168], Poisson equations [1169] and machine learning [1170]. The original VQLS work demonstrated hardware solutions on Rigetti processors for systems up to 1024×1024 . The much larger $2^{50} \times 2^{50}$ case was a numerical scaling study, not a QPU experiment [1167]. This distinction is important because variational linear solvers also inherit state-preparation, measurement, condition-number and optimization costs.

1.4.8 Quantum Singular Value Decomposition

QSVD (Quantum Singular Value Decomposition) is another variational NISQ algorithm that generalizes the classical singular value decomposition (SVD) to the quantum setting.

It allows an efficient decomposition of quantum states into a set of singular values and corresponding orthonormal bases [1171]. Some interesting algorithms make use of QSVD like a proposal for a quantum page rank algorithm [1172].

QSVD's main practical weakness is that the orthogonality of the computed singular vectors degrades as their number grows, which caps the achievable precision. A 2026 Japanese proposal restores high precision by applying a classical orthogonality correction after the variational step [1173]. The accuracy gain is real, and so is the pattern, since yet another part of the algorithm moves to the classical side.

1.4.9 Barren plateaus

Variational algorithms can encounter barren plateaus, regions in which gradients become exponentially small with system size and therefore require an exponentially growing number of samples to resolve [1165, 1174]. This is not equivalent to saying that every deep VQA has a barren plateau or that every trainable VQA must be

shallow. The behavior depends on ansatz expressivity, initialization, cost-function locality, symmetries, circuit depth and noise [1166]. For sufficiently random or expressive circuit families with global observables, one often obtains an asymptotic behavior of the form

$$\text{Var}[\partial_k C] = O(2^{-n})$$

under the assumptions of the corresponding barren-plateau theorem. By contrast, local cost functions with structured circuits of depth $O(\log n)$ can have gradients that vanish only polynomially [1166].

Research is very active to fix this problem [698, 1175] like by adding additional parameters and constraints to improve gradients in the variational training loop without resorting to inefficient overfitting [1176], with narrowing the scope to dissipative quantum algorithms based on non-unital channels [1177], using tensor networks to create the parametrized circuit [1178] or using neural networks to build the circuit as well [1179]. It seems that the barren plateau syndrome could be avoided in VQE algorithms [1176, 1180–1183].

It is a matter of balancing circuit expressivity and trainability [1184–1187].

There are structured conditions under which barren plateaus can be avoided in QAOA-like circuits [1188] and quantum convolutional neural networks [1189]. A 2025 Nature Communications perspective makes a deeper connection to dequantization: many of the structures currently used to guarantee trainability confine the dynamics to polynomially sized or otherwise classically accessible subspaces [936]. The authors explicitly present this as case-by-case evidence, not a theorem that every barren-plateau-free circuit is classically simulable. Increasing structure can improve trainability while simultaneously strengthening classical simulation methods.

A particularly relevant 2026 result comes from Mele et al. [1190]. For expectation-value estimation in typical circuits subject to local incoherent noise, they show that the influence of sufficiently early layers becomes exponentially suppressed, giving an effective logarithmic depth. For non-unital noise and local observables they also prove the absence of barren plateaus. Under the same assumptions, however, they construct an efficient classical algorithm that estimates those expectation values to constant additive accuracy with high probability over the circuit ensemble. This is a rigorous trainability-versus-simulability result for an important noisy expectation-value regime.

For QAOA, dynamical Lie algebra analysis provides a useful but model-dependent trainability diagnostic. A 2025 study shows that multi-angle QAOA for MaxCut has exponentially large Lie algebras for connected graphs other than paths and cycles, making those ansatzes highly prone to barren plateaus [1191]. A 2025 preprint reports $\Theta(4^n)$ DLA dimension almost surely for continuously weighted connected graphs except paths and

cycles, and for asymptotically almost all unweighted graphs [1192].

Careful initialization is usually offered as the cure, but a 2026 Los Alamos, EPFL and Fujitsu Research collaboration shows the situation is subtler. Their first-moment framework provides an operator-level diagnostic for when an initialization escapes the fully concentrated fixed point, and reveals that barren plateau avoidance is highly non-unique, with exponentially many families of inequivalent shifted, biased or non-symmetric initialization distributions all avoiding concentration while inducing different biases [1193]. Choosing among them therefore becomes an additional unsolved design problem rather than a settled recipe.

Other 2025-2026 work refines the diagnosis and the workarounds. Barren plateaus can arise through mechanisms beyond the usual observable concentration [1194], task-aligned symmetry in the ansatz can defeat them [1195], a non-unitary ansatz mitigates the noise-induced variety in molecular electronic transport calculations [1196], gradient troughs specific to ADAPT-VQE can be escaped with targeted operator selection strategies [1197], and a comprehensive numerical study maps the interaction between barren plateaus and over-parametrization [1198]. Tensor networks are once again the classical crutch of choice, here to prepare multivariate function states efficiently [1199]. Two results push against the general pessimism, though neither runs on hardware: parameterized quantum circuits can display the double descent phenomenon familiar from classical deep learning, with generalization on unseen data improving again as the model grows past the interpolation threshold [1200], and exponential learning separations survive for dynamical correlators estimated through operator shadows [1201].

The emerging picture can be summarized as a **trainability-expressivity-simulability triangle**. A VQA must be expressive enough to represent states or solution distributions beyond efficient classical methods, trainable with a polynomially manageable number of noisy or logical circuit evaluations, and measurable with an affordable precision budget. Making the ansatz shallower, more local, lower-entanglement or more symmetric often improves trainability, but those same restrictions can make tensor-network, neural-state or reduced-subspace simulation more efficient [935, 936, 1190]. This is the most important conceptual connection with dequantization.

1.4.10 Analog quantum computing and simulations

Quantum annealing and analog quantum computing are not the darlings of the quantum computing industry. On one hand, in quantum annealing, D-Wave has been criticized for a long time for “not being quantum” nor being in a position to bring any computing advantage.

On the other hand, analog quantum computers (programmable Hamiltonian simulation or programmable quantum simulators) are developed and commercialized by a very small number of vendors such as PASQAL and QuEra and said to have their own scalability challenges.

Still, you find many solutions running on these systems that seem closer to some quantum advantage than existing gate-based QPUs [1202]. Recent benchmarks also show that analog quantum computers currently have greater computing capacity than gate-based noisy quantum computers as discussed in the Q-Score section.

Analog quantum advantage

Recent papers also tend to demonstrate the capacity to reach a quantum advantage with a quantum simulator [1203, 1204] and with a quantum annealer [1205]. Most of these solutions rely on formulating a problem into a QUBO (Quadratic Unconstrained Binary Optimization), a generic form of binary optimization problem. It is supported by analog quantum computers including quantum annealers from D-Wave and quantum simulators from QuEra and Pasqal [1206]. But a QUBO problem can also be reformulated into a QAOA problem that can be solved on a gate-based QPU [1207].

In quantum annealers and simulators, QUBO problems are converted in a problem manageable by the hardware through the process of graph embedding that is then natively processed in the QPU. There is some more flexibility in the graph shape and form with quantum simulators than with quantum annealers.

Solving optimization problems

Various optimization problems, in turn, can be converted in QUBO models and sometimes QAOA variational models, including for solving some many-body quantum physics simulation problems like Ising models [1208], though, not fermionic systems nor chemical electronic structure simulations although some inroads are done here [1209], and even financial optimization problems [1210, 1211]. QUBO algorithms can for example solve TSP problems (traveling salesperson problem) [1212, 1213], 3SAT combinatorial problems [1214–1217], Max-Cut problems [1218], Simon’s period finding problem [1219], Maximum Independent Set problems [1220], machine learning problems like a Principal Component Analysis (PCA) [1221], graph coloring problems [1222], and job-shop scheduling problems in manufacturing where multiple jobs must be processed on different machines, each job having a specific sequence of operations and each operation requiring a specific machine for its execution.

But this doesn’t mean it delivers a practical quantum advantage vs best-in-class classical solutions. The goal is to determine an optimal schedule that minimizes the time required to complete all jobs or other performance criteria and resource constraints, aka solving the job shop problem [1223–1226]. The formulation can

even be stretched to continuous optimization, which annealers cannot address natively, by approximating an unconstrained quadratic program through a sequence of QUBO instances whose continuous weights carry the real variables, decoupling the qubit count from the required precision [1227], although the demonstration is limited to linear regression. It can also solve more exotic problems like the snake-in-the-box problem (finding the longest simple path through a binary hypercube graph without revisiting any node) and the coil-in-the-box problem (same, with a closed loop) [1228].

Sheir Yarkoni et al’s 2022 review paper on quantum annealing provided a status of the D-Wave platform usability [1229]. It inventoried a broad set of algorithms and trials related to mobility traffic flow optimization and vehicle routing problem [1230], solving the Closest String Problem (CSP) which is found in computational biology and data mining, and is about finding a string as close as possible to a set of strings (like a series of nucleotides or amino acids) [1231], DNA sequence alignment [1232], scheduling and logistics problems, finance portfolio optimization, quantum simulation, material design [1233], physics, biology, machine learning (classification, reinforcement learning, cluster analysis), matrix factorization and other finite-element design. All these algorithms are hybrid like most NISQ known algorithms. Compared to the various known gate-based NISQ algorithms, quantum annealers are more generic than usually thought. Yarkoni et al. highlighted that quantum annealing can only address specific problem formulations and implements metaheuristic quantum optimizations, meaning approximate solutions (Figure 67). Also, the number of needed physical qubits scales polynomially with the number of logical variables from the problem formulation.

Other sources found gate-based QAOA to be potentially more efficient than quantum-annealing-based QUBO [1234]. The review paper lists some improvements required at the hardware level to generate some quantum advantage such as additional qubit control, driver Hamiltonians and operators, and higher connectivity, part of which are now available in D-Wave’s Advantage2 QPU, in production since mid-2025 with 4,400 qubits.

QUBO problem formulation also has a variant, Higher-order Unconstrained Binary Optimization (HUBO), that is involved in solving more generic search and optimization problems. For example, a new variation of HUBO can reduce the number of qubits needed to run a Grover adaptive search [1235]. Also, the broader polynomial unconstrained binary optimization (PUBO) problems can be reformulated into QUBO problems or solved directly, like for a 3SAT problem and more efficiently [1236]. Two 2026 papers quantify what this buys. A systematic HUBO construction benchmarked on Gate Assignment, Maximum k -Colorable Subgraph and Integer Programming reduces qubit counts and cuts total CNOT counts by at least 89.6% against the QUBO formulation on every tested instance, with an open-

source library released to build the models [1237]. On the algorithm side, the Grover-mixer variant of QAOA improves monotonically with circuit depth on HUBO and PUBO problems where the transverse-field mixer does not, and an analytical model of its dynamics gives a classical approximation of the optimal parameters, removing much of the outer loop cost [1238]. A third route encodes the decision variables directly as qudits rather than binaries, which captures the constraints of TSP, two vehicle routing variants, graph coloring, job scheduling and Max- K -Cut without penalty terms and improves approximation ratios at comparable depth, at the price of hardware that barely exists [1239].

Q The spectral gap and its workarounds

The adiabatic theorem requires evolution to be slow compared with a scale set by the minimum spectral gap and the rate at which the Hamiltonian changes. In common simplified bounds, the runtime contains a factor proportional to $\max_s \|dH/ds\|/\Delta_{min}^2$, although tighter bounds and locally optimized schedules can change the prefactors and gap dependence. Some hard instances exhibit exponentially small minimum gaps, which can make straightforward adiabatic evolution exponentially slow. This is an instance-dependent phenomenon rather than a statement that every large system undergoes such a gap closing.

Shortcuts to Adiabaticity (STA), including counter-diabatic driving, add control terms intended to suppress diabatic transitions and reach a target state faster than a naive adiabatic schedule. The achievable speedup is platform and problem dependent and comes at the price of additional control complexity, interaction terms or pulse bandwidth. It doesn’t reduce an arbitrarily long evolution to microseconds.

In analog quantum simulators, STA techniques are used for tasks such as state preparation, transport and expansion of cold-atom systems. Related optimal-control and counter-diabatic ideas can also inform pulse design in gate-based hardware.

Two more 2026 encoding results follow the same logic of avoiding penalties: an all-valid-state HUBO encoding restricts the search space to feasible assignments by construction [1240], and a hardware-efficient method selects the auxiliary variables used in quadratization so as to minimize the added qubit and coupling overhead [1241]. On the benchmarking side, a systematic set of QUBO formulations and benchmark instances for facility location problems has been released, which is the sort of shared baseline the field still lacks [1242]. A claim of sub-exponential speedup for gate-based QUBO solving

also appeared [1243], with no hardware validation and no comparison against a strong classical solver, so it should be read as a complexity conjecture rather than a result.

The most instructive paper of this batch is a low-key one. A George Mason and Old Dominion team shows that the standard practice of translating an agent-based model into a QUBO, using Schelling’s segregation model on lollipop networks as the case study, is architecturally incompatible with quantum optimization, because observing the state of the system at every iteration, which is what an agent-based model does by definition, systematically destroys the superposition on which any advantage would rest [1244]. Reformulating the question, from simulating agent dynamics until convergence to computing the minimum number of agent moves needed for global satisfaction, exposed network symmetries hidden by both the agent-based and the QUBO formulations, and yielded a faster purely classical algorithm which then becomes the new lower bound quantum approaches must beat. It is a compact illustration of a recurring pattern: the classical baseline usually improves as soon as somebody looks at the problem structure seriously, and a good part of the quantum optimization literature is competing against classical algorithms that nobody bothered to optimize.

In detail, many algorithms tested on quantum annealers can solve sizeable problems, but usually, still under the demanding quantitative requirements levels of real-life scenarios [1245]. Some examples come mostly from the financial sector with portfolio optimizations [1246–1248]. Then we have also financial index tracking [1249] and foreign exchange optimizations [1250].

Analog quantum computers bring more flexibility on paper with the ability to define arbitrary graph trees with better connectivity. It can be used for simulating the physics of some topological spin liquid [1251], to predict which companies could fail in loans reimbursements [1252] and solve various graph problems [1253] and other machine learning problems [1254].

In a 2023 experiment in South Korea with a home-grown Rydberg simulator, researchers could solve a MIS (maximum independent set) problem using up to 141 atoms, although not seemingly in a quantum advantage regime [1255].

When reviewing all these case studies, both in the gate-based and analog quantum computing categories, one thing is striking: the most powerful solutions available are in the analog space rather than in the gate-based space. Quantum inspired classical solutions implementing linear algebra and tensor networks computing are also making classical computing currently more competitive in several areas [1256]. These are not quantum at all. Then, other use cases directly put you in the FTQC zone, requiring thousands of logical qubits and thus, up to millions to hundred million physical qubits.

Benchmarks must be analyzed carefully. Comparisons must be fair, using the best classical optimizers as references, like CPLEX, and comparing QPUs with classical systems of similar prices, not a simple PC [1257].

Scalability challenges

However, even if analog quantum computing existing use cases are closer to real-life production grade levels than the gate-based equivalents, there are still some challenges to overcome in generating a quantum advantage with analog quantum computers as summarized in Figure 66 [1258]. Quantum annealers require more tunability of qubit connections and better qubits connectivity.

	modus operandi	some scaling challenges
quantum annealers	find a ground state of an Ising model, optimization problems are mapped to Ising models (QUBO)	• qubit connectivity. • noise mitigation. • large scale tunnelling effect. • all to all long range interactions.
quantum simulators	find a ground state of an Ising model or XY quantum simulation model (with more degrees of liberty)	• laser power (neutral atoms). • atoms positioning at scale. • large scale and long range entanglement. • noise mitigation.

Figure 66: Some scaling challenges for quantum annealers and analog quantum computers. Source: © Olivier Ezratty, 2023.

Noise mitigation must also be handled [1259–1261]. Some techniques are used to benchmark the quality of entanglement of these systems [1262]. And there’s a remaining question similar as the one with NISQ systems: how far large scale quantum effects can operate, particularly, based on the tunnel effect that is at the core of quantum annealing. Neutral atoms scaling is linked to the ability to control large chunks of well-positioned entangled atoms in ultra-vacuum. The related tools are made of more powerful and stable lasers, and their related control electronics.

In [1263], Thomas Ayrat et al. from Eviden investigated in 2025 the practical and theoretical limitations of using Rydberg atom analog QPU to solve combinatorial optimization problems at scale and beyond UD-MIS problems with more general graph structures and embedding schemes like the Crossing Lattice construction, which introduces ancilla atoms and defective regions to simulate arbitrary connectivity. While this allows the analog processor to represent more complex problems, interpreting the quantum output in a way that maps back to the original problem either yields high-quality results but is computationally expensive, scaling exponentially with system size, or it ignores defective regions, is efficient but produces poor-quality solutions. They conclude that the utility of these analog QPUs for solving high-quality optimization at scale is limited unless new methods for interpretation or embedding and classical post-processing are developed.

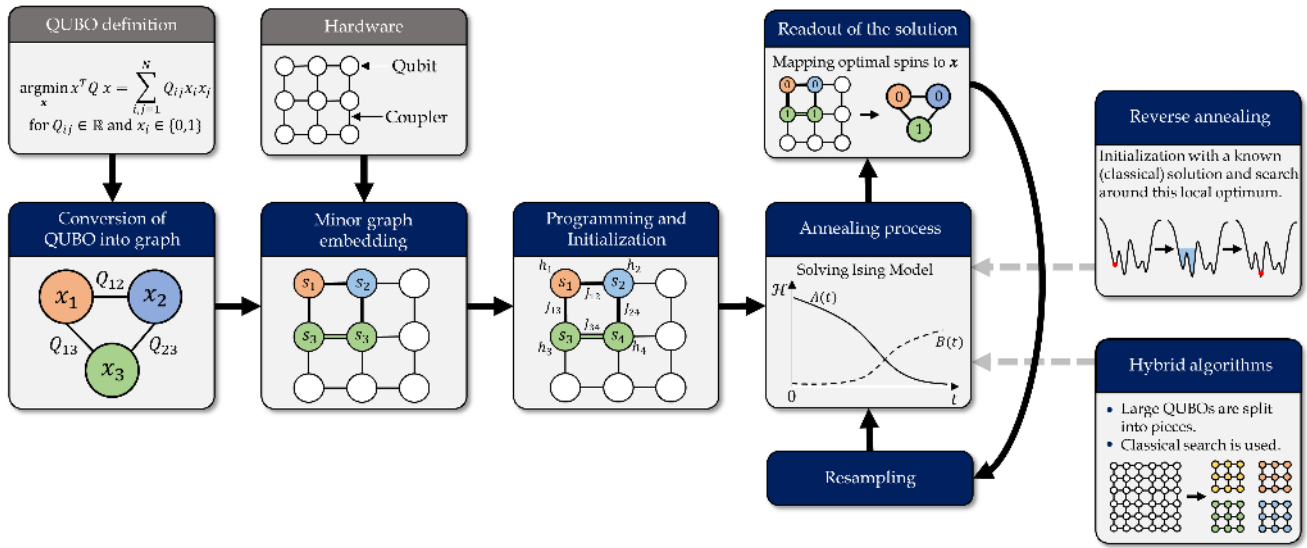


Figure 67: A quantum annealing process taking the example of an optimization problem encoded in a QUBO problem and a related graph. The graph is automatically mapped to a graph compatible with the qubit topology of a D-Wave QPU through the process of “minor graph embedding”. The graph is then encoded in the system with an initialization of the qubit weights and connections. The annealing process creates a transverse magnetic field that sets the qubits value in a superposed state and progressively removing this field, which implements the annealing process, converging the qubit towards their optimum value minimizing the total system energy. Then the qubits are read out. The process is repeated several times and the result values averaged. Variations involved reverse annealing when the graph initialization uses a known classical solution and the annealing process helps find a better solution around it, and hybrid algorithms where large QUBO problems are split classically into smaller QUBO problems. Source: Sheir Yarkoni et al. in [1229]. Added in 2023.

Another concern is about the limits of adiabaticity for solving optimization problems. The larger the system, the greater is the number of energy levels to “traverse”. This lengthens the annealing process on both quantum annealers and analog cold-atom based QPUs. This annealing can become lengthier than the coherence time of the QPU. It creates some upper bounds on the size of the problems that can be solved with these systems [1264–1267].

Digital-Analog Quantum Computing

DAQC (Digital-Analog Quantum Computing) is about implementing a hybrid gate-based and analog quantum computing model [1268, 1269] (Figure 68). DAQC is supposed to make more efficient use of quantum computing resources, is less sensitive to noise [1270], enable NISQ algorithms with fewer qubits, and run faster than regular NISQ algorithms. It is adapted to solving a broad set of problems in many body physics, optimization, machine learning [1271], and linear equations [1272].

DAQC is proposed by Kipu Quantum (Germany) and Qilimanjaro (Spain). Kipu Quantum is investigating the use of superconducting, trapped ion and neutral atoms qubits for its implementation. An implementation proposal using superconducting qubits could use SQUIDs to connect qubits in 2D matrices [1274].

It is also a computing paradigm explored with neutral atoms, where analog blocks can exploit programmable Rydberg interactions and local laser control. It can reduce digital compilation overhead for selected Hamil-

tonians [1275]. The achievable speedup remains strongly algorithm dependent [1276], and the additional analog-control requirements complicate calibration, verification and software tooling. A 2023 digitized-counterdiabatic factorization paper demonstrated a highly reduced 48-bit factorization instance on 10 Quantinuum qubits and extrapolated the approach toward RSA-2048 [1277]. That extrapolation was not a resource-estimated demonstration that a NISQ DAQC platform can factor RSA-2048.

Another paradigm mixes quantum annealing and gate-based quantum computing. It is proposed by a team of Taiwan researchers with their large-system sampling approximation (LSSA) algorithm that solves Ising problems more efficiently than quantum annealers alone [1278].

D-Wave has now entered this space directly. In a 2026 paper co-signed with Daniel Lidar, Gabriel Aeppli and Markus Müller, the company demonstrated analog-digital quantum computing on a highly multiplexed superconducting quantum annealing processor, using evolution under a fixed many-body Hamiltonian which in the weak-coupling regime is well described by an effective XY model, combined with arbitrary-basis initialization and measurement through auxiliary qubits [1279]. This widens the operation set of an annealer well beyond the uniform attenuation of quantum fluctuations, which is exactly the simplification that allowed annealers to scale to thousands of qubits in the first place, so the interesting question is how much of that scaling survives the added control requirements.

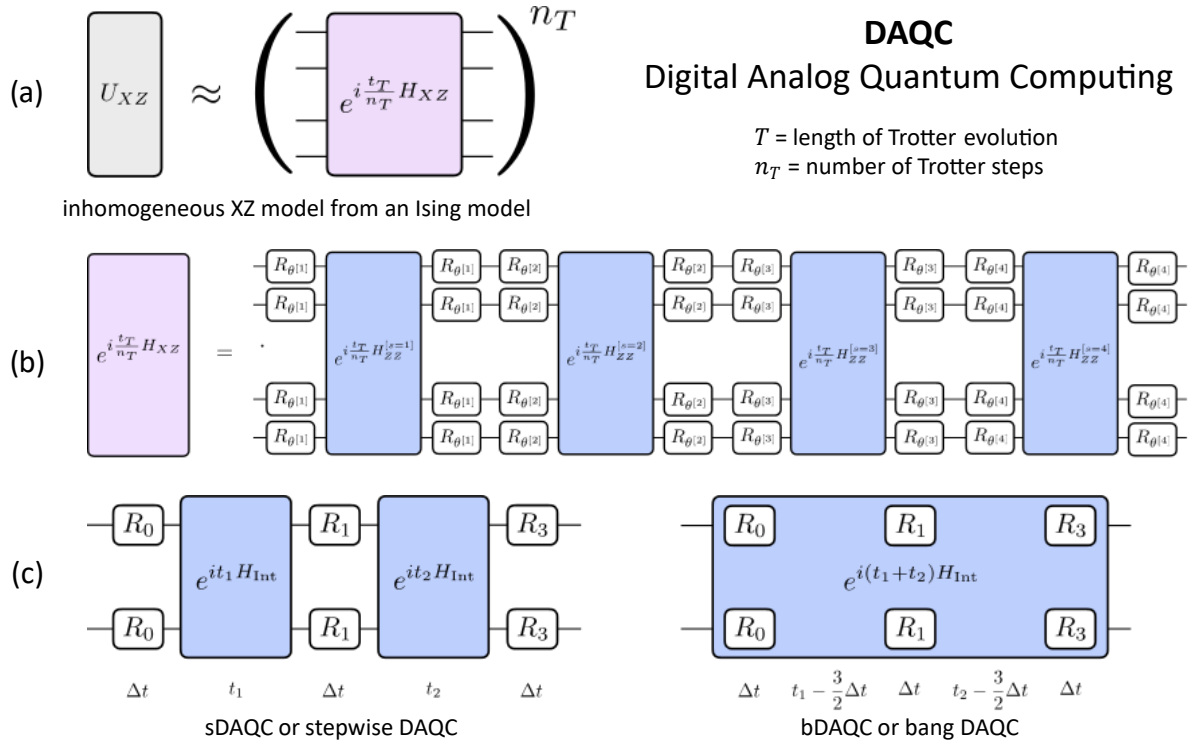


Figure 68: Description of the DAQC paradigm for solving an inhomogeneous model from an Ising model. It is decomposed in a series of Hamiltonian evolutions by a trotter decomposition of length T and n_T Trotter steps (a). Each step is decomposed in a global Hamiltonian implemented in an analog mode with analog blocks implementing entanglement, surrounded by two series of single qubit rotation gates (SQRs). It is conceptually not far from the way variational quantum algorithm ansatzes are constructed. There are two variations for this model with the stepwise DAQC model (sDAQC) with a time separation of the rotations and the analog Hamiltonian evolution and the bang DAQC (bDAQC) with several single qubit gate rotations cycles implemented during a longer Hamiltonian evolution. Source: [1273]. Assembled by Olivier Ezratty, September 2024.

Separately, Daniel Lidar built a quantum supremacy framework for a digital-analog-digital model consisting of one single-qubit layer, a single transverse-field Ising analog block and a final single-qubit layer before Z-basis readout [1280]. He proved that the output distribution stays within constant total-variation distance of an IQP circuit, for fully connected as well as bounded-degree hardware graphs matching trapped ion, neutral atom and superconducting architectures, with hardness resting on an anticoncentration property proved for all-to-all graphs and conjectured for bounded-degree ones, plus an average-case hardness conjecture on complex-temperature Ising partition functions. As with every sampling-based supremacy argument, this establishes the classical hardness of the sampling task and says nothing about the usefulness of the samples.

On the algorithm side, the hyperactive Kipu Quantum proposed a constant-depth digital-analog counterdiabatic scheme [1281], hybrid digital-analog protocols were developed for simulating multi-body interactions with trapped ions [1282], and shortcuts to adiabaticity have been optimized numerically through a hybrid control strategy [1283].

1.4.11 Digitized Counter-Diabatic Quantum Optimization

Digitized Counter-Diabatic Quantum Optimization (DCQO) is a gate-based heuristic that digitizes ideas from adiabatic quantum optimization and counter-diabatic driving [1284, 1285]. Additional approximate counter-diabatic terms are used to reduce diabatic transitions while keeping the circuit shallower than a direct digitization of a slow adiabatic path. Reported success-probability improvements are problem-, ansatz- and benchmark-dependent, so DCQO cannot be described as generically outperforming traditional optimization methods without a matched classical baseline.

DCQO is explored for solving combinatorial optimization problems like the Bin Packing Problem [1286], the MaxCut Problem, Job Shop Scheduling Problem (JSSP) and other logistics problems [1287], Portfolio Optimization, Protein Folding and Molecular Docking, and Higher-Order Unconstrained Binary Optimization (HUBO) problems.

One of the main promoters of this technique is Kipu Quantum (Germany).

1.4.12 NISQ distributed computing

Another option is to distribute a quantum computation across several QPUs using **circuit cutting** with classical reconstruction [1288–1290]. Its viability depends strongly on the entanglement geometry [1291]. Optimizing cut placement can itself be computationally hard, while the reconstruction overhead grows exponentially with the number and type of cuts in generic settings [1292, 1293]. This creates another dequantization-like tension: circuits with sparse enough entanglement to cut efficiently are often favorable for tensor-network contraction as well [1294]. For genuinely modular large-scale FTQC, coherent interconnects and teleported logical gates are a different architectural route whose crossover against circuit cutting depends on link fidelity, entanglement-generation rate and workload [1295].

Among other scenarios, it is proposed to distribute some separable quantum neural network classification tasks to several QPUs [1296] and even more generic tasks [1297]. In April 2026, two independent teams posted papers with the same title, distributed variational quantum

linear solver, within two weeks of each other. The Oak Ridge and NVIDIA version targets the large number of circuit evaluations generated by a linear-combination-of-unitaries decomposition [1298]. In the generic dense case the number of decomposition terms can grow exponentially, which makes the $O(L^2)$ expectation-evaluation structure expensive. Their CUDA-Q framework distributes these evaluations asynchronously. For the sparse structured matrices studied, a fast Walsh-Hadamard-transform Pauli decomposition with 1% co-efficient thresholding reduces the retained term count dramatically and changes the practical per-iteration scaling. This favorable reduction is structure- and threshold-dependent. It can't be generalized to arbitrary matrices.

The second paper shares the title and the objective [1299]. Both address the classical side of the loop, which is a fair reflection of where the cost actually sits. QAOA has also been evaluated across distributed quantum network topologies, where the entanglement distribution overhead between nodes becomes the limiting factor rather than the qubit count [1300].



1.4.13 Bibliography and notes

These are the bibliographical references and notes for the **NISQ algorithms** subsection.

- [42] [Topological data analysis on noisy quantum computers](#), by Ismail Yunus Akhalwaya, Lior Horesh, et al., arXiv, March 2024 (34 pages).
- [54] [The bitter truth about gate-based quantum algorithms in the NISQ era](#), by Frank Leymann and Johanna Barzen, Quantum Science and Technology, September 2020 (29 pages).
- [161] [A Quantum Approximate Optimization Algorithm](#), by Edward Farhi, Jeffrey Goldstone, and Sam Gutmann, arXiv, November 2014 (16 pages).
- [162] [A variational eigenvalue solver on a photonic quantum processor](#), by Alberto Peruzzo, Jarrod McClean, Peter Shadbolt, Man-Hong Yung, Xiao-Qi Zhou, Peter J. Love, Alán Aspuru-Guzik, and Jeremy L. O'Brien, Nature Communications, July 2014 (7 pages).
- [314] [Solving partial differential equations on near-term quantum computers](#), by Anton Simen Albino, Erick Giovanni Sperandio Nascimento, et al., arXiv, August 2022 (9 pages).
- [461] [The generative quantum eigensolver \(GQE\) and its application for ground state search](#), by Kouhei Nakaji, Alan Aspuru-Guzik, et al., arXiv, January 2024 (16 pages).
- [466] [Is quantum advantage the right goal for quantum machine learning?](#), by Maria Schuld and Nathan Killoran, arXiv, February 2023 (13 pages).
- [576] [Configured Quantum Reservoir Computing for Multi-Task Machine Learning](#), by Wei Xia, Xiaopeng Li, et al., arXiv, March 2023 (15 pages).
- [696] [Challenges and Opportunities in Quantum Machine Learning](#), by M. Cerezo, Guillaume Verdon, Hsin-Yuan Huang, Lukasz Cincio, and Patrick J. Coles, arXiv, March 2023 (14 pages).
- [698] [A Survey of Methods for Mitigating Barren Plateaus for Parameterized Quantum Circuits](#), by Michelle Gelman, arXiv, June 2024 (18 pages).
- [832] [An adaptive variational algorithm for exact molecular simulations on a quantum computer](#), by Sophia Economou et al., Nature Communications, July 2019 (9 pages).
- [833] [An adaptive variational algorithm for exact molecular simulations on a quantum computer](#), by Harper R. Grimsley, Sophia E. Economou, Edwin Barnes, and Nicholas J. Mayhall, Nature Communications, July 2019.
- [910] [Quantum Computing in the NISQ era and beyond](#), by John Preskill, arXiv, July 2018 (20 pages).
- [911] [An Overview of Noise-Adaptive Quantum Algorithms](#), by Jordan Makansi, The Quantum Insider, June 2025.
- [912] [Where are we heading with NISQ?](#), by Olivier Ezratty, arXiv, June 2023 (47 pages).
- [913] [Fast, accurate, high-resolution simulation of large-scale Fermi-Hubbard models on a digital quantum processor](#), by Gavin S. Hartnett, Khadijeh Sona Najafi, Aleksei Khindanov, Haoran Liao, Michael Schutzman, Michael R. Hush, Michael J. Biercuk, and Yuval Baum, arXiv, May 2026 (35 pages).
- [914] [Fundamental limits of quantum error mitigation](#), by Ryuji Takagi, Suguru Endo, Shintaro Minagawa, and Mile Gu, npj Quantum Information, September 2022.
- [915] [Exponentially tighter bounds on limitations of quantum error mitigation](#), by Yihui Quek, Daniel Stilck França, Sumeet Khatri, Johannes Jakob Meyer, and Jens Eisert, Nature Physics, July 2024 (11 pages).
- [916] [Scalability challenges in variational quantum optimization under stochastic noise](#), by Adelina Bärligea, Benedikt Poggel, and Jeanette Miriam Lorenz, Physical Review A 112, September 2025 (25 pages).
- [917] [Variational Quantum Algorithms](#), by Marco Cerezo, Patrick J. Coles, et al., arXiv, October 2021 (33 pages).

- [918] [Quantum Chemistry in the Age of Quantum Computing](#), by Yudong Cao, Alán Aspuru-Guzik, et al., arXiv, December 2018 (194 pages).
- [919] [Variational Quantum Linear Solver](#), by Carlos Bravo-Prieto, Ryan LaRose, M. Cerezo, Yigit Subasi, Lukasz Cincio, and Patrick J. Coles, arXiv, November 2023 (8 pages).
- [920] [A Review of Variational Quantum Algorithms: Insights into Fault-Tolerant Quantum Computing](#), by Zhirao Wang, Junxiang Huang, Runyu Ye, Qingyu Li, Qi-Ming Ding, Yiming Huang, Ting Zhang, Yumeng Zeng, Jianshuo Gao, Xiao Yuan, and Yuan Yao, arXiv, April 2026 (28 pages).
- [921] [A fault-tolerant variational quantum algorithm with limited T-depth](#), by Hasan Sayginel, Francois Jamet, Abhishek Agarwal, Dan E. Browne, and Ivan Rungger, Quantum Science and Technology, January 2024.
- [922] [Feasibility of performing quantum chemistry calculations on quantum computers](#), by Thibaud Louvet, Thomas Ayrat, and Xavier Waintal, Physical Review B, March 2026 (14 pages)  (preprint on [arXiv](#)).
- [923] [Recent Advances in Quantum Architecture Search](#), by Haozhen Situ, Zhimin He, and Lvzhou Li, arXiv, April 2026 (17 pages).
- [924] [AI Agents for Variational Quantum Circuit Design](#), by Marco Knipfer, Alexander Roman, Konstantin T. Matchev, Katia Matcheva, and Sergei Gleyzer, arXiv, February 2026 (43 pages).
- [925] [How to find expressible and trainable parameterized quantum circuits?](#), by Peter Röseler, Dennis Willsch, and Kristel Michielsen, arXiv, March 2026 (25 pages).
- [926] [Variational Gibbs State Preparation on NISQ devices](#), by Mirko Consiglio, Jacopo Settino, Andrea Giordano, Carlo Mastroianni, Francesco Plastina, Salvatore Lorenzo, Sabrina Maniscalco, John Gould, and Tony J. G. Apollaro, arXiv, July 2024 (15 pages).
- [927] [Verifiably Exact Solution of the Electronic Schrödinger Equation on Quantum Devices](#), by Scott E. Smart and David A. Mazziotti, arXiv, March 2023 (8 pages).
- [928] [Orders of magnitude reduction in the computational overhead for quantum many-body problems on quantum computers via an exact transcorrelated method](#), by Igor O. Sokolov, Werner Dobrautz, Hongjun Luo, Ali Alavi, and Ivano Tavernelli, arXiv, March 2023 (21 pages).
- [929] [Variational Quantum Search with Shallow Depth for Unstructured Database Search](#), by Junpeng Zhan, arXiv, September 2023 (16 pages).
- [930] [Training Variational Quantum Algorithms Is NP-Hard](#), by Lennart Bittel and Martin Kliesch, Physical Review Letters, September 2021  (preprint on [arXiv](#)).
- [931] [Undecidable problems associated with variational quantum algorithms](#), by Georgios Korpas, Vyacheslav Kungurtsev, and Jakub Mareček, arXiv, March 2025 (19 pages).
- [932] [Tensor-network-assisted variational quantum algorithm](#), by Junxiang Huang, Xiao Yuan, et al., arXiv, November 2023 (12 pages).
- [933] [Combining Matrix Product States and Noisy Quantum Computers for Quantum Simulation](#), by Baptiste Anselme Martin, Thomas Ayrat, François Jamet, Marko J. Rančić, and Pascal Simon, arXiv, January 2024 (14 pages).
- [934] [QUBO-based training for VQAs on Quantum Annealers](#), by Ernesto Acosta, Guillermo Botella, and Carlos Cano, arXiv, September 2025 (33 pages).
- [935] [Classical variational simulation of the Quantum Approximate Optimization Algorithm](#), by Matija Medvidović and Giuseppe Carleo, npj Quantum Information, June 2021.
- [936] [Does provable absence of barren plateaus imply classical simulability?](#), by M. Cerezo, Martin Larocca, Diego García-Martín, N. L. Diaz, Paolo Braccia, Enrico Fontana, Manuel S. Rudolph, Pablo Bermejo, Aroosa Ijaz, Supanut Thanasilp, Eric R. Anschuetz, and Zoë Holmes, Nature Communications, August 2025.
- [937] [A comprehensive survey on quantum computer usage: How many qubits are employed for what purposes?](#), by Tsubasa Ichikawa, Keisuke Fujii, et al., arXiv, October 2023 (14 pages).
- [938] [Quantum advantage with shallow circuits](#), by Sergey Bravyi, David Gosset, and Robert Koenig, arXiv, April 2017 (23 pages).
- [939] [Simulating the quantum Fourier transform, Grover's algorithm, and the quantum counting algorithm with limited entanglement using tensor-networks](#), by Marcel Niedermeier, Jose L. Lado, and Christian Flindt, arXiv, September 2024 (16 pages).
- [940] [Assessing requirements to scale to practical quantum advantage](#), by Michael E. Beverland, Alexander Vashchillo, et al., arXiv, November 2022 (41 pages).
- [941] [The cost of solving linear differential equations on a quantum computer: fast-forwarding to explicit resource counts](#), by David Jennings, Andrew T. Sornborger, et al., arXiv, November 2024 (38 pages).
- [942] [Noisy intermediate-scale quantum computers](#), by Bin Cheng, Dapeng Yu, et al., arXiv, March 2023 (50 pages).
- [943] [Measurements as a roadblock to near-term practical quantum advantage in chemistry: resource analysis](#), by Jérôme F. Gonthier, Jhonathan Romero, et al., arXiv, August 2022 (49 pages).
- [944] [Quantum Computing's dirty little secret](#), by Amara, Amara's Substack, April 2025.
- [945] [Special Session: Noisy Intermediate-Scale Quantum \(NISQ\) Computers – How They Work, How They Fail, How to Test Them?](#), by Sebastian Brandhofer, Simon Devitt, Thomas Wellens, and Ilia Polian, arXiv, January 2023 (10 pages).
- [946] [Error per single-qubit gate below \$10^{-4}\$ in a superconducting qubit](#), by Zhiyuan Li, Haifeng Yu, et al., arXiv, February 2023 (7 pages).
- [947] [Limitations of Noisy Quantum Devices in Computational and Entangling Power](#), by Yuxuan Yan, Zhenyu Du, Junjie Chen, and Xiongfeng Ma, arXiv, June 2023 (18 pages).
- [948] [NISQ computing: where are we and where do we go?](#), by Jonathan Wei Zhong Lau, September 2022 (30 pages).
- [949] [A Case for Variability-Aware Policies for NISQ-Era Quantum Computers](#), by Swamit S. Tannu and Moinuddin K. Qureshi, arXiv, May 2018 (12 pages).
- [950] [Bounding quantum gate error rate based on reported average fidelity](#), by Yuval R. Sanders et al., 2015 (14 pages).
- [951] [Variational Quantum Eigensolvers in the Era of Distributed Quantum Computers](#), by Ilia Khait, Edwin Tham, Dvira Segal, and Aharon Brodutch, arXiv, February 2023 (5 pages).
- [952] [Quantum Computing in the NISQ era and beyond](#), by John Preskill, Quantum Journal, July 2018 (20 pages).
- [953] [QAOA for Max-Cut requires hundreds of qubits for quantum speed-up](#), by G. G. Guerreschi and A. Y. Matsuura, Nature Scientific Reports, May 2019.
- [954] [Error Mitigation for Universal Gates on Encoded Qubits](#), by Christophe Piveteau, David Sutter, Sergey Bravyi, Jay M. Gambetta, and Kristan Temme, Physical Review Letters, November 2021  (preprint on [arXiv](#)).
- [955] [Optimal ancilla-free Clifford+T approximation of z-rotations](#), by Neil J. Ross and Peter Selinger, Quantum Information and Computation, September 2016 (53 pages) .
- [956] [Probabilistic Interpolation of Quantum Rotation Angles](#), by Bálint Koczor, John Morton, and Simon Benjamin, arXiv, November 2023 (15 pages).
- [957] [Towards Advantages of Parameterized Quantum Pulses](#), by Zhiding Liang, Yiyu Shi, et al., arXiv, March 2024 (15 pages).
- [958] [NAPA: Intermediate-level Variational Native-pulse Ansatz for Variational Quantum Algorithms](#), by Zhiding Liang, Yiyu Shi, et al., arXiv, January 2024 (13 pages).

- [959] [PANSATZ: Pulse-based Ansatz for Variational Quantum Algorithms](#), by Dekel Meirom and Steven H. Frankel, arXiv, March 2023 (11 pages).
- [960] [Gate-free state preparation for fast variational quantum eigensolver simulations](#), by Oinam Romesh Meitei, Sophia E. Economou, Nicholas J. Mayhall, et al., npj Quantum Information, October 2021.
- [961] [Minimizing state preparation times in pulse-level variational molecular simulations](#), by Ayush Asthana, Sophia E. Economou, Nicholas J. Mayhall, et al., arXiv, March 2022 (12 pages).
- [962] [Parameterization and optimizability of pulse-level VQEs](#), by Kyle M Sherbert, Hisham Amer, Sophia E Economou, Edwin Barnes, and Nicholas J Mayhall, arXiv, February 2025 (14 pages).
- [963] [Pulse-optimised circuit elements for scalable and noise-resilient quantum chemistry](#), by Henrik Gothen, Normann Mertig, and David R. M. Arvidsson-Shukur, arXiv, June 2026 (9 pages).
- [964] [DyPP: Dynamic Parameter Prediction to Accelerate Convergence of Variational Quantum Algorithms](#), by Satwik Kundu, Debarshi Kundu, and Swaroop Ghosh, arXiv, February 2024 (13 pages).
- [965] [DISQ: Dynamic Iteration Skipping for Variational Quantum Algorithms](#), by Junyao Zhang, Yiran Chen, et al., arXiv, July 2024 (12 pages).
- [966] [Measurements as a roadblock to near-term practical quantum advantage in chemistry: Resource analysis](#), by Jérôme F. Gonthier, Maxwell D. Radin, Corneliu Buda, Eric J. Daskocil, Clena M. Abuan, and Jhonathan Romero, Physical Review Research, August 2022.
- [967] [Large-Scale Simulation of Quantum Computational Chemistry on a New Sunway Supercomputer](#), by Honghui Shang, Zhenyu Li, et al., arXiv, July 2022 (13 pages).
- [968] [Calculating the ground state energy of benzene under spatial deformations with noisy quantum computing](#), by Wasil Sennane, Jean-Philip Piquemal, and Marko J. Rančić, arXiv, November 2022 (14 pages).
- [969] [Molecular Symmetry in VQE: A Dual Approach for Trapped-Ion Simulations of Benzene](#), by Joshua Goings, Luning Zhao, Jacek Jakowski, Titus Morris, and Raphael Pooser, arXiv, August 2023 (8 pages).
- [970] [Accelerating Fermionic System Simulation on Quantum Computers](#), by Qing-Song Li, Guo-Ping Guo, et al., arXiv, May 2025 (15 pages).
- [971] [The Quantum Approximate Optimization Algorithm Can Require Exponential Time to Optimize Linear Functions](#), by Francisco Chicano, Zakaria Abdelmoiz Dahi, and Gabriel Luque, arXiv, May 2025 (15 pages).
- [972] [Limitations of Quantum Approximate Optimization in Solving Generic Higher-Order Constraint-Satisfaction Problems](#), by Thorge Müller, Ajainderpal Singh, Frank K. Wilhelm, and Tim Bode, arXiv, November 2024 (10 pages).
- [973] [Measurements Number Scaling in the Quantum Approximate Optimization Algorithm for MaxCut: A Statistical Analysis](#), by Inbar Chefer, Uri Shaham, and Adi Makmal, arXiv, July 2026 (53 pages).
- [974] [The Cost of Improving the Precision of the Variational Quantum Eigensolver for Quantum Chemistry](#), by Ivana Miháliková, Matej Pivluska, Martin Plesch, Martin Friák, Daniel Nagaj, and Mojmir Šob, arXiv, November 2021 (25 pages).
- [975] [Resource-efficient utilization of quantum computers](#), by Ijaz Ahamed Mohammad, Matej Pivluska, and Martin Plesch, arXiv, May 2023 (12 pages).
- [976] [Towards Practical Quantum Variational Algorithms](#), by D. Wecker, M. B. Hastings, and M. Troyer, arXiv, September 2015 (10 pages).
- [977] [Quantum expectation-value estimation by computational basis sampling](#), by Masaya Kohda, September 2022 (19 pages).
- [978] [Pauli String Partitioning Algorithm with the Ising Model for Simultaneous Measurement](#), by Tomochika Kurita, Shintaro Sato, et al., arXiv, September 2022 (25 pages).
- [979] [The Variational Quantum Eigensolver: a review of methods and best practices](#), by Jules Tilly, Jonathan Tennyson, et al., arXiv, August 2022 (156 pages).
- [980] [How Parallel Circuit Execution Can Be Useful for NISQ Computing?](#), by Siyuan Niu and Aida Todri-Sanial, arXiv, December 2021 (6 pages).
- [981] [Towards an experimentally viable variational quantum eigensolver with superconducting qubits](#), by Ryan Babush, 2016 (18 slides).
- [982] [Variational Quantum Eigensolver explained](#), by Michal, November 2019.
- [983] [Benchmarking a wide range of optimisers for solving the Fermi-Hubbard model using the variational quantum eigensolver](#), by Benjamin D.M. Jones, Lana Mineh, and Ashley Montanaro, arXiv, January 2025 (43 pages).
- [984] [Weighted Approximate Quantum Natural Gradient for Variational Quantum Eigensolver](#), by Chenyu Shi, Vedran Dunjko, and Hao Wang, arXiv, April 2025 (17 pages).
- [985] [Many-Body Excited States with a Contracted Quantum Eigensolver](#), by Scott E. Smart, Davis M. Welakuh, and Prineha Narang, arXiv, May 2023 (13 pages).
- [986] [Folded Spectrum VQE : A quantum computing method for the calculation of molecular excited states](#), by Lila Cadi Tazi and Alex J.W. Thom, arXiv, February 2024 (17 pages).
- [987] [VQE-generated Quantum Circuit Dataset for Machine Learning](#), by Akimoto Nakayama, Keisuke Fujii, et al., arXiv, June 2023 (9 pages).
- [988] [VQE Method: A Short Survey and Recent Developments](#), by Dmitry A. Fedorov, Bo Peng, Niranjan Govind, and Yuri Alexeev, arXiv, August 2021 (23 pages).
- [989] [Ground-state energy estimation of the water molecule on a trapped-ion quantum computer](#), by Yunseong Nam, Christopher Monroe, Jungsang Kim, et al., npj Quantum Information, April 2020.
- [990] [How will quantum computers provide an industrially relevant computational advantage in quantum chemistry?](#), by V.E. Elfving, A. Bochevarov, et al., arXiv, September 2020 (20 pages).
- [991] [Quantifying the effect of gate errors on variational quantum eigensolvers for quantum chemistry](#), by Kieran Dalton, David R. M. Arvidsson-Shukur, et al., npj Quantum Information, January 2024.
- [992] [Limitations of Quantum Hardware for Molecular Energy Estimation Using VQE](#), by Abel Carreras, David Casanova, and Román Orús, arXiv, June 2025 (41 pages).
- [993] [Error Analysis of the Variational Quantum Eigensolver Algorithm](#), by Sebastian Brandhofer, Simon Devitt, and Ilia Polian, arXiv, January 2023 (6 pages).
- [994] [Benchmarking the Variational Quantum Eigensolver using different quantum hardware](#), by Amine Bentellis, Andrea Matic-Flierl, Christian B. Mendl, and Jeanette Miriam Lorenz, arXiv, May 2023 (6 pages).
- [995] [Practical limitations of quantum data propagation on noisy quantum processors](#), by Gaurav Saxena, Ahmed Shalabi, and Thi Ha Kyaw, arXiv, May 2024 (10 pages).
- [996] [Exponential Scaling Barriers for Variational Quantum Eigensolvers](#), by Manuel Hagelueken, Marco Roth, et al., arXiv, March 2026 (11 pages).
- [997] [Recent Developments in VQE: Survey and Benchmarking](#), by Taylor Harville, Rishu Khurana, Vitor F. Grizzi, and Cong Liu, arXiv, February 2026 (60 pages).
- [998] [Comparative study of adaptive variational quantum eigensolvers for multi-orbital impurity models](#), by Anirban

- Mukherjee, Yong-Xin Yao, et al., *Communications Physics*, January 2023.
- [999] [A self-consistent field approach for the variational quantum eigensolver: orbital optimization goes adaptive](#), by Aaron Fitzpatrick, Stefan Knecht, et al., arXiv, December 2022 (21 pages).
- [1000] [Overlap-ADAPT-VQE: practical quantum chemistry on quantum computers via overlap-guided compact Ansätze](#), by César Feniou, Jean-Philip Piquemal, et al., *Communications Physics*, July 2023.
- [1001] [Greedy Gradient-free Adaptive Variational Quantum Algorithms on a Noisy Intermediate Scale Quantum Computer](#), by César Feniou, Jean-Philip Piquemal, et al., arXiv, January 2025 (28 pages).
- [1002] [Classical Pre-optimization Approach for ADAPT-VQE: Maximizing the Potential of High-Performance Computing Resources to Improve Quantum Simulation of Chemical Applications](#), by J. Wayne Mullinax, Sophia E. Economou, Norm M. Tubman, et al., arXiv, November 2024 (12 pages).
- [1003] [Nuclear shell-model simulation in digital quantum computers](#), by A. Pérez-Obiol, A. M. Romero, J. Menéndez, A. Ríos, A. García-Sáez, and B. Juliá-Díaz, arXiv, September 2023 (16 pages).
- [1004] [Physically motivated improvements of Variational Quantum Eigensolvers](#), by Nonia Vaquero-Sabater, Abel Carerras, Román Orús, Nicholas J. Mayhall, and David Casanova, arXiv, September 2024 (36 pages).
- [1005] [Benchmarking Adaptive Variational Quantum Algorithms on QUBO Instances](#), by Gloria Turati, Maurizio Ferrari Dacrema, and Paolo Cremonesi Politecnico di Milano), arXiv, August 2023 (7 pages).
- [1006] [Qubit-efficient quantum chemistry with the ADAPT variational quantum eigensolver and double unitary downfolding](#), by Harjeet Singh, Nicholas J. Mayhall, et al., arXiv, April 2025 (14 pages).
- [1007] [K-ADAPT-VQE: Optimizing Molecular Ground State Searches by Chunking Operators](#), by Tatiana Bespalova, Oumaya Ladhari, and Guido Masella, arXiv, June 2025 (9 pages).
- [1008] [Resource-efficient energy-based operator selection in fermionic ADAPT-VQE via exact Hamiltonian transformation](#), by Emanuele Rossi, Erik Rosendahl Kjellgren, Artur F. Izmaylov, Stephan P.A. Sauer, Karl Michael Ziems, and Sonia Coriani, arXiv, June 2026 (32 pages).
- [1009] [Counterdiabatic ADAPT-VQE for molecular simulation](#), by Diego Tancara, Herbert Díaz-Moraga, and Dardo Goyeneche, arXiv, January 2026 (10 pages).
- [1010] [Continuous-variable ADAPT-VQE for bosonic lattice models](#), by Dimitrios Athanasakos, Gloria Tejedor-García, Jack Y. Araz, Mafalda Ramôa, Bharath Sambasivam, Sophia E. Economou, and Felix Ringer, arXiv, June 2026 (20 pages).
- [1011] [Ground state preparation of random all-to-all Hamiltonians using ADAPT-VQE](#), by Sabhyata Gupta, Bharath Sambasivam, Sophia E. Economou, and Raghav G. Jha, arXiv, June 2026 (12 pages).
- [1012] [Pre-optimization of quantum circuits, barren plateaus and classical simulability: tensor networks to unlock the variational quantum eigensolver](#), by Baptiste Anselme Martin and Thomas Ayrat, arXiv, February 2026 (7 pages).
- [1013] [Warm-Starting the VQE with Approximate Complex Amplitude Encoding](#), by Felix Truger, Johanna Barzen, Frank Leymann, and Julian Obst, arXiv, February 2024 (9 pages).
- [1014] [Warm-Starting and Quantum Computing: A Systematic Mapping Study](#), by Felix Truger, Johanna Barzen, Marvin Bechtold, Martin Beisel, Frank Leymann, Alexander Mandl, and Vladimir Yussupov, arXiv, March 2024 (29 pages).
- [1015] [Warm Starts, Cold States: Exploiting Adiabaticity for Variational Ground-States](#), by Ricard Puig, Berta Casas, Alba Cervera-Lierta, Zoë Holmes, and Adrián Pérez-Salinas, arXiv, February 2026 (24 pages).
- [1016] [Improving Variational Quantum Optimization using CVaR](#), by Panagiotis Kl. Barkoutsos, Giacomo Nannicini, Anton Robert, Ivano Tavernelli, and Stefan Woerner, *Quantum*, April 2020 (16 pages).
- [1017] [Classical Post-processing for Unitary Block Optimization Scheme to Reduce the Effect of Noise on Optimization of Variational Quantum Eigensolvers](#), by Xiaochuan Ding and Bryan K. Clark, arXiv, November 2024 (18 pages).
- [1018] [Towards chemical accuracy with shallow quantum circuits: A Clifford-based Hamiltonian engineering approach](#), by Jiace Sun, Lixue Cheng, and Weitang Li, arXiv, December 2023 (14 pages).
- [1019] [Neural network encoded variational quantum algorithms](#), by Jiaqi Miao, Chang-Yu Hsieh, and Shi-Xin Zhang, arXiv, August 2023 (4 pages).
- [1020] [Discrete Flow-Based Generative Models for Measurement Optimization in Quantum Computing](#), by Isaac L. Huidobro-Meezs, Jun Dai, and Rodrigo A. Vargas-Hernández, arXiv, September 2025 (28 pages).
- [1021] [Accelerated VQE: Parameter Recycling for Similar Recurring Problem Instances](#), by Tobias Rohe, Claudia Linnhoff-Popien, et al., arXiv, March 2025 (11 pages).
- [1022] [Multiclass Portfolio Optimization via Variational Quantum Eigensolver with Dicke State Ansatz](#), by J. V. S. Scurulim, Gabriel Mattos Langeloh, Victor Leme Beltran, and Samurá Brito, arXiv, August 2025 (14 pages).
- [1023] [Accelerated variational quantum eigensolver with joint Bell measurement](#), by Chenfeng Cao, Hiroshi Yano, and Yuya O. Nakagawa, arXiv, February 2024 (14 pages).
- [1024] [Low-depth Gaussian State Energy Estimation](#), by Gumaro Rendon and Peter D. Johnson, arXiv, September 2023 (12 pages).
- [1025] [TITAN: A Trajectory-Informed Technique for Adaptive Parameter Freezing in Large-Scale VQE](#), by Yifeng Peng, Xinyi Li, Samuel Yen-Chi Chen, Kaining Zhang, Zhiding Liang, Ying Wang, and Yuxuan Du, arXiv, September 2025 (15 pages).
- [1026] [Measurement-based infused circuits for variational quantum eigensolvers](#), by Albie Chan, Zheng Shi, Luca Dellantonio, Wolfgang Dür, and Christine A. Muschik, arXiv, June 2024 (26 pages).
- [1027] [Cyclic Variational Quantum Eigensolver: Escaping Barren Plateaus through Staircase Descent](#), by Hao Zhang and Ayush Asthana, arXiv, September 2025 (10 pages).
- [1028] [Variational Denoising for Variational Quantum Eigensolver](#), by Quoc Hoan Tran, Shinji Kikuchi, and Hirotaka Oshima, arXiv, November 2023 (6 pages).
- [1029] [Quantum-centric computation of molecular excited states with extended sample-based quantum diagonalization](#), by Stefano Barison, Javier Robledo Moreno, and Mario Motta, arXiv, November 2024 (15 pages).
- [1030] [Critical Limitations in Quantum-Selected Configuration Interaction Methods](#), by Peter Reinholdt, Jacob Kongsted, et al., arXiv, June 2025 (36 pages).
- [1031] [Inductive Construction of Variational Quantum Circuit for Constrained Combinatorial Optimization](#), by Hyakka Nakada, Kotaro Tanahashi, and Shu Tanaka, arXiv, January 2025 (13 pages).
- [1032] [Noise Resilience and Robust Convergence Guarantees for the Variational Quantum Eigensolver](#), by Mirko Legnini and Julian Berberich, arXiv, January 2026 (12 pages).
- [1033] [Parallelizing the Variational Quantum Eigensolver: From JIT Compilation to Multi-GPU Scaling](#), by Rylan Malarchick and Ashton Steed, arXiv, January 2026 (19 pages).
- [1034] [Benchmarking the Impact of Active Space Selection on the VQE Pipeline for Quantum Drug Discovery](#), by Zhi Yin, Xiaoran Li, Zhupeng Han, Shengyu Zhang, Xin Li, Zhihong Zhang, Runqing Zhang, Anbang Wang, and Xiaojin Zhang, arXiv, December 2025 (37 pages).

- [1035] [Photonic variational quantum eigensolver for NISQ-compatible quantum technology](#), by Kang-Min Hu, Min Namkung, and Hyang-Tag Lim, Nano Convergence, December 2025 (41 pages).
- [1036] [Benchmarking Quantum Simulations of the Lipkin-Meshkov-Glick Model Using Large Tensor Networks](#), by Maggie Bao and Brian J. McDermott, arXiv, July 2026 (22 pages).
- [1037] [Beyond Trotterization: Variational Product Formulas for Quantum Simulation](#), by Ibsal Assi, Michael Vogl, Meenu Kumari, and J. P. F. LeBlanc, arXiv, November 2025 (24 pages).
- [1038] [Auger Spectroscopy via Generative Quantum Eigensolver: A Quantum Approach to Molecular Excitations](#), by Kimberlee Keithley and Alan Aspuru-Guzik, arXiv, March 2026 (12 pages).
- [1039] [Simulating Many-Body Systems with a Projective Quantum Eigensolver](#), by Nicholas H. Stair and Francesco A. Evangelista, PRX Quantum, July 2021 (20 pages).
- [1040] [Optimizing Unitary Coupled Cluster Wave Functions on Quantum Hardware: Error Bound and Resource-Efficient Optimizer](#), by Martin Plazanet and Thomas Ayrat, arXiv, October 2024 (21 pages).
- [1041] [Non-Variational ADAPT algorithm for quantum simulations](#), by Ho Lun Tang, Sophia E. Economou, et al., arXiv, November 2024 (13 pages).
- [1042] [CANOE: Classically Assisted Non-Orthogonal Eigensolver](#), by Jihyeon Park, Collin C. D. Frink, and Matthew Otten, arXiv, March 2026 (25 pages).
- [1043] [Projector Quantum Variational Ansatz](#), by Thomas Dumontier, Robin Ollive, and Stephane Louise, arXiv, June 2026 (10 pages).
- [1044] [Comparative study of variations in quantum approximate optimization algorithms for the Traveling Salesman Problem](#), by Wenyang Qian, James P. Vary, et al., arXiv, July 2023 (18 pages).
- [1045] [Indirect Quantum Approximate Optimization Algorithms: application to the TSP](#), by Eric Bourreau, Gerard Fleury, and Philippe Lacomme, arXiv, November 2023 (8 pages).
- [1046] [Quantum Approximate Optimization Algorithm explained](#), by Michal, May 2020.
- [1047] [An Introduction to Quantum Optimization Approximation Algorithm](#), by Qingfeng Wang et al., December 2018 (16 pages).
- [1048] [QAOA Quantum Approximate Optimization Algorithm](#), by Peter Shor, MIT, June 2018 (25 slides).
- [1049] [Quantum Approximate Optimization Algorithm: Performance, Mechanism, and Implementation on Near-Term Devices](#), by Leo Zhou, Sheng-Tao Wang, Soonwon Choi, Hannes Pichler, and Mikhail D. Lukin, arXiv, November 2019 (10 pages).
- [1050] [Quantum Approximate Optimization of Non-Planar Graph Problems on a Planar Superconducting Processor](#), by Matthew P. Harrigan, Ryan Babbush, et al., arXiv, January 2021 (19 pages).
- [1051] [Lower bounding the MaxCut of high girth 3-regular graphs using the QAOA](#), by Edward Farhi, Sam Gutmann, Daniel Ranard, and Benjamin Villalonga, arXiv, March 2025 (14 pages).
- [1052] [Efficient hybrid variational quantum algorithm for solving graph coloring problem](#), by Dongmei Liu, Lili Yan, et al., arXiv, April 2025 (12 pages).
- [1053] [Modified Recursive QAOA for Exact Max-Cut Solutions on Bipartite Graphs: Closing the Gap Beyond QAOA Limit](#), by Eunok Bae, Hyukjoon Kwon, V Vijendran, and Soojoon Lee, arXiv, November 2024 (16 pages).
- [1054] [An Optimization-Free Recursive QAOA for the Binary Paint Shop Problem](#), by Gary J Mooney, Jedwin Villanueva, Bhaskar Roy Radhan, Joydip Ghosh, Charles D Hill, and Lloyd C L Hollenberg, arXiv, July 2025 (18 pages).
- [1055] [No quantum advantage implies improved bounds and classical algorithms for the binary paint shop problem](#), by Mark Goh, Lara Caroline Pereira dos Santos, and Matthias Sperl, arXiv, April 2026 (8 pages).
- [1056] [Iterative Quantum Algorithms for Maximum Independent Set: A Tale of Low-Depth Quantum Algorithms](#), by Lucas T. Brady and Stuart Hadfield, arXiv, November 2023 (16 pages).
- [1057] [Q-CHOP: Quantum constrained Hamiltonian optimization](#), by Michael A. Perlin, Ruslan Shaydulin, Benjamin P. Hall, Pierre Minssen, Changhao Li, Kabir Dubey, Rich Rines, Eric R. Anschuetz, Marco Pistoia, and Pranav Gokhale, arXiv, March 2024 (19 pages).
- [1058] [Quantum-enhanced Markov Chain Monte Carlo for Combinatorial Optimization](#), by Kate V. Marshall, Daniel J. Egger, Michael Garn, Francesca Schiavello, Sebastian Brandhofer, Christa Zoufal, and Stefan Woerner, arXiv, February 2026 (19 pages).
- [1059] [A scalable quantum-enhanced greedy algorithm for maximum independent set problems](#), by Elisabeth Wybo, Jami Rönkkö, Olli Hirviniemi, Jernej Rudi Finžgar, and Martin Leib, arXiv, January 2026 (12 pages).
- [1060] [A quantum feasibility preserving modeling for the min cut problem](#), by Ali Abbassi, Yann Dujardin, Eric Gourdin, Philippe Lacomme, and Caroline Prodhon, arXiv, February 2026 (19 pages).
- [1061] [Qubit-efficient and gate-efficient encodings of graph partitioning problems for quantum optimization](#), by Tristan Zaborniak, Prashanti Priya Angara, Vikram Khipple Mulligan, Hausi A. Müller, and Ulrike Stege, arXiv, April 2026 (12 pages).
- [1062] [Amplitude amplification-inspired QAOA: Improving the success probability for solving 3SAT](#), by Alexander Mandl, Johanna Barzen, Marvin Bechtold, Frank Leymann, and Karoline Wild, arXiv, March 2023 (25 pages).
- [1063] [Quantum speedups in solving near-symmetric optimization problems by low-depth QAOA](#), by Ashley Montanaro and Leo Zhou, arXiv, February 2025 (25 pages).
- [1064] [A Practically Scalable Approach to the Closest Vector Problem for Sieving via QAOA with Fixed Angles](#), by Ben Priestley and Petros Wallden, arXiv, March 2025 (17 pages).
- [1065] [Approximating Ground States of Quantum Hamiltonians with Snapshot-QAOA](#), by Reuben Tate, Stephan Eidenbenz, et al., arXiv, December 2024 (13 pages).
- [1066] [From the Quantum Approximate Optimization Algorithm to a Quantum Alternating Operator Ansatz](#), by Stuart Hadfield, Zhihui Wang, Bryan O’Gorman, Eleanor G. Rieffel, Davide Venturelli, and Rupak Biswas, arXiv, February 2019 (51 pages).
- [1067] [Evidence of scaling advantage for the quantum approximate optimization algorithm on a classically intractable problem](#), by Ruslan Shaydulin, Changhao Li, Shouvanik Chakrabarti, Matthew DeCross, Dylan Herman, Niraj Kumar, Jeffrey Larson, Danylo Lykov, Pierre Minssen, Yue Sun, Yuri Alexeev, Joan M. Dreiling, John P. Gaebler, Thomas M. Gatterman, Justin A. Gerber, Kevin Gilmore, Dan Gresh, Nathan Hewitt, Chandler V. Horst, Shaohan Hu, Jacob Johansen, Mitchell Matheny, Tanner Mingle, Michael Mills, Steven A. Moses, Brian Neyenhuis, Peter Siegfried, Romina Yalovetzky, and Marco Pistoia, Science Advances, May 2024.
- [1068] [Scaling of the quantum approximate optimization algorithm on superconducting qubit based hardware](#), by Johannes Weidenfeller, Caroline Tornow, Daniel J. Egger, et al., arXiv, December 2022 (25 pages).
- [1069] [Comparing Quantum Service Offerings](#), by Julian Obst, Felix Truger, et al., arXiv, December 2023 (4 pages).
- [1070] [QAOA with \$N \cdot p \geq 200\$](#) , by Ruslan Shaydulin and Marco Pistoia, arXiv, September 2023 (6 pages).

- [1071] [On the performance of Quantum Approximate Optimization](#), by Vishwanathan Akshay, 2023 (110 pages).
- [1072] [Parameter Setting Heuristics Make the Quantum Approximate Optimization Algorithm Suitable for the Early Fault-Tolerant Era](#), by Zichang He, Ruslan Shaydulin, Dylan Herman, Changhao Li, Rudy Raymond, Shree Hari Sureshbabu, and Marco Pistoia, arXiv, August 2024 (7 pages).
- [1073] [Error Propagation in NISQ Devices for Solving Classical Optimization Problems](#), by Guillermo González-García, PRX Quantum, December 2022 (17 pages).
- [1074] [Deep-Circuit QAOA](#), by Gereon Koßmann, Lennart Binkowski, Lauritz van Luijk, Timo Ziegler, and René Schwonnek, arXiv, February 2023 (19 pages).
- [1075] [A Review on Quantum Approximate Optimization Algorithm and its Variants](#), by Kostas Blekos, Alessandro Sumner, et al., arXiv, June 2023 (67 pages).
- [1076] [Feedback-Based Quantum Algorithm for Constrained Optimization Problems](#), by Salahuddin Abdul Rahman, Özkan Karabacak, and Rafal Wisniewski, arXiv, July 2024 (13 pages).
- [1077] [Quantum Alternating Operator Ansatz \(QAOA\) beyond low depth with gradually changing unitaries](#), by Vladimir Kremenetski, Anuj Apte, Tad Hogg, Stuart Hadfield, and Norm M. Tubman, arXiv, July 2023 (40 pages).
- [1078] [Enhancing Quantum Algorithms for Quadratic Unconstrained Binary Optimization via Integer Programming](#), by Friedrich Wagner, Jonas Nüßlein, and Frauke Liers, arXiv, April 2024 (29 pages).
- [1079] [Solution of SAT Problems with the Adaptive-Bias Quantum Approximate Optimization Algorithm](#), by Yunlong Yu, Chenfeng Cao, Xiang-Bin Wang, Nic Shannon, and Robert Joynt, arXiv, April 2023 (18 pages).
- [1080] [Investigating the effect of circuit cutting in QAOA for the MaxCut problem on NISQ devices](#), by Marvin Bechtold, Benjamin Weder, et al., arXiv, October 2023 (32 pages).
- [1081] [Parallel circuit implementation of variational quantum algorithms](#), by Michele Cattelan and Sheir Yarkoni, arXiv, April 2023 (12 pages).
- [1082] [Quantum Approximate Optimization Algorithm with Cat Qubits](#), by Pontus Vikstål, Laura García-Álvarez, Shruti Puri, and Giulia Ferrini, arXiv, September 2024 (15 pages).
- [1083] [Theory and Implementation of the Quantum Approximate Optimization Algorithm: A Comprehensive Introduction and Case Study Using Qiskit and IBM Quantum Computers](#), by Andreas Sturm, arXiv, January 2023 (114 pages).
- [1084] [Weighted Feedback-Based Quantum Algorithm for Excited States Calculation](#), by Salahuddin Abdul Rahman, Özkan Karabacak, and Rafal Wisniewski, arXiv, July 2024 (7 pages).
- [1085] [Dynamic-ADAPT-QAOA: An algorithm with shallow and noise-resilient circuits](#), by Nikola Yanakiev, Normann Mertig, Christopher K. Long, and David R. M. Arvidsson-Shukur, arXiv, August 2023 (15 pages).
- [1086] [Restricted Global Optimization for QAOA](#), by Peter Gleißner, Georg Kruse, and Andreas Roßkopf, arXiv, September 2023 (17 pages).
- [1087] [Quantum Dueling: an Efficient Solution for Combinatorial Optimization](#), by Letian Tang, Shujin Li, et al., arXiv, January 2024 (18 pages).
- [1088] [Detrimental Agnostic Entanglement: The Case Against Hardware-Efficient Ansätze for Combinatorial Optimization](#), by Tobias Rohe, Markus Baumann, Federico Harjes Ruiloba, Philipp Altmann, Gerhard Stenzel, and Claudia Linnhoff-Popien, arXiv, May 2026 (11 pages).
- [1089] [Fundamental Limitations of QAOA on Constrained Problems and a Route to Exponential Enhancement](#), by Chionso Onah and Kristel Michielsen, arXiv, November 2025 (35 pages).
- [1090] [Regularized Warm-Started Quantum Approximate Optimization and Conditions for Surpassing Classical Solvers on the Max-Cut Problem](#), by Zichang He, Anuj Apte, Brandon Augustino, Arman Babakhani, Abid Khan, Sivaprasad Omanakuttan, and Ruslan Shaydulin, arXiv, March 2026 (24 pages).
- [1091] [Efficient Fourier-Based Linear Combination of Unitaries and Applications in Quantum Optimization](#), by Almudena Carrera Vazquez, Daniel J. Egger, and Stefan Woerner, arXiv, May 2026 (18 pages).
- [1092] [Quantum optimization using a 127-qubit gate-model IBM quantum computer can outperform quantum annealers for nontrivial binary optimization problems](#), by Natasha Sachdeva, Michael J. Biercuk, Yuval Baum, et al., arXiv, October 2024 (19 pages).
- [1093] [Short-depth QAOA circuits and quantum annealing on higher-order ising models](#), by Elijah Pelofske, Andreas Bärttschi, and Stephan Eidenbenz, npj Quantum Information, March 2024 (preprint on doi.org).
- [1094] [Performance Benchmarking of Quantum Algorithms for Hard Combinatorial Optimization Problems: A Comparative Study of non-FTQC Approaches](#), by Santaro Kikuura, Ryoya Igata, Yuta Shingu, and Shohei Watabe, arXiv, October 2024 (6 pages).
- [1095] [Benchmarking Quantum Optimization for the Maximum-Cut Problem on a Superconducting Quantum Computer](#), by Maxime Dupont, Bhuvanesh Sundar, Bram Evert, David E. Bernal Neira, Zedong Peng, Stephen Jeffrey, and Mark J. Hodson, arXiv, January 2025 (35 pages).
- [1096] [Performant near-term quantum combinatorial optimization](#), by Titus D. Morris, Ananth Kaushik, Martin Roetteler, and Phillip C. Lotshaw, arXiv, October 2024 (8 pages).
- [1097] [Solving Combinatorial Optimization Problems with a Block Encoding Quantum Optimizer](#), by Adelina Bärligea, Benedikt Poggel, and Jeanette Miriam Lorenz, arXiv, March 2025 (12 pages).
- [1098] [Quantum Graph Optimization Algorithm](#), by Yuhan Huang, Ferris Prima Nugraha, Siyuan Jin, Yichi Zhang, Bei Zeng, and Qiming Shao, arXiv, April 2024 (11 pages).
- [1099] [Energy Landscapes for the Quantum Approximate Optimization Algorithm](#), by Boy Choy and David J. Wales, arXiv, May 2024 (20 pages).
- [1100] [Towards large-scale quantum optimization solvers with few qubits](#), by Marco Sciorilli, Lucas Borges, Taylor L. Patti, Diego García-Martín, Giancarlo Camilo, Anima Anandkumar, and Leandro Aolita, Nature Communications, January 2025.
- [1101] [Quantum Interference as a Proposal Mechanism for Combinatorial Optimization](#), by Donald J. Jacobs, arXiv, July 2026 (76 pages).
- [1102] [Benchmarking Classical and Quantum Optimization Approaches for Rider-Order Assignment](#), by Tharrmashastha SAPV, Surya Prakash Palanivel, Jasjyot Singh Gulati, and M Maruthu Pandi, arXiv, February 2026 (7 pages).
- [1103] [Experimental Workflows for Combinatorial Optimization: Towards Quantum Advantage](#), by Prashanti Priya Angara, Luis F. Rivera, Ulrike Stege, Hausi Müller, Ibrahim Shehzad, and Sean Wagner, arXiv, April 2026 (12 pages).
- [1104] [Hardware-Aware QAOA for HoneyPot Traffic Partitioning on 100+ Qubit IBM Quantum Processors](#), by Cameron V. Cogburn, Casimer DeCusatis, and Evan Spillane, arXiv, June 2026 (11 pages).
- [1105] [Setting angles in quantum approximate optimization at utility-scale](#), by Maosheng Guo, Joel Jurado Diaz, Anurag Ramesh, Conrad J. Haupt, Alberto Baiardi, Dimitrios Athanasakos, M. Emre Sahin, Oscar Wallis, George Pennington, Christian Arenz, Sebastian Brandhofer, Georgios Korpas, Ieva Čepaitė, J. A. Montañez-Barrera, Jakub Mareček, Davide Venturelli, Stephan Eidenbenz, David E. Bernal Neira, and Daniel J. Egger, arXiv, June 2026 (38 pages).
- [1106] [Noise-Induced Landscape Distortion in QAOA for Constrained Binary Optimization: Empirical Characterization](#)

- on IBM Quantum Hardware, by Dikran S. Meliksetian, arXiv, April 2026 (5 pages).
- [1107] [Generalized Short Path Algorithms: Towards Super-Quadratic Speedup over Markov Chain Search for Combinatorial Optimization](#), by Shouvanik Chakrabarti, Marco Pistoia, et al., arXiv, October 2024 (68 pages).
 - [1108] [Quantum and classical correlations in shrinking algorithms for optimization](#), by Victor Fischer, Lilly Palackal, Christian B. Mendl, et al., arXiv, December 2024 (12 pages).
 - [1109] [A Framework to Formulate Pathfinding Problems for Quantum Computing](#), by Damian Rovara, Nils Quetschlich, and Robert Wille, arXiv, April 2024 (10 pages).
 - [1110] [Efficient Encodings of the Travelling Salesperson Problem for Variational Quantum Algorithms](#), by Manuel Schnaus, Lilly Palackal, Benedikt Poggel, Xiomara Runge, Hans Ehm, Jeanette Miriam Lorenz, and Christian B. Mendl, arXiv, July 2024 (7 pages).
 - [1111] [Beyond QUBO and HOBQ formulations, solving the Travelling Salesman Problem on a quantum boson sampler](#), by Daniel Goldsmith and Joe Day-Evans, arXiv, June 2024 (16 pages).
 - [1112] [Out of the Loop: Structural Approximation of Optimisation Landscapes and non-Iterative Quantum Optimisation](#), by Tom Krüger and Wolfgang Mauere, arXiv, August 2024 (20 pages).
 - [1113] [Strategies for running the QAOA at hundreds of qubits](#), by Brandon Augustino, Madelyn Cain, Edward Farhi, Swati Gupta, Sam Gutmann, Daniel Ranard, Eugene Tang, and Katherine Van Kirk, arXiv, October 2024 (15 pages).
 - [1114] [Cross-Problem Parameter Transfer in Quantum Approximate Optimization Algorithm: A Machine Learning Approach](#), by Kien X. Nguyen, Bao Bach, and Ilya Safro, arXiv, April 2025 (7 pages).
 - [1115] [Diffusion-Enhanced Optimization of Variational Quantum Eigensolver for General Hamiltonians](#), by Shikun Zhang, Zhisong Xiao, et al., arXiv, January 2025 (9 pages).
 - [1116] [Exploring Entanglement and Optimization within the Hamiltonian Variational Ansatz](#), by Roeland Wiersema, PRX Quantum, December 2020.
 - [1117] [Warm-Starting QAOA with XY Mixers: A Novel Approach for Quantum-Enhanced Vehicle Routing Optimization](#), by Rafael S. do Carmo, João Paulo Papa, et al., arXiv, April 2025 (10 pages).
 - [1118] [The Lie Algebra of XY-mixer Topologies and Warm Starting QAOA for Constrained Optimization](#), by Steven Kordonowy and Hannes Leipold, arXiv, June 2025 (63 pages).
 - [1119] [Warm Start Adaptive-Bias Quantum Approximate Optimization Algorithm](#), by Yunlong Yu, Xiang-Bin Wang, Nic Shannon, and Robert Joynt, arXiv, March 2025 (17 pages).
 - [1120] [SCOOP: A Quantum-Computing Framework for Constrained Combinatorial Optimization](#), by Prashanti Priya Angara, Emily Martins, Ulrike Stege, and Hausi Müller, arXiv, April 2025 (11 pages).
 - [1121] [The Art of Avoiding Constraints: A Penalty-free Approach to Constrained Combinatorial Optimization with QAOA](#), by Prashanti Priya Angara, Danylo Lykov, Ulrike Stege, Yuri Alexeev, and Hausi Müller, arXiv, March 2025 (25 pages).
 - [1122] [One for All: Universal Quantum Conic Programming Framework for Hard-Constrained Combinatorial Optimization Problems](#), by Lennart Binkowski, Tobias J. Osborne, Marvin Schwiering, René Schwonnek, and Timo Ziegler, arXiv, November 2024 (14 pages).
 - [1123] [On the role of overparametrization in Quantum Approximate Optimization](#), by Daniil Rabinovich, Andrey Kardanashin, and Soumik Adhikary, arXiv, August 2025 (11 pages).
 - [1124] [A Quantum Approximate Optimization Algorithm for Local Hamiltonian Problems](#), by Ishaan Kannan, Robbie King, and Leo Zhou, arXiv, December 2024 (21 pages).
 - [1125] [Hierarchical Quantum Optimization via Backbone-Driven Problem Decomposition: Integrating Tabu-Search with QAOA](#), by Minhui Gou, Wei-Feng Zhuang, et al., arXiv, April 2025 (10 pages).
 - [1126] [Evidence of scaling advantage on an NP-Complete problem with enhanced quantum solvers](#), by Quanfeng Lu, Gui-Lu Long, et al., arXiv, August 2025 (13 pages).
 - [1127] [Quantum-Enhanced Optimization by Warm Starts](#), by Ieva Čepaitė, Niam Vaishnav, Leo Zhou, and Ashley Montanaro, arXiv, August 2025 (50 pages).
 - [1128] [Hierarchical divide and conquer quantum approach to combinatorial optimization problems with tunable reduction](#), by Mathias Schmid, Naeimeh Mohseni, and Michael J. Hartmann, arXiv, December 2025 (19 pages).
 - [1129] [Methods for non-variational heuristic quantum optimisation](#), by Stuart Ferguson and Petros Wallden, arXiv, February 2026 (12 pages).
 - [1130] [Divide-and-Conquer Neural Network Surrogates for Quantum Sampling: Accelerating Markov Chain Monte Carlo in Large-Scale Constrained Optimization Problems](#), by Yuya Kawamata, Yuichiro Nakano, and Keisuke Fujii, arXiv, April 2026 (12 pages).
 - [1131] [A SWAP-free Framework for QAOA](#), by Thiago Assis, Pedro Baptista, Laila Lopes, Diego Ferreira, and Gabriel Coutinho, arXiv, April 2026 (19 pages).
 - [1132] [Constraint-Aware Quantum Optimization via Hamming Weight Operators](#), by Yajie Hao, Qiming Ding, Xiao Yuan, and Xiaoting Wang, arXiv, January 2026 (11 pages).
 - [1133] [QAOA-Predictor: Forecasting Success Probabilities and Minimal Depths for Efficient Fixed-Parameter Optimization](#), by Rodrigo Coelho, Georg Kruse, and Jeanette Miriam Lorenz, arXiv, March 2026 (29 pages).
 - [1134] [Graph-Conditioned Meta-Optimizer for QAOA Parameter Generation on Multiple Problem Classes](#), by Kien X. Nguyen and Ilya Safro, arXiv, April 2026 (12 pages).
 - [1135] [Mechanism of Efficacy in QAOA for Random k-SAT: From Adiabatic Manifold to Sublinear Parameter Optimization](#), by Mingyou Wu and Hanwu Chen, arXiv, May 2026 (31 pages).
 - [1136] [Towards a universal QAOA protocol: Evidence of a scaling advantage in solving some combinatorial optimization problems](#), by J. A. Montanez-Barrera and Kristel Michielsen, arXiv, June 2024 (26 pages).
 - [1137] [Unlocking Quantum Optimization: A Use Case Study on NISQ Systems](#), by Andreas Sturm, Bharadwaj Mummaneni, and Leon Rullkötter, arXiv, April 2024 (50 pages).
 - [1138] [A Predictive Approach for Selecting the Best Quantum Solver for an Optimization Problem](#), by Deborah Volpe, Nils Quetschlich, Mariagrazia Graziano, Giovanna Turvani, and Robert Wille, arXiv, August 2024 (12 pages).
 - [1139] [Advantage for Discrete Variational Quantum Algorithms in Circuit Recompilation](#), by Oleksandr Kyriienko, Chukwudubem Umeano, and Zoë Holmes, arXiv, October 2025 (5 pages).
 - [1140] [ParaQAOA: Efficient Parallel Divide-and-Conquer QAOA for Large-Scale Max-Cut Problems Beyond 10,000 Vertices](#), by Po-Hsuan Huang, Xie-Ru Li, Chi Chuang, Chia-Heng Tu, and Shih-Hao Hung, arXiv, March 2026 (28 pages).
 - [1141] [Experimental implementation of continuous-variable QAOA on a quad-rail lattice cluster state](#), by Shota Yokoyama, Atsushi Sakaguchi, Jun-ichi Yoshikawa, Hiromari Nagayoshi, Warit Asavanant, Kan Takase, Takuji Hiraoka, Akira Furusawa, and Hidehiro Yonezawa, arXiv, June 2026 (14 pages).
 - [1142] [The Quest for Quantum Advantage in Combinatorial Optimization: End-to-end Benchmarking of Quantum Solvers vs. Multi-core Classical Solvers](#), by Pranav Chandarana, Alejandro Gomez Cadavid, Enrique Solano, Thorsten Koch, Stefan Woerner, and Narendra N. Hegade, arXiv, March 2026 (17 pages).

- [1143] [Postprocessing Variationally Scheduled Quantum Algorithm for Constrained Combinatorial Optimization Problems](#), by Tatsuhiko Shirai, IEEE, September 2023 (14 pages).
- [1144] [An in-principle super-polynomial quantum advantage for approximating combinatorial optimization problems via computational learning theory](#), by Niklas Pirnay et al., March 2024 (17 pages).
- [1145] [A Lyapunov Framework for Quantum Algorithm Design in Combinatorial Optimization with Approximation Ratio Guarantees](#), by Shengminjie Chen, Ziyang Li, Hongyi Zhou, Jialin Zhang, Wenguo Yang, and Xiaoming Sun, arXiv, December 2025 (21 pages).
- [1146] [Variational quantum algorithms for machine learning: theory and applications](#), by Stefano Mangini, arXiv, June 2023 (220 pages).
- [1147] [Numerical evidence against advantage with quantum fidelity kernels on classical data](#), by Lucas Slattery, Marco Pistoia, Stefan M. Wild, et al., arXiv, November 2022 (9 pages).
- [1148] [Benchmarking simulated and physical quantum processing units using quantum and hybrid algorithms](#), by Mohammad Kordzanganeh, Markus Pflichtsch, Alexey Melnikov, et al., arXiv, June 2023 (21 pages).
- [1149] [Quantum density estimation with density matrices: Application to quantum anomaly detection](#), by Diego H. Useche, Oscar A. Bustos-Brinez, Joseph A. Gallego-Mejia, and Fabio A. González, arXiv, March 2024 (27 pages).
- [1150] [Revisiting dequantization and quantum advantage in learning tasks](#), by Jordan Cotler, Hsin-Yuan Huang, and Jarrod R. McClean, arXiv, December 2021 (6 pages).
- [1151] [A quantum-inspired classical algorithm for recommendation systems](#), by Ewin Tang, Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, 217-228, June 2019 (12 pages) 📄 (preprint on [arXiv](#)).
- [1152] [Robust Dequantization of the Quantum Singular Value Transformation and Quantum Machine Learning Algorithms](#), by François Le Gall, Computational Complexity, January 2025.
- [1153] [Constrained Quantum Optimization for Extractive Summarization on a Trapped-ion Quantum Computer](#), by Pradeep Niroula, Marco Pistoia, et al., arXiv, October 2022 (16 pages).
- [1154] [QNLP in Practice: Running Compositional Models of Meaning on a Quantum Computer](#), by Robin Lorenz, Anna Pearson, Konstantinos Meichanetzidis, Dimitri Kartsaklis, and Bob Coecke, arXiv, May 2023 (38 pages).
- [1155] [Quantum algorithms for topological and geometric analysis of data](#), by Seth Lloyd, Silvano Garnerone, and Paolo Zanardi, Nature Communications, January 2016.
- [1156] [Topological data analysis and machine learning](#), by Daniel Leykam and Dimitris G. Angelakis, arXiv, July 2023 (15 pages).
- [1157] [Higher-order topological kernels via quantum computation](#), by Massimiliano Incudini, Francesco Martini, and Alessandra Di Pierro, arXiv, July 2023 (9 pages).
- [1158] [Quantum-Enhanced Topological Data Analysis: A Peep from an Implementation Perspective](#), by Ankit Khandelwal and M Girish Chandra, arXiv, February 2023 (10 pages).
- [1159] [Analyzing Prospects for Quantum Advantage in Topological Data Analysis](#), by Dominic W. Berry et al., Caltech, February 2024 (45 pages).
- [1160] [Complexity-Theoretic Limitations on Quantum Algorithms for Topological Data Analysis](#), by Alexander Schindhuber and Seth Lloyd, arXiv, January 2024 (19 pages).
- [1161] [Quantum computing and persistence in topological data analysis](#), by Casper Gyurik, Alexander Schindhuber, Robbie King, Vedran Dunjko, and Ryu Hayakawa, arXiv, October 2024 (21 pages).
- [1162] [Towards quantum advantage via topological data analysis](#), by Casper Gyurik, Chris Cade, and Vedran Dunjko, arXiv, October 2022 (7 pages).
- [1163] [Why measuring performance is our biggest blind spot in quantum machine learning](#), by and Maria Schuld, March 2022.
- [1164] [Architectural Patterns for Designing Quantum Artificial Intelligence Systems](#), by Mykhailo Klymenko, Liming Zhu, et al., arXiv, December 2024 (25 pages).
- [1165] [Noise-induced barren plateaus in variational quantum algorithms](#), by Samson Wang, Enrico Fontana, M. Cerezo, Kunal Sharma, Akira Sone, Lukasz Cincio, and Patrick J. Coles, Nature Communications, November 2021.
- [1166] [Cost function dependent barren plateaus in shallow parametrized quantum circuits](#), by M. Cerezo, Patrick J. Coles, et al., Nature Communications, March 2021.
- [1167] [Variational Quantum Linear Solver](#), by Carlos Bravo-Prieto, Ryan LaRose, M. Cerezo, Yigit Subasi, Lukasz Cincio, and Patrick J. Coles, Quantum, November 2023 (8 pages).
- [1168] [Memory-, Circuit-, and Ansatz-Efficient VQLS for CFD on Hybrid Quantum-HPC Systems](#), by Chao Lu, Muralikrishnan Gopalakrishnan Meena, Eduardo Antonio Coello Perez, Kalyana Chakravarthi Gottiparthi, and Seongmin Kim, arXiv, August 2026 (11 pages).
- [1169] [Variational Quantum Linear Solver via Block Encoding for the Poisson Equation](#), by Viraj Dsouza, Ayush Singhal, Alex Khan, Rut Lineswala, and Abhishek Chopra, arXiv, August 2026 (14 pages).
- [1170] [Variational Quantum Linear Solver enhanced Quantum Support Vector Machine](#), by Jianming Yi, Kalyani Suresh, Ali Moghiseh, and Norbert Wehn, arXiv, September 2023 (14 pages).
- [1171] [Quantum Singular Value Decomposer](#), by Carlos Bravo-Prieto, Diego García-Martín, and José I. Latorre, arXiv, June 2020 (1 pages).
- [1172] [Variational Quantum PageRank](#), by Christopher Sims, arXiv, April 2023 (9 pages).
- [1173] [High-Precision Variational Quantum SVD via Classical Orthogonality Correction](#), by Shohei Miyakoshi, Takanori Sugimoto, Tomonori Shirakawa, Seiji Yunoki, and Hiroshi Ueda, arXiv, May 2026 (24 pages).
- [1174] [Barren plateaus in quantum neural network training landscapes](#), by Jarrod R. McClean, Sergio Boixo, Vadim N. Smelyanskiy, Ryan Babbush, and Hartmut Neven, Nature Communications, November 2018.
- [1175] [Investigating and Mitigating Barren Plateaus in Variational Quantum Circuits: A Survey](#), by Jack Cunningham and Jun Zhuang, arXiv, February 2025 (12 pages).
- [1176] [Hamiltonian variational ansatz without barren plateaus](#), by Chae-Yeun Park and Nathan Killoran, arXiv, January 2024 (12 pages).
- [1177] [Scaling Quantum Algorithms via Dissipation: Avoiding Barren Plateaus](#), by Elias Zapusek, Ivan Rojko, and Florentin Reiter, arXiv, July 2025 (35 pages).
- [1178] [Synergistic pretraining of parametrized quantum circuits via tensor networks](#), by Manuel S. Rudolph, Jacob Miller, Danial Motlagh, Jing Chen, Atithi Acharya, and Alejandro Perdomo-Ortiz, Nature Communications, December 2023.
- [1179] [Enhancing Variational Quantum Circuit Training: An Improved Neural Network Approach for Barren Plateau Mitigation](#), by Zhehao Yi, Yanying Liang, and Haozhen Situ, arXiv, November 2024 (12 pages).
- [1180] [Universal effectiveness of high-depth circuits in variational eigenproblems](#), by Joonho Kim et al., PRR, June 2021 (12 pages).
- [1181] [Theory of overparametrization in quantum neural networks](#), by Martin Larocca, Nathan Ju, Diego García-Martín, Patrick J. Coles, and M. Cerezo, arXiv, September 2021 (16 pages).

- [1182] [Variational quantum simulation: a case study for understanding warm starts](#), by Ricard Puig, Zoë Holmes, et al., arXiv, March 2025 (34 pages).
- [1183] [Escaping from the Barren Plateau via Gaussian Initializations in Deep Variational Quantum Circuits](#), by Kaining Zhang, Liu Liu, Min-Hsiu Hsieh, and Dacheng Tao, arXiv, February 2025 (36 pages).
- [1184] [Connecting Ansatz Expressibility to Gradient Magnitudes and Barren Plateaus](#), by Zoë Holmes et al., PRX Quantum, January 2022 (23 pages).
- [1185] [Parallel variational quantum algorithms with gradient-informed restart to speed up optimisation in the presence of barren plateaus](#), by Daniel Mastropietro, Jakub Marecek, et al., arXiv, December 2024 (14 pages).
- [1186] [Do Quantum Neural Networks have Simplicity Bias?](#), by Jessica Pointing, arXiv, July 2024 (9 pages).
- [1187] [A Lie algebraic theory of barren plateaus for deep parameterized quantum circuits](#), by Michael Ragone, Marco Cerezo, et al., Nature Communications, August 2024.
- [1188] [The Adjoint Is All You Need: Characterizing Barren Plateaus in Quantum Ansätze](#), by Enrico Fontana, Dylan Herman, Shouvanik Chakrabarti, Niraj Kumar, Romina Yalovetzky, Jamie Heredge, Shree Hari Sureshbabu, and Marco Pistoia, arXiv, March 2024 (39 pages).
- [1189] [Absence of Barren Plateaus in Quantum Convolutional Neural Networks](#), by Arthur Pesah et al., Physical Review X, October 2021 (26 pages).
- [1190] [Noise-induced shallow circuits and the absence of barren plateaus](#), by Antonio Anna Mele, Armando Angrisani, Soumik Ghosh, Sumeet Khatri, Jens Eisert, Daniel Stilck França, and Yihui Quek, Nature Physics 22, 751-756, April 2026 (6 pages).
- [1191] [Analyzing the Quantum Approximate Optimization Algorithm: Ansätze, Symmetries, and Lie Algebras](#), by Sujay Kazi, Martín Larocca, Marco Farinati, Patrick J. Coles, M. Cerezo, and Robert Zeier, PRX Quantum 6, November 2025.
- [1192] [QAOA-MaxCut has barren plateaus for almost all graphs](#), by Rui Mao, Pei Yuan, Jonathan Allcock, and Shengyu Zhang, arXiv, December 2025 (43 pages).
- [1193] [Exponentially many initializations to avoid barren plateaus](#), by Ankit Kulshrestha, Ricard Puig, Diego García-Martín, Lukasz Cincio, Ilya Safro, Zoë Holmes, and M. Cerezo, arXiv, June 2026 (45 pages).
- [1194] [Barren Plateaus Beyond Observable Concentration](#), by Zi-Shen Li, Bujiao Wu, Xiao-Wei Li, and Man-Hong Yung, arXiv, March 2026 (12 pages).
- [1195] [Defeating Barren Plateaus with Task-Aligned Symmetry](#), by Ruipeng Xing, Yanan Li, Zhen Shang, Shengbin Wang, Yongjian Gu, and Zhimin Wang, arXiv, July 2026 (8 pages).
- [1196] [Mitigating Noise-Induced Barren Plateaus Using a Non-Unitary Ansatz: Application to Molecular Electronic Transport](#), by Sasanka Dowarah, Abeda Sultana Shamma, Yazdan Maghsoud, G. Andrés Cisneros, and Michael Kolodrubetz, arXiv, May 2026 (29 pages).
- [1197] [Strategies for Overcoming Gradient Troughs in the ADAPT-VQE Algorithm](#), by Jonas Stadelmann, Julian Übelher, Mafalda Ramôa, Bharath Sambasivam, Edwin Barnes, and Sophia E. Economou, arXiv, December 2025 (18 pages).
- [1198] [Comprehensive Numerical Studies of Barren Plateau and Overparametrization in Variational Quantum Algorithm](#), by Himuro Hashimoto, Akio Nakabayashi, Lento Nagano, Yutaro Iiyama, Ryu Sawada, Junichi Tanaka, and Koji Terashi, arXiv, February 2026 (16 pages).
- [1199] [Efficient quantum state preparation of multivariate functions using tensor networks](#), by Marco Ballarin, Juan José García-Ripoll, David Hayes, and Michael Lubasch, arXiv, November 2025 (8 pages).
- [1200] [Cautious optimism for deep parameterized quantum circuits](#), by Marie Kempkes, Elies Gil-Fuster, Carlos Bravo-Prieto, Aroosa Ijaz, Alissa Wilms, Jens Eisert, Evert van Nieuwenburg, and Vedran Dunjko, arXiv, July 2026 (21 pages).
- [1201] [The Complexity of Dynamical Correlators: Operator Shadows and Exponential Learning Separations](#), by Shao-Hen Chiew, Armando Angrisani, and Zoe Holmes, arXiv, July 2026 (38 pages).
- [1202] [Practical quantum advantage in quantum simulation](#), by Andrew J. Daley, Matthias Troyer, Peter Zoller, et al., Nature, July 2022 (preprint on [strathprints.strath.ac.uk](#)).
- [1203] [Efficiently verifiable quantum advantage on near-term analog quantum simulators](#), by Zhenning Liu, Andrew M. Childs, et al., arXiv, March 2024 (20 pages).
- [1204] [Accuracy guarantees and quantum advantage in analogue open quantum simulation with and without noise](#), by Vikram Kashyap, Rahul Trivedi, et al., arXiv, November 2024 (65 pages).
- [1205] [Scaling Advantage in Approximate Optimization with Quantum Annealing](#), by Humberto Munoz Bauza and Daniel A. Lidar, arXiv, January 2024 (11 pages).
- [1206] [Combinatorial Optimization on Gate Model Quantum Computers: A Survey](#), by Ehsan Zahedinejad and Arman Zaribafian, arXiv, August 2017 (19 pages).
- [1207] [Five Starter Problems: Solving Quadratic Unconstrained Binary Optimization Models on Quantum Computers](#), by Arul Mazumder and Sridhar Tayur, arXiv, June 2025 (47 pages).
- [1208] [Learning-Driven Annealing with Adaptive Hamiltonian Modification for Solving Large-Scale Problems on Quantum Devices](#), by Sebastian Schulz, Dennis Willsch, and Kristel Michielsen, arXiv, February 2025 (20 pages).
- [1209] [Variational Quantum Annealing for Quantum Chemistry](#), by Ka-Wa Yip, Kübra Yeter-Aydeniz, and Sijia S. Dong, arXiv, March 2025 (18 pages).
- [1210] [Optimized QUBO formulation methods for quantum computing](#), by Dario De Santis, Salvatore Tirone, Stefano Marmi, and Vittorio Giovannetti, arXiv, June 2024 (31 pages).
- [1211] [Towards an Automatic Framework for Solving Optimization Problems with Quantum Computers](#), by Deborah Volpe, Nils Quetschlich, Mariagrazia Graziano, Giovanna Turvani, and Robert Wille, arXiv, June 2024 (12 pages).
- [1212] [Improving Performance in Combinatorial Optimization Problems with Inequality Constraints: An Evaluation of the Unbalanced Penalization Method on D-Wave Advantage](#), by J. A. Montanez-Barrera, Pim van den Heuvel, Dennis Willsch, and Kristel Michielsen, arXiv, May 2023 (8 pages).
- [1213] [Quantum Annealing and Graph Neural Networks for Solving TSP with QUBO](#), by Haoqi He, arXiv, February 2024 (9 pages).
- [1214] [Solving \(Max\) 3-SAT via Quadratic Unconstrained Binary Optimization](#), by Jonas Nüßlein, Sebastian Zielinski, Thomas Gabor, Claudia Linnhoff-Popien, and Sebastian Feld, arXiv, February 2023 (14 pages).
- [1215] [Influence of Different 3SAT-to-QUBO Transformations on the Solution Quality of Quantum Annealing: A Benchmark Study](#), by Sebastian Zielinski, Sebastian Feld, et al., arXiv, May 2023 (9 pages).
- [1216] [Transformation-Dependent Performance-Enhancement of Digital Annealer for 3-SAT](#), by Christian Münch, Fritz Schinkel, Sebastian Zielinski, and Stefan Walter, arXiv, December 2023 (10 pages).
- [1217] [Solving Max-3SAT Using QUBO Approximation](#), by Sebastian Zielinski, Sebastian Feld, et al., arXiv, September 2024 (11 pages).
- [1218] [Accuracy and Performance Evaluation of Quantum, Classical and Hybrid Solvers for the Max-Cut Problem](#), by Jaka Vodeb, Vid Eržen, Timotej Hrga, and Janez Povh, arXiv, December 2024 (19 pages).

- [1219] [Simon's Period Finding on a Quantum Annealer](#), by Reece Robertson, Emery Doucet, Zakaria Mzaouali, Krzysztof Domino, Bartłomiej Gardas, and Sebastian Deffner, arXiv, April 2025 (6 pages).
- [1220] [Analog QAOA with Bayesian Optimisation on a neutral atom QPU](#), by Simone Tibaldi, Lucas Leclerc, Davide Vodola, Edoardo Tignone, and Elisa Ercolessi, arXiv, January 2025 (12 pages).
- [1221] [Quantum Annealing for Robust Principal Component Analysis](#), by Ian Tomeo, Andreas Savakis, et al., arXiv, January 2025 (20 pages).
- [1222] [Optimizing QUBO generation parameters for NP problems and their impact on D-Wave convergence](#), by Toru Fujii, Koshi Komuro, and Kaito Tomari, arXiv, April 2025 (22 pages).
- [1223] [Solving job-shop scheduling problems with QUBO-based specialized hardware](#), by Jiachen Zhang, 2022 (9 pages).
- [1224] [Quantum algorithm for process parallel flexible job shop scheduling](#), by Berend Denkena, 2021 (15 pages).
- [1225] [Solving flexible job shop scheduling problems in manufacturing with Quantum Annealing](#), by Philipp Schworm, 2023 (11 pages).
- [1226] [Solving the Job Shop Scheduling Problem: QUBO model and Quantum Annealing](#), by Riad Aggoune et al., March 2023 (4 pages).
- [1227] [Quadratic Continuous Quantum Optimization](#), by Sascha Mücke, Thore Gerlach, and Nico Piatkowski, 2025 IEEE International Conference on Quantum Artificial Intelligence, pages 216-221, December 2025 (6 pages).
- [1228] [QUBO formulation for the Snake-in-the-box and Coil-in-the-box problems](#), by Federico Fuidio, Eduardo Canale, and Rafael Sotelo, arXiv, September 2024 (10 pages).
- [1229] [Quantum Annealing for Industry Applications: Introduction and Review](#), by Sheir Yarkoni, Elena Raponi, Thomas Bäck, and Sebastian Schmitt, arXiv, June 2022 (43 pages).
- [1230] [Hybrid Meta-Solving for Practical Quantum Computing](#), by Domenik Eichhorn, Jeanette Miriam Lorenz, et al., arXiv, May 2024 (11 pages).
- [1231] [Quantum Annealing Solutions for the Closest String Problem with D-Wave Systems](#), by Chandeepta Dissanayake, arXiv, October 2023 (10 pages).
- [1232] [Modified Multiple Sequence Alignment Algorithm on Quantum Annealers \(MAQ\)](#), by Melody Lee, arXiv, March 2024 (12 pages).
- [1233] [Quantum Annealing Algorithms for Estimating Ising Partition Functions](#), by Haowei Li, Zhiyuan Yao, and Xingze Qiu, arXiv, April 2025 (8 pages).
- [1234] [Quantum Approximate Optimization Algorithm: Performance, Mechanism, and Implementation on Near-Term Devices](#), by Leo Zhou et al., PRX, 2020 (23 pages).
- [1235] [Accelerating Grover Adaptive Search: Qubit and Gate Count Reduction Strategies with Higher-Order Formulations](#), by Yuki Sano, Kosuke Mitarai, Naoki Yamamoto, and Naoki Ishikawa, arXiv, May 2024 (11 pages).
- [1236] [Boosting quantum annealing performance through direct polynomial unconstrained binary optimization](#), by Sebastian Nagies, Michael Johanning, Philipp Hauke, et al., arXiv, April 2025 (18 pages).
- [1237] [Resource-efficient quantum optimization via higher-order encoding](#), by Frederik Koch, Shahram Panahian, Rick Mukherjee, Joseph Doetsch, and Dieter Jaksch, EPJ Quantum Technology, May 2026 (24 pages).
- [1238] [Applying a Grover-mixer quantum alternating-operator-ansatz algorithm to higher-order unconstrained binary optimization problems](#), by Evgeniy O. Kiktenko, Elizaveta V. Krendeleva, and Aleksey K. Fedorov, Physical Review A, May 2026 (13 pages) (preprint on arXiv).
- [1239] [Encoding Matters: Benchmarking Binary and D-ary Representations for Quantum Combinatorial Optimization](#), by Shashank Sanjay Bhat, Peiyong Wang, Joseph West, and Udaya Parampalli, arXiv, February 2026 (39 pages).
- [1240] [All-valid-state HOBQ encoding for constrained combinatorial optimization on NISQ devices](#), by Juncheng Wang, Jun-ichi Shirakashi, Tetsuo Shibuya, and Hiroshi Imai, arXiv, June 2026 (14 pages).
- [1241] [Quantum Hardware-Efficient Selection of Auxiliary Variables for QUBO Formulations](#), by Damian Rovara, Lukas Burgholzer, and Robert Wille, arXiv, November 2025 (7 pages).
- [1242] [Quantum Optimization in Loc\(Q\)ation Science: QUBO Formulations, Benchmark Problems, and a Computational Study](#), by Felix P. Broesamle and Stefan Nickel, arXiv, February 2026 (45 pages).
- [1243] [Achieving Sub-Exponential Speedup in Gate-Based Quantum Computing for Quadratic Unconstrained Binary Optimization](#), by Ying-Wei Tseng and Wen-Zhi Zhang, arXiv, February 2026 (12 pages).
- [1244] [Navigating Quantum Missteps in Agent-Based Modeling: A Schelling Model Case Study](#), by C. Nico Barati, Arie Croitoru, Ross Gore, Michael Jarret, William Kennedy, Andrew Maciejunes, Maxim A. Malikov, and Samuel S. Mendelson, arXiv, November 2025 (24 pages).
- [1245] [Neural Networks for Programming Quantum Annealers](#), by Samuel Bosch, Bobak Kiani, Rui Yang, Adrian Lupascu, and Seth Lloyd, arXiv, August 2023 (15 pages).
- [1246] [Dynamic Portfolio Optimization with Real Datasets Using Quantum Processors and Quantum-Inspired Tensor Networks](#), by Samuel Mugel, Enrique Lizaso, Roman Orus, et al., arXiv, December 2021 (13 pages).
- [1247] [Hybrid Quantum Investment Optimization with Minimal Holding Period](#), by Samuel Mugel, Enrique Lizaso, Roman Orus, et al., arXiv, December 2021 (6 pages).
- [1248] [Comparing Classical-Quantum Portfolio Optimization with Enhanced Constraints](#), by Salvatore Certo, Anh Dung Pham, and Daniel Beaulieu, arXiv, March 2022 (6 pages).
- [1249] [Financial Index Tracking via Quantum Computing with Cardinality Constraints](#), by Samuel Palmer, Roman Orus, Harish Naik, Samuel Mugel, et al., arXiv, August 2022 (8 pages).
- [1250] [Finding the Optimal Currency Composition of Foreign Exchange Reserves with a Quantum Computer](#), by Martin Vesely, arXiv, March 2023 (30 pages).
- [1251] [Probing topological spin liquids on a programmable quantum simulator](#), by G. Semechin, December 2021 (21 pages) (preprint on arXiv).
- [1252] [Financial Risk Management on a Neutral Atom Quantum Processor](#), by Lucas Leclerc, Didier M'tamon, et al., arXiv, April 2024 (17 pages).
- [1253] [Quantum evolution kernel: Machine learning on graphs with programmable arrays of qubits](#), by Louis-Paul Henry, Physical Review A, September 2021 (preprint on arXiv).
- [1254] [Circumventing Traps in Analog Quantum Machine Learning Algorithms Through Co-Design](#), by Rodrigo Araiza Bravo, Susanne F. Yelin, et al., arXiv, August 2024 (10 pages).
- [1255] [Quantum Computing Dataset of Maximum Independent Set Problem on King's Lattice of over Hundred Rydberg Atoms](#), by Kangheun Kim, Minhyuk Kim, Juyoung Park, Andrew Byun, and Jaewook Ahn, arXiv, November 2023 (10 pages).
- [1256] [Mean-Field Approximate Optimization Algorithm](#), by Aditi Misra-Spieldenner, Frank K. Wilhelm, et al., arXiv, September 2023 (17 pages).
- [1257] [Quantum Annealing for Combinatorial Optimization: A Benchmarking Study](#), by Seongmin Kim, Sang-Woo Ahn, In-Saeng Suh, Alexander W. Dowling, Eungkyu Lee, and Tengfei Luo, arXiv, April 2025 (21 pages).
- [1258] [Going Beyond Gadgets: The Importance of Scalability for Analogue Quantum Simulators](#), by Dylan Harley, Daniel

- Stilck França, Albert H. Werner, Matthias Christandl, et al., arXiv, August 2024 (59 pages).
- [1259] [Analog Errors in Quantum Annealing: Doom and Hope](#), by Adam Pearson, Anurag Mishra, Itay Hen, and Daniel Lidar, arXiv, July 2019 (16 pages).
 - [1260] [Post-Error Correction for Quantum Annealing Processor using Reinforcement Learning](#), by Tomasz Śmierczalski, Bartłomiej Gardas, et al., arXiv, March 2022 (14 pages).
 - [1261] [Boosting the Performance of Quantum Annealers using Machine Learning](#), by Jure Brencelj, Dragan Mihailović, Viktor Kabanov, Ljupčo Todorovski, Sašo Džeroski, and Jaka Vodeb, arXiv, March 2022 (14 pages).
 - [1262] [Solving optimization problems with local light shift encoding on Rydberg quantum annealers](#), by Kapil Goswami, Rick Mukherjee, Herwig Ott, and Peter Schmelcher, arXiv, December 2023 (18 pages).
 - [1263] [Approximate combinatorial optimization with Rydberg atoms: the barrier of interpretability](#), by Christian de Correia, Thomas Ayrat, and Corentin Bertrand, arXiv, July 2025 (21 pages).
 - [1264] [Erratum: Lower Bounds on Quantum Annealing Times \[Phys. Rev. Lett. 130, 140601 \(2023\)\]](#), by Luis Pedro García-Pintos, Lucas T. Brady, Jacob Bringewatt, and Yi-Kai Liu, Physical Review Letters, June 2024 (3 pages).
 - [1265] [Tighter Lower Bounds on Quantum Annealing Times](#), by Luis Pedro García-Pintos, Mrunmay Sahasrabudhe, and Christian Arenz, arXiv, October 2024 (9 pages).
 - [1266] [Quantum annealing versus classical solvers: Applications, challenges and limitations for optimisation problems](#), by Finley Alexander Quinton, Hongyu Zhang, et al., arXiv, April 2025 (17 pages).
 - [1267] [Scaling Advantage in Approximate Optimization with Quantum Annealing](#), by Humberto Munoz-Bauza and Daniel Lidar, Physical Review Letters, April 2025 (8 pages).
 - [1268] [Digital-analog quantum computation](#), by Adrian Parra-Rodriguez et al., January 2019 (12 pages).
 - [1269] [Tight bound for the total time in digital-analog quantum computation](#), by Mikel Garcia-de-Andoin and Mikel Sanz, arXiv, December 2025 (7 pages).
 - [1270] [Mitigating noise in digital and digital-analog quantum computation](#), by Paula García-Molina, Ana Martin, Mikel Garcia de Andoin, and Mikel Sanz, Communications Physics, October 2024.
 - [1271] [Digital-Analog Quantum Machine Learning](#), by Lucas Lamata, arXiv, November 2024 (4 pages).
 - [1272] [Digital-Analog Co-Design of the Harrow-Hassidim-Lloyd Algorithm](#), by Ana Martin et al., June 2023 (7 pages)  (preprint on arXiv).
 - [1273] [Digital-Analog Quantum Computing and Algorithms](#), by Ana Martin, arXiv, January 2024 (141 pages).
 - [1274] [Superconducting Circuit Architecture for Digital-Analog Quantum Computing](#), by J. Yu, J. C. Retamal, M. Sanz, E. Solano, and F. Albarrán-Arriagada, arXiv, May 2022 (15 pages).
 - [1275] [Mitigating noise in digital and digital-analog quantum computation](#), by Paula García-Molina, Ana Martin, Mikel Garcia de Andoin, and Mikel Sanz, arXiv, October 2024 (12 pages).
 - [1276] [Enhancing Quantum Annealing in Digital-Analog Quantum Computing](#), by Tadashi Kadowaki, arXiv, April 2024 (9 pages).
 - [1277] [Digitized-counterdiabatic quantum factorization](#), by Narendra N. Hegade and Enrique Solano, arXiv, January 2023 (3 pages).
 - [1278] [Hybrid Gate-Based and Annealing Quantum Computing for Large-Size Ising Problems](#), by Chen-Yu Liu and Hsi-Sheng Goan, arXiv, August 2022 (14 pages).
 - [1279] [Analog-Digital Quantum Computing with Quantum Annealing Processors](#), by Rahul Deshpande, Majid Kheirkhah, Chris Rich, Richard Harris, Jack Raymond, Emile Hoskinson, Pratik Sathe, Andrew J. Berkley, Stefan Paul, Brian Barch, Daniel A. Lidar, Markus Müller, Gabriel Aeppli, Andrew D. King, and Mohammad H. Amin, arXiv, March 2026 (26 pages).
 - [1280] [Digital-Analog-Digital Quantum Supremacy](#), by Daniel A. Lidar, arXiv, December 2025 (10 pages).
 - [1281] [Constant Depth Digital-Analog Counterdiabatic Quantum Computing](#), by Balaganchi A. Bhargava, Shubham Kumar, Anne-Maria Visuri, Paolo A. Erdman, Enrique Solano, and Narendra N. Hegade, arXiv, January 2026 (15 pages).
 - [1282] [Hybrid digital-analog protocols for simulating quantum multi-body interactions](#), by Or Katz, Alexander Schuckert, Tianyi Wang, Eleanor Crane, Alexey V. Gorshkov, and Marko Cetina, arXiv, December 2025 (22 pages).
 - [1283] [Numerically Optimizing Shortcuts to Adiabaticity: A Hybrid Control Strategy](#), by Bo Xing, Jesús G. Parejo, Sofía Martínez-Garaot, Paola Cappellaro, and Mikel Palmero, arXiv, April 2026 (24 pages).
 - [1284] [Shortcuts to adiabaticity: Concepts, methods, and applications](#), by D. Guéry-Odelin, Reviews of Modern Physics, October 2019  (preprint on hdl.handle.net).
 - [1285] [Taming quantum systems: A tutorial for using shortcuts-to-adiabaticity, quantum optimal control, and reinforcement learning](#), by Callum W. Duncan, Pablo M. Poggi, Marin Bukov, Nikolaj Thomas Zinner, and Steve Campbell, arXiv, January 2025 (62 pages).
 - [1286] [Digitized Counter-Diabatic Quantum Optimization for Bin Packing Problem](#), by Ruoqian Xu, Sebastián V. Romero, Jialiang Tang, Yue Ban, and Xi Chen, arXiv, February 2025 (12 pages).
 - [1287] [Digitized Counterdiabatic Quantum Algorithms for Logistics Scheduling](#), by Archismita Dalal, Gernot Füchsel, et al., arXiv, February 2025 (13 pages).
 - [1288] [Quantum Circuit Cutting: Complexity and Optimization](#), by Yuval Idan, Eitan Zahavi, Elad Mentovich, Eliahu Cohen, and Shmuel Zaks, arXiv, April 2026 (20 pages).
 - [1289] [Decomposition of Multi-Qubit Gates for Circuit Cutting](#), by Ryota Tamura, Tomoya Kashimata, Yohei Hamakawa, Kosuke Tatsumura, and Hiroshi Imai, arXiv, March 2026 (8 pages).
 - [1290] [Scalable Circuit Cutting: A Framework for Combined Gate and Wire Cuts Using Gate Groups](#), by Fiona Jiali Fröhler, Yannick Stade, Christian Ufrecht, Daniel D. Scherer, and Robert Wille, arXiv, August 2026 (7 pages).
 - [1291] [Entanglement geometry separates circuit cutting, classical hardness, and trainability](#), by Maria Gragera Garces, Sabina Drăgoi, and Lirandë Pira, arXiv, July 2026 (4 pages).
 - [1292] [Understanding the Scalability of Circuit Cutting Techniques for Practical Quantum Applications](#), by Songqinghao Yang and Prakash Murali, arXiv, November 2024 (12 pages).
 - [1293] [Divide and Conquer for Combinatorial Optimization and Distributed Quantum Computation](#), by Teague Tomesh, Margaret Martonosi, et al., arXiv, October 2023 (12 pages).
 - [1294] [TensorQC: Towards Scalable Distributed Quantum Computing via Tensor Networks](#), by Wei Tang and Margaret Martonosi, arXiv, February 2025 (14 pages).
 - [1295] [Gate Teleportation vs Circuit Cutting in Distributed Quantum Computing](#), by Shobhit Gupta, Nikolay Sheshko, Daniel J. Dilley, Alvin Gonzales, Manish K. Singh, and Zain H. Saleem, arXiv, October 2025 (9 pages).
 - [1296] [Scalable Quantum Neural Networks for Classification](#), by Jindi Wu, Zeyi Tao, and Qun Li, arXiv, August 2022 (11 pages).
 - [1297] [Enabling Multi-programming Mechanism for Quantum Computing in the NISQ Era](#), by Siyuan Niu and Aida Todri-Sanial, arXiv, February 2023 (34 pages).
 - [1298] [Distributed Variational Quantum Linear Solver](#), by Chao Lu, Pooja Rao, Muralikrishnan Gopalakrishnan Meena,

- and Kalyana Chakaravarthi Gottiparthi, arXiv, April 2026 (11 pages).
- [1299] [Distributed Variational Quantum Linear Solver](#), by Tong Shen, Zeru Zhu, and Ji Liu, arXiv, April 2026 (13 pages).
- [1300] [Performance Analysis of QAOA Across Distributed Quantum Network Topologies Using SwitchQNet](#), by Samanvay Sharma and Rodney Van Meter, arXiv, July 2026 (7 pages).

This **NISQ algorithms** subsection contains 403 bibliographical references and notes containing at least 7,621 pages.

1.5 Quantum speedups

One key potential benefit from quantum computing is with quantum speedups. In complexity theories, theoretical asymptotic performance gains of quantum algorithms, whether exponential, superpolynomial, polynomial, quadratic, and constant refer to asymptotic scalings. They depend on the definition of the input size and on the computational or query model being compared. Also, they should embed the cost of encoding the data used by the circuit in an end-to-end analysis. We have seen in multiple other parts of this book on algorithms that there is a difference between an asymptotic advantage and a practical one. On existing NISQ or future FTQC quantum hardware, what counts for an end-user is at which problem size is a quantum solution faster than the best possible classical one, and at what cost.

1.5.1 Complexity class vs speedup gains

For an explicit vector of N complex values, a classical fast Fourier transform typically requires $O(N \log N)$ arithmetic operations. A quantum Fourier transform (QFT) on an n -qubit state, with $N = 2^n$ amplitudes, can be implemented exactly with $O(n^2)$ elementary controlled rotations, or approximately with fewer gates. However, the QFT acts on amplitudes and does not output all N Fourier coefficients. Therefore, comparing $O(N \log N)$ directly with $O((\log N)^2)$ does not by itself establish an end-to-end exponential algorithmic speedup [1301]. Shor’s factoring algorithm is a separate result whose total gate complexity is polynomial in the input bit length [31].

You can also visualize graphically how computing time grows for several common asymptotic functions in Figure 69. It is using O which gives an asymptotic upper bound. There are also big Ω for asymptotic lower bound, and big Θ for asymptotically tight bound. Small o denotes a strict asymptotic upper bound. Worst-case, best-case, and average-case complexity are different notions: each of those running-time functions can itself be described using O , Ω , or Θ notation.

A computationally useful quantum algorithm seeks an advantage in an end-to-end resource such as time, number of queries, memory, energy, or solution quality relative to the best relevant classical method. In this subsection, the main focus is time complexity and time to solution from a theoretical standpoint, and not a practical one.

A change from a complexity polynomial in N to one polynomial in $\log N$ is exponential when the true input size is $n = \log_2 N$, provided that the compared algorithms solve the same task under comparable access and output assumptions. This distinction is important for quantum subroutines such as QFT, phase estimation, HHL-type linear solvers, and quantum machine-learning algorithms, where state preparation and readout can

dominate the end-to-end complexity [1301, 1302]. The time scales are more meaningful in Table 8.

The classic example is Shor’s algorithm. With N being the integer to factor, its bit length $n = \lceil \log_2 N \rceil$ is the relevant input size. Shor’s algorithm factors N in quantum polynomial time in n [31]. The best known classical general-purpose factoring algorithm, the general number field sieve, has heuristic subexponential complexity of the form $\exp(O(n^{1/3}(\log n)^{2/3}))$ [1303]. Thus, relative to the best known classical factoring methods, Shor provides a superpolynomial but not strictly exponential speedup when complexity is expressed in the input bit length n .

Deutsch-Jozsa provides an exponential separation in deterministic black-box query complexity for its promised Boolean-function problem. One quantum oracle query is enough, when an exact deterministic classical algorithm may require exponentially many queries in the input bit length. This does not imply a constant end-to-end running time for an explicitly constructed oracle. The implementation cost of the oracle and any data-access mechanism must also be included, particularly if it relies on some form of qRAM which may have a significant space and time cost.

The practical value of a speedup depends on both the asymptotic separation and the crossover problem size. A polynomial improvement can be decisive at useful scales, while even a superpolynomial separation can remain impractical if the quantum prefactor or required problem size is too large.

1.5.2 Non-Clifford gates

The gate set is one of the resources that determines both classical simulability and fault-tolerant cost (Figure 70, Figure 71). Circuits using only Clifford gates and Pauli-basis measurements can be simulated efficiently on a classical computer [699]. Adding a suitable non-stabilizer resource, often described as magic or contextuality, such as a T -type magic state, makes the computation universal [1305]. This does not mean that every exponential quantum speedup requires magic or contextuality. Shor and general QPE-based algorithms use controlled implementations of problem-specific unitaries and, for nontrivial scalable instances, generally require non-Clifford resources when compiled fault tolerantly. Grover’s oracle and diffusion operator generally contain multi-controlled phase operations whose implementation cost must be included.

1.5.3 Maximally entangled states

Entanglement is an important computational resource, but maximal N -qubit entanglement is neither a general necessary nor a sufficient condition for exponential quantum speedup. For pure-state circuit computations, Jozsa and Linden showed that an exponential speedup requires

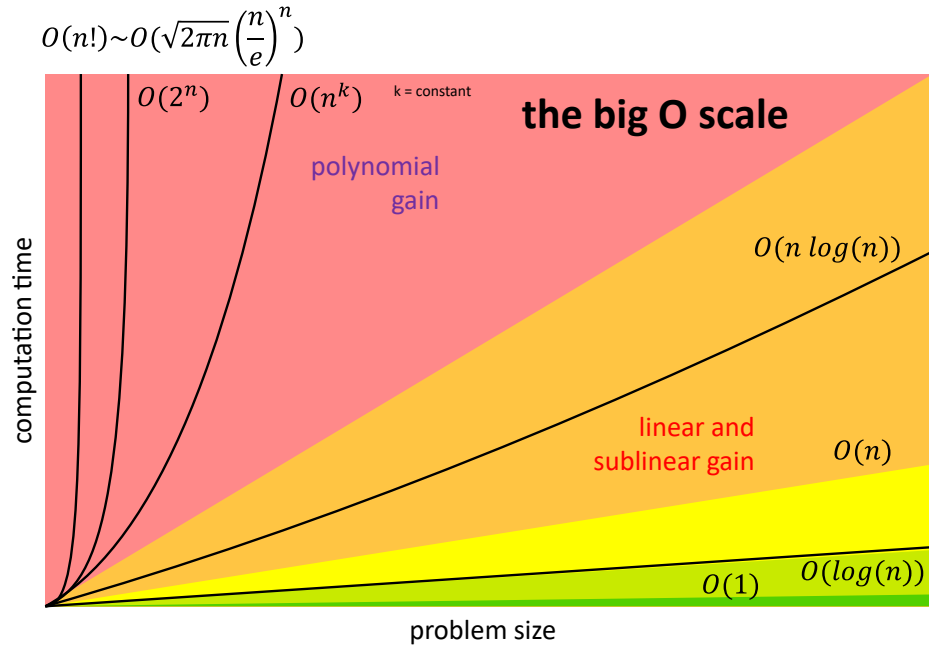


Figure 69: Another view of common asymptotic growth functions. The factorial growth $n!$ is asymptotically described by Stirling’s formula, $n! \sim \sqrt{2\pi n} (n/e)^n$. Such a running time grows faster than c^n for every fixed constant c , but this property alone does not determine what quantum speedup is possible for a particular problem. Source: Wikipedia, reformatted with additions by Olivier Ezratty, © 2022-2026.

Complexity	n	$n \log_2 n$	n^2	n^3	1.5^n	2^n	$n!$
$n = 10$	< 1 s	< 1 s	< 1 s	< 1 s	< 1 s	< 1 s	4 s
$n = 30$	< 1 s	< 1 s	< 1 s	< 1 s	< 1 s	18 min	8.4×10^{18} y
$n = 50$	< 1 s	< 1 s	< 1 s	< 1 s	11 min	36 y	9.6×10^{50} y
$n = 100$	< 1 s	< 1 s	< 1 s	1 s	12,900 y	4.0×10^{16} y	3.0×10^{144} y
$n = 1\text{K}$	< 1 s	< 1 s	1 s	17 min	10^{163} y	10^{288} y	10^{2554} y
$n = 10\text{K}$	< 1 s	< 1 s	2 min	12 d	10^{1747} y	10^{2997} y	10^{35646} y
$n = 100\text{K}$	< 1 s	2 s	3 h	32 y	10^{17596} y	10^{30090} y	10^{456560} y
$n = 1\text{M}$	1 s	20 s	12 d	31,689 y	10^{176078} y	10^{301016} y	$10^{5565695}$ y

Table 8: Complexity classes and time scales. The numerical values assume approximately 10^6 elementary operations per second and are illustrative rather than hardware-independent complexity bounds. Source for the complexity discussion: [1304].

multipartite entanglement involving an unbounded number of subsystems, and Vidal showed that computations with sufficiently restricted entanglement can be simulated efficiently classically [701, 1306]. If a pure-state computation factorizes permanently into independent blocks of at most k qubits, a direct state-vector simulation can exploit that factorization and store only $O((N/k)2^k)$ amplitudes rather than 2^N . This illustrates why bounded entanglement can aid classical simulation, but Hilbert-space dimension alone does not quantify quantum speedup.

Multipartite entanglement has many inequivalent structures and measures [1307]. An absolutely maximally entangled (AME) state is a pure multipartite state whose reductions are maximally mixed for every bipartition

in which the smaller subsystem contains at most half of the parties [1308]. Other work studies how much and what kind of entanglement is required by quantum algorithms [1309], and how multipartite entanglement can be certified experimentally [1310, 1311].

A notable 2015 experiment prepared a non-Gaussian entangled ensemble of about 3,100 atoms and certified an entanglement depth of $2,910 \pm 190$ atoms, while also reconstructing a negative Wigner function [1312]. This result was not a quantum-computing demonstration. Contextuality has been identified as a computational resource in particular magic-state models [1313]. Entanglement creates nonclassical correlations, not an instantaneous physical interconnection that can transmit information or causal influence between distant qubits.

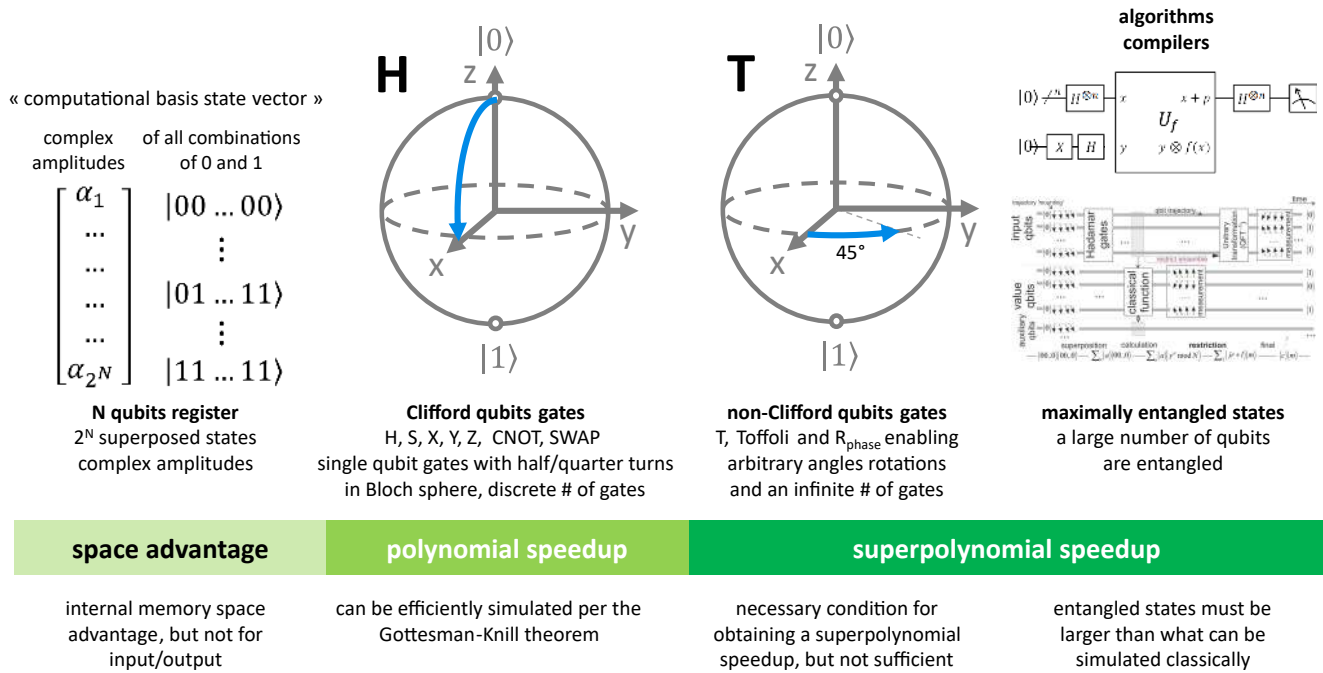


Figure 70: The role of several gate-based quantum computing supporting features in bringing some potential quantum speedup. The 2^N Hilbert-space dimension brings an internal processing space advantage which does not automatically translate into a speed advantage, particularly when considering end-to-end resource estimates including problem data encoding and circuit sampling. Clifford circuits with stabilizer inputs and Pauli measurements are efficiently classically simulable per the Gottesman-Knill theorem. It can't support an end-to-end computational advantage. Non-Clifford gates and entanglement are necessary but not sufficient to enable some superpolynomial speedup. Entanglement should deal with a sufficiently large number of physical or logical qubits to escape the classical simulability zone. © Olivier Ezratty, 2022-September 2026.

Local operations cannot be used to signal through entanglement alone.

1.5.4 Superpolynomial and exponential speedups

We introduce here the notion of superpolynomial speedup. A function is superpolynomial if it eventually grows faster than every polynomial. A quantum speedup is superpolynomial when the ratio between the best comparable classical cost and the quantum cost grows superpolynomially with the input size. Exponential speedup is one form of superpolynomial speedup, but superpolynomial growth need not be exponential.

Common asymptotic running-time classes include polynomial $O(n^k)$ for constant k , quasi-polynomial $\exp((\log n)^{O(1)})$, subexponential $\exp(o(n))$, for example $O(2^{n^\alpha})$ with $0 < \alpha < 1$, and exponential $O(2^{cn})$ for constant $c > 0$. A speedup is a comparison, not a running-time class by itself. It should be expressed as a ratio or a comparison between classical and quantum resource scalings for the same task. The term “strong quantum speedup” is usually reserved for a comparison against the best classical algorithm available, rather than being a synonym for exponential speedup [1314].

Other structured algebraic problems have efficient quantum algorithms and strong evidence of quantum advan-

tage, but one should distinguish this from an unconditional proof of exponential speedup over all classical algorithms. Examples include **estimating Gauss sums**, for which efficient quantum algorithms are known and classical hardness is supported by reductions from discrete logarithms [1315], and additive approximation of the **Jones polynomial** at specified roots of unity, which is BQP-complete in the formulation of Aharonov, Jones, and Landau [1316]. The Jones polynomial is a knot invariant, but it is not a complete or unique fingerprint: distinct knots can have the same Jones polynomial. Conversely, if two knots have different Jones polynomials, they cannot be equivalent.

Provable exponential quantum separations are clearest in oracle and query-complexity models. Candidate end-to-end superpolynomial advantages often exploit specific algebraic, spectral, or dynamical structure. Grover search deliberately assumes an unstructured black-box search space and provides only a quadratic query speedup. Factoring and hidden-subgroup algorithms exploit algebraic structure, while quantum-simulation algorithms exploit the structure of quantum dynamics. Structure can also help classical algorithms. For example, low-entanglement quantum states may be tractable with tensor-network methods. Therefore, the relevant comparison is always against the best classical method that exploits the same available structure.

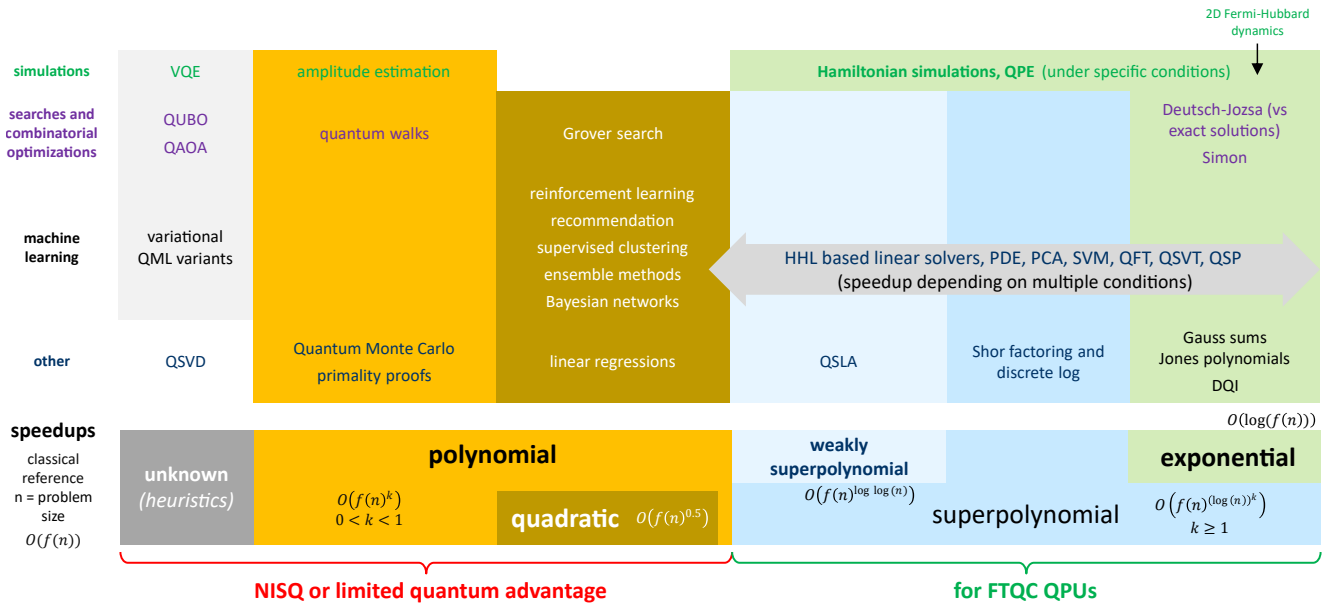


Figure 71: Indicative taxonomy of polynomial, superpolynomial, and exponential quantum speedups. The classification is problem- and access-model-dependent. In particular, speedups attributed to QPE, HHL-type solvers, QFT-based methods, QSP/QSVT, and quantum machine learning require end-to-end accounting of state preparation, oracle or block-encoding cost, conditioning, accuracy, number of circuit shots, readout, plus any potential QEC/FTQC additional cost. Only a few algorithms like Deutsch-Jozsa, Shor integer factoring and discrete log, and 2D Fermi-Hubbard dynamics can be classified in the exponential speedup range. © Olivier Ezratty, 2022-September 2026.

It is also necessary to account for randomized, approximation, and heuristic classical algorithms, which can substantially improve practical time to solution even when they do not change the worst-case complexity class. Many quantum algorithms are probabilistic because measurement produces samples from a distribution. That algorithmic randomness should be distinguished from physical errors and noise in the quantum processor.

Quantum algorithms are therefore only one route to improving difficult computations. Progress in quantum algorithms often stimulates better classical algorithms, implementations, and hardware, which continuously moves the baseline against which quantum advantage must be assessed. This is an ever moving target.

A good example is the rapid progress of classical optimization heuristics, including simulated-bifurcation approaches [1317]. Constant-factor benchmark improvements can move the practical crossover point substantially even when they do not change asymptotic complexity. In the terminology introduced for quantum-speedup benchmarking, a “strong quantum speedup” compares a quantum algorithm with the best classical algorithm for the same problem, not just with a selected classical comparable [1314].

A polynomial speedup is not intrinsically minor. A quadratic or higher-degree polynomial improvement can have substantial practical value if the crossover occurs at a useful problem size and computing time, and if the quantum implementation has manageable constants and overheads. So far, however, for both NISQ and FTQC, there are no known such quantum algorithms.

1.5.5 Theoretical vs practical speedups

But the details must be accounted for carefully [367, 1314] (Figure 72):

- The quantum and classical comparison should use the same problem definition, input distribution or worst-case promise, target accuracy, success probability, and output requirements. Asymptotic analysis is not synonymous with worst-case analysis. Both asymptotic scaling and finite-size time-to-solution matter, since favorable asymptotic scaling can be overwhelmed by large prefactors at relevant problem sizes [367, 1320, 1321].
- Check whether the claimed quantum advantage has been “dequantized” by a classical quantum-inspired algorithm, as has happened for several quantum machine-learning and linear-algebra proposals discussed in Table 6.
- The quantum speedup is affected by the number of circuit repetitions, or shots, needed to estimate the requested output to a target statistical precision and confidence. In Google’s 2019 Sycamore random-circuit-sampling experiment, one million samples of a circuit instance took about 200 s, while the largest reported 53-qubit, 20-cycle elided data set contained 30 million samples over ten circuit instances [1322]. For direct Monte Carlo estimation of an expectation value with bounded variance, the statistical error typically decreases as $O(1/\sqrt{M})$ after M independent shots, so reducing the error by a factor of ten may require about one hundred times more samples.

what slows down quantum computing?

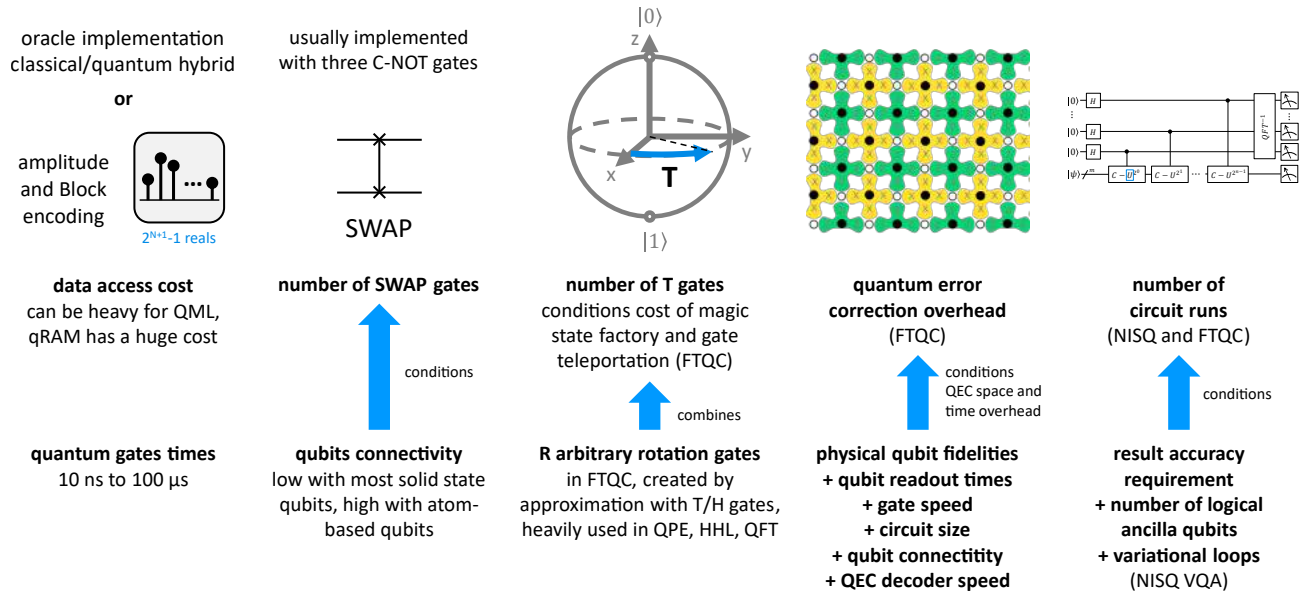


Figure 72: Quantum computing speedup must be computed after considering all the various hardware and circuit constraints which condition computing time. Many of these considerations are discussed in this book, particularly in the QEC and FTQC part. © Olivier Ezratty, 2022-2026.

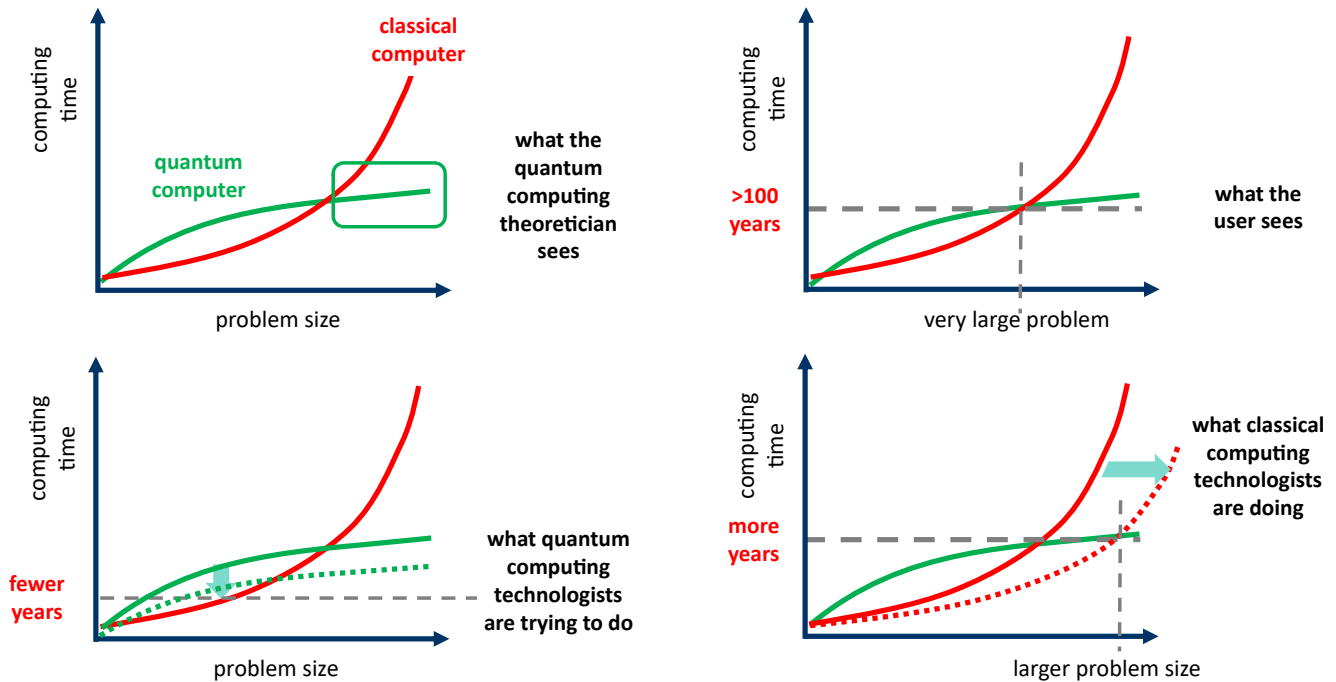


Figure 73: A theoretical quantum speedup can be impractical if the quantum algorithm becomes faster than its classical counterpart only at problem sizes for which both approaches take an unreasonable time. The crossover is a moving target: quantum hardware, algorithms, compilation, and parallelization may improve, while classical algorithms and hardware also improve. At small and intermediate scales, a quantum implementation can be slower because of large prefactors such as gate latency, fault-tolerance overhead, state preparation, and data loading, even when its asymptotic scaling is better. Source: Olivier Ezratty, 2023-2026, inspired by [367].

- Quantum advantage does not depend simply on how many qubits use Hadamard gates. Relevant quantities include the number of logical data and ancilla qubits, circuit depth, non-Clifford count or non-Clifford depth, connectivity and routing overhead,

state-preparation complexity, and the amount and structure of entanglement generated during the computation.

Q Non-Clifford resources, universality and fault tolerance

The relation between quantum speedup, universal gate sets, and magic is more precise when stated in terms of classical simulability rather than continuous motion in Hilbert space.

- Standard circuit-model quantum computation does not require a physically continuous set of gates. A finite discrete universal gate set can approximate arbitrary unitaries to any desired accuracy.
- The Gottesman-Knill theorem states that stabilizer circuits composed of Clifford operations, stabilizer-state preparation, and Pauli-basis measurements can be simulated efficiently classically [699]. The relevant limitation is therefore stabilizer structure, not the absence of continuous Bloch-sphere rotations.
- Adding a discrete non-Clifford resource is enough for universality. For example, Clifford+ T is a universal gate set, even though T is itself a fixed $\pi/4$ phase rotation. Hence arbitrary-angle native rotations are not a necessary condition for theoretical exponential quantum speedup [1305].
- On many NISQ platforms, continuously parameterized physical rotations are convenient native controls. Their availability does not by itself imply computational advantage because noisy shallow circuits can still be classically tractable or provide no useful separation.
- In many fault-tolerant architectures based on stabilizer codes, non-Clifford logical operations are costly and can be implemented using magic-state distillation followed by state injection or gate teleportation. They are not generally “transversally injected”. Moreover, a universal set of logical gates cannot all be transversal for a quantum code that detects arbitrary single-subsystem errors [1318].
- Arbitrary logical rotations can be approximated by sequences over a finite universal gate set. Solovay-Kitaev establishes polylogarithmic approximation overhead for suitable gate sets, while modern Clifford+ T synthesis methods can be substantially more efficient in practice. The resulting non-Clifford count can nevertheless dominate fault-tolerant resource estimates [1319].
- Data encoding and quantum-state preparation must also be accounted for. They are particularly important in quantum machine learning and quantum linear-algebra algorithms, where a claimed speedup can depend on assuming efficient state preparation or sample/query access to classical data [1302, 1323]. Similar input-state preparation costs also arise in quantum chemistry and simulation.
- The same applies to oracle-based algorithms, particularly when their complexity assumes coherent random access to a large classical data set. Scalable fault-tolerant qRAM with the required capacity, latency, and error rate is not currently available, so its construction and access costs cannot be treated as free.
- Large-scale fault-tolerant quantum computing adds space and time overheads for logical qubits encoding, error syndrome extraction, error syndrome decoding, logical gate operations, and the generation and teleportation of non-Clifford states. Under fault-tolerance threshold assumptions this overhead need not destroy the computational complexity separation, but in practical resource estimates it can erase a modest polynomial or quadratic advantage over relevant problem sizes [1324].
- Assess the crossover problem size and the corresponding wall-clock time. If the quantum method overtakes the best classical counterpart only when both runtimes are operationally irrelevant, the asymptotic advantage has little practical value [367]. See Figure 73. Resource estimates for new quantum optimization algorithms should therefore include the prefactors and end-to-end costs that determine the crossover with classical optimizers, not only asymptotic query or gate counts.
- Watch theoretical advances that delimit where quantum speedups are possible. Huffstutler *et al.* developed a framework for ruling out superpolynomial quantum query speedups for certain classes of partial Boolean functions when specific promise-aware complexity measures or completion properties satisfy their criteria [1325].

Assessing a practical quantum speedup therefore requires an end-to-end model of time to solution, including all dominant quantum and classical costs.

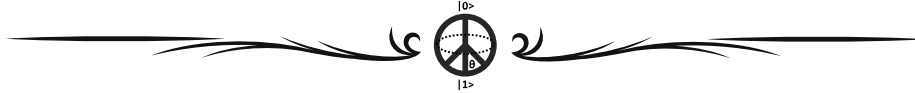
We can summarize the kinds of benefits coming from quantum computing over time like this:

Better. A quantum implementation may, for a specific benchmark and fixed resource budget, produce a solution of higher quality or accuracy than a chosen classical method. Such a quality advantage has to be demonstrated against the best relevant classical baseline under the same data-access, accuracy, and resource assumptions. There is no generic result that NISQ quantum machine-learning or optimization algorithms explore the

computational space “better” than classical algorithms [1302, 1326].

Faster. A practical speed advantage exists only when the total time to solution is lower than for the best classical alternative after including state preparation, data access, compilation, circuit execution, repetitions, error mitigation or error correction, decoding, and classical orchestration. Whether this occurs in a NISQ or FTQC regime is problem-dependent.

Beyond. In the longer term, sufficiently large fault-tolerant quantum computers may access problem regimes that are practically inaccessible to classical supercomputers, particularly for some quantum-simulation and cryptanalytic tasks. There is no universal threshold such as 100 logical qubits: the crossover depends on logical-qubit count, logical error rate, circuit depth, non-Clifford resources, target precision, and continuing improvements in classical algorithms and hardware.



1.5.6 Bibliography and notes

These are the bibliographical references and notes for the **Quantum speedups** subsection.

- [31] [Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer](#), by Peter W. Shor, SIAM Journal on Computing, October 1997 (26 pages).
- [367] [Disentangling Hype from Practicality: On Realistically Achieving Quantum Advantage](#), by Torsten Hoefler et al., Communications of the ACM, May 2023.
- [699] [Improved simulation of stabilizer circuits](#), by Scott Aaronson and Daniel Gottesman, Physical Review A, November 2004 (15 pages).
- [701] [Efficient classical simulation of slightly entangled quantum computations](#), by Guifre Vidal, Physical Review Letters, October 2003 (39 pages).
- [1301] [A review on quantum Fourier transform](#), by Tomás Barros, Pablo Álvarez Sánchez, Bárbara Vidal, and Mauricio Solar, Quantum Information Processing, January 2026.
- [1302] [Quantum Principal Component Analysis Only Achieves an Exponential Speedup Because of Its State Preparation Assumptions](#), by Ewin Tang, Physical Review Letters, August 2021 (preprint on [arXiv](#)).
- [1303] [Factoring integers with the number field sieve](#), by Joe P. Buhler, Hendrik W. Lenstra Jr., and Carl Pomerance, The Development of the Number Field Sieve, Lecture Notes in Mathematics 1554, August 1993 (45 pages).
- [1304] [Algorithm design](#), by John Kleinberg and Eva Tardos, Pearson Addison Wesley, 2006 (432 pages).
- [1305] [Universal quantum computation with ideal Clifford gates and noisy ancillas](#), by Sergey Bravyi and Alexei Kitaev, Physical Review A, February 2005 (14 pages) (preprint on [arXiv](#)).
- [1306] [On the role of entanglement in quantum-computational speed-up](#), by Richard Jozsa and Noah Linden, Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences, August 2003 (22 pages) (preprint on [arXiv](#)).
- [1307] [A brief introduction to multipartite entanglement](#), by Ingemar Bengtsson and Karol Zyczkowski, arXiv, December 2016 (37 pages).
- [1308] [Symmetry and Classification of Multipartite Entangled States](#), by Adam Burchardt, arXiv, April 2022 (126 pages).
- [1309] [How Much Entanglement Do Quantum Optimization Algorithms Require?](#), by Yanzhu Chen, Sophia E. Economou, et al., arXiv, July 2023 (15 pages).
- [1310] [Quantifying multiparticle entanglement with randomized measurements](#), by Sophia Ohnemus, Heinz-Peter Breuer, and Andreas Ketterer, arXiv, July 2022 (17 pages).
- [1311] [Scalable estimation of pure multi-qubit states](#), by Luciano Pereira, Leonardo Zambrano, and Aldo Delgado, npj Quantum Information, May 2022.
- [1312] [Entanglement with negative Wigner function of almost 3,000 atoms heralded by one photon](#), by Robert McConnell, Hao Zhang, Jiazhong Hu, Senka Ćuk, and Vladan Vuletić, Nature, March 2015 (4 pages) (preprint on [arXiv](#)).
- [1313] [Contextuality supplies the ‘magic’ for quantum computation](#), by Mark Howard, Joel Wallman, Victor Veitch, and Joseph Emerson, Nature, June 2014 (preprint on [arXiv](#)).
- [1314] [Defining and detecting quantum speedup](#), by Troels F. Rønnow, Zhihui Wang, Joshua Job, Sergio Boixo, Sergei V. Isakov, David Wecker, John M. Martinis, Daniel A. Lidar, and Matthias Troyer, Science, July 2014 (5 pages) (preprint on [arXiv](#)).
- [1315] [Efficient Quantum Algorithms for Estimating Gauss Sums](#), by Wim van Dam and Gadiel Seroussi, arXiv, July 2002 (11 pages).
- [1316] [A Polynomial Quantum Algorithm for Approximating the Jones Polynomial](#), by Dorit Aharonov, Vaughan Jones, and Zeph Landau, Algorithmica, November 2009 (27 pages) (preprint on [arXiv](#)).
- [1317] [Combinatorial optimization by simulating adiabatic bifurcations in nonlinear Hamiltonian systems](#), by Hayato Goto, Kosuke Tatsumura, and Alexander R. Dixon, Science Advances, April 2019 (8 pages).
- [1318] [Restrictions on Transversal Encoded Quantum Gate Sets](#), by Bryan Eastin and Emanuel Knill, Physical Review Letters, March 2009 (4 pages) (preprint on [arXiv](#)).
- [1319] [The Solovay-Kitaev algorithm](#), by Christopher M. Dawson and Michael A. Nielsen, Quantum Information and Computation, January 2006 (15 pages).
- [1320] [Quantifying Grover speed-ups beyond asymptotic analysis](#), by Chris Cade, Marten Folkertsma, Ido Niesen, and Jordi Weggemans, arXiv, September 2023 (57 pages).
- [1321] [Realistic Runtime Analysis for Quantum Simplex Computation](#), by Sabrina Ammann, Timo de Wolff, et al., arXiv, November 2023 (39 pages).
- [1322] [Quantum supremacy using a programmable superconducting processor](#), by Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C. Bardin, et al., Nature, October 2019 (6 pages).
- [1323] [Power of data in quantum machine learning](#), by Hsin-Yuan Huang, Michael Broughton, Masoud Mohseni, Ryan Babbush, Sergio Boixo, Hartmut Neven, and Jarrod R. McClean, Nature Communications, May 2021.
- [1324] [Focus beyond Quadratic Speedups for Error-Corrected Quantum Advantage](#), by Ryan Babbush, March 2021 (11 pages).
- [1325] [A Framework for Ruling out Quantum Speedups](#), by Thomas Huffstutler, Upendra Kapshikar, David Miloschewsky, and Supartha Podder, 51st International

Symposium on Mathematical Foundations of Computer Science, LIPIcs 386, August 2026 (17 pages).

- [1326] [The Complexity of NISQ](#), by Sitan Chen, Jordan Cotler, Hsin-Yuan Huang, and Jerry Li, arXiv, October 2022 (37 pages).

This **Quantum speedups** subsection contains 30 bibliographical references and notes containing at least 1,029 pages.

1.6 Complexity classes

So far, we have reviewed a lot of the most common quantum algorithms and their theoretical acceleration.

Quantum computing is sometimes presented as a miracle solution to extend computing capacities beyond the limits of classical supercomputing. For some structured problems, quantum algorithms can provide substantial asymptotic speedups over the best-known classical algorithms. But what is the nature of the problems that may benefit from a quantum computer? And above all, what are the computational limits of quantum computers?

We will see that practical limits shift over time as algorithms and hardware improve, while the mathematical definitions of complexity classes are fixed. This deals with **computational complexity theory**, an abstract field of computer science and mathematics involving classes such as P, NP and BQP. Computer scientists have been discussing for more than half a century whether $\mathbf{P} = \mathbf{NP}$ or not.

This is the science of computational complexity classes. Behind the mathematics of complexity lie key technical and philosophical considerations about the limits of computation. Complexity classes can often be pictured as partially nested sets. They classify computational problems according to asymptotic resource requirements, most commonly time or memory space, as a function of the input length n , usually measured in bits. Typical growth rates are constant, logarithmic, linear, polynomial and exponential. Polynomial time, $O(n^k)$ for a fixed constant k , is conventionally used as the boundary of efficient computation in complexity theory, although a large exponent or prefactor can still make an algorithm impractical.

Quantum computing can provide polynomial, superpolynomial or exponential speedups for particular computational models and structured problems, but there is no general rule turning classically exponential problems into polynomial-time quantum problems. For example, Shor's factoring algorithm runs in quantum polynomial time, whereas the best-known classical factoring algorithms are subexponential rather than proven exponential. Exponential quantum separations are known in oracle and query models, while unconditional separations between BQP and major classical classes remain open [34]. Problems requiring exponential or double-exponential resources, and problems limited by exponential memory, therefore remain far beyond practical computation even with very large fault-tolerant quantum computers.

These limitations have an indirect impact on predictions about the creation of some omniscient artificial intelligence capable of transcending human reasoning and solving all problems. This hypothetical AGI (Artificial General Intelligence) will be limited by the data and concepts that feed it and by the impossibility of solving all complex problems.

Humanity will continue to confront computationally intractable and undecidable tasks. It will not be able to solve all the complex problems around! Quantum computing does not allow us to dominate nature, to put the whole Universe in equations and to predict how it will run with quantum precision. Chance and the unexpected will continue to play a role in a very non-deterministic world, and for the better. It is a small lesson in humility for humanity.

Problem Complexity Classes

To dive into complexity classes, you need to define the main classes of problems by level of complexity. Here I am trying to simplify complexity, this time in the literal sense of the word.

Complexity classes classify defined computational problems according to resource bounds, particularly at asymptotic levels. Some hard problems have a combinatorial search structure, such as SAT, while others arise from algebra, graphs, numerical computation, physics or many-body systems. Brute-force search and oracle models are useful tools, but they are not the basis of complexity classes in general.

Problem classes use the notion of Turing's deterministic and nondeterministic machines. Turing machines are conceptual models of computers created by Alan Turing before the Second World War (Figure 74).

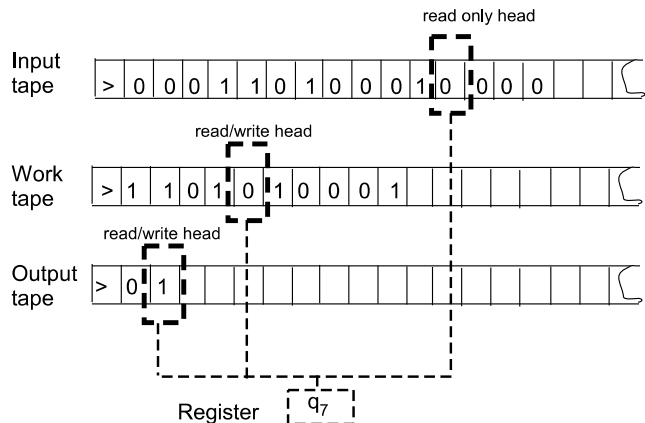


Figure 74: The Turing machine. Source: [1327].

A standard Turing machine consists of a finite control, a read/write head and an unbounded tape containing symbols. Program instructions are encoded in the finite control or, for a universal Turing machine, can themselves be represented as data on the tape. Multi-tape variants are often convenient and are polynomially equivalent to the standard model. Turing machines are used to define computability and many classical complexity classes [1327].

Computers are all metaphorically Turing machines, with limited space and memory, reproducing this logic by reading program instructions and managing data in random access memory (RAM) or persistent storage (hard disk, SSD, ...).

Associated with the notion of a Turing machine is the **Church-Turing thesis**, named after Alonzo Church and Alan Turing. It states, informally, that every effectively calculable function can be computed by a Turing machine. Equivalent formal models include lambda calculus and recursive functions. The thesis concerns computability, not whether a computation is efficient.

In a deterministic Turing machine, each machine configuration determines at most one next configuration. In a nondeterministic Turing machine, a configuration can have several possible successors (Figure 75). A decision problem is accepted if at least one computation branch reaches an accepting state. Nondeterminism is a mathematical model of branching computation, not a physical machine that evaluates all branches in parallel.

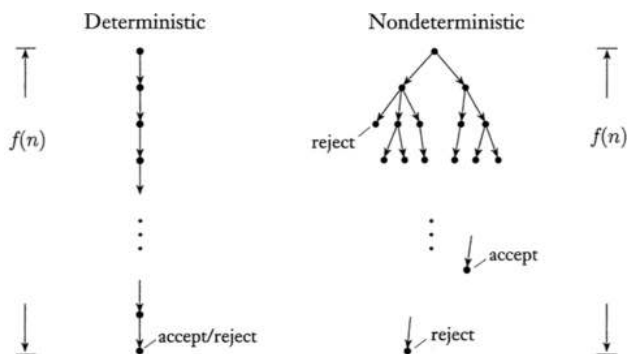


Figure 75: Deterministic and nondeterministic Turing machines.

The relation between deterministic and nondeterministic resource bounds is itself a central subject of complexity theory.

1.6.1 Generic complexity classes

Complexity can refer to several resources, notably computation time, memory space, communication, circuit depth or query count. Time and space complexity are distinct, and which one becomes the practical bottleneck depends on the algorithm, architecture and input size. However, in many situations, time and space trade-offs can be made.

Membership in a complexity class is defined by the existence of an algorithm or computational model satisfying a specified asymptotic resource bound. The best-known algorithms determine what is currently known about membership and practical tractability, but they do not change the definition of the class.

Standard complexity classes such as P, NP and PSPACE are defined using computational models such as Turing machines or circuits. Black-box or oracle models define a different but closely related notion, query complexity, and are particularly useful for proving relative separations and lower bounds.

So here are these classes by increasing levels of complexity knowing that we will spend more time on NP related classes (Figure 76).

L, also called deterministic logarithmic space or DLOGSPACE, is the class of decision problems solvable by a deterministic Turing machine using $O(\log n)$ working memory. The time may be polynomial even though the additional working space is logarithmic. Representative problems include reachability in undirected graphs and several pointer-manipulation problems.

NL is the class of decision problems solvable using $O(\log n)$ space on a nondeterministic Turing machine. Whether $L = NL$ is unknown. A canonical NL-complete problem is directed s - t reachability.

LIN is the class of decision problems that can be solved by a deterministic Turing machine in linear time. It is a subset of the P class.

P, also called PTIME, is the class of decision problems solvable by a deterministic Turing machine in polynomial time $O(n^k)$ for some fixed constant k . P is conventionally associated with tractable computation, although large exponents and prefactors can still be impractical. Examples include graph reachability, shortest-path decision variants and primality testing. Sorting, matrix multiplication and matrix inversion are function problems rather than decision problems, but their standard algorithms also run in polynomial time.

BPP is a class of problems that can be solved by random approaches (“Bounded-Error Probabilistic Polynomial-Time”). It would seem that $BPP = P$ but this has not yet been demonstrated.

PP, Probabilistic Polynomial Time, is the class of decision problems for which a polynomial-time probabilistic Turing machine accepts yes-instances with probability greater than $1/2$ and accepts no-instances with probability at most $1/2$. Unlike BPP, the gap from $1/2$ may be exponentially small.

FP, “Function Polynomial Time” represents the set of function problems that can be computed by a deterministic Turing machine in polynomial time.

NP is the class of decision problems for which a yes-instance has a polynomial-size certificate that can be verified in polynomial time by a deterministic machine. Equivalently, NP is the class solvable in polynomial time by a nondeterministic Turing machine. P is contained in NP. NP problems are not known in general to require exponential classical time. Brute-force algorithms for many NP-complete problems do take exponential time, but better exact and heuristic algorithms often exist. Quantum superposition also does not make the outcomes of 2^n candidate computations simultaneously readable. For unstructured search, Grover’s algorithm provides a quadratic rather than an exponential speedup.

Examples include the decision version of the Steiner tree problem, many constrained scheduling decision problems, SAT and Hamiltonian cycle. Literal matching of

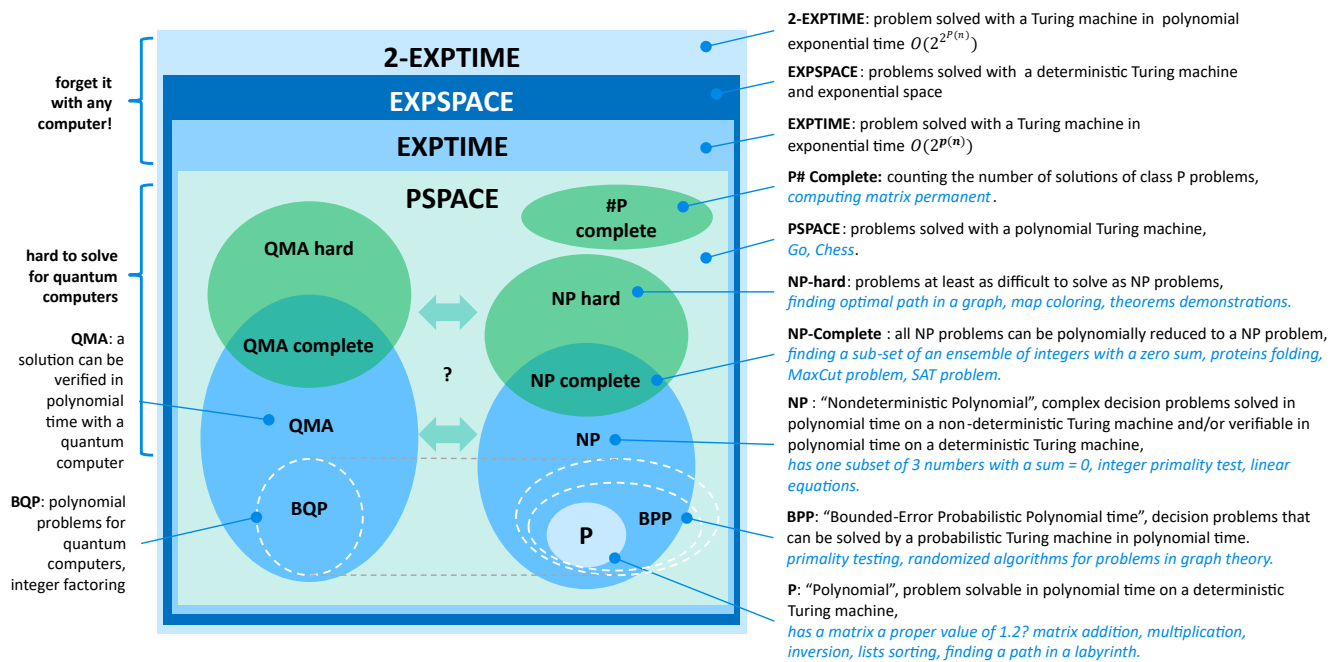


Figure 76: Schematic view of selected classical and quantum complexity classes. Proof that all NP class is within PSPACE in [1328], and PSPACE in EXPTIME [1327, 1329]. © Olivier Ezratty, 2021-September 2026.

a DNA subsequence in a genome is polynomial-time and is therefore not a useful example of an NP-hard problem unless additional combinatorial constraints are specified.

Hard combinatorial problems are omnipresent in logistics, planning, production, transportation, telecom, utilities, finance and cryptography. A **decidable** decision problem is one for which an algorithm halts on every input and returns the correct yes or no answer. Decidability does not imply practical feasibility. A decidable problem may require prohibitive time or memory.

An exponential problem has an elegant solution if one can find one solution that has a polynomial or, at best, linear duration. Polynomial times scale better than exponential times!

A big debate has been going on since 1971 (Stephen Cook et Leonid Levin) as to whether class P equals class NP. If $P = NP$, it would be as simple to find a result when one can also simply verify it [1330–1332]. The consensus is that P is not equal to NP [1333].

Determining whether $P = NP$ is one of the seven Millennium Prize Problems announced by the Clay Mathematics Institute in 2000, with a \$1M prize for a correct solution (Figure 77). Other Millennium Prize Problems include proving or disproving the Riemann hypothesis and establishing existence and smoothness properties for the three-dimensional incompressible Navier-Stokes equations.

On the P vs NP side, the wording of the challenge provides an example of such a problem: you must allocate 50 rooms for two students to 400 candidates, but some candidates must not live in the same room [1334]. The combinatorial choice of 100 students among 400 is huge,

so exhaustive search is impractical. A candidate room allocation can be verified efficiently, which places the corresponding decision problem in NP. If $P = NP$, every problem in NP would have a polynomial-time deterministic algorithm. If $P \neq NP$, at least some problems in NP are not in P, and in particular no NP-complete problem is in P. This does not mean that every problem in NP is hard, since P is already a subset of NP. In practice, exact algorithms, approximation algorithms and heuristics can solve many useful instances without exhausting the full search space. A polynomial-time algorithm $O(n^k)$ can also be impractical when k or the prefactor is large, so asymptotic tractability and engineering feasibility are different notions.

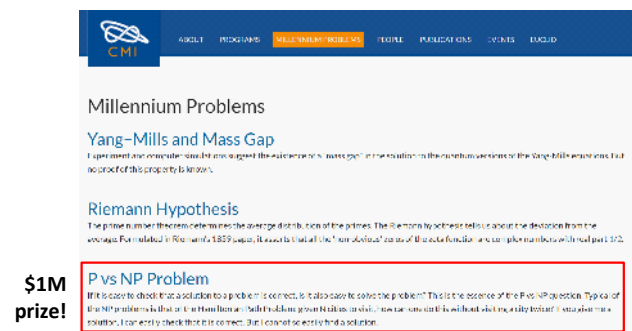


Figure 77: The Millennium challenge and P vs NP problem. Source: [Clay Mathematics Institute mathematical challenges](#).

A decision problem is **NP-complete** if it is in NP and every problem in NP can be reduced to it in polynomial time. SAT and 3SAT are canonical examples. No polynomial-time classical algorithm is known for any NP-complete problem. Likewise, no polynomial-time

quantum algorithm is known for general NP-complete problems, and whether NP is contained in BQP remains open.

Two related but distinct optimization problems are **bin packing** and **knapsack**. Bin packing asks how to pack items into the minimum number of fixed-capacity bins. Knapsack asks which items to select so as to maximize total value while respecting one or more capacity constraints. Packing a car trunk is a useful visual analogy, but it can represent different formal problems depending on the objective and constraints.

Common variants include the 0-1 knapsack problem, where each item is selected at most once [1335–1337], the bounded knapsack problem, where each item type has a finite multiplicity, and the unbounded knapsack problem, where arbitrarily many copies of each item type may be selected. Fractional knapsack is a different variant in which fractions of items may be selected and admits a greedy polynomial-time solution. In contrast, 0-1 and unbounded knapsack are NP-hard optimization problems, although pseudo-polynomial dynamic-programming algorithms and approximation schemes are available.

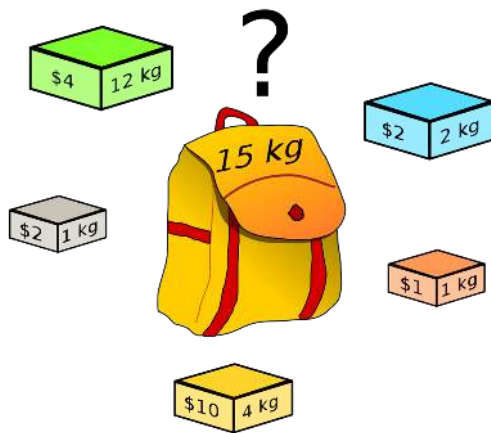


Figure 78: The famous bin-packing problem. Source: Wikipedia.

Knapsack problems variants are the 0-1 Quadratic Knapsack Problem (QKP), an extension of 0-1-KP which aims to maximize a quadratic objective function, and can for example include additional profits or costs coming from pairs of items being selected together. Then, the 0-1 Multidimensional Knapsack Problem (MDKP), or Multiple-Constraint Knapsack Problem which extends the classical 0-1-KP problem by introducing multiple linear resource constraints, like maximizing the weight, volume, or price of the packed items. As shown in [1338], it seems way more difficult to obtain a quantum advantage with solving MDKP than the GKP problems. And the comparison was made with a single CPU AMD workstation using 128 GB of memory which would cost about \$2K, way less than any FTQC QPU, and the considered QPU was supposed to support gate cycles of 1 ns that are entirely out of sight for any qubit modality,

particularly when considering the cost of quantum error correction.

Complete NP problems also include the **subset sum problem** of finding a subset of a set of integers whose sum is equal to an arbitrary integer.

The generalized **Minesweeper** consistency problem asks whether a partially revealed board is compatible with some placement of mines. This decision problem is NP-complete [1339]. The familiar Windows game is a finite instance of this broader problem family (Figure 80).

Some formal protein-folding models are NP-complete. In particular, optimal folding in the three-dimensional hydrophobic-hydrophilic lattice model is NP-complete [1340]. This does not establish that real biological protein folding, without a precise computational formulation and accuracy criterion, is generically an NP-complete problem.

Such worst-case hardness results indicate that exact optimization can become difficult as model size grows. If one NP-complete decision problem had a polynomial-time algorithm, every problem in NP could be solved in polynomial time through polynomial-time reductions. This does not mean that all solutions of every NP-complete problem would automatically be enumerated.

Graph coloring contains several distinct problems (Figure 82). For vertex coloring, deciding whether a general graph is colorable with a fixed number $k \geq 3$ is NP-complete, while 2-colorability is in P [1341]. Finding the chromatic number is NP-hard. Edge coloring is also computationally hard in general, with determination of the chromatic index NP-complete even for cubic graphs [1342]. For planar maps, the four-color theorem guarantees that four colors always suffice, while deciding whether three colors suffice remains NP-complete for suitable planar graph formulations.

- **Graph node** coloring has applications in frequency assignment and compiler register allocation. The k -colorability decision problem is NP-complete for fixed $k \geq 3$, while finding the minimum number of colors is NP-hard.
- **Edge** coloring can model frequency or channel assignment when adjacent links must use different resources. Contrary to the previous version, edge coloring is not polynomial-time in general. Determining the chromatic index is NP-complete, including for cubic graphs [1342].
- **Planar map** coloring corresponds to vertex coloring of a planar adjacency graph. Four colors always suffice by the four-color theorem. Deciding whether three colors suffice is NP-complete for general planar graphs, subject to the precise graph formulation.

For a complexity class C , a problem is **C-hard** if every problem in C can be reduced to it under the specified

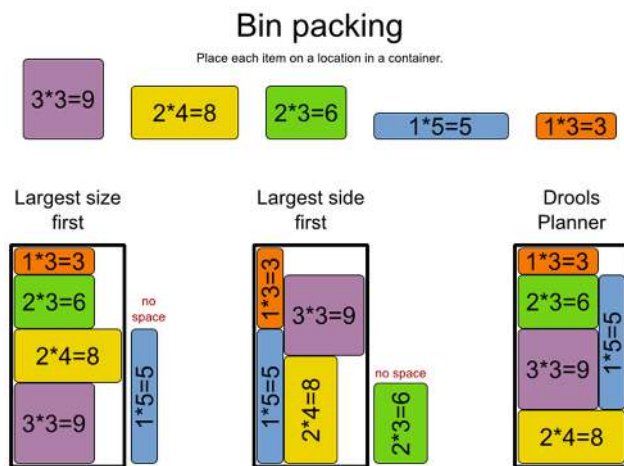


Figure 79: Ever filled your car's trunk when going to vacation? Source: [Stackoverflow](#).

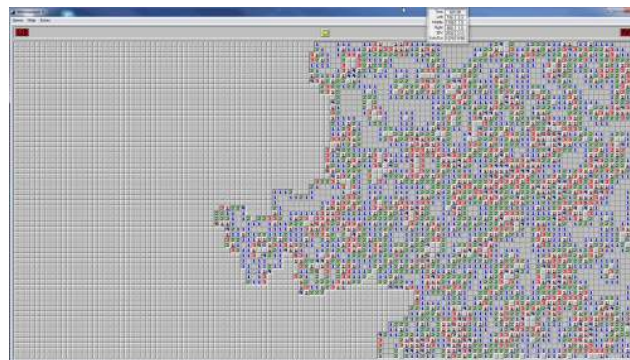


Figure 80: The generalized Minesweeper consistency problem is NP-complete [1339]. Source of the illustration.

reduction. A problem is **C-complete** if it is both C-hard and a member of C. C-hard problems do not form a subclass of C and may lie outside C altogether [1343].

A problem is **NP-hard** if every problem in NP can be reduced to it in polynomial time. Many important NP-hard problems are optimization problems, but NP-hardness is not restricted to optimization. An NP-hard problem need not itself be a decision problem or belong to NP.

The optimization version of the **traveling salesperson problem** asks for the shortest tour visiting each city once and returning to the starting city. It is NP-hard. Exact algorithms do not necessarily enumerate every possible tour. The Held-Karp dynamic-programming algorithm, for example, runs in $O(n^2 2^n)$ time and $O(n 2^n)$ space [1344].

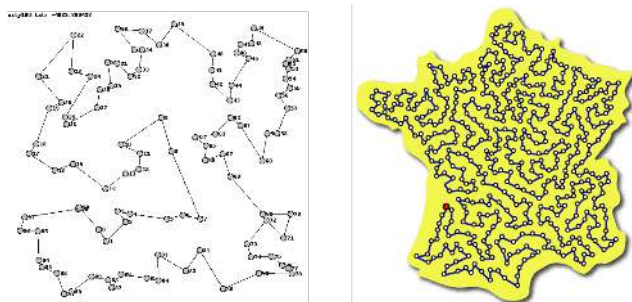


Figure 81: The TSP (traveling salesperson problem). What is the shortest possible route that visits each place exactly once and returns to the origin place is a NP-hard problem. Deciding whether there is a solution of at most length L is NP-complete. There is no known quantum algorithm that solves a TSP problem efficiently. And classical solution using heuristics fare well for very large problems.

The corresponding decision problem asks whether there exists a tour whose total cost is at most a threshold L . A proposed tour can be checked in polynomial time,

so this decision problem is in NP and is NP-complete (Figure 81).

A Hamiltonian cycle is a cycle that visits every vertex exactly once. Deciding whether a general graph contains one is NP-complete. TSP adds edge costs and asks for a minimum-cost Hamiltonian tour. Its decision version is NP-complete and its optimization version is NP-hard.

Straight enumeration of tours scales factorially, while the classic Held-Karp dynamic-programming algorithm uses $O(n^2 2^n)$ time [1344]. This is an exact upper bound, not a known optimal lower bound. Modern branch-and-cut solvers can prove optimality for much larger structured instances, including tours with tens of thousands of locations, even though worst-case exponential scaling remains.

There is no known polynomial-time quantum algorithm for exact general TSP. TSP is not a special case of the shortest-path problem. Classical shortest path between two vertices is in P, whereas TSP requires a tour through all vertices and is NP-hard. Quantum algorithms can accelerate some graph problems, but the speedup and complexity depend strongly on the precise problem and access model [1345].

Generalized versions of several **Nintendo** games, where the level size is treated as an unbounded input, have been shown NP-hard or NP-complete [1346]. These asymptotic results do not apply to one fixed commercial game instance. No general polynomial-time quantum algorithms are known for NP-hard problems.

Another NP-hard combinatorial optimization problem is LABS, Low Autocorrelation Binary Sequences. Its goal is to find a binary sequence of length n , with entries $+1$ or -1 , that minimizes the sum of squared off-peak aperiodic autocorrelations. The search space contains 2^n sequences before exploiting symmetries. Exact branch-and-bound calculations have determined optima up to lengths of order $n \sim 10^2$, while larger instances rely on

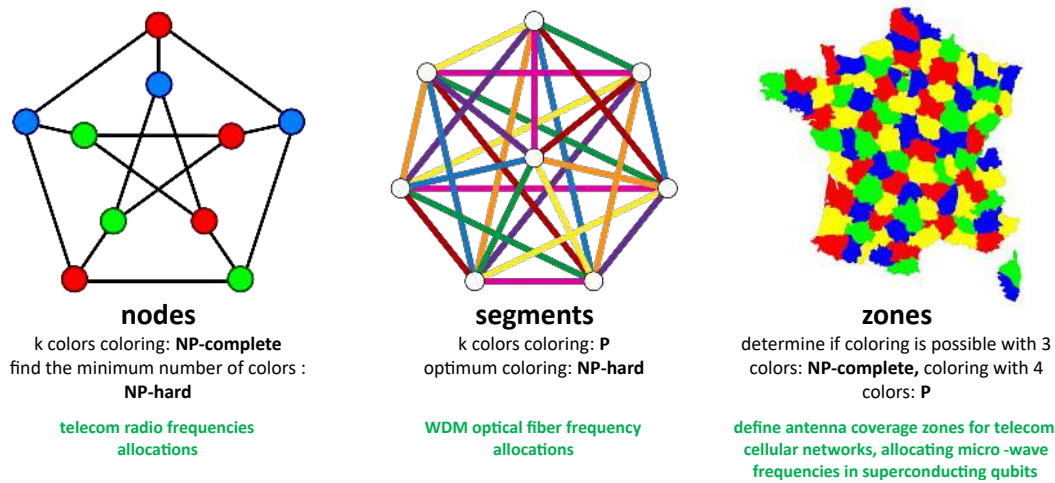


Figure 82: Complexity of selected vertex, edge and planar map coloring problems.

heuristics. LABS has applications in signal processing, communications and statistical physics. A 2025 paper presented a NISQ-oriented solver [1347], but this does not establish scalable quantum advantage.

PSPACE is the class of decision problems solvable using polynomial space on a deterministic Turing machine. NPSPACE is the analogous nondeterministic polynomial-space class. Savitch's theorem implies that $\text{NPSPACE} = \text{PSPACE}$ [1348].

quantum computers, and distinguished generalized games from fixed board sizes. **EXPTIME** is the class of decision problems solvable by a deterministic Turing machine in time $2^{p(n)}$ for some polynomial p . Generalized $n \times n$ chess is EXPTIME-complete [1349]. These complexity results concern families of games whose board size grows with the input. They are unrelated to the finite standard chess and Go boards played by Deep Blue or AlphaGo. Since BQP is contained in PSPACE and PSPACE is strictly contained in EXPTIME, an EXPTIME-complete problem cannot be in BQP.

NEXPTIME, also called NEXP, is the class of decision problems solvable in exponential time by a nondeterministic Turing machine. The definition is a time bound and does not require an “unlimited memory” qualifier.

EXPSpace is the class of problems that can be solved in exponential space. In other words, they are difficult to access on today's and even tomorrow's machines.

2-EXPTIME, also called 2-EXP, is the class of decision problems solvable by a deterministic Turing machine in double-exponential time $2^{2^{p(n)}}$ for some polynomial p . It strictly contains EXPTIME by the deterministic time-hierarchy theorem.

#P, introduced by Leslie G. Valiant, counts the number of accepting computation paths of a nondeterministic polynomial-time machine, equivalently the number of witnesses of an NP relation. It is a counting class rather than a decision-problem class. The associated hardness notions are **#P-hard** and **#P-complete**.

Valiant proved in 1979 that computing the permanent of a 0-1 matrix is #P-complete [1350]. Scott Aaronson later gave a linear-optical proof of the #P-hardness of the permanent [1351]. The permanent also plays a central role in the complexity of boson sampling, discussed later in the section on photonic qubits.

The classes PSPACE, EXPTIME, NEXPTIME, EXPSpace and 2-EXPTIME contain many artificial and natural problems from logic, games, verification, automata and other domains. Their definitions concern asymptotic computational resources, not specifically the prediction of strongly interacting physical systems.

One cannot assign a complexity class to “simulating a whole living cell” without first defining a precise computational problem, its input encoding, required output, accuracy and time horizon. A physically complex system is not automatically associated with a higher complexity class. The computational complexity depends on the formal task being asked.

There are many other problem complexity classes that won't be described here: EXP (problems solvable by deterministic Turing machines in exponential time), MIP (Multi-prover Interactive Proofs), RP (Randomized Polynomial time), RE (Recursively Enumerable [1352]), ZPP (Zero-error Probabilistic Polynomial time), SL (Symmetric Logspace), NC (problems that can be efficiently solved in parallel [1353]), AC^0 , TC^0 , MA (Merlin Arthur, where Merlin sends a proof, and Arthur which is a randomized polynomial-time verifier checks it with bounded error), AM (Arthur Merlin, where Arthur sends random bits first, then Merlin replies and Arthur verifies the response using randomness and limited interaction), and SZK (Statistical Zero Knowledge, a class of problems that admit interactive proofs where the verifier learns nothing beyond the truth of the statement)! You can find them in the [Complexity Zoo](#) site which inventories the zoo of problem complexity classes (Figure 83). There seems to be over a hundred of them [1327].

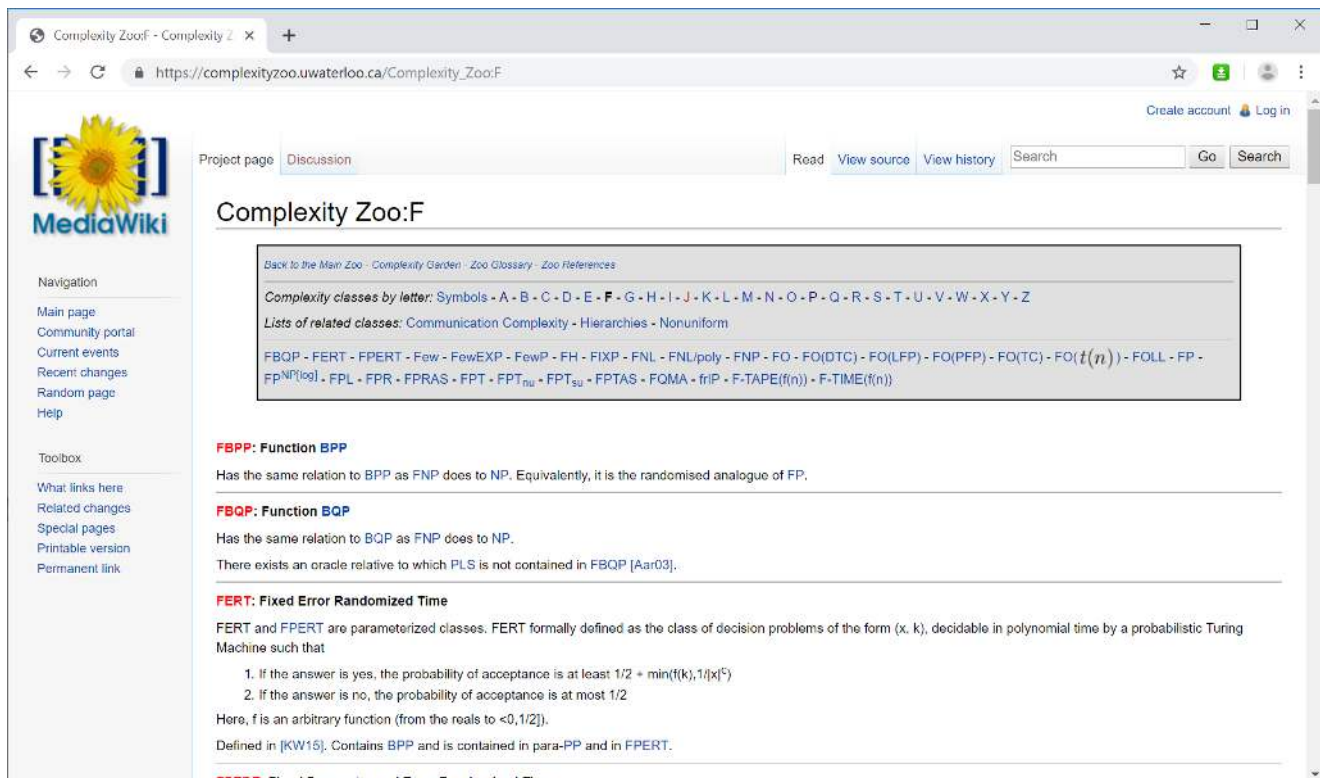


Figure 83: There is even a zoo website for complexity classes! Source: the [Complexity zoo](https://complexityzoo.uwaterloo.ca/Complexity_Zoo:F)

1.6.2 Quantum complexity classes

Let's now discuss the classification of problems that are theoretically addressable by quantum computers, the correspondence with the *above* classes being still a problem that is not entirely solved! The classification is different because quantum computers can provide some potentially exponential speedups versus classical computers.

These classes are idealized asymptotic models and do not directly include hardware limitations such as decoherence, gate errors, control latency or error-correction overhead. Fault-tolerant quantum error correction can suppress logical error rates below a target value when physical error rates and architecture satisfy suitable thresholds, but it does so with substantial qubit, gate and time overhead. It does not remove computation-time complexity.

PH, the Polynomial Hierarchy, is the union of an infinite sequence of complexity classes built from alternating existential and universal polynomial-time quantifiers. Its first levels contain NP and coNP, and PH is contained in PSPACE.

BQP, Bounded-Error Quantum Polynomial Time, is the class of decision problems solvable by a uniform polynomial-size quantum circuit family with error probability at most $1/3$, where the constant can be reduced exponentially by repetition. P is contained in BQP [34]. Whether BQP is strictly larger than P, whether BQP is contained in NP, and whether NP is contained in BQP are all open in the unrelativized setting. Raz and Tal

proved an *oracle separation*: there exists an oracle O for which BQP^O is not contained in PH^O [1354].

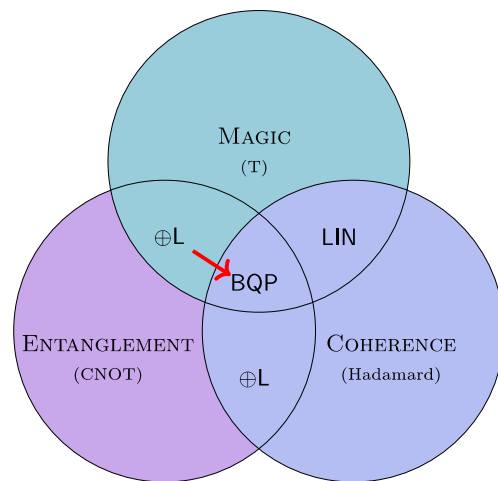


Figure 84: Illustrative quantum-resource view involving entanglement, coherence and magic. This is not a formal characterization of BQP. Source: [1355]. Added in 2024.

PQP, Probabilistic Quantum Polynomial Time, is the unbounded-error quantum analogue of PP. It uses a strict acceptance threshold at $1/2$ without requiring a constant gap, and $PQP = PP$ [1356]. This should not be confused with Aaronson's result **PostBQP** = **PP**, which concerns quantum computation with postselection [1357].

PostBQP is the class of decision problems solvable in quantum polynomial time with postselection and

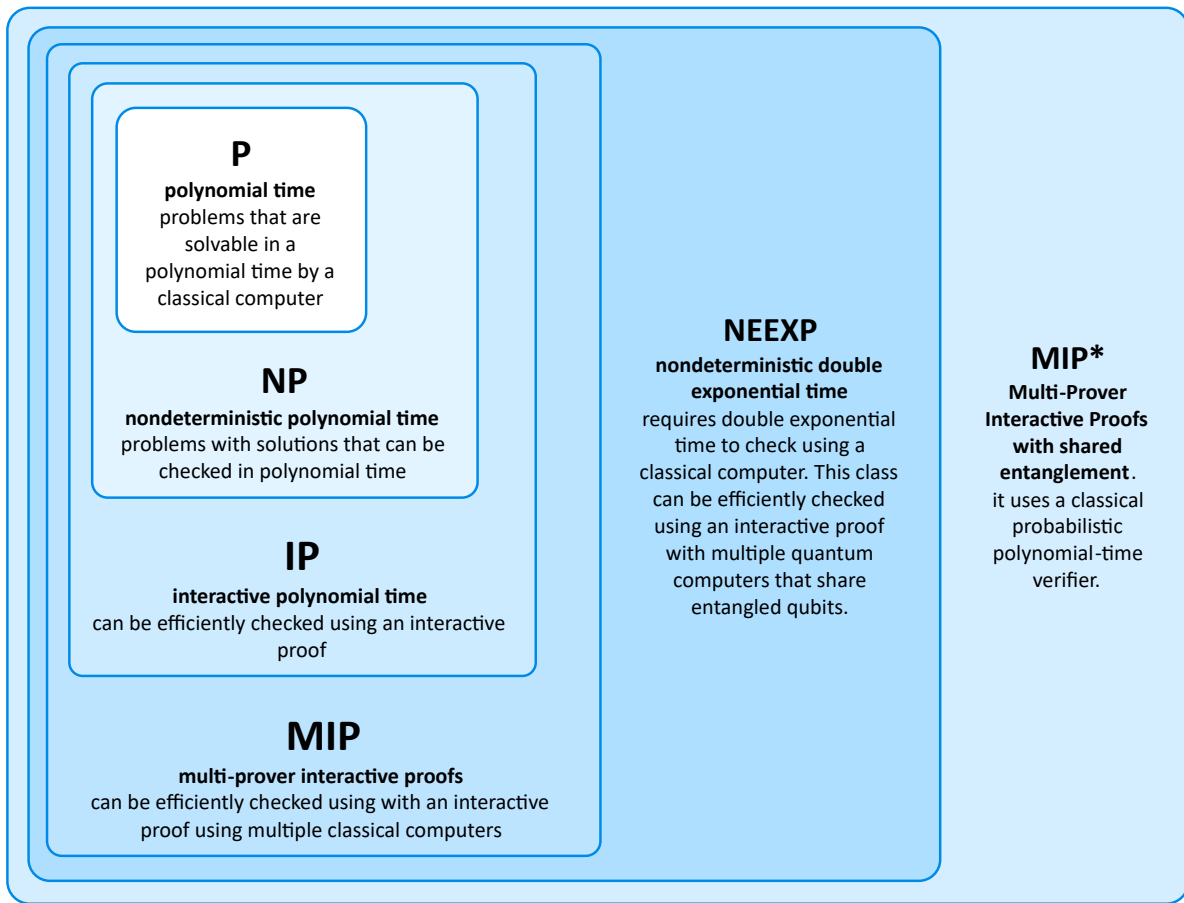


Figure 85: NEEXP and entangled multi-prover interactive proofs. © Olivier Ezratty, 2022-September 2026.

bounded error. Postselection allows conditioning on a specified measurement outcome even when that outcome may have exponentially small probability. Aaronson proved that $\text{PostBQP} = \text{PP}$ [1357].

NISQ is also the name of a formal complexity class introduced by Chen, Cotler, Huang and Li to capture efficient classical computation with access to noisy quantum circuits [1358]. The peer-reviewed paper was published in 2023.

The model satisfies $\text{BPP} \subseteq \text{NISQ} \subseteq \text{BQP}$ by construction, while strictness of these containments is not known unconditionally. Chen et al. provide oracle evidence separating NISQ from both BPP and BQP. In their noise model, unstructured search does not retain Grover's quadratic query advantage, while Bernstein-Vazirani still exhibits a logarithmic query advantage [1358]. These statements concern that formal NISQ model and should not be interpreted as a complete complexity classification of all present-day quantum hardware.

What about the relation between gate-based quantum computing and adiabatic quantum computation? Universal adiabatic quantum computation is polynomially equivalent to the standard circuit model [1359]. This theorem does *not* imply that present commercial quantum annealers, such as D-Wave systems operating with restricted stoquastic annealing Hamiltonians, are compu-

tationally equivalent to a universal gate-based quantum computer.

QMA, Quantum Merlin Arthur, is the quantum analogue of NP in which Merlin supplies a polynomial-size quantum witness and a polynomial-time quantum verifier accepts yes-instances with probability at least $2/3$ while accepting no-instances with probability at most $1/3$. NP and BQP are contained in QMA. A problem is QMA-complete when it is both in QMA and QMA-hard. The canonical k -local Hamiltonian ground-energy decision problem is QMA-complete, including the 2-local case [190]. This does not mean that every electronic-structure calculation is QMA-complete. Quantum phase estimation can estimate eigenvalues to finite precision when one can efficiently implement controlled Hamiltonian time evolutions $U(t) = e^{-iHt}$, or controlled powers of an appropriate unitary. Practical advantage also depends on efficient state preparation, sufficient overlap with the target eigenstate, Hamiltonian simulation cost and the required precision. Actually, while computing a ground state if QMA-complete, its implementation in the QPE separates the prover which is classical and prepares the many-body system state with a good overlap and the verifier which is the QPE and is in BQP. The exponential burden then sits in the classical part of the algorithm.

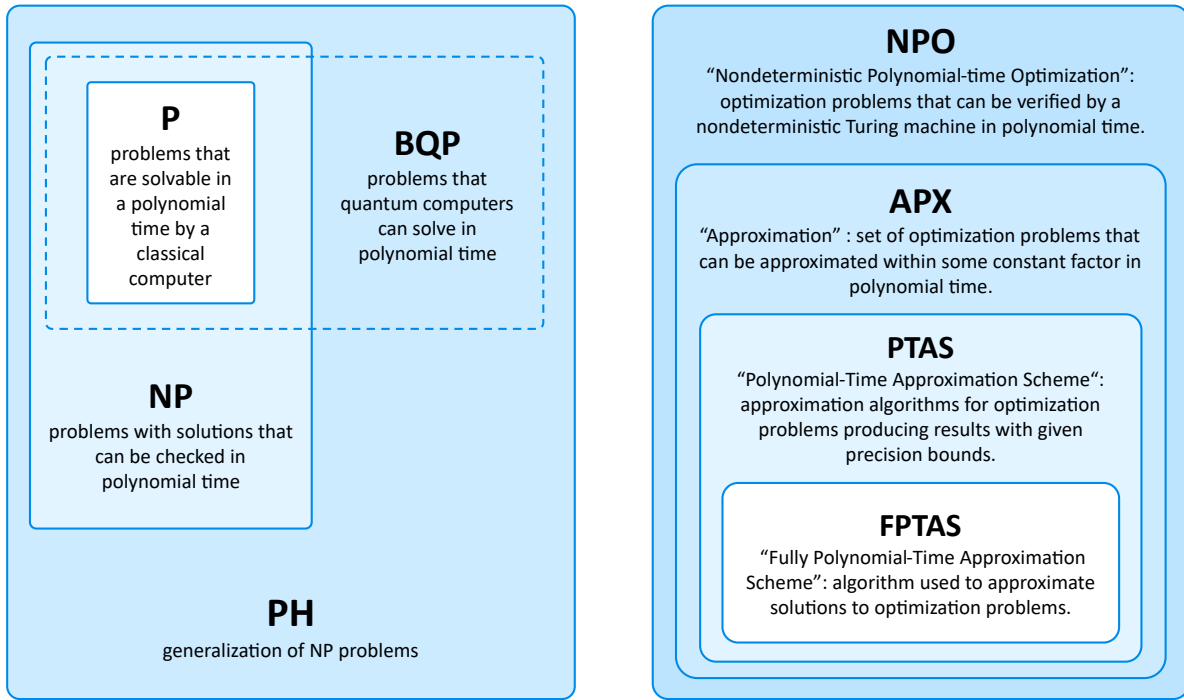


Figure 86: Schematic relation of BQP to selected classical complexity classes. NPO, APX, PTAS and FPTAS are optimization or approximation notions different from decision classes on the left. © Olivier Ezratty, 2022-September 2026.

QCMA is the class in which Merlin supplies a polynomial-size *classical* witness that is checked by a polynomial-time quantum verifier. Thus $\text{NP} \subseteq \text{QCMA} \subseteq \text{QMA}$. Whether $\text{QCMA} = \text{QMA}$ is unknown without oracles. Oracle constructions can separate the two classes relative to specific oracles [1360].

Many publications point out the limitations of quantum algorithms and computers. For decision problems, an asymptotic quantum advantage over efficient randomized classical computation would be exhibited by a problem in BQP that is not in BPP. A separation from PH would be a much stronger result.

For an exponentially large unstructured search space, a quadratic quantum speedup changes $O(2^n)$ scaling only to roughly $O(2^{n/2})$, which remains exponential. This is one reason generic quantum search does not make NP-complete problems efficiently solvable [1361, 1362]. No general polynomial-time quantum algorithms are known for NP-complete problems, although their exclusion from BQP has not been proved.

Finally, **NEEXP**, nondeterministic double-exponential time, is a class of decision problems solvable by a nondeterministic machine in time $2^{2^{p(n)}}$ for some polynomial p . Natarajan and Wright showed that NEEXP is contained in MIP^* , an interactive-proof class in which a polynomial-time classical verifier interacts with multiple noncommunicating provers that may share entanglement (Figure 85) [1363]. This is a statement about efficient verification with powerful provers, not an efficient algorithm for solving NEEXP problems.

There are many other quantum classes like QNC, QAC, Qpoly and QTC that I won't detail here [1364].

Some problems are undecidable, i.e., they cannot be solved by an algorithm, no matter how much time you have. This is the case for the determination of the end of a Turing machine program.

There is no algorithm that decides for every arbitrary program and input whether execution eventually halts. The result $\text{MIP}^* = \text{RE}$ shows that the languages recognizable by classical polynomial-time verifiers interacting with multiple noncommunicating provers sharing entanglement coincide with recursively enumerable languages [1365]. This connects entangled interactive proofs to the computability frontier represented by RE, but it does not make undecidable problems algorithmically solvable.

In the same vein, **Rice's theorem** states that every nontrivial semantic property of the partial function computed by an arbitrary program is undecidable in general. It does not imply that automatic bug detection or program verification is impossible for all restricted languages or specified properties. Static analysis, model checking, proof assistants and formal methods can certify many well-defined properties of specific programs, especially in critical systems such as aerospace.

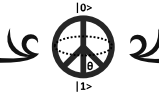
In quantum computing, you can also be exposed to two other complexities:

Krylov complexity measures how an initially simple state or operator spreads over a Krylov basis under dynamics. For a state, the Krylov basis is generated from repeated action of the Hamiltonian followed by Lanczos orthogonalization. For an operator in the Heisenberg picture, the relevant generator is the Liouvillian super-operator $\mathcal{L}(O) = [H, O]$, so the basis is generated by nested commutators rather than by simply applying H

to the operator. Krylov complexity is used to characterize operator growth, quantum chaos and open-system dynamics [1366, 1367].

Kolmogorov complexity is the length of the shortest program, for a fixed universal description language, that

outputs a target object. Different universal machines change the value only by an additive constant, but Kolmogorov complexity is itself uncomputable in general. Quantum analogues can be used to discuss the algorithmic information content or description complexity of quantum states and operations [1368].



1.6.3 Bibliography and notes

These are the bibliographical references and notes for the **Complexity classes** subsection.

- [34] [Quantum Complexity Theory](#), by Ethan Bernstein and Umesh Vazirani, SIAM Journal on Computing, October 1997 (63 pages)
- [190] [The Complexity of the Local Hamiltonian Problem](#), by Julia Kempe, Alexei Kitaev, and Oded Regev, SIAM Journal on Computing, May 2006 (28 pages) (preprint on [arXiv](#)).
- [1327] [Computational Complexity: A Modern Approach](#), by Sanjeev Arora and Boaz Barak, Princeton University, 2007 (489 pages) (preprint on [Stanford](#)).
- [1328] [Computational Complexity: A Modern Approach](#), by Sanjeev Arora and Boaz Barak, Cambridge University Press, January 2007 (489 pages).
- [1329] [Introduction to the Theory of Computation](#), by Michael Sipser, Cengage Learning, February 2012 (482 pages).
- [1330] [The complexity of theorem-proving procedures](#), by Stephen Cook of the University of Toronto, 1971 (8 pages).
- [1331] [Reducibility among combinatorial problems](#), by Richard M. Karp, Complexity of Computer Computations, 1972 (19 pages) .
- [1332] [Complexity and calculability](#), by Anca Muscholl of the LaBRI, 2017 (128 slides).
- [1333] [Fifty Years of P vs. NP and the Possibility of the Impossible](#), by Lance Fortnow, 2021 (11 pages).
- [1334] [The P versus NP problem](#), by Stephen Cook, (12 pages).
- [1335] [A quantum algorithm for solving 0-1 Knapsack problems](#), by Sören Wilkening, Tobias J. Osborne, et al., arXiv, November 2024 (13 pages).
- [1336] [Hybrid classical-quantum branch-and-bound algorithm for solving integer linear problems](#), by Claudio Sanavio, Edoardo Tignone, and Elisa Ercolessi, arXiv, November 2023 (11 pages).
- [1337] [Quantum tree generator improves QAOA state-of-the-art for the knapsack problem](#), by Paul Christiansen, Lennart Binkowski, Debora Ramacciotti, and Sören Wilkening, arXiv, July 2025 (10 pages).
- [1338] [A quantum search method for quadratic and multidimensional knapsack problems](#), by Sören Wilkening and Tobias J. Osborne et al, arXiv, March 2025 (7 pages).
- [1339] [Minesweeper is NP-complete](#), by Richard Kaye, The Mathematical Intelligencer, March 2000 (7 pages) .
- [1340] [Protein Folding in the Hydrophobic-Hydrophilic \(HP\) Model is NP-Complete](#), by Bonnie Berger and Tom Leighton, Journal of Computational Biology, March 1998 (14 pages) .
- [1341] [Some simplified NP-complete graph problems](#), by Michael R. Garey, David S. Johnson, and Larry J. Stockmeyer, Theoretical Computer Science, February 1976 (31 pages).
- [1342] [The NP-Completeness of Edge-Coloring](#), by Ian Holyer, SIAM Journal on Computing, November 1981 (3 pages) .
- [1343] [Calculability and Complexity- Some Results I Know](#), by Etienne Grandjean of the University of Caen, 2017 (43 slides).
- [1344] [A Dynamic Programming Approach to Sequencing Problems](#), by Michael Held and Richard M. Karp, Journal of the Society for Industrial and Applied Mathematics, March 1962 (15 pages) .
- [1345] [Advances in quantum algorithms for the shortest path problem](#), by Adam Wesolowski and Stephen Piddock, arXiv, October 2024 (16 pages).
- [1346] [Classic Nintendo Games are \(Computationally\) Hard](#), by Greg Aloupis, Erik D. Demaine, Alan Guo, and Giovanni Viglietta, arXiv, February 2015 (36 pages).
- [1347] [A competitive NISQ and qubit-efficient solver for the LABS problem](#), by Marco Sciorilli, Giancarlo Camilo, Thiago O. Maciel, Askery Canabarro, Lucas Borges, and Leandro Aolita, arXiv, June 2025 (9 pages).
- [1348] [Relationships between nondeterministic and deterministic tape complexities](#), by Walter J. Savitch, Journal of Computer and System Sciences, April 1970 (16 pages).
- [1349] [Computing a perfect strategy for n x n chess requires time exponential in n](#), by Aviezer S. Fraenkel and David Lichtenstein, Journal of Combinatorial Theory, Series A, September 1981 (16 pages) .
- [1350] [The complexity of computing the permanent](#), by Leslie G. Valiant, Theoretical Computer Science, January 1979 (13 pages).
- [1351] [A Linear-Optical Proof that the Permanent is #P-Hard](#), by Scott Aaronson, arXiv, September 2011 (12 pages).
- [1352] [The quantum smooth label cover problem is undecidable](#), by Eric Culf, Kieran Mastel, Connor Paddock, and Taro Spirig, arXiv, October 2025 (38 pages).
- [1353] [Quantum precomputation: parallelizing cascade circuits and the Moore-Nilsson conjecture is false](#), by Adam Bene Watts, Joseph Slote, et al., arXiv, October 2025 (10 pages).
- [1354] [Oracle Separation of BQP and PH](#), by Ran Raz and Avishay Tal, Journal of the ACM, August 2022 (21 pages) .
- [1355] [On the role of coherence for quantum computational advantage](#), by Hugo Thomas, Pierre-Emmanuel Emeriau, Elham Kashefi, Harold Ollivier, and Ulysse Chabaud, arXiv, October 2024 (20 pages).
- [1356] [Analysis of Quantum Functions](#), by Tomoyuki Yamakami, International Journal of Foundations of Computer Science, October 2003 (38 pages) .
- [1357] [Quantum computing, postselection, and probabilistic polynomial-time](#), by Scott Aaronson, Proceedings of the Royal Society A, September 2005 (10 pages) .
- [1358] [The complexity of NISQ](#), by Sitan Chen, Jordan Cotler, Hsin-Yuan Huang, and Jerry Li, Nature Communications, September 2023 (6 pages).
- [1359] [Adiabatic Quantum Computation is Equivalent to Standard Quantum Computation](#), by Dorit Aharonov, Wim van Dam, Julia Kempe, Zeph Landau, Seth Lloyd, and Oded Regev, SIAM Journal on Computing, January 2007 (29 pages) (preprint on [arXiv](#)).

- [1360] [Separating QMA from QCMA with a classical oracle](#), by John Bostanci, Jonas Haferkamp, Chinmay Nirkhe, and Mark Zhandry, arXiv, November 2025 (91 pages).
- [1361] [The Limits of Quantum Computers](#), by Scott Aaronson, 2008 (16 pages).
- [1362] [NP-complete Problems and Physical Reality](#), by Scott Aaronson, 2005 (23 pages).
- [1363] [NEEXP in MIP*](#), by Anand Natarajan and John Wright, arXiv, September 2019 (122 pages).
- [1364] [The power of shallow-depth Toffoli and qudit quantum circuits](#), by Alex Bredariol Grilo, Elham Kashefi, Damian Markham, and Michael de Oliveira, arXiv, April 2024 (29 pages).
- [1365] [MIP*=RE](#), by Zhengfeng Ji, Anand Natarajan, Thomas Vidick, John Wright, and Henry Yuen, arXiv, November 2022 (223 pages).
- [1366] [Krylov complexity in open quantum systems](#), by Chang Liu, Physical Review Research, August 2023.
- [1367] [Krylov Complexity](#), by Eliezer Rabinovici, Adrián Sánchez-Garrido, Ruth Shir, and Julian Sonner, arXiv, July 2025 (90 pages).
- [1368] [Quantum Kolmogorov complexity and quantum correlations in deterministic-control quantum Turing machines](#), by Mariano Lemus, Ricardo Faleiro, Paulo Mateus, Nikola Paunković, and André Souto, arXiv, January 2024 (29 pages).

This **Complexity classes** subsection contains 44 bibliographical references and notes containing at least 2,625 pages.

1.7 Quantum inspired algorithms

Quantum-inspired algorithms are classical algorithms whose design is inspired by quantum physics methods and interference management, but not programmed as quantum algorithms run through a classical emulator [1369]. Many quantum-inspired algorithms use tensor networks, which we will describe later in this part. Quantum-inspired algorithms can be parallelized in CPU and GPU architectures [1370]. Digital quantum annealers can also be considered as implementing quantum-inspired techniques, but at the hardware level, and classically.

Lecture notes from Waintal et al. present these techniques as plain linear algebra on exponentially large vectors and matrices, deliberately decoupled from the many-body physics context where they were invented. It covers MPS and MPO algorithms, DMRG, tensor cross interpolation (TCI) which learns an MPS representation of a known function, and the “quantics” representation, with three worked applications: emulating a quantum computer, emulating a quantum annealer, and solving Poisson, diffusion and Gross-Pitaevskii equations [1371]. Annealing-inspired classical solvers have their own 2026 coverage, with simulated bifurcation applied to quantum annealing [1372], multi-objective variants [1373] and NISQ-era annealing-inspired heuristics [1374].

Creating a quantum-inspired algorithm is sometimes said to rely on “dequantizing” or the “de-quantization” of a quantum algorithm [1375]. There is even a quantum-inspired version of the NISQ QAOA algorithm [1256]. As Alastair Abbott and Cristian S. Calude wrote in 2010, “*de-quantization helps formulate conditions to determine if a quantum algorithm provides a real speed-up over classical algorithms. These conditions can be used to develop new quantum algorithms more effectively (by avoiding features that could allow the algorithm to be efficiently classically simulated), as well as providing the potential to create new classical algorithms (by using features which have proved valuable for quantum algorithms)*”. In the framework studied in their paper, they found that algorithms whose entanglement remains suitably bounded can be de-quantized [1376]. This is not a universal criterion for classical simulability. For example, fermionic Gaussian states can be highly entangled and still admit efficient classical simulation because they have a compact covariance-matrix representation [1377].

A 2026 result illustrates the mechanism. Hastings’ short-path algorithm, a variant of adiabatic optimization analyzed by Dalzell et al. in 2023, was believed to solve MAX- k -CSP constraint satisfaction problems in $2^{(1-c)n/2}$ time, suggesting a super-quadratic advantage. In 2026, Le Gall and Tamaki identified the purely classical conditioning-and-search mechanism hidden in that analysis and derived classical algorithms running in $2^{(1-c')n}$ with $c' > c$, removing the claimed super-quadratic separation and, as a byproduct, producing

a new quantum-inspired classical technique for these problems [1378].

1.7.1 Applications

Quantum inspired algorithms can be helpful in multiple domains:

- Recommendation systems, like with the famous solution from Ewin Tang in 2018 [761].
- Various machine learning tasks like images classification [1379], and others [1380–1383]. A 2026 critical review from Guillermo Valverde and Alejandro Pozas-Kerstjens covers the two ways tensor networks enter machine learning, either as standalone learning architectures or as decompositions of existing neural network layers, and assesses where the claimed gains in efficiency, explainability and privacy actually hold [1384]. Other 2026 examples include synthetic aperture radar (SAR) object classification with tensor networks [1385] and data re-uploading networks transposed to classical hardware [1386]. A notable dequantization result targets the quantum ridgelet transform, a QML algorithm that samples hidden nodes to extract a sparse trainable subnetwork (a “winning lottery ticket”) from a large shallow network in $O(D)$ time in the data dimension D , where the naive classical version costs $\exp[O(D)]$. A fully classical sampler achieves the same task in $O(\text{poly}(D))$, removing the exponential dependence and with it the claimed quantum speedup [1387].
- Optimizing and compressing large language models (LLMs) and convolutional neural networks [1388–1390]. Quantum inspired self-attention has also been dropped into a full autoregressive language modeling pipeline, with reported gains of $15.5\times$ on character error rate and $13\times$ on cross-entropy loss versus standard attention, at the cost of $2.6\times$ longer inference. The result is however obtained on a GPT-1 scale model, which is three orders of magnitude below current frontier models, so it says little about behavior at scale [1391].
- Solving linear algebra problems [1392–1394], Partial Differential Equations (PDEs) [1395] and fluid dynamics simulations [1396–1398]. The PDE work increasingly relies on the **quantics** representation, described later in this part, where tailoring the tensor network algorithms to the scale hierarchy instead of using stock DMRG brings faster and more robust convergence on Poisson and Schrödinger problems in two, three and four dimensions, with simulations reaching 2^{80} grid points [1399]. Non-unitary transforms are also within reach, unlike on a gate-based quantum computer. A discrete Laplace transform was decomposed into a non-unitary damping transform followed by a QFT, both compressed into a single MPO, handling $N = 2^{30}$ input points and up

to 2^{60} output points [1400]. There is even a 2026 proposal for the dequantization of QSVT [1401].

- Pretraining variational ansatz circuits initialization using tensor networks and conversion to quantum circuits to avoid barren plateaus [1178].
- Simulating quantum systems [1402, 1403], simulating molecular excited states [1404], quantum dynamics simulations [1405], simulating material deformations [1406], simulating spin-boson models with tensor networks using process trees to simulate non-Markovian quantum processes with long-range correlations [1407], spin model simulations [1408], for computing the ground states of frustrated Hubbard and Heisenberg models which helps study superconducting materials, including high T_c materials (high critical temperature superconductors) [1409], for solving the Gross-Pitaevskii nonlinear partial differential equation that is used for simulating the dynamics of a Bose-Einstein condensates and also for studying superfluidity [1410], and even to simulate fermionic many-body systems [1411–1413]. In 2026, Neural quantum states (NQS) with a residual convolutional architecture simulated real-time quench dynamics of the 3D transverse-field Ising model on lattices of up to 1,000 spins, a record for real-time dynamics beyond 1D, and the first large-scale numerical demonstration of the 3D quantum Kibble-Zurek mechanism [1414]. And a three-level tensor network compression, combining a global purification of the density matrix, compression of inter-mode entanglement and an encoding of Fock occupation numbers, simulated the Lindblad dynamics of superconducting circuits with arbitrarily high mode occupations. It handled large cat qubits and transmon ionization, bringing a computation previously run on a supercomputer down to laptop level [1415]. Trajectory-based MPS simulations of open systems did show that physically equivalent stochastic versions of the same Lindblad dynamics do not reduce total cost but redistribute it between per-trajectory bond dimension and the number of trajectories needed for statistical convergence [1416].
- Emulating or replacing flagship quantum algorithms. In 2026 altogether, an iterative MPS formulation of Grover reused a single Grover operator instead of materializing $O(\sqrt{N})$ layers, running about $15\times$ faster than the plain Grover circuit on an MPS backend at 29 qubits under NVIDIA CUDA-Q [1417], a Gaussian boson sampling algorithm for graph isomorphism was dequantized into a classical test that uses efficiently computable statistical properties of the sampler to rule out isomorphism [1418], and a GPU-parallel iterative qubit coupled cluster (iQCC) solver emulated 100 to 124 qubit chemistry circuits for ruthenium catalysts in 1.2 to 45 hours, beating DMRG accuracy, which pushed the chemistry advantage frontier well past the commonly assumed 50-qubit mark [1419].

- Reconstructing (or learning) quantum states from experimental data using Matrix Product Operators (MPOs) [1420].
- Noise characterization of quantum computers beyond the capabilities of quantum tomography [1421], and learning hardware Hamiltonians, as done for the interactions between Rydberg atoms in neutral atom processors [1422].
- Operational research and combinatorial optimizations including search and solving traveling salesperson problems (TSP) [1423–1428]. Quantinuum benchmarked product state and MPS ansatz minimization of a quantum annealing Hamiltonian, embedded in an iterated local search metaheuristic, on maximum cut instances of up to 50,000 variables, outperforming classical iterated local search, QAOA and other variational quantum-inspired solvers [1429]. Other variants use qudit-based imaginary time evolution [1430] and target Sudoku and MaxCut [1431], while noncommutative polynomial optimization provides scalable certification of quantum spin system ground states [1432].
- Optimizing energy grids as in [1433].
- Weather forecasts, as planned by Planette.ai (2022, USA, \$2.4M), a company specialized in long-range weather prediction technology, that was selected in August 2025 by NASA for the development of Qubit-Cast, a new AI and quantum-inspired weather forecasting system.
- In cryptocurrencies management, for categorizing Bitcoin addresses [1434].
- Cybersecurity and threat detection [1435].

In industry markets, quantum software startups find in quantum-inspired algorithms a way to monetize their know-how while waiting for sufficiently powerful quantum computers. There are applications in various markets:

- Financial services with portfolio optimization and options pricing [1436], with applications developed by QuantFi and Multiverse Computing.
- Chemistry [1437, 1438]. Quantum inspired algorithms even exist that transpose the complicated gaussian boson sampling photonic based technique to simulate molecular vibronic spectra. Molecular vibronic spectra come from light absorption that depends on the transitions between the different electronic states of the molecule and on changes in its vibrational energy [1439]. This is the domain where the classical baseline moves fastest. Nvidia along with PNNL and others reported in 2026 mixed-precision spin-adapted ab initio DMRG on Nvidia Blackwell GPUs, interfaced with ORCA, delivering high accuracy ground state energies for the Fe_4S_4 cluster on

a CAS(54,36) model space, a problem listed on the IBM and RIKEN Quantum Advantage Tracker, plus CAS-SCF orbital optimizations up to CAS(89,102) for $\text{Fe}_5\text{S}_{12}\text{H}_4^{5-}$. Their conclusion was that DMRG numbers should be the mandatory classical reference whenever quantum advantage is claimed, and even the best DMRG codes are still far from exploiting GPUs fully [1440]. The same message came from the parallel iQCC results on ruthenium catalysts [1419]. Supporting tooling includes the ten-year PySCF retrospective [1441], the GPU-accelerated ByteQC package [1442], stochastic tensor contraction [1443], tree tensor network states computing thousands of vibrational eigenstates in full dimensionality [1444] and quantum-inspired Ising machines applied to electronic structure [1445].

- Healthcare with applications developed by many startups like Qubit Pharmaceuticals, Aqemia and Rahko in chemical simulations.
- Cybersecurity [1446] and manufacturing [1447].

That is fine, unless the related communication is misleading. In many cases, you must read the fine print to discover that a « quantum algorithm » or a “quantum computing use case” solving such and such problem is a classical quantum-inspired solution.

1.7.2 Tensor networks

In our presentation of basic linear algebra, we described the tensor product, or Kronecker product, used to combine the Hilbert spaces of several quantum systems. An N -qubit pure state is represented by a vector of 2^N complex amplitudes. Tensors are also heavily used in classical deep learning, including with the famous TensorFlow SDK from Google (which is progressively replaced by the JAX framework).

Describing a generic many-body quantum state requires exponentially many amplitudes and is therefore hard to compute classically. Tensor networks factorize these large tensors into networks of smaller tensors and can “zip” their representation. For an MPS of N sites with local dimension d and a typical bond dimension χ , storage scales roughly as $O(Nd\chi^2)$ instead of $O(d^N)$. This can provide an exponential reduction in parameters and operations when χ remains moderate [1448].

One important reason why tensor networks can efficiently represent many physically relevant states is the so-called area law. For ground states of many local Hamiltonians, the entanglement entropy $S(A)$ of a spatial region A scales approximately with the size of its boundary $|\partial A|$, rather than with its volume $|A|$ [1449]. For an MPS with bond dimension χ , the entanglement entropy across a cut is bounded by roughly $\log \chi$. Area-law states can therefore often be represented with a moderate bond dimension, whereas volume-law entanglement generally requires χ to grow exponentially with

system size. Area laws constrain the classical compressibility of particular quantum states. They do not imply that entanglement exists only between neighbors, nor that highly entangled states cannot exist [1449–1451].

A second compression mechanism is not related to quantum physics. In the quantics representation, also labeled quantized tensor train (QTT), a continuous variable discretized on 2^n grid points is encoded on n binary digits, each digit supported by one chain tensor. Digit 1 sets the coarsest scale and digit n the finest, so the tensor network indexes scales rather than particles, and the number of tensor indices grows only logarithmically with the number of grid points. Functions that are smooth at each scale can acquire a low bond dimension, which is why the technique fits multi-scale PDEs. The resulting computational cost is not generically logarithmic, since it also depends on the QTT ranks. It is favorable only when they remain sufficiently small. Unlike the many-body case where all spins play equivalent roles, the degrees of freedom are not equivalent here, and adapting the algorithms to that hierarchy following multigrid methods speeds up convergence significantly [1371, 1399]. Tensor cross interpolation (TCI) is the companion learning algorithm that builds such an MPS from a black-box function without ever enumerating its 2^n values [1371].

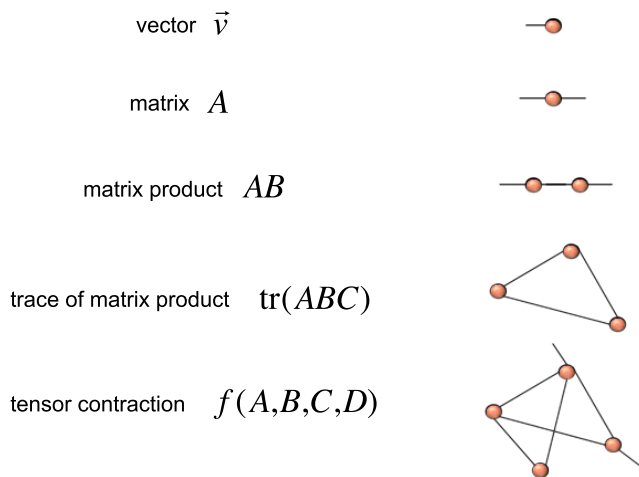


Figure 87: Basics of tensor networks operations. Source: [1452]

Tensor networks have various use-cases in many-body quantum physics digital simulations, for classical simulations of quantum computers, in chemistry, as well as in machine learning and applied mathematics [1453]. This is an entirely new world of mathematics.

How is it connected to quantum computing [1454]? Most of the tensor network literature deals with classical computing but tensor matrix products can be implemented with quantum computing [1455, 1456].

Some companies like Multiverse Computing make a lot of use of tensor networks techniques in quantum-inspired algorithms, including for the optimization of classical

LLM trained models. Classical tensor networks are also used in quantum code emulators simulating noisy qubits.

A 45-page 2026 review from Multiverse goes over the IBM, D-Wave and Google quantum advantage experiments and on the tensor network simulations that were built to reproduce them. It identifies the regimes and system features that remain hard for current tensor network methods, which is also a map of where a defensible advantage claim could still be made [1457]. Each advantage claim resets a classical benchmark that then has to be re-established, so the claim is only as strong as the classical effort spent against it.

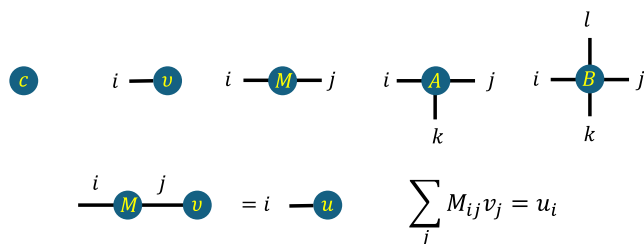


Figure 88: Tensor network operations visualization. Source: [1458].

In the graphical notation shown in Figure 88, a node represents a tensor and each edge represents one of its indices. An edge joining two nodes denotes an index contracted between the corresponding tensors, while a dangling edge denotes a free index. Vectors and matrices are rank-1 and rank-2 tensors, respectively, and larger networks represent successive tensor contractions.

Then, tensor network techniques are represented with these graphical views. As shown in Figure 89, the main techniques are the following, keeping in mind that MPS, TTN, PEPS, TPS and MERA are variational ansätze, meaning families of states, whereas DMRG is the optimization algorithm that is usually run on top of an MPS ansatz:

- **MPS** (Matrix Product States), represented in a 1D series of links between serialized tensors [1459]. Their infinite and boundary variants are still being formalized, as in a 97-page 2025 study of uniform MPS with a boundary from Ignacio Cirac and David Pérez-García [1460].
- **DMRG**, introduced by Steven R. White in 1992, is today usually formulated as a variational optimization over an MPS ansatz. It sweeps through the tensor network while optimizing one or two local tensors at a time. Schmidt decompositions, usually implemented through singular value decompositions (SVD), truncate the bond dimension while retaining the dominant correlations and entanglement across each cut [1461, 1462]. The Clifford Augmented DMRG variant (CA-DMRG) is a method used in quantum chemistry that integrates Clifford transformations that can be efficiently emulated. It helps simulate more complex electron interactions with higher accuracy and lower computational cost

than standard DMRG in the regimes studied [1463]. In 2026, Örs Legeza, John Gunnels and collaborators reported high-accuracy DMRG calculations for Fe_4S_4 in a $\text{CAS}(54,36)$ space and CAS-SCF orbital optimizations up to $\text{CAS}(89,102)$ for $\text{Fe}_5\text{S}_{12}\text{H}_4^{5-}$, within a full orbital space containing 331 electrons in 451 orbitals [1440]. DMRG is now a key classical reference for strongly correlated electronic-structure benchmarks, alongside other state-of-the-art classical methods depending on the physical regime.

- **TTN** (Tree Tensor Networks), tensor networks with hierarchical tree connectivity that can efficiently capture correlations without loops. They are used for many-body quantum simulations, including some 2D problems [1464, 1465], and in molecular spectroscopy to compute thousands of vibrational eigenstates in full dimensionality [1444].
- **PEPS** (Projected Entangled Pair States), a generalization of MPS to higher-dimensional lattice connectivity, most notably 2D lattices. A site tensor typically carries one physical index and several virtual indices connecting it to neighboring sites. PEPS and related 2D tensor-network contractions are useful to simulate quantum circuits, as in [1466], which studies a local unitary cluster Jastrow (LUCJ) ansatz reaching 52 qubits for an N_2 molecule and 72 qubits for a 4Fe-4S molecule, with the largest circuit involving roughly 1800 $XX + YY$ rotations, and the discrete-time dynamics of a domain-wall quench within a 2D Heisenberg model, up to 164 qubits on an IBM heavy-hex architecture and 105 qubits on the Google Willow processor topology.
- **TPS** (Tensor Product States) [1467].
- **MERA** (Multiscale Entanglement Renormalization Ansatz), which can also be used in quantum computing [1468].

These tools use the technique of tensor networks contraction [1469]. The graphical representation of gate-based quantum circuits happens to be a special case of tensor networks. Marc-André Desrosiers explains how we build tensor network algorithms using these representations [1470]. The ZX Calculus graphical language is also derived from the tensor network formalism [1471].

Contraction remains the practical bottleneck for 2D and beyond, where exact contraction is exponential. Recent work improves the approximation side with partitioned expansions of approximate contractions from Evenbly, Gray and Chan [1472], replica tensor trains [1473], sketched Hadamard products of tensor trains [1474] and provable convergence guarantees derived from algorithmic locality [1475]. Shadow tomography ideas have also been ported to classical tensor network simulations to estimate many observables from few contractions [1476].

Other known contributors to the development and use of MPS are Edwin Miles Stoudenmire, who did his PhD

Tensor Networks

e.g. *RO, Annals of Physics* **349** (2014) 117–158

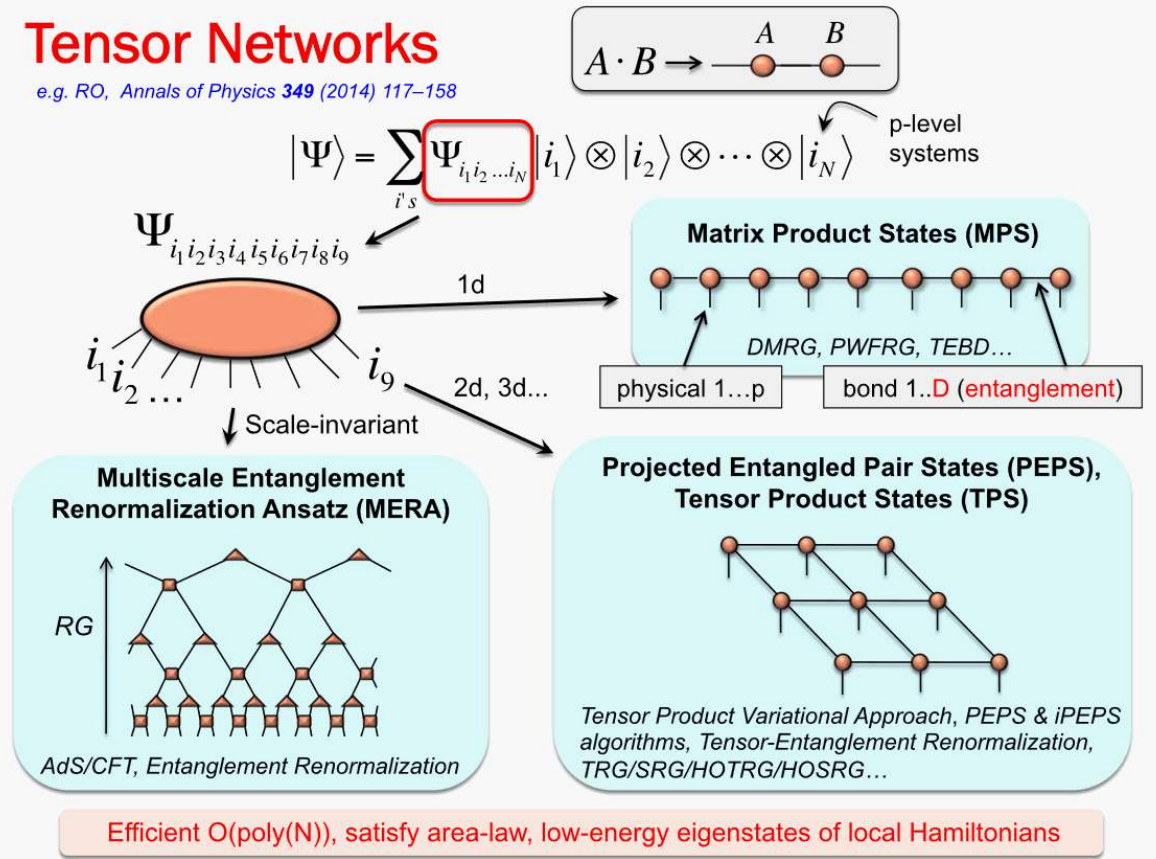


Figure 89: How tensor networks are graphically represented using the above notation.

with Steven White, and Antoine Georges, both from the Flatiron Institute in New York, and Ulrich Schollwöck from LMU München who also contributed to the development of DMRG.

Tensor Networks can be used to optimize a variational e^{-iHt} Hamiltonian time evolution simulations quantum circuits. A 2025 paper describes the method that decomposes a 1D quantum state into subproblems with infinite MPS (iMPS). The quantum part of the solution is based on a variational ansatz that is optimized with gradient-based methods like L-BFGS (Limited-memory Broyden-Fletcher-Goldfarb-Shanno). It is using unit-cell, composed of parameterized 2-qubit gates or specific Rz-based gates. This provides a $5.2\times$ reduction in T-count. The solution cannot however be generalized since it works only on 1D problems and its quantum advantage is not demonstrated [1477].

Tensor Network methods can also be used *within* quantum circuits. A proposal from China uses a quantum version of matrix product state (qMPS) that is embedded in a variational quantum algorithms. Positioned as a quantum version of the DMRG method, it can reduce the number of qubits used vs a VQE and could help distribute algorithms across multiple QPUs although such decompositions cast doubts on any potential quantum advantage [1478].

About 80 classical tools implement Tensor networks like **Google TensorNetwork** which relies on Tensor-

Flow and was released in 2019 [1479] or the **QTensor** library from the Argonne National Laboratory that is used for quantum circuits emulations [1480, 1481]. Before quantum computers bring some quantum advantage, tensor network based techniques remain key contenders to enable classical many-body simulations, particularly thanks to many architectural and implementation improvements, particularly around DMRG based techniques [1482].

The dispersion is real and now measured. A 2026 survey counts over 50 DMRG implementations alone and benchmarks eight of them under a common performance-oriented framework, finding that comparable codes differ by large factors on identical problems, which is a warning for anyone using a single tensor network baseline to rebut a quantum advantage claim [1483]. On the symbolic side, SeQuant automates the derivation and numerical evaluation of tensor algebra expressions used in many-body methods [1484].

After some time spent understanding how tensor networks work [1485–1489], you may wonder where quantum computing plays a role here. Tensor networks are used as design and analysis tools for quantum error correction codes, topological computing, condensed matter physics [1490] and quantum machine learning algorithms [1491]. They are also used in quantum circuit emulation, where tensor contractions can avoid storing a full state vector or density matrix [1492]. Their limits be-

come apparent for sufficiently large and deep circuits whose relevant tensor-network bond dimensions grow too rapidly. FTQC should eventually reach regimes that cannot be simulated end-to-end this way, but tensor networks will remain relevant for restricted circuits, verification, state preparation, error-correction research and many-body calculations.

Xanadu’s PennyLane framework contains tensor network circuit templates [1493]. Zapata Computing is also working on using tensor networks to prepare parametrized quantum circuits used in hybrid algorithms [1494]. Tensor networks may soon benefit from optimization techniques like the one developed by DeepMind, AlphaTensor [1495]. Tensor networks can also be combined with quantum circuits in a hybrid manner [1496].

Other tensor networks use cases include the ansatz preparation of chemical simulation using MPS [1497, 1498] and the creation of explainable and efficient classical machine learning solutions [1499].

Finally, tensor networks and other quantum-inspired techniques can modify the complexity comparison between classical and quantum algorithms. A 2024 paper focusing on linear regression, supervised clustering, principal component analysis, recommendation systems and Hamiltonian simulation proves quadratic lower bounds in the Frobenius norm of the underlying matrix for the quantum inspired classical access model it studies, while the corresponding quantum complexities are linear in that norm [1500]. This establishes that a quadratic quantum-classical separation can be unavoidable in that model, but it does not upper-bound the possible quantum speedup to be only polynomial. Also, many dequantization results show that apparent exponential speedups can collapse to polynomial speedups for structured inputs, including some low-rank matrix problems, when comparable data-access assumptions are imposed on the classical algorithm.

1.7.3 Other quantum inspired techniques

Pauli propagation is a technique that we already mentioned in the emulator subsection. It is another quantum inspired technique that extends the reach obtained with MPS and DMRG methods [1501]. Instead of tracking the state, it evolves the observable in the Heisenberg picture as a sum of Pauli strings and truncates the strings that contribute negligibly, so the cost is driven by the number of retained strings rather than by entanglement.

The 2026 extensions go in three directions. Symmetry merging exploits the fact that, when the circuit has a symmetry, many Pauli strings evolve redundantly, so only one representative per group orbit needs to be propagated, which cuts memory by a factor set by the orbit size and improves stability under truncation and noise [1502]. The method has been ported from real-time to imaginary-time evolution, which targets ground states rather than dynamics [1503], and to real-time correla-

tors, the quantities that feed spectroscopy [1504]. A vectorization formulation recasts the whole Heisenberg-picture simulation in a form that unifies these variants [1505].

Majorana propagation is the fermionic counterpart, applying the same successive truncations to Heisenberg evolution but on Majorana monomials instead of Pauli strings, with monomial length as the truncation criterion. Algorithmiq built ADAPT-VMPE on top of it, a fully classical eigensolver that constructs ground state ansätze iteratively with polynomial cost in qubit count and iteration count under its truncation and approximation assumptions, and generated up to 100-qubit ansätze for a strongly correlated photosensitizer currently in human clinical trials for cancer treatment [1506].

Matchgate circuits define a third classical simulability class, distinct from stabilizer circuits and from low-entanglement tensor networks. They generate fermionic Gaussian states, which can be highly entangled and yet remain classically tractable because they are fully described by a covariance matrix of size $2N$ by $2N$ instead of an exponential vector state. A 2026 paper derives optimal gate counts for preparing an arbitrary pure fermionic Gaussian state, gives circuits that saturate those bounds, and introduces a classical simulation algorithm that manipulates the generating circuits directly, plus an extension to t -doped Gaussian circuits where a few non-matchgate gates are added [1377]. The doping parameter plays a role analogous to T -count for Clifford circuits, providing a tunable measure of departure from an efficiently simulable class.

Neural quantum states (NQS) are another interesting quantum inspired technique. They were originally developed as parameterized neural-network representations of complex quantum wavefunctions $\psi_\theta(x)$, including both amplitude and phase information. Their Born probabilities $|\psi_\theta(x)|^2$ define high-dimensional probability distributions that can be sampled with suitable architectures. NQS are used to simulate quantum systems on classical hardware, enabling approximate solutions to many-body problems in condensed matter physics [1507–1509], including Hubbard models [1510]. Related neural architectures, such as restricted Boltzmann machines (RBM, [1511]) and autoregressive models, are also used for Monte Carlo sampling and variational inference in statistical physics.

Two 2026 results framed what NQS can and cannot do. On the capability side, transformer wave functions applied to the J_1 - J_2 Heisenberg model on triangular and square lattices of up to 20 by 20 sites show that the V-score, a variational accuracy measure, decays as a power law in training compute, and that curves for different system sizes collapse onto a single one after rescaling, so the transformer ansatz is size-consistent for these systems [1512, 1513]. On the limitation side, Nisarga Paul proved that feed-forward NQS acting on n spins with k scalar nonlinearities obey $S \leq ck \log n$ on the entanglement entropy of any subregion, under analyticity assumptions.

This is the NQS analog of an entanglement-capacity bound and it rules out volume-law entanglement for NQS with $O(1)$ nonlinearities [1514]. The practical reading is that NQS are not an unconditional escape from the entanglement barrier. Increasing their entanglement capacity requires increasing network complexity, through depth, width, the number of nonlinear units or the architecture itself. NQS have also been benchmarked in non-stabilizer regimes on atomic nuclei, where magic rather than entanglement is the limiting resource [1515], and an 85-page review covers the nuclear many-body applications [1516].

These examples also show that entanglement alone does not determine classical simulation hardness. MPS methods are often limited by bond-dimension growth and entanglement, Pauli propagation by the proliferation of relevant Pauli strings, fermionic Gaussian simulation by departures from Gaussianity, and neural quantum states by their representational and optimization complexity. Different classical methods therefore fail at different boundaries, which is why quantum advantage must always be compared against the strongest classical method adapted to the structure of the problem.

Quantum dynamics simulations using the space evolving block decimation (SEBD) method exploit the “*system causal light-cone structure by interleaving time and space evolution and applying projective measurements as soon as local subsystems reach the target physical time, suppressing entanglement growth*”, as proposed in [1517]. This mitigates entanglement growth in the time-evolved state, leading to smaller tensor network bond dimensions. The mechanism is conceptually related to the causal locality captured by Lieb-Robinson bounds, which we described in a subsection from the quantum engineering part of this book.

Entanglement barriers provide the complementary theoretical picture. A 2026 study maps satisfiability problems onto MPS evolution and shows that the entanglement growth that kills the simulation is itself dictated by the computational complexity of the instance, which turns bond dimension into a complexity diagnostic rather than a mere numerical nuisance [1518]. On the algorithmic side, imaginary time evolution, the workhorse for reaching ground states classically, has been accelerated with implicit stepping on logarithmic grids by Zima et al. [1519].



1.7.4 Bibliography and notes

These are the bibliographical references and notes for the **Quantum inspired algorithms** subsection.

- [761] [A quantum-inspired classical algorithm for recommendation systems](#), by Ewin Tang, arXiv, May 2019 (32 pages).
- [1178] [Synergistic pretraining of parametrized quantum circuits via tensor networks](#), by Manuel S. Rudolph, Jacob Miller, Danial Motlagh, Jing Chen, Atithi Acharya, and Alejandro Perdomo-Ortiz, Nature Communications, December 2023.
- [1256] [Mean-Field Approximate Optimization Algorithm](#), by Aditi Misra-Spieldenner, Frank K. Wilhelm, et al., arXiv, September 2023 (17 pages).
- [1369] [Quantum-inspired algorithms in practice](#), by Juan Miguel Arrazola, Alain Delgado, Bhaskar Roy Bardhan, and Seth Lloyd, arXiv, August 2020 (24 pages).
- [1370] [Massively Parallel Tensor Network State Algorithms on Hybrid CPU-GPU Based Architectures](#), by Andor Menczer and Örs Legeza, arXiv, May 2023 (18 pages).
- [1371] [Who can compete with quantum computers? Lecture notes on quantum inspired tensor networks computational techniques](#), by Xavier Waintal, Chen-How Huang, and Christoph W. Groth, arXiv, January 2026 (69 pages).
- [1372] [Simulated Bifurcation Quantum Annealing](#), by Jakub Pawłowski, Paweł Tarasiuk, Jan Tuziowski, Lukasz Paweł, and Bartłomiej Gardas, arXiv, April 2026 (18 pages).
- [1373] [Multi-Objective Optimization by Quantum-Annealing-Inspired Algorithms](#), by Xian-Zhe Tao, Pavel Mosharev, and Man-Hong Yung, arXiv, April 2026 (14 pages).
- [1374] [Quantum annealing inspired algorithms for the NISQ Era](#), by Rijul Sachdeva, Vrinda Mehta, Manpreet Singh Jattana, Kristel Michielsen, and Fengping Jin, arXiv, April 2026 (10 pages).
- [1375] [Sampling-based sublinear low-rank matrix arithmetic framework for dequantizing quantum machine learning](#),

by Nai-Hui Chia, Ewin Tang, Chunhao Wang, et al., arXiv, July 2023 (77 pages).

- [1376] [Understanding the Quantum Computational Speed-up via De-quantisation](#), by Alastair A. Abbott and Cristian S. Calude, arXiv, June 2010 (12 pages).
- [1377] [Matchgate circuit representation of fermionic Gaussian states: optimal preparation, approximation, and classical simulation](#), by Marc Langer, Raul Morral-Yepes, Adam Gammon-Smith, Frank Pollmann, and Barbara Kraus, arXiv, March 2026 (39 pages).
- [1378] [Dequantizing Short-Path Quantum Algorithms](#), by François Le Gall and Suguru Tamaki, arXiv, April 2026 (45 pages).
- [1379] [AutoQML: Automatic Generation and Training of Robust Quantum-Inspired Classifiers by Using Genetic Algorithms on Grayscale Images](#), by Sergio Altares-López, Juan José García-Ripoll, and Angela Ribeiro, arXiv, August 2022 (13 pages).
- [1380] [Robust Dequantization of the Quantum Singular value Transformation and Quantum Machine Learning Algorithms](#), by François Le Gall, arXiv, December 2024 (56 pages).
- [1381] [Quantum-Inspired Machine Learning: a Survey](#), by Larry Huynh, Jin Hong, Ajmal Mian, Hajime Suzuki, Yanqiu Wu, and Seyit Camtepe, arXiv, September 2023 (59 pages).
- [1382] [Exploring Tensor Network Algorithms as a Quantum-Inspired Method for Quantum Extreme Learning Machine](#), by Payal D. Solanki and Anh Pham, arXiv, April 2025 (15 pages).
- [1383] [Quantum-Inspired Weight-Constrained Neural Network: Reducing Variable Numbers by 100x Compared to Standard Neural Networks](#), by Shaozhi Li, M Sabbir Salek, Binayak Roy, Yao Wang, and Mashrur Chowdhury, arXiv, December 2024 (13 pages).
- [1384] [Quantum-inspired tensor networks in machine learning models](#), by Guillermo Valverde, Igor Garcia-Olaizola, Gi-

- annicola Scarpa, and Alejandro Pozas-Kerstjens, arXiv, April 2026 (28 pages).
- [1385] [Quantum-Inspired Robust and Scalable SAR Object Classification](#), by Maximilian Scharf, Marco Trenti, Felix Bock, Pdraig Davidson, Tobias Brosch, Benjamin Rodrigues de Miranda, Sigurd Huber, and Timo Felser, arXiv, April 2026 (6 pages).
 - [1386] [Q-RUN: Quantum-Inspired Data Re-uploading Networks](#), by Wenbo Qiao, Shuaixian Wang, Peng Zhang, Yan Ming, and Jiaming Zhao, arXiv, December 2025 (24 pages).
 - [1387] [Winning Lottery Tickets in Neural Networks via a Quantum-Inspired Classical Algorithm](#), by Natsuto Isogai, Hayata Yamasaki, Sho Sonoda, and Mio Murao, arXiv, May 2026 (28 pages).
 - [1388] [CompactifAI: Extreme Compression of Large Language Models using Quantum-Inspired Tensor Networks](#), by Andrei Tomut, Samuel Mugel, Roman Orus, et al., arXiv, May 2024 (5 pages).
 - [1389] [Hyper Intelligence Releases Quantum-Inspired Algorithm Designed to Reduce Cost of LLMs](#), by Matt Swayne, The Quantum Insider, November 2023.
 - [1390] [Tensor network compressibility of convolutional models](#), by Sukhbinder Singh, Saeed S. Jahromi, and Roman Orus, arXiv, August 2024 (40 pages).
 - [1391] [Quantum-Inspired Self-Attention in a Large Language Model](#), by Nikita Kuznetsov, Niyaz Ismagilov, and Ernesto Campos, arXiv, February 2026 (8 pages).
 - [1392] [Towards Optimizing the Expected Performance of Sampling-Based Quantum-Inspired Algorithms](#), by Hyunho Cha and Jungwoo Lee, arXiv, January 2025 (29 pages).
 - [1393] [An improved quantum-inspired algorithm for linear regression](#), by András Gilyén, Zhao Song, and Ewin Tang, arXiv, June 2022 (21 pages).
 - [1394] [Linear algebra at exponential scale via tensor network dimension reduction](#), by Chris Camano, Ethan N. Epperly, Raphael A. Meyer, and Joel A. Tropp, arXiv, June 2026 (44 pages).
 - [1395] [Fast and Flexible Quantum-Inspired Differential Equation Solvers with Data Integration](#), by Lucas Arenstein, Martin Mikkelsen, and Michael Kastoryano, arXiv, May 2025 (23 pages).
 - [1396] [Complete quantum-inspired framework for computational fluid dynamics](#), by Raghavendra D. Peddinti, Leandro Aolita, et al., arXiv, September 2023 (13 pages).
 - [1397] [Technical report on a quantum-inspired solver for simulating compressible flows](#), by Raghavendra Dheeraj Peddinti, Stefano Pisoni, Egor Tiunov, Alessandro Marini, and Leandro Aolita, arXiv, June 2025 (10 pages).
 - [1398] [Quantum-Inspired Computational Fluid Dynamics for Transient Turbulent Compressible Flows](#), by Shang Xian Matthew Lee, Melissa Kozul, Muhammad Usman, Martin Sevier, Matthew L. Sims-Goh, and Richard D. Sandberg, arXiv, August 2026 (18 pages).
 - [1399] [Tailoring tensor network techniques to the quantics representation for highly inhomogeneous problems and few body problems](#), by Jheng-Wei Li, Nicolas Jolly, and Xavier Waintal, arXiv, April 2026 (17 pages).
 - [1400] [Quantum-Inspired Algorithms beyond Unitary Circuits: the Laplace Transform](#), by Noufal Jaseem, Sergi Ramos-Calderer, Gauthameshwar S., Dingzu Wang, Jose Ignacio Latorre, and Dario Poletti, arXiv, January 2026 (9 pages).
 - [1401] [A Quantum-Inspired Dequantization Method for Diagonally Weighted Matrix Functions: Application to Learning with Optimized Random Features](#), by Natsuto Isogai, Mio Murao, and Hayata Yamasaki, arXiv, September 2026 (18 pages).
 - [1402] [Classical algorithms for many-body quantum systems at finite energies](#), by Yilun Yang, J. Ignacio Cirac, and Mari Carmen Bañuls, arXiv, April 2022 (11 pages).
 - [1403] [Constant-depth preparation of matrix product states with adaptive quantum circuits](#), by Kevin C. Smith, Abid Khan, Bryan K. Clark, S.M. Girvin, and Tzu-Chieh Wei, arXiv, October 2024 (25 pages).
 - [1404] [Efficient Excited-State Calculations for Molecules Based on Contextual Subspace Method and Symmetry Optimizations](#), by Qianjun Yao and He Li, arXiv, June 2025 (8 pages).
 - [1405] [Reweighted Time-Evolving Block Decimation for Improved Quantum Dynamics Simulations](#), by Sayak Guha Roy and Kevin Slagle, arXiv, December 2024 (14 pages).
 - [1406] [Quantum-Inspired Solver for Simulating Material Deformations](#), by Mazen Ali, Aser Cortines, Siddhartha Morales, Samuel Mugel, Mireia Olave, Roman Orus, Samuel Palmer, and Hodei Usabiaga, arXiv, February 2025 (12 pages).
 - [1407] [Capturing long-range memory structures with tree-geometry process tensors](#), by Neil Dowling, Gregory A. L. White, et al., arXiv, October 2024 (8 pages).
 - [1408] [How a Classical Computer Beat a Quantum Computer at Its Own Game](#), by Mara Johnson-Groh, October 2024.
 - [1409] [Density-Matrix Mean-Field Theory](#), by Junyi Zhang and Zhengqian Cheng, arXiv, September 2024 (28 pages).
 - [1410] [Solving the Gross-Pitaevskii equation on multiple different scales using the quantics tensor train representation](#), by Marcel Niedermeier, Adrien Moulinas, Thibaud Louvet, Jose L. Lado, and Xavier Waintal, arXiv, July 2025 (13 pages).
 - [1411] [Fermionic tensor network contraction for arbitrary geometries](#), by Yang Gao, Garnet Kin-Lic Chan, et al., arXiv, December 2024 (9 pages).
 - [1412] [Matrix product states and first quantization](#), by Jheng-Wei Li and Xavier Waintal, arXiv, April 2024 (6 pages).
 - [1413] [Classical Algorithms for Constant Approximation of the Ground State Energy of Local Hamiltonians](#), by François Le Gall, arXiv, July 2025 (27 pages).
 - [1414] [Real-time Dynamics in 3D for up to 1000 Qubits with Neural Quantum States: Quenches and the Quantum Kibble-Zurek Mechanism](#), by Vighnesh Dattatraya Naik, Zheng-Hang Sun, and Markus Heyl, arXiv, April 2026 (14 pages).
 - [1415] [A multilevel tensor network compression technique for simulating Lindblad dynamics in superconducting circuits](#), by Adrien Moulinas and Xavier Waintal, arXiv, May 2026 (18 pages).
 - [1416] [Computational regimes in matrix-product-state-based quantum trajectory simulations](#), by Aaron Sander, Simon Cichy, Martin Eigel, Jens Eisert, Maximilian Frohlich, Tom Peham, and Robert Wille, arXiv, June 2026 (13 pages).
 - [1417] [Iterative Matrix Product State Simulation for Scalable Grover's Algorithm](#), by Mei Ian Sam, Tzu-Ling Kuo, and Tai-Yue Li, arXiv, January 2026 (5 pages).
 - [1418] [A Quantum-Inspired Algorithm for Graph Isomorphism](#), by Innes L. Maxwell, Stefan N. van den Hoven, and Jelmer J. Renema, arXiv, December 2025 (12 pages).
 - [1419] [Parallel iQCC Enables 200 Qubit Scale Quantum Chemistry on Accelerated Computing Platforms Surpassing Classical Benchmarks in Ruthenium Catalysts](#), by Seyyed Mehdi Hosseini Jenab, Brandon Henderson, and Scott N. Genin, arXiv, March 2026 (14 pages).
 - [1420] [Learning mixed quantum states in large-scale experiments](#), by Matteo Votto, Marko Ljubotina, Cécilia Lancien, J. Ignacio Cirac, Peter Zoller, Maksym Serbyn, Lorenzo Piroli, and Benoît Vermersch, arXiv, July 2025 (20 pages).
 - [1421] [Tensor network noise characterization for near-term quantum computers](#), by Stefano Mangini et al., Algorithmiq, August 2024 (21 pages).
 - [1422] [Learning interactions between Rydberg atoms](#), by Olivier Simard, Anna Dawid, Joseph Tindall, Michel Ferrero, and Antoine Georges, arXiv, August 2025 (20 pages).

- [1423] [Tensor networks based quantum optimization algorithm](#), by V.Akshay, Ar.Melnikov, A.Termanova, and M.R.Perelshtein, arXiv, April 2024 (15 pages).
- [1424] [Fast Numerical Solver of Ising Optimization Problems via Pruning and Domain Selection](#), by Langyu Li, Daoyi Dong, and Yu Pan, arXiv, September 2024 (8 pages).
- [1425] [Traveling Salesman Problem from a Tensor Networks Perspective](#), by Alejandro Mata Ali, Iñigo Perez Delgado, and Aitor Moreno Fdez. de Leceta, arXiv, July 2024 (13 pages).
- [1426] [Amplitude-Ensemble Quantum-Inspired Tabu Search Algorithm for Solving 0/1 Knapsack Problems](#), by Kuo-Chun Tseng, Wei-Chun Huang, et al., arXiv, March 2024 (7 pages).
- [1427] [Knapsack and Shortest Path Problems Generalizations From A Quantum-Inspired Tensor Network Perspective](#), by Sergio Muñiz Subiñas, Jorge Martínez Martín, Alejandro Mata Ali, Javier Sedano, and Ángel Miguel García-Vico, arXiv, June 2025 (14 pages).
- [1428] [Tensor Network Generator-Enhanced Optimization for Traveling Salesman Problem](#), by Ryo Sakai and Chen-Yu Liu, arXiv, February 2026 (11 pages).
- [1429] [Variational \(matrix\) product states for combinatorial optimization](#), by Guillermo Preisser, Conor Mc Keever, and Michael Lubasch, arXiv, December 2025 (8 pages).
- [1430] [Quantum-Inspired Optimization through Qudit-Based Imaginary Time Evolution](#), by Erik M. Asgrim and Ahsan Javed Awan, arXiv, December 2025 (11 pages).
- [1431] [A Quantum-Inspired Algorithm for Solving Sudoku Puzzles and the MaxCut Problem](#), by Max B. Zhao and Fei Li, arXiv, October 2025 (29 pages).
- [1432] [Scalable Ground-State Certification of Quantum Spin Systems via Structured Noncommutative Polynomial Optimization](#), by Jie Wang, David Jansen, Irene Frerot, Marc-Olivier Renou, Victor Magron, and Antonio Acin, arXiv, April 2026 (37 pages).
- [1433] [Renewable energy forecasting using optimized quantum temporal model based on Ninja optimization algorithm](#), by Mona Ahmed Yassen, Hossam El.Deen Salah Mostafa, et al., Scientific Reports, April 2025 (24 pages)  (preprint on doi.org).
- [1434] [Efficient Bitcoin Address Classification Using Quantum-Inspired Feature Selection](#), by Ming-Fong Sie, Yen-Jui Chang, Chien-Lung Lin, Ching-Ray Chang, and Shih-Wei Liao, arXiv, November 2024 (19 pages).
- [1435] [Oracle and Entanglement's seQure Launch Quantum-Inspired Threat Detection Platform on Oracle Cloud](#), by Cierra Choucair, The Quantum Insider, May 2025.
- [1436] [Application of Tensor Neural Networks to Pricing Bermudan Swaptions](#), by Raj G. Patel, Samuel Mugel, Roman Orus, et al., arXiv, March 2024 (16 pages).
- [1437] [Efficient molecular conformation generation with quantum-inspired algorithm](#), by Yunting Li, Man-Hong Yung, et al., arXiv, April 2024 (29 pages).
- [1438] [Enhancing GPU-acceleration in the Python-based Simulations of Chemistry Framework](#), by Xiaojie Wu, Weihao Gao, et al., arXiv, July 2024 (40 pages).
- [1439] [Quantum-inspired classical algorithm for molecular vibronic spectra](#), by Changhun Oh, Youngrong Lim, Yat Wong, Bill Fefferman, and Liang Jiang, arXiv, March 2023 (16 pages).
- [1440] [Hunting for quantum advantage in electronic structure calculations is a highly non-trivial task](#), by Ors Legeza, Andor Menczer, Miklos Antal Werner, Sotiris S. Xantheas, Frank Neese, Martin Ganahl, Cole Brower, Samuel Rodriguez Bernabeu, Jeff Hammond, and John Gunnels, arXiv, March 2026 (9 pages).
- [1441] [The Python Simulations of Chemistry Framework: 10 years of an open-source quantum chemistry project](#), by Qiming Sun and Garnet Kin-Lic Chan, arXiv, March 2026 (47 pages).
- [1442] [ByteQC: GPU-Accelerated Quantum Chemistry Package for Large-Scale Systems](#), by Zhen Guo, Ji Chen, and Dingshun Lv, arXiv, February 2025 (38 pages).
- [1443] [Stochastic tensor contraction for quantum chemistry](#), by Jiace Sun and Garnet Kin-Lic Chan, arXiv, February 2026 (42 pages).
- [1444] [Accurate, full-dimensional computations of thousands of complex vibrational eigenstates with tree tensor network states](#), by Henrik R. Larsson, Brieuc Le De, and Gino E. Gamboni, arXiv, May 2026 (13 pages).
- [1445] [Quantum-Inspired Ising Machines for Quantum Chemistry Calculations](#), by Mahmood Hasani, Hadis Salasi, and Negar Ashari Astani, arXiv, December 2025 (31 pages).
- [1446] [Tensor Networks for Explainable Machine Learning in Cybersecurity](#), by Borja Aizpurua, Samuel Palmer, and Roman Orus, arXiv, April 2025 (12 pages).
- [1447] [Quantum-inspired Techniques in Tensor Networks for Industrial Contexts](#), by Alejandro Mata Ali, Iñigo Perez Delgado, and Aitor Moreno Fdez. de Leceta, arXiv, April 2024 (13 pages).
- [1448] [Tensor network states to compress the many body problem](#), by Antoine Tilloy, November 2021 (15 slides).
- [1449] [Colloquium: Area laws for the entanglement entropy](#), by J. Eisert et al., 2010 (28 pages)  (preprint on arXiv).
- [1450] [Area Laws for Entanglement](#), by Fernando G.S.L. Brandão et al., 2014 (56 slides).
- [1451] [The resource theory of tensor networks](#), by Matthias Christandl, Freek Witteveen, et al., arXiv, December 2024 (66 pages).
- [1452] [Introduction to Tensor Network States and Methods](#), by Román Orús, 2020 (229 slides).
- [1453] [Survey on Computational Applications of Tensor Network Simulations](#), by Marcos Díez García and Antonio Márquez Romero, arXiv, August 2024 (38 pages).
- [1454] [Tensor Network Techniques for Quantum Computation](#), by Mario Collura, Guglielmo Lami, Nishan Ranabhat, and Alessandro Santini, arXiv, March 2025 (200 pages).
- [1455] [Tensor networks for quantum computing](#), by Aleksandr Berezutskii, Yuri Alexeev, et al., Nature, July 2025  (preprint on arXiv).
- [1456] [Variational preparation of normal matrix product states on quantum computers](#), by Ben Jaderberg, George Pennington, Kate V. Marshall, Lewis W. Anderson, Abhishek Agarwal, Lachlan P. Lindoy, Ivan Rungger, Stefano Mensa, and Jason Crain, arXiv, March 2025 (15 pages).
- [1457] [Quantum Advantage: a Tensor Network Perspective](#), by Augustine Kshetrimayum, Saeed S. Jahromi, Sukhbinder Singh, and Roman Orus, arXiv, March 2026 (45 pages).
- [1458] [Lecture 1: tensor network states](#), by Philippe Corboz, (56 slides).
- [1459] [Introduction to matrix-product states and tensor networks](#), by Grégoire Misguich, arXiv, June 2026 (64 pages).
- [1460] [Uniform matrix product states with a boundary](#), by Marta Florido-Llinas, Alvaro M. Alhambra, David Perez-Garcia, and J. Ignacio Cirac, arXiv, December 2025 (97 pages).
- [1461] [Density matrix renormalization group, 30 years on](#), by Frank Verstraete, Miles E. Stoudenmire, et al., Nature Reviews Physics, April 2023 .
- [1462] [Density-matrix renormalization group: a pedagogical introduction](#), by G. Catarina and Bruno Murta, arXiv, April 2023 (30 pages).
- [1463] [Clifford augmented density matrix renormalization group for *ab initio* quantum chemistry](#), by Lizhong Fu, Honghui Shang, Jinlong Yang, and Chu Guo, arXiv, June 2025 (6 pages).
- [1464] [Time evolution of the quantum Ising model in two dimensions using Tree Tensor Networks](#), by Wladislaw Krinitsin, Niklas Tausendpfund, Markus Heyl, Matteo Rizzi, and Markus Schmitt, arXiv, May 2025 (12 pages).

- [1465] [Simulating quantum dynamics in two-dimensional lattices with tensor network influence functional belief propagation](#), by Gunhee Park, Johnnie Gray, and Garnet Kin-Lic Chan, arXiv, April 2025 (16 pages).
- [1466] [Simulating and Sampling from Quantum Circuits with 2D Tensor Networks](#), by Manuel S. Rudolph and Joseph Tindall, arXiv, July 2025 (13 pages).
- [1467] [Tensor-product representations for string-net condensed states](#), by Zheng-Cheng Gu, Michael Levin, Brian Swingle, and Xiao-Gang Wen, arXiv, September 2008 (11 pages).
- [1468] [Probing Entanglement Scaling Across a Quantum Phase Transition on a Quantum Computer](#), by Qiang Miao, Marko Cetina, et al., arXiv, March 2025 (16 pages).
- [1469] [Approximate Contraction of Arbitrary Tensor Networks with a Flexible and Efficient Density Matrix Algorithm](#), by Linjian Ma, Matthew Fishman, Miles Stoudenmire, and Edgar Solomonik, arXiv, December 2024 (42 pages).
- [1470] [Basics of Tensor Network - An overview of tensors and renormalization](#), by Samuel Desrosiers, Glen B. Evenbly, and Thomas E. Baker, November 2018 (5 pages).
- [1471] [Barren plateaus in quantum tensor network optimization](#), by Enrique Cervero Martín, Michael Lubasch, et al., arXiv, April 2023 (26 pages).
- [1472] [Partitioned Expansions for Approximate Tensor Network Contractions](#), by Glen Evenbly, Johnnie Gray, and Garnet Kin-Lic Chan, arXiv, December 2025 (18 pages).
- [1473] [Replica Tensor Train](#), by Miha Srdinsek, Gabriel Gouraud, and Xavier Waintal, arXiv, April 2026 (18 pages).
- [1474] [Recursive Sketched Interpolation: Efficient Hadamard Products of Tensor Trains](#), by Zhaonan Meng, Yuehaw Khoo, Jiajia Li, and E. Miles Stoudenmire, arXiv, February 2026 (20 pages).
- [1475] [Algorithmic Locality via Provable Convergence in Quantum Tensor Networks](#), by Siddhant Midha, Yifan F. Zhang, Daniel Malz, Dmitry A. Abanin, and Sarang Gopalakrishnan, arXiv, April 2026 (5 pages).
- [1476] [Shadow tomography for classical tensor network simulations](#), by Jiace Sun and Garnet Kin-Lic Chan, arXiv, June 2026 (18 pages).
- [1477] [Learning Circuits with Infinite Tensor Networks](#), by Joe Gibbs and Lukasz Cincio, arXiv, June 2025 (13 pages).
- [1478] [Matrix Product State on a Quantum Computer](#), by Yong Liu, Guangyao Huang, Yizhi Wang, and Junjie Wu, arXiv, June 2025 (8 pages).
- [1479] [The landscape of software for tensor computations](#), by Christos Psarras and Paolo Bientinesi et al), arXiv, June 2022 (16 pages).
- [1480] [Performance Evaluation and Acceleration of the QTensor Quantum Circuit Simulator on GPUs](#), by Danylo Lykov, Yuri Alexeev, et al., arXiv, April 2022 (8 pages).
- [1481] [QArchSearch: A Scalable Quantum Architecture Search Package](#), by Ankit Kulshrestha, Danylo Lykov, Ilya Safro, and Yuri Alexeev, arXiv, October 2023 (10 pages).
- [1482] [Boosting the effective performance of massively parallel tensor network state algorithms on hybrid CPU-GPU based architectures via non-Abelian symmetries](#), by Andor Menczer and Örs Legeza, arXiv, September 2023 (16 pages).
- [1483] [Performance Benchmarking: Software for the Density Matrix Renormalization Group](#), by Per Sehlstedt, Paolo Bientinesi, and Lars Karlsson, arXiv, July 2026 (31 pages).
- [1484] [SeQuant Framework for Symbolic and Numerical Tensor Algebra. I. Core Capabilities](#), by Bimal Gaudel and Edward F. Valeev, arXiv, November 2025 (30 pages).
- [1485] [Tensor Networks in a Nutshell](#), by Jacob Biamonte and Ville Bergholm, arXiv, July 2017 (34 pages).
- [1486] [Lectures on Quantum Tensor Networks](#), by Jacob Biamonte, arXiv, January 2020 (178 pages).
- [1487] [Hand-waving and Interpretive Dance: An Introductory Course on Tensor Networks](#), by Jacob C. Bridgeman and Christopher T. Chubb, arXiv, May 2017 (62 pages).
- [1488] [Tensor Network Contractions](#), by Maciej Lewenstein, 2020 (160 pages).
- [1489] [Tensor Network Algorithms: a Route Map](#), by Mari Carmen Bañuls, arXiv, May 2022 (14 pages).
- [1490] [Applications of Tensor Networks: Machine Learning & Quantum Computing](#), by Edwin Miles Stoudenmire, 2018 (142 slides).
- [1491] [Towards Quantum Machine Learning with Tensor Networks](#), by William Huggins, Piyush Patel, K. Birgitta Whaley, and E. Miles Stoudenmire, arXiv, July 2018 (10 pages).
- [1492] [TensorCircuit: a Quantum Software Framework for the NISQ Era](#), by Shi-Xin Zhang, Shengyu Zhang, et al., arXiv, January 2023 (43 pages).
- [1493] [Tensor-Network Quantum Circuits](#), by Xanadu, June 2022.
- [1494] [Synergy Between Quantum Circuits and Tensor Networks: Short-cutting the Race to Practical Quantum Advantage](#), by Manuel S. Rudolph, Alejandro Perdomo-Ortiz, et al., arXiv, August 2023 (15 pages).
- [1495] [Discovering faster matrix multiplication algorithms with reinforcement learning](#), by Alhussein Fawzi et al., Nature, October 2022 (17 pages).
- [1496] [Tensor Quantum Programming](#), by A. Termanova, M. Perelshtein, et al., arXiv, March 2024 (17 pages).
- [1497] [A quantum eigenvalue solver based on tensor networks](#), by Oskar Leimkuhler and K. Birgitta Whaley, arXiv, May 2024 (24 pages).
- [1498] [Preparing matrix product states via fusion: constraints and extensions](#), by David T. Stephen and Oliver Hart, arXiv, May 2024 (15 pages).
- [1499] [Tensor Networks for Interpretable and Efficient Quantum-Inspired Machine Learning](#), by Shi-Ju Ran, November 2023 (12 pages).
- [1500] [Lower bounds for quantum-inspired classical algorithms via communication complexity](#), by Nikhil S. Mande and Changpeng Shao, arXiv, December 2024 (20 pages).
- [1501] [Pauli Propagation: A Computational Framework for Simulating Quantum Systems](#), by Manuel S. Rudolph, Tyson Jones, Yanting Teng, Armando Angrisani, and Zoë Holmes, arXiv, May 2025 (32 pages).
- [1502] [Leveraging Symmetry Merging in Pauli Propagation](#), by Yanting Teng, Su Yeon Chang, Manuel S. Rudolph, and Zoe Holmes, arXiv, December 2025 (17 pages).
- [1503] [Pauli Propagation for Imaginary Time Evolution](#), by Rafael Gomez-Lurbe and Armando Perez, arXiv, January 2026 (9 pages).
- [1504] [Efficient computation of real-time correlators using Pauli Propagation](#), by Alexander F. Kemper and Nicholas J. Mayhall, arXiv, July 2026 (8 pages).
- [1505] [Quantum simulation in the Heisenberg picture via vectorization](#), by Shao-Hen Chiew, Armando Angrisani, Zoe Holmes, and Giuseppe Carleo, arXiv, February 2026 (43 pages).
- [1506] [Scalable quantum circuit generation for iterative ground state approximation using Majorana Propagation](#), by Rahul Chakraborty, Aaron Miller, Anton Nykanen, Ozlem Salehi, Fabio Tarocco, Fabijan Pavosevic, Pi A. B. Haase, Martina Stella, and Adam Glos, arXiv, March 2026 (47 pages).
- [1507] [Transformer Wave Function for two dimensional frustrated magnets: emergence of a Spin-Liquid Phase in the Shastry-Sutherland Model](#), by Luciano Loris Viteritti, Federico Becca, et al., arXiv, April 2025 (14 pages).
- [1508] [Neural Network Quantum States for the Interacting Hofstadter Model with Higher Local Occupations and Long-Range Interactions](#), by Fabian Döschl, Annabelle Bohrdt, et al., arXiv, October 2024 (11 pages).

- [1509] [Empowering deep neural quantum states through efficient optimization](#), by Ao Chen and Markus Heyl, Nature Physics, July 2024 (15 pages).
- [1510] [Superconductivity in the t-t' Hubbard Model from Symmetry-Preserving Neural-Network Quantum States](#), by Riccardo Rende, Luciano Loris Viteritti, and Antoine Georges, arXiv, August 2026 (8 pages).
- [1511] [Basis dependence of Neural Quantum States for the Transverse Field Ising Model](#), by Ronald Santiago Cortes, Nils Niggemann, et al., arXiv, December 2025 (23 pages).
- [1512] [Scaling Laws for Neural-Network Quantum States](#), by Riccardo Rende, Alessandro Sinibaldi, Luciano Loris Viteritti, Roeland Wiersema, Antoine Georges, and Giuseppe Carleo, arXiv, June 2026 (7 pages).
- [1513] [Information-Theoretic Scaling Laws of Neural Quantum States](#), by Yiming Lu, Sriram Bharadwaj, Dikshant Rathore, and Di Luo, arXiv, March 2026 (27 pages).
- [1514] [Bound on Entanglement in Neural Quantum States](#), by Nisarga Paul, Physical Review Letters, vol 136, 120403, March 2026 📄 (preprint on [arXiv](#)).
- [1515] [Neural Quantum States in Non-Stabilizer Regimes: Benchmarks with Atomic Nuclei](#), by James W. T. Keeble, Alessandro Lovato, and Caroline E. P. Robin, arXiv, March 2026 (13 pages).
- [1516] [Neural-network quantum states for the nuclear many-body problem](#), by Alessandro Lovato, Giuseppe Carleo, Bryce Fore, Morten Hjorth-Jensen, Jane Kim, Arnau Rios, and Noemi Rocco, arXiv, February 2026 (85 pages).
- [1517] [Investigating a Quantum-Inspired Method for Quantum Dynamics](#), by Bo Xiao, Benedikt Kloss, and E. Miles Stoudenmire, arXiv, December 2025 (4 pages).
- [1518] [Entanglement Barriers from Computational Complexity: Matrix-Product-State Approach to Satisfiability](#), by Tim Pokart, Frank Pollmann, and Jan Carl Budich, arXiv, February 2026 (17 pages).
- [1519] [Fast Tensor Network Imaginary Time Evolution by Implicit Stepping on Logarithmic Grids](#), by John P. Zima, E. Miles Stoudenmire, Steven R. White, Olivier Parcollet, and Jason Kaye, arXiv, June 2026 (9 pages).

This **Quantum inspired algorithms** subsection contains 154 bibliographical references and notes containing at least 3,623 pages.

1.8 Quantum algorithms key takeaways

- Quantum algorithms have been created since the early 1990s, many years before any quantum computer was prototyped in a research laboratory. They manipulate quantum amplitudes through coherent unitary transformations, interference and measurement, and in many important algorithms, entanglement. Their mathematical description relies on vectors, matrices and linear operators acting in Hilbert spaces whose dimension grows exponentially with the number of qubits.
- The main algorithm classes are oracle based and search algorithms, optimization algorithms, quantum physics simulation algorithms, and quantum machine learning algorithms. There is also a more generic class of algorithms solving linear systems and partial differential equations.
- Quantum algorithms are interesting when they provide some quantum advantage compared to their equivalent best-in-class classical equivalents. It can be a theoretical and practical speedup, better result quality such as chemical simulation accuracy or better heuristics, less required training data in some machine learning cases, or, potentially, a lower computing cost including energy consumption. Quantum speedups must be practical and incorporate all classical and quantum computing costs, including error detection and mitigation (NISQ) or QEC/FTQC overheads. A polynomial quantum speedup can be a limiting factor, like with Grover's search algorithm, particularly if its oracle implementation is not well specified.
- Another key aspect of quantum algorithms is data loading and/or preparation. It is often overlooked and can have a significant time cost. For algorithms relying on amplitude encoding, loading an arbitrary N -dimensional classical vector generically requires $O(N)$ resources in one form or another unless exploitable structure or an efficient state-preparation mechanism is available. Some algorithms assume the use of qRAM, which is not yet available at useful scale. These constraints make quantum computing not well adapted to big-data workloads such as conventional large-language model (LLM) training and inference. Any serious resource estimate should include the circuit and data-access costs required to prepare the quantum state that is then processed by components such as phase estimation, amplitude amplification or quantum simulation.
- Relatively few quantum algorithms provide a theoretical computing time exponential or superpolynomial speedup, the main one being Shor's integer factoring and discrete log algorithms. Any exponential speedup for most other classes of algorithms like HHL to solve linear systems is usually conditioned by specific criteria like data sparsity and the existence of some structure in the problem. The 2024 DQI (Decoded Quantum Interferometry) from Google generated a lot of hope to solve optimization problems efficiently, but the related resource estimates are quite significant.
- Gate-based computers, quantum annealers and analog quantum-simulator algorithms are all hybrid, combining a classical (preparation) part and a quantum part (a quantum circuit, or an annealing process). A special breed of quantum algorithms are the variational quantum algorithms (VQA) and their variants for optimizations (QAOA), chemical simulations (VQE) and machine learning (QML) that target NISQ systems (noisy intermediate quantum computers). These combine classical preparation and adjustment of a Hamiltonian that computes a cost function with several classical optimization steps and quantum computing circuit executions. Algorithms based on a Quantum Fourier Transform (QFT) and modular exponentiation require a fault-tolerant quantum computer with quantum error correction. But they are also hybrid since they need some classical software that prepares the circuit to run, a task that may be highly resource demanding and in some cases, that requires an HPC.
- Quantum-inspired algorithms run on classical computers and borrow mathematical representations, sampling techniques or algorithmic ideas from quantum information and many-body physics. Tensor networks, including MPS and DMRG, form one major family, but other approaches include sample-and-query dequantization, neural quantum states, Pauli and Majorana propagation, matchgate and fermionic-Gaussian simulation, and annealing-inspired optimization. They can improve classical computing solutions and are useful for classical simulations of quantum systems. Progress in these methods continuously raises the classical baseline against which quantum advantage must be evaluated, including in quantum chemistry where increasingly large active spaces can now be treated classically.

2 Quantum software development tools

We now need to explore quantum computing software to implement the various algorithms we've just uncovered. It is a completely new world with very different paradigms. Quantum algorithms still require programming, programming languages and development environments. Quantum software engineering is an emerging domain with tons of tools to create, invent or reinvent for code writing, debugging, testing, verification, certification, compilation and optimization. It also involves resource estimation and benchmarking tools, tools for securing quantum computing and cloud and infrastructure operation tools.

2.1 Development tool classes

Gate-based quantum software is organized in layers with, starting from the bottom, the physical qubits followed by low-level machine language to drive them at the physical level (microwaves length and frequencies, readouts, etc.) (Figure 90).

Next comes high level quantum source code, which is in fact a kind of macro-assembler, able to take advantage of function libraries with ready-to-use algorithms (quantum Fourier transforms, phase estimates, etc.) and finally, high-level languages or libraries tailored to specific business needs.

In the lower layers between machine language and macro-assembler are functions for converting quantum gates into a set of universal quantum gates supported by quantum hardware as well as error correction code systems that may require the execution of a large number of quantum gates. A quantum compiler also implements many optimizations by for example removing quantum gate sequences that do not change the state of a qubit, such as two consecutive Hadamard or X (NOT) gates. It also arranges these quantum gates to minimize the number of quantum gate steps in the solution. Quantum software architecture is generally hybrid. They manage side by side the execution of classical and quantum code.

At the last stage, the classical computer is used to control the execution of quantum algorithms, if only to trigger the quantum gates at the right time, sequentially.

Analog computing (quantum annealing, analog quantum simulation) software stacks are a bit different with the various tools used to convert the problem into some Ising or QUBO model and its graph embedding onto the hardware qubit layout that is more or less flexible (Figure 91).

We can identify some major classes of tools for creating quantum software: graphic programming tools, scripting languages, intermediate languages, machine languages, compilers, and application libraries [1520].

2.1.1 Graphical programming tools

They allow us to visually define the sequence of quantum gates to create algorithms and execute them on quantum accelerators.

These tools can also emulate, run and visualize the status of qubits when their number is reasonable with various methods: Bloch sphere, register state and density matrix.

One example of such tools is the [IBM Quantum Experience](#) Composer that is available online since 2016 (Figure 92). Code can be executed on one of the many IBM quantum systems available in the cloud or on the Qiskit Aer emulator you can install on your own hardware.

There are also graphical simulators of qubits that can be used to understand how to chain quantum gates on a few qubits and visualize the result visually. The most open one is the open-source tool [Quirk](#) which can simulate up to 16 qubits.

It works online and can be downloaded to run on your own computer locally. Figure 93 shows an [example](#) of a quantum Fourier transform performed in Quirk. It was developed by **Craig Gidney**, now a Google engineer specialized in algorithms and error correction codes.

Quantum circuits graphical representation can quickly become complicated with over 30 qubits and a large number of gate cycles. It can be thus interesting to visualize code in a simplified manner with a higher level of abstraction with gate block functions. Such a tool named Quantivine has been proposed by a team of researchers in 2023 [1521] (Figure 94).

Presented in 2025, **MuqcsCraft** from Ecole de technologie supérieure in Montreal, Canada, is a graphical tool akin to Quirk, which adds a representation of the register vector state and its evolution over time for small circuits. It is very useful for teaching and learning quantum circuit design [1522].

ZX-calculus is a graphical programming language that uses topological composition rules. It was created in 2008 by Bob Coecke and Ross Duncan [1523]. It visualizes the modifications made to a set of qubits and is based on transformations applicable to the geometric representation of quantum gates that simplify models (Figure 95).

It is particularly useful for programming a quantum computer in MBQC (measurement base quantum computing), for visually model error correction codes [1524] and design braiding models in the topological quantum

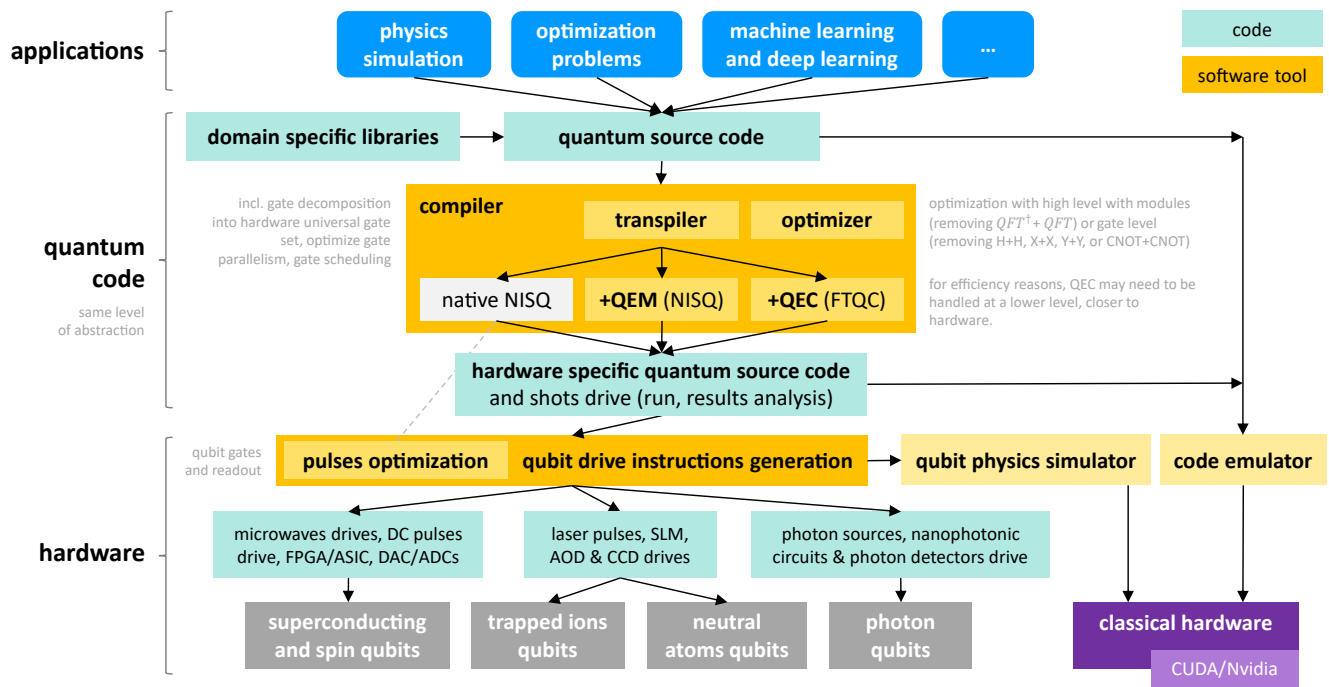


Figure 90: Classification of quantum software engineering tools in the code and compilation+emulation foodchain. © Olivier Ezratty, 2023-2025.

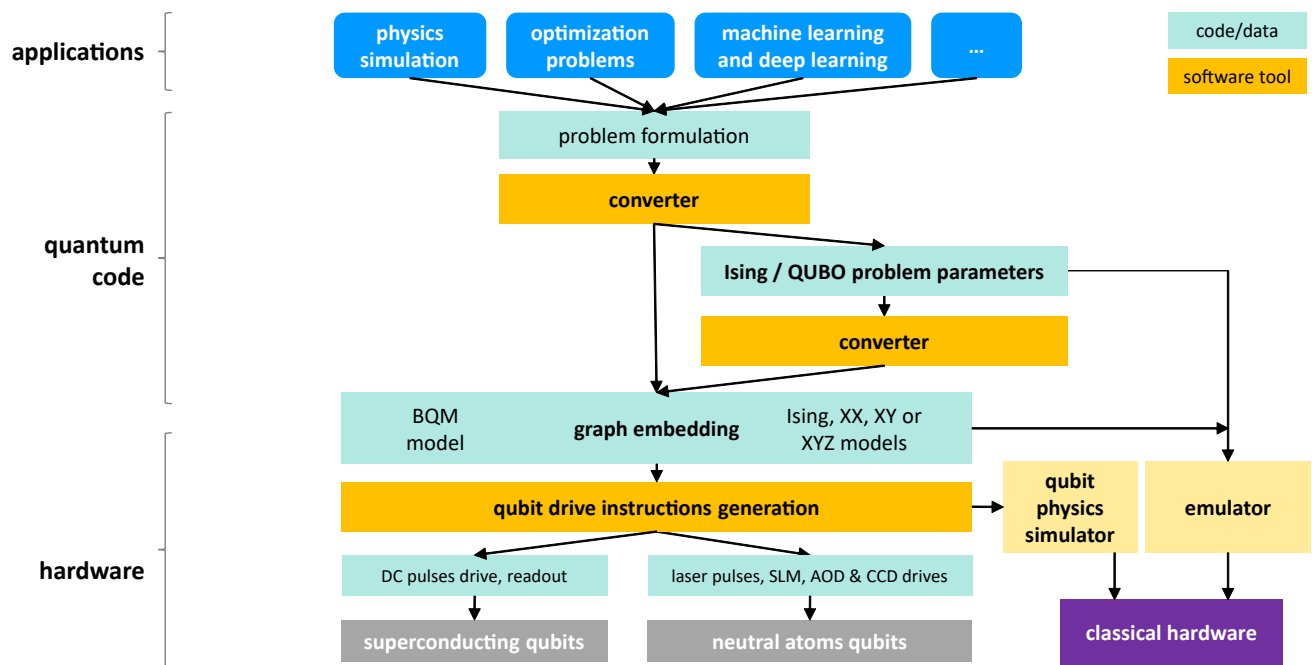


Figure 91: Software tools for analog computing. © Olivier Ezratty, 2023.

computing paradigm [1525]. It can also help optimize quantum code compilation [1526–1528].

Contributors to the ZX-Calculus work include researchers from Loria under the responsibility of Simon Perdrix, a research laboratory located in Nancy and Dominic Horsman then at UGA LIG in Grenoble and now at Oxford University [1529, 1530].

ZX has its own zoo of extensions:

- **PyZX** is a Python tool created in 2019 that implements ZX-Calculus principles for the creation, visualization, and automated rewriting of large-scale quantum circuits.
- **SZC-calculus** (Scalable ZX-calculus) is a high-level extension of ZX-calculus for the design and verification of quantum computations with qubits registers. Among other things, it can be used to describe graph states used in MBQC and error correcting codes [1531].

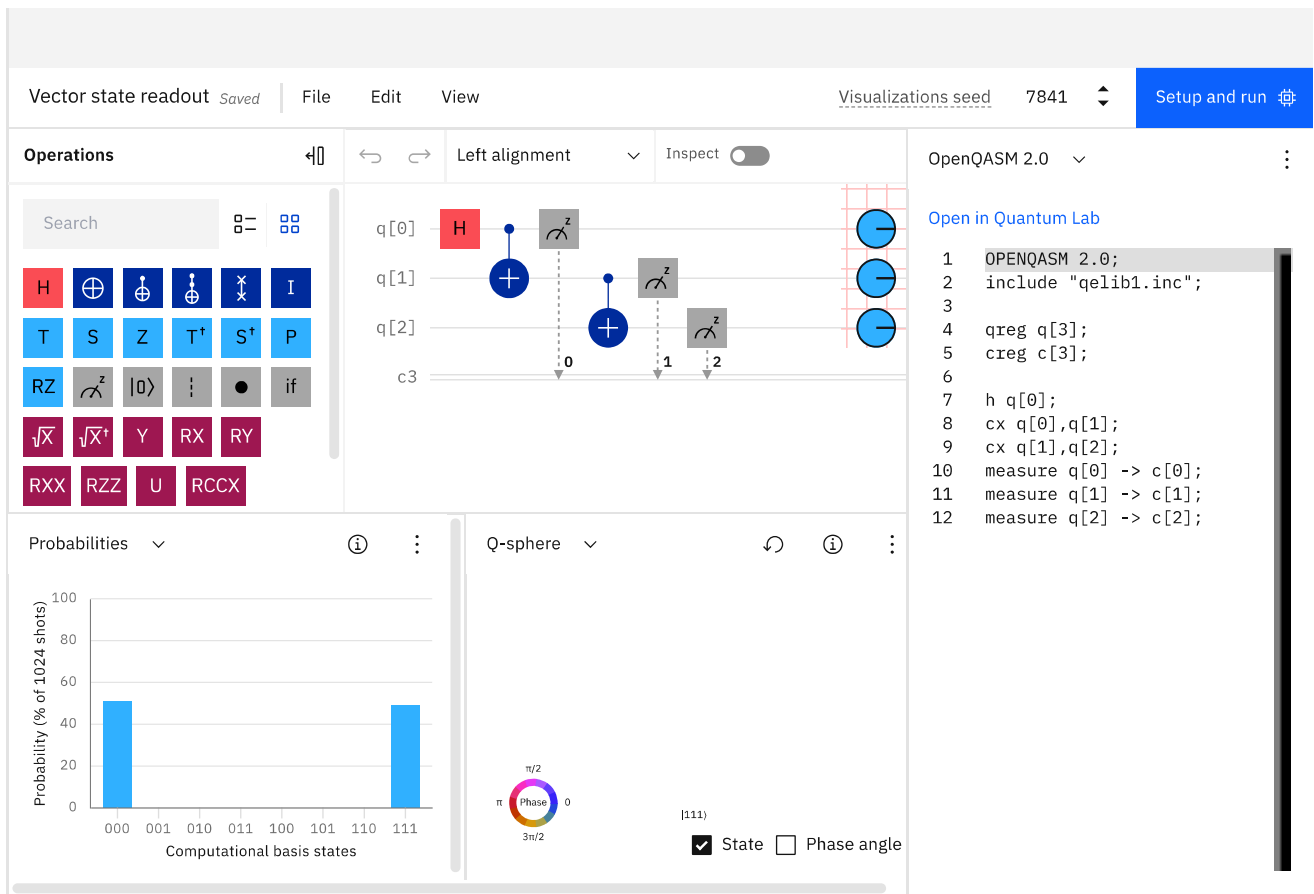


Figure 92: IBM Quantum Experience visual interface. Source: IBM.



Figure 93: Quirk, an open-source visual quantum-programming tool that runs in a browser and supports up to 16 qubits in vector-state mode. It runs on a laptop and even a smartphone. It was developed by Craig Gidney before he joined Google. Source: Quirk Algassert.

- **ZXH-calculus** is a graphical language based on ZX-calculus that helps model many-body states [1532].
- **ZW-calculus** is a variant that allows better entanglement modeling and **ZH-calculus** is used for the generalization of MBQC programming models thanks to the addition of a box implementing the Hadamard gate and an easier integration of Toffoli gates in

its diagrams. It is associated with a development tool, **Quantomatic**, created by Aleks Kissinger and Vladimir Zamdzhiev, then at the University of Oxford. Zamdzhiev is now at Inria in France [1533].

- They organize their own international conference, the **QPL** (Quantum Physics and Logic) but it covers broader topics than ZX calculus. [1531]

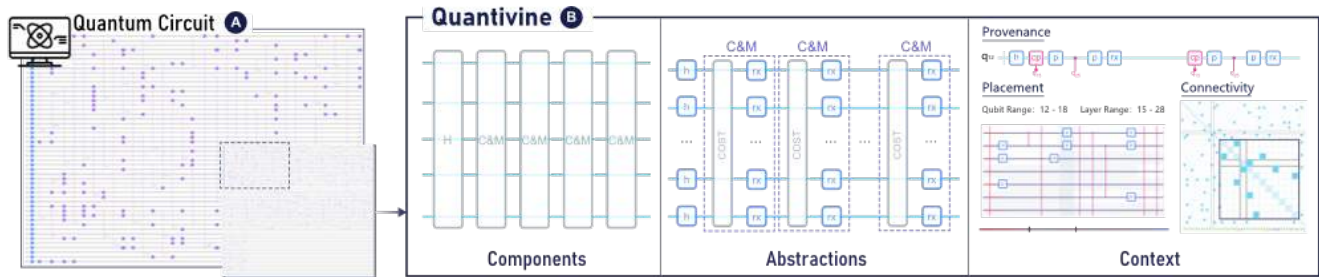


Figure 94: Quantivine is a graphical tool to simplify a large circuit representation with circuit blocks, here with an example using 50 qubits. Source: [1521].

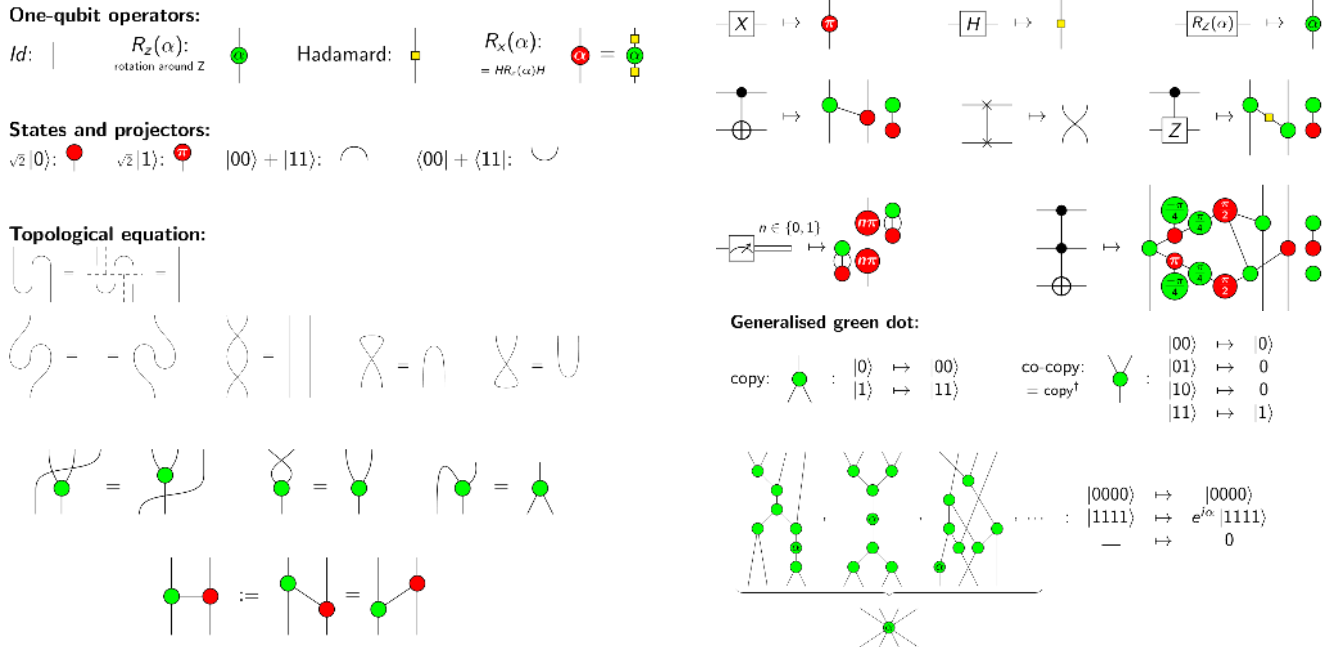


Figure 95: ZX calculus graphical language key operations. Source: [1529].

- **DisCoPy** (Distributional Compositional Python) is an open-source toolbox for computing with string diagrams and functors [1534]. Its diagram data structure allows us to encode various kinds of quantum processes, with functors for classical simulation and optimization, as well as compilation and evaluation on quantum hardware. It supports ZX calculus and its variants and linear optical quantum computing. Its photonics module, still under development as of May 2022, builds linear optical circuits and connects with the quantum simulator **Perceval** from Quandel. DisCoPy encodes arbitrary string diagrams and interprets these for computation on various classical (NumPy, JAX, TensorFlow, PyTorch, Sympy) and quantum systems (Perceval, PyZX). DisCoPy is of particular use with quantum natural language processing (QNLP) and quantum machine translation applications [1535].

2.1.2 Scripting languages

They are used to program quantum algorithms in text mode. These tools allow us to associate classical pro-

gramming with the chaining of quantum functions conditioned by the state of variables in classical memory or coming from mid-circuit measurement. Such code can contain repetitive sequences that would be cumbersome to input in a graphical representation of the algorithm (Figure 97).

Two programming styles are commonly encountered in quantum software: imperative and functional programming (Figure 96):

- **Imperative languages** are procedural programming languages where step-by-step algorithms are described. They include the usual languages such as C, C++, PHP or Java, and also C# from Microsoft.
- **Functional programming** emphasizes functions, immutable data and expression evaluation. Pure functional languages avoid mutable state and typically use recursion or higher-order functions instead of imperative loops, while multi-paradigm languages can mix functional and imperative constructs. This style can make some transformations and compiler analyses more explicit.

Many traditional programming languages can be used for imperative or functional programming, especially if they use function pointers or support event-driven logic. JavaScript supports functional-programming patterns, including higher-order functions and callbacks. jQuery is a JavaScript library, not a separate programming language. The same mix of styles is possible in C++.

Table 1: A selection of some quantum programming languages.

Name	Style	Notes
QCL	Imperative	Has classical sublanguage, multiple high-level programming features.
qGCL	Imperative	Emphasis on algorithm derivation and verification.
LanQ	Imperative	Full operational semantics, proven type soundness.
Quipper	Functional	Focus on scalability, plans to include linear types for static checks (currently done at run-time).
QPL	Functional	Statically typed, denotational semantics in terms of CPOs of superoperators.
QML	Functional	Linearly typed, focused on weakening - not contraction. Quantum control and quantum data.
Qumin	Functional	Two sublanguages (untyped and linearly typed). Focus on ease of use and clean, functional style of programming.

Figure 96: Imperative and functional quantum programming languages. Source: [1536].

The most common programming language used with quantum libraries is Python. It provides language constructs, data types, for-loops and branching, modularity, and classes. It can help build repetitive code structures, which would be harder than with graphical circuit design. It can also be used to create automatic testing tools.

2.1.3 Frameworks

Along these classical programming languages, developers use quantum software development kits and frameworks. Higher-level SDKs such as Qiskit or pyQuil generate circuits and hybrid workflows, while languages and intermediate representations such as OpenQASM, Quil or QIR provide lower-level exchange and compilation formats. A compiler or transpiler lowers the higher-level representation toward the target backend. As shown in Table 9, the source-code size varies by roughly a factor of three across these frameworks for the small workloads considered.

Beyond gate-based quantum computing, there are also some domain-specific languages devised to drive quantum simulators [1537].

2.1.4 Machine languages

At the lowest software levels are assembly-like instruction sets, controller microcode and pulse-level descriptions. Assembly or intermediate representations specify gates, measurements and control flow, while lower control layers translate these instructions into microwave, voltage or optical pulses and detector operations. These layers are hardware dependent and are normally hidden from quantum algorithm developers.

Language	Shor	TFIM		Heisenberg	
		Trotter	LCU	Trotter	LCU
Cirq	127	45	203	53	203
CUDA-Q	114	38	196	40	191
Guppy	144	45	215	62	216
PennyLane	48	36	180	37	181
PyQuil	182	50	216	58	217
Q#	173	24	316	42	318
Qiskit	136	33	195	35	196
Qualtran	111	35	181	64	183
Qrisp	50	61	149	63	150
Silq	87	15	137	25	140

Table 9: Ten frameworks and the corresponding number of lines of source code required to solve standard problems (Shor factoring, transverse-field Ising model or TFIM, and a Heisenberg model). Source: [1538].

2.1.5 Compilers, transpilers and optimizers

Quantum compilers transform a quantum program through several abstraction levels toward a target backend (Figure 99). Depending on the software stack, compilation may stop at a target instruction set, while lower control software generates pulses and readout commands. In FTQC stacks, compilation can also include logical-gate synthesis, quantum error correction constraints and magic-state resources.

Circuit synthesis is a technique used by compilers or other tools to generate quantum circuits implementing a known unitary U , a Boolean oracle or a target gate [1539]. It decomposes the target operation into a chosen gate set that can then be compiled [1540]. Methods include Cartan decomposition and related Lie-group techniques [1541, 1542]. The synthesis must be optimized because many different circuits can implement the same unitary transformation. Two-qubit gate count is a common cost metric [1543, 1544]. Early exact synthesis methods focused on two-qubit operations [1545, 1546]. General synthesis extends to n qubits, but an arbitrary n -qubit unitary has on the order of 4^n real parameters and generic exact synthesis therefore requires an exponential number of elementary gates, not a polynomial number [1547]. Polynomial-size synthesis is possible for structured families of unitaries that admit compact descriptions [1548, 1549]. Reinforcement learning is also explored [1550–1553], including to reduce T-count in FTQC settings [1554–1558]. Transformer models are being explored as well [1559], with correctness requiring explicit verification because generated circuits can be invalid or suboptimal [1560, 1561]. In August 2026, Mothe and Gühne introduced a framework that can certify that a target unitary cannot be implemented with a given circuit architecture and gate set [1562].

Q Can I run my classical code faster with a quantum computers?

Well, practically, no. Theoretically, yes.

Converting classical code to run it on a quantum computer will not generally make it faster. It will usually run much slower due to slow quantum hardware clock speeds. Quantum gates take much more time than any operation running on a standard microprocessors. Then, you must add the error correction overhead. Finally, quantum algorithms speedups are conditioned by many characteristics: the problem structure, the ability to solve it with using interferences in a quantum register, and the use of “magic” (non trivial Clifford gates).

While code porting is out of question, creating a quantum algorithm solving a problem currently solve with classical may make sense. In that case, the solution is entirely different even though it usually contains a classical part, the one that is used to generate the quantum circuit to run on a quantum computer.

At the gate level, circuit synthesis can be applied to:

- CNOT gates, because many QPUs do not implement CNOT as their native entangling gate and instead use CZ, ECR, iSWAP, Mølmer-Sørensen or other native interactions. A CNOT can be synthesized from the native gate set. Limited connectivity can additionally require routing operations such as SWAPs, with a standard SWAP decomposition using three CNOTs [1563]. Some architectures can reduce routing overhead through nonlocal or teleportation-based gates [1564].
- Toffoli gates and more generally n-control-NOT gates ($C^n(X)$) using a minimum number of CNOT gates [1565–1568] (Figure 102) and T gates [1569, 1570].
- Arbitrary single-qubit rotations can be parameterized, up to a global phase, by an Euler decomposition such as $U(\alpha, \beta, \gamma) = R_z(\alpha)R_y(\beta)R_z(\gamma)$. In FTQC, these gates are approximated using a FTQC gate set, usually with Clifford and T gates or another combination. The Solovay-Kitaev theorem provides a general polylogarithmic approximation solution for this decomposition, but practical synthesis now often uses more efficient number-theoretic, probabilistic or optimization-based methods. There are dedicated techniques for small-angle rotations [1571] and for “pre-synthesis”, which rearranges a circuit to merge compatible rotations before their fault-tolerant synthesis [1572, 1573].
- Pauli exponentials are unitaries of the form $e^{-i\theta P/2}$, where P is a Pauli string, namely a tensor prod-

uct of single-qubit Pauli operators I , X , Y and Z acting on n qubits [1574, 1575]. For example, $P = X_1 \otimes Z_2 \otimes Z_3 \otimes Y_4$ acts on four qubits. Since $P^2 = I$, $e^{-i\theta P/2} = \cos(\theta/2)I - i \sin(\theta/2)P$. A multi-qubit Pauli exponential can create entanglement, depending on the input state, the Pauli weight and the rotation angle. Hamiltonians are commonly decomposed as sums of Pauli strings, and product-formula or related simulation methods implement the corresponding time evolution as products of Pauli exponentials. They are also common in NISQ variational ansatzes.

- Matchgates which are two-qubit gates commonly used in quantum chemistry for simulating non-interacting fermions and spin chains [1576].
- Diagonal unitaries, which are used for example as phase separators in QAOA circuits [1577].
- Stabilizer circuits used for syndrome extraction and other operations in quantum error correction [1578].

Transpilation. A transpiler maps a circuit or intermediate representation to the gate set, connectivity and timing constraints of a target backend, while lower control layers may subsequently translate scheduled operations into pulses (Figure 98). Transpilation can become computationally expensive for large circuits and therefore needs its own heuristics and optimizations [1579, 1580]. In NISQ systems it can use calibration data, native error rates, connectivity, gate durations and crosstalk information [1581], with additional pulse-level optimization where supported [1582]. Transpilation remains relevant in FTQC, but the target changes from physical native gates to logical instructions, code-deformation or lattice-surgery primitives, magic-state operations, routing and fault-tolerant scheduling.

Optimization. Compilers carry out optimizations specific to certain types of algorithms [1586–1590] and hardware-specific architectures like trapped ions and cold atoms QPUs [1591–1596]. Many advances can reduce the number of gates used and create shallower circuits. It can be done with qubit mapping which optimizes SWAP insertion techniques that are used heavily when qubit connectivity is poor, like with converting SWAP gates into 3 CNOTs and removing redundant combinations of CNOTs [1597–1599], replacing CNOTs by $\sqrt{i\text{SWAP}}$ gates [1600], optimizing CNOT circuits generation [1601–1607], transpiling that takes qubit connectivity into account [1608–1610], and optimizing the balance between CNOTs and T gates, particularly for fault-tolerant schemes that are very costly with T gates like Hamiltonian simulations [1611]. This is for example done by AlphaTensor which was codeveloped by Google DeepMind and Quantinuum [1612, 1613]. It is about minimizing the T-count, the number of T gates used in the algorithm or its constituent parts like with quantum Fourier transforms [1614–1617] and freeing ancillas for reuse through automated uncomputation [1618–1620].

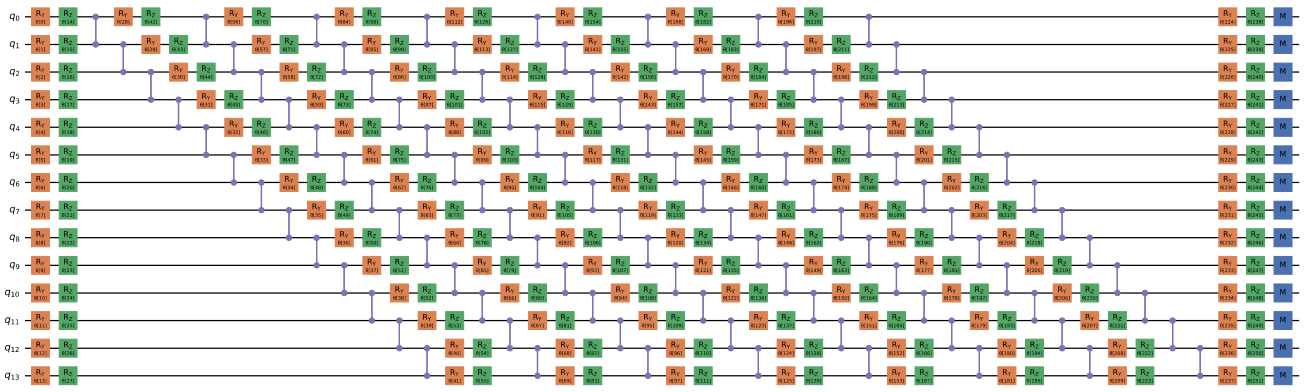


Figure 97: Example of repetitive pattern in a quantum circuit. Source: [1583].

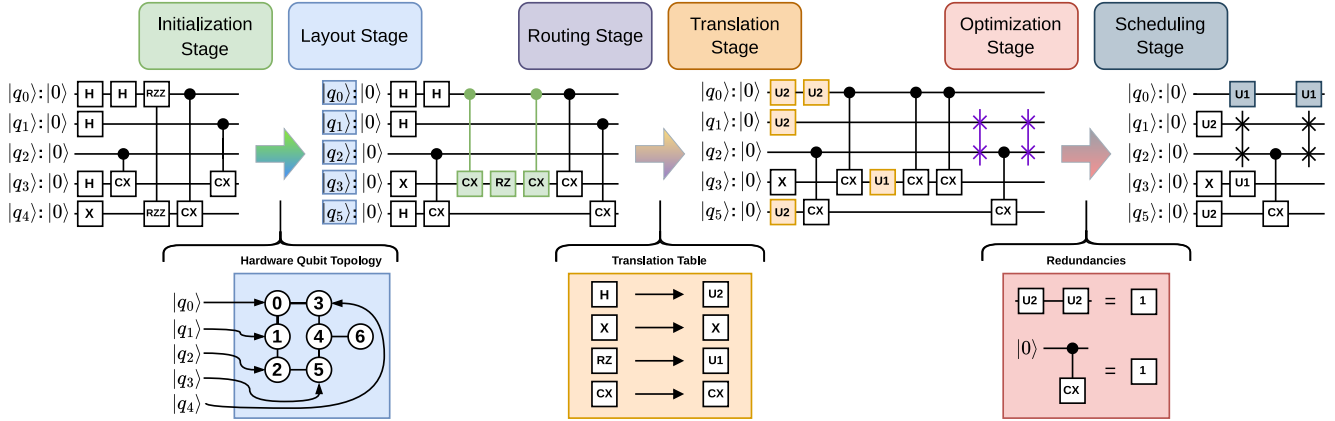


Figure 98: The steps of transpilation in Qiskit, with the translation of the circuit to native gates happening in the middle of the process at the translation stage. Source: [1584].

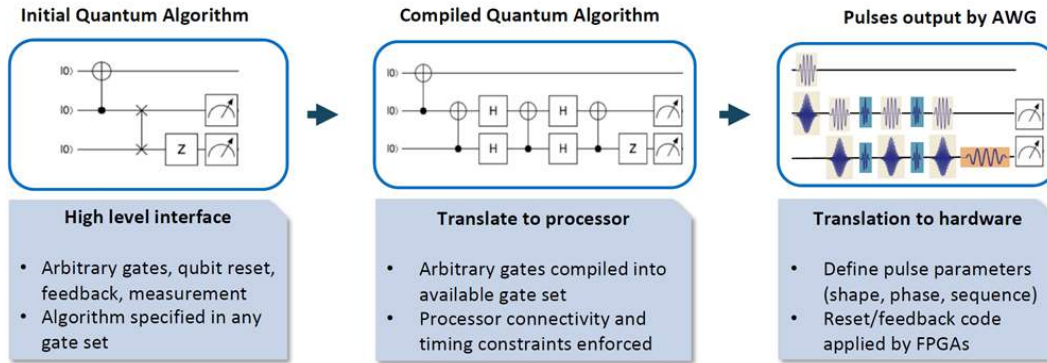


Figure 99: The various roles of a quantum code compiler, first to translate high-level gate codes into primitive gates supported by the quantum processor, and then to turn these gates into low-level electronic controls driving qubit gates and readout. Source: [1585].

There are also various optimization strategies to find the optimized block encoding technique [67], distribute computing across multiple quantum computers [1621], compiling techniques optimizations using precompilation and predicting schemes [1622], precomputation methods to optimize circuit-execution timing and making use of the gate teleportation mechanism [1623], to maximize the overall circuit fidelity [1624], for crosstalk and decoherence mitigation in superconducting frequency-tunable quantum chips [1625], to integrate quantum error detection mechanisms [1626], for the compilation

of a VQE algorithm for NISQ QPUs [1627], for QAOA [1628], and to minimize the effect of noise (in a NISQ regime) [1629] and for the QFT [1630]. These optimizations minimize the number of expensive gates (two-qubit gates, or arbitrary angle single-qubit rotations, T) while also minimizing the approximation error of the compiled circuit to the ideal target unitary [1631].

Various machine and deep learning techniques, including reinforcement learning, are also used in optimization [1612, 1632–1636]. Reinforcement learning can also

manage quantum error correction circuit compression [1637].

Noise-adaptive mapping techniques can take into account qubit calibration data and avoid bad two-qubit gate combinations although in the long term, it can have counter-intuitive detrimental effects that are well documented in [1638].

Quantum compilers and optimizers could even potentially run themselves on a quantum computer although it would demand more quantum computing resources than the target algorithm [1639].

Retargetability describes a compiler’s ability to map the same source program or intermediate representation to multiple quantum processor backends. A 2026 benchmark reports strong retargetability for Quantinuum’s TKET and IBM’s Qiskit [1640].

FTQC compilation is different from NISQ compilation because the compiler targets logical operations implemented through a quantum error correction architecture ([1641] explains well this process). Given a physical error model and a target algorithmic failure probability generally linked to the circuit size, resource estimation and compilation help determine an adequate code distance and the required data and ancilla qubits. Depending on the non-Clifford content after synthesis, the stack must provision magic-state factories and schedule state injection or gate teleportation [1642]. Depending on the code and hardware connectivity, it may also schedule lattice surgery, patch routing, transversal logical operations and physical qubit transport. These resources must be coordinated with accounting for the error syndrome decoder latency and classical feed-forward [1643] (Figure 100). Modular systems additionally require circuit partitioning and inter-processor communication [1644]. The resulting problem is a space-time-resource scheduling problem, potentially including energy constraints. This field remains relatively immature, and the classical cost of compiling and optimizing very large FTQC workloads is still poorly quantified.

Circuit scheduling. Compilers determine operation start times under gate-duration, connectivity, measurement, reset, feed-forward and hardware-resource constraints. In NISQ circuits, the critical path and idle periods should remain compatible with the relevant T_1 and T_2 times. In FTQC, the overall computation is not limited by the physical-qubit coherence time since error correction runs continuously to extend the lifetime of logical qubits for the circuit execution duration. Scheduling is instead constrained by QEC cycle time, decoder and feed-forward latency, logical operation time, routing and the availability of resources such as magic states.

Low-level compiling. At the lowest control level, scheduled operations are translated into hardware commands such as microwave and voltage pulses, laser pulses, SLM or AOD patterns, detector triggers and readout acquisition sequences.

All these programming tools come from either research labs or from quantum computer vendors such as IBM, Rigetti and D-Wave [1649]. Like Bull/Eviden’s AQASM, these compilation tools can be cross-platform and support different gate-based quantum computer architectures. Quantum programming languages are generally able to combine classical procedural programming with quantum registers and gates programming. They allow parallel management of classical memory with quantum registers.

Compiling cost. As quantum circuits size grows, their compiling, transpiling and optimization costs may become a concern. Academics are starting to investigate this topic to uncover potential bottlenecks, particularly in the FTQC regime. Balancing optimization classical costs and quantum circuit execution time savings will become a common trade-off. As detailed in [1650], for 100-qubit circuits, it seems that when optimizing circuit compilation, the classical compilation computing time may exceed quantum circuit execution (Figure 101).

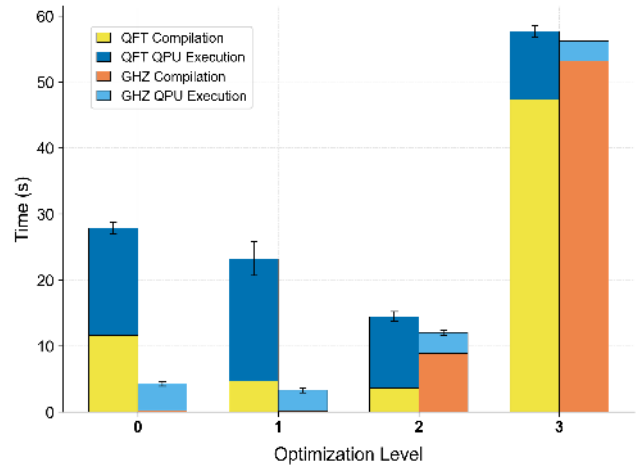


Figure 101: A comparison of the time cost of compiling and quantum circuit execution for a QFT and the preparation of a GHZ state with increasing optimization from 0 to 3. It shows that with maximum optimization, the compiling time cost becomes much larger than the circuit execution time. The caveat here is that the estimates were based on running a QFT and GHZ state preparation circuits only 30 times and the compilation was achieved using an HPC node with a single 48-core CPU. In real life, you could use a more powerful classical configuration on one end, and run the circuit two orders of magnitude more often. Source: [1650].

Quantum compilation can be either a fixed or an input-dependent cost. A parameterized circuit can often be compiled once and reused for many parameter values, while state preparation, oracles, block encodings or data-dependent circuit generation can force recompilation or partial re-optimization. This is not unique to quantum computing, since classical JIT compilation and specialization can also depend on runtime data. Shor order finding has a modular-arithmetic subcircuit, while generic QPE applies controlled powers U^{2^k} of an arbitrary unitary. HHL uses phase estimation with Hamiltonian simulation [158]. Compilation cost should

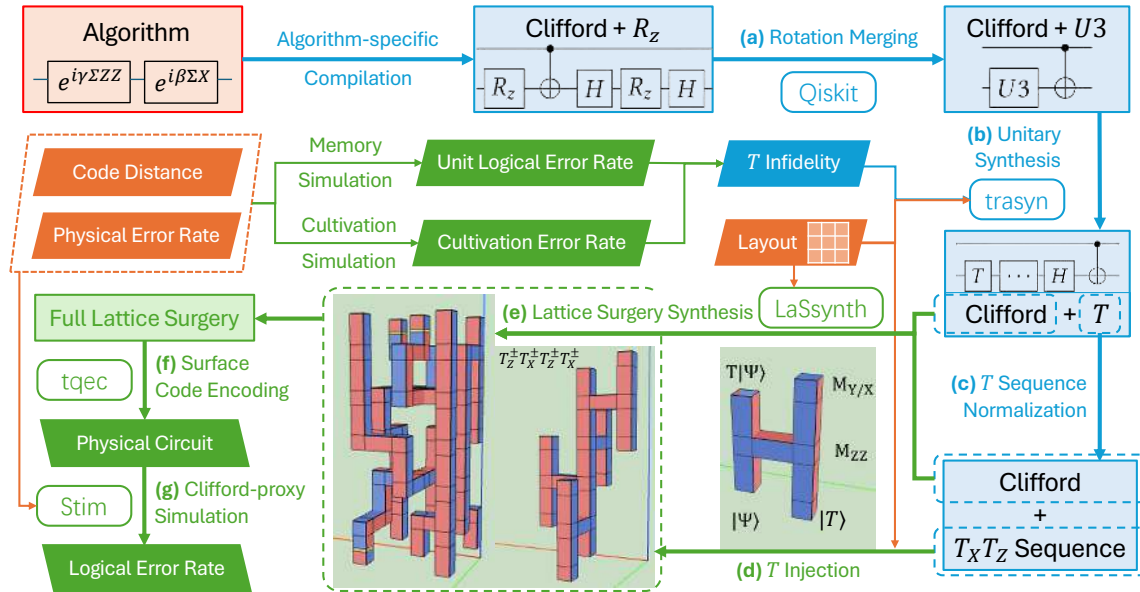


Figure 100: Example of FTQC compilation pipeline and parameters. Source: [1645]. [1646–1648] also provide insights on the topic and architecture of QEC/FTQC compilation.

therefore be accounted for when it depends materially on the workload, but it may be amortized over repeated circuit executions. We still lack robust classical resource estimates for compiling and optimizing very large FTQC workloads, particularly when QEC transformations are integrated into the compilation stack [1651, 1652].

Code obfuscation. Like in classical computing, it will become useful to make sure that a compiled quantum circuit can’t be reverse engineered to avoid IP theft and other cyberattacks. This is becoming an investigated research area [1653]. But the other way around is done, with reverse-engineering compiled circuits for the sake of explainability or interpretability, using some machine-learning-based decompiling techniques [1654].

2.1.6 Application-specific frameworks

On top of scripting languages used to program quantum computers, annealers or simulators sit a wealth of application-specific frameworks aimed at solving particular problems. The bulk of these frameworks are proposed by the many software vendors we inventory in a special part of this book. They usually support various underlying hardware architectures from classical digital emulators to QPUs running on-premises or in the cloud, the best ones being able to support heterogeneous classical and quantum platforms for the test and implementation of hybrid algorithms.

The most common proposed frameworks cover generic optimization problems (mostly using variational techniques mixing classical and quantum computing like QAOA and VQE/VQA), and others for chemical simulations and financial optimizations, which have been identified as the first go-to-markets for quantum computing so far. Some application-specific frameworks are

also proposed by the large vertically integrated quantum computing vendors like IBM (with their Qiskit frameworks). Others can also be proposed by research labs like Q² Chemistry which comes from USTC in China [1655].

2.1.7 Emulators and simulators

Emulators are software and/or hardware tools that emulate the execution of quantum algorithms on classical computers. Their qubits emulation capacity is closely related to the amount of memory available and the emulation mode that is implemented, but also on pure computing power [1656].

Terminology

The terminology is not standardized. “Quantum circuit simulator” is widely used in the literature for software that executes a quantum circuit on classical hardware, while “quantum simulator” also denotes a physical or numerical system used to study quantum many-body dynamics. This book uses “quantum emulator” for classical execution of quantum-computing code and reserves “quantum simulation” for the simulation of quantum physical systems [1657]. Figure 104 makes this convention explicit and separates high-level circuit emulation from lower-level models of qubit physics and noise [1658, 1659]. A Bell-test circuit can of course be emulated classically, with sampled measurements generated from the computed probability distribution [1660].

Analog quantum simulators, including neutral-atom platforms, directly implement programmable many-body Hamiltonians such as Ising or XY models. Optimization problems can be mapped to Ising or QUBO forms. Digital quantum simulation represents quantum

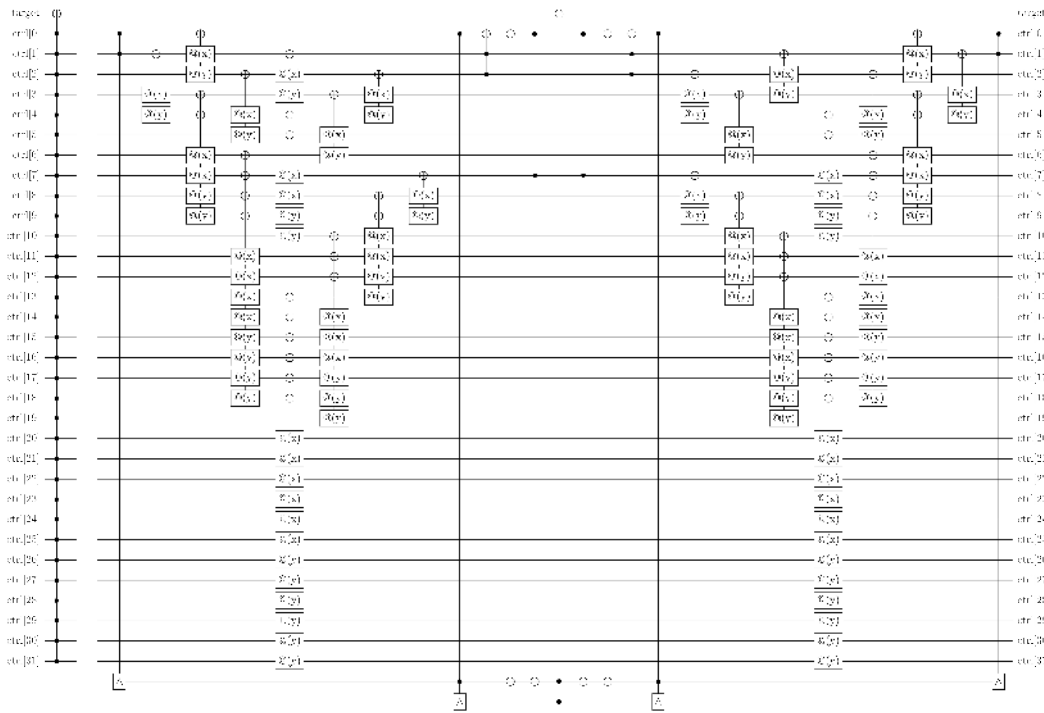


Figure 102: Decomposition of a 32-controlled NOT gate, meaning 32 control qubits plus one target, into 61 Toffoli gates and $\mathcal{O}(\log_2 n)$ depth using two clean ancilla qubits. Source: Craig Gidney in [1567]

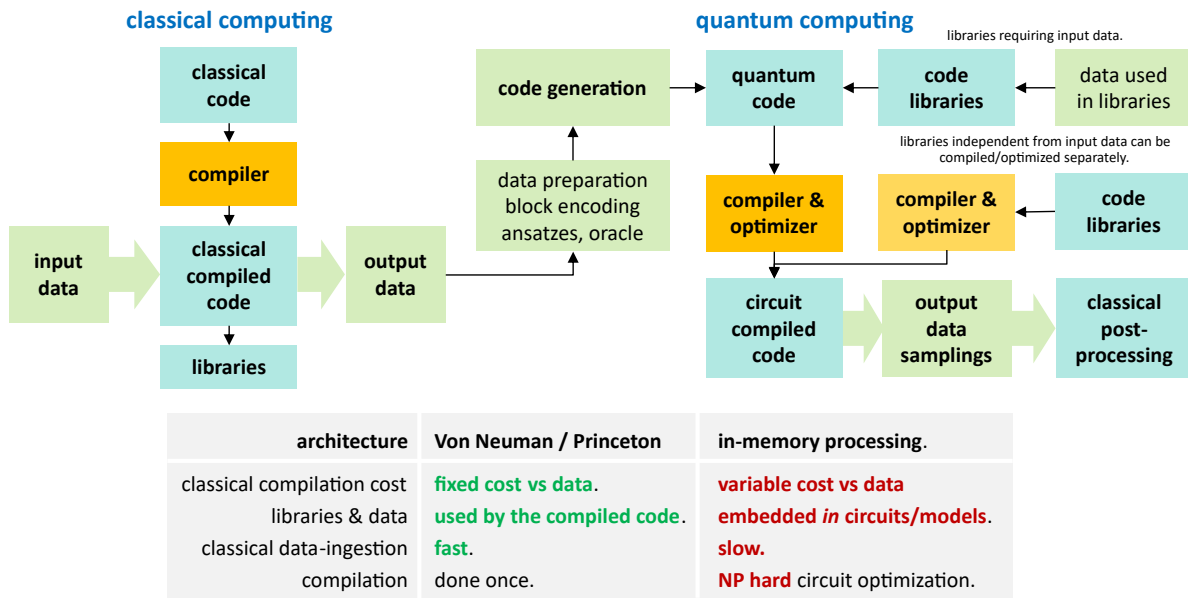


Figure 103: Some differences between classical and quantum computing. Classical compiled code usually consumes input data at run time, although JIT compilation and specialization can make classical compilation input dependent. Quantum programs can similarly separate reusable compiled circuit structure from run-time parameters, while other workloads embed data in state preparation, oracles, block encodings or variational parameters and may require partial recompilation. Several quantum-compilation subproblems, including routing and some synthesis or optimization formulations, are NP-hard. © Olivier Ezratty, 2024-2026.

dynamics with qubits and quantum gates, using methods such as product formulas, variational algorithms or, in FTQC, phase-estimation and qubitization-based methods. Classical many-body simulation includes exact diagonalization, tensor networks, quantum Monte Carlo and electronic-structure methods. Density-functional

theory is a classical electronic-structure approximation, not a quantum-circuit simulator.

Quantum code emulation serves multiple purposes: learning how to program, visualizing the internal data of a quantum algorithm (which is not possible on a real quantum computer), developing quantum algorithms, and testing them at small scale. It can also be used

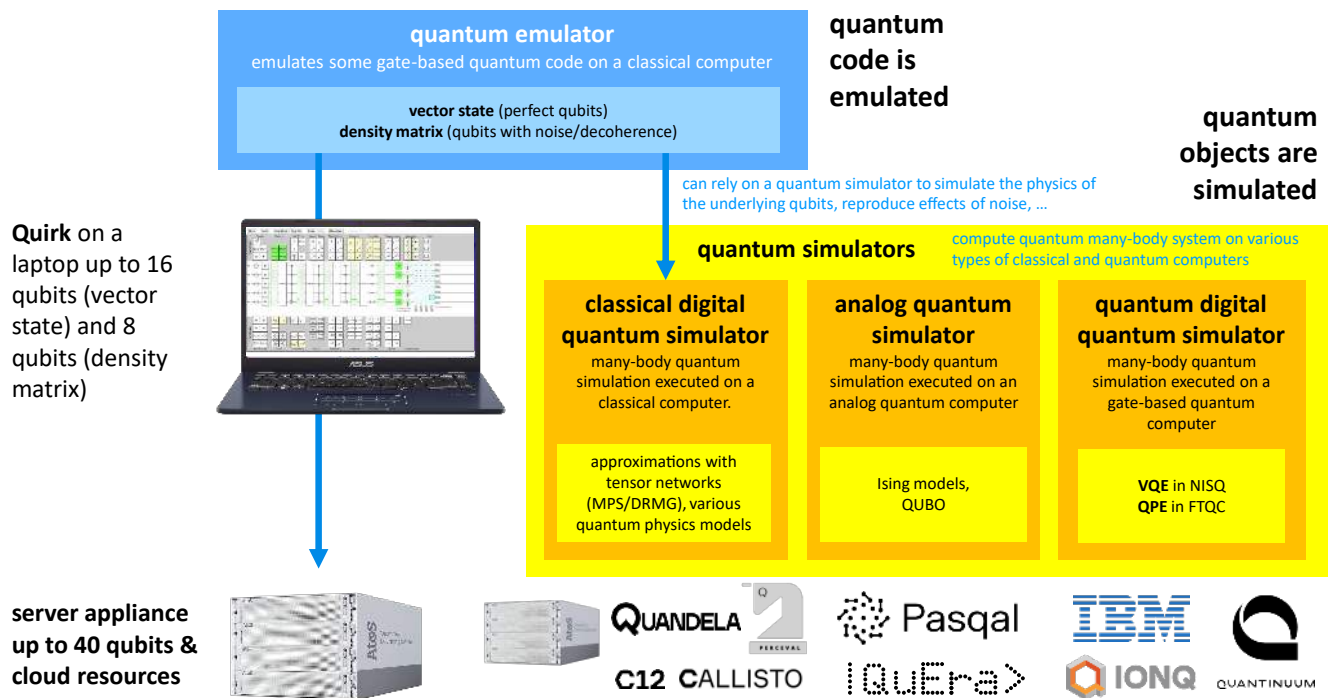


Figure 104: Separating emulation from simulation in quantum computing lingua. Source: Olivier Ezratty, 2023-2026.

to simulate how quantum hardware behaves at a low level, reproducing digitally their defects like noise and other imperfections. For NISQ devices, comparison with the emulated equivalent circuit can help identify the circuit regimes in which hardware output becomes indistinguishable from a chosen noise baseline [1661].

Some tools include lower-level models of sources, devices, noise or qubit physics. Perceval, for example, models linear-optical circuits in mode and Fock-state representations and can include source and detector imperfections, while other tools model the dynamics of specific solid-state qubits, for example with silicon quantum dots qubits [1662]. A random-number generator is needed when the emulator samples measurement shots.

Emulation methods

There are various methods for emulating a circuit of N qubits and a given circuit depth, related to the state-vector and density-matrix formalisms introduced earlier [1663, 1664]. Their cost depends strongly on the circuit structure, entanglement, topology, noise model and on whether the task is to compute amplitudes, expectation values or samples. There is no single ordering from hardest to simplest that applies to all circuits.

Density-matrix emulation stores a $2^N \times 2^N$ complex matrix, namely 4^N complex values or 2^{2N+1} real floating-point values. With complex128 arithmetic this is 2^{2N+4} bytes before implementation overhead. It is therefore much more memory intensive than a pure-state vector. Density matrices are useful for general mixed-state noise and decoherence models, although trajectories and other methods can sometimes avoid storing the full density matrix.

Quantum state-vector emulation stores the 2^N complex amplitudes of an N -qubit pure state. With complex128 arithmetic this requires $2^N \times 16 = 2^{N+4}$ bytes before implementation overhead. A 30-qubit state vector therefore occupies 16 GB, while 50 qubits require 16 PB. One-, two- and three-qubit gates act on 2×2 , 4×4 and 8×8 subspaces without requiring storage of a full $2^N \times 2^N$ matrix. Practical limits depend on memory, communication and gate throughput. Laptops typically reach the high-20s to about 30 qubits, while distributed HPC state-vector engines can exceed 40 qubits and specialized very large-memory systems can reach around 50 qubits [1665]. This approach is also called a Schrödinger-style simulation.

A state-vector emulator keeps the full pure-state vector and can therefore support strong-simulation tasks such as computing selected amplitudes or output probabilities. Strong simulation describes the computational task of evaluating probabilities, while weak simulation means sampling outcomes from the output distribution [1666]. In other words, strong simulation outputs exact mathematical values (amplitudes or probabilities), while weak simulation outputs a sample (sequences of measurement bits).

Tensor-network techniques exploit circuit or state structure to avoid storing the full 2^N state vector [1667–1670]. Tensor-network contraction can be exact, in which case its worst-case cost can still be exponential, or approximate when bond dimensions or contraction paths are truncated. It can support amplitude, probability, expectation-value or sampling tasks depending on the contraction strategy. In September 2019, Alibaba used tensor-network methods to simulate a 70-qubit

Bristlecone-style circuit of depth 34 using 1,449 cloud instances, corresponding to 127,512 CPU processors [1671]. Approximate compression can trade accuracy for computational cost [1672]. In 2021, a Sunway-based implementation simulated Sycamore random-circuit sampling workloads in hundreds of seconds [1673, 1674]. Figure 107 illustrates how these structured methods reach qubit counts far beyond full state-vector storage for suitable circuits.

Machine learning can be broadly used to optimize circuit emulators. It can optimize the underlying tensor network processes themselves with solving the combinatorial problems tensor networks struggle with, particularly for determining the contraction order which is the sequence in which tensors are multiplied. This is done using deep reinforcement learning (RL) and graph neural networks (GNNs) [1675]. The technique is used by Nvidia in their cuTensorNet library. Machine learning can build a surrogate model that predicts the output of a quantum circuit. A GNN or Transformer is trained on a large dataset of quantum circuits and their corresponding output distributions. The model can then quickly infer the expected results [1676]. This is particularly useful for variational quantum circuits where the circuit structure stays the same but only rotation parameters change when using the classical optimizer. Before a circuit even reaches the emulator, machine learning can also optimize its structure to make it easier to simulate. Machine learning can also help simulate qubit noise in the emulator [1677]. Reinforcement learning agents can be trained to apply ZX-calculus rules to decompose the circuit into a much smaller, equivalent version, which the software emulator can then process faster [1678]. Finally, machine learning-based circuit placement (QCP) algorithms can reduce the number of SWAP gates needed, which in turn reduces the number of tensor operations for the emulator [1679].

Here are less common emulation methods:

Weak simulation means generating samples from a quantum circuit's output distribution, either exactly or approximately. It does not mean storing amplitudes without their phases, and it does not require storing all 2^N output probabilities. A weak simulator may use tensor networks, stabilizer decompositions, Monte Carlo methods or other specialized techniques to draw bit strings directly. Strong simulation instead evaluates specified probabilities, marginals or expectation values [1666, 1680].

Feynman-based circuit emulation of simulation methods, particularly those leveraging the path integral formulation, view the qubit register quantum state evolution as a sum over all possible “paths” or histories a quantum system could take from an initial state to a final state. Each path is assigned a complex amplitude, and the total amplitude for a particular outcome is the sum of amplitudes of all paths leading to that outcome.

Pauli propagation is a classical simulation framework that propagates observables, expressed as sums of Pauli strings, backward through a circuit in the Heisenberg picture. Exact propagation can become intractable when the number of Pauli terms grows rapidly, so practical methods truncate small or high-weight contributions and become approximate. This approach is particularly useful for expectation values and is not intrinsically limited by one-dimensional geometry or low entanglement, which can make it complementary to MPS and DMRG methods [1501]. Its efficiency depends on the circuit family, noise, observable and Pauli-term growth.

Hybrid quantum computing emulation emulates the quantum part, and executes the classical part of a hybrid algorithm, as developed in China on a Sunway supercomputer [967].

LLM-based emulation has been proposed for small Grover-algorithm instances [1681]. The reported experiment was limited to 20-qubit instances and did not establish a scaling or performance advantage over established classical simulation methods.

Classical analog emulators could be used for Grover searches emulation [1682]. They use analog electronic circuits with operational amplifiers, resistors and capacitors. It is not conceptually absurd given that many quantum computing concepts are not far from classical analog electronics (interferences, amplification, ...). This is good for learning, but this tool would not scale well, the number of lines growing exponentially with the number of qubits (and also, probably, quantum gates).

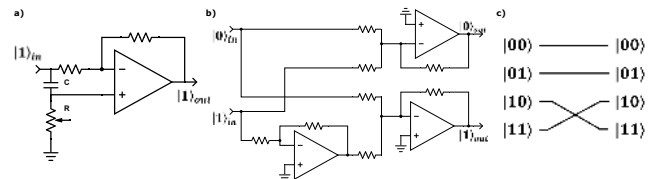


Figure 105: An emulator of a Grover circuit using classical analog electronics. It was tested with only 3 qubits. Source: [1682].

Emulation resources

For a full state-vector emulator, memory grows as 2^N and for a full density matrix as 4^N , while structured methods can scale very differently (Figure 107). A 16 GB machine can support a 30-qubit complex128 state vector in theory [1683], before simulator and operating-system overhead. Practical limits are usually a little lower. Quirk's 16-qubit state-vector and 8-qubit density-matrix limits are implementation and user-interface limits rather than practical memory limits.

A quantum processor may execute an individual circuit shot faster than a full state-vector emulator even at moderate sizes. However, this does not directly create a computational quantum advantage. Classical emulators can compute and reuse the complete state vector, and specialized classical methods may be much faster than

state-vector simulation. No universal crossover exists at a given qubit count, since classical emulation complexity depends strongly on circuit depth, entanglement, gate structure (Clifford vs non-Clifford) and the required output.

Still, a few ideal qubits have been emulated even on early microcomputers, for example two qubits on a Commodore 64 [1684]. The state-vector memory itself is tiny at these scales. A 2024 benchmark of classical quantum circuit emulators showed that ideal randomized circuits with 19 qubits and 19 layers can be emulated in ms to seconds on a single HPC node, depending on the simulator and on GPU acceleration [1685, 1686]. With four Nvidia A100 GPUs, simulation times reach about the second scale only around 26 qubits for these Quantum-Volume-style circuits. Full state-vector simulation scales approximately exponentially with the number of qubits, but circuits below about 20 qubits remain inexpensive to emulate on modern classical hardware (Figure 106).

Specialized server appliances such as Bull/Eviden/Atos'QLM fit in a datacenter rack, are designed to manage a very large amount of memory and can support the emulation of up to 40 qubits. More than 40 qubits can be emulated on massively parallel architectures and supercomputers or on a large number of clusters.

For full state-vector emulation, both memory capacity and memory bandwidth become major bottlenecks. A 50-qubit complex128 state vector requires 16 PB, about 18 PB in decimal units. A 96-qubit state vector would require 2^{46} times more memory than a 50-qubit one, namely 2^{100} bytes, about 1.27×10^{30} bytes. Exponential state-space growth therefore rapidly outruns improvements in conventional memory capacity.

Nevertheless, the number of emulated qubits on supercomputers is still constantly increasing. China research teams have been the most active in this emulation race, particularly at Alibaba and Huawei, with several records set in 2018 up to 2021. The most recent advances are about distributing quantum code emulation on multiple CPUs and GPUs [1687, 1688].

Origin Quantum, a multi-role (hardware, software) startup from China in partnership with the Guang-Can Guo team from the University of Science and Technology of China, emulated 64 qubits with a 22-depth algorithm on a cluster of 128 nodes in 2018 [1689]. They used a method to transform combinations of CZ gates (conditional Pauli Z gates) and single-qubit gates into simpler sub-circuits that do not need to be interleaved. They also projected that the method could simulate 72 qubits at depth 23 on a supercomputer running for 16 hours, an estimate rather than a demonstrated run. This work shows that two key parameters condition the emulation capabilities in classical computers: not only the number of qubits but also the number of quantum gate sequences. The larger the number of qubits emulated, the fewer quantum gate sequences we can simulate.

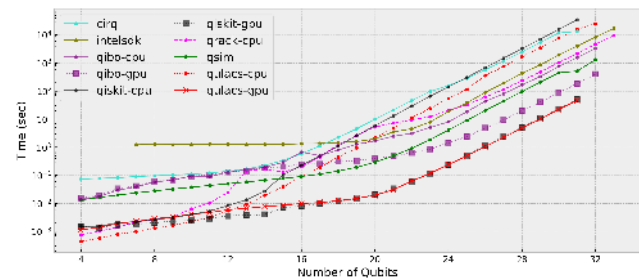


Figure 106: Comparative performance of CPU- and GPU-based emulators for running square random circuits of width N and depth N , of the type used in Quantum Volume benchmarking. The Quantum Volume metric itself is 2^N only when a processor successfully executes the corresponding width- N , depth- N benchmark circuit. Source: [1685, 1686].

Alibaba achieved a second record in 2018 with 81 qubits and 40 quantum gate sequences [1692]. Their Taizhang simulation exploited a method created by Igor Markov and Shi Yaoyun in 2005 that allows a quantum algorithm to be distributed over a farm of thousands of servers [1693]. The Alibaba Quantum Laboratory is managed by the same Shi Yaoyun, a professor at the University of Michigan. Their simulations included 100 qubits over 35 layers ($10 \times 10 \times 35$), 121 qubits over 31 layers ($11 \times 11 \times 31$) and 144 qubits over 27 layers ($12 \times 12 \times 27$).

Also in 2018, **Huawei** reported another record with its “HiQ Cloud” service, capable of emulating 42 to 169 qubits [1694]. The method was like the one used by Alibaba. The 42 qubits were simulated in “full amplitude” mode. 81 qubits were simulated with “a single amplitude” and 169 qubits with a single amplitude and a small number of quantum gates. Other records have been broken in the USA in 2019, such as **Google**’s record with NASA, the University of Illinois and the Oak Ridge laboratory with 49 to 121 qubits on the IBM Summit [1695].

IBM researchers reported a 56-qubit simulation in 2017 using Vulcan, an IBM Blue Gene/Q supercomputer operated by Lawrence Livermore National Laboratory, rather than an IBM-owned production system. The Oak Ridge National Laboratory team developed **XACC** (eXtreme-scale ACCelerator), a framework for hybrid workflows combining classical HPC resources with gate-based or annealing quantum backends [1696, 1697].

Dell announced in 2021 its hybrid solution combining classical computing and quantum emulation using its Dell EMC PowerEdge R740xd server appliance and IBM’s Qiskit Runtime.

Quokka Computing (2024, Australia) is a company created by Chris Ferrie which launched a software and hardware emulator supporting up to 30 qubits. It is a simple USB dongle using a system-on-chip processor akin to the ones you can find in cheap smartphones and with 4 GB of memory. It runs attached to your laptop. The project was launched on Kickstarter and positioned initially as a “quantum computer”. There was of course some backlash from the quantum computing community.

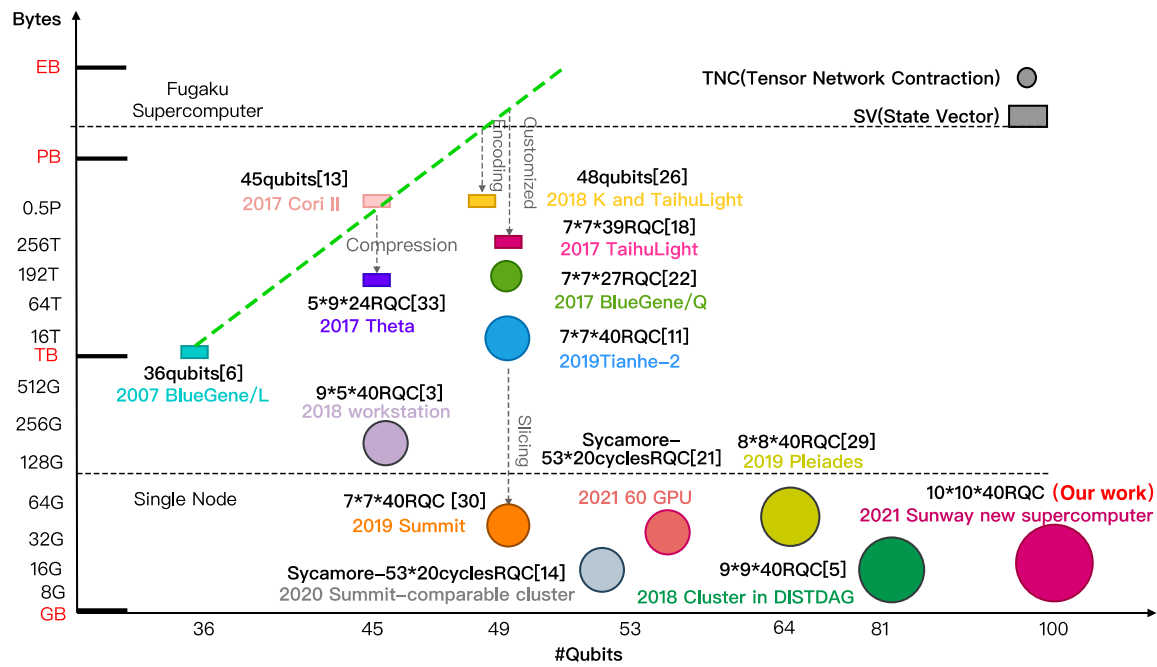


Figure 107: The capacities of various quantum emulators in number of qubits (X) and memory capacity (Y). The size of the circles and rectangles corresponds to the circuit depth. However, the real capacity to simulate a quantum circuit depends on many more parameters than the circuit depth and breadth. Are at play their gates, symmetries, entanglement structure, input state and measured observables [1690]. Source: [1673].

Methods		Memory	Run time	Approx. or exact	Noiseless or noisy	Application regime
Full state	State-vector	worst $O(2^N)$	worst $O(m2^N)$	Exact	Noiseless [#]	General, good for small circuits [*]
	Density-matrix	worst $O(2^{2N})$	worst $O(m2^{2N})$	Exact	Both	General, good for small circuits ⁺
MPS state/MPO		$O(N\chi^2)$	$O(N\chi^6)$	Approx.	Noisy	General, good for shallow circuits
Tensor network		On demand	$O(e^W)$	Both	Both	General, good for shallow circuits
Stabilizer		$O(e^{m_T})$	$O(e^{m_T})$	Approx.	Both	Circuits dominated with Clifford gates, particularly in QEC

Figure 108: Table showing representative classical memory and run-time scalings for gate-based quantum emulation methods. Full state-vector simulation is exact and structure agnostic for pure states but scales as 2^N in memory. Density-matrix simulation scales as 4^N and supports general mixed-state noise. Stabilizer simulation is polynomial and can be much cheaper than state-vector simulation, but exact stabilizer methods are restricted to Clifford circuits. MPS and general tensor-network methods can reach much larger qubit counts when entanglement and circuit structure remain favorable, and can be either exact or approximate. Source: [1691].

It is somewhat useless as a dedicated hardware, but the software is worthwhile.

Bull

France



Bull, formerly Atos and Eviden, has sold since 2017 an Intel and Nvidia-based quantum emulator appliance, Qaptiva, formerly named Quantum Learning Machine (QLM). It is also one of the few remaining HPC vendors.

It has been widely adopted worldwide with 39 customers as of mid-2025. Bull customers include the US DoE's ORNL, CEA, the University of Reims, the cybersecurity research department of the University of Applied Sciences of Upper Austria in Hagenberg, the UK Hartree Science and Technology Facility Centre (STFC), the C-DAC (Centre for Development of Advanced Computing) in India, in its Quantum Computing Experience Center, in Japan, in Finland at the CSC IT Center for Science Kvasi in collaboration with IQM, in the new

Quantum Integration Centre (QIC) from LRZ (Leibniz Supercomputing Centre) of the Bavarian Academy of Sciences and in Spain (CESGA) and at CERN in Geneva. In June 2020, Bull added support for Nvidia GPGPUs by integrating 2 to 32 Nvidia GPUs in its Qaptiva appliance. It multiplied its computing power by 12 compared to the Intel-based initial version. This system was first delivered in December 2020 to the Irish HPC center (ICHEC).

In May 2019, Atos launched myQLM, a quantum programming tool for researchers, students, and developers. It is a Python-based development environment that allows users to simulate quantum programs on their own computer. Programming is carried out in AQASM (Atos Quantum Assembly Language) and pyAQSM. To access a number of qubits that exceeds the current capacity of PCs, i.e., more than 20 qubits, developers can run their code on an Atos Quantum Learning Machine simulator in the cloud, but at a charge. Atos also enables the sharing of quantum practices, libraries and application codes. Atos offers one of the open-source translators of myQLM code to other quantum programming environments. In September 2020, this software offer became free of charge to all audiences [1698]. myQLM supports the emulation of the three quantum computing paradigms: quantum annealing, quantum simulation and gate-based programming. It supports several modes: state-vector emulation using various compression methods (stabilizers, binary decision diagrams, tensor networks MPS) as well as the simulation of qubit noise. In 2023, Bull (Eviden) extended its emulation capabilities to over 100 qubits thanks to the use of tensor networks and added a new framework for integrating HPCs with quantum computers. It introduced Q-Pragma, a framework supporting C++ that helps HPC centers utilize quantum algorithms and implement HPC-Quantum hybridization [1699]. In November 2024, Bull (Eviden) extended its state-vector emulation support to 44 qubits with Qaptiva HPC, the version of the Qaptiva emulator that runs on HPC clusters. Bull's offering also contains Qaptiva Access which schedules quantum workloads across code emulation on Qaptiva server appliances and HPC clusters, and real quantum computing on QPUs.

Bull participates in the European Flagship projects **AQTION** (quantum accelerator), **PASQuanS** (analog quantum simulator) and **NEASQC** (NExt ApplicationS of Quantum Computing, which they coordinate).

It is also heavily involved in the EuroHPC project, which includes the **European Processor Initiative**, an initiative to develop a processor adapted to the needs of supercomputers and on-board as well as autonomous vehicles.

And, of course, in the **(HPC|QS)** project which will deploy in Finland, Germany and France three hybrid quantum solutions combining a supercomputer and a quantum computer.

Nvidia

USA, 1993



Nvidia progressively became an important player in quantum computing through its GPUs and DGX systems, the cuQuantum simulation libraries, the CUDA-Q hybrid quantum-classical programming platform and GPU-QPU integration technologies such as DGX Quantum and NVQLink. QODA was the original name of CUDA-Q.

GPGPU: Nvidia is the worldwide leader with its powerful general purpose GPU chips. It started in 2017 with its V100 (Volta) chip, followed by its A100 (Ampere) in 2020, its H100 (Hopper) in 2022 and its B100 (Blackwell) in 2024 with 208 billion transistors. Each of these chips have more transistors, ALU (arithmetic-logic units) and tensors (matrix adders/multipliers). As a reminder, Nvidia's CUDA is a parallel-computing platform and API that helps low-level developers directly access the features of its GPGPUs and (game-based) GPU. Nvidia became a worldwide leader with its GPGPUs in at least four domains: machine and deep learning (including LLMs), crypto-mining, scientific computing running on HPCs, and then, quantum code emulation [1700].

Since 2023, the arm-based Grace CPU complements the H100 or B100 in an integrated fashion on multi-die chips. It enables Nvidia to avoid integrating AMD or Intel chips in its servers. Nvidia GPGPU are integrated in third-party hardware or on its own DGX server appliances, usually based on eight of their GPUs, HBM fast memory, NVLink inter-CPU/GPU connectivity (7.2TB/s full duplex bandwidth), and Mellanox-based InfiniBand Spectrum-X connectivity.

Nvidia GPGPUs are now all over the place in over half of the TOP500 HPCs in the world, and is supported by various quantum emulation cloud offerings from JUQCS-G (Jülich, supporting 50 qubits since 2025 [1701]), Qgate (NVAITC), Qiskit-AER (IBM), QuEST (Oxford), SV1 (Amazon Web Services) and Vulcan (QC Ware). In March 2024, Nvidia announced that Japan's new ABCI-Q Fujitsu supercomputer built for AIST is using 2,000 Nvidia H100 chips and enables quantum emulations. It will deploy in 2025.

In 2022, Pasqal (France) deployed an on-premises Nvidia DGX POD to run digital simulations of its quantum simulator using cuQuantum. Qubit Pharmaceuticals (France) also deployed Nvidia DGX servers to run its own quantum emulator.

In March 2023, Nvidia formed a partnership with Quantum Machines to combine their DGX Quantum server appliance with Quantum Machines' OPX+ quantum qubit control hardware. The new DGX is using the Grace Hopper GH200 powerful chip including both custom arm cores Grace CPUs and GH100 GPUs introduced in May 2023 [1703]. Both systems are connected at the PCIe bus level enabling very low sub-microsecond

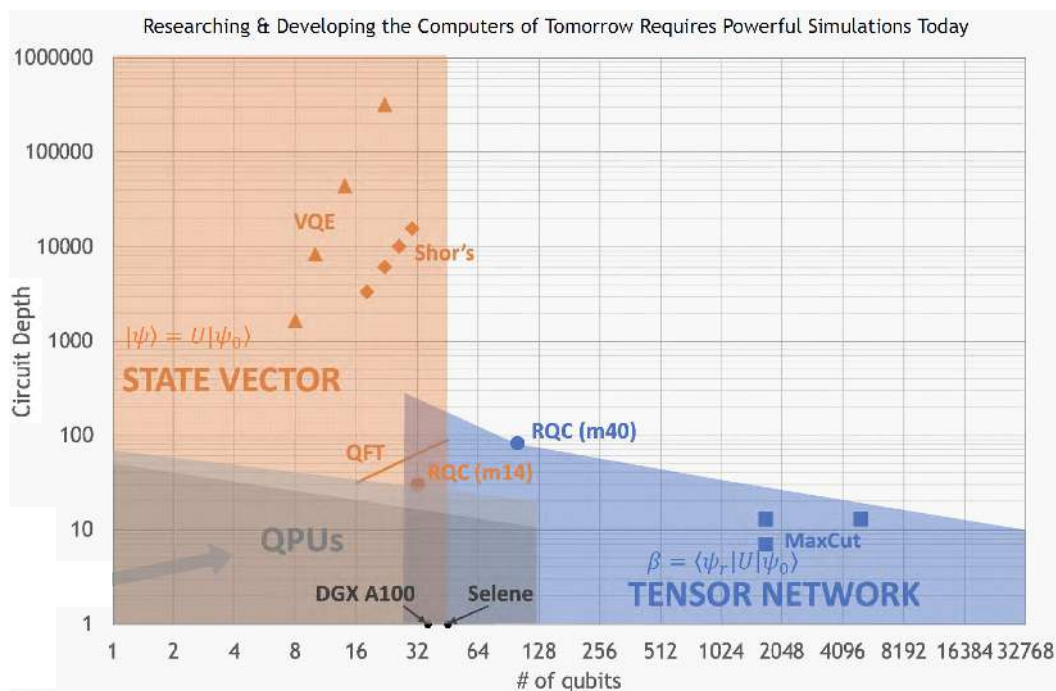


Figure 109: Nvidia positioning the scope of state vector quantum emulation in a regime with fewer than 32 qubits but no limitation in the circuit depth (on the y-axis), and tensor-network emulation, which can reach hundreds of qubits for shallow circuits. This last solution can reach hundreds of qubits for favorable low-entanglement or otherwise compressible circuits, but its cost can become prohibitive as contraction complexity grows, so it is not a general substitute for NISQ hardware. The chart shows that classical emulation has a broader scope than existing NISQ quantum computers (in grey). The Shor point in the scatter plot corresponds to running Shor integer factoring algorithm on very small RSA keys, not the sought-after RSA-2048 key. It is the same with the VQE point which corresponds to rather small chemical simulation requirements. Source: Nvidia [1702]. Added in 2023.

latency between Nvidia GPUs and QPUs. In the future, it will enable fast implementation of quantum error correction codes.

Quantum software. Most Nvidia quantum software tools are related to quantum code emulation, but they can also drive QPUs (Figure 110).

cuQuantum: is an SDK running on top of Nvidia GPGPUs that implements gates-based programming emulation. It was announced in April 2021, in beta in November 2021 [1704] and released in March 2022 [1705].

It supports both state-vector emulation with tens of qubits (**cuStateVec**) and a less resources-hungry tensor-network based emulation that supports up to thousands of qubits (Figure 109). They integrated cuStateVec into qsim, Google Quantum AI's state vector simulator that can be used through Cirq. cuStateVec can also be used with Qiskit Aer, IBM's emulation framework [1706].

CUDA-Q: is Nvidia's open-source hybrid quantum-classical programming platform for building, simulating and executing quantum applications across classical accelerators and QPU backends. It is now used by multiple quantum hardware vendors for QEC simulation (QC Design), for QEC integration (Inflection), quantum error mitigation (QEDMA).

nvq++: it is the CUDA-Q C++ compiler driver. CUDA-Q lowers quantum kernels through MLIR-based intermediate representations and can emit QIR. QIR is an LLVM-based intermediate representation for hybrid quantum-classical programs, not a low-level machine language.

QODA: the Quantum Optimized Device Architecture was introduced by Nvidia in July 2022 as a unified hybrid quantum-classical programming platform [1702]. Nvidia renamed QODA to CUDA Quantum, now CUDA-Q, in March 2023. It should therefore be treated as the historical name of CUDA-Q rather than a separate platform [1707].

NVQLink: is Nvidia's open GPU-QPU interconnect architecture for tightly coupled hybrid systems. Nvidia specifies up to 400 Gb/s GPU-QPU throughput and latency below 4 μ s, targeting low-latency control, hybrid algorithms and real-time quantum error correction. It was introduced in October 2025 and developed with a broad hardware and research ecosystem [1708].

Nvidia Quantum Cloud: it is the service platform announced in March 2024 that supports CUDA-Q. It integrates third-party software components like the Generative Quantum Eigensolver, developed by the University of Toronto, which uses a LLM to find the ground-state energy of a molecule more quickly, Classiq's compiler,

and QC Ware’s Promethium quantum chemistry and molecular simulation library.

Nvidia’s software tools adopters include QC Ware (for quantum chemistry and QML using cuQuantum on the Nvidia A100-based Lawrence Berkeley National Laboratory Perlmutter supercomputer launched in 2021), ORNL (using cuQuantum in TNQVM, a framework for tensor network quantum circuit simulations), Xanadu (using cuQuantum in their PennyLane framework for QML and quantum chemistry), Classiq (in their Quantum Algorithm Design platform) and Zapata Computing (in Orquestra). Nvidia also works with Google Quantum AI, IBM, IonQ and Pasqal.

Nvidia also proposes various useful libraries: cuSOLVER, a library for decompositions and linear system solutions for both dense and sparse matrices, it is based on cuBLAS, a GPU-accelerated library for accelerating AI and HPC applications, and cuFFT, a CUDA-based library to run fast-Fourier transforms.

cuPQC: Nvidia’s cuPQC SDK provides GPU-accelerated post-quantum cryptography primitives. Its public-key library supports the NIST-standardized ML-KEM family, derived from CRYSTALS-Kyber, and ML-DSA, derived from CRYSTALS-Dilithium, including key generation, encapsulation/decapsulation and digital signatures.

In March 2025, Nvidia announced the creation of its Accelerated Quantum Research Center, or NVAQC, in Boston, mostly for studying the integration of QPUs with AI supercomputers.

Emulators inventory

Beyond the solutions mentioned above, many software emulation tools are available and are mostly always open-source [1656]. Here they are in alphabetical order.

Amazon Braket emulator supports state vector, tensor network and density matrix emulations with respectively SV1, TN1, and DM1 [1709].

DiaQ is a full state vector emulator from North Carolina State University and the Oak Ridge National Laboratory [1710]. It is optimized for the use of sparse matrices.

Fermioniq (2021, the Netherlands, €300K) is the developer of Ava, a tensor network based quantum emulator running on GPUs, still in full state vector mode. Its computing speed scales much better (lower) than Nvidia’s cuState state vector emulator and support up to a hundred qubits with up to 7000 gates. It can also use tensor networks for compression [1711]. The company created by Harry Buhrman, Ido Niessen, Chris Cade and Jürgen Sandig is a spin-off of the University of Amsterdam, Centrum Wiskunde & Informatica (CWI) and QuSoft. The emulator is cloud based. Since November 2024, Ava is available through the NVIDIA CUDA-Q cloud platform.

Fujitsu has a 39-qubit CPU-based state vector quantum simulator hardware that is based on 512 PrimeHPC

FX700 2U server nodes using the arm based A64FX Fujitsu CPUs. This hardware used the Qulacs software emulator, created by Osaka University and QunaSys. This hardware and software offering was initially launched in March 2022. In February 2024, Fujitsu announced that its quantum emulator would speed up by x200 quantum-classical hybrid algorithms.

It enables simultaneous and distributed processing of a large number of quantum circuits emulations on 1024 Fujitsu A64FX chips connected with each other by InfiniBand. This acceleration was due to the use of a NISQ VQE algorithms and with emulating 36 qubits. Emulation was distributing the many Pauli strings observable circuits on these chips [1712].

Google’s state vector based emulator qsim is written in C++ and can simulate 30 qubits on a laptop and up to 40 qubits in Google Cloud or on a 90 core Intel Xeon workstation.

Hyperion is a state vector emulator developed by Qubit Pharmaceuticals running on GPUs and used to evaluate chemical simulation algorithms at small scale [68, 1713].

Intel IQS (formerly qHiPSTER) created by Intel in 2016 [1714]. It supports up to 42 qubits pure states in state vector simulation mode. As of 2023, Intel released its Quantum SDK 1.0. with a new emulator supporting C++ language [1715].

LIQUi|> is a software architecture and tool suite for quantum computing developed by the Quantum Architectures and Computation Group (QuArC) at Microsoft Research starting back in 2016. It includes a programming language, optimization and scheduling algorithms, and quantum simulators. LIQUi|> can be used to translate a quantum algorithm written in the form of a high-level program into the low-level machine instructions for a quantum device.

phase2 is a full-state vector emulator coming from Germany that can be distributed on multiple CPUs and GPUs [1716].

Psitrum is a software emulator developed in Matlab in Saudi Arabia. It computes the algorithm matrix, the density matrix of the simulated qubit register and the output state vector. It can simulate quantum noise and has a graphical circuit designer. It provides density matrices visualization tools [1717].

QEA, *aka* Quantum Emulation Accelerator, is a state vector coming from Japan that optimizes classical resources (memory allocation, data sharing, flexible arithmetic) [1718].

Qandle is a PyTorch compatible state-vector simulation for quantum circuits from LMU Munich [1719].

Q-Gear is a state vector emulator converting Qiskit instructions into CUDA-Q to support Nvidia GPGPUs [1720].

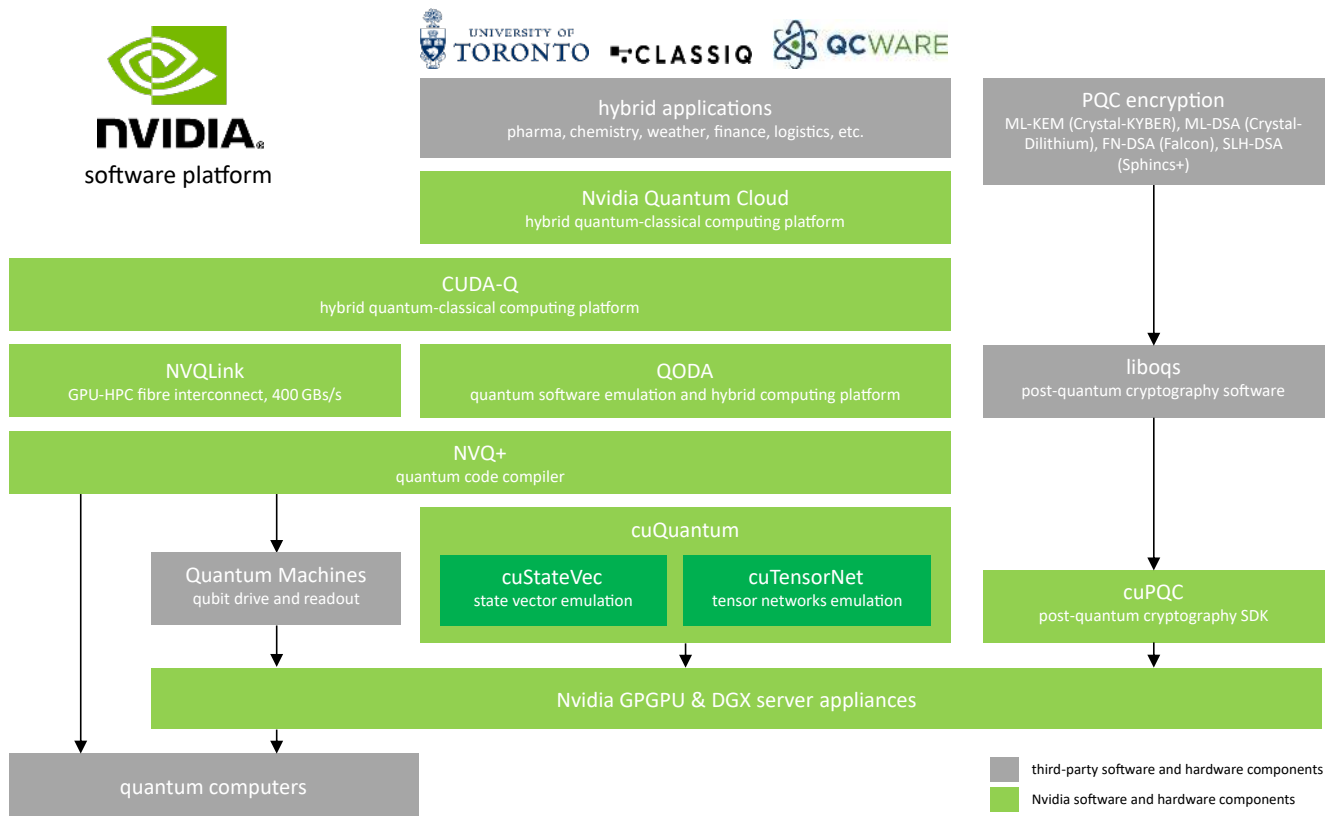


Figure 110: Nvidia quantum platform, assembled by Olivier Ezratty, September 2024-September 2026.

QCLAB+ is a quantum circuit emulator from the Lawrence Berkeley National Laboratory that works on Nvidia GPUs [1721].

Qibo is an open-source framework for quantum emulation with qibojit for full-state vector simulation, qibotn for tensor-network simulation and Clifford for specialized execution. It was created out of TII in Abu Dhabi and in Italy. It can help teach how Bell inequalities are violated [1722]. It was complemented in 2025 by Qiboml, a library for orchestrating quantum and classical components in hybrid machine learning workflows [1723].

Qimax is a GPU-based emulator from Japan which uses an extended stabilizer formalism through a novel encoding-decoding architecture optimized for parallel computation [1724].

Qiskit Aer is IBM's emulation software solution launched in December 2018. It supports simulations in state vector mode and density matrix mode as well as in the more exotic matrix product state (adapted to weakly entangled states) and stabilizer modes (supporting only Clifford group gates).

QOkita is a QAOA circuit emulator designed by JPMorganChase and Argonne Lab with significant speedup compared to classical state-vector emulators [1725].

Qrack is a C++ quantum bit and gate simulator, with the ability to support arbitrary numbers of entangled qubits — up to system limitations [1726]. Suitable for embedding in other projects, the Qrack QInterface

contains a full and performant collection of standard quantum gates, as well as variations suitable for register operations and arbitrary rotations. The developers of Qrack maintain a fork of the ProjectQ quantum computer compiler which can use Qrack as the simulator, generally. This stack is also compatible with the SimulaQron quantum network simulator. Further, it maintains a QrackProvider for Qiskit. Both ProjectQ and Qiskit integrations for Qrack support the PennyLane stack. For Qiskit, a fork of the Qiskit plugin provides support for a “QrackDevice”.

QSeqSim, a Qiskit-integrated symbolic backend that simulates while-loop quantum programs and their induced sequential quantum circuits [1727].

Quantum Computer Emulator (QCE) is a software tool that emulates various hardware designs of quantum computers [1728]. QCE simulates the physical processes that govern the operation of a hardware quantum processor, strictly according to the laws of quantum mechanics. QCE also provides an environment to debug and execute quantum algorithms under realistic experimental conditions. The software contains a Graphical User Interface (GUI) and the simulator itself.

Quantum Programming Studio is a web-based graphical user interface designed to allow users to construct quantum algorithms and obtain results by simulating directly in the browser or by executing on real quantum computers. The circuit can be exported to multiple quantum programming languages/frameworks

and can be executed on various simulators and quantum computers.

Quantum Rings Inc (2022, USA) is a startup created by Bob Wold (CEO) and Venkat Kasirajan (CTO) that provides a tensor network based emulator supporting up to 100 qubits on a laptop with 64 GB of memory. It supports Qiskit and QASM. It is in private preview since December 2023 and is provided as a SaaS (cloud) offering. They are quite optimistic, planning from \$230M to \$450M of yearly revenue by 2030.

Quantum Tea (2023, Italy) is an academic open-source software project, *aka* Q-TEA for “Quantum Tensor network Emulator Applications”. It is a Python software suite for tensor-network simulation developed by the Quantum Information and Matter group at the University of Padova. The suite is organized as four packages: Quantum Tea Leaves (qtealeaves) is the lower-level engine for solving the Schrödinger and Lindblad equations, Quantum Matcha Tea (qmatchatea) runs quantum circuits as tensor networks with Qiskit integration and noise models for emulating realistic devices, Quantum Red Tea (qredtea) is the tooling layer that switches between tensor backends (NumPy, PyTorch) [1729] and hardware (CPU, GPU, TPU) and Quantum Chai Tea (qchaitea) tensor-network machine learning. VulQano, targets quantum circuit compilation and QUBO solving. INFN also provides a “QuantumTEA-as-a-Service” cloud offering and HPC integration on CINECA clusters.

Qubit101 simulator is a user-friendly quantum circuit editor and simulator. The tool helps users to create, modify and save quantum circuits.

Along with this, users can simulate its effect over a pre-defined quantum state, watch the evolution of the state stage by stage, together with the possible measurements results, use other quantum circuits as gates, so complex circuits can be easily created and finally simulate an almost arbitrary number of qubits. Supported platforms include Rigetti Forest, IBM Qiskit, Google Cirq and TensorFlow Quantum, Microsoft Quantum Development Kit, Amazon Braket and more.

Queen V2 is a state vector emulator optimized for Nvidia GPGPUs coming from NTU Taiwan [1730].

QuEST (Quantum Exact Simulation Toolkit) is an open-source C/C++ high-performance quantum circuit simulator created by Simon Benjamin’s Quantum Technology Theory Group at Oxford University. It supports multithreading, MPI distribution and GPU acceleration including CUDA, rather than “QUDA instead of CUDA”. For a single complex128 state vector, 2 GiB stores at most 27 qubits and 256 GiB at most 34 qubits before overhead. Larger state vectors require distributed memory across nodes. QuEST also supports density matrices and noise channels, with hardware-scale limits determined by the aggregate memory and backend configuration [1731].

QuIDDPro is a fast, scalable, and easy-to-use computational interface for generic quantum circuit simulation.

It supports state vectors, density matrices, and related operations using the Quantum Information Decision Diagram (QuIDD) data structure. Software packages including Matlab, Octave, QCSim, and libquantum, have also been used to simulate quantum circuits.

Quirk runs in your browser and even on your smartphone. It was developed in 2016 by Craig Gidney before he joined Google in 2017. It is very useful to learn quantum circuit encoding and understanding what’s happening on the data in qubits. Then, you move to learning how to code with scripting languages like Python for larger codes since Quirk is limited to supporting 16 qubits.

Qymera is an original proposal from TU Delft that uses relational databases (ala SQL) to run emulation. It was created mostly for educational purposes [1732].

SandboxAQ and Google used DMRG-style tensor-network methods on TPU-v3 hardware to study quantum many-body ground states [1733]. The reported bond dimension $\chi = 2^{16} = 65,536$ is not a “number of entanglements”. It is the maximum Schmidt-rank dimension retained across a tensor-network bond. For an MPS, it bounds the bipartite entanglement entropy across that cut by $\log_2 \chi = 16$ bits, while the computational cost grows polynomially with χ and can therefore still be very large [1734].

ScaffCC is a compiler and scheduler adapted to the Scaffold programming language supporting the LLVM infrastructure. It supports QASM.

SimQubit is a GUI quantum circuit simulator, written on top of the Q++ (sourceforge.net/projects/qplusplus) quantum templates. It allows editing of quantum circuits and applying them to quantum states, with multiple ways to view the output probabilities.

SimulaQron is a quantum circuit simulator developed by Stephanie Wehner et al. at QuTech supporting up to 34 qubits.

SOFT is a circuit emulator from China launched in 2026 that supports noisy quantum circuits containing non-Clifford gates. It can be used to simulate QEC and FTQC circuits including magic state factories [1735].

staq is another emulator available as a GitHub repository, staq is a modern C++ library for the synthesis, transformation, optimization and compilation of quantum circuits authored by softwareQ Inc (Canada, 2017). under the MIT License. It is usable either through the provided binary tools or as a header-only library that can be included to provide direct support for parsing & manipulating circuits written in the OpenQASM circuit description language. Inspired by Clang, staq is designed to manipulate OpenQASM syntax trees directly, rather than through an intermediate representation which makes retrieving the original source code impossible. OpenQASM circuits can be inspected and transformed (in most cases) without losing the original source structure. This makes staq ideally suited

for source-to-source transformations, where only specific changes are desired. Likewise, this allows translations to other common circuit description languages and libraries to closely follow the OpenQASM source.

TANQ-Sim is a density matrix based simulator from the Pacific Northwest National Laboratory, Oak Ridge National Laboratory and Microsoft [1736].

TensorCircuit-NG is a scalable, tensor-based quantum emulator that fuses quantum circuits, tensor networks, and neural networks into a single end-to-end differentiable computational graph. Built natively on JAX, TensorFlow, and PyTorch, it leverages automatic differentiation, just-in-time compilation, and vectorized parallelism to efficiently simulate complex quantum operations. It can simulate gate-based circuits, analog dynamics, qudit systems, and support noise modeling [1737].

TornadoQSim is a quantum circuit emulator written in Java that can leverage various hardware acceleration architectures and backends [1738].

UNIQuE is an emulator from the University of Maryland launched in 2024 which natively support algorithm patterns like a QFT and QPE [1739].

Wolfram Quantum Framework is a “paclet”, a quantum code emulator package for the Wolfram Language Paclet Repository. It contains a library of pre-built quantum functions like the Grover search algorithm and a QFT. It is integrated with the Wolfram Cloud offering and Amazon Braket.

In a last resort, you can also develop your own emulator [1740].

Special emulators

There are then QPU emulators and/or simulators that support specific quantum hardware platforms:

- For cold atoms, with **Pulser**, a cold atoms simulator developed by Pasqal [1741], **MIMIQ** from QPerfect, and another one created by the University of Illinois at Urbana-Champaign and HRL Laboratories [1742].
- For photonic quantum computers with **Perceval**, a photon qubits emulator and simulator developed by Quandela, **Piquasso**, an open-source emulator of photonic quantum computers [1743], **SOQCS** also for photon qubits simulations [1744], for boson sampling and Gaussian boson sampling experiments [1745–1749]. In that case, these are indeed simulators since they simulate the lower-level operations of Boson sampling physics.
- For emulating chemical simulation circuits containing many rotation gates [1750] or for supporting fSim gates, which enable fermionic simulations [1751].
- For hybrid quantum computing simulations [1752].
- For emulating analog quantum computing on a gate-based quantum computer [1753]. Under this book’s

terminology, this is emulation rather than physical quantum simulation.

- Stim is an emulator made for simulating quantum error correction codes, created by Craig Gidney from Google in 2021 [1754].
- The emulation (or simulation?) of superconducting qubit modules with... superconducting qubits [1755].
- Emulating qudit systems [1756].
- Using in-memory processing in specialized Upmem classical DRAM chips [1757]. It accelerates emulation and can reduce its energy cost.
- Using FPGAs, like FQsun from Japan and AMARETTO from Italy, the main benefit being its power efficiency [1758, 1759].

Other proposed academic emulators are dedicated to algorithms like the quantum linear solver applied to sparse data, expanding the scope of emulation capacities on classical hardware [1760], the combinatorial NchooseK model [1761], or for quantum SVM [1762].

Some emulators are adapted to NISQ workloads [939, 1763], while specialized approximate or structure-exploiting methods can report simulations with hundreds or even about 1,000 qubits [1764, 1765]. Such qubit counts are not about arbitrary full state-vector emulation because they depend on circuit structure, entanglement, approximation and the requested observable or sampling task. Other tools specifically model device defects [1766] or particular algorithms such as QAOA and QAOA-based Max-Cut [1767, 1768].

The lower a quantum emulator supports the quantum simulation of its underlying qubits, the closer it resembles a qubit physical circuit EDA (Electronic Design Automation) software tool, like the ones proposed by Cadence and Siemens/Mentor Graphics. Solid state qubit EDA are still poorly integrated, and many qubit designers are still relying on custom EDA tools like for this physics simulator of bosonic qubits inner dynamics [1769].

2.1.8 Resource estimators

Resource estimators convert an algorithmic workload and a set of hardware and QEC assumptions into resource metrics such as logical and physical qubit counts, non-Clifford resources, space-time volume, error budgets and runtime. Inputs can include logical gate counts or a circuit, physical error rates, gate and measurement times, connectivity or architecture assumptions, the QEC code and the target algorithmic failure probability [1773]. They are particularly useful for prospective FTQC studies, where the required hardware does not yet exist at application scale.

- **QuRE** was proposed in 2013 by USC, UCSB and Berkeley [1774].

Q Are there resource estimators for NISQ circuits?

There is no broadly accepted, hardware-agnostic NISQ resource estimator playing the same role as FTQC estimators. NISQ performance depends strongly on calibration data, noise, shot budgets, optimizer convergence and error-mitigation overhead. Subcircuit Volumetric Benchmarking (SVB), proposed in 2025 by Sandia researchers, combines decomposition of an algorithm into executable chunks with volumetric benchmarking of those chunks [1770]. It is therefore closer to a hybrid benchmarking and feasibility-estimation methodology than to a universal NISQ resource estimator.

It is possible that demanding NISQ variational algorithms with relatively deep ansatzes will benefit from early stage FTQC platforms. This is what JPMorganChase is expecting for running large scale QAOA algorithms [1771]. This projection may be optimistic. It assumes random 8-SAT instances and a polynomial speedup obtained by combining QAOA with amplitude amplification. Under the study's stated surface-code, error-rate and solver assumptions, the modeled crossover against classical solvers occurs at 179 variables and about 15 hours of runtime, on an FTQC machine with 73.91 million physical qubits each at a physical error rate (PER) of 10^{-3} [1772].

- **Microsoft** launched its Azure Quantum Resource Estimator in November 2022 to estimate the qubit count, T-gate count and execution time for a given algorithm targeting an FTQC platform [940]. It has been open-source since January 2024 [1775]. It takes as input, on top of the algorithm circuit, various QPU architecture specifications: qubit fidelities, connectivity, and the used quantum error correction code. It is related to the rQOPS metric, or reliable Quantum Operations Per Second which measures how many reliable operations can be executed in a second [1776]. The Microsoft resource estimator was used by Alice&Bob, showcasing the need for FTQC architectures to run commonplace algorithms [1777] and for some estimations for the implementation of the HHL algorithms [1778]. Other more specialized resource estimators exist like for chemistry applications [1779]. In a 2024 preprint, researchers in Poland used Microsoft Azure Resource estimator to compare the cost of running a quantum kernel on a QPU and in a classical emulator [1780]. They found that the quantum version could outperform the classical emulator with 37 qubits but at a threshold of 10^{10} seconds = 316 years. A quantum energy advantage would come with 32 qubits, at 11.57 days but with 28 GWh, which is a lot of energy!
- **pyLIQTR** (Lincoln Laboratory Quantum algorithm Test and Research) is a library to build quantum circuits developed by the MIT that can help in making algorithms resource estimates.
- **QUALTRAN** (quantum algorithms translator) is a resource estimator launched by Google in 2023. It helps express quantum algorithms, analyze them to output resource counts, do numerical simulations like for error propagation, produce documentation, use libraries of subroutines and estimate the cost of quantum error correction [1781].
- **BenchQ** is a resource estimator that Zapata AI developed for DARPA which considers qubit connectivity. It was completed in June 2024 by three physics and chemical simulation case studies for benchmarking published as part of a DARPA project [446, 1782]. Zapata went out of business in 2024, to be recreated in 2025.
- **Bartiq** is a resource estimator launched by PsiQuantum in May 2024, in connection with their QREF (quantum resource estimator format) [1783] and their Active Volume architecture [1784, 1785].
- **Rigetti** et al. proposed a resource estimator for FTQC algorithms in June 2024. It estimates the physical resources needed for the execution of some quantum algorithms (QFT, transverse-Ising models and 2D Fermi-Hubbard models), including the constraints of a modular superconducting hardware architecture. It predicts the size, power consumption, and runtime of these algorithms, in a range reaching a couple hundred logical qubits and between one and two million physical qubits [1786]. The tool seems specific to a specific graph state and distributed MBQC architecture which is not commonplace with superconducting qubit vendors. Rigetti published another paper with FTQC resource estimates using a Fermi-Hubbard model to simulate strongly correlated systems, with providing interesting estimates

of classical resources consumption by the DoE (in the 100Ks physical qubits) [1787].

- **AutoQuREO** is a resource estimator introduced in August 2026 by Fujitsu India built around an user-defined abstraction of the quantum computing stack, a modular library of reusable stack components enabling rapid full-stack prototyping, surrogate modeling of layer-wise resources via algorithmic profiling and neuro-symbolic learning, and integrated multi-constraints optimization that embeds QRE directly into deployment pipeline. It provides an example based on a FTQC QPE and a NISQ QAOA [1788].
- **Xanadu PennyLane** also contains a resource estimator which was introduced in October 2025 [1789].

Resource estimators are useful for optimizing both algorithms and QPU architectures. Some workloads require many statistically independent repetitions and can therefore be parallelized across multiple QPUs, while others are dominated by a single long fault-tolerant computation. Parallelization should therefore be estimated from the algorithm’s required circuit shot count, together with the cost, footprint and interconnect constraints of the target systems [1791].

Figure 112 compiles a set of algorithmic “logical” resource estimates for key documented quantum algorithms and their resource estimates expressed in quantum gates (all, T, Toffoli), related target qubit error rates ranges, and physical or logical qubit ranges. It shows that many useful industry cases are way above the 100-128 logical qubits that are in the mid-term FTQC QPU vendor roadmaps. It still shows up with serious papers providing resource estimates like a 2025 paper from JPMorgan Chase proposing an FTQC-based quantum algorithm for tensor problems, including tensor PCA and planted kXOR, that requires 900 logical qubits, $\sim 10^{15}$ gates and $\sim 10^{12}$ gate depth to bring some quantum advantage [1799].

In September 2025, NERSC published a survey associating resource estimates for scientific workloads with public vendor roadmaps [1790]. NERSC, the National Energy Research Scientific Computing Center, is a US Department of Energy Office of Science user facility operated as a division of Lawrence Berkeley National Laboratory. The survey compares high-energy physics, chemistry and materials workloads (Figure 111). Its comparison with vendor roadmaps is a projection rather than a guarantee, so the suggested 5-to-10-year application window depends on roadmaps being achieved and on the resource-estimation assumptions remaining valid. The report also discusses system-level performance metrics such as sustained throughput across representative workloads.

They also highlight that “*Most quantum algorithms and applications allow a trade-off between a reduction in the number of qubits for an increase in the number of quantum gates (or vice versa). A well-known example is Hamiltonian simulation, a key quantum-algorithmic kernel used in the majority of applications we discuss. Implementing a Hamiltonian simulation using a Trotter decomposition typically requires significantly fewer qubits and significantly more quantum gates compared to a method that combines qubitization with Quantum Signal Processing (QSP) at a similar error level.*”

Finally, a 2025 study on quantum-assisted resource estimation [1800] proposed using smaller quantum processors to estimate simulation errors, including Trotter errors, more tightly than analytical worst-case bounds. For the examples studied, the resulting error estimates reduced projected depth and runtime requirements by about three orders of magnitude for 100-qubit target systems. This is evidence for a potential quantum advantage in a specific resource-estimation subtask, not a general result that resource estimation itself provides quantum advantage.

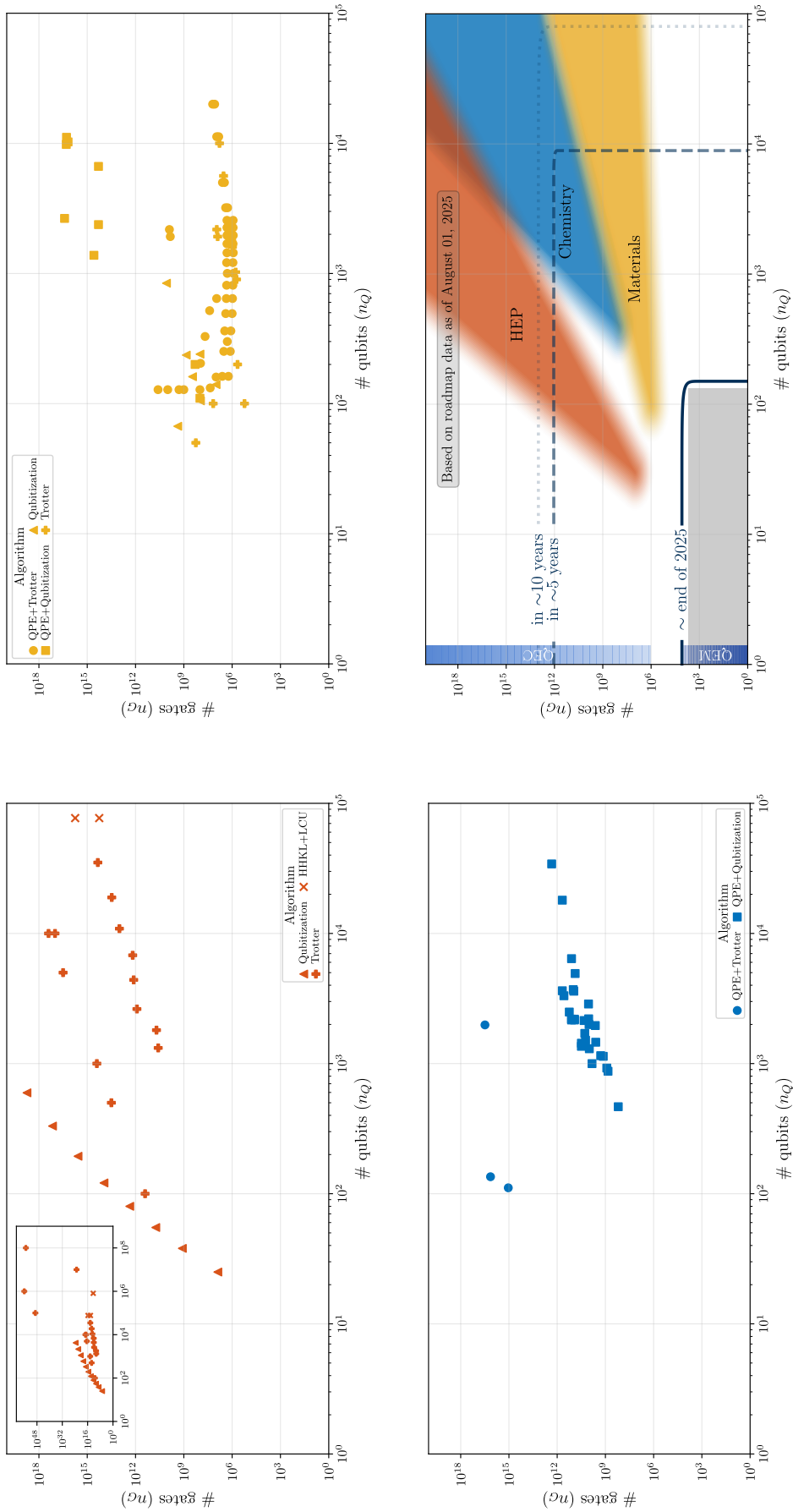


Figure 111: Charts from the DoE NERSC published in September 2025 showing resource estimates for high-energy physics simulations, chemical simulations and materials simulations in logical qubit number (X) and number of gate operations (Y). It then compares it to published QPU vendor roadmaps. It highlights that viable use cases in these three categories could be implemented in a five-to-ten-year window, provided that the vendors respect their roadmaps. Source: [1790].

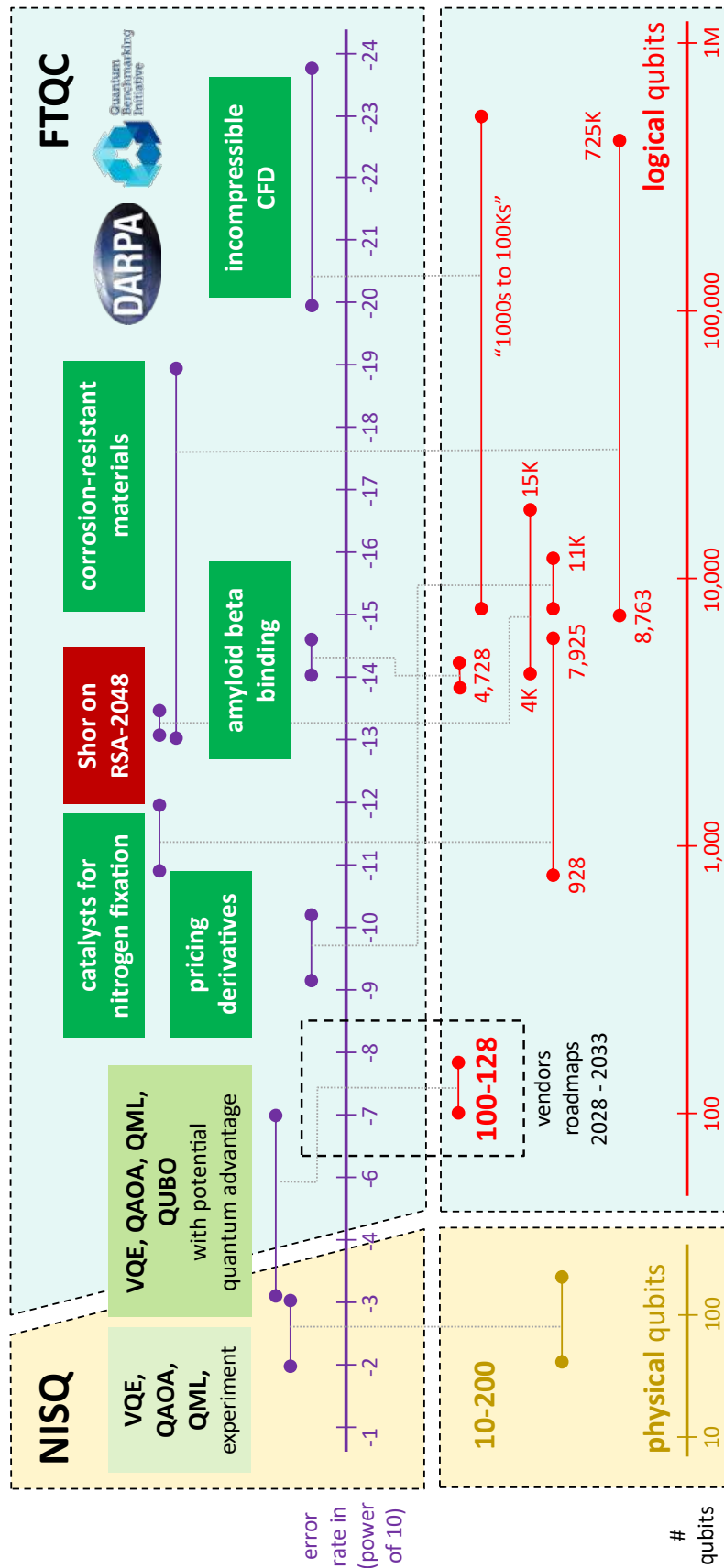
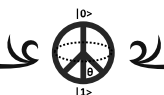


Figure 112: Compilation of some algorithmic-level resource estimates for key algorithms which have some industry relevance with data from 2024. The top part contains some typical algorithms, and their resource estimated ranges in gates, T gates or Toffoli gates. These resource estimates correspond to an error rate for the qubits (physical in NISQ and logical in FTQC) in the purple scale crossed by the dotted lines. In a simplified manner, it corresponds usually to the two-qubit gate error rates which are usually the higher vs single qubit gates. Below in orange and red are the related resource estimates in physical (NISQ) and logical (FTQC) qubits, regardless of the implementation. Physical qubits (in NISQ) and logical qubits (in FTQC) are on a similar log scale. What determines the physical per logical qubit count is not shown in the chart. The best algorithms for early FTQC systems may be NISQ variational algorithms which require deeper circuits than NISQ QPU + QEM can enable. Case study sources: TDA [1792], catalysts discovery [1782], derivative pricing [1793], integer factoring [1794], amyloid beta binding [1795], simulating corrosion resistant materials [1796], catalysts discovery [1797], and computational fluid dynamics [1798]. © Olivier Ezratty, 2024-2025.



2.1.9 Bibliography and notes

These are the bibliographical references and notes for the **Development tools classes** subsection.

- [67] [Quantum compilation framework for data loading](#), by Guillermo Alonso-Linaje, Utkarsh Azad, Jay Soni, Jarrett Smalley, Leigh Lapworth, and Juan Miguel Arrazola, arXiv, December 2025 (21 pages).
- [68] [Sparse Quantum State Preparation for Strongly Correlated Systems](#), by César Feniou, O. Adjoua, B. Claudon, Julien Zylberman, E. Giner, and Jean-Philip Piquemal, The Journal of Physical Chemistry Letters, March 2024 (27 pages)  (preprint on [HAL](#)).
- [158] [Quantum algorithm for linear systems of equations](#), by Aram W. Harrow, Avinandan Hassidim, and Seth Lloyd, Physical Review Letters, October 2009 (4 pages).
- [446] [Feasibility of accelerating incompressible computational fluid dynamics simulations with fault-tolerant quantum computers](#), by John Penuel, Amara Katabarwa, Peter D. Johnson, Collin Farquhar, Yudong Cao, and Michael C. Garrett, arXiv, September 2024 (70 pages).
- [939] [Simulating the quantum Fourier transform, Grover's algorithm, and the quantum counting algorithm with limited entanglement using tensor-networks](#), by Marcel Niedermeier, Jose L. Lado, and Christian Flindt, arXiv, September 2024 (16 pages).
- [940] [Assessing requirements to scale to practical quantum advantage](#), by Michael E. Beverland, Alexander Vashchilo, et al., arXiv, November 2022 (41 pages).
- [967] [Large-Scale Simulation of Quantum Computational Chemistry on a New Sunway Supercomputer](#), by Honghui Shang, Zhenyu Li, et al., arXiv, July 2022 (13 pages).
- [1501] [Pauli Propagation: A Computational Framework for Simulating Quantum Systems](#), by Manuel S. Rudolph, Tyson Jones, Yanting Teng, Armando Angrisani, and Zoë Holmes, arXiv, May 2025 (32 pages).
- [1520] [Quantum Computing Toolkit From Nuts and Bolts to Sack of Tools](#), by Himanshu Sahu and Hari Prabhat Gupta, arXiv, March 2023 (27 pages).
- [1521] [Quantivine: A Visualization Approach for Large-scale Quantum Circuit Representation and Analysis](#), by Zhen Wen, Wei Chen, et al., arXiv, July 2023 (11 pages).
- [1522] [Visualizing Quantum Circuits: State Vector Difference Highlighting and the Half-Matrix](#), by Michael J. McGuffin and Jean-Marc Robert, arXiv, October 2025 (11 pages).
- [1523] [Interacting Quantum Observables: Categorical Algebra and Diagrammatics](#), by Bob Coecke and Ross Duncan, arXiv, April 2011 (81 pages).
- [1524] [Equivalence Classes of Quantum Error-Correcting Codes](#), by Andrey Boris Khesin and Alexander Li, arXiv, June 2024 (28 pages).
- [1525] [The ZX-calculus as a Language for Topological Quantum Computation](#), by Fatimah Rita Ahmadi and Aleks Kissinger, arXiv, August 2023 (14 pages).
- [1526] [Effective Compression of Quantum Braided Circuits Aided by ZX-Calculus](#), by Michael Hanks, Physical Review X, November 2020.
- [1527] [A Review on Quantum Circuit Optimization using ZX-Calculus](#), by Tobias Fischbach, Pierre Talbot, and Pascal Bourvry, arXiv, September 2025 (26 pages).
- [1528] [Automated quantum circuit optimization with randomized replacements](#), by Marcin Szyniszewski, Aleks Kissinger, Noah Linden, and Paul Skrzypczyk, arXiv, January 2026 (7 pages).
- [1529] [Completeness of the ZX-Calculus](#), by Renaud Vilmart et al., 2018 (123 slides).
- [1530] [Completeness of the ZX-Calculus](#), by Emmanuel Jeandel, Simon Perdrix, and Renaud Vilmart, arXiv, June 2020 (72 pages).
- [1531] [SZX-calculus: Scalable Graphical Quantum Reasoning](#), by Titouan Carette, Dominic Horsman, and Simon Perdrix, arXiv, April 2019 (29 pages).
- [1532] [AKLT-states as ZX-diagrams: diagrammatic reasoning for quantum states](#), by Richard D. P. East, John van de Wetering, Nicholas Chancellor, and Adolfo G. Grushin, arXiv, December 2021 (29 pages).
- [1533] [Quantomatic: A Proof Assistant for Diagrammatic Reasoning](#), by Aleks Kissinger and Vladimir Zamdzhiev, arXiv, October 2015 (11 pages).
- [1534] [DisCoPy for the quantum computer scientist](#), by Alexis Toumi, Giovanni de Felice, and Richie Yeung, arXiv, May 2022 (6 pages).
- [1535] [Towards Machine Translation with Quantum Computers](#), by Irene Vicente Nieto, 2021 (48 pages).
- [1536] [Qumin, a minimalist quantum programming language](#), by Alexander Singh, Konstantinos Giannakis, and Theodore Andronikos, arXiv, April 2017 (34 pages).
- [1537] [SimuQ: A Framework for Programming Quantum Hamiltonian Simulation with Analog Compilation](#), by Yuxiang Peng, Jacob Young, Pengyu Liu, and Xiaodi Wu, arXiv, November 2023 (34 pages).
- [1538] [A Survey of Quantum Programming Languages](#), by Quan Do, Hersh Gupta, Xiyuan Cao, Aarav Pabla, Pranav Singamsetty, Evan O'Grady, Keli Huang, and Jens Palsberg, arXiv, June 2026 (27 pages).
- [1539] [Quantum Circuit Synthesis and Compilation Optimization: Overview and Prospects](#), by Yan Ge, Yan Junchi, et al., arXiv, June 2024 (32 pages).
- [1540] [Qsyn: A Developer-Friendly Quantum Circuit Synthesis Framework for NISQ Era and Beyond](#), by Mu-Te Lau, Chung-Yang Ric Huang, et al., arXiv, November 2024 (12 pages).
- [1541] [Recursive Cartan decompositions for unitary synthesis](#), by David Wierichs, Maxwell West, Roy T. Forestano, M. Cerezo, and Nathan Killoran, arXiv, June 2025 (33 pages).
- [1542] [RedCarD: A Quantum Assisted Algorithm for Fixed-Depth Unitary Synthesis via Cartan Decomposition](#), by Omar Alsheikh, A. F. Kemper, et al., arXiv, December 2025 (6 pages).
- [1543] [On Exact Sizes of Minimal CNOT Circuits](#), by Jens Emil Christensen, Søren Fuglede Jørgensen, Andreas Pavlogianis, and Jaco van de Pol, arXiv, April 2025 (34 pages).
- [1544] [Heuristic and Optimal Synthesis of CNOT and Clifford Circuits](#), by Mark Webster, Stergios Koutsoumpas, and Dan E Browne, arXiv, March 2025 (26 pages).
- [1545] [Exact two-qubit universal quantum circuit](#), by Jun Zhang, Jiri Vala, Shankar Sastry, and K. Birgitta Whaley, arXiv, May 2003 (5 pages).
- [1546] [Minimal Universal Two-qubit Quantum Circuits](#), by Vivek V. Shende, Igor L. Markov, and Stephen S. Bullock, arXiv, March 2004 (8 pages).
- [1547] [Quantum Circuits for General Multiqubit Gates](#), by Mikko Möttönen, Juha J. Vartiainen, Ville Bergholm, and Martti M. Salomaa, Physical Review Letters, September 2004 (4 pages)  (preprint on [arXiv](#)).
- [1548] [Near-optimal quantum circuit construction via Cartan decomposition](#), by Maximilian Balthasar Mansky, PRA, November 2023 (9 pages).

- [1549] [Optimizing Gate Decomposition for High-Level Quantum Programming](#), by Evandro C. R. Rosa, Eduardo I. Duzzioni, and Rafael de Santiago, arXiv, March 2025 (21 pages).
- [1550] [Unitary Synthesis of Clifford+T Circuits with Reinforcement Learning](#), by Sebastian Rietsch, Daniel D. Scherer, et al., arXiv, September 2024 (12 pages).
- [1551] [Practical and efficient quantum circuit synthesis and transpiling with Reinforcement Learning](#), by David Kremer, Victor Villar, Hanhee Paik, Ivan Duran, Ismael Faro, and Juan Cruz-Benito, arXiv, February 2025 (19 pages).
- [1552] [Unitary Synthesis with AlphaZero via Dynamic Circuits](#), by Xavier Valcarce, Bastien Grivet, and Nicolas Sangouard, arXiv, August 2025 (5 pages).
- [1553] [Pauli Network Circuit Synthesis with Reinforcement Learning](#), by Ayushi Dubal, David Kremer, Simon Martiel, Victor Villar, Derek Wang, and Juan Cruz-Benito, arXiv, March 2025 (13 pages).
- [1554] [Optimizing the non-Clifford-count in unitary synthesis using Reinforcement Learning](#), by David Kremer, Ali Javadi-Abhari, and Priyanka Mukhopadhyay, arXiv, September 2025 (32 pages).
- [1555] [Quantum Circuit Pre-Synthesis: Learning Local Edits to Reduce T-count](#), by Daniele Lizzio Bosco, Lukasz Cincio, Giuseppe Serra, and M. Cerezo, arXiv, January 2026 (5 pages).
- [1556] [T Count as a Numerically Solvable Minimization Problem](#), by Marc Grau Davis, Ed Younis, Mathias Weiden, Hyeonrak Choi, and Dirk Englund, arXiv, May 2026 (6 pages).
- [1557] [Multi-Qubit Dyadic Phase Fixing for Fault-Tolerant Quantum Compilation](#), by Justin Kalloor, Mathias Weiden, Ed Younis, John Kubitowicz, and Costin Iancu, arXiv, June 2026 (11 pages).
- [1558] [Quantum Circuit for General Unitary: Improved T-count via Block Flattening and Dilation](#), by Pei Yuan, Shengyu Zhang, and Wei Zi, arXiv, August 2026 (13 pages).
- [1559] [When Close Enough Is Not Enough: Autoregressive Drift in Quantum Circuit Synthesis](#), by Mehdi Saeedi, Eddie Richter, and Paul Hartke, arXiv, July 2026 (11 pages).
- [1560] [Gatekeepers and Hallucinations: A Layered Evaluation Framework for LLM-Driven Quantum Circuit Generation](#), by Christopher Coleman and Sharon Marfatia, arXiv, June 2026 (7 pages).
- [1561] [RubriQ: Rubric-Guided Group Relative Policy Optimization for Constraint-Aware Quantum Circuit Synthesis](#), by Ziqing Guo and Ziwen Pan, arXiv, July 2026 (17 pages).
- [1562] [Witnessing the architecture of quantum circuits](#), by Raphaël Mothe and Otfried Gühne, arXiv, August 2026 (7 pages).
- [1563] [Improving the Fidelity of CNOT Circuits on NISQ Hardware](#), by Dohun Kim, Minyoung Kim, Sarah Meng Li, and Michele Mosca, arXiv, May 2024 (67 pages).
- [1564] [SWAP-less Implementation of Quantum Algorithms](#), by Berend Klaver, Stefan Rombouts, Michael Fellner, Anette Messinger, Kilian Ender, Katharina Ludwig, and Wolfgang Lechner, arXiv, August 2024 (10 pages).
- [1565] [Polylogarithmic-depth controlled-NOT gates without ancilla qubits](#), by Baptiste Claudon, Fabrice Debbaesch, Alberto Peruzzo, Jean-Philip Piquemal, et al., Nature Communications, July 2024.
- [1566] [Quantum circuit for multi-qubit Toffoli gate with optimal resource](#), by Junhong Nie, Wei Zi, and Xiaoming Sun, arXiv, February 2024 (14 pages).
- [1567] [Rise of conditionally clean ancillae for efficient quantum circuit constructions](#), by Tanuj Khattar and Craig Gidney, arXiv, May 2025 (23 pages).
- [1568] [Cost-Effective Realization of n-Bit Toffoli Gates for IBM Quantum Computers Using the Bloch Sphere Approach and IBM Native Gates](#), by Ali Al-Bayaty and Marek Perkowski, arXiv, October 2024 (32 pages).
- [1569] [Multi-qubit Toffoli with exponentially fewer T gates](#), by David Gosset, Robin Kothari, and Chenyi Zhang, arXiv, October 2025 (26 pages).
- [1570] [Adaptive Clifford+T Decomposition of Large Toffoli Gates with One Clean Ancilla](#), by Abhoy Kole, Majd Assaad, Till Schnittka, and Rolf Drechsler, arXiv, May 2026 (6 pages).
- [1571] [More efficient Clifford+T synthesis for small-angle rotations and application to Trotterization](#), by Marius Bothe, Christoph Sünderhauf, Michael J. Witham, Earl T. Campbell, and Nick S. Blunt, arXiv, June 2026 (43 pages).
- [1572] [A Reservoir Computing Approach to Quantum Gate Synthesis](#), by Francesco Caravelli, Roberto Menta, and Antonio Sannia, arXiv, July 2026 (12 pages).
- [1573] [Optimal Approximation of Single Qubit Rotations within a Quantum Circuit](#), by Gilad Kishony, Avi Elazari, Ron Cohen, and Lior Gazit, arXiv, May 2026 (10 pages).
- [1574] [Decomposition Algorithm of an Arbitrary Pauli Exponential through a Quantum Circuit](#), by Maximilian Balthasar Mansky, Victor Ramos Puigvert, Santiago Londoño Castillo, and Claudia Linnhoff-Popien, arXiv, May 2023 (9 pages).
- [1575] [Shortest Path in Pauli Forest – An Algorithm for Decomposing Pauli Exponentials to Quantum Circuits](#), by Lauri Vuorenkoski and Arianne Meijer-van de Griend, arXiv, May 2026 (9 pages).
- [1576] [Matchgate synthesis via Clifford matchgates and T gates](#), by Berta Casas, Paolo Braccia, Élie Gouzien, M. Cerezo, and Diego García-Martín, arXiv, February 2026 (19 pages).
- [1577] [Asymptotically Optimal Circuit Depth for Diagonal Unitary Synthesis and Compilation on Two-Dimensional Grids](#), by Chengzhuo Xu, Xiao Chen, Zhihao Liu, and Zhigang Li, arXiv, June 2026 (24 pages).
- [1578] [Synthesis and Optimization of Encoding Circuits for Fault-Tolerant Quantum Computation](#), by Tom Peham, Matthew Steinberg, Robert Wille, and Sascha Heußen, arXiv, May 2026 (33 pages).
- [1579] [QASMTans: A QASM based Quantum Transpiler Framework for NISQ Devices](#), by Fei Hua, Travis S. Humble, Ang Li, et al., arXiv, August 2023 (10 pages).
- [1580] [Comparative Study of Quantum Transpilers: Evaluating the Performance of qiskit-braket-provider, qBraid-SDK, and Pytket Extensions](#), by Mohamed Messaoud Louamri, Mohamed Taha Rouabah, et al., arXiv, April 2025 (11 pages).
- [1581] [SurgeQ: A Hybrid Framework for Ultra-Fast Quantum Processor Design and Crosstalk-Aware Circuit Execution](#), by Xinxuan Chen, Hongxiang Zhu, Zhaohui Yang, Zhaofeng Su, Jianxin Chen, Feng Wu, and Hui-Hai Zhao, arXiv, January 2026 (7 pages).
- [1582] [QASMTans: An End-to-End QASM Compilation Framework with Pulse Generation for Near-Term Quantum Devices](#), by Aaron Hoyt, Meng Wang, Fei Hua, Chunshu Wu, Chenxu Liu, Muqing Zheng, Samuel Stein, Drew Rebar, Yufei Ding, Travis S. Humble, and Ang Li, arXiv, February 2026 (27 pages).
- [1583] [Dynamical Mean Field Theory for Real Materials on a Quantum Computer](#), by Johannes Selisko, Ivano Tavernelli, Thomas Eckl, et al., arXiv, April 2024 (25 pages).
- [1584] [Transpiler Autotuning with Predictive Models for Quantum Circuit Optimization](#), by Piotr Malkowski, Domenik Eichhorn, Joshua Ammermann, Rinor Kelmendi, Nick Poser, Patrick Hopf, and Ina Schaefer, arXiv, July 2026 (30 pages).
- [1585] [How about quantum computing?](#), by Bert de Jong, June 2019 (47 slides).
- [1586] [Kernpiler: Compiler Optimization for Quantum Hamiltonian Simulation with Partial Trotterization](#), by Ethan Decker, Gushu Li, et al., arXiv, April 2025 (15 pages).
- [1587] [Constant-time hybrid compilation of Shor’s algorithm with quantum just-in-time compilation](#), by David Ittah, Jackson Fraser, Josh Izaac, and Olivia Di Matteo, arXiv, April 2025 (7 pages).

- [1588] [Partial Compilation of Variational Algorithms for Noisy Intermediate-Scale Quantum Machines](#), by Pranav Gokhale, Frederic T. Chong, et al., arXiv, September 2019 (13 pages).
- [1589] [Full-stack quantum computing systems in the NISQ era: algorithm-driven and hardware-aware compilation techniques](#), by Medina Bandic, Sebastian Feld, and Carmen G. Almudever, arXiv, April 2022 (6 pages).
- [1590] [Towards Application-Aware Quantum Circuit Compilation](#), by Nils Quetschlich, Robert Wille, et al., arXiv, June 2024 (8 pages).
- [1591] [TrapSIMD: SIMD-Aware Compiler Optimization for 2D Trapped-Ion Quantum Machines](#), by Jixuan Ruan, Yufei Ding, et al., arXiv, April 2025 (15 pages).
- [1592] [Quantum Compiler Design for Qubit Mapping and Routing: A Cross-Architectural Survey of Superconducting, Trapped-Ion, and Neutral Atom Systems](#), by Chenghong Zhu, Xin Wang, et al., arXiv, May 2025 (43 pages).
- [1593] [Computational Capabilities and Compiler Development for Neutral Atom Quantum Processors: Connecting Tool Developers and Hardware Experts](#), by Ludwig Schmid, Robert Wille, et al., arXiv, April 2024 (32 pages).
- [1594] [Parallax: A Compiler for Neutral Atom Quantum Computers under Hardware Constraints](#), by Jason Ludmir and Tirthak Patel, arXiv, October 2024 (13 pages).
- [1595] [General circuit mapping algorithm for neutral atom quantum computers](#), by Neven Gentil, Lous S. Rianne, and Aida Todri-Sanial, arXiv, June 2026 (24 pages).
- [1596] [Search Smarter, Not Harder: A Scalable, High-Quality Zoned Neutral Atom Compiler](#), by Yannick Stade, Lukas Burgholzer, and Robert Wille, arXiv, December 2025 (7 pages).
- [1597] [Architecture aware compilation of quantum circuits via lazy synthesis](#), by Simon Martiel and Timothée Goubault de Brugière, arXiv, June 2022 (32 pages).
- [1598] [Not All SWAPs Have the Same Cost: A Case for Optimization-Aware Qubit Routing](#), by Ji Liu, Peiyi Li, and Huiyang Zhou, arXiv, May 2022 (17 pages).
- [1599] [Robust Qubit Mapping Algorithm via Double-Source Optimal Routing on Large Quantum Circuits](#), by Chin-Yi Cheng, Chung-Yang Ric Huang, et al., arXiv, August 2024 (26 pages).
- [1600] [MIRAGE: Quantum Circuit Decomposition and Routing Collaborative Design using Mirror Gates](#), by Evan McKinney, Michael Hatridge, and Alex K. Jones, arXiv, August 2023 (13 pages).
- [1601] [Decoding techniques applied to the compilation of CNOT circuits for NISQ architectures](#), by Timothée Goubault de Brugière, Marc Baboulin, Benoît Valiron, Simon Martiel, and Cyril Allouche, arXiv, January 2022 (31 pages).
- [1602] [Energy risk analysis with Dynamic Amplitude Estimation and Piecewise Approximate Quantum Compiling](#), by Kumar J. B. Ghosh, Corey O'Meara, et al., arXiv, June 2023 (25 pages).
- [1603] [Shallower CNOT circuits on realistic quantum hardware](#), by Timothée Goubault de Brugière and Simon Martiel, arXiv, March 2023 (28 pages).
- [1604] [Algorithmic Theory of Qubit Routing](#), by Takehiro Ito, Naonori Kakimura, Naoyuki Kamiyama, Yusuke Kobayashi, and Yoshio Okamoto, arXiv, May 2023 (14 pages).
- [1605] [Treespilation: Architecture- and State-Optimised Fermion-to-Qubit Mappings](#), by Aaron Miller, Adam Glos, and Zoltán Zimborás, arXiv, July 2024 (25 pages).
- [1606] [Connectivity-aware Synthesis of Quantum Algorithms](#), by Florian Dreier, Christoph Fleckenstein, Gregor Aigner, Michael Fellner, Reinhard Stahn, Martin Lanthaler, and Wolfgang Lechner, arXiv, January 2025 (38 pages).
- [1607] [CNOT-Optimal Clifford Synthesis as SAT](#), by Irfansha Shaik and Jaco van de Pol, arXiv, April 2025 (27 pages).
- [1608] [Optimizing QAOA circuit transpilation with parity twine and SWAP network encodings](#), by J. A. Montanez-Barrera, Yanjun Ji, Michael R. von Spakovsky, David E. Bernal Neira, and Kristel Michielsen, arXiv, May 2025 (10 pages).
- [1609] [QContext: Context-Aware Decomposition for Quantum Gates](#), by Ji Liu, Max Bowman, Pranav Gokhale, Sidharth Dangwal, Jeffrey Larson, Frederic T. Chong, and Paul D. Hovland, arXiv, February 2023 (10 pages).
- [1610] [Optimal Layout Synthesis for Deep Quantum Circuits on NISQ Processors with 100+ Qubits](#), by Irfansha Shaik and Jaco van de Pol, arXiv, July 2024 (18 pages).
- [1611] [Non-Clifford Fusion: T-Gate Optimization for Quantum Simulation](#), by Yingheng Li, Xulong Tang, Paul Hovland, and Ji Liu, arXiv, October 2025 (13 pages).
- [1612] [Quantum Circuit Optimization with AlphaTensor](#), by Francisco J. R. Ruiz, Pushmeet Kohli, et al., arXiv, March 2024 (25 pages).
- [1613] [Reusability Report: Optimizing T-count in General Quantum Circuits with AlphaTensor-Quantum](#), by Remmy Zen, Maximilian Nägele, and Florian Marquardt, arXiv, November 2025 (7 pages).
- [1614] [Gaussian Elimination versus Greedy Methods for the Synthesis of Linear Reversible Circuits](#), by Timothée Goubault de Brugière, Marc Baboulin, Benoît Valiron, Simon Martiel, and Cyril Allouche, arXiv, January 2022 (27 pages).
- [1615] [Reducing the Depth of Linear Reversible Quantum Circuits](#), by Timothée Goubault de Brugière, Marc Baboulin, Benoît Valiron, Simon Martiel, and Cyril Allouche, arXiv, January 2022 (22 pages).
- [1616] [T-count optimization of approximate quantum Fourier transform](#), by Byeongyong Park and Doyeol Ahn, arXiv, July 2024 (21 pages).
- [1617] [Optimal Hadamard gate count for Clifford+T synthesis of Pauli rotations sequences](#), by Vivien Vandaele, Simon Martiel, Simon Perdrix, and Christophe Vuillot, arXiv, February 2024 (32 pages).
- [1618] [Automated Auxiliary Qubit Allocation in High-Level Quantum Programming](#), by Evandro C. R. Rosa, Jerusa Marchi, Eduardo I. Duzzioni, and Rafael de Santiago, arXiv, December 2024 (24 pages).
- [1619] [Scalable Memory Recycling for Large Quantum Programs](#), by Israel Reichental, Ravid Alon, Lior Preminger, Matan Vax, and Amir Naveh, arXiv, March 2025 (10 pages).
- [1620] [Recycling qubits in near-term quantum computers](#), by Galit Anikeeva, Isaac H. Kim, and Patrick Hayden, arXiv, December 2020 (4 pages).
- [1621] [Qurzon: A Prototype for a Divide and Conquer Based Quantum Compiler](#), by Turbasu Chatterjee, Arnav Das, Shah Ishmam Mohtashim, Amit Saha, and Amlan Chakrabarti, arXiv, November 2021 (11 pages).
- [1622] [Reducing the Compilation Time of Quantum Circuits Using Pre-Compilation on the Gate Level](#), by Nils Quetschlich, Lukas Burgholzer, and Robert Wille, arXiv, July 2023 (11 pages).
- [1623] [Accelerating Quantum Algorithms with Precomputation](#), by William J. Huggins and Jarrod R. McClean, arXiv, February 2024 (25 pages).
- [1624] [Efficient Quantum Circuit Compilation for Near-Term Quantum Advantage](#), by Yuchen Guo and Shuo Yang, arXiv, June 2025 (13 pages).
- [1625] [CAMEL: Physically Inspired Crosstalk-Aware Mapping and gate scheduling for Frequency-Tunable Quantum Chips](#), by Bin-han Lu, Guo-ping Guo, et al., arXiv, October 2024 (14 pages).
- [1626] [QuantEM: The quantum error management compiler](#), by Ji Liu, Quinn Langfitt, Mingyoung Jessica Jeng, Alvin Gonzales, Noble Agyeman-Bobie, Kaiya Jones Siddharth Vijaymurugan, Daniel Dille, Zain H. Saleem, Nikos Haravallas, and Kaitlin N. Smith, arXiv, September 2025 (5 pages).

- [1627] [Policy Gradient Approach to Compilation of Variational Quantum Circuits](#), by David A. Herrera-Martí, arXiv, September 2022 (17 pages).
- [1628] [Coqa: Blazing Fast Compiler Optimizations for QAOA](#), by Yuchen Zhu, Zhiding Liang, et al., arXiv, August 2024 (7 pages).
- [1629] [Resilience-Runtime Tradeoff Relations for Quantum Algorithms](#), by Luis Pedro García-Pintos, Yi-Kai Liu, et al., arXiv, August 2024 (21 pages).
- [1630] [Optimizing Quantum Fourier Transformation \(QFT\) Kernels for Modern NISQ and FT Architectures](#), by Yuwei Jin, Eddy Z. Zhang, et al., arXiv, August 2024 (16 pages).
- [1631] [Application Scale Quantum Circuit Compilation and Optimization](#), by Justin Kalloor, Mohan Sarovar, et al., arXiv, October 2025 (17 pages).
- [1632] [Leveraging Quantum Machine Learning Generalization to Significantly Speed-up Quantum Compilation](#), by Alon Kukliansky, Lukasz Cincio, Ed Younis, and Costin Iancu, arXiv, August 2024 (15 pages).
- [1633] [Reinforcement Learning Based Quantum Circuit Optimization via ZX-Calculus](#), by Jordi Riu, Jan Nogué, Gerard Vilaplana, Artur Garcia-Saez, and Marta P. Estarellas, arXiv, May 2025 (22 pages).
- [1634] [Quantum circuit optimization with deep reinforcement learning](#), by Thomas Fösel, Murphy Yuezhen Niu, Florian Marquardt, and Li Li, arXiv, March 2021 (10 pages).
- [1635] [Quantum Compiling with Reinforcement Learning on a Superconducting Processor](#), by Z. T. Wang, S. P. Zhao, et al., arXiv, June 2024 (26 pages).
- [1636] [Toward Scalable Quantum Compilation for Modular Architecture: Qubit Mapping and Reuse via Deep Reinforcement Learning](#), by Sokea Sang, Leanhok Hour, and Youngsun Han, arXiv, June 2025 (37 pages).
- [1637] [Quantum Circuit Discovery for Fault-Tolerant Logical State Preparation with Reinforcement Learning](#), by Remmy Zen, Markus Müller, Florian Marquardt, et al., arXiv, May 2024 (34 pages).
- [1638] [Revisiting Noise-adaptive Transpilation in Quantum Computing: How Much Impact Does it Have?](#), by Yuqian Huo, Jinbiao Wei, Christopher Kverne, Mayur Akewar, Janki Bhimani, and Tirthak Patel, arXiv, July 2025 (10 pages).
- [1639] [Quantum circuit compilation with quantum computers](#), by Davide Rattacaso, Daniel Jaschke, Marco Ballarin, Ilaria Siloi, and Simone Montangero, arXiv, August 2024 (5 pages).
- [1640] [Investigating Retargetability Claims for Quantum Compilers](#), by Luke Southall, Joshua Ammermann, Rinor Kelmendi, Domenik Eichhorn, and Ina Schaefer, arXiv, January 2026 (14 pages).
- [1641] [A Platform-aware Compilation Framework for Fault-tolerant Quantum Computation](#), by Srushti Patil, Susan X. Chen, Andreas Juul Bay-Smidt, Stefan Alaric Schäffer, Peter Krogstrup, Stefano Paesani, and Gemma C. Solomon, arXiv, September 2026 (31 pages).
- [1642] [Harvest: Resource-Aware Quantum Compilation for Magic State Protocols](#), by Jannik Pflieger, Aleksandra Świerkowska, Emmanouil Giortamis, and Pramod Bhatotia, arXiv, August 2026 (14 pages).
- [1643] [Stalls and Speculation: Pipelined Execution for Fault Tolerant Quantum Computation](#), by Aditi Awasthi, Gokul Subramanian Ravi, and Jonathan Mark Baker, arXiv, June 2026 (7 pages).
- [1644] [Chipmunk: A Fault-Tolerant Compiler for Chiplet Quantum Architectures](#), by Peter Wegmann, Aleksandra Świerkowska, Emmanouil Giortamis, and Pramod Bhatotia, arXiv, March 2026 (14 pages).
- [1645] [Compilation Pipeline for Predicting Algorithmic Break-Even in an Early-Fault-Tolerant Surface Code Architecture](#), by Tianyi Hao, Joseph Sullivan, Sivaprasad Omanakuttan, Michael A. Perlin, and Ruslan Shaydulin, arXiv, November 2025 (3 pages).
- [1646] [Low-cost algorithm-to-execution framework for surface-code quantum computing](#), by Yunxin Tang, Zixuan Huo, Junxiang Huang, Zhou You, Zhirao Wang, Zhenai Ding, Yumeng Zeng, Yangyu Lu, Yiming Huang, Zongkang Zhang, Haipeng Xie, Ying Li, Jinzhao Sun, Xiao Yuan, and Yuan Yao, arXiv, September 2026 (31 pages).
- [1647] [Quantum Compiler Design for Fault-Tolerant Quantum Computing](#), by Chenghong Zhu, Jiahao Chen, Keming He, Hongshun Yao, Zhaohui Yang, Jin-Guo Liu, Anbang Wu, Xiaotong Ni, Xingsheng Luan, Zhuo Fu, Shenggen Zheng, and Xin Wang, arXiv, September 2026 (60 pages).
- [1648] [CUDA-Q Logical: Retargetable Compilation for Fault-Tolerant Quantum Computing](#), by Alexander McCaskey, Justin Lietz, Adam Holmes, Kohei Nakaji, Vadym Kliuchnikov, Yifan Hong, Amalee Wilson, Andres Paz, Bettina Heim, Bruno Schmitt, and Krysta M. Svore, arXiv, September 2026 (67 pages).
- [1649] [Opportunities and Challenges in Intermediate-Scale Quantum Computing](#), by Fred Chong, 2018 (34 slides).
- [1650] [Breaking Down Quantum Compilation: Profiling and Identifying Costly Passes](#), by Felix Zilk, Alessandro Tundo, Vincenzo De Maio, and Ivona Brandic, arXiv, May 2025 (6 pages).
- [1651] [Optimizing compilation of error correction codes for 2xN quantum dot arrays and its NP-hardness](#), by Anthony Micciche, Anasua Chatterjee, Andrew McGregor, and Stefan Krastanov, arXiv, January 2025 (21 pages).
- [1652] [Deterministic Fault-Tolerant State Preparation for Near-Term Quantum Error Correction: Automatic Synthesis Using Boolean Satisfiability](#), by Ludwig Schmid, Tom Peham, Lucas Berent, Markus Müller, and Robert Wille, arXiv, January 2025 (7 pages).
- [1653] [OPAQUE: Obfuscating Phase in Quantum Circuit Compilation for Efficient IP Protection](#), by Anees Rehman, Vincent Langford, and Yuntao Liu, arXiv, February 2025 (6 pages).
- [1654] [DeQompile: quantum circuit decompilation using genetic programming for explainable quantum architecture search](#), by Shubing Xie, Aritra Sarkar, and Sebastian Feld, arXiv, April 2025 (19 pages).
- [1655] [Q²Chemistry: A quantum computation platform for quantum chemistry](#), by Yi Fan, Jie Liu, Xiongzi Zeng, Zhiqian Xu, Honghui Shang, Zhenyu Li, and Jinlong Yang, arXiv, August 2022 (32 pages).
- [1656] [List of QC simulators](#), by Quantiki, Quantiki website, 2007.
- [1657] [Disentangling quantum emulation and quantum simulation](#), by Olivier Ezratty, January 2023.
- [1658] [Universal Quantum Emulator](#), by Iman Marvian and Seth Lloyd, arXiv, March 2024 (12 pages).
- [1659] [Simulation of Quantum Computers: Review and Acceleration Opportunities](#), by Alessio Cicero, Mohammad Ali Maleki, Muhammad Waqar Azhar, Anton Frisk Kockum, and Pedro Trancoso, arXiv, March 2025 (33 pages).
- [1660] [Bell's Inequality Violation on MATLAB](#), by Somi Yun, Fakhar Zaman, et al., Kyung Hee University, 2020.
- [1661] [Large-Scale Quantum Circuit Simulation on an Exascale System for QPU Benchmarking](#), by J. A. Montanez-Barrera and Kristel Michielsen, arXiv, April 2026 (9 pages).
- [1662] [QArray+: A physics-informed GPU-accelerated simulator for quantum dot arrays](#), by Pranav Vaidhyanathan, Barnaby van Straaten, Alice Petrillo, Rahul Marchand, Edwin De Nicolo, Menno Veldhorst, Brucek Khailany, Taylor L. Patti, and Natalia Ares, arXiv, September 2026 (19 pages).
- [1663] [Classical Simulation of Intermediate-Size Quantum Circuits](#), by Jianxin Chen, Fang Zhang, Cupjin Huang, Michael Newman, and Yaoyun Shi, arXiv, May 2018 (12 pages).
- [1664] [What limits the simulation of quantum computers?](#), by Yiqing Zhou, E. Miles Stoudenmire, and Xavier Waintal, arXiv, March 2020 (14 pages).

- [1665] [Benchmarking Quantum Computer Simulation Software Packages: State Vector Simulators](#), by Amit Jamadagni, Andreas M. Läuchli, and Cornelius Hempel, arXiv, July 2024 (19 pages).
- [1666] [Classical simulation of Gaussian quantum circuits with non-Gaussian input states](#), by Ulysse Chabaud, Giulia Ferrini, Frédéric Grosshans, and Damian Markham, Physical Review Research, July 2021.
- [1667] [Enabling large-depth simulation of noisy quantum circuits with positive tensor networks](#), by Ambroise Müller, Thomas Ayrat, and Corentin Bertrand, arXiv, February 2024 (17 pages).
- [1668] [Distributed Memory Techniques for Classical Simulation of Quantum Circuits](#), by Ryan LaRose, arXiv, June 2018 (11 pages).
- [1669] [Distributed Tensor Network Library for Quantum Computing Emulation](#), by Jakub Adamski and Oliver Thomson Brown, arXiv, May 2025 (12 pages).
- [1670] [Harnessing CUDA-Q's MPS for Tensor Network Simulations of Large-Scale Quantum Circuits](#), by Gabin Schieffer, Stefano Markidis, and Ivy Peng, arXiv, January 2025 (10 pages).
- [1671] [Alibaba Cloud Quantum Development Platform: Large-Scale Classical Simulation of Quantum Circuits](#), by Fang Zhang, Yaoyun Shi, et al., arXiv, September 2019 (5 pages).
- [1672] [Full-State Quantum Circuit Simulation by Using Data Compression](#), by Xin-Chuan Wu, 2020 (29 slides).
- [1673] [Closing the "Quantum Supremacy" Gap: Achieving Real-Time Simulation of a Random Quantum Circuit Using a New Sunway Supercomputer](#), by Yong Liu, Dexun Chen, et al., arXiv, November 2021 (18 pages).
- [1674] [China's exascale quantum simulation not all it appears](#), by Nicole Hemsoth, November 2021.
- [1675] [Hyper-optimized tensor network contraction](#), by Johnnie Gray1 and2 and Stefanos Kourtis1 and3 and4, Quantum, March 2021 (22 pages).
- [1676] [Output Prediction of Quantum Circuits based on Graph Neural Networks](#), by Yuxiang Liu, Fanxu Meng, Lu Wang, Yi Hu, Zaichen Zhang, and Xutao Yu, arXiv, March 2026 (24 pages).
- [1677] [AI-Powered Noisy Quantum Emulation: Generalized Gate-Based Protocols for Hardware-Agnostic Simulation](#), by Matthew Ho, Jun Yong Khoo, Adrian M. Mak, and Stefano Carrazza, arXiv, February 2025 (12 pages).
- [1678] [Optimizing ZX-diagrams with deep reinforcement learning](#), by Maximilian Nägeli and Florian Marquardt, Machine Learning: Science and Technology, September 2024 (14 pages).
- [1679] [Using Reinforcement Learning to Perform Qubit Routing in Quantum Compilers](#), by Matteo G. Pozzi, Steven J. Herbert, Akash Sengupta, Robert D. Mullins University of Cambridge Computer Laboratory, Cambridge Quantum Computing, Department of Engineering, and University of Cambridge), arXiv, July 2020 (13 pages).
- [1680] [Just Like the Real Thing: Fast Weak Simulation of Quantum Computation](#), by Stefan Hillmich, Igor L. Markov, and Robert Wille, arXiv, July 2020 (6 pages).
- [1681] [GroverGPT: A Large Language Model with 8 Billion Parameters for Quantum Searching](#), by Haoran Wang, Pingzhi Li, Min Chen, Jinglei Cheng, Junyu Liu, and Tianlong Chen, arXiv, February 2025 (12 pages).
- [1682] [Classical analog circuit emulation of quantum Grover search algorithm](#), by Samuel Feldman, Hassam Ghazali, and Andrey Rogachev, arXiv, June 2025 (6 pages).
- [1683] [A complex128 encodes the real and imaginary components of complex numbers each on a 64-bit floating-point number representing 8 bytes. The total makes 16 bytes = 128 bits.](#)
- [1684] [Quantum Computing on a Commodore 64 in 200 Lines of BASIC](#), by Davide Gessa, July 2023.
- [1685] [A comparison of HPC-based quantum computing simulators using Quantum Volume](#), by Lourens van Niekerk, Dhiraaj Kumar, Aasish Kumar Sharma, Tino Meisel, Martin Leandro Paleico, and Christian Boehme, arXiv, December 2024 (11 pages).
- [1686] [Validating quantum computers using randomized model circuits](#), by Andrew W. Cross, Lev S. Bishop, Sarah Sheldon, Paul D. Nation, and Jay M. Gambetta, Physical Review A, September 2019 (13 pages) (preprint on arXiv).
- [1687] [Scalable parallel simulation of quantum circuits on CPU and GPU systems](#), by Guolong Zhong, Yi Fan, and Zhenyu Li, arXiv, September 2025 (36 pages).
- [1688] [GPU-Accelerated Quantum Simulation of Stabilizer Circuits](#), by Muhammad Osama, Dimitrios Thanos, and Alfons Laarman, arXiv, March 2026 (36 pages).
- [1689] [Researchers successfully simulate a 64-qubit circuit](#), by Science China Press, June 2018.
- [1690] [The cost of simulating classically tractable quantum circuits and dynamics](#), by Su Yeon Chang, Supanut Thanasilp, Zoë Holmes, and M. Cerezo, arXiv, September 2026 (50 pages).
- [1691] [A Herculean task: Classical simulation of quantum computers](#), by Xiaosi Xu, Simon Benjamin, Jinzhao Sun, Xiao Yuan, and Pan Zhang, arXiv, February 2023 (14 pages).
- [1692] [Alibaba Says Its New "Tai Zhang" Is the World's Most Powerful Quantum Circuit Simulator](#), by Mos Zhang, Medium, May 2018.
- [1693] [Simulating quantum computation by contracting tensor networks](#), by Igor L. Markov and Yaoyun Shi, arXiv, July 2009 (21 pages).
- [1694] [Huawei Unveils Quantum Computing Simulation HiQ Cloud Service Platform](#), by Huawei, Huawei, October 2018.
- [1695] [Establishing the Quantum Supremacy Frontier with a 281 Pflop/s Simulation](#), by Benjamin Villalonga, Dmitry Lyakh, Sergio Boixo, Hartmut Neven, Travis S. Humble, Rupak Biswas, Eleanor G. Rieffel, Alan Ho, and Salvatore Mandrà, arXiv, May 2020 (14 pages).
- [1696] [Eclipse Science and Open Source Software for Quantum Computing](#), by Alex McCaskey, Eclipse Foundation, 2017.
- [1697] [A Language and Hardware Independent Approach to Quantum-Classical Computing](#), by Alexander J. McCaskey, Eugene F. Dumitrescu, Dmitry Liakh, Mengsu Chen, Wuchun Feng, and Travis S. Humble, arXiv, July 2018 (15 pages).
- [1698] [Atos roadmap in The Atos Quantum Program - Paving the way to quantum-accelerated HPC](#), by Jean-Pierre Panziera, June 2021 (10 slides).
- [1699] [Eviden makes significant progress in enhancing the scalability of Qubit emulation and platform capabilities |Eviden](#), by Eviden, November 2023.
- [1700] [Quantum Computer Simulations at Warp Speed: Assessing the Impact of GPU Acceleration](#), by Jennifer Faj, Ivy Peng, Jacob Wahlgren, and Stefano Markidis, arXiv, July 2023 (11 pages).
- [1701] [Universal Quantum Simulation of 50 Qubits on Europe's First Exascale Supercomputer Harnessing Its Heterogeneous CPU-GPU Architecture](#), by Hans De Raedt, Kristel Michielsen, Thomas Lippert, et al., arXiv, November 2025 (17 pages).
- [1702] [Bringing GPU acceleration to Hybrid Quantum-Classical Computing](#), by François Courteille and Sam Stanwyck, Teratec TQCI seminar, January 2023.
- [1703] [Nvidia's grace-hopper hybrid systems bring huge memory to bear](#), by Timothy Prickett Morgan, May 2023.
- [1704] [NVIDIA Teams With Google Quantum AI, IBM and Other Leaders to Speed Research in Quantum Computing](#), by Sam Stanwyck, November 2021.
- [1705] [Nvidia Unveils Onramp to Hybrid Quantum Computing](#), by Timothy Costa, March 2022.

- [1706] [cuQuantum SDK: A High-Performance Library for Accelerating Quantum Science](#), by Harun Bayraktar, Ali Charara, David Clark, Saul Cohen, Timothy Costa, Yao-Lung L. Fang, Yang Gao, Jack Guan, John Gunnels, Azzam Haidar, Andreas Hehn, Markus Hohnerbach, Matthew Jones, Tom Lubowe, Dmitry Lyakh, Shinya Morino, Paul Springer, Sam Stanwyck, Igor Terentyev, Satya Varadhan, Jonathan Wong, and Takuma Yamaguchi, arXiv, August 2023 (13 pages).
- [1707] [Multi-GPU-Enabled Hybrid Quantum-Classical Workflow in Quantum-HPC Middleware: Applications in Quantum Simulations](#), by Kuan-Cheng Chen, Xiaoren Li, Xiaotian Xu, Yun-Yuan Wang, and Chen-Yu Liu, arXiv, March 2024 (8 pages).
- [1708] [Platform Architecture for Tight Coupling of High-Performance Computing with Quantum Processors](#), by Shane A. Caldwell, Travis Humble, et al., arXiv, October 2025 (43 pages).
- [1709] [Simulating quantum circuits with Amazon Braket](#), by Katharine Hyatt et al., September 2021.
- [1710] [DiaQ: Efficient State-Vector Quantum Simulation](#), by Srikar Chundury, Jiajia Li, In-Saeng Suh, and Frank Mueller, arXiv, April 2024 (11 pages).
- [1711] [Non-zero noise extrapolation: accurately simulating noisy quantum circuits with tensor networks](#), by Anthony P. Thompson, Arie Soeteman, Chris Cade, and Ido Niesen, arXiv, January 2025 (33 pages).
- [1712] [Simulator Demonstration of Large Scale Variational Quantum Algorithm on HPC Cluster](#), by Mikio Morita, Yoshinori Tomita, Junpei Koyama, and Koichi Kimura, arXiv, September 2024 (11 pages).
- [1713] [High Performance Quantum Emulation for Chemistry Applications with Hyperion](#), by Olivier Adjoua, Siwar Badredine, César Feniou, Igor Chollet, Diata Traore, Guillaume Michel, and Jean-Philip Piquemal, arXiv, April 2026 (15 pages).
- [1714] [qHiPSTER: The Quantum High Performance Software Testing Environment](#), by Mikhail Smelyanskiy, Nicolas P. D. Sawaya, and Alán Aspuru-Guzik, arXiv, May 2016 (9 pages).
- [1715] [Quantum Researchers Use Intel Quantum SDK to Explore Complex Problem Solving](#), by Scott Bair, February 2023.
- [1716] [phase2: Full-State Vector Simulation of Quantum Time Evolution at Scale](#), by Marek Miller, Jakob Günther, Freek Witteveen, Matthew S. Teynor, Mihael Erakovic, Markus Reiher, Gemma C. Solomon, and Matthias Christandl, arXiv, April 2025 (31 pages).
- [1717] [Psitrum: An Open Source Simulator for Universal Quantum Computers](#), by Mohammed Alghadeer, Eid Aldawsari, Raja Selvarajan, Khaled Alutaibi, Sabre Kais, and Fahhad H Alharbi, arXiv, February 2023 (28 pages).
- [1718] [QEA: An Accelerator for Quantum Circuit Simulation with Resources Efficiency and Flexibility](#), by Van Duy Tran, Tuan Hai Vu, Vu Trung Duong Le, Hoai Luan Pham, and Yasuhiko Nakashima, arXiv, May 2025 (6 pages).
- [1719] [Qandle: Accelerating State Vector Simulation Using Gate-Matrix Caching and Circuit Splitting](#), by Gerhard Stenzel, Sebastian Zielinski, Michael Kölle, Philipp Altmann, Jonas Nüßlein, and Thomas Gabor, arXiv, April 2024 (8 pages).
- [1720] [Q-GEAR: Improving quantum simulation framework](#), by Ziqing Guo, Ziwen Pan, and Jan Baleski, arXiv, April 2025 (10 pages).
- [1721] [QCLAB++: Simulating Quantum Circuits on GPUs](#), by Roel Van Beeumen, Daan Camps, and Neil Mehta, arXiv, February 2023 (13 pages).
- [1722] [Simulating Bell inequalities with Qibo](#), by Isabella Masina, Giuseppe Lo Presti, Matteo Robbiati, and Michele Grossi, arXiv, April 2025 (22 pages).
- [1723] [Qiboml: towards the orchestration of quantum-classical machine learning](#), by Matteo Robbiati, Stefano Carrazza, et al., arXiv, October 2025 (14 pages).
- [1724] [Qimax: Efficient quantum simulation via GPU-accelerated extended stabilizer formalism](#), by Vu Tuan Hai, Bui Cao Doanh, Le Vu Trung Duong, Pham Hoai Luan, and Yasuhiko Nakashima, arXiv, May 2025 (11 pages).
- [1725] [Fast Simulation of High-Depth QAOA Circuits](#), by Danylo Lykov, Marco Pistoia, et al., arXiv, September 2023 (10 pages).
- [1726] [Exact and approximate simulation of large quantum circuits on a single GPU](#), by Daniel Strano, Benn Bollay, Aryan Blaauw, Nathan Shammah, William J. Zeng, and Andrea Mari, arXiv, September 2023 (10 pages).
- [1727] [QSeqSim: A Symbolic Simulator for Qiskit While Loops Using Sequential Quantum Circuits](#), by Zihao Li, Ji Guan, and Mingsheng Ying, arXiv, May 2026 (26 pages).
- [1728] [QCE: A Simulator for Quantum Computer Hardware](#), by Kristel Michiels and Hans de Raedt, Turkish Journal of Physics, February 2022 (29 pages).
- [1729] [Benchmarking Quantum Red TEA on CPUs, GPUs, and TPUs](#), by Daniel Jaschke, Marco Ballarin, Nora Reinić, Luka Pavešić, and Simone Montangero, arXiv, February 2025 (15 pages).
- [1730] [QueenV2: Future of Quantum Circuit Simulation](#), by Chuan-Chi Wang, arXiv, September 2024 (5 pages).
- [1731] [The Virtual Quantum Device \(VQD\): A tool for detailed emulation of quantum computers](#), by Cica Gustiani, Tyson Jones, and Simon C. Benjamin, arXiv, February 2025 (22 pages).
- [1732] [Qymera: Simulating Quantum Circuits using RDBMS](#), by Tim Littau and Rihan Hai, arXiv, June 2025 (4 pages).
- [1733] [Density Matrix Renormalization Group with Tensor Processing Units](#), by Martin Ganah et al., February 2023 (17 pages).
- [1734] [Matrix product states and projected entangled pair states: Concepts, symmetries, theorems](#), by J. Ignacio Cirac, David Pérez-García, Norbert Schuch, and Frank Verstraete, Reviews of Modern Physics, December 2021.
- [1735] [SOFT: a high-performance simulator for universal fault-tolerant quantum circuits](#), by Riling Li, Keli Zheng, Yiming Zhang, Huazhe Lou, Shenggang Ying, Ke Liu, and Xiaoming Sun, arXiv, December 2025 (9 pages).
- [1736] [TANQ-Sim: Tensorcore Accelerated Noisy Quantum System Simulation via QIR on Perlmutter HPC](#), by Ang Li, Chenxu Liu, Samuel Stein, In-Saeng Suh, Muqing Zheng, Meng Wang, Yue Shi, Bo Fang, Martin Roetteler, and Travis Humble, arXiv, October 2024 (14 pages).
- [1737] [TensorCircuit-NG: A Universal, Composable, and Scalable Platform for Quantum Computing and Quantum Simulation](#), by Shi-Xin Zhang, Yu-Qin Chen, Weitang Li, Jiace Sun, Wei-Guo Ma, Pei-Lin Zheng, Yu-Xiang Huang, Qi-Xiang Wang, Hui Yu, Zhuo Li, Xuyang Huang, Zong-Liang Li, Zhou-Quan Wan, Shuo Liu, Jiezhong Qiu, Jiaqi Miao, Zixuan Song, Yuxuan Yan, Kazuki Tsuoka, Pan Zhang, Lei Wang, Heng Fan, Chang-Yu Hsieh, Hong Yao, and Tao Xiang, arXiv, February 2026 (33 pages).
- [1738] [TornadoQSim: An Open-source High-Performance and Modular Quantum Circuit Simulation Framework](#), by Ales Kubicek, Athanasios Stratikopoulos, Juan Fumero, Nikos Foutris, and Christos Kotselidis, arXiv, May 2023 (29 pages).
- [1739] [Introducing UNIQe: The Unconventional Noiseless Intermediate Quantum Emulator](#), by Reece Robertson and Dan Ventura, arXiv, September 2024 (25 pages).
- [1740] [How to Write a Simulator for Quantum Circuits from Scratch: A Tutorial](#), by Michael J. McGuffin, Jean-Marc Robert, and Kazuki Ikeda, arXiv, June 2025 (12 pages).
- [1741] [Cloud on-demand emulation of quantum dynamics with tensor networks](#), by Kemal Bidzhiev, Anne-Claire Le Henaff, Alvin Sashala Naik, et al., arXiv, February 2023 (12 pages).

- [1742] [Simulating Neutral Atom Quantum Systems with Tensor Network States](#), by James Allen, Matthew Otten, Stephen Gray, and Bryan K. Clark, arXiv, May 2025 (17 pages).
- [1743] [Piquasso: A Photonic Quantum Computer Simulation Software Platform](#), by Zoltán Kolarovszki, Zoltán Zimborás, et al., arXiv, March 2025 (30 pages).
- [1744] [SOQCS: A Stochastic Optical Quantum Circuit Simulator](#), by Javier Osca and Jiri Vala, arXiv, January 2024 (23 pages).
- [1745] [Classical algorithm for simulating experimental Gaussian boson sampling](#), by Changhun Oh, Minzhao Liu, Yuri Alexeev, Bill Fefferman, and Liang Jiang, Nature Physics, June 2024 (preprint on [arXiv](#)).
- [1746] [On classical simulation algorithms for noisy Boson Sampling](#), by Changhun Oh, Liang Jiang, and Bill Fefferman, arXiv, January 2023 (29 pages).
- [1747] [Simulating Gaussian boson sampling quantum computers](#), by Alexander S. Delliós, Margaret D. Reid, and Peter D. Drummond, arXiv, August 2023 (16 pages).
- [1748] [Gate-based quantum simulation of Gaussian bosonic circuits on exponentially many modes](#), by Alice Barthe, M. Cerezo, Andrew T. Sornborger, Martin Larocca, and Diego García-Martín, arXiv, July 2024 (14 pages).
- [1749] [Efficient Classical Algorithms for Simulating Gaussian Boson Sampling on Graphs](#), by Yexin Zhang, Tongyang Li, et al., arXiv, May 2025 (30 pages).
- [1750] [Hybrid Method of Efficient Simulation of Physics Applications for a Quantum Computer](#), by Carla Rieger, Albert T. Schmitz, Gehad Salem, Massimiliano Incudini, Sofia Vallecorsa, Anne Y. Matsuura, Michele Grossi, and Gian Giacomo Guerreschi, arXiv, July 2026 (24 pages).
- [1751] [ffsim: Faster simulation of fermionic quantum circuits](#), by Kevin J. Sung, Inho Choi, Mirko Amico, Bartholomew Andrews, Esra Ayantuna, Yukio Kawashima, Wan-Hsuan Lin, David Omanovic, Samuele Piccinelli, Javier Robledo Moreno, Abdullah Ash Saki, James Shee, Soyoung Shin, Minh C. Tran, Kento Ueda, Haimeng Zhang, and Mario Motta, arXiv, May 2026 (24 pages).
- [1752] [iQuantum: A Case for Modeling and Simulation of Quantum Computing Environments](#), by Hoa T. Nguyen, Muhammad Usman, and Rajkumar Buyya, arXiv, March 2023 (10 pages).
- [1753] [Cost of Emulating a Small Quantum Annealing Problem in the Circuit-Model](#), by Javier Gonzalez-Conde, Zachary Morrell, Marc Vuffray, Tameem Albash, and Carleton Coffrin, arXiv, June 2025 (24 pages).
- [1754] [Stim: a fast stabilizer circuit simulator](#), by Craig Gidney, arXiv, June 2021 (26 pages).
- [1755] [Quantum computer-aided design: digital quantum simulation of quantum processors](#), by Thi Ha Kyaw, William D. Oliver, Gian Giacomo Guerreschi, Alán Aspuru-Guzik, et al., arXiv, October 2021 (17 pages).
- [1756] [GCAMPS: A Scalable Classical Simulator for Qudit Systems](#), by Ben Harper, Azar C. Nakhl, Thomas Quella, Martin Sevier, and Muhammad Usman, arXiv, November 2025 (9 pages).
- [1757] [PIMutation: Exploring the Potential of PIM Architecture for Quantum Circuit Simulation](#), by Dongin Lee, Won Woo Ro, et al., arXiv, March 2025 (8 pages).
- [1758] [FQsun: A Configurable Wave Function-Based Quantum Emulator for Power-Efficient Quantum Simulations](#), by Tuan Hai Vu, Vu Trung Duong Le, Hoai Luan Pham, Quoc Chuong Nguyen, and Yasuhiko Nakashima, arXiv, March 2025 (15 pages).
- [1759] [AMARETTO: Enabling Efficient Quantum Algorithm Emulation on Low-Tier FPGAs](#), by Christian Conti, Deborah Volpe, Mariagrazia Graziano, Maurizio Zamboni, and Giovanna Turvani, arXiv, November 2024 (4 pages).
- [1760] [Scalable Program Implementation and Simulation of the Large-Scale Quantum Algorithm: \$1024 \times 1024\$ Quantum Linear Solver and Beyond](#), by Zhao-Yun Chen, Cheng Xue, Xi-Ning Zhuang, Tai-Ping Sun, Huan-Yu Liu, Ye Li, Yu-Chun Wu, and Guo-Ping Guo, arXiv, March 2023 (17 pages).
- [1761] [Implementing NChooseK on IBM Q Quantum Computer Systems](#), by Harsh Khetawat, May 2019 (15 pages).
- [1762] [Validating Large-Scale Quantum Machine Learning: Efficient Simulation of Quantum Support Vector Machines Using Tensor Networks](#), by Kuan-Cheng Chen, Tai-Yue Li, Yun-Yuan Wang, Simon See, Chun-Chieh Wang, Robert Wille, Nan-Yow Chen, An-Cheng Yang, and Chun-Yu Lin, arXiv, January 2025 (23 pages).
- [1763] [Classical simulations of noisy variational quantum circuits](#), by Enrico Fontana, Manuel S. Rudolph, Ross Duncan, Ivan Rungger, and Cristina Cirstoiu, arXiv, June 2023 (13 pages).
- [1764] [Dynamical cluster-based optimization of tensor network algorithms for quantum circuit simulations](#), by Andrea De Girolamo, Paolo Facchi, Peter Rabl, Saverio Pascazio, Cosmo Lupo, and Giuseppe Magnifico, arXiv, February 2025 (17 pages).
- [1765] [Towards practical and massively parallel quantum computing emulation for quantum chemistry](#), by Honghui Shang, Yi Fan, Li Shen, Chu Guo, Jie Liu, Xiaohui Duan, Fang Li, and Zhenyu Li, arXiv, March 2023 (13 pages).
- [1766] [Approximation Algorithm for Noisy Quantum Circuit Simulation](#), by Mingyu Huang, Ji Guan, Wang Fang, and Mingsheng Ying, arXiv, November 2023 (8 pages).
- [1767] [CUAOA: A Novel CUDA-Accelerated Simulation Framework for the QAOA](#), by Jonas Stein, Jonas Blenninger, David Bucher, Peter J. Eder, Elif Çetiner, Maximilian Zorn, and Claudia Linnhoff-Popien, arXiv, December 2024 (6 pages).
- [1768] [Towards Optimizations of Quantum Circuit Simulation for Solving Max-Cut Problems with QAOA](#), by Yu-Cheng Lin, Chuan-Chi Wang, Chia-Heng Tu, and Shih-Hao Hung, arXiv, December 2023 (7 pages).
- [1769] [Adiabatic elimination for composite open quantum systems: reduced model formulation and numerical simulations](#), by François-Marie Le Régent and Pierre Rouchon, arXiv, November 2023 (25 pages).
- [1770] [Benchmarking quantum computers with any quantum algorithm](#), by Stefan K. Seritan, Aditya Dhumentarao, Aidan Q. Wilber-Gauthier, Kenneth M. Rudinger, Antonio E. Russo, Robin Blume-Kohout, Andrew D. Baczewski, and Timothy Proctor, arXiv, August 2025 (15 pages).
- [1771] [Performance of Quantum Approximate Optimization with Quantum Error Detection](#), by Zichang He, David Amaro, Ruslan Shaydulin, and Marco Pistoia, arXiv, May 2025 (6 pages).
- [1772] [Threshold for Fault-tolerant Quantum Advantage with the Quantum Approximate Optimization Algorithm](#), by Sivaprasad Omanakuttan, Marco Pistoia, et al., arXiv, April 2025 (30 pages).
- [1773] [Is Circuit Depth Accurate for Comparing Quantum Circuit Runtimes?](#), by Matthew Tremba, Ji Liu, and Paul Hovland, arXiv, May 2025 (8 pages).
- [1774] [QuRE: The Quantum Resource Estimator Toolbox](#), by Martin Suchara, October 2013 (8 pages).
- [1775] [Design Fault Tolerant Quantum Computing applications with the open-source Resource Estimator](#), by Fabrice Frachon et al., January 2024.
- [1776] [Using Azure Quantum Resource Estimator for Assessing Performance of Fault Tolerant Quantum Computation](#), by Wim van Dam, Mariia Mykhailova, and Mathias Soeken, arXiv, May 2024 (6 pages).
- [1777] [Alice&Bob tests azure quantum resource estimator, highlighting the need for fault-tolerant qubits](#), by Blaise Vignon, November 2022.

- [1778] [An Early Investigation of the HHL Quantum Linear Solver for Scientific Applications](#), by Muqing Zheng, Chenxu Liu, Samuel Stein, Xiangyu Li, Johannes Mülmenstädt, Yousu Chen, and Ang Li, arXiv, June 2025 (21 pages).
- [1779] [QREChem: Quantum Resource Estimation Software for Chemistry Applications](#), by Matthew Otten, Stephen K. Gray, et al., arXiv, March 2024 (8 pages).
- [1780] [In Search of Quantum Advantage: Estimating the Number of Shots in Quantum Kernel Methods](#), by Artur Miroszewski, Marco Fellous Asiani, Jakub Mielczarek, Bertrand Le Saux, and Jakub Nalepa, arXiv, July 2024 (13 pages).
- [1781] [Expressing and Analyzing Quantum Algorithms with Qualtran](#), by Matthew P. Harrigan, Ryan Babbush, Nicholas C. Rubin, et al., arXiv, September 2024 (39 pages).
- [1782] [Feasibility of accelerating homogeneous catalyst discovery with fault-tolerant quantum computers](#), by Nicole Bellonzi, Alán Aspuru-Guzik, et al., arXiv, June 2024 (27 pages).
- [1783] [PsiQuantum announces QREF and Bartiq: Open-Source Software for Better Tools, Libraries and Datasets](#), by Psi-Quantum, May 2024.
- [1784] [A Scheduler for the Active Volume Architecture](#), by Sam Heavey and Athena Caesura, arXiv, 2026 (24 pages).
- [1785] [Active volume: An architecture for efficient fault-tolerant quantum computers with limited non-local connections](#), by Daniel Litinski and Naomi Nickerson, arXiv, November 2022 (42 pages).
- [1786] [Superconducting qubits at the utility scale: the potential and limitations of modularity](#), by S. N. Saadatmand, Josh Y. Mutus, et al., arXiv, June 2025 (34 pages).
- [1787] [Quantifying fault tolerant simulation of strongly correlated systems using the Fermi-Hubbard model](#), by Anjali A. Agrawal, Alexander F. Kemper, et al., arXiv, June 2024 (18 pages).
- [1788] [AutoQuREO: A Framework for Automated Quantum Resource Estimation and Optimization](#), by Harshkumar Oza, Aritra Sarkar, Syed Naqi Abbas, Rahul Bhowmick, Aryan Prakash, Prateek P Kulkarni, and Krishna Kumar Sabapathy, arXiv, August 2026 (47 pages).
- [1789] [How to estimate the resource requirements of quantum algorithms](#), by Jay Soni and Anton Naim Ibrahim, Xanadu PennyLane, January 2026.
- [1790] [Quantum Computing Technology Roadmaps and Capability Assessment for Scientific Computing - An analysis of use cases from the NERSC workload](#), by Daan Camps, Nicholas J. Wright, et al., arXiv, September 2025 (69 pages).
- [1791] [Utilizing Resource Estimation for the Development of Quantum Computing Applications](#), by Nils Quetschlich, Mathias Soeken, Prakash Murali, and Robert Wille, arXiv, August 2024 (7 pages).
- [1792] [Towards Quantum Advantage on Noisy Quantum Computers](#), by Ismail Yunus Akhalwaya et al., arXiv, March 2024 (34 pages).
- [1793] [A Threshold for Quantum Advantage in Derivative Pricing](#), by Shouvanik Chakrabarti, Rajiv Krishnakumar, Guglielmo Mazzola, Nikitas Stamatopoulos, Stefan Woerner, and William J. Zeng, Quantum, June 2021 (41 pages).
- [1794] [How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits](#), by Craig Gidney and Martin Ekerå, Quantum Journal, 2019 (28 pages).
- [1795] [Quantum Resources Required for Binding Affinity Calculations of Amyloid Beta](#), by Matthew Otten, Travis Humble, Nicole Bellonzi, Norm Tubman, and Nick Rubin, arXiv, June 2024 (15 pages).
- [1796] [Quantum Computing for Corrosion-Resistant Materials and Anti-Corrosive Coatings Design](#), by Nam Nguyen, Christopher N. Beall, Thomas S. Humble, and Nicholas P. Lubbers, arXiv, June 2024 (50 pages).
- [1797] [Feasibility of Accelerating Homogeneous Catalyst Discovery with Fault-Tolerant Quantum Computers](#), by Nicole Bellonzi, Matthew Otten, Norm Tubman, Nick Rubin, and Thomas S. Humble, arXiv, June 2024 (43 pages).
- [1798] [Feasibility of Accelerating Incompressible Computational Fluid Dynamics Simulations with Fault-Tolerant Quantum Computers](#), by John Penuel, Nicole Bellonzi, Matthew Otten, Norm Tubman, Nick Rubin, and Thomas S. Humble, arXiv, June 2024 (70 pages).
- [1799] [End-to-end quantum algorithms for tensor problems](#), by Enrico Fontana, Shouvanik Chakrabarti, et al., arXiv, October 2025 (53 pages).
- [1800] [Quantum Advantage in Resource Estimation](#), by William A. Simon and Peter J. Love, arXiv, December 2025 (32 pages).

This **Development tools classes** subsection contains 289 bibliographical references and notes containing at least 5,192 pages.

2.2 Academic quantum development tools

Here is an overview of the main quantum languages created to date, starting with languages that are independent of hardware architectures and that often originate from research laboratories. They have the disadvantage that they are not generally linked to cloud quantum computer offerings. The related researchers are the equivalents of the Kernighan and Richie (creators of the C language) and Bjarne Stroustrup (creator of C++) in the quantum realm! A good number of these languages come from Europe.

Classical concept	Quantum analogue
classical machine model variables variable assignments classical input	hybrid quantum architecture quantum registers elementary gates quantum measurement
subroutines argument and return types local variables dynamic memory	operators quantum data types scratch registers scratch space management
boolean expressions conditional execution selection conditional loops	quantum conditions conditional operators quantum if-statement quantum forking

Figure 113: Classical and quantum programming concepts. Source: [1801].

- **BQSKit** aka Berkeley Quantum Synthesis Toolkit is a quantum code compiler, transpiler and optimizer. Its QFactor circuit optimizer is using tensor networks [1802].

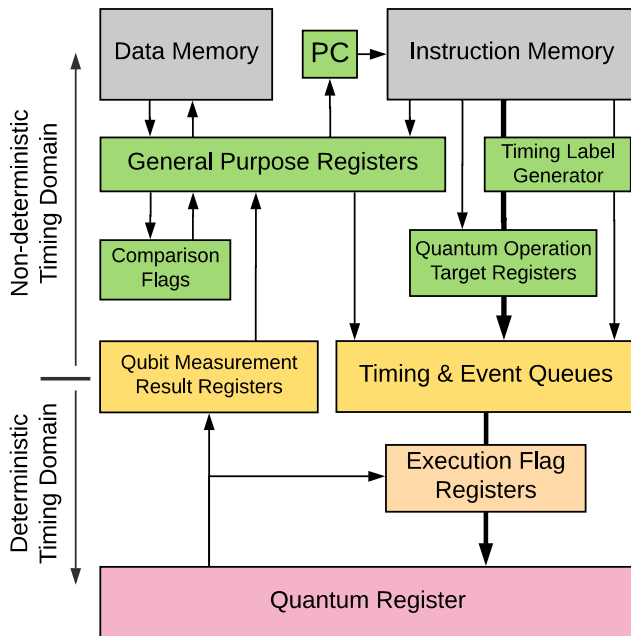


Figure 114: eQASM architecture, from Delft University. Source: [1803].

- **eQASM** is an intermediate quantum machine language from Delft University of Technology and QuTech, a research institute founded by TU Delft and TNO. It sits in between high-level programming tools (QASM) and the quantum accelerator. It is a compiled language, hence the “e” for executable. The compiler manages the dependencies with hardware implementation specifics. Tests have been carried out with a 7-qubit superconducting chip (Figure 114).
- **MCBeth** is a language created at Yale University and Chicago tailored for MBQC (measurement based quantum computing) programming [1804]. It can represent, program and simulate measurement-based and cluster state computation. Its compiled code can be executed directly on MBQC hardware as well as on traditional gate-based QPUs. This language is based on the initial work by Vincent Danos, Elham Kashefi and Prakash Panangaden in 2007 [1805].
- **MindSpore Quantum** is a full-stack hybrid quantum computing framework supporting gate-based programming, NISQ algorithms (ansatz library, VQE, QAOA, QNN) and an emulator. It comes from China [1806].
- **OpenQL** is an open-source quantum programming language created by TU Delft in 2020. It includes a high-level quantum programming language, its associated quantum compiler and a low-level assembly language, cQASM [1807].
- **ProjectQ** is a scripting language from ETH Zurich that takes the form of an open-source Python framework, released on GitHub since 2016. It includes a compiler that converts quantum code into C++ language for execution in a quantum simulator with a traditional processor [1808]. Launched in early 2017, it supports IBM’s quantum computers via their OpenQASM language, which is normal since ETH Zurich is a partner of the latter, as well as simulation on a traditional computer via a C++ implementation that supports up to 28 qubits (Figure 115). ProjectQ is compatible with OpenFermion from Rigetti and Google.
- **QCL** or Quantum Computation Language has syntax and data types close to those of the C language. This language is one of the first for quantum programming, created in 1998 by **Bernhard Omer** from the Austrian Institute of Technology in Vienna. It is described in a review paper which positions very well the conceptual differences between classical and quantum programming languages and methods (Figure 113).
- **Q Language** is an extension of the C++ language that provides classes for programming quantum gates (Hadamard, CNOT, SWAP, QFT for quantum Fourier transform) [1809].

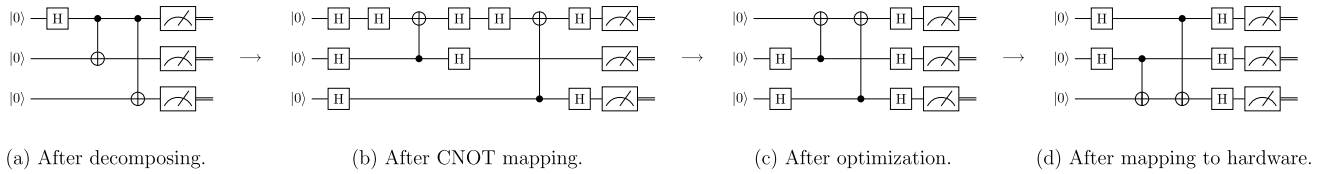


Figure 5: Individual stages of compiling an entangling operation for the IBM back-end. The high-level Entangle-gate is decomposed into its definition (Hadamard gate on the first qubit, followed by a sequence of controlled NOT gates on all other qubits). Then, the CNOT gates are remapped to satisfy the logical constraint that controlled NOT gates are allowed to act on one qubit only, followed by optimizing and mapping the circuit to the actual hardware.

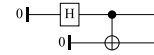
Figure 115: ProjectQ compiler entangling gates decomposition. Source: [1808].

- **Q.js** is a graphical quantum emulator launched in 2019, running in JavaScript and thus running in a browser [1810].
- **QCLAB** and **QCLAB++** are object-oriented Matlab toolboxes for constructing, representing, and simulating quantum circuits, the latter supporting C++ [1811].
- **QFC** and **QPL** are two functional languages defined by Peter Selinger, from Canada, the first one using a graphical syntax and the second one using a textual syntax [1812].
- **QML** is a functional programming language created by Thorsten Altenkirch and Jonathan Grattage (UK) [1813].
- **qGCL** or **Quantum Guarded Command Language** was created by Paolo Zuliani of the University of Newcastle.
- **QIRO** (Quantum Intermediate Representation for Optimization) is a two-dialect language proposal to enable quantum-classical co-optimizations [1814].
- **Qubiter** is an open-source language developed in Python that can be used on top of IBM's OpenQASM and Google's OpenFermion. It was created in 2017.
- **QNET** is a language from Stanford University created in 2012, which simulates the operation of quantum networks.
- **QOS**, a quantum operating system from Germany which is actually a compiler and runtime for NISQ applications which can implement multiprogramming features and reduce the effects of noise [1815].
- **Qrisp** is an open-source high-level quantum programming framework created at Fraunhofer FOKUS in Berlin and now an incubating Eclipse Foundation project under the EPL-2.0 license, with contributions from HSRM, IQM, eleQtron and DLR [1816]. Python-based, it replaces qubits and gates with typed quantum variables (integers, floats, modular numbers) and functions, and handles automatic uncomputation and qubit recycling. It compiles down to standard circuits that run on IBM, IQM, Quantinuum, Rigetti or

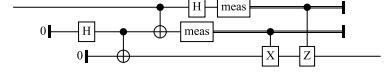
AQT quantum systems. Its Jasp extension connects to JAX and Catalyst for hybrid real-time control.

- **Quipper** is a language created in 2013 that builds on the classic **Haskell**^w language, created in 1990, to which it provides extensions in the form of data types and function libraries [1817]. It manipulates a software version of qRAM, an addressable quantum memory register, that is essential for the execution of algorithms such as Grover and QMLs (Figure 116). The language does not seem to have evolved since 2016. One of its creators is Benoît Valiron who teaches quantum programming at CentraleSupélec in France [1818].

```
bell100 :: Circ (Qubit, Qubit)
bell100 = do
  a <- plus_minus False
  (a,b) <- share a
  return (a,b)
```



```
teleport :: Qubit -> Circ Qubit
teleport q = do
  (a,b) <- bell100
  (x,y) <- alice q a
  b <- bob b (x,y)
  return b
```



```
teleport_generic_labeled :: (QData qa) => qa -> Circ qa
teleport_generic_labeled q = do
  comment_with_label "ENTER: bell100" q "q"
  (a,b) <- bell100_generic (qc_false q)
  comment_with_label "ENTER: alice" (a,b) ("a","b")
  (x,y) <- alice_generic q a
  comment_with_label "ENTER: bob" (x,y) ("x","y")
  b <- bob_generic b (x,y)
  return b
```

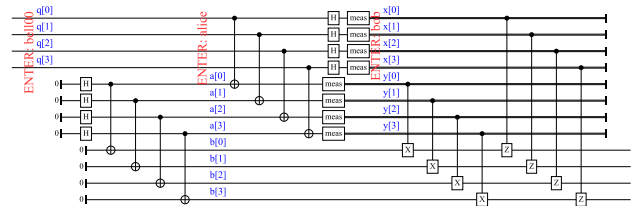


Figure 116: Example of Quipper code or an X and Z measurement, a teleportation and a QFT. Source: [1817].

- **Qumin** is a minimalist open-source quantum language designed in Greece in 2017 [1536].
- **Qunity** is a language created in 2022 at the Universities of Maryland and Chicago, and at AWS [1819].

Its goal is to unify quantum and classical programming concepts in a single language. Its syntax uses familiar programming constructs that can have both quantum and classical effects like summing linear operators, using exception handling syntax with projective measurements and using aliasing to induce entanglement. It can also automatically construct reversible subroutines from irreversible quantum algorithms through the uncomputation of “garbage” outputs. It can for example create full quantum oracle functions for algorithms like Grover, Deutsch-Jozsa and Simon. Qunity is still being developed and will be compiled to generate OpenQasm lower level code.

- **QuTiP** (Quantum Toolbox in Python) is another open-source quantum code emulation tool developed by Paul Nation of IBM, Robert Johansson of Rakuten and Franco Nori of RIKEN (Japan) and the University of Michigan. The project started in 2011. It targets superconducting qubits. The current version is the 5.0 [1820].
- **QWire** is another quantum programming language close to Quipper, launched in 2018, from the University of Pennsylvania [1821]. It is associated with a formal proof solution.
- **Scaffold** is a language developed at Princeton University [1822]. It is used to program traditional code which is then automatically transformed into quantum gates via its C2QG (Classical code to Quantum Gates) function. In particular, Scaffold can generate QASM. It can be interesting to develop oracles for search algorithms. Figure 117 contains a sample Scaffold code, almost easy to understand! Its development was also funded by IARPA.
- **ScaleQC** is a framework developed by Princeton researchers for hybrid quantum and classical computing [1823].
- **Silq** is a concise and static quantum programming language proposed by a ETH Zurich team [1824]. Quantum implementation languages of **lambda calculus**, conceptualized by Alonzo Church and Stephen Cole Kleene during the 1930s, followed. This type of computation makes it possible to solve very complex and NP-complete problems, the class of problems that can be verified in polynomial time and whose resolution requires exponential time on classical computers and potentially polynomial time on quantum computers [1825]! Researchers at the University of Chicago’s Enabling Practical-scale Quantum Computation (**EPIQC**) laboratory proposed a compiler that can improve the speed and reliability of quantum computers by a factor of 10. Here again, the compiler has to adapt to the underlying hardware architecture [1826]. The team used Google’s TensorFlow library to optimize the physical control parameters of the qubits.

```
// Pauli X, Pauli Y, Pauli Z, Hadamard, S, and T gates
gate X(qreg input[1]);
gate Y(qreg input[1]);
gate Z(qreg input[1]);
gate H(qreg input[1]);
gate S(qreg input[1]);
gate T(qreg input[1]);

// Daggered gates
gate Tdag(qreg input[1]);
gate Sdag(qreg input[1]);

// CNOT gate defined on two 1-qubit registers
gate CNOT(qreg target[1], qreg control[1]);

// Toffoli (CCNOT) gate
gate Toffoli(qreg target[1], qreg control1[1], qreg control2[1]);

// Rotation gates
gate Rz(qreg target[1], float angle);           //Arbitrary Rotation

// Controlled rotation
gate controlledRz(qreg target[1], qubit control[1], float angle);

// One-qubit measurement gates
gate measZ(qreg input[1], bit data);
gate measX(qreg input[1], bit data);

//One-qubit prepare gates: initializes to 0
gate prepZ(qreg input[1]);
gate prepX(qreg input[1]);

//Fredkin (controlled swap) gate
gate fredkin(qreg targ[1], qreg control1[1], qreg control2[1])
```

Figure 117: Scaffold code example. Source: [1822].

- **TWIST** is a language created in 2021 by MIT’s CSAIL lab that enforces how qubits are entangled or not, handles the notion of purity (a set of qubits not influenced by others) and enables the creation of safer programs. It introduces μQ , a small functional quantum language. Its creators plan to devise a higher-level abstraction language using TWIST [1829]. Although this project was jointly funded by IBM Research, it remains unclear whether IBM could reuse it in its quantum software toolbox.
- **Yao.jl** is a package for the Julia language used for creating quantum circuits.

Let’s add here the academic compilers that are dedicated to analog quantum computing:

- **SimuQ** which was created in 2023 at the University of Maryland [1537]. Positioned as a framework, it supports Hamiltonian programming and pulse-level compilation to target analog quantum simulators. It is using an abstract analog instruction set (AAIS) intermediate layer. Its target QPUs are those from IonQ, QuEra and even IBM superconducting qubits.
- **TurboQ**, launched in 2025 by the University of Pennsylvania and the DoE Lawrence Berkeley National Laboratory [1830]. It optimizes the optimization step that is quite costly with SimuQ, with compilation and optimization times exceeding 6 hours for a system using 100 qubits.

Most quantum programming software tools are open-source (Figure 119, Figure 120). Their differentiation is mainly concentrated on documentation and tutorials

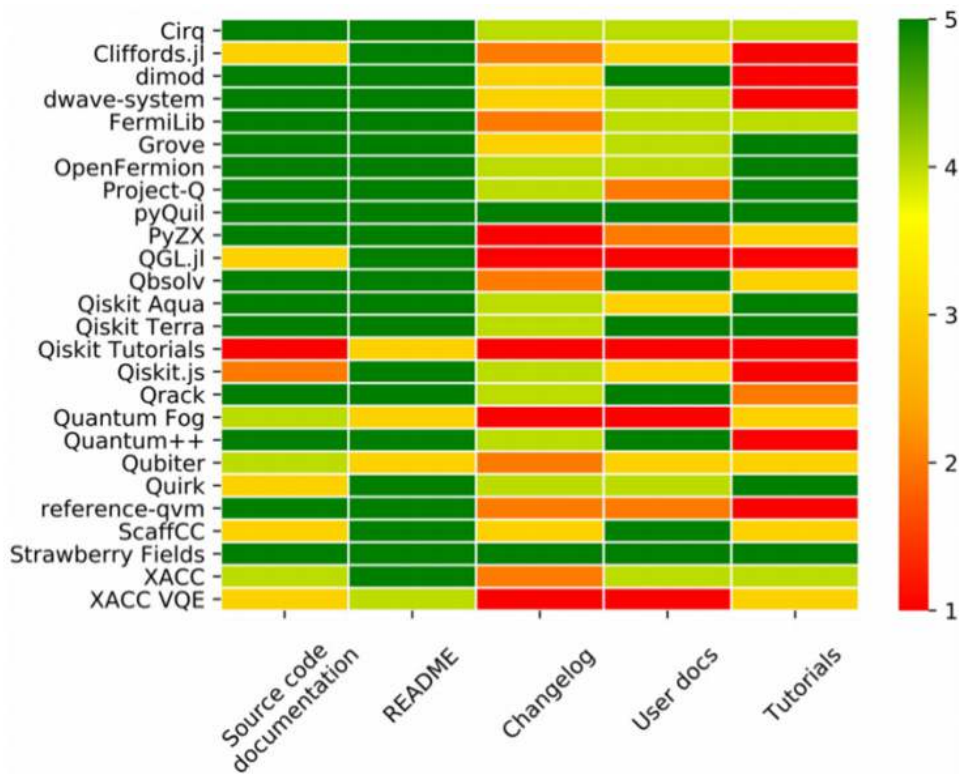


Figure 118: Heatmap of various quantum coding tools and their quality figures of merit in 2018. Source: [1827].

Open source quantum (2016 -)

2016	QETLAB	Matlab	University of Waterloo,Canada
2016	Liquil>	F#	Microsoft
2016	Quantum Fog	Python	Artiste-qb
2016	Qubiter	Python	Artiste-qb
2016	IBM Q Experience	-	IBM
2017	ProjectQ	Python	ETH Zurich
2017	Forest (QUIL)	Python	Rigetti
2017	QISKit	Python	IBM
2017	Quantum Optics.jl	Julia	Universität Innsbruck
2017	PsiQuaSP	C++.	Gegg M, Richter M
2018	Strawberry Fields	Python	Xanadu, Canada
2018	Quantum Dev Kit	Q#.	Microsoft
2018	<u>QCGPU</u>	Rust, OpenCI	Adam Kelly
2018	<u>NetKet</u>	C++	The Simons Foundation
2018	OpenFermion	Python	Google, Harvard,UMich, ETH ..

https://github.com/markf94/os_quantum_software

Figure 119: Summary timeline of the appearance of various quantum development tools as of 2021. Source: [1828], which provides an excellent overview of development tools covering the entire quantum software creation cycle, including the thorny issues of debugging and testing.

[1827] (Figure 118). However, in practice, few commercial application developers use the languages discussed in this section. Instead, they are hooked to the languages and toolkits provided by commercial quantum computer vendors listed afterwards. They are easily locked into “full stack” approaches that are proprietary in practice although also open sourced in principle.

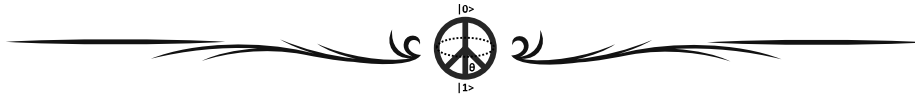
The most interesting thing about all this is that many development tools allow us to get our hands on small-scale quantum algorithms before scalable and usable

quantum computers are available. And most of them are also open-sourced and free to install and use.

Some optimization tools can also be mentioned here like **CutQC** which distributes in an optimized way a large quantum circuit onto several (non-connected) QPUs and classical platforms (CPU or GPU) for co-processing [1831]. It enables the use of NISQ QPUs at their optimum regime, when a small number of qubits have a sufficient fidelity. Obviously, it doesn’t generate an equivalent system to the sum of the qubits of the used QPUs.

Year	Language	Reference(s)	Semantics	Host Language	Paradigm
1996	Quantum Lambda Calculi	[188]	Denotational	lambda Calculus	Functional
1998	QCL	[214–217]		C	Imperative
2000	qGCL	[252, 329–331]	Operational	Pascal	Imperative
2003	λ_q	[295, 296]	Operational	Lambda Calculus	Functional
2003	Q language	[31, 32]		C++	Imperative
2004	QFC (QPL)	[256–258]	Denotational	Flowchart syntax (Textual syntax)	Functional
2005	QPAIlg	[143, 164]		Process calculus	Other
2005	QML	[10, 11, 115]	Denotational	Syntax similar to Haskell	Functional
2004	CQP	[103–105]	Operational	Process calculus	Other
2005	cQPL	[187]	Denotational		Functional
2006	LanQ	[196–199]	Operational	C	Imperative
2008	NDQJava	[312]		Java	Imperative
2009	Cove	[237]		C#	Imperative
2011	QuECT	[47]		Java	Circuit
2012	Scaffold	[1, 140]		C (C++)	Imperative
2013	QuaFL	[166]		Haskell	Functional
2013	Quipper	[116, 117]	Operational	Haskell	Functional
2013	Chisel-Q	[182]		Scala	Imperative, functional
2014	LIQUi ⟩	[306]	Denotational	F#	Functional
2015	Proto-Quipper	[244, 247]		Haskell	Functional
2016	QASM	[220]		Assembly language	Imperative
2016	FJQuantum	[81]		Feather-weight Java	Imperative
2016	ProjectQ	[124, 279, 285]		Python	Imperative, functional
2016	pyQuil (Quil)	[272]		Python	Imperative
2017	Forest	[61, 272]		Python	Declarative
2017	OpenQASM	[66]		Assembly language	Imperative
2017	qPCF	[222, 224]		Lambda calculus	Functional
2017	QWIRE	[225]	Denotational	Coq proof assistant	Circuit
2017	cQASM	[149]		Assembly language	Imperative
2017	Qiskit	[4, 242]		Python	Imperative, functional
2018	IQu	[223]		Idealized Algol	Imperative
2018	Strawberry Fields	[150, 151]		Python	Imperative, functional
2018	Blackbird	[150, 151]		Python	Imperative, functional
2018	QuantumOptics.jl	[161]		Julia	Imperative
2018	Cirq	[284]		Python	Imperative, functional
2018	Q#	[282]		C#	Imperative
2018	Q SI⟩	[181]		.Net language	Imperative
2020	Silq	[34]	Operational		Imperative, functional
2020	Quingo	[286]		Python	Imperative

Figure 120: Timeline of quantum programming tools. Source: [1828].



2.2.1 Bibliography and notes

These are the bibliographical references and notes for the **Academic development tools** subsection.

- [1536] [Qumin, a minimalist quantum programming language](#), by Alexander Singh, Konstantinos Giannakis, and Theodore Andronikos, arXiv, April 2017 (34 pages).
- [1537] [SimuQ: A Framework for Programming Quantum Hamiltonian Simulation with Analog Compilation](#), by Yuxiang Peng, Jacob Young, Pengyu Liu, and Xiaodi Wu, arXiv, November 2023 (34 pages).
- [1801] [Structured Quantum Programming](#), by Bernhard Omer, September 2009 (130 pages).

- [1802] [Berkeley Quantum Synthesis Toolkit](#), by LBL, Lawrence Berkeley Lab, 2019.
- [1803] [eQASM: An Executable Quantum Instruction Set Architecture](#), by X. Fu, Leo DiCarlo, K. Bertels, et al., arXiv, March 2019 (13 pages).
- [1804] [MCBeth: A Measurement Based Quantum Programming Language](#), by Aidan Evans, Seun Omonije, Robert Soulé, and Robert Rand, arXiv, June 2022 (24 pages).
- [1805] [The Measurement Calculus](#), by Vincent Danos, Elham Kashefi, and Prakash Panangaden, arXiv, April 2007 (46 pages).
- [1806] [MindSpore Quantum: A User-Friendly, High-Performance, and AI-Compatible Quantum Computing Framework](#), by Xusheng Xu, Man-Hong Yung, et al., arXiv, July 2024 (57 pages).

- [1807] [OpenQL : A Portable Quantum Programming Framework for Quantum Accelerators](#), by N. Khammassi, C. G. Al-mudever, et al., arXiv, May 2020 (13 pages).
- [1808] [ProjectQ: An Open Source Software Framework for Quantum Computing](#), by Damian Steiger et al., 2018 (13 pages).
- [1809] [Toward an architecture for quantum programming](#), by Stefano Bettelli, arXiv, 2003 (23 pages).
- [1810] [Quantum Programming: JavaScript \(Q.js\) - a drag and drop circuit editor](#), by Stewart, 2020.
- [1811] [QCLAB: A Matlab Toolbox for Quantum Computing](#), by Sophia Keip, Daan Camps, and Roel Van Beeumen, arXiv, March 2025 (12 pages).
- [1812] [Towards a Quantum Programming Language](#), by Peter Selinger, Mathematical Structures in Computer Science, November 2002 .
- [1813] [A functional quantum programming language](#), by Thorsten Altenkirch and Jonathan Grattage, arXiv, February 2005 (15 pages).
- [1814] [Enabling Dataflow Optimization for Quantum Programs](#), by David Ittah, Thomas Häner, Vadym Kliuchnikov, and Torsten Hoefer, arXiv, August 2021 (15 pages).
- [1815] [QOS: A Quantum Operating System](#), by Emmanouil Gior-tamis, Francisco Romão, Nathaniel Tornow, and Pramod Bhatotia, arXiv, April 2025 (19 pages).
- [1816] [Qrisp: A Framework for Compilable High-Level Program-ming of Gate-Based Quantum Computers](#), by Raphael Seidel, Sebastian Bock, René Zander, Matic Petrič, Niklas Steinmann, Nikolay Tcholtchev, and Manfred Hauswirth, arXiv, June 2024 (19 pages).
- [1817] [An Introduction to Quantum Programming in Quipper](#), by Alexander S. Green, Peter LeFanu Lumsdaine, Neil J. Ross, Peter Selinger, and Benoît Valiron, arXiv, April 2013 (15 pages).
- [1818] [Quantum Computation Model and Programming Paradigm](#), by Benoit Valiron, Journées LTP, December 2018 (67 slides).
- [1819] [Qunity: A Unified Language for Quantum and Classical Computing \(Extended Version\)](#), by Finn Voichick, Liyi Li, Robert Rand, and Michael Hicks, arXiv, March 2025 (79 pages).
- [1820] [QuTiP 5: The Quantum Toolbox in Python](#), by Neill Lam-bert, Franco Nori, et al., arXiv, December 2024 (70 pages).
- [1821] [QWIRE: a core language for quantum circuits](#), by Jennifer Paykin et al., POPL 17 proceedings, January 2017.
- [1822] [Scaffold: Quantum Programming Language](#), by Ali Javadi Abhari et al., Princeton University, July 2012.
- [1823] [ScaleQC: A Scalable Framework for Hybrid Computation on Quantum and Classical Processors](#), by Wei Tang and Margaret Martonosi, arXiv, July 2022 (12 pages).
- [1824] [Swiss scientists launch high-level quantum computing lan-guage](#), by ETH Zurich, ETH Zurich, June 2020.
- [1825] [A lambda calculus for quantum computation with classical control](#), by Peter Selinger et al., arXiv, 2004 (15 pages).
- [1826] [Optimized Compilation of Aggregated Instructions for Re-alistic Quantum Computers](#), by Yunong Shi et al., Pro-ceedings ASPLOS Optimized Compilation of Aggregated Instructions for Realistic Quantum Computers, April 2019 (14 pages)  (preprint on [arXiv](#)).
- [1827] [Open-source software in quantum computing](#), by Mark Fingerhuth, Thomas Babej, and Peter Wittek, PLOS, De-cember 2018 (28 pages).
- [1828] [Quantum Software Engineering: Landscapes and Horizons](#), by Jianjun Zhao, arXiv, December 2021 (34 pages).
- [1829] [Twist: Sound Reasoning for Purity and Entanglement in Quantum Programs](#), by Charles Yuan et al., January 2022 (32 pages).
- [1830] [QTurbo: A Robust and Efficient Compiler for Analog Quan-tum Simulation](#), by Junyu Zhou, Yuhao Liu, Shize Che, Anupam Mitra, Efehan Kökcü, Ermal Rrapaj, Costin Iancu, and Gushu Li, arXiv, June 2025 (15 pages).
- [1831] [Cutting Quantum Circuits to Run on Quantum and Classi-cal Platforms](#), by Wei Tang and Margaret Martonosi, arXiv, May 2022 (10 pages).

This **Academic development tools** subsection contains 33 bib-liographical references and notes containing at least 791 pages.

2.3 Vendor quantum-development tools

Even before general-purpose quantum computers become operational at an exploitable scale, competition over software platforms has already begun. Most major quantum computing vendors have adopted some degree of end-to-end vertical integration, from quantum processors and control systems to compilers, software development kits, simulators and cloud execution. This is particularly visible at IBM, Microsoft, Rigetti and D-Wave. The updated stacks shown in Table 10 help understand how these ecosystems emerged.

A vendor stack typically combines an intermediate or low-level representation, a Python-oriented SDK, compilation and transpilation tools, simulators, cloud access to QPUs and increasingly workflow services for hybrid quantum-classical execution. The boundaries between these layers are less rigid than they were in the late 2010s. Open standards such as OpenQASM and QIR, and cross-vendor frameworks such as Qiskit, Cirq and PennyLane, now coexist with vendor-specific execution services.

There is still no dominant hardware-independent abstraction that completely frees developers from understanding quantum data, quantum operations and hardware constraints. This remains a debated software engineering problem, particularly because classical abstractions such as unrestricted memory and control flow do not map directly to quantum computation [1834]. Several languages and DSLs provide higher-level abstractions, including Qwerty from Georgia Tech [1835, 1836], Rhyme from Switzerland [1837], and Classiq from Israel. Domain-specific approaches can synthesize or transform circuits from descriptions above the gate-sequence level [1838]. For FTQC algorithm design and resource analysis, **Qualtran** provides reusable algorithmic building blocks, hierarchical representations and architecture-independent resource counts [1839].

Open source alone is therefore not a sufficient differentiator between platforms. Relevant distinctions include which layers remain proprietary, who controls the open-source project, the license, the portability of intermediate representations, the supported hardware backends, and whether cloud execution and higher-level optimization services are tied to one vendor.

2.3.1 D-Wave

D-Wave’s current software stack is centered on **Ocean**, its Python SDK, and the **Leap** cloud service. As of August 2026, the Ocean SDK is version 9.4.0. Its main packages include **dimod** for binary and constrained quadratic models, **dwave-system** for QPU samplers and embedding composites, **dwave-cloud-client** for access to the Solver API, **dwave-hybrid** for user-defined hybrid workflows, and **dwave-optimization** for nonlinear optimization models. The current package inventory is maintained in the [Ocean documentation](#).

QUBO and Ising formulations remain central because they map naturally to the programmable Ising Hamiltonian of a D-Wave annealer. Ocean represents these problems as BQM (Binary Quadratic Model) objects. The newer stack also supports CQM (Constrained Quadratic Model) problems through hybrid solvers and nonlinear models through **dwave-optimization**. Communication with remote QPUs and hybrid solvers is handled through the Leap Solver API, or SAPI, via a REST interface exposed by **dwave-cloud-client**. The older QMI terminology and third-party QMASM language are best regarded as historical interfaces rather than the normal development path in 2026.

The open-source **dwave-hybrid** framework remains available for building custom classical-quantum workflows. D-Wave also provides managed hybrid solvers in Leap. In February 2026 its nonlinear hybrid solver was renamed **Stride Hybrid Solver**. As of August 2026, the current Stride release was version 1.38. Version 1.37 had added support for **dwave-optimization** 0.7.2, while Leap also maintained managed BQM and CQM hybrid solvers. These managed hybrid solvers combine classical algorithms with calls to a QPU. The role of the quantum annealer in the obtained performance is not well known. Current service changes are documented in the [Leap release notes](#).

D-Wave also released a [PyTorch plug-in](#) in 2025, aimed at hybrid machine-learning workflows such as restricted Boltzmann machines. This is an additional integration layer rather than a replacement for Ocean.

The relationship between D-Wave annealing and universal quantum computation needs a qualification. Ideal adiabatic quantum computation is computationally universal and polynomially equivalent to the circuit model under appropriate assumptions [1840]. This does not mean that a present D-Wave transverse-field annealer implements arbitrary gate-model circuits. Standard forward annealing also does not require an initial state close to the solution. Starting from a classically supplied candidate state is instead a feature of **reverse annealing**, which is useful for local refinement around that candidate.

Leap remains D-Wave’s cloud access layer for QPUs, hybrid solvers, account management and development workflows. The Leap V2 prices shown in Figure 123 are historical 2021 values and do not necessarily reflect current pricing. D-Wave’s current QPU fleet includes Advantage and Advantage2 systems, while exact solver identifiers and working graphs can change between service releases.

2.3.2 IBM

IBM’s software platform is centered on **Qiskit**, **OpenQASM** and IBM Quantum cloud services. Qiskit remains one of the most widely used quantum software

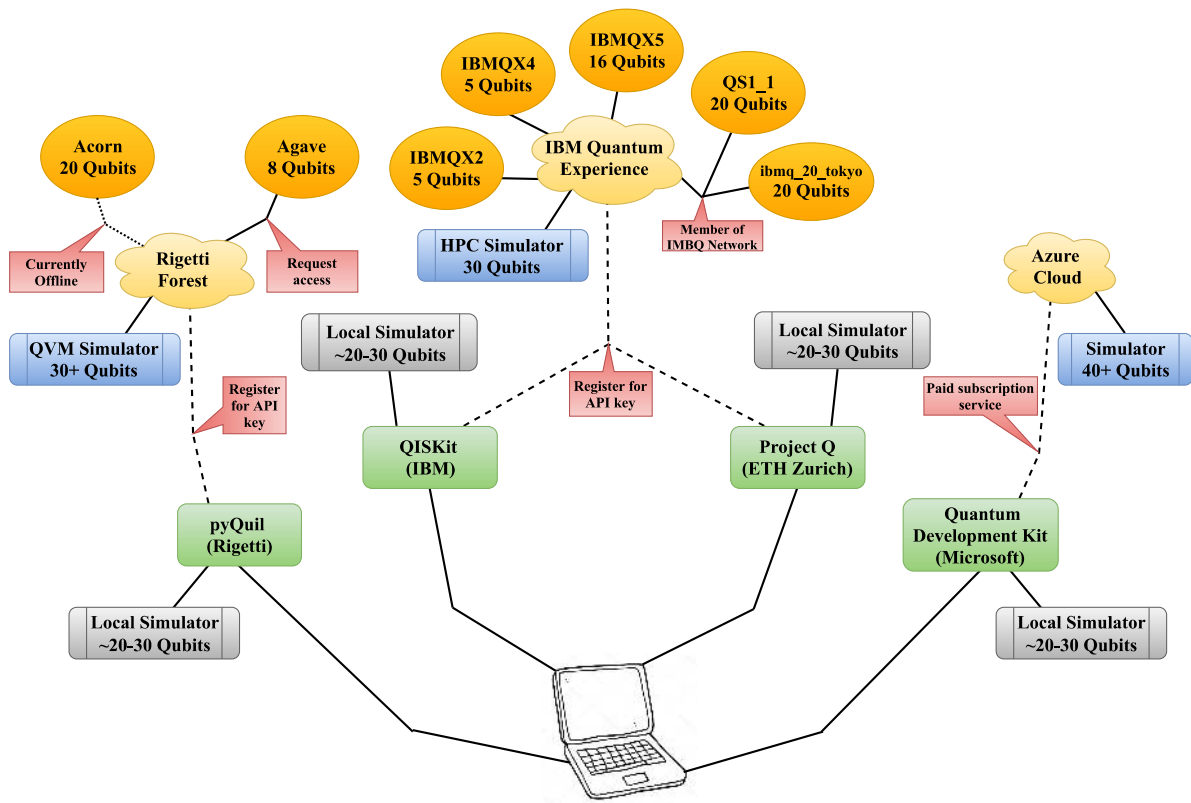


Figure 121: This 2019 diagram provides an historical snapshot of early gate-level quantum software platforms. Source: [1832].

	IBM	rigetti	D-Wave The Quantum Computing Company™	XANADU	Google	Microsoft	amazon	Bull
quantum emulators and simulators	Qiskit Aer	QVM	dimod, dwave-samplers	PennyLane Lightning	qsim, Cirq simulators	QDK simulators	LocalSimulator, SV1, DM1	myQLM, Qaptiva HPC
visual programming and integrated development environments	IBM Quantum Composer	QCS JupyterLab	Leap IDE, Codespaces		Cirq, Google Colab	VS Code QDK extension	Braket notebooks	
thematic quantum libraries (chemistry, finance, machine learning, ...)	Qiskit Nature, ML, Optimization, Functions		dwave-optimization, hybrid solvers	PennyLane QML, qchem	OpenFermion, Qualtran, TensorFlow Quantum	QDK Chemistry, QDK-EC, resource estimator	PennyLane, Hybrid Jobs	myQLM-fermion, optimization libraries
generic quantum libraries / full-stack	Qiskit	Quil SDK	Ocean SDK	PennyLane	Cirq	Microsoft QDK	Braket SDK	Qaptiva, myQLM
high level machine language (quantum circuits)	Qiskit circuits	pyQuil	NLM, CQM, BQM	PennyLane QNodes, Catalyst	Cirq circuits, Qualtran	Q#, Python QDK	Braket Circuit API	PyAQASM
low level machine language	OpenQASM 3	Quil, Quil-T	Ising, QUBO	MLIR, QIR	device-native gates	OpenQASM, QIR	OpenQASM 3, JAQCD	AQASM
qubits and quantum gates	superconducting	superconducting	quantum annealing	photonic qumodes	superconducting, IonQ	IonQ, Quantinuum, Pasqal, Rigetti	AQT, IonQ, IQM, QuEra, Rigetti	hardware-agnostic

Table 10: The current software stacks from large quantum vendors and cloud providers. Based on a 2018 schema found in [1833]. © Olivier Ezratty, 2020-September 2026.

frameworks, as illustrated by the 2024 Unitary Foundation survey in Figure 125.

OpenQASM is an open quantum assembly language introduced in 2017 [1841]. OpenQASM 3 added classical control flow, external functions, timing constructs and a richer description of quantum programs [1842]. Version 3.1 was the current language specification in August 2026.

Qiskit is primarily a Python SDK for constructing, transforming and executing quantum circuits [1843]. Qiskit 1.0 was released in 2024. As of September 2026, the current patch release is Qiskit 2.5.2. The older architectural description based on separately installed **Qiskit Terra** is obsolete because Terra was folded into the main Qiskit package. **Qiskit Aqua** was deprecated in 2021 and its application-level functionality was split into more specialized packages. The current SDK also exposes a

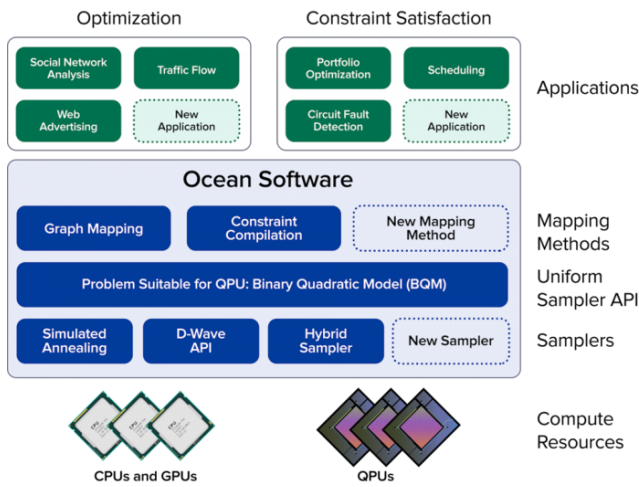


Figure 122: D-Wave's Ocean software architecture. The image is a conceptual view. Current software package names and versions are available in the Ocean documentation. Source: D-Wave.

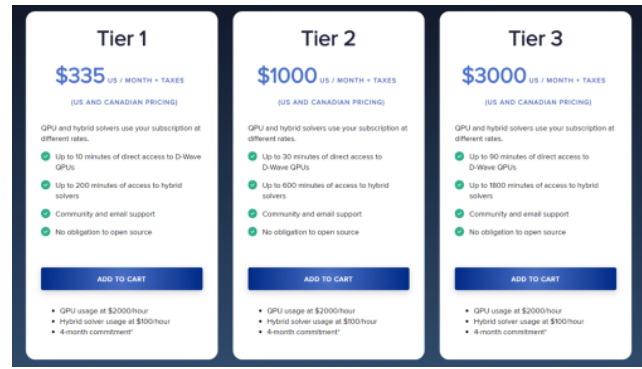


Figure 123: Historical D-Wave Leap pricing from 2021. These prices are not current and are retained only to document the evolution of the service.

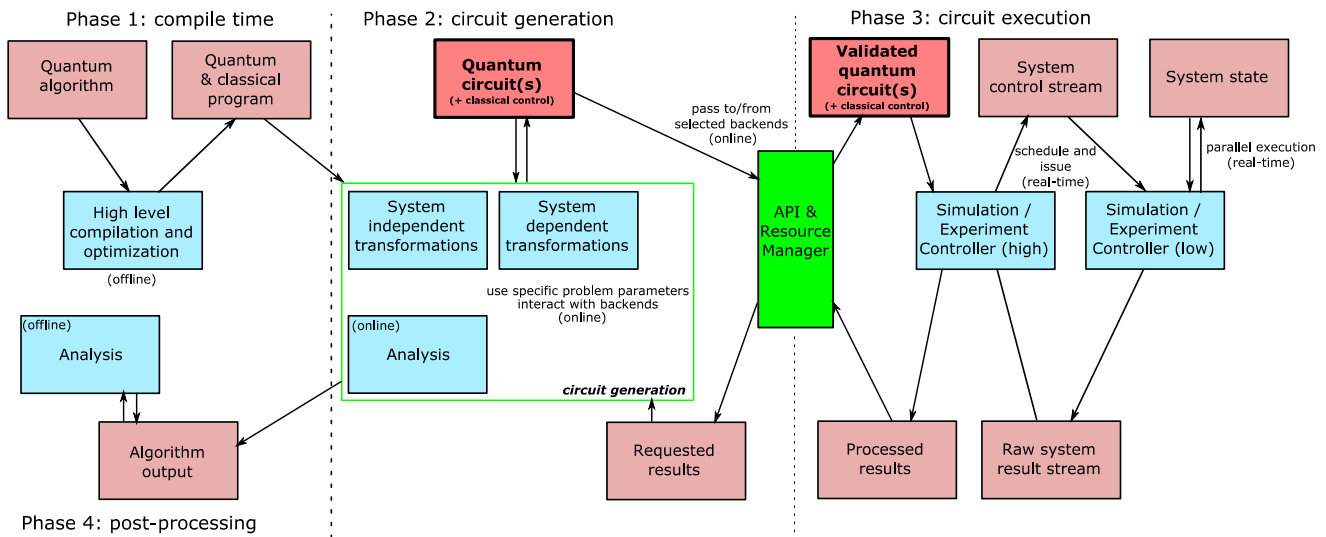


Figure 124: Historical Qiskit block diagram of processes and abstractions used to transform and execute a quantum algorithm. Source: [1841].

C API to parts of its Rust-based core. Current releases are documented at [IBM Quantum documentation](#).

The main current software layers can be summarized as follows. **Qiskit Aer** is the local high-performance simulator package. For an unrestricted dense state-vector simulation, memory scales as $O(2^n)$. With complex double-precision amplitudes, the state vector alone needs about 16×2^n bytes, or roughly 16 GiB at 30 qubits and 32 GiB at 31 qubits, before simulator overhead. Tensor-network, stabilizer and other specialized methods can reach much larger qubit counts when the circuit structure allows it. IBM retired its hosted cloud simulators in May 2024, which makes the former fixed cloud-simulator qubit limits obsolete.

IBM Quantum Compute Service is the name adopted in July 2026 for what had previously been called

the Qiskit Runtime service. The open-source `qiskit-ibm-runtime` client remains the IBM-specific provider layer, and the naming change did not require changes to existing applications. The service executes Qiskit primitives close to IBM quantum hardware and supports modes such as jobs, sessions and batches. IBM's 2021 claim of a 120x reduction in the wall-clock time of a particular variational workflow should therefore be kept as a historical systems result rather than interpreted as a generic algorithmic speedup [1844].

Qiskit Functions is a catalog of circuit-level and application-level services introduced in preview in 2024. It remains active in 2026 and includes IBM and third-party functions for compilation, error suppression and mitigation, chemistry, optimization and other workflows. **Qiskit Serverless** provides remote orchestration of long-running hybrid workloads and classical preprocess-

ing or post-processing. It is still labeled experimental. The hosted environment documented in 2026 provides a normal node with 24 virtual CPU cores and 120 GB of RAM. These services are described in the [current IBM Quantum services documentation](#).

Qiskit Addons are a newer modular layer for research capabilities that should be distinguished from Serverless. The current catalog includes circuit cutting, sample-based quantum diagonalization, approximate quantum compilation, multi-product formulas, operator backpropagation, Pauli propagation and several error-mitigation techniques. In particular, circuit cutting is implemented in `qiskit-addon-cutting`, not by Qiskit Serverless itself. Circuit cutting trades quantum circuit width or connectivity for classical reconstruction and sampling overhead. Its usefulness therefore depends on the circuit and on the cuts chosen, rather than on a blanket requirement that the target state be weakly entangled. See the [Qiskit Addons catalog](#).

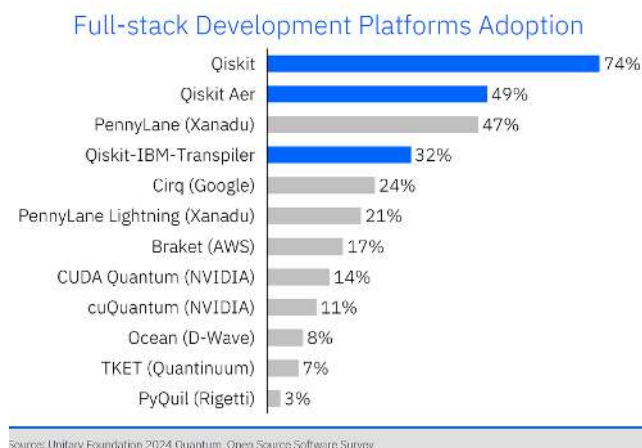


Figure 125: Qiskit was used by about three fourths of respondents in the 2024 Unitary Foundation survey of open-source quantum software.

Qiskit Machine Learning remains an open-source package for quantum machine learning [1845]. It was not first released in 2025, so the 2025 date should be associated with the cited software paper rather than with the creation of the library. **Qiskit Slurm** remains a reference integration for hybrid HPC-QPU workflows [1846].

Qiskit Code Assistant was launched as an LLM-based coding assistant in 2024 [1847–1849], but IBM discontinued the hosted preview service on May 29, 2026 and archived the VS Code and JupyterLab extensions. The trained models remain available for local use. This supersedes the previous description of Code Assistant as an active IBM cloud service.

Dynamic circuits combine mid-circuit measurements, feed-forward and conditional quantum operations. IBM has progressively integrated them into its execution stack since 2022. In 2024, IBM researchers demonstrated virtual CNOT interactions spanning two QPUs using local operations and classical communication, cut Bell-pair

quasi-probability decompositions, and real-time classical feed-forward [1850]. There was no quantum link and no physical Bell pair distributed between the QPUs. The experiment recreated the statistics of long-range entangling operations by circuit cutting and dynamic circuits. IBM researchers also proposed benchmarking methods for dynamic circuits [1851].

2.3.3 Rigetti

Rigetti’s current developer platform is **Quantum Cloud Services (QCS)**. Its programming model is centered on **Quil**, an instruction language that supports quantum gates, measurements, classical memory and control flow [1853]. Pulse-level extensions are described by **Quil-T**. The current **Quil SDK** combines **pyQuil**, the **quile** optimizing compiler and the **QVM** quantum virtual machine. Rigetti describes pyQuil as its flagship Python interface for constructing, compiling and running Quil programs on either the QVM or Rigetti QPUs. Current documentation is available at [Rigetti QCS](#).

The older **Forest** name now survives mainly in legacy installer and repository names, while current QCS documentation presents the **Quil SDK** as the developer suite. Likewise, **Grove** was an early library of algorithm examples and is no longer the organizing layer of the Rigetti stack. The current Quil SDK can be installed on Linux, macOS and Windows, so the former statement that Rigetti’s tools were not cross-platform is no longer correct. Current QCS documentation nevertheless recommends Rigetti’s hosted JupyterLab environment for QPU execution, because local installations are intended mainly for QVM development and do not have the network access needed to submit directly to Rigetti QPUs.

The simple Hadamard example in Figure 130 illustrates an ideal quantum random bit. Starting from $|0\rangle$, an ideal Hadamard prepares $(|0\rangle + |1\rangle)/\sqrt{2}$ and a computational-basis measurement returns 0 or 1 with probability 1/2. Real hardware can introduce bias through preparation, gate and measurement errors.

OpenFermion is a cross-vendor open-source Python library for representing fermionic and bosonic models and for quantum chemistry workflows. It began as a collaborative project with contributions from Google and many academic and industrial groups, and its published software paper has a large multi-institution author list [1854]. OpenFermion maps electronic-structure and second-quantized problems into representations useful for quantum algorithms. It does not by itself make an arbitrary chemistry algorithm executable on every quantum hardware architecture.

2.3.4 Google

Google’s main open-source gate-model framework remains **Cirq**, a Python library launched in 2018. Cirq 1.7.0 was released in June 2026. The former **Cirq-FT**

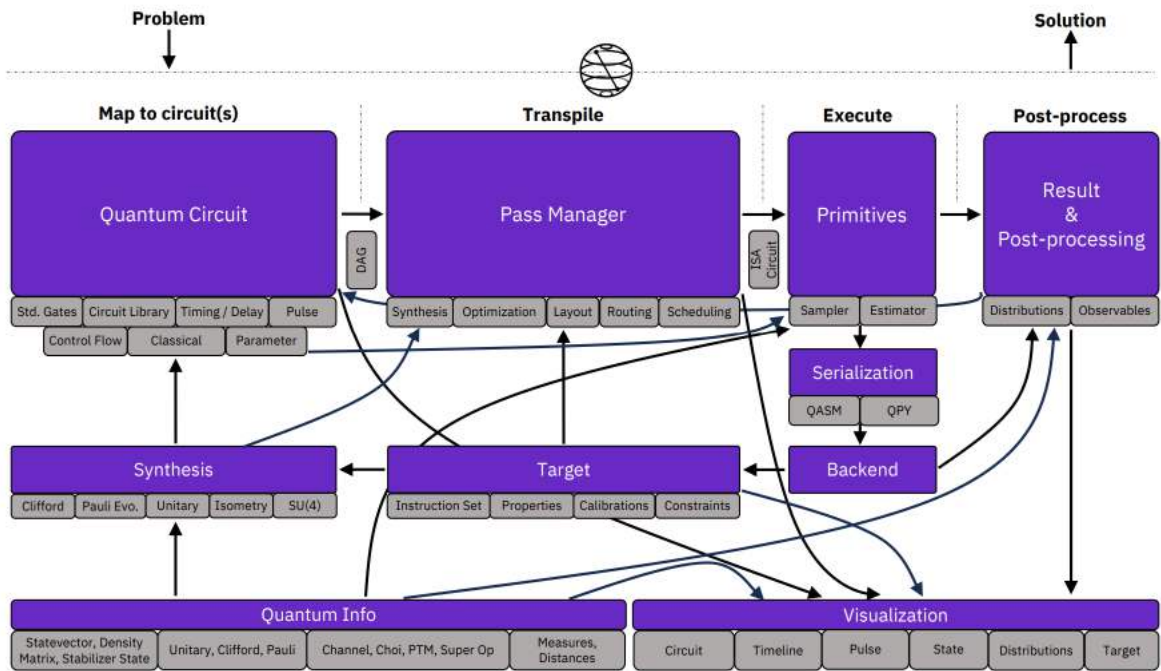


Figure 126: Qiskit architecture from [1843]. The figure predates some of the Qiskit 2.x service and Addon reorganizations described in the text.

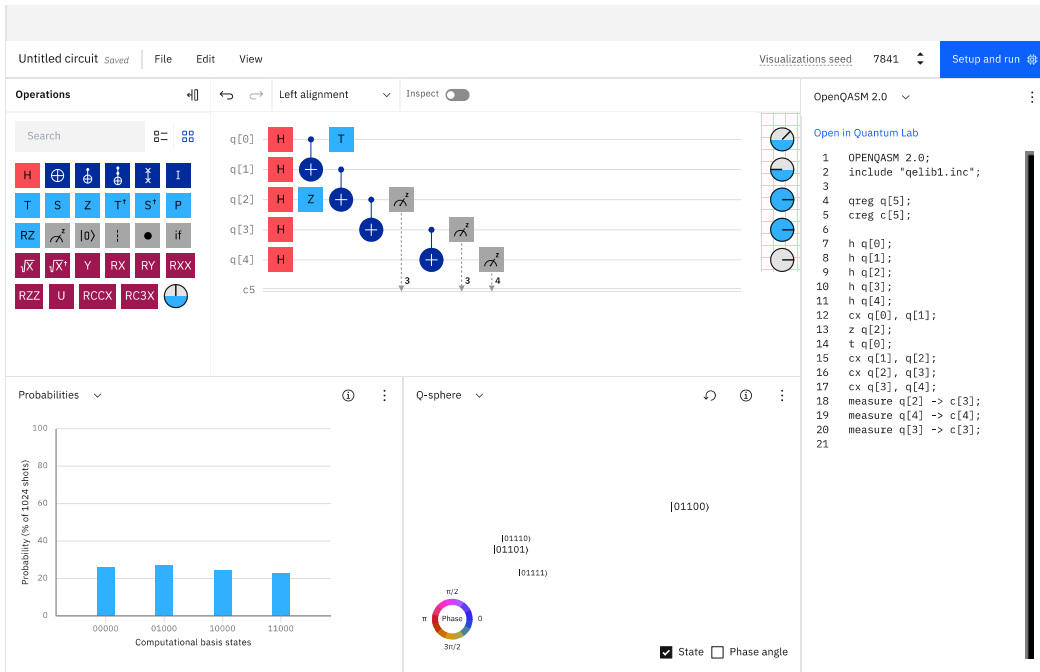


Figure 127: IBM Quantum Composer graphical circuit editor, October 2023. The screenshot is historical, while graphical circuit editing remains part of IBM's web tooling.

package was removed from Cirq after its functionality moved toward **Qualtran**, a separate open-source library for expressing, testing and analyzing fault-tolerant algorithms and their resource requirements [1839]. Current Cirq releases are listed at [GitHub](#).

Google also maintains **qsim**, a state-vector emulator integrated with Cirq. qsim uses single-precision complex amplitudes, so the state vector itself scales as about 8×2^n bytes, or roughly 8 GB at 30 qubits and 16 GB at

31 qubits, before workspace, copies and other overhead. Google continues to maintain **TensorFlow Quantum**, launched in 2020 for hybrid quantum-classical machine-learning research [1856]. Its current release was version 0.7.6 as of September 2026.

Google superconducting processors can now be accessed programmatically through Google's **Quantum Computing Service** and Quantum Engine APIs. This is not a public cloud service comparable to unrestricted

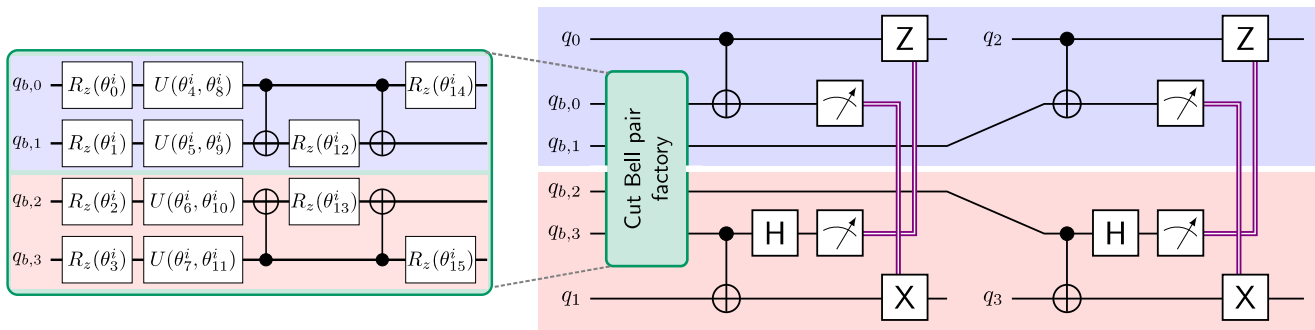


Figure 128: A virtual CNOT construction using local operations and classical communication, cut Bell-pair quasi-probability decompositions, measurement and real-time classical correction. No physical quantum link connects the two QPUs in this experiment. The construction is discussed in [1850] and is related to gate-teleportation methods [1852].

```

from pyquil.gates import X, CNOT, H, Z, RX, I
from pyquil.api import QVMConnection
from pyquil.quil import Program
import numpy as np

qvm = QVMConnection()

alice_register = 0
ancilla_register = 1

flip_correction_branch = Program(X(1))
phase_correction_branch = Program(Z(1))

prog = (Program()
        .inst(H(0))
        .inst(CNOT(0, 1))
        .inst(RX(0.2 * np.pi, 2))
        .inst(CNOT(2, 0))
        .inst(H(2))
        .measure(0, alice_register)
        .measure(2, ancilla_register)
        .if_then(alice_register, flip_correction_branch)
        .if_then(ancilla_register, phase_correction_branch))

qvm.run_and_measure(prog, list(prog.get_qubits()), trials=10)

```

```

H 0
CNOT 0 1
RX(pi/5) 2
CNOT 2 0
H 2
MEASURE 0 [0]
MEASURE 2 [1]
JUMP-WHEN @THEN1 [0]
JUMP @END2
LABEL @THEN1
X 1
LABEL @END2
JUMP-WHEN @THEN5 [1]
JUMP @END6
LABEL @THEN5
Z 1
LABEL @END6

```

Figure 129: Historical pyQuil example and the lower-level Quil program generated from it. Source: [Quantum Cloud Computing](#) by Johannes Otterbach, Rigetti, January 2018.

```

from pyquil.gates import H

# The Hadamard produces what is called a superposition state
coin_program = Program(H(0))
wavefunction = quantum_simulator.wavefunction(coin_program)

print("H|0> = ", wavefunction)
print("With outcome probabilities\n", wavefunction.get_outcome_probs())

```

● — Hadamard gate on qubit 1

```

H|0> = (0.7071067812+0j)|0> + (0.7071067812+0j)|1>
With outcome probabilities
{'0': 0.49999999999999989, '1': 0.49999999999999989}

```

● — readout output

Figure 130: Historical Hadamard-gate example programmed with pyQuil. Source: [pyQuil Documentation Release 1.9.1.dev0](#), Rigetti, July 2018.

commercial access. As of July 2026, hardware access remained restricted to approved groups, generally in collaboration with a Google sponsor. The current access conditions are documented at [Google Quantum AI](#).

FOREST: Tools for experimental quantum programming

forest.rigetti.com

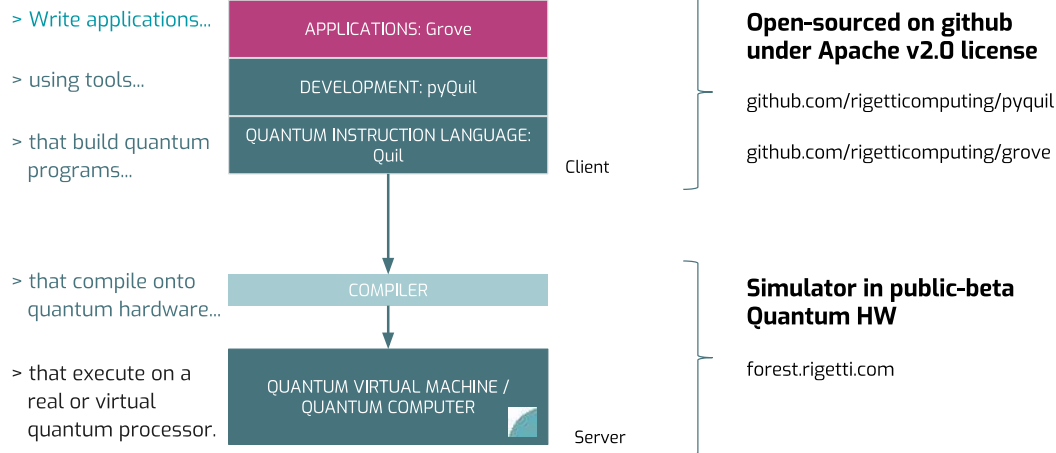


Figure 131: Historical Rigetti Forest software platform. Source: [1855]. Forest has since been superseded as the main branding by Rigetti QCS and the Quil SDK.

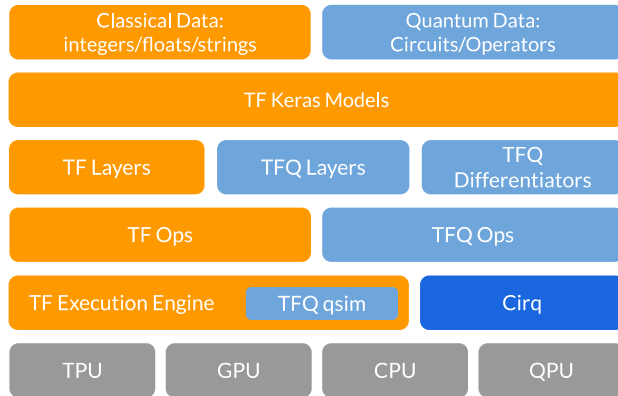


Figure 132: Historical TensorFlow Quantum hybrid quantum-classical software architecture in 2020. Source: [1856].

2.3.5 Microsoft

Microsoft's quantum software is organized around **Azure Quantum**, the **Microsoft Quantum Development Kit** (QDK), **Q#** and the LLVM-based **Quantum Intermediate Representation** (QIR). Microsoft's research on topological qubits is a separate hardware program.

Q# remains Microsoft's open-source quantum programming language [1857]. The modern QDK was substantially rewritten around Rust and WebAssembly in 2023 and 2024. As of August 2026, the QDK VS Code extension directly supports Q# and OpenQASM, including circuit visualization, resource estimation, job submission, code completion and debugging. The QDK Python package supports Q#, OpenQASM, Qiskit and Cirq, with PennyLane interoperability through QIR. Current language support is documented at [Microsoft Learn](https://microsoft.com/learn/quantum).

The QDK includes local simulation, circuit visualization and the **Azure Quantum Resource Estimator**, which estimates logical and physical resources for fault-tolerant algorithms under configurable error-correction assumptions (Figure 133). The estimator's own documented input formats are narrower than the QDK's general language interoperability, currently Q#, Cirq, OpenQASM and QIR. Microsoft also released **QDK/Chemistry** in 2026 for chemistry-oriented workflows [1858]. The current QDK portfolio also includes **QDK-EC**, a software suite for quantum error-correction research, and **QDK Learning** for educational material. The **Quantum Katas** remain educational examples and exercises rather than a core execution layer [1859]. In January 2026, Microsoft added AI-based features in its QDK and its integration with Visual Studio Code and GitHub Copilot.

QIR is an LLVM-based intermediate representation introduced in 2020 to provide a common compiler interface between quantum languages and execution targets. The QIR Alliance was formed in 2021 and is hosted by the Linux Foundation [1861]. QIR is used by multiple compilers, SDKs and hardware integrations. Extensions and alternatives for distributed quantum computing, including NetQIR, continue to be studied [1862, 1863].

Azure Quantum currently provides access to QPUs and simulators from **IonQ**, **Pasqal**, **Quantinuum** and **Rigetti**. In April 2026, listed hardware included IonQ Forte systems with 36 qubits, Pasqal Fresnel with 100 atoms, Quantinuum H2 targets up to 32 qubits, and Rigetti Cepheus with 108 qubits. These are service snapshots and can change. Current targets are listed at [Microsoft Learn](https://microsoft.com/learn/quantum). Pricing is provider-specific and changes over time. The old fixed IonQ figures of \$0.00003 for a one-qubit gate and \$0.0003 for a two-qubit gate are

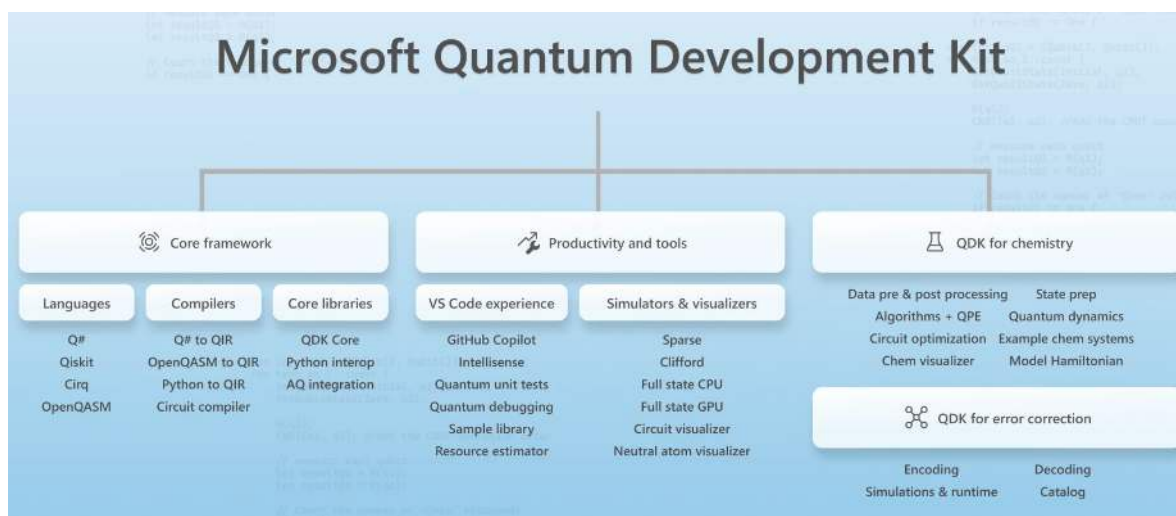


Figure 133: Microsoft QDK structure as of September 2026. Source: [1860].

no longer a reliable description. Azure’s current IonQ offering uses a provider-defined token pricing model.

Azure Quantum Elements remains Microsoft’s chemistry and materials platform. It combines HPC, AI and computational chemistry services and can incorporate quantum-computing workflows where appropriate. It should not be described simply as a classical implementation of quantum chemistry.

2.3.6 Amazon

Amazon’s quantum software offering is organized around **Amazon Braket**. The service provides a Python SDK, managed notebooks, local and on-demand simulators, QPU access, **Hybrid Jobs** for managed hybrid workloads, and **Braket Direct** for reservations and direct access workflows [1864]. Braket also supports OpenQASM 3 and analog Hamiltonian simulation for hardware such as QuEra’s neutral-atom system.

As of September 2026, Amazon Braket proposes QPU access from **AQT**, **IonQ**, **IQM**, **QuEra** and **Rigetti**. The active device list includes AQT IBEX-Q1, IonQ Forte systems, IQM Garnet and Emerald, QuEra Aquila, Rigetti Ankaa-3 and the 108-qubit Rigetti Cepheus-1-108Q. OQC is no longer in the current Braket device list. AWS also added native CUDA-Q support to Braket notebook instances in 2025 and Python 3.12 support for Braket notebooks and managed Base, CUDA-Q, PennyLane and TensorFlow containers in 2026. The live device inventory is maintained at [Amazon Braket documentation](#).

AWS published an example in 2024 showing how users could build a Braket-oriented coding assistant with Claude and retrieval-augmented generation [1865]. This was an example architecture that developers could assemble, not a distinct first-party “AWS Braket Code Assistant” product in the same sense as the Braket SDK or Hybrid Jobs.

2.3.7 IonQ

IonQ provides direct access through the **IonQ Quantum Cloud**, its REST API and a set of SDK integrations. The current developer documentation supports **Qiskit** through `qiskit-ionq`, **Cirq** through `cirq-ionq`, **PennyLane**, `qBraid` and **CUDA-Q**. IonQ also maintains a low-level typed Python client, `ionq-core`, for direct access to its cloud API. The SDK documentation is available at docs.ionq.com.

IonQ introduced a rewritten **Quantum OS** and a **Hybrid Services** suite in December 2024. Hybrid Services includes workload management, a solver service, sessions for iterative workloads and an IonQ SDK for integration with the IonQ Quantum Cloud API. IonQ hardware can also be accessed through cloud partners including Amazon Braket and Azure Quantum.

2.3.8 Intel

Intel’s current software offering is primarily a simulation and compiler toolchain rather than a generally available cloud service attached to Intel production QPUs. The **Intel Quantum SDK** is a C++-oriented full stack in simulation, with a compiler, runtime and multiple simulator backends. The current release is version 1.1.1. Version 1.1 added the QExpr representation and FLEQ functional-language extensions, compiler pass control, qubit mapping and scheduling. Current release notes are available at [Intel’s Quantum SDK page](#).

The open-source **Intel Quantum Simulator** (IQS), formerly qHiPSTER, is a distributed-memory high-performance simulator written in C++ with optional Python bindings. Published benchmarks included simulations up to 42 qubits on HPC systems [1866]. As with all dense state-vector simulators, attainable qubit count depends mainly on aggregate memory and communication bandwidth, with state-vector memory growing as $O(2^n)$.

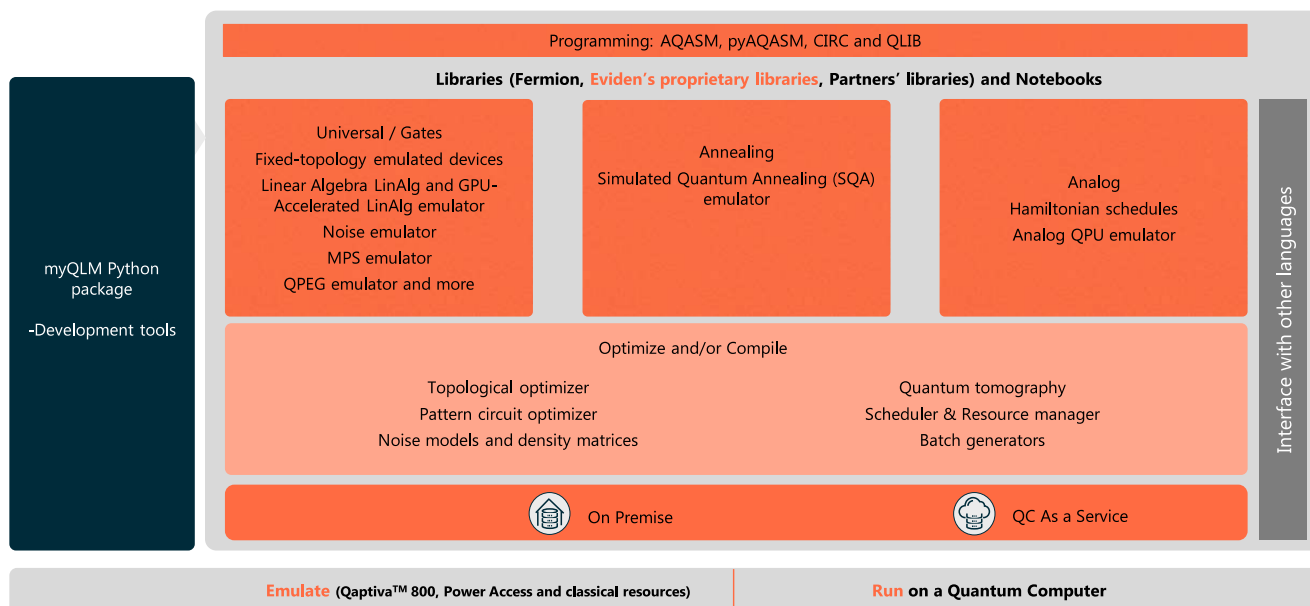


Figure 134: Bull software platform. Source: [1867]. The diagram predates Qaptiva HPC, launched in November 2024, and the current Qaptiva portfolio described in the text.

2.3.9 Huawei

Huawei’s **HiQ** cloud simulator, launched in 2018 [1694], is now discontinued. The actively maintained open-source framework associated with Huawei’s MindSpore ecosystem is **MindQuantum**. Version 0.12.0 is the current documented release as of September 2026. It provides circuit construction, quantum simulation, compilation and mapping tools, variational quantum algorithms and integration with MindSpore for hybrid quantum-classical machine learning.

MindQuantum 0.12 added a beta abstract **QPU** class for real-hardware execution. Hardware providers implement a sampling interface while MindQuantum supplies shot-based expectation-value and gradient estimation. This is an important evolution beyond the older HiQ simulator-only description. Current release notes are available [1868]. Simulator qubit counts depend on the simulation method, circuit depth, entanglement and available memory.

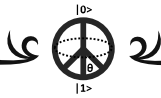
2.3.10 Bull

Bull comprises the former Atos Advanced Computing activities, which operated under the Eviden brand before their acquisition by the French State in March 2026. Its quantum software and emulation portfolio is branded **Qaptiva**. The current portfolio includes the **Qaptiva Quantum Emulation Appliance**, **Qaptiva**

HPC, **Qaptiva Access Server**, the free **myQLM** Python environment, and consulting and training services. Qaptiva Access Server acts as an orchestration hub between emulators, QPUs and HPC resources. The current portfolio is described at [Eviden’s Qaptiva pages](#).

Qaptiva HPC, launched in November 2024, runs the emulation stack on distributed HPC clusters rather than on a dedicated Qaptiva appliance. Eviden reported state-vector emulations of up to 43 qubits in the French HQI infrastructure and describes the product as supporting more than 40 qubits on suitable HPC resources. This is a dense-emulation result which can’t be compared with the much larger qubit counts mentioned for tensor-network simulations. In 2023, Eviden announced tensor-network emulation beyond 100 qubits for some NISQ circuits. Such a qubit count is highly circuit-dependent because tensor-network cost is controlled by entanglement structure, graph topology and contraction complexity, not by qubit count alone.

The older Atos QLM and Qaptiva 800 names remain relevant historically. The programming environment retains QLM-derived Python APIs and AQASM concepts, while **Q-Pragma** provides a C++ pragma-based approach for hybrid quantum-classical programs [1869]. Bull’s compiler work also includes architecture-aware transformations such as Lazy Synthesis [1597]. Current Qaptiva positioning is broader than the former description based only on AQASM, PyAQASM and a dedicated appliance.



2.3.11 Bibliography and notes

These are the bibliographical references and notes for the **Industry Vendors development tools** subsection.

- [1597] [Architecture aware compilation of quantum circuits via lazy synthesis](#), by Simon Martiel and Timothée Goubault de Brugière, arXiv, June 2022 (32 pages).
- [1694] [Huawei Unveils Quantum Computing Simulation HiQ Cloud Service Platform](#), by Huawei, Huawei, October 2018.
- [1832] [Overview and Comparison of Gate Level Quantum Software Platforms](#), by Ryan LaRose, arXiv, March 2019 (24 pages).
- [1833] [Quantum Computing languages landscape](#), by Alba Cervera-Lierta of the Quantum World Association, Medium, September 2018.
- [1834] [Productive Quantum Programming Needs Better Abstract Machines](#), by Santiago Núñez-Corrales, Olivia Di Matteo, John Dumbell, Marcus Edwards, Edoardo Giusto, Scott Pakin, and Vlad Stirbu, arXiv, April 2025 (11 pages).
- [1835] [Qwerty: A Basis-Oriented Quantum Programming Language](#), by Austin J. Adams, Sharjeel Khan, Jeffrey S. Young, and Thomas M. Conte, arXiv, April 2024 (30 pages).
- [1836] [ASDF: A Compiler for Qwerty, a Basis-Oriented Quantum Programming Language](#), by Austin J. Adams, Thomas M. Conte, et al., arXiv, January 2025 (20 pages).
- [1837] [Quantum types: going beyond qubits and quantum gates](#), by Tamás Varga, Yaiza Aragonés-Soria, and Manuel Oriol, arXiv, August 2024 (4 pages).
- [1838] [Automated Quantum Algorithm Synthesis](#), by Amy Rouillard, Matt Lourens, and Francesco Petruccione, arXiv, March 2025 (9 pages).
- [1839] [Expressing and Analyzing Quantum Algorithms with Qualtran](#), by Matthew P. Harrigan, Ryan Babbush, Nicholas C. Rubin, et al., arXiv, September 2024 (39 pages).
- [1840] [Adiabatic quantum computation](#), by Tameem Albash and Daniel A. Lidar, Reviews of Modern Physics, January 2018 (64 pages)  (preprint on [arXiv](#)).
- [1841] [Open Quantum Assembly Language](#), by Andrew W. Cross, Jay M. Gambetta, et al., arXiv, July 2017 (24 pages).
- [1842] [OpenQASM 3: A broader and deeper quantum assembly language](#), by Andrew W. Cross, Jay M. Gambetta, Blake R. Johnson, et al., arXiv, March 2022 (60 pages).
- [1843] [Quantum computing with Qiskit](#), by Ali Javadi-Abhari, Simon Martiel, Lev S. Bishop, Andrew W. Cross, Blake R. Johnson, Jay M. Gambetta, et al., arXiv, June 2024 (19 pages).
- [1844] [IBM's roadmap for building an open quantum software ecosystem](#), by Karl Wehden et al., February 2021.
- [1845] [Qiskit Machine Learning: an open-source library for quantum machine learning tasks at scale on quantum hardware and classical simulators](#), by M. Emre Sahin, Edoardo Altamura, Oscar Wallis, Stephen P. Wood, Anton Dekusar, Declan A. Millar, Takashi Imamichi, Atsushi Matsuo, and Stefano Mensa, arXiv, May 2025 (6 pages).
- [1846] [Quantum resources in resource management systems](#), by Iskandar Sitdikov, Munetaka Ohtani, et al., arXiv, June 2025 (7 pages).
- [1847] [Qiskit Code Assistant: Training LLMs for generating Quantum Computing Code](#), by Nicolas Dupuis, Luca Buratti, Sanjay Vishwakarma, Aitana Viudes Forrat, David Kremer, Ismael Faro, Ruchir Puri, and Juan Cruz-Benito, arXiv, May 2024 (4 pages).
- [1848] [Introducing Qiskit Code Assistant](#), by Daniella Garcia Almeida et al., October 2024.
- [1849] [Qiskit HumanEval: An Evaluation Benchmark For Quantum Code Generative Models](#), by Sanjay Vishwakarma, Francis Harkins, Siddharth Golecha, Vishal Sharathchandra Bajpe, Nicolas Dupuis, Luca Buratti, David Kremer, Ismael Faro, Ruchir Puri, and Juan Cruz-Benito, arXiv, June 2024 (8 pages).
- [1850] [Combining quantum processors with real-time classical communication](#), by Almudena Carrera Vazquez, Caroline Tornow, Diego Ristè, Stefan Woerner, Maika Takita, and Daniel J. Egger, Nature, November 2024.
- [1851] [Randomized Benchmarking Protocol for Dynamic Circuits](#), by Liran Shirizly, Luke C. G. Govia, and David C. McKay, arXiv, August 2024 (13 pages).
- [1852] [Quantum Teleportation is a Universal Computational Primitive](#), by Daniel Gottesman and Isaac L. Chuang, arXiv, August 1999 (6 pages).
- [1853] [A Practical Quantum Instruction Set Architecture](#), by Robert S. Smith, Michael J. Curtis, and William J. Zeng, arXiv, February 2017 (15 pages).
- [1854] [OpenFermion: The electronic structure package for quantum computers](#), by Jarrod R. McClean, Ryan Babbush, et al., Quantum Science and Technology, June 2020 (22 pages)  (preprint on [arXiv](#)).
- [1855] [Quantum Cloud Computing](#), by Johannes Otterbach, Rigetti, January 2018 (105 pages).
- [1856] [TensorFlow Quantum: A Software Framework for Quantum Machine Learning](#), by M Broughton, arXiv, 2020 (39 pages).
- [1857] [Q#: Enabling scalable quantum computing and development with a high-level domain-specific language](#), by Krysta M. Svore, Martin Roetteler, et al., arXiv, March 2018 (11 pages).
- [1858] [QDK/Chemistry: A Modular Toolkit for Quantum Chemistry Applications](#), by Nathan A. Baker, Boqin Zhang, et al., arXiv, January 2026 (32 pages).
- [1859] [Learn at your own pace with Microsoft Quantum Katas](#), by Microsoft, Microsoft, July 2018.
- [1860] [Powerful new developer tools increase the versatility of the Microsoft Quantum platform](#), by Matthias Troyer, Microsoft Azure Quantum Blog, January 2026.
- [1861] [New Quantum Intermediate Representation Alliance Serves as Common Interface for Quantum Computing Development](#), by Linux Foundation, November 2021.
- [1862] [NetQIR: An Extension of QIR for Distributed Quantum Computing](#), by F. Javier Cardama, Jorge Vázquez-Pérez, César Piñeiro, Tomás F. Pena, Juan C. Pichel, and Andrés Gómez, arXiv, April 2025 (16 pages).
- [1863] [Review of intermediate representations for quantum computing - The Journal of Supercomputing](#), by F. Javier Cardama, Jorge Vázquez-Pérez, César Piñeiro, Juan C. Pichel, Tomás F. Pena, and Andrés Gómez, SpringerLink, January 2025 (21 pages).
- [1864] [Amazon Braket launches Braket Direct, a program to dive deeper into quantum computing](#), by AWS, November 2023.
- [1865] [Accelerate quantum software development on Amazon Braket with Claude-3](#), by Yaroslav Kharkov, September 2024.
- [1866] [Intel Quantum Simulator: A cloud-ready high-performance simulator of quantum circuits](#), by Gian Giacomo Guerreschi, Justin Hogaboam, Fabio Baruffa, and Nicolas P. D. Sawaya, Quantum Science and Technology, May 2020 (14 pages).
- [1867] [Towards a High Performance Hybrid Computing](#), by Julien Mellaerts, December 2023 (33 slides).

- [1868] [MindQuantum 0.12 documentation](#), by MindSpore Quantum, MindSpore Quantum documentation, May 2026.
- [1869] [A pragma based C++ framework for hybrid quantum/classical computation](#), by Arnaud Gazda and Oceane Koska, arXiv, March 2024 (36 pages).

This **Industry Vendors development tools** subsection contains 40 bibliographical references and notes containing at least 691 pages.

2.4 Quantum software engineering

Quantum software engineering is an emerging discipline encompassing the whole middleware and software stacks involved in the creation and deployment of quantum software [1870].

It contains languages and framework designs, compilers, circuit synthesis, transpilers and optimizers, debugging methods and tools, software certification and verification, code porting and refactoring, how to secure quantum computing operations, and how to design optimized quantum software development workflows [1871].

It also embeds the various use cases of machine learning in quantum computing. We cover here the topics not already handled in the above parts of this long laundry list.

The field’s open-source base is itself growing and evolving. A recent empirical study of foundational quantum software repositories on GitHub found that most concepts do not stay confined to their originating repository: contributors migrate across projects and carry ideas with them, with some repositories evolving into sustained communities while others fold their contributions into larger established ecosystems [1872].

The software community also organizes itself to influence research policy, which is still dominated by hardware funding in most places in the world. The **Quantum Software Alliance** (QSA), led by the University of Edinburgh and launched in February 2026 by 21 academic centers with about 26 members by September 2026, published a ten-year Global Research Agenda (2026-2036) covering algorithms, compilers, verification and benchmarking, fault tolerance with hybrid HPC integration, and distributed computing (quantumsoftwarealliance.org).

2.4.1 Certification and verification

The verification and certification of quantum algorithms and the result of their use is an important topic [1873]. The factorization of integer numbers is obviously easy to verify. But when a quantum algorithm is used to simulate physical interactions such as those of atoms in molecules, it is less obvious. This is the field of Quantum characterization, verification, and validation (QCVV), a set of techniques to probe, describe, and assess the behavior of quantum information processing devices [1874].

A recent illustration combines an FPGA accelerator with LLM-guided parameter search to certify entanglement through CHSH inequality violation on a Qiskit simulator, reaching 99.94% of the theoretical bound of $2\sqrt{2}$, though the demonstration remains confined to ideal simulation and has yet to be tried on real NISQ hardware [1875].

Theoretical work shows that it is possible to prove in polynomial time that the result of a quantum algorithm

is accurate [1876, 1877]. Unfortunately, we cannot explain in detail the origin of the result by breaking it down. Nor can we prove that the result found, however valid it may be, is the best of all if there are several good ones. On top of that, we must make a distinction between error corrected hardware (FTQC) and noisy systems (NISQ). Surprisingly, while FTQC regimes will be inaccessible to classical hardware emulation and make verification difficult, verification is also complicated for noisy systems, particularly when they repeat some sequence of code iteratively driven by hybrid variational algorithms. At this point, quantum verification is out of feasibility, requiring either a very large overhead in qubit numbers [1878] or some quantum network connecting two QPUs, without some qubit overhead but using a quantum communication link between a client QPU and a server QPU [1879].

The other key point is to make sure, when using a remote quantum computer, that the recovered result corresponds to the submitted calculation and that an intruder did not interfere in the history nor was able to alter the calculation on the quantum computer side. Many research inroads have been made in the last 15 years for that respect. One of the methods consists in relying on the concept of **blind quantum computing** devised in 2009 by Anne Broadbent, Joseph Fitzsimons and Elham Kashefi [1880, 1881] and the related **Secure Delegated Quantum Computing** [1882, 1883]. Another method consists in using QPU fingerprinting to make sure that the QPU you are using is the right one [1884].

A useful entry point into this whole family of protocols is a 2026 synthesis that breaks verification down into three separable building blocks (remote state preparation, trap-based deviation detection, and error-correcting embedding) and tracks how successive protocols have relaxed their hardware requirements, moving away from single-photon sources toward implementations closer to near-term devices [1885].

CEA LIST announced in June 2020 that it had created **QBRICKS**, an environment for the specification, programming and formal verification of quantum algorithms. They used to do this for critical embedded systems where certification by formal proof is particularly important. They are now entering the field of quantum programming and have experimented their model with the quantum phase estimation algorithm (QPE, which feeds into Shor’s algorithm for integer factorization), the full Shor algorithm, and Grover’s algorithm [1886, 1887].

One of the major advances in the explainability of quantum algorithms comes from researcher **Urmila Mahadev**, whose work between 2012 and 2018 has led to the creation of a method for verifying quantum computer processing. She was a postdoc at Berkeley and supported by Scott Aaronson and Umesh Vazirani, two eminent researchers in quantum algorithmic research. Her work is about proving that a quantum computer

performs the tasks it was asked to do. She shows that a classical computer coupled to a simple quantum computer can verify in a polynomial way the results of a quantum computer [1888–1890]. The method exploits a technique of post-quantum cryptography that the verifier cannot break (LWE: Learning With Errors). LWEs are part of the Lattice-based cryptography (EN) or Euclidean networks (FR) class. The method was recently improved by an Austria team to work with an untrusted quantum computer [1891]. There is a limit to how far this line of work can go: a 2026 paper shows that Mahadev’s four-message protocol cannot be compressed into a single-message, non-interactive scheme based on LWE or any other falsifiable assumption, which sets a hard floor under the message complexity of classical verification of quantum computation [1892].

Other quantum programs verifiers [1893] from research laboratories include the **Path-sum** framework from the University of Waterloo [1894], **VOQC** (Verified Optimizer for Quantum Circuits) from the University of Maryland, itself based on **SQIR** (Small Quantum Intermediate Representation) supporting verification [1895] and **QHL** from Tsinghua University [1896].

A related task, circuit equivalence checking, used to validate that a compiler’s optimizations preserved the original circuit’s behavior, is addressed by **Manjushri**, a tool presented in 2026 using weighted binary decision diagrams as compact circuit fingerprints. With random Clifford+T circuits, it is about 8 to 10 times faster than the competing ECMC tool up to circuit depth-30, and stays reliable up to depth-38 for circuits with 128 qubits, though its success rate vanishes beyond that depth [1897].

2.4.2 Testing and debugging

Any computer software and quantum software require a set of quality control processes. Like most human-originated creations, they are prone to bugs and errors. Some classical computing methods can be reused for this respect, but many require some adaptation to the specifics of quantum computing, whether done on gate-based systems or on analog quantum simulators [1898, 1899].

A quantum circuit is not easy to debug! It will certainly require new debugging tools and approaches. And most quantum code bugs are “quantum” in nature and not easy to spot with traditional methods [1901].

We now have some inventory of the sources of common bugs in quantum circuits [1902] (Figure 137).

Code slicing is one proposed debugging method, as shown in Figure 135 [1903]. It consists in dividing the quantum circuits in chunks that are analyzed sequentially with running the circuit until chunk 1, then 2, 3 and so on, and doing sampling based measurement at each step, providing the developer knows what he’s looking for in these intermediate steps in his code. Then

you need to analyze your amplitude vector, and in some case, with under 6 qubits, your density matrix. Hopefully, most quantum code is highly “patternized” with streams of unitary encoding, modular exponentiations, QFT and inverse QFT, amplitude amplification, and so on.

Category	Count
Compatibility	3,181
Functional	2,445
Quantum-Specific Issue	1,550
Usability	1,471
Syntax Errors	1,324
Miscellaneous	1,076
Logical	548
Error Handling Defects	270
Integration Level	229
Performance	219
Critical Defects	189
Runtime Error	103
Calculation Errors	83
Out of Bound	75
Security	59
Boundary Related Errors	46
Communication Errors	42

Figure 137: Distribution of Bug Categories in the Automatically Classified Dataset. Source: [1902].

For the moment, simple circuits can be analyzed and debugged with a quantum emulator running on a classical computer, to understand how the qubit register vector state evolves step-by-step. Circuit emulators can however generate their own bugs for various reasons, mostly linked to the classical emulator infrastructure [1904].

But when quantum circuits are in the advantage regime, beyond any classical emulation capacity, other means will have to be used, like with the Proq and Bloq methods [1905] which compute expectation value of Pauli operators but may not easily scale well for large circuits. Could LLMs come to the rescue? As of 2026, some pilot projects are testing their efficiency [1906].

2.4.3 Verification and validation

Software quality control usually goes through two main steps: verification and validation (Figure 136).

Verification deals with verifying that the code will run as expected. It includes checking code documentation, designs, circuits and the various software components or patterns that are used. Verification also deals with making sure that the specifications are correctly implemented by the system. It responds to the question: are we building the product right?

In classical programming, good programmers and code reviewers can spot an error by just looking at the code. Code inspection tools can also detect undeclared variables or variables used in the wrong context. These errors are way more difficult to detect visually on a quantum circuit, particularly with large ones. It may require the use of code decomposition in modules or patterns, like in object-oriented programming, and of

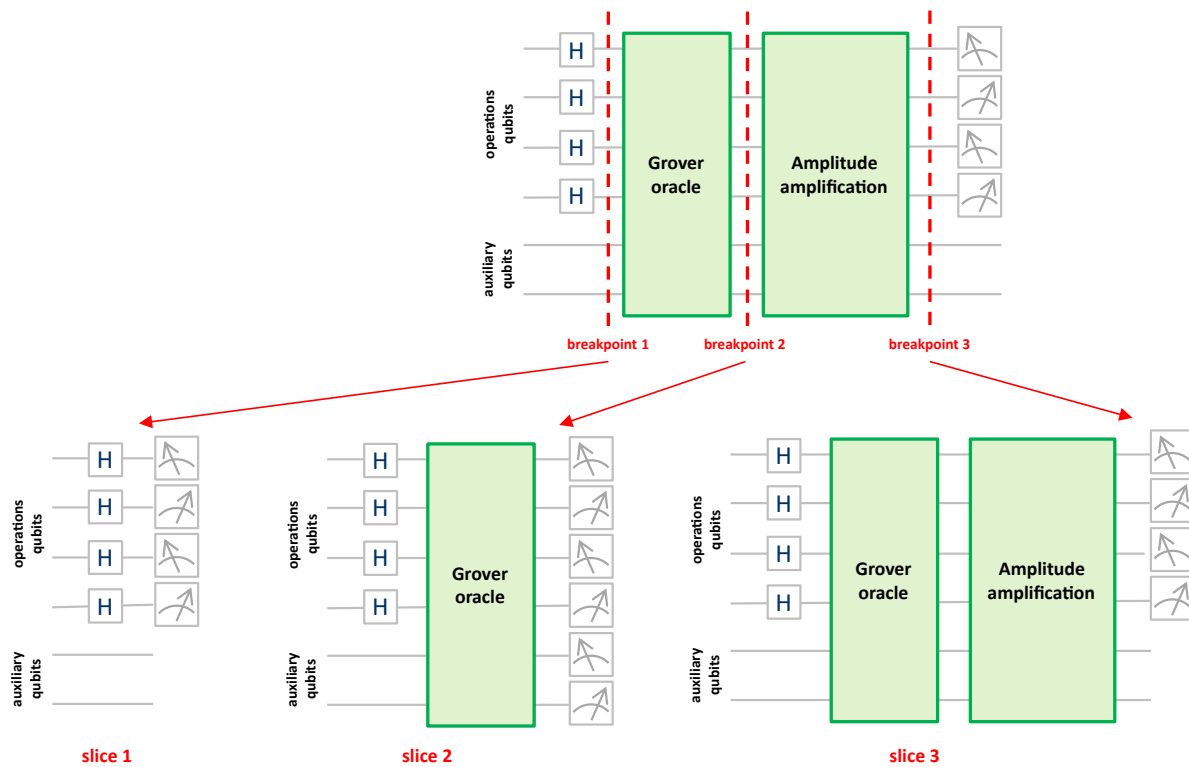


Figure 135: A quantum code debugging approach with code slicing taking the example of a Grover oracle and amplification cycle. In classical computing and with interpreted languages like JavaScript or PHP, we can use breakpoints and continue. With quantum computing, we use breakpoints and must restart all computing to stop one step later, depicted here, which can be costly since the circuit will be executed several times and its output classically post-processed or averaged. In the case of Grover’s algorithm to find a single solution, the amplification effect will be easy to observe when sampling the output of the circuit after each amplification cycle, showing one value of the state vector that will be growing at the expense of all others. Source: [1900].

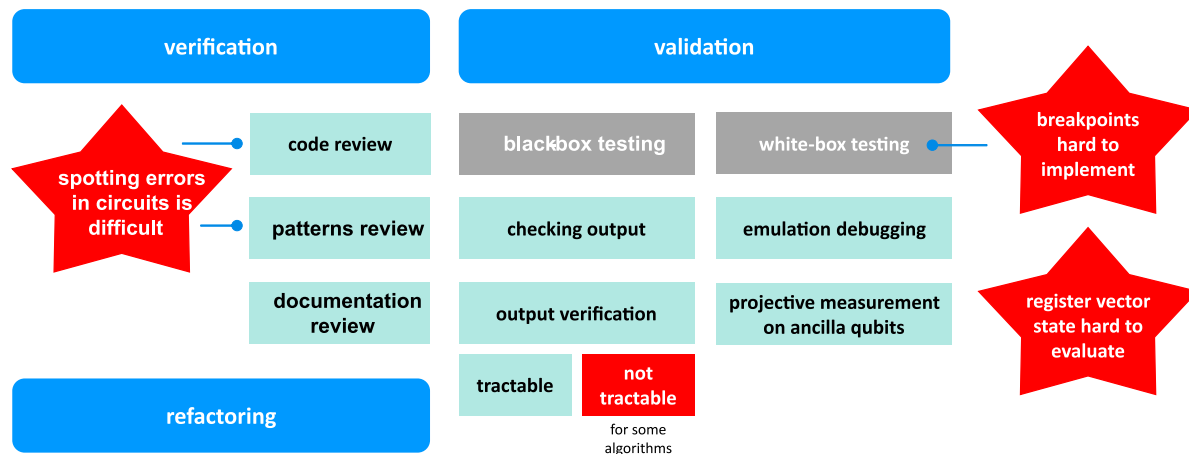


Figure 136: Some of the challenges with quantum software engineering. © Olivier Ezratty, 2022-2023.

recursively defined quantum circuits, which happen for most circuits designed for a large number of physical or logical qubits [1907]. There are also proposals for testing quantum code with treating program variables as symbolic variables. This is called “concolic testing” [1908].

Validation concerns the program output and making sure it works as planned. It includes testing and validating the code against the user’s needs. It responds to the question: are we building the right product?

Quantum computing results validation is usually fast like with integer factoring (it requires a simple classical multiplication) or a Grover search (it requires checking the Oracle once in a classical way). Another way to handle validation is to run a complete or partial classical simulation of the quantum experiment or circuit up to a certain number of qubits, or photon modes in the case of gaussian boson samplings [1909].

But some circuit validations may need to be done on a quantum computer, like with a boson sampling or with a

QMA prover (Quantum-Merlin-Arthur). On top of that, contrarily to classical computers, quantum computations errors can also come from hardware imperfections and the fateful quantum noise [1910]. Compilers, code optimizers and even error correction codes can also generate software bugs and amplify some errors.

Quantum software bugs can have various sources: errors in the data preparation (which is itself based on quantum gates), incorrect operations and transformations, incorrect compositions and iterations and incorrect qubits deallocations (or “uncomputations”).

During validation, testing uses the white-box and black-box approaches.

White box testing tests internal data structures and program flow and may include some interactive debugging. One solution is to decompose manually or automatically the quantum code into accumulated slices of code with its incremental different parts like data preparation, oracle execution and amplitude amplifications in the case of a Grover algorithm, as shown in Figure 135. The debugging tool will run each accumulated slice followed by a measurement one after the other [1900]. It is not a classical pause-play like in classical interpreters like JavaScript but “pause” and “play again from the start”, probably with several shots being run and their results averaged to get a sampled output.

Black-box testing looks at the functionality, ignoring the inner workings of the software, making sure the expected output is obtained with a given input. How about using interactive debugging in the white-box approach? Right now, it can be done on quantum emulators but is limited by their computing/memory capacity.

But good measurement, just to get a state in the computational basis would require running the code many times. And even way more times if we’d need to reconstitute the full vector state. Then, to move to the next series of gates, the circuit would have to be re-run the same number of times. And again and again and again. It would be worse if we were to check the entanglement within the register. Deciding if a register is separable is an NP-hard problem. One way to proceed is to implement unit testing by splitting the code into trusted blocks and patterns [1911, 1912].

Other debugging tools can involve projective measurements on ancilla qubits or even gentle measurement techniques [1913, 1914]. And this deals just with classic gate-based quantum computing. Analog quantum computing and special techniques like MBQC or FBQC (from PsiQuantum) will mandate specific debugging techniques and tools.

Quokka# is an open-source Python library for quantum circuit analysis released in 2026 which contains a circuit validation step [1915].

Finally, let’s mention that many quantum algorithms are hybrid and aggregate classical and quantum algorithms, which requires another set of discipline and tools.

Research is going on in all these dimensions around the world. These are strategic components for quantum computing.

2.4.4 Code porting

Some developers are asking whether classical computing code could be ported to run on a quantum computer. I suspect it does not make any sense at the programming language and framework levels given the handles objects are quite different. But why not at a high abstraction level, when using libraries that could be implemented in both classical and quantum implementations. It is what Horizon Quantum is offering with its toolkits. In that case, we would be close in both cases to the notion of no code programming. So, ...no code porting!

2.4.5 Code refactoring

Code refactoring is a technique consisting in modifying a program with making it more readable, better structured, removing clutter and making it less prone to bugs.

The technique is used in classical programming and could also be used in quantum code development. The underlying optimizations are different from the ones generated by compilers, transpilers and optimizers since it modifies the code itself instead of transforming it into a code more readily usable for the QPU itself in a one-way process [1916] (Figure 138).

The category can also include code clones detections [1917], which become commonplace as quantum developers reuse and modify smaller code segments or software patterns. This could be alleviated with more modular and adaptable code structures in quantum development tools, higher abstraction levers. It is probable that there are also lots of code clones in the quantum circuits classical pre- and post-processing tasks.

2.4.6 Securing quantum computing

Securing quantum computing infrastructures is an emerging discipline. But what are the threats? They seem numerous, at least, as soon as a QPU is a shared resource in the cloud, as shown in Figure 139. They are still very theoretical.

A first one, when using QPUs on the cloud is, how to be sure that your circuit is executed in a real QPU and not in some emulator? That’s a question if you are not in a quantum advantage regime. This is called a proof of quantumness (PoQ), with various proposed techniques [1918].

For example, side channel attacks and fault injection techniques (*aka* jailbreaking) could be physically perpetrated on a QPU with analyzing the pulse control signals send to a quantum processor and reverse engineered to capture the related quantum code [1919–

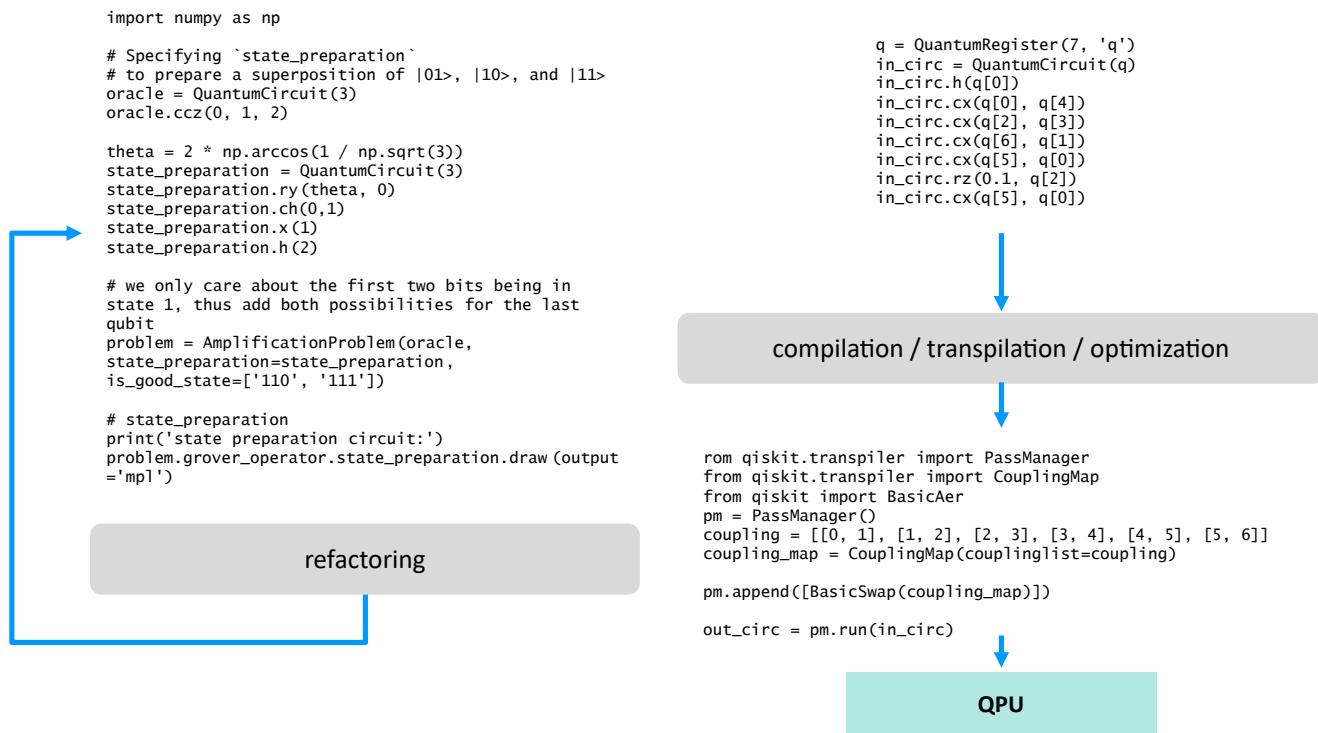


Figure 138: Difference between code refactoring and code compilation, transpilation, and optimization. © Olivier Ezratty, 2023-2025.

[1927]. This would require some malicious capacity to hook hardware devices on the various parts of a QPU. Seemingly, these QPUs will not be easy to access in cloud infrastructures. Other attacks involve cross-talk, residual coupling operations on one qubit which disturb its neighbors [1928]. Spying could be perpetrated in a much simpler way by capturing the quantum code itself while being transmitted to the cloud operator.

A newer attack angle targets quantum error correction rather than the qubits. In FTQC regimes, the error syndrome stream sent to the classical decoder contains recognizable “gate fingerprints”, so an eavesdropper on the decoder link can infer which logical circuit ran without touching the processor [1929]. On real hardware, these building blocks can also be chained together. One demonstration on a trapped-ion machine combined side-channel reconnaissance, crosstalk characterization, and adversarial example crafting into a single end-to-end attack against a deployed quantum neural network, with a parallel run on superconducting hardware confirming the chain is not platform-specific [1930].

Other attacks can involve some malicious untrusted compiler or malicious attacks of QML models [1932–1938], defended against with purification or clustering-based detectors [1939, 1940], or even on classical machine learning models used in quantum error correction [1941] and readout discrimination [1942]. Untrusted compilers could go as far as stealing chip manufacturing intellectual property and know-how [1943]!

Weird techniques could allow fingerprinting or watermarking cloud QPUs and enable their identification for

subsequent malicious usage [1944–1946], an idea now extended to circuits produced by generative models themselves so their ownership survives optimization, editing, or redistribution [1947]. Spying techniques could try to measure the value of qubits that would be improperly reset after a shot run by another customer, or when several shots are mutualized in a single QPU [1948]. Indeed, a given QPU could potentially be shared to run several distinct and disconnected quantum circuits simultaneously given it may become scarce resources. This is called “multi-tenant computing” (MTC). In such settings, there are also crosstalk attacks which “*can enable malicious actors to interfere with neighboring quantum circuits, leading to faulty computations, data leakage, or program disruption*” [1949–1952]. From a security standpoint, it looks like a nightmare. So, some researchers are looking at ways to partition these QPUs in a secure way, like with the Fair and Reliable Partitioning (FRP) method [1297, 1953, 1954].

Other concerns exist on the way to secure quantum machine learning algorithms against data-based attacks [1955, 1956] including adversarial attacks [1957–1959] and model reverse engineering [1960]. Malicious attacks could even potentially happen on your code in compilers with Trojan insertions [1961], particularly when these are used through various third party services.

Thus, proposals like code obfuscation techniques using random circuits and their uncomputation [1962–1965], a split compilation technique [1966], and even related to pulse controls [1967] as well as qubit error fingerprinting [1968].

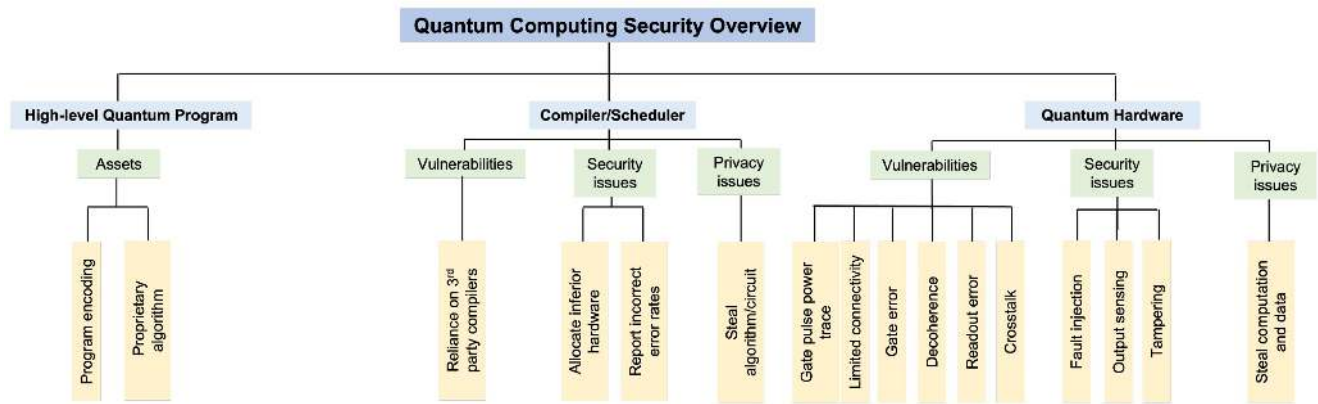


Figure 139: Overview of various threats affecting the security of your quantum code. Source: [1931].

These are fundamental research attacks which may not be practical at all. One key defense can rely on quantum blind computing as we'll see later in the quantum telecom part of this book.

2.4.7 Machine learning in quantum computing

If you search for “machine learning” and “deep learning” in this book, you will find plenty references. You have either “quantum machine learning” which deals with quantum algorithms putting in place some machine learning techniques, and “classical machine learning” which is used elsewhere [1969], like in quantum physics [1970].

Let's sort this out by creating four main categories related to classical machine learning use cases [1971] in the context of quantum computing, as shown in Figure 140 [1969]. Two recent surveys cover this ground in more depth than we can here: one reviews the interface between AI and quantum information from both directions [1972], and another, focused specifically on reinforcement learning, surveys its use in state preparation, gate design, circuit architecture search, feedback control and error correction [1973].

Machine learning “within” deals with using it to design hardware and processes, independently of a given algorithm or quantum circuit.

- Qubit control like when Google use some machine learning techniques to optimize the drive frequency allocation of its Sycamore chip qubits in 2019, to limit the effects of crosstalk. It also works at the pulse level [1974, 1975]. Reinforcement learning is used to optimize qubit control [1976, 1977]. Transformer neural networks can also help optimize qubit control fields, better than recurrent neural networks thanks to their long context windows [1978].
- Post-processing superconducting readout in-phase and quadrature digitized signals coming out of ADCs (Analog to Digital Converters) which was using SVMs and now can use deep learning networks [1979, 1980].

- Deep transfer learning can be used for qubit characterization, like with fluxonium superconducting qubits [1981].
- Classical vision LLMs can interpret qubit calibration plots [1982].
- Machine learning can also help design photonic chip circuits [1983, 1984], which is not surprising given AI-augmented EDA (Electronic Design Automation) tools already exist to design classical circuits [1985].
- The agentic-AI trend also shows up in quantum computing. A study done in the UK introduces the k-agents framework for creating laboratory-related knowledge agents. It serves to automate experiments to perform a QPU single-qubit and two-qubit gate calibration and a characterization of a GHZ state. The k-agents succeeded in orchestrating the experiments with interpreting the multimodal experimental results, and guiding the execution of the experiment with closed-loop feedback [1986].
- A similar agentic approach at the University of Chicago combines an LLM with a lab knowledge base to implement and run superconducting qubit control-and-measurement sequences directly, letting the model both deploy standard protocols and adapt them to new experimental procedures [1987].

Machine learning “in” deals with using it to prepare quantum circuits.

- Deep reinforcement learning can be used to optimize sorting quantum algorithms [1988].
- It is used to feed data and create quantum circuits, like to prepare and optimize an ansatz for a variational algorithm like QAOA [1989] or QUBO, using reinforcement learning [1990], generative pretraining on high-performing ansatzes [1991], or a closed-loop LLM agent that iterates on feature maps and VQE ansatz against literature and experimental feedback [1992]. One proposal is about using LLMs to generate quantum circuits solving 3SAT problems [1993]. The

Q-Bridge proposal is an LLM-guided code translation tool that converts classical machine learning implementations into executable QML variants [1994].

- It is also used in quantum gate synthesis to implement a given Hamiltonian, with quantum code transpilation and optimization. Various techniques are tested using supervised machine learning [1995], simple neural networks [1996, 1997], reinforcement learning [1998–2000] and LLMs [2001–2006].
- It can even predict the entangling capacity of parameterized quantum circuits [2007]. Most QML solutions mix a classical machine learning part and a quantum part [2008].
- Various breeds of AI can help design quantum error correction codes like approximate quantum error correction (AQEC) codes [2009].
- Since 2026, LLMs and reinforcement learning techniques help discover new qLDPC error correction codes [2010–2012].

Machine learning “out” deals with using the data generated by quantum circuits for whatever purpose [2013, 2014].

- Quantum errors mitigation like the ZNE (zero noise extrapolation) technique. Variational ansatz tuning happens here, creating a new version of a quantum circuit. In a way, this machine learning “out” becomes “in” [2015, 2016].
- Predict which quantum programs a particular quantum computer can reliably execute, a technique known as quantum capability learning which can use convolutional neural networks [2017] or graph neural networks [2018]. It uses physics-aware networks which classically model the noise in the given circuit. The training data coming from actual circuit execution on a QPU is compared to emulated circuit execution. Many known quantum algorithms generate data that can be used by classical machine learning: the Quantum Kernel Methods for Support Vector Machines (SVM) that maps data into higher-dimensional spaces that feed classical SVMs for classification tasks, QAOA, Quantum Boltzmann Machines, Quantum Principal Component Analysis and Quantum Generative Adversarial Networks.
- Learning quantum circuits is the process of designing and optimizing quantum circuits using machine learning techniques. It is widely used in quantum machine learning [2019].

- Characterizing quantum systems including quantum property prediction and the construction of surrogates for quantum states which model classically an approximate behavior or properties of complex quantum systems [2020]. IBM demonstrated this at scale, training neural networks on data from a 115-qubit processor to predict ground-state observables of a 2D Heisenberg model for Hamiltonian parameters never seen during training [2021]. Similar interpretable representation approaches have been used to uncover previously unreported phenomena in Rydberg-atom and fermionic experimental data [2022], and to attribute a quantum state to the generative circuit that produced it [2023].

Generative AI and LLM-based chatbots are generic helper tools for quantum computing scientists. They can help teach, learn, create quantum circuits [2024, 2025], analyze quantum code [2026], create charts, analyze scientific papers, analyze data, write papers and the like. LLMs can also drive test generation directly: a knowledge-guided fuzzer grounded in a quantum library’s own codebase found bugs that domain-specific fuzzers missed [2027]. ChatGPT, CoPilot (based on ChatGPT), Google Gemini, Claude from Anthropic are the most famous tools in that category and their capabilities are growing quickly. Some initiatives have been launched to create quantum specific LLM-based chatbots, particularly for helping quantum circuits programming [2028], including fine-tuning pipelines meant to instill genuine circuit-simulation reasoning rather than pattern matching [2029]. They fare well when there’s a lot of available code on the web, which explains why coding results are better on IBM Qiskit than with Xanadu Strawberry Fields in a 2025 experiment [2025], a picture confirmed by a 2026 benchmark spanning Qiskit, PennyLane and Cirq, where one-shot Pass@1 scores ranged from 42.9% to 59.5% depending on the framework and rose to 66.7–83.3% once the model could see and repair its own runtime errors [2030].

Beyond code, LLM-assisted workflows have started to contribute to genuinely new quantum algorithms: one case study credits an agentic system with helping expand a human intuition about rational approximation into the sign-embedding framework now used for a family of quantum linear-algebra algorithms, while leaving the decisive judgment calls to the human researchers [2031]. The same generative capabilities cut both ways for research integrity: it has been shown that a general-purpose chatbot, given only the governing physics equations, can fabricate qubit-control curves, Josephson-effect traces and even Majorana-fermion-like signatures that are hard to distinguish from genuine experimental data, without any training on specialized datasets [2032].

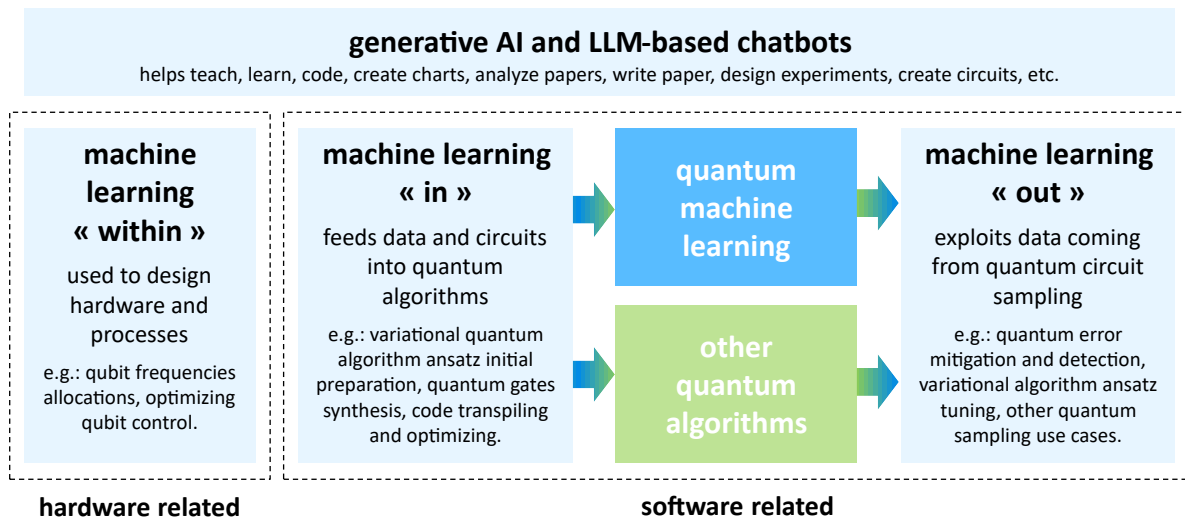
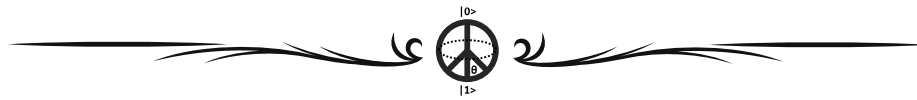


Figure 140: A way to frame the various use cases of machine learning in the context of quantum computing. Source: Olivier Ezratty, June 2024.



2.4.8 Bibliography and notes

These are the bibliographical references and notes for the **Quantum software engineering** subsection.

- [1297] [Enabling Multi-programming Mechanism for Quantum Computing in the NISQ Era](#), by Siyuan Niu and Aida Todri-Sanial, arXiv, February 2023 (34 pages).
- [1870] [Quantum Software Engineering: Roadmap and Challenges Ahead](#), by Juan M. Murillo, Jose Garcia-Alonso, Enrique Moguel, Johanna Barzen, Frank Leymann, Shaukat Ali, Tao Yue, Paolo Arcaini, Ricardo Pérez Castillo, Ignacio García Rodríguez de Guzmán, Mario Piattini, Antonio Ruiz-Cortés, Antonio Brogi, Jianjun Zhao, Andriy Miranskyy, and Manuel Wimmer, arXiv, December 2024 (49 pages).
- [1871] [Toolchain for Faster Iterations in Quantum Software Development](#), by Otso Kinanen, Andrés D. Muñoz-Moller, Vlad Stirbu, and Tommi Mikkonen, arXiv, July 2025 (28 pages).
- [1872] [Repositories, Contributors, and Continuity: An Empirical Study of Foundational Quantum Software](#), by Vincent Gierisch, Nicole Hoess, Ralf Ramsauer, and Wolfgang Maurer, arXiv, July 2026 (4 pages).
- [1873] [Logical accreditation: a framework for efficient certification of fault-tolerant computations](#), by James Mills, Adithya Sireesh, Dominik Leichtle, Joschka Roffe, and Elham Kashefi, arXiv, August 2025 (43 pages).
- [1874] [Quantum Characterization, Verification, and Validation](#), by Robin Blume-Kohout, Timothy Proctor, and Kevin Young, arXiv, March 2025 (12 pages).
- [1875] [Quantum Software Engineering in Practice: FPGA and AI Integration for Quantum Certification](#), by Marcos Guillermo Lammers, José Manuel Suárez, Adrián Pousa, Luis Mariano Bibbó, and Alejandro Fernández, arXiv, July 2026 (10 pages).
- [1876] [How to Verify a Quantum Computation](#), by Anne Broadbent, 2016 (37 pages).
- [1877] [Verification of quantum computation: An overview of existing approaches](#), by Alexandru Gheorghiu, Theodoros Kapourniotis, and Elham Kashefi, arXiv, July 2018 (65 pages).
- [1878] [Classical Verification of Quantum Computations in Linear Time](#), by Jiayu Zhang, arXiv, June 2024 (122 pages).
- [1879] [Verifying BQP Computations on Noisy Devices with Minimal Overhead](#), by Dominik Leichtle et al., PRX Quantum, October 2021 (16 pages).
- [1880] [Universal blind quantum computation](#), by Anne Broadbent, Joseph Fitzsimons, and Elham Kashefi, arXiv, December 2009 (20 pages).
- [1881] [Unifying Quantum Verification and Error-Detection: Theory and Tools for Optimisations](#), by Theodoros Kapourniotis, Elham Kashefi, Dominik Leichtle, Luka Music, and Harold Ollivier, arXiv, May 2024 (68 pages).
- [1882] [Verification of Quantum Computations without Trusted Preparations or Measurements](#), by Elham Kashefi, Dominik Leichtle, Luka Music, and Harold Ollivier, Advanced Quantum Technologies, May 2026 (12 pages) 📄 (preprint on HAL) The paper was initially published on arXiv in March 2024.
- [1883] [Towards practical secure delegated quantum computing with semi-classical light](#), by Boris Bourdoncle, Pierre-Emmanuel Emeriau, Paul Hilaire, Shane Mansfield, Luka Music, and Stephen Wein, arXiv, September 2024 (44 pages).
- [1884] [Fast Fingerprinting of Cloud-based NISQ Quantum Computers](#), by Kaitlin N. Smith, Frederic T. Chong, et al., arXiv, November 2022 (12 pages).
- [1885] [Verification of Quantum Computations: Hardware-Efficient Security Proofs](#), by Harold Ollivier, arXiv, July 2026 (91 pages).
- [1886] [Toward certified quantum programming](#), by Sébastien Bardin et al., 2018 (4 pages).
- [1887] [An Automated Deductive Verification Framework for Circuit-building Quantum Programs](#), by Christophe Charetton et al., 2021 (30 pages).
- [1888] [Classical Verification of Quantum Computations](#), by Urmila Mahadev, arXiv, December 2023 (54 pages).
- [1889] [Interactive Proofs for Quantum Computations](#), by Dorit Aharonov, Michael Ben-Or, Elad Eban, and Urmila Mahadev, arXiv, April 2017 (75 pages).
- [1890] [Graduate Student Solves Quantum Verification Problem](#), by Erica Klarreich, Quanta Magazine, October 2018.

- [1891] [Towards experimental classical verification of quantum computation](#), by Roman Stricker, Thomas Monz, et al., arXiv, March 2022 (19 pages).
- [1892] [Separating Non-Interactive Classical Verification of Quantum Computation from Falsifiable Assumptions](#), by Mohammed Barhoush, Tomoyuki Morimae, Ryo Nishimaki, and Takashi Yamakawa, Cryptology ePrint Archive, February 2026 (36 pages).
- [1893] [Formal Methods for Quantum Programs: A Survey](#), by Christophe Charetton, Sébastien Bardin, Dongho Lee, Benoît Valiron, Renaud Vilmart, and Zhaowei Xu, arXiv, April 2022 (76 pages).
- [1894] [Towards Large-scale Functional Verification of Universal Quantum Circuits](#), by Matthew Amy, arXiv, January 2019 (21 pages).
- [1895] [A Verified Optimizer for Quantum Circuits](#), by Keshu Hietala, Robert Rand, Shih-Han Hung, Xiaodi Wu, and Michael Hicks, arXiv, November 2020 (36 pages).
- [1896] [Quantum Hoare logic with classical variables](#), by Yuan Feng and Mingsheng Ying, arXiv, April 2021 (44 pages).
- [1897] [Manjushri: A Tool for Equivalence Checking of Quantum Circuits](#), by Xuan Du Trinh, Meghana Sistla, Nengkun Yu, and Thomas Reps, arXiv, January 2026 (20 pages).
- [1898] [Testing and Debugging Quantum Programs: The Road to 2030](#), by Neilson Carlos Leite Ramalho, Higor Amario de Souza, and Marcos Lordello Chaim, arXiv, October 2024 (45 pages).
- [1899] [A Survey on Testing and Analysis of Quantum Software](#), by Matteo Paltenghi and Michael Pradel, arXiv, October 2024 (36 pages).
- [1900] [A Tool For Debugging Quantum Circuits](#), by Sara Ayman Metwalli and Rodney Van Meter, arXiv, May 2022 (11 pages).
- [1901] [A Comprehensive Study of Bug Fixes in Quantum Programs](#), by Junjie Luo, Pengzhan Zhao, Zhongtao Miao, Shuhan Lan, and Jianjun Zhao, arXiv, January 2022 (8 pages).
- [1902] [Bug Classification in Quantum Software: A Rule-Based Framework and Its Evaluation](#), by Mir Mohammad Yousuf and Shabir Ahmad Sofi, arXiv, June 2025 (25 pages).
- [1903] [On the need for effective tools for debugging quantum programs](#), by Olivia Di Matteo, arXiv, February 2024 (5 pages).
- [1904] [Understanding Bugs in Quantum Simulators: An Empirical Study](#), by Krishna Upadhyay, Moshood Fakorede, and Umar Farooq, arXiv, March 2026 (26 pages).
- [1905] [Bloch Vector Assertions for Debugging Quantum Programs](#), by Noah H. Oldfield, Christoph Laaber, and Shaikat Ali, arXiv, June 2025 (40 pages).
- [1906] [QBugLM: An Agentic Benchmarking Framework for LLM-based Quantum Software Debugging](#), by An B. B. Pham, Hoa T. Nguyen, and Muhammad Usman, arXiv, June 2026 (7 pages).
- [1907] [Verification of Recursively Defined Quantum Circuits](#), by Mingsheng Ying and Zhicheng Zhang, arXiv, November 2024 (64 pages).
- [1908] [Quantum Concolic Testing](#), by Shangzhou Xia, Jianjun Zhao, Fuyuan Zhang, and Xiaoyu Guo, arXiv, June 2025 (21 pages).
- [1909] [Validation tests of Gaussian boson samplers with photon-number resolving detectors](#), by Alexander S Dellios, Margaret D Reid, and Peter D Drummond, Quantum Science and Technology, September 2025 (25 pages).
- [1910] [Formal Verification vs. Quantum Uncertainty](#), by Robert Rand, 2019 (12 pages).
- [1911] [On the Feasibility of Quantum Unit Testing](#), by Andriy Miranskyy, José Campos, Anila Mjeda, Lei Zhang, and Ignacio García Rodríguez de Guzmán, arXiv, July 2025 (18 pages).
- [1912] [QUT: A Unit Testing Framework for Quantum Subroutines](#), by Mykhailo V. Klymenko, Thong Hoang, Hoa Nguyen, Samuel A. Wilkinson, Bahar Goldozian, Xing Zhenchang, Qinghua Lu, Muhammad Usman, and Liming Zhu, arXiv, September 2025 (19 pages).
- [1913] [Debugging Quantum Processes Using Monitoring Measurements](#), by Yangjia Li and Mingsheng Ying, arXiv, March 2014 (7 pages).
- [1914] [Projection-Based Runtime Assertions for Testing and Debugging Quantum Programs](#), by Gushu Li, 2020 (29 pages).
- [1915] [Quokka#: Quantum Computing with #SAT](#), by Jingyi Mei, Dekel Zak, Muhammad Osama, Tim Coopmans, and Alfons Laarman, arXiv, May 2026 (19 pages).
- [1916] [On Refactoring Quantum Programs](#), by Jianjun Zhao, arXiv, June 2023 (8 pages).
- [1917] [Unveiling Code Clones in Quantum Programming: An Empirical Study with Qiskit](#), by Kenta Manoku and Jianjun Zhao, arXiv, January 2025 (6 pages).
- [1918] [How to Verify that a Small Device is Quantum, Unconditionally](#), by Giulio Malavolta and Tamer Mour, arXiv, May 2025 (39 pages).
- [1919] [Hacking quantum computers with row hammer attack](#), by Fernando Almaguer-Angeles, Pedro R. Dieguez, Akshata Shenoy H., and Marcin Pawłowski, arXiv, March 2025 (9 pages).
- [1920] [QubitHammer Attacks: Qubit Flipping Attacks in Multi-tenant Superconducting Quantum Computers](#), by Yizhuo Tan, Navnil Choudhury, Kanad Basu, and Jakub Szefer, arXiv, April 2025 (14 pages).
- [1921] [Masking Countermeasures Against Side-Channel Attacks on Quantum Computers](#), by Jason T. LeGrow, Travis Morrison, Jamie Sikora, and Nicolas Swanson, arXiv, January 2025 (8 pages).
- [1922] [Exploration of Quantum Computer Power Side-Channels](#), by Chuanqi Xu, Ferhat Erata, and Jakub Szefer, arXiv, May 2023 (17 pages).
- [1923] [Classification of Quantum Computer Fault Injection Attacks](#), by Chuanqi Xu, Ferhat Erata, and Jakub Szefer, arXiv, September 2023 (7 pages).
- [1924] [Security Attacks Abusing Pulse-level Quantum Circuits](#), by Chuanqi Xu and Jakub Szefer, arXiv, November 2024 (18 pages).
- [1925] [Understanding Side-Channel Vulnerabilities in Superconducting Qubit Readout Architectures](#), by Satvik Maurya, Swamit Tannu, et al., arXiv, May 2024 (7 pages).
- [1926] [Pulse-Level Simulation of Crosstalk Attacks on Superconducting Quantum Hardware](#), by Syed Emad Uddin Shubha and Tasnuva Farheen, arXiv, July 2025 (4 pages).
- [1927] [Radio-Frequency Side-Channel Analysis of a Trapped-Ion Quantum Computer](#), by Giorgio Grigolo, Dorian Schiffer, Lukas Gerster, Martin Ringbauer, and Paul Erker, arXiv, March 2026 (11 pages).
- [1928] [Schrodinger's Toolbox: Exploring the Quantum Rowhammer Attack](#), by Devon Campbell, arXiv, September 2025 (6 pages).
- [1929] [Anticipating Decoder Side-channel Attacks in Fault-tolerant Quantum Computers](#), by Shashvat Shukla, Dan E. Browne, and Shin Nishio, arXiv, July 2026 (10 pages).
- [1930] [An End-to-End Multi-Stage Kill-Chain Attack on Quantum Neural Networks: Demonstration on Trapped-Ion Hardware](#), by Cedric Brüggmann, Daniel Herr, Daniel Ohl de Mello, Pascal Debus, Maximilian Wendlinger, Kilian Tschärke, Juris Ullmanis, Alexander Erhard, Arthur Schmidt, and Fabian Petsch, arXiv, July 2026 (19 pages).
- [1931] [A Primer on Security of Quantum Computing Hardware](#), by Swaroop Ghosh, Suryansh Upadhyay, and Abdullah Ash Saki, arXiv, April 2025 (27 pages).
- [1932] [SQUASH: A SWAP-Based Quantum Attack to Sabotage Hybrid Quantum Neural Networks](#), by Rahul Kumar,

- Wenqi Wei, Ying Mao, Junaid Farooq, Ying Wang, and Juntao Chen, arXiv, June 2025 (9 pages).
- [1933] [Exploring the Vulnerabilities of Machine Learning and Quantum Machine Learning to Adversarial Attacks using a Malware Dataset: A Comparative Analysis](#), by Mst Shapna Akter, Razvan Voicu, et al., arXiv, May 2023 (10 pages).
- [1934] [TrojanNet: Detecting Trojans in Quantum Circuits using Machine Learning](#), by Subrata Das and Swaroop Ghosh, arXiv, June 2023 (9 pages).
- [1935] [QDoor: Exploiting Approximate Synthesis for Backdoor Attacks in Quantum Neural Networks](#), by Cheng Chu, Fan Chen, Philip Richerme, and Lei Jiang, arXiv, February 2024 (9 pages).
- [1936] [Quantum Neural Networks under Depolarization Noise: Exploring White-Box Attacks and Defenses](#), by David Winderl, Nicola Franco, and Jeanette Miriam Lorenz, arXiv, December 2023 (11 pages).
- [1937] [Realizing Quantum Adversarial Defense on a Trapped-ion Quantum Processor](#), by Alex Jin, Tarun Dutta, Anh Tu Ngo, Anupam Chattopadhyay, and Manas Mukherjee, arXiv, March 2025 (19 pages).
- [1938] [HarmQ: Harmonic Backdoor Attacks Against Quantum Neural Networks](#), by Junrui Zhang, Zemin Chen, Chunsheng Xin, Hongyi Wu, and Rui Ning, arXiv, July 2026 (8 pages).
- [1939] [Defending Quantum Classifiers against Adversarial Perturbations through Quantum Autoencoders](#), by Emma Andrews, Sahan Sanjaya, and Prabhat Mishra, arXiv, April 2026 (11 pages).
- [1940] [QSentry: Backdoor Detection for Quantum Neural Networks via Measurement Clustering](#), by Shuolei Wang, Zimeng Xiao, Jinjing Shi, Heyuan Shi, Shichao Zhang, and Xuelong Li, arXiv, November 2025 (15 pages).
- [1941] [Fooling the Decoder: An Adversarial Attack on Quantum Error Correction](#), by Jerome Lenssen and Alexandru Paler, arXiv, April 2025 (8 pages).
- [1942] [Fault Injection Attacks on Machine Learning-based Quantum Computer Readout Error Correction](#), by Anthony Etim and Jakub Szefer, arXiv, December 2025 (11 pages).
- [1943] [E-LoQ: Enhanced Locking for Quantum Circuit IP Protection](#), by Yuntao Liu, Jayden John, and Qian Wang, arXiv, January 2025 (10 pages).
- [1944] [Short Paper: Device- and Locality-Specific Fingerprinting of Shared NISQ Quantum Computers](#), by Allen Mi, Shuwen Deng, and Jakub Szefer, arXiv, February 2022 (5 pages).
- [1945] [Multi-Stage Watermarking for Quantum Circuits](#), by Min Yang, Xiaolong Guo, and Lei Jiang, arXiv, April 2024 (9 pages).
- [1946] [Shallow Implementation of Quantum Fingerprinting with Application to Quantum Finite Automata](#), by Mansur Ziatdinov, Aliya Khadieva, and Kamil Khadiev, arXiv, December 2024 (21 pages).
- [1947] [Q-Tag: Watermarking Quantum Circuit Generative Models](#), by Yang Yang, Yuzhu Long, Han Fang, Zhaoyun Chen, Zhonghui Li, Weiming Zhang, and Guoping Guo, arXiv, February 2026 (13 pages).
- [1948] [Securing Reset Operations in NISQ Quantum Computers](#), by Allen Mi, 2022 (15 pages).
- [1949] [Crosstalk Attack Resilient RNS Quantum Addition](#), by Bhaskar Gaur and Himanshu Thapliyal, arXiv, October 2024 (5 pages).
- [1950] [SWAP Attack: Stealthy Side-Channel Attack on Multi-Tenant Quantum Cloud System](#), by Wei Jie Bryan Lee, Siyi Wang, Suman Dutta, Walid El Maouaki, and Anupam Chattopadhyay, arXiv, February 2025 (9 pages).
- [1951] [Quantum Software Security Challenges within Shared Quantum Computing Environments](#), by Samuel Ovaskainen, Majid Haghparast, and Tommi Mikkonen, arXiv, July 2025 (6 pages).
- [1952] [Pulse-to-Circuit Characterization of Stealthy Crosstalk Attack on Multi-Tenant Superconducting Quantum Hardware](#), by Syed Emad Uddin Shubha and Tasnuva Farheen, arXiv, September 2025 (12 pages).
- [1953] [A Case for Multi-Programming quantum Computers](#), by Poulami Das, 2019 (13 pages).
- [1954] [SHARE: Secure Hardware Allocation and Resource Efficiency in Quantum Systems](#), by Suryansh Upadhyay and Swaroop Ghosh, arXiv, May 2024 (10 pages).
- [1955] [QuMoS: A Framework for Preserving Security of Quantum Machine Learning Model](#), by Zhepeng Wang, Jinyang Li, Zhirui Hu, Blake Gage, Elizabeth Iwasawa, and Weiwen Jiang, arXiv, October 2023 (10 pages).
- [1956] [QTrojan: A Circuit Backdoor Against Quantum Neural Networks](#), by Cheng Chu, Lei Jiang, Martin Swamy, and Fan Chen, arXiv, February 2023 (5 pages).
- [1957] [Towards quantum enhanced adversarial robustness in machine learning](#), by Maxwell T. West, Muhammad Usman, et al., Nature Machine Intelligence, May 2023 📄 (preprint on arXiv).
- [1958] [Adversarial attacks on hybrid classical-quantum Deep Learning models for Histopathological Cancer Detection](#), by Biswaraj Baral, Reek Majumdar, Bhavika Bhalgamiya, and Taposh Dutta Roy, arXiv, September 2023 (7 pages).
- [1959] [Adversarial Robustness Guarantees for Quantum Classifiers](#), by Neil Dowling, Kavan Modi, et al., arXiv, May 2024 (12 pages).
- [1960] [AI-driven Reverse Engineering of QML Models](#), by Archisman Ghosh and Swaroop Ghosh, arXiv, August 2024 (7 pages).
- [1961] [Quantum Trojan Insertion: Controlled Activation for Covert Circuit Manipulation](#), by Jayden John, Lakshman Golla, and Qian Wang, arXiv, February 2025 (7 pages).
- [1962] [Randomized Reversible Gate-Based Obfuscation for Secured Compilation of Quantum Circuit](#), by Subrata Das and Swaroop Ghosh, arXiv, June 2023 (11 pages).
- [1963] [Toward Privacy in Quantum Program Execution On Untrusted Quantum Cloud Computing Machines for Business-sensitive Quantum Needs](#), by Tirthak Patel, Devesh Tiwari, et al., arXiv, July 2023 (14 pages).
- [1964] [Obfuscation of Unitary Quantum Programs](#), by Mi-Ying Huang and Er-Cheng Tang, arXiv, July 2025 (56 pages).
- [1965] [Obfuscation of Arbitrary Quantum Circuits](#), by Miryam Mi-Ying Huang and Er-Cheng Tang, arXiv, January 2026 (39 pages).
- [1966] [Security Evaluation of Quantum Circuit Split Compilation under an Oracle-Guided Attack](#), by Hongyu Zhang and Yuntao Liu, arXiv, November 2025 (6 pages).
- [1967] [Hardware Architecture for a Quantum Computer Trusted Execution Environment](#), by Theodoros Trochatos, Chuanqi Xu, Sanjay Deshpande, Yao Lu, Yongshan Ding, and Jakub Szefer, arXiv, August 2023 (13 pages).
- [1968] [Quantum Computer Fingerprinting using Error Syndromes](#), by Vincent Mutolo, Devon Campbell, Quinn Manning, Henri Witold Dubourg, Ruibin Lyu, Simha Sethumadhavan, Daniel Rubenstein, and Salvatore Stolfo, arXiv, June 2025 (14 pages).
- [1969] [Artificial Intelligence for Quantum Computing](#), by Yuri Alexeev, Timothy Costa, et al., Nature, December 2025 (19 pages).
- [1970] [Deep Learning in Classical and Quantum Physics](#), by Timothy Heightman and Marcin Płodzień, arXiv, August 2025 (205 pages).
- [1971] [Machine Learning for Quantum Computing Specialists](#), by Daniel Goldsmith and M M Hassan Mahmud, arXiv, April 2024 (32 pages).
- [1972] [When AI meets quantum information: A comprehensive review](#), by Min Chen, Yu Gan, Xin Jin, Yuqing Li, Junqi Wang, Zeguan Wu, Yunfei Wang, Bingzhi Zhang, Priyama Srivastava, Tianlong Chen, Ankit Kulshrestha, Yuan Liu,

- Juan José Mendoza-Arenas, Kaushik P. Seshadreesan, Sarvagya Upadhyay, Xueyue Zhang, Quntao Zhuang, and Junyu Liu, arXiv, July 2026 (62 pages).
- [1973] [Reinforcement Learning for Quantum Technology](#), by Marin Bukov and Florian Marquardt, arXiv, January 2026 (60 pages).
- [1974] [Robust Quantum Control using Reinforcement Learning from Demonstration](#), by Shengyong Li, Pengtao Song, et al., arXiv, April 2025 (11 pages).
- [1975] [Fast Machine Learning for Quantum Control of Microwave Qudits on Edge Hardware](#), by Flor Sanders, Gaurav Agarwal, Luca Carloni, Giuseppe Di Guglielmo, Andy C. Y. Li, and Gabriel N. Perdue, arXiv, June 2025 (11 pages).
- [1976] [Entanglement engineering of optomechanical systems by reinforcement learning](#), by Li-Li Ye, Christian Arenz, Joseph M. Lukens, and Ying-Cheng Lai, arXiv, December 2024 (17 pages).
- [1977] [Preparing Schrödinger cat states in a microwave cavity using a neural network](#), by Hector Hutin, Yonatan Cohen, Audrey Bienfait, Florian Marquardt, Benjamin Huard, et al., arXiv, September 2024 (20 pages).
- [1978] [Quantum feedback control with a transformer neural network architecture](#), by Pranav Vaidhyanathan, Florian Marquardt, Mark T. Mitchison, and Natalia Ares, arXiv, November 2024 (9 pages).
- [1979] [KLiNQ: Knowledge Distillation-Assisted Lightweight Neural Network for Qubit Readout on FPGA](#), by Xiaorang Guo, Tigran Bunarjyan, Dai Liu, Benjamin Lienhard, and Martin Schulz, arXiv, March 2025 (7 pages).
- [1980] [Machine Learning for Estimation and Control of Quantum Systems](#), by Hailan Ma, Bo Qi, Ian R. Petersen, Re-Bing Wu, Herschel Rabitz, and Daoyi Dong, arXiv, March 2025 (23 pages).
- [1981] [Automatic Characterization of Fluxonium Superconducting Qubits Parameters with Deep Transfer Learning](#), by Huan-Hsuan Kung, Yen-Hsiang Lin, et al., arXiv, March 2025 (11 pages).
- [1982] [QCalEval: Benchmarking Vision-Language Models for Quantum Calibration Plot Understanding](#), by Shuxiang Cao, Alán Aspuru-Guzik, Krysta Svore, et al., arXiv, April 2026 (29 pages).
- [1983] [Enhanced Photonic Chip Design via Interpretable Machine Learning Techniques](#), by Lirandê Pira, Airin Antony, Nayanthara Prathap, Daniel Peace, and Jacqueline Romero, arXiv, May 2025 (15 pages).
- [1984] [Digital Discovery of 100 diverse Quantum Experiments with PyTheus](#), by Carlos Ruiz-Gonzalez, Mario Krenn, et al., Quantum, December 2023 (58 pages).
- [1985] [The Dawn of AI-Native EDA: Opportunities and Challenges of Large Circuit Models](#), by Lei Chen et al., arXiv, May 2024 (28 pages).
- [1986] [Agents for self-driving laboratories applied to quantum computing](#), by Shuxiang Cao, Peter Leek, Alán Aspuru-Guzik, et al., arXiv, June 2025 (37 pages).
- [1987] [Large Language Model-Assisted Superconducting Qubit Experiments](#), by Shiheng Li, J. M. Miller, P. J. Lee, G. Andersson, C. R. Conner, Y. J. Joshi, B. Karimi, A. M. King, H. L. Malc, H. Mishra, H. Qiao, M. Ryu, X. Wu, S. Xing, H. Yan, J. Shi, and Andrew N. Cleland, arXiv, March 2026 (10 pages).
- [1988] [Faster sorting algorithms discovered using deep reinforcement learning](#), by Daniel J. Mankowitz, David Silver, et al., Nature, June 2023.
- [1989] [Transfer learning of optimal QAOA parameters in combinatorial optimization](#), by J. A. Montanez-Barrera, Dennis Willsch, and Kristel Michielsen, arXiv, May 2025 (15 pages).
- [1990] [Reinforcement Learning for Variational Quantum Circuits Design](#), by Simone Foderà, Gloria Turati, Riccardo Nembrini, Maurizio Ferrari Dacrema, and Paolo Cremonesi, arXiv, September 2024 (11 pages).
- [1991] [Q3SAT-GPT: A Generative Model for Discovering Quantum Circuits for the 3-SAT Problem](#), by Pratim Ugale, Ilya Tyagin, Karunya Shirali, Kien X. Nguyen, and Ilya Safro, arXiv, April 2026 (13 pages).
- [1992] [An LLM System for Autonomous Variational Quantum Circuit Design](#), by Kenya Sakka, Wataru Mizukami, and Kosuke Mitarai, arXiv, June 2026 (63 pages).
- [1993] [Q3SAT-GPT: A Generative Model for Discovering Quantum Circuits for the 3-SAT Problem](#), by Pratim Ugale, Ilya Tyagin, Karunya Shirali, Kien X. Nguyen, and Ilya Safro, arXiv, April 2026 (13 pages).
- [1994] [Q-Bridge: Code Translation for Quantum Machine Learning via LLMs](#), by Runjia Zeng, Priyabrata Senapati, Ruixiang Tang, Dongfang Liu, and Qiang Guan, arXiv, March 2026 (26 pages).
- [1995] [Fast and Noise-aware Machine Learning Variational Quantum Eigensolver Optimiser](#), by Akib Karim, Shaobo Zhang, and Muhammad Usman, arXiv, March 2025 (11 pages).
- [1996] [Neural-Network-Based Design of Approximate Gottesman-Kitaev-Preskill Code](#), by Yexiong Zeng, Physical Review Letters, February 2025 (preprint on arXiv).
- [1997] [Making Neural Networks More Suitable for Approximate Clifford+T Circuit Synthesis](#), by Mathias Weiden, Justin Kalloor, John Kubiatawicz, and Costin Iancu, arXiv, April 2025 (10 pages).
- [1998] [Scaling the Automated Discovery of Quantum Circuits via Reinforcement Learning with Gadgets](#), by Jan Olle, Oleg M. Yevtushenko, and Florian Marquardt, arXiv, March 2025 (14 pages).
- [1999] [Designing Shadow Tomography Protocols by Natural Language Processing](#), by Yadong Wu, Pengfei Zhang, Ce Wang, Juan Yao, and Yi-Zhuang You, arXiv, September 2025 (6 pages).
- [2000] [Investigation of Automated Design of Quantum Circuits for Imaginary Time Evolution Methods Using Deep Reinforcement Learning](#), by Ryo Suzuki and Shohei Watabe, arXiv, April 2026 (11 pages).
- [2001] [Enhancing LLM-based Quantum Code Generation with Multi-Agent Optimization and Quantum Error Correction](#), by Charlie Campbell, Hao Mark Chen, Wayne Luk, and Hongxiang Fan, arXiv, April 2025 (7 pages).
- [2002] [Quantum State Preparation via Large-Language-Model-Driven Evolution](#), by Qing-Hong Cao, Zong-Yue Hou, Ying-Ying Li, Xiaohui Liu, Zhuo-Yang Song, Liang-Qi Zhang, Shutao Zhang, and Ke Zhao, arXiv, May 2025 (4 pages).
- [2003] [QAOA-GPT: Efficient Generation of Adaptive and Regular Quantum Approximate Optimization Algorithm Circuits](#), by Ilya Tyagin, Marwa H. Farag, Kyle Sherbert, Karunya Shirali, Yuri Alexeev, and Ilya Safro, arXiv, April 2025 (11 pages).
- [2004] [LLM-Guided Ansätze Design for Quantum Circuit Born Machines in Financial Generative Modeling](#), by Yaswitha Gujju, Romain Harang, and Tetsuo Shibuya, arXiv, September 2025 (5 pages).
- [2005] [FlowQ-Net: A Generative Framework for Automated Quantum Circuit Design](#), by Jun Dai, Michael Rizvi-Martel, and Guillaume Rabusseau, arXiv, October 2025 (19 pages).
- [2006] [Aligning Quantum Operators with Large Language Models](#), by Rogerio Feris, Yunchao Liu, Pengyuan Li, Hang Hua, and David Kremer, arXiv, June 2026 (7 pages).
- [2007] [Machine-Learning Insights into the Entanglement-trainability Correlation of Parameterized Quantum Circuits](#), by Shikun Zhang, Yang Zhou, Zheng Qin, Rui Li, Chunxiao Du, Zhisong Xiao, and Yongyou Zhang, arXiv, May 2025 (12 pages).
- [2008] [Synergy between noisy quantum computers and scalable classical deep learning](#), by Simone Cantori, Andrea Mari,

- David Vitali, and Sebastiano Pilati, arXiv, April 2024 (17 pages).
- [2009] [Noise-strength-adapted approximate quantum codes inspired by machine learning](#), by Shuwei Liu, Shiyu Zhou, Zi-Wen Liu, and Jinmin Yi, arXiv, March 2025 (17 pages).
- [2010] [Multi-agent discovery of practical quantum LDPC codes](#), by Dongheng Qian and Tianyi Li, arXiv, August 2026 (20 pages).
- [2011] [Evolutionary Discovery of Bivariate Bicycle Codes with LLM-Guided Search](#), by Juan Cruz-Benito, Andrew W. Cross, David Kremer, and Ismael Faro, arXiv, June 2026 (42 pages).
- [2012] [Discovering highly efficient low-weight quantum error-correcting codes with reinforcement learning](#), by Austin Yubo He and Zi-Wen Liu, arXiv, February 2025 (18 pages).
- [2013] [Attention to Quantum Complexity](#), by Hyejin Kim, William D. Oliver, Yuri D. Lensky, Kilian Q. Weinberger, Eun-Ah Kim, et al., arXiv, November 2024 (38 pages).
- [2014] [Online learning of quantum processes](#), by Asad Raza, Matthias C. Caro, Jens Eisert, and Sumeet Khatri, arXiv, June 2024 (72 pages).
- [2015] [Simultaneous Discovery of Quantum Error Correction Codes and Encoders with a Noise-Aware Reinforcement Learning Agent](#), by Jan Olle, Remmy Zen, Matteo Puviani, and Florian Marquardt, arXiv, April 2024 (23 pages).
- [2016] [Improving Quantum Computer Error Correction with Deep Learning](#), by Louis Castel, arXiv, May 2025.
- [2017] [Learning a quantum computer's capability](#), by Daniel Hothem, Kevin Young, Tommie Catanach, and Timothy Proctor, arXiv, October 2024 (20 pages).
- [2018] [What is my quantum computer good for? Quantum capability learning with physics-aware neural networks](#), by Daniel Hothem, Ashe Miller, and Timothy Proctor, arXiv, February 2025 (24 pages).
- [2019] [Learning shallow quantum circuits](#), by Hsin-Yuan Huang, Jarrod R. McClean, et al., arXiv, January 2024 (10 pages).
- [2020] [Artificial intelligence for representing and characterizing quantum systems](#), by Yuxuan Du, Barry C. Sanders, et al., arXiv, September 2025 (32 pages).
- [2021] [Learning ground state observables from quantum computing experiments](#), by Ben Jaderberg, Freya Shah, Minjun Jeon, M. Emre Sahin, Christa Zoufal, and Kunal Sharma, arXiv, June 2026 (20 pages).
- [2022] [Discovering quantum phenomena with Interpretable Machine Learning](#), by Paulin de Schoulepnikoff, Hendrik Poulsen Nautrup, Hans J. Briegel, and Gorka Muñoz-Gil, arXiv, April 2026 (23 pages).
- [2023] [Identification of quantum generative circuits with parallel quantum neural network](#), by Zheping Wu, Xiaopeng Huang, Hengyue Jia, Haobin Shi, and Wei-Wei Zhang, arXiv, March 2026 (11 pages).
- [2024] [Pushing the Limits of LLMs in Quantum Operations](#), by Dayton C. Closser and Zbigniew J. Kabala, arXiv, July 2025 (17 pages).
- [2025] [Programming Quantum Computers with Large Language Models](#), by Elena R. Henderson, Jessie M. Henderson, Joshua Ange, and Mitchell A. Thornton, arXiv, June 2025 (15 pages).
- [2026] [Exploring LLM-Driven Explanations for Quantum Algorithms](#), by Giordano d'Aloisio, Federica Sarro, et al., arXiv, September 2024 (7 pages).
- [2027] [KQFuzz: Knowledge-Guided Fuzzing for Quantum Libraries via Large Language Models](#), by Fuyuan Xia, Qixin Zhang, Chenhao Ying, Haojin Zhu, Shuai Wang, Yuan Luo, Pingchuan Ma, and Yuxuan Du, arXiv, July 2026 (17 pages).
- [2028] [Architecture for a Trustworthy Quantum Chatbot](#), by Yaiza Aragonés-Soria and Manuel Oriol, arXiv, March 2025 (15 pages).
- [2029] [Fine-Tuning Large Language Models for Quantum Reasoning](#), by Katherine Ip, Casey R. Myers, Udaya Paramalli, James Quach, and Peiyong Wang, arXiv, June 2026 (30 pages).
- [2030] [QuanBench+: A Unified Multi-Framework Benchmark for LLM-Based Quantum Code Generation](#), by Ali Slim, Haydar Hamieh, Jawad Kotaich, Yehya Ghosn, Mahdi Chehimi, Ammar Mohanna, Hasan Abed Al Kader Hammoud, and Bernard Ghanem, arXiv, April 2026 (24 pages).
- [2031] [From Meta Idea to Advanced Mathematical Discovery – Human-AI Co-Discovery of Sign-Embedding Quantum Algorithms](#), by Yanqiao Wang, Jin-Peng Liu, Peng Li, and Yang Liu, arXiv, June 2026 (35 pages).
- [2032] [Realistic quantum device data synthesized by consumer AI and how to identify it](#), by S. M. Frolov and O. V. Kravchenko, arXiv, June 2026 (22 pages).

This **Quantum software engineering** subsection contains 164 bibliographical references and notes containing at least 3,649 pages.

2.5 Benchmarking

Benchmarking quantum computing is becoming a strategic tool for both vendors and users. In the IT space, vendors have continuously relied on benchmarking to showcase new hardware advances and customers rely on it to assess the cost/benefit ratio of emerging technologies. Benchmarks drive the competition between vendors and technologies [2033]. To help you navigate this burgeoning field, a Quantum Benchmark Zoo was created in 2025 by Valentin Gilbert [2034] (Figure 142). See also a benchmarking segmentation and review in [2035].

2.5.1 Classical computing benchmarking

In classical computing, floating-point throughput is measured in **FLOP/s** (floating-point operations per second). The **TOP500** ranking uses the High Performance LINPACK (HPL) benchmark, which measures the sustained performance of a system when solving a dense system of linear equations. **FIRESTARTER** is primarily a processor stress-testing and power-validation tool rather than a general performance benchmark. **MLPerf** is a benchmark suite for machine-learning training and inference on CPUs, GPUs and accelerators. **ImageNet** and **MNIST** are datasets that are widely used in machine-learning evaluations, not benchmarks by themselves.

2.5.2 Quantum computing benchmarking

Quantum computing benchmarking can have several purposes as shown in Figure 143:

- Comparing quantum computing with classical computing requires best-in-class algorithms and implementations on both sides. Both are moving targets, particularly considering classical accelerators and emulation algorithms keep improving. A quantum advantage should therefore correspond to a specified task, accuracy target and resource metric. Google’s 2019 random-circuit-sampling experiment used linear cross-entropy benchmarking (XEB) for this purpose. Direct XEB and Quantum Volume verification can require classically computing ideal output probabilities, which becomes expensive as circuits grow. Clifford-circuit variants remain efficiently classically simulable and can therefore support scalable verification [2036], but they probe Clifford dynamics and do not represent the full classical hardness of generic non-Clifford random circuits.
- Comparing different quantum computers, which compete with each other, to solve particular tasks. It can be done with quantum computers using the same programming paradigm like gate-based systems (e.g. IBM’s quantum volume), or even, using different paradigms (annealing, simulation, gate-based; e.g. Eviden Q-score). An objective comparison can be

made by solving one or a broader set of specific problems and assessing the maximum problem size addressable by competing solutions (e.g., Eviden Q-score, QED-C and IonQ’s algorithmic qubits).

- Comparing different characteristics. The most common is computing time, but other metrics will become important as well such as precision, energy spent, weight, environmental footprint and total cost (e.g. DARPA’s benchmarking project RFP won by Raytheon BBN and the Quantum Energy Initiative’s Green 500 benchmark proposal).
- Comparing different algorithms solving a given problem on similar or different quantum hardware. Such comparisons already exist, although they remain difficult to normalize because compilation, error mitigation, classical optimization and hardware-specific implementations can dominate the end-to-end result.

There are many and diverse benchmarking tools around. Following a bottom-to-top system approach, one first encounters hardware specifications and characterization metrics such as qubit count, one- and two-qubit gate errors, readout errors, connectivity, coherence times and crosstalk. System-level benchmarks and throughput metrics, such as quantum volume and CLOPS, combine several of these effects into higher-level figures of merit [2037, 2038] (Figure 141). The analogy with classical computing is therefore only partial: qubit count and connectivity resemble hardware specifications, while randomized benchmarking, layer fidelity, quantum volume and CLOPS are measured characterization or performance metrics. Then, some are benchmarking hardware capabilities at a higher level and regardless of the use case (quantum volume from IBM). At last, others are assessing the capability to solve one or several typical use cases, and usually, their maximum size (QED-C, Atos Q-score, IonQ algorithmic qubits, SupermarQ). At some point, benchmarking will be even more tricky when mixing quantum and classical computers [2039].

Table 11 and Table 12 consolidate these proposals, sorted into low-level, multiple-use-case, single-use-case and other benchmarks.

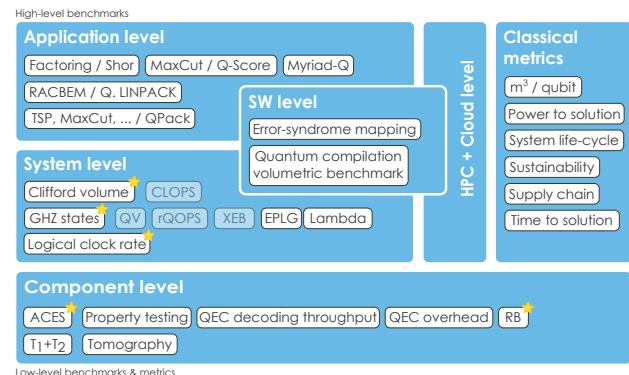


Figure 141: From low-level to high-level benchmarking proposals. Source: [2039].

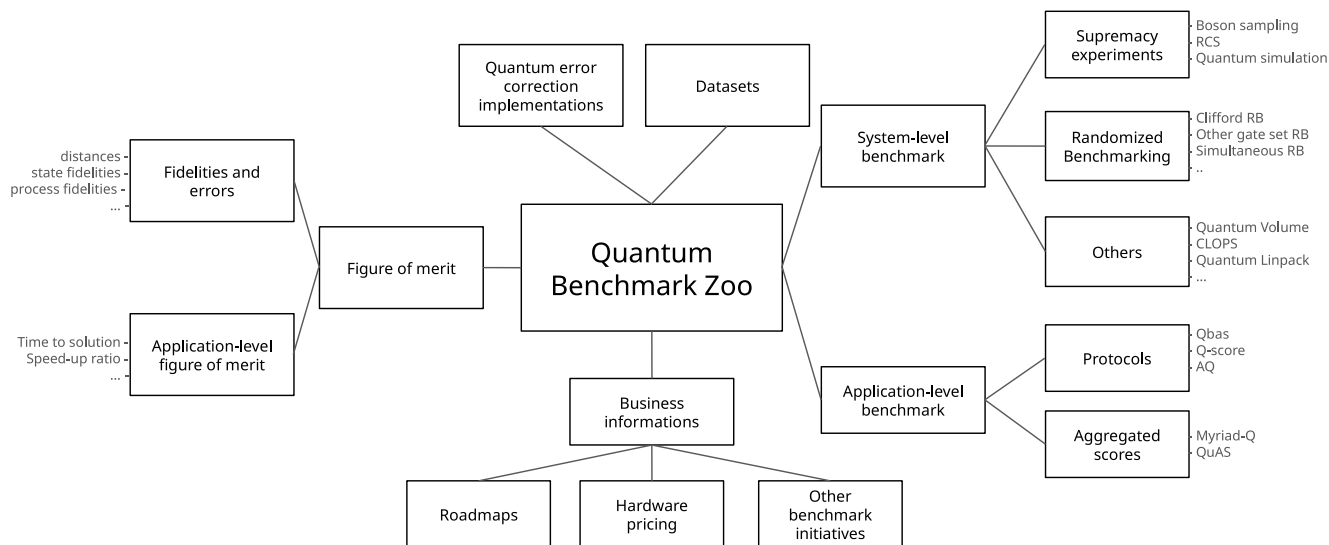


Figure 142: The structure of the Quantum Benchmark Zoo. For each item, the QBZ website provides a list of benchmarks, their characteristics and scientific references. Source: [2040].

	multiple QPUs solving the same problem with the same algorithm	multiple QPUs solving the same problem but with different algorithms	multiple QPUs solving the same problem with different computing paradigms	QPU and a classical computing solution solving a given problem	future QPU solution and an existing QPU and/or classical solution solving a given problem
paradigm			different	totally different	totally different
application	same	same	same	same	≈ the same
algorithms					
compiler					everything is different
control		potentially different	potentially different	everything is different	
processor	potentially different				and we are comparing actual energy consumption vs theoretical estimates
enabling					

Figure 143: Different types of benchmark comparisons and what is different or similar. For most comparisons when the hardware and computing paradigms are different, you need to use application-level benchmarking techniques. The algorithms may be the same only when you compare two gate-based quantum computers. © Olivier Ezratty, September 2025.

These tools are created by different kinds of players, sometimes working together: research laboratories such as Sandia National Laboratories, software and hardware vendors, industry consortia such as QED-C, academic alliances such as the Quantum Software Alliance, and standards bodies such as IEEE. The combination of these communities is important.

Vendors have commercial incentives that can influence which benchmarks they promote, particularly when a metric favors one hardware architecture over another. This matters because quantum-computing platforms differ widely in gate times, fidelities, connectivity, native gate sets and scaling constraints.

Some benchmarks can also be misleading if improperly extrapolated in the future, given all technologies don't have the same upscaling potential. An equivalent to Moore's law can't be applied in a simplistic way to all

these technologies' roadmaps. In all cases, quantum computing benchmarking will be difficult to interpret without the right technical background. You can be easily fooled by vendors given the complexity of the matter.

2.5.3 IBM Quantum Volume

Starting late 2021, IBM used three systems-level metrics and benchmarks to characterize its quantum computers: the scale with the number of qubits, the speed measured in CLOPS (circuit layer operations per second) and qubits quality measured with their homemade quantum volume metric.

The quantum volume was introduced in 2017 and was adopted by **Quantinuum** in March 2020 and afterwards by **IonQ** in October 2020, although they later

what and whom	what	pros	cons	timing / adoption
low level benchmarking				
IBM quantum volume	largest successfully executed square random circuit, reported as 2^m	compact system-level measure sensitive to fidelity, connectivity, compilation and crosstalk	exact heavy-output verification becomes classically expensive as m grows	original concept in 2017, measurable protocol published in 2019 [1686, 2041]
Cisco MBQC quantum volume	maps physical MBQC noise to logical circuit errors for CV GKP cluster-state processors	extends circuit-oriented metrics to measurement-based photonic architectures	demonstrated for a specific CV GKP cluster-state model	preprint in 2022, peer-reviewed publication in 2023 [2042]
IBM CLOPS and CLOPS _h	circuit layers operations per seconds	complements QV for speed	N/A	announced in November 2021 [2043–2045]
IBM error per layered gate (EPLG)	gate-level error metric derived from Layer Fidelity	incorporates errors measured while disjoint two-qubit layers run across a connected qubit subset	vendor-defined metric and dependent on the selected qubit subset and layer construction	introduced in November 2023 [2046, 2047]
Error Per Circuit Layer (EPCL)	circuit-level benchmark	analyzes inter-register crosstalk	doubles qubit requirements	Duke University September 2026 proposal [2048]
randomized & cross-entropy benchmarks (XEB)	random circuits with statistically defined output tests	sensitive to aggregate circuit errors and useful for sampling experiments	verification can become classically expensive at large width and depth, and the metric is task-specific	XEB used prominently by Google in 2019 and later work [2049, 2050]
cycle benchmarking	estimates process fidelity of repeated gate cycles	scalable characterization of correlated and context-dependent errors	low-level characterization rather than an application benchmark	published in 2019 by an international academic team [2051]
GHZ benchmarks	tests fidelity of GHZ states with n qubits	ability to test mid-scale entanglement	sometimes done using post-selection, doesn't scale well	QuEra, Quantinuum, others, 2023–2024 [2052, 2053]
scalable benchmarks for gate-based QC	six low-level structured circuits tests	tested 21 configurations from IBM, IonQ and Rigetti	low-level benchmark not usage based	published in 2021, QuSoft, Cambridge, Caltech [2054]
PQF (photonic quality factor)	assess performance of linear optics single photons multimode QPUs	covers many NISQ photon qubit implementations	limited to a specific photonic qubit configuration	published in 2022 by Quandela [2055]
entanglement-based volumetric benchmark	estimate size of maximum entangled qubit state	entanglement is a key feature of quantum acceleration	narrow and not usage oriented	proposed in 2022 by DoE Oak Ridge et al [2056]
Clifford volume and free fermion volume	volumetric tests using random Clifford and free fermion circuits	classically verifiable at scale, jointly universal gate set	recent, no adoption so far	published in December 2025, Wigner Research Centre and AQT [2057]
QLOPS	quantum logical operations per second in an FTQC stack	incorporates logical-qubit count, syndrome-cycle rate and fault-tolerance overheads	depends on the QEC scheme, decoder throughput/latency and assumed logical operation mix	proposed in 2025 and peer-reviewed in 2026 [2058]
single use case benchmarking				
Eviden Q-score	maximum standardized Max-Cut instance size meeting the benchmark criterion	application-oriented, hardware-paradigm independent, candidate solutions are easy to score	limited to a specific optimization family and does not by itself establish quantum advantage	published in 2020, Atos, with later extensions [2059–2061]
DoE ORNL	chemical simulation	works on existing superconducting hardware	limited to three 2-atom molecules simulations	published in 2020, DoE [2062]
Zapata benchmark for fermionic quantum simulations	one-dimensional Fermi Hubbard model (FHM) VQE running on NISQ	tested on Google Sycamore with its tunable couplers	narrow use case	proposed in March 2020 [2063]
HamilToniQ	QAOA benchmark using quantum error mitigation	targets a widely used NISQ algorithm	single algorithm, NISQ only	published in 2024 [2064]
MBQS (many-body quantum score)	maximum qubit count reproducing transverse-field Ising correlations after a quench	works on gate-based and analog QPUs, scalable	one physics use case	published in 2026, CEA and Eviden Quantum Lab [2065]
Quantum WalkScore	scoring largest problem size to solve a graph nodefinding problem	works for both NISQ and FTQC	based on quadratic speedup algorithms, with poor/no practical advantage	proposed in September 2026 by Thales [2066]

Table 11: Inventory of quantum computing benchmarking proposals, from low-level system benchmarks to multi application-level ones. © Olivier Ezratty, 2022–September 2026.

tweaked it to create their own custom algorithmic qubit benchmark.

Quantum volume is an integer that associates the quantity of qubits and the number of quantum gates that can be executed consecutively with a reasonable error rate. Indeed, having N qubits but being limited by

what and whom	what	pros	cons	timing / adoption
multiple use cases benchmarking				
QED-C supported benchmark	set of low-level algorithms benchmarks	breadth of use cases	complicated visualization	published in 2021, QED-C, Princeton, HQS, QCI, IonQ, D-Wave, Sandia Labs [2067–2069]
IonQ algorithmic qubits	largest box with width N and CX-count depth N^2 for which all benchmark circuits pass	based on several algorithm families rather than only random circuits	suite- and threshold-dependent single-number metric	formalized by IonQ in 2022 [2070]
SupermarQ from Super.tech	scalable hardware-agnostic suite of application-level circuits	uses feature vectors to characterize workload coverage and reports application-level results	suite-based benchmark, with no single universal figure of merit	published at IEEE HPCA in 2022 [2071]
QPack by TU Delft	three sets of problems (MaxCut, TSP, DSP)	measures different metrics	adoption	proposed in April 2022, extended in 2024 with QuAS [2072, 2073]
MQT Bench	circuits at several abstraction levels, from QFT and QPE to Grover, Shor and HHL	benchmarks the design automation tools and not only the QPUs	circuits suite, no single figure of merit	published in 2022-2023, TU Munich [2074]
QUARK	open-source application benchmarking framework	driven by industry end-users, covers optimization and QML	few application classes	published in 2022, QUTAC consortium [2075, 2076]
QuSquare	four system and application level quality tests	explicit relevance, reproducibility, fairness, verifiability and scalability criteria	recent, no adoption so far	published in December 2025, University of the Basque Country and BCAM [2077, 2078]
EU Quantum Flagship KPIs	Clifford volume, GHZ states, Shor period finding and Bell-state QEC gain	scalable, spans the NISQ and FTQC regimes	recent, no adoption so far	published in December 2025 [2079]
other benchmarks				
Unitary Fund Metriq	repository of benchmark results	N/A	N/A	announced in May 2022, formalized in 2026 [2080]
Benchpress	SDKs, compilers and transpilers evaluation with 1,066 tests	benchmarks the software stack and not the hardware	says nothing on qubits quality	published in 2024-2025 by IBM [2081]
DARPA Quantum Benchmarking	application benchmarks and hardware-specific resource estimation for quantum computing	utility-driven, end-to-end resource estimates for selected applications	program-specific studies rather than a universal single metric	launched in 2021, with results published through 2024, while QBI is a distinct later DARPA initiative [2082]
IEEE P7131	quantum computing performance metrics and performance benchmarking	active standardization project		active in 2026 [2083]
Quantum Energy Initiative	QC energetics benchmarking consolidated approach, QGreen500 proposal could consolidate cryogeny benchmarks	methodology (MNR) to optimize QC energetics, first analysis done with superconducting qubits	research and industry must build coordination around this goal	joint research/industry Quantum Energy Initiative launched in 2022, IEEE Working Group P3329 launched in 2023 [2084]
BACQ	application and low-level full-stack benchmarking proposal	covers many use cases and figures of merit, includes energetics performance	participants are so far only from the France quantum ecosystem	project launched in 2023 by Eviden, CEA, CNRS, Thales, Teratec and LNE [2085]
QUOPS	full stack approach “quantum universal operation performance system”	cross-platform and can compare NISQ and FTQC systems	confusing naming, with Quops.	September 2023 DoE and Quantum proposal [2086]

Table 12: Second part of the inventory of quantum computing benchmarking proposals, with single application-level ones and other transversal initiatives. © Olivier Ezratty, 2022-September 2026.

the number of quantum gates that can be used can be detrimental to the execution of many quantum algorithms. Some are greedy for quantum gates, others are not. Some algorithms can thus be satisfied with a limited number of quantum gates, such as Deutsch-Jozsa’s, which uses a single oracle query in the standard query model. The compiled gate depth, however, depends on how the oracle itself is implemented.

This quantum volume number is supposed to aggregate four performance factors:

- The **number of physical qubits** of the processor.
- The **number of quantum gates** that can be chained consecutively without the error rate being detrimental to the results.
- The **connectivity between these qubits**, which will impact the length of execution of an algorithm and potentially improve quantum volume for qubits with high connectivity such as with trapped ions. It reduces the number of SWAP gates that are required when connectivity is limited.

- The **number of quantum gates** that can be executed in parallel.

The 2019 Quantum Volume protocol uses random square circuits of width m and depth m . For each circuit, the ideal output probabilities define the “heavy” half of the bit strings. The hardware passes at a given m when the measured heavy-output probability is greater than $2/3$ with the prescribed statistical confidence [1686]. The $2/3$ threshold is therefore a benchmark criterion placed between the uniform-output value $1/2$ and the higher heavy-output probability expected from an ideal random circuit. Repeated shots estimate the heavy-output probability and its confidence interval.

The 2017 proposal defined an architecture-neutral volume approximately as $V_Q = \max_n \min[n, d(n)]^2$, where $d(n)$ is the achievable circuit depth for a selected n -qubit subset [2041]. The 2019 measurable protocol instead reports $QV = 2^m$, where m is the largest square random circuit size that passes the heavy-output test [1686]. Figure 144 compares these two definitions.

The diagram in Figure 145 from a paper by Robin Blume-Kohout and Kevin Young specifies how the m (number of qubits) and the d (computational depth) are evaluated [2087].

The number of qubits obtained to evaluate the quantum volume is much lower than the total number of qubits available: 8 for 16 in this case. The benchmark allows only 8 series of quantum gates in a row over 8 qubits, for 38 with only two qubits. In its 2017 version, the quantum volume was the grey square area containing the red lined squares.

In its 2019 version, a passing size $m = 8$ is reported as $QV = 2^8 = 256$, instead of the 2017 value $8^2 = 64$. Although 2^m is numerically equal to the dimension of the Hilbert space of m qubits, Quantum Volume is not a number of superposed states a quantum processor can “manage”. It is a convention for reporting the largest passing square random-circuit size.

When IBM stated that their 27-qubit processor had a quantum volume equal to 128, it meant that they only managed to validate their benchmark with 7 qubits among these 27 qubits.

IonQ announced in 2020 a quantum volume of 4,194,304, or 2^{22} . This means that a selected 22-qubit subset passed the 2019 Quantum Volume heavy-output test at width and depth 22. The result benefited from the high connectivity of trapped-ion systems, where pairwise entangling gates can generally be implemented between arbitrary ions within a chain. Superconducting processors usually have sparse, local coupling graphs, so compiling an arbitrary-pair random layer can require additional routing gates. The comparison is architecture-dependent, however, and all-to-all trapped-ion connectivity itself acquires control and scaling costs as systems grow.

When you look at the relative progress of QV between IBM, IonQ and Honeywell/Quantinuum systems, you

see a clear difference: IBM has a tougher time to use all its qubits while trapped ions do it better, although with a rather limited number of qubits [2088] (Figure 145).

But if and when IBM fixes some scalability issues with their qubits, like with reducing qubit crosstalk, they can potentially increase their QV to higher levels than those from trapped ions vendors.

The standard Quantum Volume protocol requires classical computation of the ideal output probabilities in order to identify the heavy outputs. Exact state-vector verification therefore becomes impractical as the circuit width m approaches roughly 50 qubits, although the precise limit depends strongly on the classical algorithm and hardware. This is a verification bottleneck for the conventional protocol, not a statement that the numerical Quantum Volume is limited to 50. The corresponding reported value would be $QV = 2^m$. Quantum Volume was explicitly designed as a near-term, non-fault-tolerant system benchmark [1686]. Proposals such as parity-based tests aim to extend device benchmarking beyond direct classical verification [2089].

Quantum computing scientists are circumspect about the interest of this indicator, which is too simplistic. Like any aggregate benchmark, Quantum Volume can hide architecture- and workload-specific strengths and bottlenecks.

This use is contested by Scott Aaronson, a specialist in complexity theory and quantum algorithms [2090]. His criticism is that a single aggregate number can obscure the underlying hardware characteristics and that demonstrated Quantum Volumes remained classically tractable. The classical-verification issue arises when the passing circuit width $m = \log_2(QV)$ becomes large, not when the numerical QV itself exceeds 50. Scott Aaronson believed that this quantum volume indicator, which is a marketing simplification tool from IBM, should be avoided. He prefers a description of systems characteristics like the number of qubits, their connectivity, their coherence time (T_1 , T_2), their reset, one and two-qubit gates and readout fidelities. With most vendors, these indicators are generally found in the scientific publications of researchers but not always in the vendors marketing literature. However, IBM publishes most of this data on their quantum systems available on Q Experience so you can have it all.

A Cisco Quantum Lab team first posted in 2022, then published in 2023, an extension of Quantum Volume to measurement-based photonic quantum processors [2042]. The framework maps physical noise in a continuous-variable cluster state with Gottesman-Kitaev-Preskill encoding to logical gate error channels, then evaluates circuit-oriented metrics from the effective logical model. At a lower level, a volumetric benchmark for entangled graph states evaluates multipartite entanglement quality across a qubit set [2056]. Such benchmarks address different layers of the stack and are not interchangeable figures of merit.

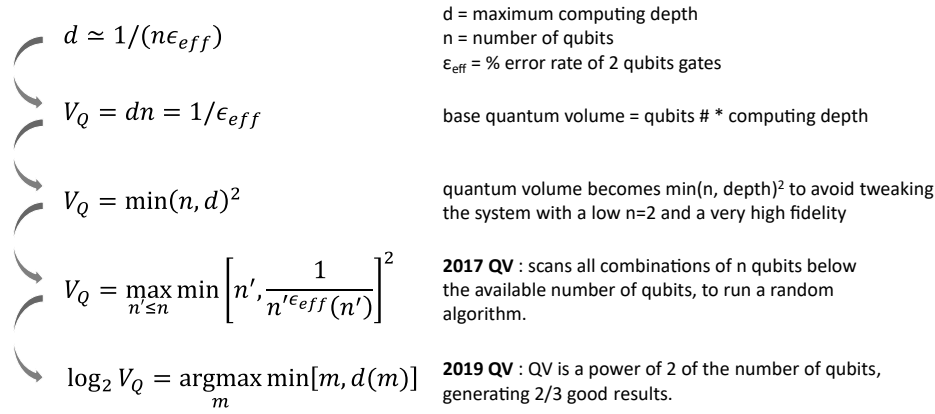


Figure 144: How is/was IBM's 2017 and 2019 quantum volumes calculated. © Olivier Ezratty, 2021-September 2026.

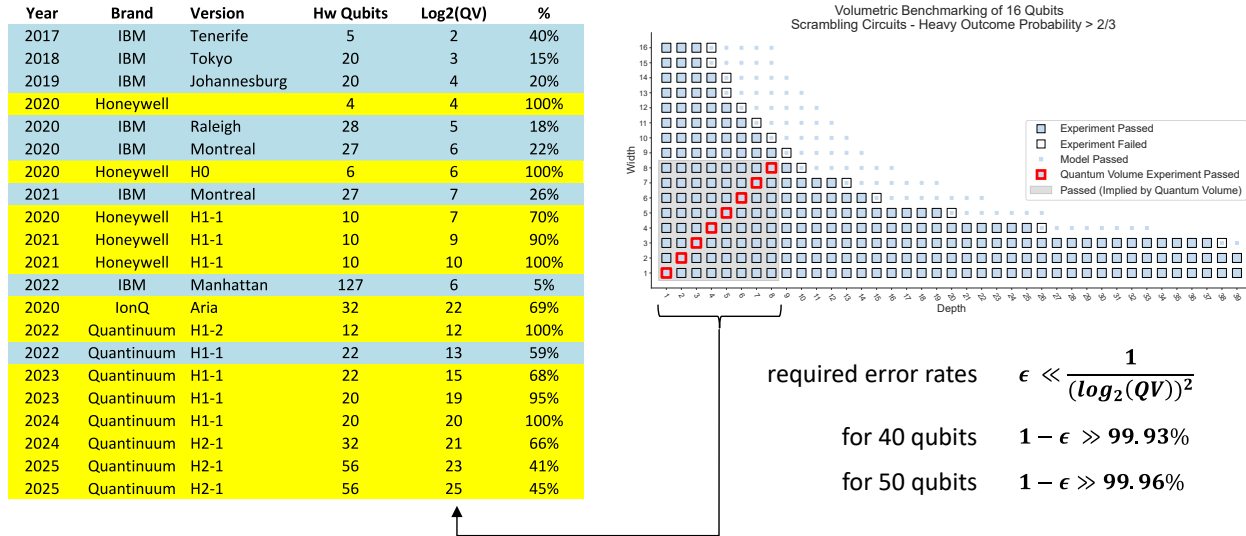


Figure 145: Evolution of systems quantum volumes over time and better visualization of how a quantum volume is evaluated. Source: [2087]. © Olivier Ezratty, 2021-2025

2.5.4 IBM CLOPS

In November 2021, **IBM** introduced CLOPS, circuit layer operations per second, as a system-throughput metric for repeatedly executing parameterized Quantum Volume-style circuits [2043]. It is not a direct analogue of a CPU clock frequency because it includes QPU execution together with parts of the classical control and parameter-update loop. In the original methodology, $CLOPS = MKSD/T$, where M is the number of circuit templates, K the number of parameter updates, S the number of shots, D the circuit depth and T the total measured execution time [2044].

IBM's 2022 roadmap associated higher CLOPS with faster execution and tighter classical-quantum integration. Dynamic circuits add mid-circuit measurement, conditional operations and feedforward. They do not by themselves create long-range qubit connectivity or eliminate the routing gates required by a processor's coupling graph.

An extension of QV is the proposed method "feature-metric benchmarking" by USA researchers [2091].

In 2023, an IBM team proposed a definition of what is a good benchmark insisting on its randomness and platform independence, and what are the appropriate and non-appropriate benchmark optimizations techniques to obtain a good QV [2092]. Another team, led by the Unitary Fund, did a benchmark evaluating the impact of a quantum mitigation technique, ZNE, on the quantum volume of four IBM QPUs. It increased it by a power of 2, or in $\log_2()$ parlance by only one qubit [2093].

IBM published a preprint on a new benchmarking technique named Layer Fidelity (LF), adapted to >100 qubits QPUs, different from their quantum volume. It considers the effect of crosstalk between qubits and can help assess QPUs capabilities when using quantum error mitigation techniques [2047].

2.5.5 IBM EPLG

IBM introduced **Layer Fidelity** in 2023 as a processor-scale randomized-benchmarking metric and **EPLG**, error per layered gate, as a gate-level quantity derived from it [2046, 2047]. They are not two names for the

same metric. For an N -qubit chain in IBM’s definition, $\text{EPLG} = 1 - \text{LF}^{1/(N-1)}$. Layer Fidelity is sensitive to errors that appear when disjoint two-qubit layers are executed across a connected subset, including crosstalk. IBM simultaneously introduced CLOPS_h , whose hardware-aware layer contains only two-qubit gates that can actually run in parallel on the device architecture [2045].

2.5.6 QLOPS

QLOPS, quantum logical operations per second, is a proposed FTQC hardware benchmark created in 2026 that estimates logical-computation throughput while explicitly incorporating fault-tolerance overheads [2058]. It is not simply a logical-gate clock rate. The framework depends on the number of logical qubits, syndrome-extraction cycle rate, code and logical-operation implementation, physical error parameters, and classical decoder throughput and latency. Application runtime still requires a separate logical resource estimate, including the operation mix, circuit structure and number of repetitions required by the algorithm.

2.5.7 IonQ algorithmic qubits

IonQ introduced “algorithmic qubits” ($\#AQ$) as an algorithm-oriented alternative to random-circuit metrics such as Quantum Volume. Its formal definition is volumetric rather than the simple mnemonic $\min(\#\text{qubits}, \sqrt{\#\text{gates}})$ [2070]. For each circuit c in the benchmark suite, the width w_c is the number of qubits and the depth coordinate d_c is the number of CX gates. IonQ defines $\#AQ = N$ as the largest box with $w_c \leq N$ and $d_c \leq N^2$ for which every included circuit satisfies the statistical success criterion $F_c - \epsilon_c > 1/e \simeq 0.37$ [2070]. The benchmark therefore depends on the selected algorithm suite, its fidelity measure and the pass threshold. It has no exact conversion to Quantum Volume.

The volumetric plot in Figure 146 visualizes each benchmark circuit by width, CX-count depth and measured fidelity. A single $\#AQ$ value is extracted from the largest fully passing $N \times N^2$ box.

IonQ reported $\#AQ20$ for Aria in 2022. The value means that all benchmark instances lying inside the corresponding width/depth box passed the criterion. Higher gate and SPAM fidelities can expand the passing region, while qubit count sets an absolute bound on the width. Because the benchmark suite can evolve, $\#AQ$ values should be compared only when the benchmark release and test protocol are specified.

2.5.8 Other systems level benchmarks and metrics

Other various low-level systems benchmarks are worth mentioning:

Cycle benchmarking, introduced in 2019, is a scalable protocol for estimating the process fidelity of repeated multi-qubit gate cycles and characterizing context-dependent errors. It is not specifically an entanglement-quality benchmark [2051].

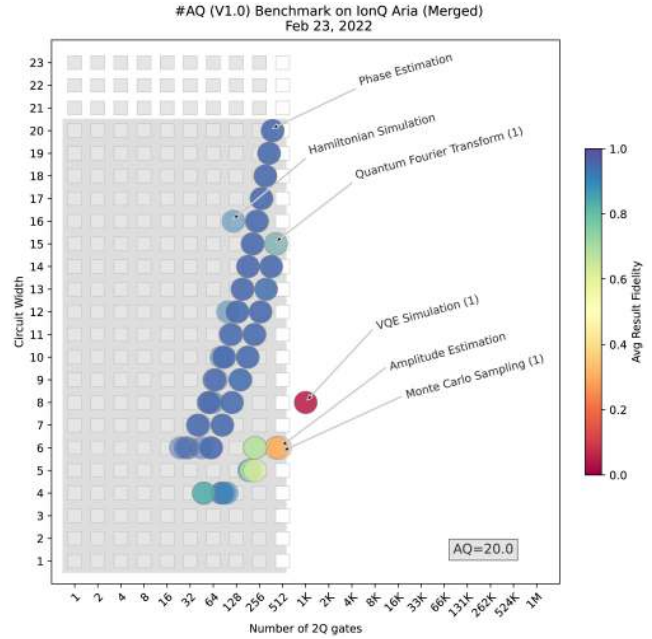


Figure 146: IonQ’s volumetric plot in 2022. Source: [2070].

A proposal made by a team from QuSoft (the Netherlands), the University of Cambridge (UK) and Caltech (USA) in April 2021 is bound to measure the performance of universal quantum computers in a hardware-agnostic way with six structured circuit tests (Bell test, Schrödinger’s microscope, Mandelbrot, line drawing, matrix inversion and platonic fractals). It is quite complex to interpret and reading out the graphical results is not straightforward, nor connected to an application need [2054].

Sandia Laboratories proposed a variation of randomized benchmarking that works in the quantum advantage regime [2094–2097].

Oak Ridge National Laboratory and several US universities proposed in 2022 a volumetric benchmark qualifying the quality of qubit entanglement, that was tested first on IBM QPUs [2056].

University of California Berkeley, HRL Laboratories and the **University of Chicago** devised randomized-circuit methods that probe noise beyond purely Clifford circuits [2098]. Non-Clifford resources are essential for universal fault-tolerant quantum computation and for many algorithms, but their presence alone does not imply exponential speedup. Other extensions include universal randomized benchmarking and

scalable protocols supporting universal gate sets [2050, 2099].

V-score is a dimensionless variational-accuracy metric for many-body ground-state calculations, $V = N \text{Var}(E)/(E - E_\infty)^2$, proposed and validated on a broad benchmark dataset [2100]. Lower values correspond empirically to more accurate variational states for the model classes studied. The work identifies several two-dimensional frustrated systems as difficult targets for current classical variational methods, while one-dimensional systems are often handled efficiently by tensor-network methods such as DMRG.

In the NISQ realm, **PyQBench** is a benchmarking tool proposed by the Institute of Theoretical and Applied Informatics from the Polish Academy of Sciences [2101].

Let’s also mention a framework for constructing predictive models from benchmarking data, and two case studies demonstrating how to do so with error models and a machine learning network [2102] and convolutional neural networks based model to predict NISQ QPU capabilities [2017].

There are also benchmark proposals to assess long-range entanglement [2053, 2103], which can be based on large GHZ states [2052], and using peaked circuit sampling, meaning, the tested circuit generates a high concentration in one entry from the computational basis and does not require costly shot-based estimation. It does not tell anything, though, on the usefulness of such sampling circuits [2104].

A broader question is whether benchmarking results are reproducible. A German team manually audited a curated sample of 127 quantum computing papers on code availability, environment specification, documentation, hardware description and executability, finding that only 24.4% provided code artefacts and that 64.5% of those failed to run in a clean environment. An automated large-scale screening of close to 5,000 papers corroborated this with a 26.8% code-availability rate [2105]. This shows why vendor or paper benchmark results cannot be taken at face value.

Quantum Fidelity-per-Cost (QFC) is a metric proposed by Yale researchers that combines the Kullback-Leibler divergence from an ideal output distribution, the shot count and the monetary cost of an execution into a single cross-provider score. It was evaluated across 14 cloud QPU access paths with 12 distinct physical quantum computers on AWS, IBM, IQM and OQC, turning hardware comparison into an explicit purchasing decision rather than a pure physics exercise [2106].

CrossBench is a 2026 proposal to generate generalized crosstalk benchmarks for quantum computers, echoing the crosstalk sensitivity already built into IBM’s EPLG metric described above but aiming for a platform-independent test [2107].

Two 2025-2026 proposals extend the quantum-volume family of volumetric benchmarks. **Clifford Volume**

and **Free Fermion Volume** are complementary benchmarks from a Wigner Research Centre and AQT team that test random Clifford and free-fermion operations, chosen because both classes are efficiently simulable classically, enabling verification at scale, together form a universal gate set, and serve as primitives in shadow tomography and quantum chemistry [2057]. A separate Japanese proposal defines a related notion of “quantum anomaly” as a benchmarking quantity for quantum computers [2108].

A UCL team proposed in 2025 an alternative to task-based benchmarks, testing violations of macrorealism, the classical assumption that a system has definite properties measurable without disturbance, via two consecutive mid-circuit parity measurements on N qubits. Because the predicted violation is independent of N under ideal conditions, the quantum-to-classical transition observed as N grows becomes a scalable, task-agnostic quantumness metric. It was demonstrated on an IBM QC up to $N = 38$ qubits, about an order of magnitude more qubits than earlier tests of this kind [2109].

A collaboration involving Duke University, PNNL and others implemented a 14-vertex graph-coloring non-local game on four trapped-ion processors, a benchmark whose classical performance bound is information-theoretic and holds unconditionally, regardless of the classical computer’s resources. One system exceeded the classical bound with statistical significance, the first such violation for this game type on real hardware, while the other systems matched the best superconducting results on the same game [2110].

A cautionary result worth flagging to anyone building or trusting entanglement benchmarks: a Nordic team showed in 2026 that arbitrarily small, adversarially chosen measurement errors can produce a false positive certification of high-dimensional entanglement, undermining the usual assumption that experimental measurements are implemented exactly [2111].

In the FTQC realm, various benchmarking techniques are proposed which consider the mid-circuit qubit measurements that are used in quantum error correction [2112, 2113]. Two 2026 benchmark suites target this regime more specifically: one evaluates logical computation under hardware-motivated and biased noise models [2114], and a companion suite from an overlapping author team benchmarks fault-tolerant compilation and architecture choices together [2115]. In that regime, benchmarking should not mandate some classical emulation for comparisons, which may be inaccessible. It may though require a full stack approach, starting with the application circuit, before its compilation [2116]. Quantum error correction codes can also be benchmarked [2117] and even magic state distillation [2118].

The Q-score procedure

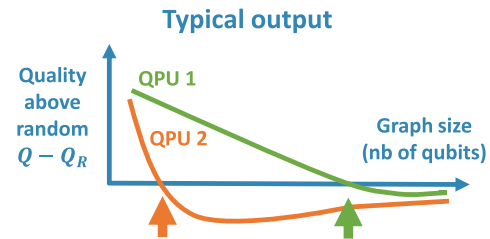
For a **given QPU**. For increasing graph size N :
Get average quality (value of MAXCUT cost function) $Q_R(N)$ of a random solver.

Repeat $P = 500$ times:

- Pick a random (Erdős-Rényi) graph G_N of size N
- Apply **QAOA** procedure with **COBYLA** optimization (random init.) and **MAXCUT** cost function H , get quality $Q = \langle \psi | H | \psi \rangle$ of final state of optimized circuit
- Return quality $Q(G_N)$

Average over the P qualities $Q(G_N)$ to get average $Q(N)$.

As soon as the quality becomes lower than random ($Q(N) \leq Q_R(N)$ with statistical confidence) and under a time limit, return N .
This is the Q-score.



The quality above random usually decreases with the number of qubits because of decoherence in NISQ processors.

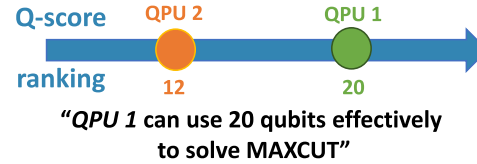


Figure 147: Bull/Eviden/Atos Qscore calculation method. Source: Atos. 2021.

2.5.9 Hardware specific benchmarks

Some very specialized benchmarks are proposed that target specific quantum hardware platforms. These are mostly research tools not bound to be used in the industry. One may still need to agree on the question being asked and what comparisons are made and for solving which kind of problem and which success criteria [2119].

For example, a benchmark was proposed by the **University of Sussex**, **Universal Quantum** (UK) and **Qu&Co** (the Netherlands) in 2022 to better compare trapped-ions and superconducting qubits in a FTQC regime for chemical simulations, where the trade-offs between qubit connectivity, fidelities and gate-times are complicated to assess. It shows that in some situations, slower gates are not damaging run-time, provided enough physical qubits are available in the QPU [2120]. But one can wonder whether many-to-many qubit connectivity can itself scale with the number of qubits when using trapped ions.

The **University of Strathclyde** proposed in 2024 a benchmark for gate-based neutral atoms QPUs starting with digitally emulating it using the QuEST toolkit [2121].

In 2023, a team in China proposed to benchmark qubit leakage, a particular error that is not corrected by regular QEC codes [2122].

In 2024, a German team proposed to benchmark particular photon-based systems using non-Gaussian bosonic circuits [2123].

A **Technion** team published in 2026 a series of protocol-based benchmarks that define explicit quantumness thresholds at the protocol level rather than the gate level, so that a chip either clears the threshold or does not. They first compared superconducting and ion-trap technology on their best sub-chips [2124], then applied the same methodology to compare two generations of IBM

hardware, the older Eagle and the newer Heron architectures [2125]. This protocol-level approach sidesteps the need for classical emulation while still giving a common, hardware-agnostic language for comparison.

HyQBench is a 2026 benchmark suite for hybrid continuous-variable/discrete-variable (CV-DV) quantum computing from North Carolina State University and Rutgers, implemented with Bosonic Qiskit and QuTiP. It covers eight representative circuits across three abstraction levels, from primitives (cat-state and GKP-state preparation) to algorithms (CV-DV QFT, VQE, QAOA) to applications (Jaynes-Cummings-Hubbard simulation, Shor's integer factoring), and one benchmark (cat-state preparation) was validated on Sandia's QSCOUT trapped-ion platform with a measured fidelity of 0.71 [2126].

2.5.10 Eviden Q-score

Atos (Bull since 2026) proposed its Q-score benchmark in December 2020. It is based on determining the maximum size of a standardized problem that can be solved on a given hardware on any quantum programming paradigm [2059]. The first selected problem is the classical combinatorial **Max-Cut** (Figure 147). Its variations are used to solve the traveling salesperson problem or various graphs problems with applications in logistics, industry and finance. It can also be used to handle clustering in quantum machine learning. The Q-score benchmark is evaluated by using a hybrid classical+quantum algorithm like with QAOA (Quantum Approximate Optimization Algorithm).

This benchmark creates a simple metric based on the largest standardized Max-Cut instance for which the prescribed performance criterion is met, and it can be applied across different computing paradigms [2059]. Evaluating the cut value of a candidate bit string is

polynomial in the graph size, so benchmark outputs can be scored efficiently. Certifying that a candidate is the exact optimum of a general Max-Cut instance is not polynomially tractable unless standard complexity assumptions fail. The Q-score software tools are open-source and published on [GitHub](#).

In August 2022, a Dutch team evaluated **D-Wave** 2000Q and Advantage annealers and obtained a record of 70Q and 140Q, which is much better than existing gate-based QPUs [2060, 2127]. The same team proposed a Q-score extension with a Max-Clique combinatorial problem [2061, 2128].

Hybrid solutions may under some conditions outperform particular classical baselines. The Q-score values quoted in this section are not a ranking: before comparing them, the hardware, the solver type, whether the run used a physical device or a simulator, the time budget and the success criterion all have to be stated, and these differ across annealers, gate-based devices and classical digital simulators. A team from **Pasqal** implemented the Q-score workflow on a digital simulator of its neutral-atom architecture and reported a Q-score of 80Q [2129]. The study used classical machine learning to reduce the number of quantum-model evaluations. This result was obtained with a simulator.

In October 2023, the first Q-Score was published by a gate-based quantum computer vendor, IQM, with 8Q obtained with their latest 20-qubit QPU [2130].

The Q-score would need to be completed by other benchmarks such as one for quantum chemistry simulation (number of atoms in molecules) and another on the bit length of the largest integer that can be factored under a specified time and success criterion. It may happen as part of the BACQ project mentioned later.

2.5.11 Other application level benchmarks

Many application-oriented benchmarks have been proposed in the last few years (Figure 148) [2078, 2131].

The DoE **ORNL** (Oak Ridge National Laboratory) proposed in 2020 a benchmark for chemical simulation [2062]. It uses three two-atom molecules, NaH, KH and RbH, on then-available IBM and Rigetti superconducting systems. The proposed benchmark set is narrow and is not a generic molecular-simulation benchmark.

QED-C is supporting a series of application oriented benchmarks proposed by researchers from Princeton, HQS, QCI, IonQ, D-Wave and Sandia Labs in the USA [2067]. It mixes the volumetric benchmarking method from IBM and a comparison of performance with various standard algorithms. They did some comparisons on actual quantum hardware from IBM, Rigetti, HQS and IonQ. It was launched in October 2021. In 2023, they extended their standard proposal with an optimization algorithms benchmark supporting NISQ QAOA and quantum annealing [2068]. In 2024, researchers proposed an extension of the QED-C benchmarks with a

VQE implementation of a Hydrogen Lattice simulation, of a supervised machine-learning classification applied to a binary classification problem and including methods to include optimization and error mitigation in the benchmarking workflow [2069].

SupermarQ is a scalable, hardware-agnostic benchmark suite introduced by researchers from Princeton, Super.tech, the University of Chicago, Intel and Northwestern University and published at IEEE HPCA in 2022 [2071]. It uses representative application-level circuits and feature vectors to characterize workload coverage, and it was evaluated on several quantum hardware platforms. Its value is primarily as a suite rather than as a single-number ranking.

QuAS, the Quantum Application Score, is an application-level benchmark proposed by TU Delft researchers [2073]. It combines the Q-score and their QPack benchmarking set that is based on the dominating set problem (DSP) and the traveling salesperson problem (TSP) and using QAOA as a benchmarking tool. It measures the maximum problem size a quantum computer can solve, the required computing runtime and the achieved accuracy [2072, 2132, 2133]. As of 2024, QuAS was tested on D-Wave, IBM, Rigetti and the Quantum Inspire cloud platform.

HamLib is a library of qubit Hamiltonian problem instances for benchmarking quantum algorithms and hardware, not a library of ready-to-use circuits [2134]. It contains instances ranging from 2 to 1000 qubits across spin models, fermionic and bosonic models, molecular structure, Max-Cut, SAT variants and the traveling salesperson problem.

Many-body Quantum Score (MBQS) is a scalable application-level benchmark from a CEA/Eviden Quantum Lab team, applicable to both gate-based and analog QPUs. It quantifies the maximum number of qubits for which a processor can reliably reproduce correlation functions of the transverse-field Ising model after a quantum quench, and it was first tested on a commercial neutral-atom device [2065]. A related, narrower proposal standardizes a test of many-body coherence specifically for gate-based platforms [2135].

QuantumBench is a 2025 benchmark for general quantum problem solving proposed by a Japan-based team including IBM Quantum and Fujitsu-linked researchers [2136].

Measuring what matters is a scalable application-level benchmarking framework proposed by researchers formerly at IonQ and Microsoft, positioned, like QED-C, as a general methodology rather than a single fixed test [2137].

A Max-Planck/Barcelona team proposed in 2026 a noise-aware benchmark of quantum algorithms for ground-state preparation, directly relevant to the VQE and quantum chemistry use cases discussed throughout this book [2138].

On the simulator side, two 2026 proposals benchmark quantum circuit simulation itself rather than physical hardware. **InferQ**, from Delft University of Technology, is a database-oriented benchmark that compiles quantum circuit simulation into SQL workloads executable by relational database engines, releasing a dataset of over 200,000 circuits for systematic query-optimization research [2139]. A separate proposal from the University of Washington and University of New Mexico instead targets the verification side of quantum simulation, using classically simulable “stabilizer scars” to bound the fidelity of otherwise classically intractable non-equilibrium simulations, a benchmark aimed squarely at quantum-advantage claims in this domain [2140].

DARPA launched its **Quantum Benchmarking** program in 2021 to quantify progress toward computational applications with economic utility and to estimate the hardware-specific resources required to execute them. The program developed application benchmarks and end-to-end resource-estimation tools for quantum computing, with detailed studies in areas including chemistry, materials and nonlinear differential equations [2082]. The program is now complete.

DARPA’s **Quantum Benchmarking Initiative** (QBI), announced later, is a distinct program that evaluates whether candidate fault-tolerant quantum-computing approaches could reach industrially useful scale.



Another similar approach, but a narrower one, was proposed by researchers from **Brookhaven** and **Pacific Northwest** National Laboratories from the DoE which estimates hardware resources needed for key algorithms [2141].

HamilToniQ (UK) is an application-oriented benchmark focused on QAOA with quantum error mitigation in the NISQ regime [2064].

The **BACQ project**, launched in France in 2023, is a high-level application-oriented benchmarking project that also uses low-level technical figures of merit [2085]. The considered problem classes include optimization, linear-system solving, quantum-physics simulation and integer factorization. The benchmark considers figures of merit such as computation time, latency, problem size, approximation ratio, success probability, accuracy, fidelity and power consumption. It is intended to cover gate-based NISQ and FTQC systems as well as analog quantum processors. The project is run by Bull, CEA, CNRS, Thales, Teratec and LNE.

Other use-cases benchmarks have also been created: Agnostiq created a benchmark dedicated to optimizing financial portfolio using the Quantum Approximate

Optimization Algorithm (QAOA) [2142], German researchers looked at industry applications [2143], and Zapata Computing created one benchmarking on fermionic simulations [2063].

There are other benchmarks for Hamiltonian simulations [2144], QAOA [2145], VQE [2146, 2147], Fermi-Hubbard simulations [2148], materials design [2149], quantum optimization algorithms [868, 2150–2154], Grover searches [2155–2157], quantum machine learning [2158–2163], reinforcement learning [2164], image recognition [2165], solving TSP problems [2166], for the DAQC hybrid analog/gate-based quantum computing paradigm [2167], for analog simulations of bosonic and fermionic dynamics [2168], and even for comparing classical reinforcement learning methods used for tuning variational circuit ansätze [2169].

On benchmarking quantum machine learning algorithms, several researchers warn of various benchmark biases in QML as well as in classical machine learning can be unduly tweaked to one model’s advantage [466, 1163, 2170]. It is not a surprise, then, that most multiple-application benchmarks do not include some QML applications by default but focus more on chemical simulation and optimization problems.

2.5.12 International standard organizations

Many standards organizations are engaged in quantum technologies, well beyond benchmarking. International bodies such as ISO and IEC are organized through national standards bodies, while IEEE standards projects are organized through sponsor committees and working groups. In Europe, ETSI and CEN/CENELEC also develop quantum-related standards [2171].

IEC and ISO created **IEC/ISO JTC 3 Quantum Technologies** in 2024, with a scope spanning quantum computing and simulation, metrology, sources, detectors and communications. Quantum-information-technology work previously carried by ISO/IEC JTC 1/WG 14 was transferred to JTC 3.

IEEE maintains several active quantum standards projects. Relevant examples include P2995 for quantum algorithm design and development, P3120 for quantum computing architecture, P1913 for software-defined quantum communication, P7130 for quantum-technologies definitions, P7131 for quantum computing performance metrics and performance benchmarking, and P3329 for quantum computing energy efficiency. In September 2026, P7131 and P3329 remain active projects rather than completed 2024 standards.

The **EU Quantum Flagship** published in December 2025 its own suite of scalable benchmarks, developed as Key Performance Indicators (KPIs) and designed to remain applicable across both NISQ and future FTQC architectures. It defines four core benchmarks: large multi-qubit circuit execution via the Clifford Volume benchmark described above, scalable multipartite en-

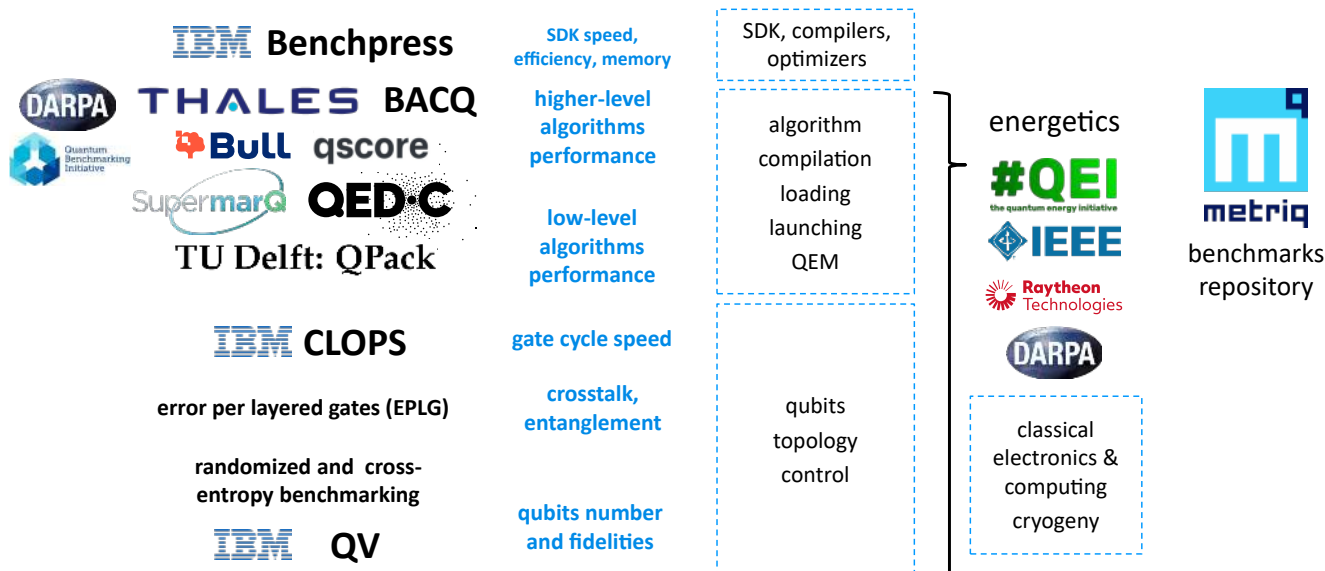


Figure 148: Various levels of abstraction in quantum computing benchmarking. © Olivier Ezratty, 2023-September 2026.

tanglement generation through GHZ-state preparation, a benchmark based on Shor's period-finding subroutine applied to simple functions, and a Bell-state-based protocol quantifying the benefit of quantum error correction [2079].

CEN/CENELEC established **CEN-CLC/JTC 22 Quantum Technologies** in 2022. It builds on the earlier Focus Group on Quantum Technologies, which operated from 2020 to 2023.

ITU-T's Focus Group on Quantum Information Technology for Networks (FG-QIT4N) operated from 2019 to November 2021 and produced nine technical reports. It is no longer an active focus group, although quantum-network standardization continues in ITU-T study groups.

ETSI is also active in areas including QKD and post-quantum cryptography.

2.5.13 Benchmarking tools

Metriq from **Unitary Fund** was announced in May 2022 and formally introduced in March 2026 [2080]. It is a repository of benchmarking results which showcases interesting charts making them easy to grasp. The Unitary Fund then launched in June 2024 the Open Quantum Benchmark Committee to standardize quantum computing benchmarks with Metriq. It has the initial participation of Indiana University, PSI in Switzerland, University of Milano-Bicocca, Pasqal, Thales, Nvidia, IBM, the University of British Columbia, qBraid, QPerfect and AWS.

MQT Bench aka the Munich Quantum Toolkit is a multiple-abstraction level suite of benchmarking tools covering both low-level abstraction building blocks (QFT, QPE, amplitude estimation) and higher-level ones (Grover, Shor, HHL) [2074].

Arline (Germany) is also proposing its own benchmarking tools suite that is used to compare compiler optimizers [2172].

QUARK (Germany) is an industry application-oriented benchmarking proposal based on an open-source framework, from the QUTAC consortium members including BMW, Capgemini, Fraunhofer and German universities [2075]. It covers combinatorial optimizations and quantum machine learning case studies [2076].

Benchpress is a benchmarking tool proposed by IBM in 2024 to evaluate SDKs and mainly compilers and optimizers. Seven SDKs were evaluated using 1066 code tests. Qiskit, QTS (Qiskit Transpiler Service extension, also from IBM) and Tket from Quantinuum fared best in this benchmark [2081].

A tool-oriented complement to fixed benchmark suites is software that helps researchers build their own. A 2025 open-source tool lets users create scalable benchmarks directly from quantum algorithms rather than relying on a fixed, pre-selected test suite [2173].

Software quality itself is also becoming a benchmarking target rather than just hardware. A Chinese team proposed in 2026 a benchmark for quantum software testing based on scalable quantum programs, extending the classical software-testing benchmarking tradition to the quantum stack [2174].

QuSquare is a 2025 quality-oriented benchmark suite from the same University of the Basque Country/BCAM group behind [2078]. It targets the pre-FTQC systems specifically, with four tests spanning system and application levels: partial Clifford randomized, multipartite entanglement, transverse field Ising model Hamiltonian simulation and a data re-uploading quantum neural network test, which seek relevance, reproducibility, fairness, verifiability and scalability as the group defined in its earlier standardization proposal [2077].

2.5.14 Quantum supremacy and advantage

Quantum supremacy, advantage and utility are various terms used to qualify the superiority of quantum computers as compared with the most powerful supercomputers. They must be defined carefully.

Quantum supremacy was an expression coined by John Preskill in a paper presented at the Solvay Congress in 2011 [2175]. It is achieved when a problem, useful or not, can be solved only with a quantum algorithm running on a quantum computer, and there is no known classical algorithm for the most powerful supercomputers that could solve it within a humanly relevant time [2176]. Quantum supremacy was a goal for researchers and vendors like Google and it became claims starting in October 2019.

Quantum supremacy does not mean that a given quantum computer is supremely more powerful than all its contemporary supercomputers. The term is applicable to a combination of a specific problem and related quantum algorithm, a given quantum computer, and the best-in-class available classical algorithms adapted to the most powerful available supercomputers. The criteria are moving targets. Supercomputers have not said their last word, and classical algorithms are also being improved [2177, 2178].

Robert König (Technical University of Munich), David Gosset (University of Waterloo, Canada) and Sergey Bravyi (IBM) demonstrated in October 2018 that quantum computers could perform operations inaccessible to conventional computers but based only on the case of a particular algorithm [938, 2179].

Others are devising “*proofs of quantumness*”, interactive or cryptographic protocols intended to convince a classical verifier that a prover has quantum computational capability [2180]. The relevant soundness statement is normally formulated against classical probabilistic polynomial-time adversaries, often under explicit cryptographic assumptions. “Less than polynomial time” is not the definition of tractability. Polynomial-time computation is the standard complexity-theory notion of efficient computation.

Some D-Wave and Google benchmarks carried out in 2015 and showing the superiority of the quantum solution were then contradicted by the creation of algorithms optimized for supercomputers under certain conditions. In a few years, it will certainly come into play for a few algorithms that cannot have optimized supercomputer equivalents.

Google’s 2019 quantum-supremacy experiment used a 53-qubit Sycamore processor to perform random circuit sampling, with linear cross-entropy benchmarking (XEB) used to estimate the fidelity of the sampled distribution [2049]. The reported full-circuit XEB value was about 2×10^{-3} , or 0.2%. This is a fidelity-like correlation metric, not a 0.2% probability of obtaining a

“good” answer. Sycamore generated one million samples in about 200 s and three million in about 600 s.

Google initially estimated that comparable classical sampling would require roughly 10,000 years on Summit, while IBM argued shortly afterward that a different classical strategy could reduce this to about 2.5 days [2181]. Classical algorithms and hardware subsequently improved dramatically. A peer-reviewed 2025 National Science Review paper generated three million uncorrelated samples at XEB 2×10^{-3} in 86.4 s on 1,432 Nvidia A100 GPUs, consuming 13.7 kWh [2182]. This provides a measured classical result faster than the original Sycamore experiment. The six-second classical runtime refers to resource estimates for a different optimized simulation scenario.

Cristian and Elena Calude of the University of Auckland in New Zealand then argued that a high-performance limit, that of a precise quantum computer, is compared to a low limit which is the best performance in solving the same problem in a supercomputer [2183].

Quantum supremacy is thus a comparison between the demonstrated performance of a quantum computer and the assumption that no equivalent task can be achieved on a classical computer. On top of that, the supremacy claim was of course misunderstood for a quantum advantage, up to leading some to think that quantum computers were not far from breaking RSA keys. The disappointment came in 2020 when Google published several papers experimenting with Sycamore for real use cases like in computational chemistry. These cases couldn’t use more than 20 qubits and were as a result far off any quantum practical advantage as compared with best-in-class classical solutions.

Then, many researchers tried to simulate Google’s supremacy classically using various breeds of tensor networks and compression techniques.

A 2020 paper from Yiqing Zhou (University of Illinois), Edwin Miles Stoudenmire (Flatiron Institute) and Xavier Waintal (CEA-IRIG) stated that emulating Sycamore’s processes on a classical computer could use some compression technique to take into account the qubit’s noise. It could even be done on a simple microcomputer [2184]. This corresponded, however, to a 95% emulated gate fidelity. With a 99% fidelity matching Sycamore’s capability, it would still require a couple hundred cores and some TB of memory, fitting in a datacenter rack. There would be an energy advantage for Sycamore but going down from $500,000\times$ vs the IBM Summit to about only $\times 325$ for a 500-core cluster server.

A research team from China published an even better performance in 2021, using an improved tensor contraction technique and running for 15 hours on a cluster of 512 GPUs [2185]. In 2021, a China team classically simulated the Google Sycamore cross-entropy benchmark with a single Nvidia A100 GPGPU running for 149 days, reporting a simulation fidelity of 73.9% for the amplitudes it computed, while the Sycamore experi-

ment reported an XEB fidelity of about 0.2%. The two percentages are not the same quantity and should not be compared directly [2185, 2186].

Another team, from Alibaba, found a way to optimize Sycamore’s emulation in September 2021 to reach only 20 days of computing on a system equivalent to the IBM Summit [2187]. And yet another one in October 2021 could simulate it on the new Sunway supercomputer in 304 seconds, using a tensor compression technique [1673]. The China team behind this was awarded the 2021 ACM Gordon Bell Prize. Their work was later contradicted by ORNL researchers who had developed Google’s supremacy classical simulation in 2019 [1674]. Late 2022 until 2024, another compression technique was tested using 41,932,800 cores for 8.5 days [2188]. Meanwhile, other China academics explain how to tweak random quantum circuits benchmarks to obtain a quantum supremacy [2189].

In summer 2022, Xavier Waintal, Edwin Miles Stoudenmire and collaborators including Thomas Ayrar showed that noisy random circuits in the Sycamore regime can become substantially easier for tensor-network methods because noise limits the effective entanglement and fidelity that must be reproduced [2190, 2191]. Their favorable scaling applies to the noisy, finite-fidelity regime and to the approximation criterion studied. It is not a general theorem that arbitrary random quantum circuits can be simulated in polynomial time. Related complexity-theory work similarly shows that sufficiently noisy random-circuit sampling can lose its asymptotic hardness under specified noise models [2192–2194]. The central point is that increasing circuit size without maintaining sufficiently low error rates does not by itself preserve a computational separation.

Classical simulation claims deserve the same scrutiny as quantum advantage claims. A December 2025 two-author preprint from **Quantum Rings**, a vendor of a classical quantum circuit simulator, reported a 53-qubit 20-cycle Sycamore circuit sampled with 2.5 million shots on a single Nvidia A100 GPU plus 100 CPU jobs in about 1h15, with a linear XEB score of 0.549 and a claimed speedup of 6.95×10^7 [2195]. But it seemed that these numbers were misleading as the speedup was measured against Google’s own 2019 Sycamore estimate, which was superseded by Willow in 2024, and the XEB comparison opposes an ideal noiseless simulation to a noisy hardware run whose XEB was 0.002 by construction.

In October 2022 and May 2023, Gil Kalai et al. produced an in-depth and well-crafted detailed analysis of Google’s supremacy experiment and related protocols raising doubts on some statistical data [2196, 2197].

Google extended random circuit sampling to 70 qubits in work published in 2024, emphasizing the competition between circuit fidelity and classical simulation cost [2198]. In December 2024, Google reported a random-circuit-sampling experiment on its 105-qubit Willow processor

taking under five minutes. Google estimated that reproducing the benchmark with then-current classical methods on **Frontier**, not Aurora, would require approximately 10^{25} years. This is a classical cost estimate based on a specified simulation model, not a measured supercomputer runtime.

In September 2025, it proposed another use case for a quantum advantage, with some generative quantum advantage, tested on a 68-qubit version of the Sycamore chip, seemingly a 72-qubit chip with 4 defective qubits. One of the proposed tests consisted in creating a quantum circuit that has the same effect as an unknown unitary transformation U , a very theoretical construct given that for most problems, you know how to construct U in the first place like with a quantum phase estimation [2199].

In October 2025, Google announced its **Quantum Echoes** experiment, measuring out-of-time-order correlators (OTOCs) on 65 of the 103 qubits of a Willow chip and claiming a measurement more than 10,000 times faster than the best classical methods it had tested [2200]. One classical method that was not tested at the time was tensor networks with belief propagation (TNBP), which approximates the OTOC by evolving a projected entangled pair state and truncating it in a belief propagation gauge. A Google Quantum AI team with the Donostia International Physics Center closed that gap in April 2026, combining scaling arguments and explicit numerical simulations to show that TNBP cannot reproduce the second order OTOC on 65 qubits at any feasible cost, since the OTOC circuits generate enough entanglement to be largely incompressible [2201]. The argument extends to any Schrödinger picture tensor network evolution. The obvious caveat is that the rebuttal comes from the same organization that made the initial claim, which is not the independent replication that supremacy claims deserve.

In November 2025, **Quantinuum** used its 98-qubit Helios trapped-ion system to measure superconducting pairing correlations in three regimes of Fermi-Hubbard models: non-equilibrium pairing induced by an electromagnetic field in the half-filled square lattice model, d-wave pairing in an approximate ground state of the checkerboard model at 1/6 doping, and s-wave pairing in a bilayer model relevant to nickelate superconductors [2202]. These pairing correlations are off-diagonal and thus not accessible to the local density measurements used in most quantum simulation experiments, which is why they had not been measured before on a quantum computer. This is a quantum simulation result fed with a problem Hamiltonian, so it is closer to a useful advantage than any sampling experiment, but the classical simulability of these specific instances still has to be settled by independent teams.

In March 2026, an **IBM Quantum** team reported the first case where a sample-based quantum diagonalization algorithm outperformed off-the-shelf classical selected configuration interaction heuristics. They con-

structed a class of local Hamiltonians with sparse ground states, showed that representative classical heuristics fail on a specific 49-qubit instance, then solved it with the sample-based Krylov quantum diagonalization algorithm (a variant of SQD) running on an IBM Heron r3 processor, using quantum samples taken from a grid of time-evolved states [2203]. The paper mentions that the same problem is also solvable classically with two iterative solvers that the authors designed specifically to target their Hamiltonian construction. The advantage therefore holds against standard heuristics, not against the best classical algorithm, and the Hamiltonian family was engineered for the quantum method in the first place.

In 2018, IBM researchers demonstrated that quantum supremacy was assured in the long run, even with quantum computers that can chain a finite and constrained number of quantum gates [938]. In December 2020, they published a theoretical model that could prove some quantum advantage, solving binary function problems, and tested on a low scale on a 27-qubit superconducting system [2204]. These various, sometimes convoluted, performances are very hard to compare and evaluate.

Many quantum-supremacy experiments use sampling tasks deliberately chosen to be hard to simulate classically, including random circuit sampling and boson sampling. These experiments do have instance parameters, such as circuit gates or interferometer settings, but they generally do not process externally supplied application data or solve a recognized end-user problem. Computational advantage and practical utility are separate criteria. A mathematically well-defined quantum-classical separation can therefore be valid while having little immediate practical value. Conversely, a useful quantum algorithm must be assessed end to end, including state preparation, data loading, repetitions, classical post-processing and, in the FTQC regime, error-correction overhead.

Another structural weakness of many supremacy claims is the verification bottleneck. Checking that a sampling experiment was correctly executed usually requires a full classical simulation of the ideal computation, which is precisely the task assumed to be intractable, so the verifiable regime and the simulable regime coincide. Proposals are emerging to break this circularity. **Hidden Code Sampling**, from IBM Quantum, the University of Chicago, the University of Maryland and Berkeley, uses a quantum error correcting code to make the output distribution conditionally peaked, so that verification costs far less than a full simulation, with exact sampling shown to be classically hard unless the polynomial hierarchy collapses [2205]. It is still a theoretical construct with no hardware implementation, and its average case hardness relies on a conjecture, as with every random circuit sampling proposal.

We cannot avoid mentioning the debates around the term supremacy with its bad social and political meaning. One replacement proposal hasn't been adopted yet,

which consists of using the simpler term **primacy** with the same meaning [2206].

Q Boson samplings can also be simulated classically

Boson sampling and Gaussian boson sampling (described in the photonic qubits part of this book) are also in the realm of quantum supremacy experiments, particularly in China.

Conversely, a Google team did show that the 2021 Chinese gaussian boson sampling (GBS) supremacy experiment could be efficiently simulated classically (still, quadratically) [2207]. Another team, from the UK, achieved a similar feat in 2022, in which they did simulate a 100-mode and up to 60 click detection events GBS on a $\sim 100,000$ -core supercomputer [2208]. A 2026 paper could simulate a Gaussian boson sampling up to 1,152 modes corresponding to the Jiuzhang 3.0 experiment published in 2023 [2209, 2210].

Quantum advantage is used inconsistently in the literature. In the broad computational sense, it means that a quantum implementation outperforms the best known classical method for a specified task and resource metric. A *practical* quantum advantage normally adds a usefulness requirement and an end-to-end comparison involving time to solution, accuracy, cost, energy or another operational figure of merit. This is generally a stronger engineering requirement than demonstrating a computational separation on a contrived sampling task.

There are currently fewer practical-advantage claims than sampling-based supremacy claims. Some use the term for narrowly defined resource comparisons or in domains beyond quantum computing [2211, 2212].

Negative results are as informative as positive ones and there are few of them in optimization. A July 2026 semiclassical analysis of QAOA on the Sherrington-Kirkpatrick spin glass model, using the truncated Wigner approximation with a large spin S , found that the optimal balance is reached for S of order the circuit depth p , with the final energy converging to the Parisi value like $\log(p)/p$. The semiclassical version, which is a classical simulation, slightly outperforms the true spin-1/2 QAOA, and removing the initial semiclassical noise and reoptimizing the angles even yields a faster $1/p$ convergence, leaving no room for a quantum advantage on approximate spin glass optimization [2213].

On the applications side, published annealing advantages remain fragile. A Korean team benchmarked D-Wave's hybrid constrained quadratic model solver against Gurobi and CPLEX on an Arctic ship routing problem built from Copernicus sea ice data discretized on an H3 hexagonal grid, and reported 10 to 100 times

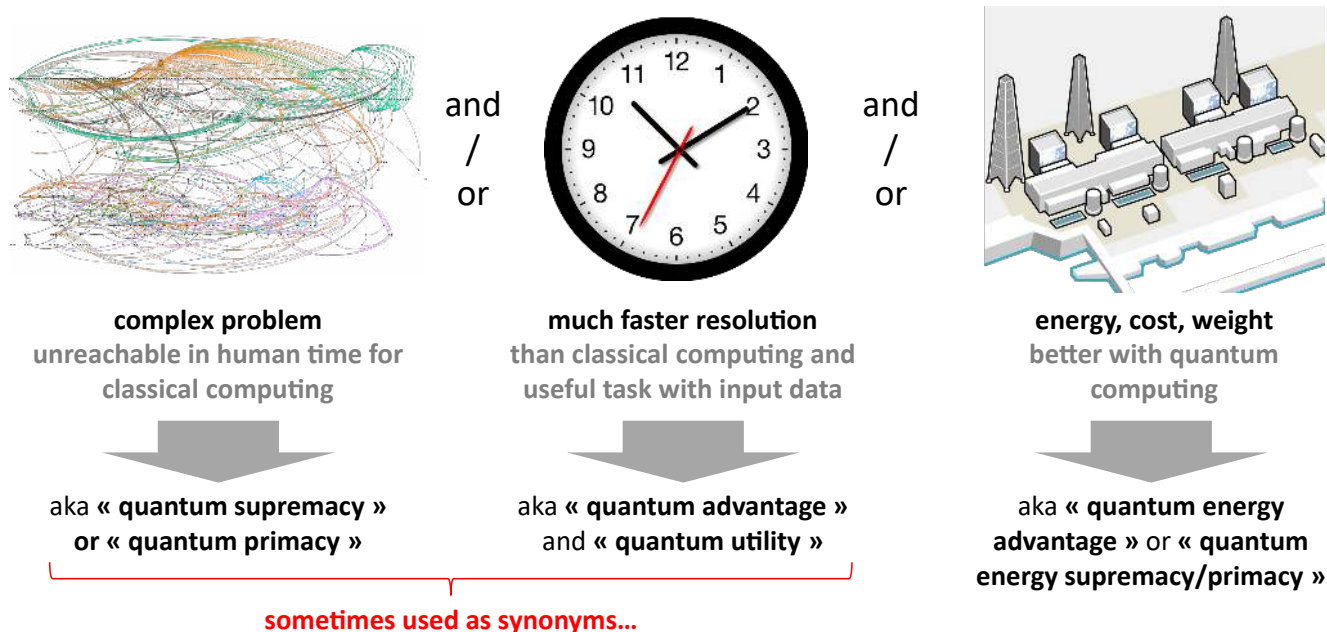


Figure 149: Trying to define quantum supremacy (or primacy) and quantum advantage. © Olivier Ezratty, 2022-2024.

faster convergence, but for route quality gains of about 10% on smoothness and 1% on length [2214]. As is often the case in such comparisons, it opposed a heuristic hybrid solver returning feasible solutions to classical exact mixed-integer solvers that also prove optimality. And D-Wave's hybrid solvers do most of their work classically, so this is a solver engineering result and not a real certifiable quantum advantage demonstration.

One first example comes from an interesting work from a team of researchers from France and Edinburgh announced in February 2021, including Eleni Diamanti and Iordanis Kerenidis [2215, 2216]. It involved a photonics-based experiment that didn't do any real calculation and was about putting in place a QMA (Quantum Merlin Arthur) verification protocol, consisting of an interactive test that requires, through a quantum network, the verification of the solution of a complex NP-complete optimization problem without having to communicate the whole solution. The breakthrough that made this possible was the creation of a system encoding the solution result with partial information about the solution to be verified from one network node to another. The protocol compresses a large vector state describing the partial information on the solution, involving some entanglement and multi-mode photons quantum communications. This compression protocol would make it possible to verify the results in much less time. No actual verification was done on the other end of the system.

We have here a quantum advantage coming from the way to connect a quantum computer solving an NP-complete SAT problem and another quantum computer verifying the solution with partial information. Both computers do not exist yet. Another view on this would be that it proposes an architecture to verify a solution to an NP-problem on an end-to-end solution.

Another example of quantum computing advantage could be reached by feeding a quantum computer with quantum-sensor data transmitted "quantumly" instead of classically [44] (Figure 151). That is what a team from Google and other researchers demonstrated in 2021 and is shown in Figure 150 [741]. This requires some form of quantum memory that is still to be created. It must also consider the effect of noise in the quantum sensors and transmission device [2217].

There is no universal qubit-count threshold for practical quantum advantage. The required scale depends on the problem, algorithm, error rates, connectivity, gate speed, compilation overhead and the classical baseline. In fault-tolerant architectures, the relevant resource is usually the number and quality of logical qubits together with logical cycle time and non-Clifford resource overhead, rather than a raw physical-qubit count. Researchers have also shown that quadratic algorithmic speedups can be erased by large constant factors from error correction and other end-to-end overheads on early FTQC systems [1324].

The quantum advantage label is also used in non-computing domains, like with quantum sensing, when some technique is generating better accuracy or precision than classical sensors.

Quantum advantages will have a significant market impact when they become more general and cover a broad class of quantum algorithms and related situations instead of being narrow and with lots of conditions on the input data (sparse, symmetries, ...).

Quantum utility was introduced by IBM in 2023 for computations at scales where quantum hardware can produce scientifically meaningful results beyond straightforward brute-force simulation, without requiring a demonstrated advantage over the best classical

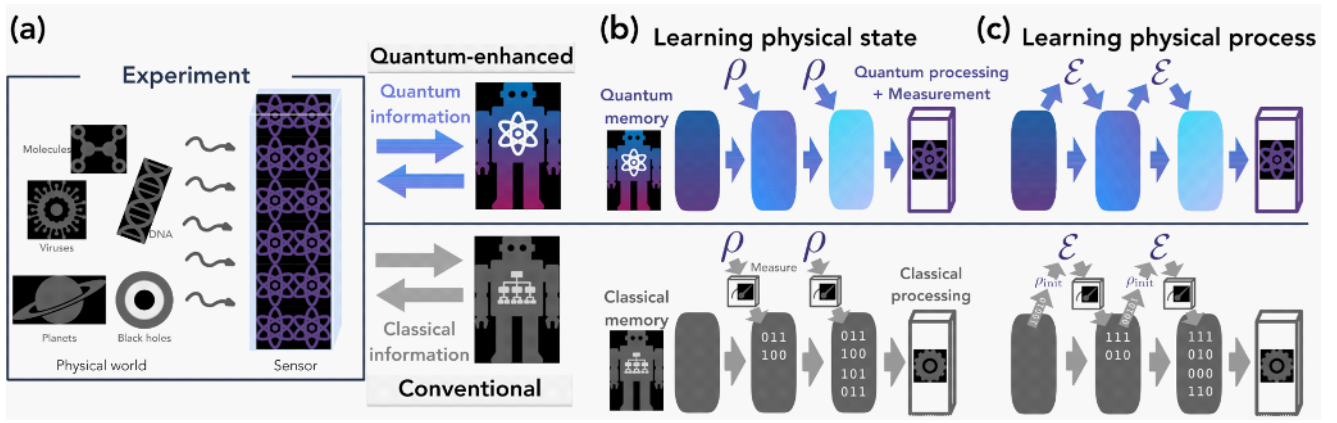


Figure 150: A quantum advantage can come from connecting quantum sensors and quantum computers, avoiding the tedious steps of quantum-to-classical and classical-to-quantum data conversions. Source: [45], with 40 Sycamore qubits.

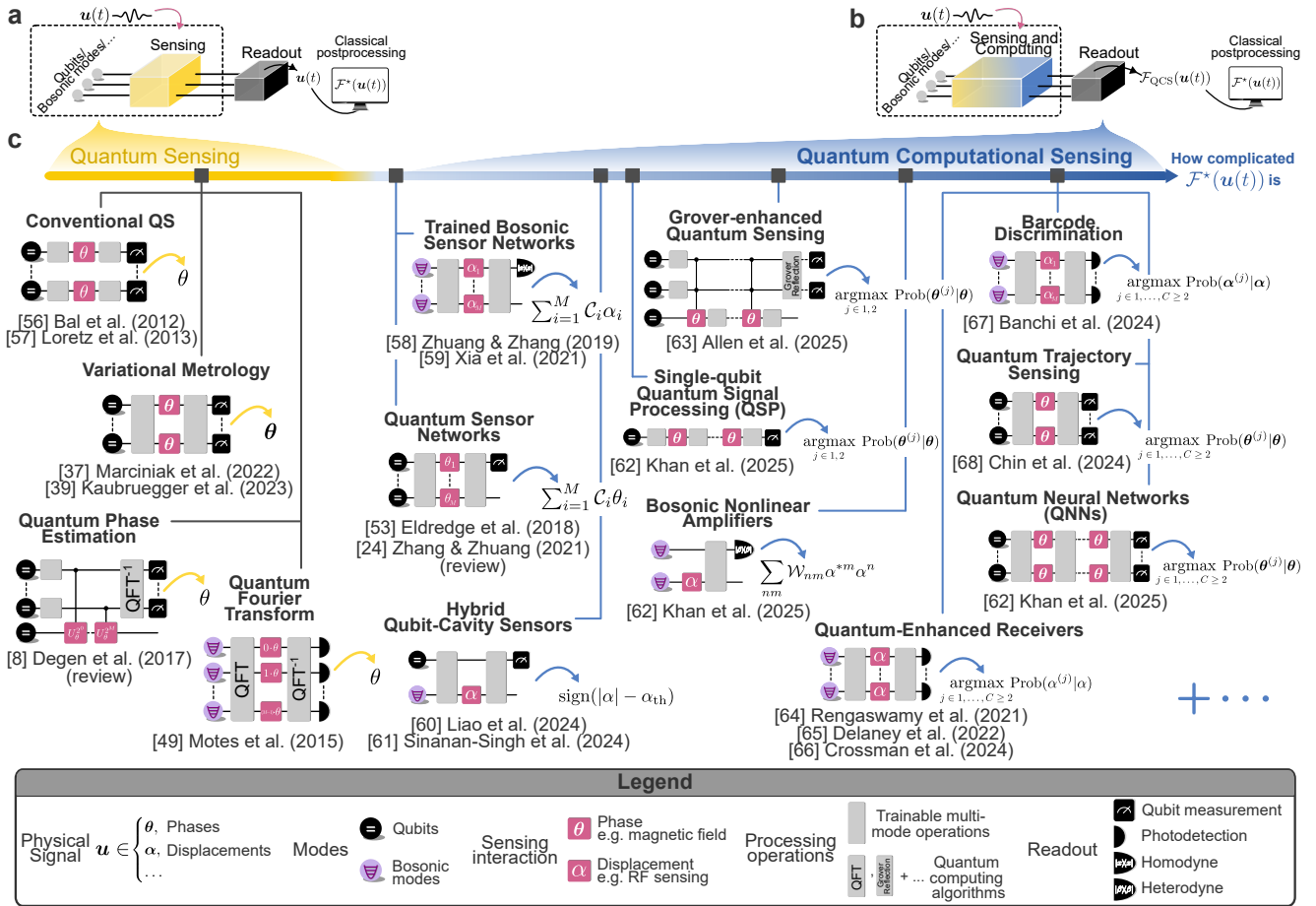


Figure 151: A broad set of quantum sensing and computing protocols. Source: [44].

algorithm [2218]. It should therefore not be treated as equivalent to quantum advantage.

Quantum dominance is not a standardized term. When it is used, it generally denotes a stronger form of quantum-classical separation in which the corresponding classical computation is considered infeasible [2223]. The exact meaning should be defined whenever the term is used.

Quantum energy advantage is another threshold that may arise someday when on top of some computing

time benefits, we could highlight the fact that quantum computers consume much less energy than supercomputers for solving similar problems (Figure 149). This is still a subject of research and is addressed as part of the Quantum Energy Initiative [2084].

Table 13 consolidates selected quantum-supremacy and quantum-advantage announcements since 2019 [2224]. The early claims were dominated by sampling and proof-of-principle tasks rather than end-user applications. From 2022 onward, more experiments and propos-

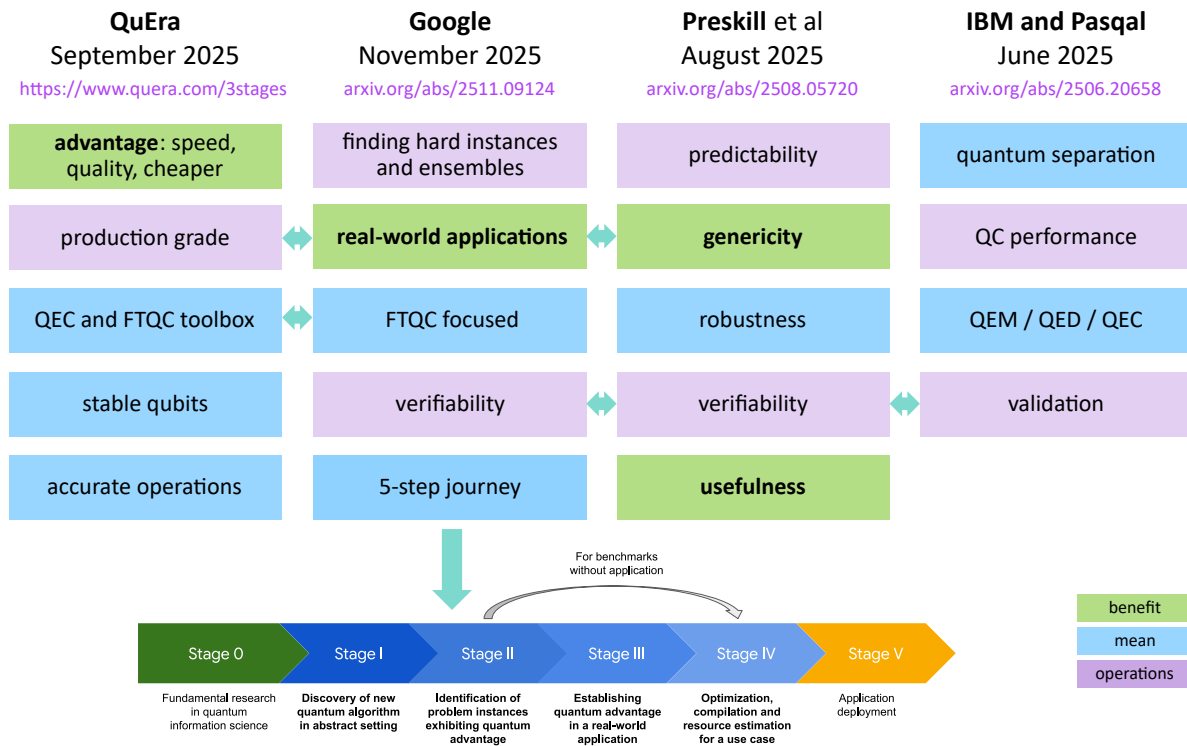


Figure 152: The way different academics and vendors position the criteria to obtain a practical quantum advantage. Sources: Google [2219], John Preskill [2220], IBM and Pasqal [2221] and QuEra [2222]. © Olivier Ezratty, December 2025.

als began to target application-relevant Hamiltonians, learning tasks and resource comparisons, although most remained narrow and strongly dependent on the chosen classical baseline.

The IBM et al. example below was touting a quantum advantage, but it would require a NISQ QPU with 96-qubits and 99.99% 2-qubit gates and measurement fidelities which is extrapolated from a 12-qubit Quantinuum QPU having such fidelities. One key problem is that these trapped ion QPUs are quite hard to scale.

Advantage genericity. Many quantum advantage claims are positioned in the future, in scientific papers title starting with “Toward”. The claims are unfortunately never generic. It must always relate to some specific problem, algorithm, (future) hardware configuration and to many data constraints. For example, the problem data must be sparse or show some symmetry. This lack of advantage genericity is problematic. We can hope that in the future, such advantages will become more generic, when we can say that, in general, quantum computers can solve a broad spectrum of chemical simulation or optimization problems in a reasonable time, that is not at all accessible to supercomputers.

Time scales. Finally, speedup related quantum advantages must be practical and not theoretical. It does not matter much that you obtain some polynomial or exponential speedup to run a chemical simulation of a useful molecule if it lasts 1,500 years! If and when we are able to build scalable FTQC computers with thousands of refined logical qubits, the practical quantum computing time scales for real useful problems will become a key concern and generate new scientific challenges.

Late 2025, various hardware vendors and academic teams published various papers to describe the high-level conditions for obtaining a practical quantum advantage. They adopted distinct approaches and focused on different figures of merit on the expected benefits, on the means to obtain them (stable qubits, error correction, mitigation and error detection in the NISQ regime for IBM), and the operational dimension (industrial size for the data used, verifiability and validation, practical performance). It is interesting here to distinguish between the technical means (in blue), the operational constraints (in purple) and the user benefit (in green) (Figure 152).

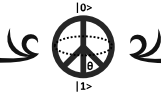
who and when	architecture	algorithm	input data	comment	classically simulated?
Google , Arute et al. October 2019 [2225]	Sycamore, 53 superconducting qubits	cross entropy benchmarking	none	running a random gates algorithm	matched and surpassed classically , 3M uncorrelated samples in 86.4 s on 1,432 A100 GPUs for 13.7 kWh, versus 600 s for 3M Sycamore samples [2182], after [1673, 2181, 2184, 2185, 2190]
China December 2020 [2226]	70 photons modes GBS (Gaussian Boson Sampling)	interferometer photons mixing	none	running a random physical process	contested , a classical squashed-states model explains the data better than the intended squeezed light model [2227]
IBM Research December 2020 [2204]	IBM 27 superconducting qubits	symmetric Boolean functions	SLSB3 function parameters	theoretical demonstration of quantum advantage	stands , the shallow-circuit separation is unconditional and needs no hardness conjecture, but it is proven on a 27-qubit toy instance
Kerenidis, Diamanti et al. March 2021 [2215]	multi-mode photon dense encoding of verified solution	Quantum Merlin Arthur based verification	output from some quantum computation (not implemented)	no actual computing done in the experiment	not applicable, no computation was executed, so there is nothing to dequantize
China April 2021 [892]	Quantum walk on 62 superconducting qubits	simple quantum walk	simulating a 2-photons Mach-Zehnder interferometer	no quantum advantage at all	trivially classical , a 2-photon walk reduces to a small linear-optics matrix computable in microseconds
University of Arizona May 2021 [2224]	supervised learning assisted by an entangled sensor network	variational algorithm, classical computing	data extracted from three entangled squeezed light photonic sensors	not a quantum “computing” advantage per se	not applicable, the learning step is already classical, the claimed gain sits in the sensing front end
China June 2021 [2228]	66 superconducting qubits and 110 couplers, Zuchongzhi 2.0, then 2.1	cross entropy benchmarking	none	56 used qubits 60 used qubits	refuted , exact tensor network contraction of the 60-qubit 24-cycle circuit on the Sunway supercomputer [2188], same contraction pipeline as for Sycamore [2185]
China September 2021 [2229]					
China June 2021 [2230]	144 photons modes GBS and up to 113 detected events	interferometer photons mixing	none	parametrizable photon phases could lead to a programmable system	refuted , quadratic-cost approximation [2207], about 100,000 cores [2208], then an MPS sampler at bond dimension 10^4 beating the experiment on correlation functions [1745]
Google, AWS, Harvard et al. December 2021 [45]	quantum sensors feeding a quantum computer	learning about the principal component of a noisy state	quantum output from quantum sensors	requires some quantum memory	no classical counterpart , the input is quantum data, so the exponential separation holds by construction rather than by hardness conjecture
Xanadu June 2022 [2231]	216 squeezed photons modes GBS (Gaussian Boson Sampling)	time domain multiplexing and interferometer photons mixing	programmable GBS with 1,296 parameters	first programmable GBS	refuted , the same MPS sampler reproduces the $M = 216$ and $M = 288$ Borealis configurations with better correlation functions than the hardware [1745]
IBM et al. September 2022 [1160, 2232, 2233]	hybrid algorithm that could run on NISQ QPUs	QML-TDA unsupervised machine learning technique for extracting valuable shape-related data features	small data sets related to cosmic microwave background	exponential speedup, resilient to noise, requires 96-qubit QPU with 2Q gate and measurement fidelity of 99.99%...	dequantized , the exponential speedup does not survive complexity-theoretic analysis, the hard regime is #P-hard and the tractable regime admits classical algorithms [1160]
IBM June 2023 [2234]	127 qubits	trotterized time evolution of a 2D transverse-field Ising model	none	presented as a utility and not an advantage, quickly classically emulated.	refuted within weeks , belief propagation tensor networks matched to the lattice geometry [2235], sparse Pauli dynamics on a laptop [2236] and PEPO in the Heisenberg picture [2237]
Google June 2023 [2238]	70 qubits on 72 Sycamore chipset	cross entropy benchmarking	none	running a random gates algorithm	not explicitly reproduced, but noisy random circuit sampling admits polynomial [2192] and quasi-polynomial [2194] classical algorithms

continued on next page

who and when	architecture	algorithm	input data	comment	classically simulated?
Google December 2024 [2239]	105 qubits on Willow chipset	cross entropy benchmarking	none	running a random gates algorithm	no published simulation, but the 10²⁵ years figure is a tensor contraction cost estimate and not a measured runtime
D-Wave March 2025 [2240]	D-Wave Advantage-2 annealer, 4,400 qubits	quench dynamics of 1, 2 and 3D infinite-dimensional spin glasses	problem Hamiltonian	largely matched classically in 2026	largely reproduced by lattice-specific belief propagation tensor networks on a workstation [2241] and by time-dependent variational Monte Carlo [2242, 2243], D-Wave disputes the coverage of the hardest geometries and observables [2244]
China August 2025 [2245]	Jiuzhang 4.0	Gaussian boson sampling	none	random interferometer	designed to escape the lossy-MPS regime of [1745] with 92% source efficiency, but Jiuzhang 3.0 and its 1,152 modes have been simulated [2210]
Google September 2025 [2199]	Sycamore 68-qubits	generative quantum model	bitstream generation, learning to generate compressed simulation circuits, learning to generate quantum states	very theoretical constructs, practical use cases not demonstrated	no published simulation , the tasks are too recent and too abstract to have attracted a dedicated classical attack
Google Echoes October 2025 [2200]	Willow 103-qubits used with 65 qubits	Measuring Out-of-Time-Order Correlators (OTOC)	random circuit	some observables, but no practical utility	stands as of mid-2026 , OTOC circuits are shown too entangled for tensor networks with belief propagation [2201], though that rebuttal was authored by Google itself
BlueQubit random peaked circuits October 2025 [2246]	Quantinuum H2 56-qubit QPU	random peaked circuits	none	circuits are easier to classically verify	broken , an MPO contraction exploiting the mirrored circuit structure extracts the peak of the largest instance in about one hour on a single GPU, roughly half the QPU runtime [2247]
PhaseCraft October 2025 [2248, 2249]	Quantinuum H2 56-qubit and on 72 qubits in a Google Willow QPU	time-dynamics of a 2D Fermi-Hubbard fermionic many-body systems	problem Hamiltonian	computing quantum dynamics	mostly matched , the authors themselves benchmarked against tensor network and operator propagation baselines that reach most of the same phenomenology [2249]
Quantinuum November 2025 [2202]	Quantinuum Helios 98 qubits	Fermi-Hubbard models	Fermi-Hubbard problem	measuring pairing correlations	not yet settled , no independent classical reproduction of the off-diagonal pairing correlations has been published

Table 13: Inventory of past quantum advantages/supremacies announcements, their underlying characteristics and the classical simulation work that answered them.

© Olivier Ezratty, 2021-2026. The last column records what was achieved classically, with the hardware and timing details carried in the notes of the corresponding references. Green marks a claim that still stands, red a claim that was matched, refuted or dequantized. The IBM et al. September 2022 claim is also discussed in [2233] and contested in [1160]. The Google, AWS and Harvard December 2021 experiment used 40 Sycamore qubits.



2.5.15 Bibliography and notes

These are the bibliographical references and notes for the **Benchmarking** subsection.

- [44] [Quantum Computational-Sensing Advantage](#), by Saeed A. Khan, Sridhar Prabhu, Logan G. Wright, and Peter L. McMahon, arXiv, July 2025 (27 pages).
- [45] [Quantum advantage in learning from experiments](#), by Hsin-Yuan Huang, Michael Broughton, Jordan Cotler, Sitan Chen, Jerry Li, Masoud Mohseni, Hartmut Neven, Ryan Babbush, Richard Kueng, John Preskill, and Jarrod R. McClean, Science, June 2022 (6 pages).
- [466] [Is quantum advantage the right goal for quantum machine learning?](#), by Maria Schuld and Nathan Killoran, arXiv, February 2023 (13 pages).
- [741] [Quantum advantage in learning from experiments](#), by Hsin-Yuan Huang, Michael Broughton, Jordan Cotler, Sitan Chen, Jerry Li, Masoud Mohseni, Hartmut Neven, Ryan Babbush, Richard Kueng, John Preskill, and Jarrod R. McClean, arXiv, December 2021 (6 pages).
- [868] [Benchmarking a heuristic Floquet adiabatic algorithm for the Max-Cut problem](#), by Etienne Granet and Henrik Dreyer, arXiv, April 2024 (6 pages).
- [892] [Quantum walks on a programmable two-dimensional 62-qubit superconducting processor](#), by Ming Gong, Jian-Wei Pan, et al., arXiv, July 2021 (13 pages).
- [938] [Quantum advantage with shallow circuits](#), by Sergey Bravyi, David Gosset, and Robert Koenig, arXiv, April 2017 (23 pages).
- [1160] [Complexity-Theoretic Limitations on Quantum Algorithms for Topological Data Analysis](#), by Alexander Schmidhuber and Seth Lloyd, arXiv, January 2024 (19 pages).
- [1163] [Why measuring performance is our biggest blind spot in quantum machine learning](#), by and Maria Schuld, March 2022.
- [1324] [Focus beyond Quadratic Speedups for Error-Corrected Quantum Advantage](#), by Ryan Babbush, March 2021 (11 pages).
- [1673] [Closing the “Quantum Supremacy” Gap: Achieving Real-Time Simulation of a Random Quantum Circuit Using a New Sunway Supercomputer](#), by Yong Liu, Dexun Chen, et al., arXiv, November 2021 (18 pages).
- [1674] [China’s exascale quantum simulation not all it appears](#), by Nicole Hemsoth, November 2021.
- [1686] [Validating quantum computers using randomized model circuits](#), by Andrew W. Cross, Lev S. Bishop, Sarah Sheldon, Paul D. Nation, and Jay M. Gambetta, Physical Review A, September 2019 (13 pages)  (preprint on [arXiv](#)).
- [1745] [Classical algorithm for simulating experimental Gaussian boson sampling](#), by Changhun Oh, Minzhao Liu, Yuri Alexeev, Bill Fefferman, and Liang Jiang, Nature Physics, June 2024  (preprint on [arXiv](#)).
- [2017] [Learning a quantum computer’s capability](#), by Daniel Hothem, Kevin Young, Tommie Catanach, and Timothy Proctor, arXiv, October 2024 (20 pages).
- [2033] [The Race to Quantum Advantage Depends on Benchmarking](#), by Matt Langione, Jean-François Bobier, Lisa Kraye, Hanl Park, and Amit Kumar, February 2022.
- [2034] [Quantum Benchmark Zoo](#), by Valentin Gilbert et al., Quantum Benchmark Zoo, June 2025.
- [2035] [Quantum Computer Benchmarking: An Explorative Systematic Literature Review](#), by Tobias Rohe, Federico Harjes Ruiloba, Sabrina Egger, Sebastian von Beck, Jonas Stein, and Claudia Linnhoff-Popien, arXiv, September 2025 (79 pages).
- [2036] [Linear Cross Entropy Benchmarking with Clifford Circuits](#), by Jianxin Chen, Dawei Ding, Cupjin Huang, and Linghang Kong, arXiv, June 2022 (30 pages).
- [2037] [A Review and Collection of Metrics and Benchmarks for Quantum Computers: definitions, methodologies and software](#), by Deep Lall, Ivan Rungger, et al., arXiv, February 2025 (124 pages).
- [2038] [Estimating the Power of a Quantum Computer](#), by Brandon Rodenburg, arXiv, January 2025 (14 pages).
- [2039] [Systematic benchmarking of quantum computers: status and recommendations](#), by Jeanette Miriam Lorenz, Ziyad Amodjee, Mattia Giardini, et al., arXiv, March 2025 (19 pages).
- [2040] [A meta-benchmark for quantum computers](#), by Valentin Gilbert, Teratec TQCI Seminar 24-25th June 2025, June 2025 (86 slides).
- [2041] [Quantum Volume](#), by Lev Bishop et al., 2017 (5 pages).
- [2042] [Quantum Volume for Photonic Quantum Processors](#), by Yuxuan Zhang, Daoheng Niu, Alireza Shabani, and Hassan Shapourian, Physical Review Letters, March 2023 (5 pages)  (preprint on [arXiv](#)).
- [2043] [Quality, Speed, and Scale: three key attributes to measure the performance of near-term quantum computers](#), by Andrew Wack, Hanhee Paik, Ali Javadi-Abhari, Petar Jurcevic, Ismael Faro, Jay M. Gambetta, and Blake R. Johnson, arXiv, October 2021 (8 pages).
- [2044] [Driving quantum performance: more qubits, higher Quantum Volume, and now a proper measure of speed](#), by Jay Gambetta et al., November 2021.
- [2045] [CLOPS: Benchmarking System Speed at Utility Scale](#), by Andrew Wack, arXiv, August 2026 (14 pages).
- [2046] [Updating how we measure quantum quality and speed](#), by Andrew Wack and David McKay, IBM blog, November 2023.
- [2047] [Benchmarking Quantum Processor Performance at Scale](#), by David C. McKay, Ian Hincks, Emily J. Pritchett, Malcolm Carroll, Luke C. G. Govia, and Seth T. Merkel, arXiv, November 2023 (15 pages).
- [2048] [Characterizing Large Scale Quantum Systems with Error Per Circuit Layer](#), by Travis Hurant, Kenneth R. Brown, et al., arXiv, September 2026 (18 pages).
- [2049] [Computational advantage of quantum random sampling](#), by Dominik Hangleiter and Jen Eisert, Review of Modern Physics, July 2023  (preprint on [refubium.fu-berlin.de](#)).
- [2050] [Randomized Benchmarking Beyond Groups](#), by Jianxin Chen, Dawei Ding, and Cupjin Huang, arXiv, December 2022 (36 pages).
- [2051] [Characterizing large-scale quantum computers via cycle benchmarking](#), by Alexander Erhard, Rainer Blatt, et al., Nature Communications, November 2019 (7 pages).
- [2052] [Characterization of entanglement on superconducting quantum computers of up to 414 qubits](#), by John F Kam, Haiyue Kang, Charles D Hill, Gary J Mooney, and Lloyd C L Holtenberg, arXiv, October 2024 (25 pages).
- [2053] [Generating multipartite nonlocality to benchmark quantum computers](#), by Jan Lennart Bönsel, Otfried Gühne, and Adán Cabello, arXiv, January 2025 (14 pages).
- [2054] [Scalable Benchmarks for Gate-Based Quantum Computers](#), by Arjan Cornelissen, Johannes Bausch, and András Gilyén, arXiv, April 2021 (54 pages).
- [2055] [Assessing the quality of near-term photonic quantum devices](#), by Rawad Mezher and Shane Mansfield, arXiv, February 2022 (30).

- [2056] [An entanglement-based volumetric benchmark for near-term quantum hardware](#), by Kathleen E. Hamilton, Raphael Pooser, et al., arXiv, September 2022 (21 pages).
- [2057] [Clifford Volume and Free Fermion Volume: Complementary Scalable Benchmarks for Quantum Computers](#), by Attila Portik, Orsolya Kálmán, Thomas Monz, and Zoltán Zimborás, arXiv, December 2025 (6 pages).
- [2058] [Benchmarking fault-tolerant quantum computing hardware via QLOPS](#), by Linghang Kong, Fang Zhang, and Jianxin Chen, ACM Transactions on Quantum Computing, April 2026 (14 pages).
- [2059] [Benchmarking quantum co-processors in an application-centric, hardware-agnostic and scalable way](#), by Simon Martiel, Thomas Ayrat, and Cyril Allouche, arXiv, July 2021 (12 pages).
- [2060] [Evaluating the Q-score of Quantum Annealers](#), by Ward van der Schoot, Frank Phillipson, et al., arXiv, August 2022 (8 pages).
- [2061] [Extending the Q-score to an Application-level Quantum Metric Framework](#), by Ward van der Schoot, Rob Kooij, et al., arXiv, October 2024 (11 pages).
- [2062] [ORNL researchers advance performance benchmark for quantum computers](#), by ORNL, January 2020.
- [2063] [An application benchmark for fermionic quantum simulations](#), by Pierre-Luc Dallaire-Demers, Michał Stechły, Jerome F. Gonthier, Ntwali Toussaint Bashige, Jonathan Romero, and Yudong Cao, arXiv, March 2020 (14 pages).
- [2064] [HamilToniQ: An Open-Source Benchmark Toolkit for Quantum Computers](#), by Xiaotian Xu, Kuan-Cheng Chen, and Robert Wille, arXiv, April 2024 (11 pages).
- [2065] [Many-body Quantum Score: a scalable benchmark for digital and analog quantum processors and first test on a commercial neutral atom device](#), by Harold Erbin, Pierre-Louis Burdeau, Corentin Bertrand, Thomas Ayrat, and Grégoire Misguich, arXiv, January 2026 (31 pages).
- [2066] [Quantum WalkScore: Benchmarking Quantum Computers on the Graph Nodefinding Problem](#), by Noé Olivier and Michel Nowak, arXiv, September 2026 (18 pages).
- [2067] [Application-Oriented Performance Benchmarks for Quantum Computing](#), by Thomas Lubinski, Sonika Johri, Paul Varosy, Jeremiah Coleman, Luning Zhao, Jason Necaise, Charles H. Baldwin, Karl Mayer, and Timothy Proctor, arXiv, January 2023 (33 pages).
- [2068] [Optimization Applications as Quantum Performance Benchmarks](#), by Thomas Lubinski, Carleton Coffrin, Catherine McGeoch, Pratik Sathe, Joshua Apanavicius, and David E. Bernal Neira, arXiv, February 2024 (31 pages).
- [2069] [Quantum Algorithm Exploration using Application-Oriented Performance Benchmarks](#), by Thomas Lubinski, Pranav S. Mundada, et al., arXiv, February 2024 (21 pages).
- [2070] [Algorithmic Qubits: A Better Single-Number Metric](#), by IonQ, February 2022.
- [2071] [SupermarQ: A Scalable Quantum Benchmark Suite](#), by Teague Tomesh, Frederic T. Chong, et al., Proceedings of the 2022 IEEE International Symposium on High-Performance Computer Architecture, April 2022 (17 pages)  (preprint on arXiv).
- [2072] [QPack: Quantum Approximate Optimization Algorithms as universal benchmark for quantum computers](#), by Koen Mesman, Zaid Al-Ars, and Matthias Möller, arXiv, April 2022 (24 pages).
- [2073] [QuAS: Quantum Application Score for benchmarking the utility of quantum computers](#), by Koen J. Mesman, Ward van der Schoot, Matthias Möller, and Niels M. P. Neumann, arXiv, June 2024 (9 pages).
- [2074] [MQT Bench: Benchmarking Software and Design Automation Tools for Quantum Computing](#), by Nils Quetschlich, Lukas Burgholzer, and Robert Wille, arXiv, July 2023 (14 pages).
- [2075] [QUARK: A Framework for Quantum Computing Application Benchmarking](#), by Jernej Rudi Finžgar, Philipp Ross, Leonhard Hölcher, Johannes Klepsch, and Andre Luckow, arXiv, August 2022 (12 pages).
- [2076] [Application-Oriented Benchmarking of Quantum Generative Learning Using QUARK](#), by Florian J. Kiwit, Marwa Marso, Philipp Ross, Carlos A. Riofrío, Johannes Klepsch, and Andre Luckow, arXiv, August 2023 (10 pages).
- [2077] [QuSquare: Scalable Quality-Oriented Benchmark Suite for Pre-Fault-Tolerant Quantum Devices](#), by David Aguirre, Rubén Peña, and Mikel Sanz, arXiv, December 2025 (46 pages).
- [2078] [Benchmarking Quantum Computers: Towards a Standard Performance Evaluation Approach](#), by Arturo Acuaviva, David Aguirre, Rubén Peña, and Mikel Sanz, Quantum Science and Technology, February 2026 (38 pages).
- [2079] [The EU Quantum Flagship's Key Performance Indicators for Quantum Computing](#), by Zoltán Zimborás, Thomas Monz, Frank Wilhelm-Mauch, et al., arXiv, December 2025 (34 pages).
- [2080] [Metric: A Collaborative Platform for Benchmarking Quantum Computers](#), by Alessandro Cosentino, Changhao Li, Vincent Russo, Bradley A. Chase, Tom Lubinski, Siyuan Niu, Neer Patel, Nathan Shammah, and William J. Zeng, arXiv, March 2026 (41 pages).
- [2081] [Benchmarking the performance of quantum computing software for quantum circuit creation, manipulation and compilation](#), by Paul D. Nation, Ali Javadi-Abhari, et al., Nature Computational Science, April 2025 (10 pages).
- [2082] [Publications highlighting potential impact of quantum computing in specific applications](#), by DARPA, DARPA website, July 2024.
- [2083] [P7131 - Standard for Quantum Computing Performance Metrics & Performance Benchmarking](#), by IEEE, IEEE.
- [2084] [Quantum technologies need a Quantum Energy Initiative](#), by Alexia Auffèves, PRX Quantum, June 2022.
- [2085] [BACQ – Application-oriented Benchmarks for Quantum Computing](#), by Frédéric Barbaresco, Laurent Rioux, Christophe Labreuche, Michel Nowak, Noé Olivier, Damien Nicolazic, Olivier Hess, Anne-Lise Guilmin, Robert Wang, Tanguy Sassolas, Stéphane Louise, Kyrylo Snizhko, Grégoire Misguich, Alexia Auffèves, Robert Whitney, Emmanuelle Vergnaud, and Félicien Schopfer, arXiv, March 2024 (14 pages).
- [2086] [Benchmarking the computational power of quantum computers](#), by Timothy Proctor, Robin Blume-Kohout, et al., arXiv, September 2026 (46 pages).
- [2087] [A volumetric framework for quantum computer benchmarks](#), by Robin Blume-Kohout and Kevin C. Young, arXiv, November 2020 (16 pages).
- [2088] [Quantum Volume in Practice: What Users Can Expect from NISQ Devices](#), by Elijah Pelofske, Andreas Bärtzsch, and Stephan Eidenbenz, arXiv, August 2023 (25 pages).
- [2089] [Benchmarking quantum devices beyond classical capabilities](#), by Rafał Bistron, Marcin Rudziński, Ryszard Kukulski, and Karol Życzkowski, arXiv, February 2025 (29 pages).
- [2090] [Turn down the quantum volume](#), by Scott Aaronson, Scott Aaronson blog, February 2020.
- [2091] [Featuremetric benchmarking: Quantum computer benchmarks based on circuit features](#), by Timothy Proctor, Anh Tran, Xingxin Liu, Aditya Dhumuntarao, Stefan Seritan, Alaina Green, and Norbert M Linke, arXiv, April 2025 (23 pages).
- [2092] [Defining Standard Strategies for Quantum Benchmarks](#), by Mirko Amico, David C. McKay, et al., arXiv, March 2023 (12 pages).
- [2093] [Increasing the Measured Effective Quantum Volume with Zero Noise Extrapolation](#), by Elijah Pelofske, William J. Zeng, et al., arXiv, July 2024 (16 pages).

- [2094] [Measuring the Capabilities of Quantum Computers](#), by Timothy Proctor, Kenneth Rudinger, Kevin Young, Erik Nielsen, and Robin Blume-Kohout, arXiv, January 2022 (4 pages).
- [2095] [Scalable randomized benchmarking of quantum computers using mirror circuits](#), by Timothy Proctor, Robin Blume-Kohout, Kevin Young, et al., arXiv, October 2022 (4 pages).
- [2096] [Hands-on Introduction to Randomized Benchmarking](#), by Ana Silva and Eliska Greplova, arXiv, April 2025 (77 pages).
- [2097] [QPU-scale randomized benchmarking via Bell-pair injection](#), by Haripriya Pettugani, María Aguado-Yáñez, As-tryd Park, Daniel Bultrini, and James R. Wootton, arXiv, June 2026.
- [2098] [Benchmarking near-term quantum computers via random circuit sampling](#), by Yunchao Liu, Matthew Otten, Roozbeh Bassirianjahromi, Liang Jiang, and Bill Fefferman, arXiv, April 2022 (43 pages).
- [2099] [Demonstrating scalable randomized benchmarking of universal gate sets](#), by Jordan Hines, Marie Lu, Ravi K. Naik, Akel Hashim, Jean-Loup Ville, Brad Mitchell, John Mark Kriekebaum, David I. Santiago, Stefan Seritan, Erik Nielsen, Robin Blume-Kohout, Kevin Young, Irfan Siddiqi, Birgitta Whaley, and Timothy Proctor, arXiv, October 2023 (15 pages).
- [2100] [Variational benchmarks for quantum many-body problems](#), by Dian Wu, Giuseppe Carleo, et al., Science, October 2024 (6 pages) 📄 (preprint on [HAL](#)).
- [2101] [PyQBench: a Python library for benchmarking gate-based quantum computers](#), by Konrad Jałowicki, Paulina Lewandowska, and Łukasz Paweła, arXiv, March 2023 (49 pages).
- [2102] [Predictive Models from Quantum Computer Benchmarks](#), by Daniel Hothem, Jordan Hines, Karthik Nataraj, Robin Blume-Kohout, and Timothy Proctor, arXiv, May 2023 (6 pages).
- [2103] [Benchmarking quantum gates and circuits](#), by Vinay Tripathi, Daria Kowsari, Kumar Saurav, Haimeng Zhang, Eli M. Levenson-Falk, and Daniel A. Lidar, arXiv, February 2025 (52 pages).
- [2104] [On verifiable quantum advantage with peaked circuit sampling](#), by Scott Aaronson and Yuxuan Zhang, arXiv, May 2024 (18 pages).
- [2105] [Works on My QPU: Reproducibility in Quantum Computing Research](#), by Dominik Köster, Maja Franz, Benjamin Zec, Nicole Hoess, Ralf Ramsauer, and Wolfgang Maurerer, arXiv, July 2026.
- [2106] [Quantum Fidelity-per-Cost: A Metric for Evaluation of Quantum Computing Systems](#), by Siddarth Shinde and Jakub Szefer, arXiv, July 2026 (7 pages).
- [2107] [CrossBench: Generalized Crosstalk Benchmark Generation for Quantum Computers](#), by Jaden Hawley and Chi-Ren Shyu, arXiv, April 2026 (7 pages).
- [2108] [Quantum anomaly for benchmarking quantum computing](#), by Tomoya Hayata and Arata Yamamoto, arXiv, March 2026 (11 pages).
- [2109] [How “Quantum” is your Quantum Computer? Macrorealism-based Benchmarking via Mid-Circuit Parity Measurements](#), by Ben Zindorf, Lorenzo Braccini, Debarshi Das, and Sougato Bose, arXiv, November 2025 (12 pages).
- [2110] [Nonlocal Games as Cross-Platform Quantum Benchmarks: Exceeding unconditional classical bounds on trapped-ion processors](#), by Anton T. Than, Norbert M. Linke, et al., arXiv, March 2026.
- [2111] [Faking entanglement with imperceptible measurement deviations](#), by Jaime Moreno, Elna Svegborn, Simon Morelli, Markus Hiekkamäki, Lea Kopf, Robert Fickler, and Armin Tavakoli, arXiv, June 2026 (9 pages).
- [2112] [Quantum Benchmarking via Random Dynamical Quantum Maps](#), by Daniel Volya and Prabhat Mishra, arXiv, April 2024 (7 pages).
- [2113] [A generalized cycle benchmarking algorithm for characterizing mid-circuit measurements](#), by Zhihan Zhang, Senrui Chen, Yunchao Liu, and Liang Jiang, arXiv, October 2024 (30 pages).
- [2114] [FTPrimitiveBench: A Benchmark Suite For Logical Computation Under Hardware-Motivated and Biased Noise Models](#), by Shuwen Kan, Adrian Harkness, Zefan Du, Rod Rofougaran, Sean Garner, Chenxu Liu, Ying Mao, and Samuel Stein, arXiv, May 2026 (34 pages).
- [2115] [FTCircuitBench: A Benchmark Suite for Fault-Tolerant Quantum Compilation and Architecture](#), by Adrian Harkness, Samuel Stein, et al., arXiv, January 2026 (54 pages).
- [2116] [Scalable Full-Stack Benchmarks for Quantum Computers](#), by Jordan Hines and Timothy Proctor, arXiv, December 2023 (9 pages).
- [2117] [Magic Mirror on the Wall, How to Benchmark Quantum Error Correction Codes, Overall ?](#), by Avimita Chatterjee and Swaroop Ghosh, arXiv, September 2024 (12 pages).
- [2118] [Efficient benchmarking of logical magic state](#), by Su-un Lee, Ming Yuan, Senrui Chen, Kento Tsubouchi, and Liang Jiang, arXiv, May 2025 (26 pages).
- [2119] [A methodology for comparing and benchmarking quantum devices](#), by Jessica Park, Susan Stepney, and Irene D’Amico, arXiv, May 2024 (15 pages).
- [2120] [The impact of hardware specifications on reaching quantum advantage in the fault tolerant regime](#), by Mark Webber et al., January 2022 (15 pages).
- [2121] [Benchmarking the algorithmic performance of near-term neutral atom processors](#), by K. McInroy, N. Pearson, and J. D. Pritchard, arXiv, February 2024 (14 pages).
- [2122] [Leakage Benchmarking for Universal Gate Sets](#), by Bujiao Wu, Jianxin Chen, et al., arXiv, April 2024 (27 pages).
- [2123] [Classical simulation of non-Gaussian bosonic circuits](#), by Beatriz Dias and Robert Koenig, arXiv, March 2024 (43 pages).
- [2124] [Benchmarking Quantum Computers via Protocols, Comparing Superconducting and Ion-Trap Quantum Technology](#), by Nitay Mayo, Tal Mor, and Yossi Weinstein, arXiv, April 2026 (37 pages).
- [2125] [Benchmarking Quantum Computers via Protocols, Comparing IBM’s Heron vs IBM’s Eagle](#), by Nitay Mayo, Tal Mor, and Yossi Weinstein, arXiv, March 2026 (42 pages).
- [2126] [HyQBench: A Benchmark Suite for Hybrid CV-DV Quantum Computing](#), by Shubdeep Mohapatra, Yuan Liu, Eddy Z. Zhang, and Huiyang Zhou, arXiv, March 2026 (13 pages).
- [2127] [Milestones on the Quantum Utility Highway](#), by Catherine C. McGeoch and Pau Farre, arXiv, May 2023 (34 pages).
- [2128] [New developments around Atos Q-score](#), by Philippe Duluc, August 2022.
- [2129] [Efficient protocol for solving combinatorial graph problems on neutral-atom quantum processors](#), by Wesley da Silva Coelho, Mauro D’Arcangelo, and Louis-Paul Henry, arXiv, August 2022 (16 pages).
- [2130] [Finland launches a 20-qubit quantum computer development towards more powerful quantum computers continues](#), by IQM, IQM news, October 2023.
- [2131] [Benchmarking quantum computers](#), by Timothy Proctor, Kevin Young, Andrew D. Baczewski, and Robin Blume-Kohout, arXiv, July 2024 (16 pages).
- [2132] [QPack Scores: Quantitative performance metrics for application-oriented quantum computer benchmarking](#), by Huub Donkers, Koen Mesman, Zaid Al-Ars, and Matthias Möller, arXiv, May 2022 (23 pages).
- [2133] [QPack: A cross-platform quantum benchmark-suite, Quantitative performance metrics for application oriented quantum computer benchmarking](#), by Huub Donkers, 2022 (67 pages).

- [2134] [HamLib: A library of Hamiltonians for benchmarking quantum algorithms and hardware](#), by Nicolas PD Sawaya, Daan Camps, et al., *Quantum*, December 2024.
- [2135] [Standardized test of many-body coherence in gate-based quantum platforms](#), by Yi Teng, Orazio Scarlatella, Shiyu Zhou, Armin Rahmani, Claudio Chamon, and Claudio Castelnovo, arXiv, March 2025 (5 pages).
- [2136] [QuantumBench: A Benchmark for Quantum Problem Solving](#), by Shunya Minami, Tatsuya Ishigaki, Ikko Hamamura, Taku Mikuriya, Youmi Ma, Naoaki Okazaki, Hiroya Takamura, Yohichi Suzuki, and Tadashi Kadowaki, arXiv, October 2025 (11 pages).
- [2137] [Measuring what matters: A scalable framework for application-level quantum benchmarking](#), by Willie Aboumrad and Martin Roetteler, arXiv, April 2026 (79 pages).
- [2138] [Benchmark of quantum algorithms for ground state preparation in the presence of noise](#), by Daniel Molpeceres, Sirui Lu, J. Ignacio Cirac, and Barbara Kraus, arXiv, June 2026 (12 pages).
- [2139] [InferQ: A Database-Oriented Benchmark for Quantum Circuits Simulation](#), by Andrei Iliescu, Aadi Patwardhan, and Rihan Hai, arXiv, July 2026.
- [2140] [Benchmarking quantum simulation at scale](#), by Jeremy Hartse, Mohsin Raza, Shravan Shravan, Ivan H. Deutsch, and Niklas Mueller, arXiv, July 2026 (18 pages).
- [2141] [On the importance of scalability and resource estimation of quantum algorithms for domain sciences](#), by Vincent R. Pascuzzi, Ning Bao, and Ang Li, arXiv, May 2022 (5 pages).
- [2142] [Wasserstein Solution Quality and the Quantum Approximate Optimization Algorithm: A Portfolio Optimization Case Study](#), by Jack S. Baker and Santosh Kumar Radha, arXiv, February 2022 (21 pages).
- [2143] [BenchQC – Scalable and modular benchmarking of industrial quantum computing applications](#), by Florian Geissler, Jeanette Miriam Lorenz, et al., arXiv, April 2025 (14 pages).
- [2144] [AppQSim: Application-oriented benchmarks for Hamiltonian simulation on a quantum computer](#), by Etienne Granet and Henrik Dreyer, arXiv, March 2025 (24 pages).
- [2145] [Evaluating the performance of quantum processing units at large width and depth](#), by J. A. Montanez-Barrera, Kristel Michielsen, and David E. Bernal Neira, arXiv, May 2025 (19 pages).
- [2146] [Classical Benchmarks for Variational Quantum Eigensolver Simulations of the Hubbard Model](#), by Antonios M. Alvertis, Abid Khan, Thomas Iadecola, Peter P. Orth, and Norm Tubman, arXiv, May 2025 (19 pages).
- [2147] [Classical Benchmarks of a Symmetry-Adapted Variational Quantum Eigensolver for Real-Time Green's Functions in Dynamical Mean-Field Theory](#), by Aadi Singh, Chakradhar Rangi, and Ka-Ming Tam, arXiv, February 2026 (11 pages).
- [2148] [Probing quantum advantage for solving the Fermi-Hubbard model with entropy benchmarking](#), by Pauline Besserve and Raúl García-Patrón, arXiv, October 2025 (14 pages).
- [2149] [A Practical Cross-Platform, Multi-Algorithm Study of Quantum Optimisation for Configurational Analysis of Materials](#), by Kieran McDowall, Phalgun Lolur, Konstantinos Georgopoulos, et al., arXiv, May 2025 (33 pages).
- [2150] [A Comparative Study of Quantum Optimization Techniques for Solving Combinatorial Optimization Benchmark Problems](#), by Monit Sharma and Hoong Chuin Lau, arXiv, March 2025 (30 pages).
- [2151] [Quantum Optimization Benchmark Library - The Intractable Decathlon](#), by Thorsten Koch, Christa Zoufal, et al., arXiv, April 2025 (66 pages).
- [2152] [Towards Robust Benchmarking of Quantum Optimization Algorithms](#), by David Bucher, Nico Kraus, Jonas Blenninger, Michael Lachner, Jonas Stein, and Claudia Linnhoff-Popien, arXiv, May 2024 (12 pages).
- [2153] [QOPTLib: a Quantum Computing Oriented Benchmark for Combinatorial Optimization Problems](#), by Eneko Osaba and Esther Villar-Rodriguez, arXiv, April 2024 (15 pages).
- [2154] [Fair Benchmarking of Optimisation Applications](#), by Frank Phillipson, arXiv, December 2025 (6 pages).
- [2155] [GRADE: Grover-based Benchmarking Toolkit for Assessing Quantum Hardware](#), by Shay Manor, Millan Kumar, Priyank Behera, Azain Khalid, and Oliver Zeng, arXiv, April 2025 (13 pages).
- [2156] [Quantum search on noisy intermediate-scale quantum devices](#), by Kun Zhang, Kwangmin Yu, and Vladimir Korepin, arXiv, September 2022 (12 pages).
- [2157] [GroverFigureOfMerit: An Agnostic Figure of Merit for Quantum Backend Characterization in the NISQ Era](#), by Tiago Restucha, Marcos Guillermo Lammers, and Alejandro Fernández, arXiv, July 2026.
- [2158] [Quantum vs. classical: A comprehensive benchmark study for predicting time series with variational quantum machine learning](#), by Tobias Fellner, David Kreplin, Samuel Tovey, and Christian Holm, arXiv, April 2025 (20 pages).
- [2159] [QMetric: Benchmarking Quantum Neural Networks Across Circuits, Features, and Training Dimensions](#), by Silvie Illéssová, Tomasz Rybotycki, and Martin Beseda, arXiv, June 2025 (15 pages).
- [2160] [Better than classical? The subtle art of benchmarking quantum machine learning models](#), by Joseph Bowles, Shah Nawaz Ahmed, and Maria Schuld, arXiv, March 2024 (41 pages).
- [2161] [Benchmarking quantum machine learning kernel training for classification tasks](#), by Diego Alvarez-Estevéz, arXiv, January 2025 (19 pages).
- [2162] [Benchmarking a Tunable Quantum Neural Network on Trapped-Ion and Superconducting Hardware](#), by Djamil Lakhdar-Hamina, Xingxin Liu, Richard Barney, Sarah H. Miller, Alaina M. Green, Norbert M. Linke, and Victor Galitski, arXiv, July 2025 (6 pages).
- [2163] [Benchmarking Encoding Families in Quantum Neural Networks Under Fixed Circuit Area for Frequency Spectrum and Trainability](#), by Martyna Czuba, Patrick Holzer, and Hein Zay Yar Oo, arXiv, March 2026 (28 pages).
- [2164] [Benchmarking Quantum Reinforcement Learning](#), by Georg Kruse, Rodrigo Coelho, Andreas Rosskopf, Robert Wille, and Jeanette Miriam Lorenz, arXiv, March 2025 (10 pages).
- [2165] [Benchmarking MedMNIST dataset on real quantum hardware](#), by Gurinder Singh, Hongni Jin, and Kenneth M. Merz Jr, arXiv, March 2025 (15 pages).
- [2166] [Application-Driven Benchmarking of the Traveling Salesperson Problem: a Quantum Hardware Deep-Dive](#), by Amine Bentellis, Benedikt Poggel, and Jeanette Miriam Lorenz, arXiv, July 2025 (11 pages).
- [2167] [Benchmarking Digital-Analog Quantum Computation](#), by Vicente Pina Canelles, Adrian Auer, et al., arXiv, July 2023 (5 pages).
- [2168] [Benchmarking bosonic and fermionic dynamics](#), by Jadwiga Wilkens, Marios Ioannou, Ellen Derbyshire, Jens Eisert, Dominik Hangleiter, Ingo Roth, and Jonas Haferkamp, arXiv, August 2024 (5 pages).
- [2169] [BenchRL-QAS: Benchmarking reinforcement learning algorithms for quantum architecture search](#), by Azhar Ikhtiarudin, Aditi Das, Param Thakkar, and Akash Kundu, arXiv, July 2025 (13 pages).
- [2170] [The Benchmark Lottery](#), by Mostafa Dehghani, Yi Tay, Alexey A. Gritsenko, Zhe Zhao, Neil Houlsby, Fernando Diaz, Donald Metzler, and Oriol Vinyals, arXiv, July 2021 (34 pages).
- [2171] [Towards European Standards for Quantum Technologies](#), by O. van Deventer, N. Spethmann, M. Loeffler, M. Amoretti, R. van den Brink, N. Bruno, P. Comi, N. Farrugia, M. Gramegna, B. Kassenberg, W. Kozłowski, T. Länger, T. Lindstrom, V. Martin, N. Neumann, H. Papadopoulos,

- S. Pascasio, M. Peev, R. Pitwon, M.A. Rol, P. Traina, P. Venderbosch, F. K. Wilhelm-Mauch, and A. Jenet, arXiv, October 2022 (55 pages).
- [2172] [Arline Benchmarks: Automated Benchmarking Platform for Quantum Compilers](#), by Y. Kharkov, A. Ivanova, E. Mikhantiev, and A. Kotelnikov, arXiv, February 2022 (27 pages).
- [2173] [Software for Creating Scalable Benchmarks from Quantum Algorithms](#), by Noah Siekierski, Stefan Seritan, Neer Patel, and Timothy Proctor, arXiv, November 2025 (17 pages).
- [2174] [Benchmarking Quantum Software Testing with Scalable Quantum Programs](#), by Yuechen Li, Minqi Shao, Xiyuan Li, Jianjun Zhao, and Kai-Yuan Cai, arXiv, July 2026.
- [2175] [Quantum Computing and the Entanglement Frontier](#), by John Preskill, arXiv, November 2012 (18 pages).
- [2176] [Quantum supremacy: Some fundamental concepts](#), by Man-Hong Yung, January 2019 (2 pages).
- [2177] [Quantum Algorithms Struggle Against Old Foe: Clever Computers](#), by Ariel Bleicher, February 2018.
- [2178] [A brief history of quantum vs classical computational advantage](#), by Ryan LaRose, arXiv, December 2024 (18 pages).
- [2179] [First proof of quantum computer advantage](#), by Technical University Munich, October 2018.
- [2180] [Simpler Proofs of Quantumness](#), by Zvika Brakerski, Venkata Koppula, Umesh Vazirani, and Thomas Vidick, arXiv, May 2020 (12 pages).
- [2181] [Leveraging Secondary Storage to Simulate Deep 54-qubit Sycamore Circuits](#), by Edwin Pednault, John A. Gunnels, Giacomo Nannicini, Lior Horeish, and Robert Wisnieff, arXiv, October 2019 (39 pages).
- [2182] [Leapfrogging Sycamore: harnessing 1432 GPUs for 7x faster quantum random circuit sampling](#), by Xian-He Zhao, Han-Sen Zhong, Feng Pan, Zi-Han Chen, Rong Fu, Zhongling Su, Xiaotong Xie, Chaoxing Zhao, Pan Zhang, Wanli Ouyang, Chao-Yang Lu, Jian-Wei Pan, and Ming-Cheng Chen, National Science Review, March 2025.
- [2183] [The Road to Quantum Computational Supremacy](#), by Cristian S. Calude and Elena Calude, arXiv, December 2019 (15 pages).
- [2184] [What limits the simulation of quantum computers?](#), by Yiqing Zhou et al., PRX, November 2020 (14 pages).
- [2185] [Solving the sampling problem of the Sycamore quantum circuits](#), by Feng Pan, Keyang Chen, and Pan Zhang, arXiv, August 2022 (17 pages).
- [2186] [Simulating the Sycamore quantum supremacy circuits](#), by Feng Pan and Pan Zhang, arXiv, March 2021 (9 pages).
- [2187] [Efficient parallelization of tensor network contraction for simulating quantum computation](#), by Cupjin Huang, Jianxin Chen, et al., Nature Computational Science, September 2021.
- [2188] [Validating quantum-supremacy experiments with exact and fast tensor network contraction](#), by Yong Liu, Yaojian Chen, Chu Guo, Jiawei Song, Xinmin Shi, Lin Gan, Wenzhao Wu, Wei Wu, Haohuan Fu, Xin Liu, Dexun Chen, Zhifeng Zhao, Guangwen Yang, and Jiangang Gao, arXiv, January 2024 (5 pages).
- [2189] [How to Design a Classically Difficult Random Quantum Circuit for Quantum Computational Advantage Experiments](#), by He-Liang Huang, Youwei Zhao, and Chu Guo, arXiv, January 2025 (11 pages).
- [2190] [Density-Matrix Renormalization Group Algorithm for Simulating Quantum Circuits with a Finite Fidelity](#), by Thomas Ayrat et al., PRX Quantum, August 2022 (26 pages).
- [2191] [Quantum supremacy: Is it true that current quantum computers can beat classical computers?](#), by Thomas Ayrat, May 2023.
- [2192] [A polynomial-time classical algorithm for noisy random circuit sampling](#), by Dorit Aharonov, Xun Gao, Zeph Landau, Yunchao Liu, and Umesh Vazirani, arXiv, November 2022 (27 pages).
- [2193] [New Algorithm Closes Quantum Supremacy Window](#), by Ben Brubaker, January 2023.
- [2194] [Classically Sampling Noisy Quantum Circuits in Quasi-Polynomial Time under Approximate Markovianity](#), by Yifan F. Zhang, Su-un Lee, Liang Jiang, and Sarang Gopalakrishnan, arXiv, October 2025 (32 pages).
- [2195] [Revisiting Quantum Supremacy: Simulating Sycamore-Class Circuits Using Hybrid CPU/GPU HPC Workloads](#), by Bob Wold and Venkateswaran Kasirajan, arXiv, December 2025 (6 pages).
- [2196] [Google's Quantum Supremacy Claim: Data, Documentation, and Discussion](#), by Gil Kalai, Yosef Rinott, and Tomer Shoham, arXiv, January 2023 (32 pages).
- [2197] [Questions and Concerns About Google's Quantum Supremacy Claim](#), by Gil Kalai, Yosef Rinott, and Tomer Shoham, arXiv, May 2023 (49 pages).
- [2198] [Phase transitions in random circuit sampling](#), by A. Morvan, S. Boixo, et al., Nature, October 2024.
- [2199] [Generative quantum advantage for classical and quantum problems](#), by Hsin-Yuan Huang, Michael Broughton, Norhan Eassa, Hartmut Neven, Ryan Babbush, and Jarrod R. McClean, arXiv, September 2025 (49 pages).
- [2200] [Observation of constructive interference at the edge of quantum ergodicity](#), by Google Quantum AI, Collaborators, and Dmitry A. Abanin, Nature, October 2025 (76 pages).
- [2201] [Tensor Networks with Belief Propagation Cannot Feasibly Simulate Google's Quantum Echoes Experiment](#), by Pablo Bermejo, Benjamin Villalonga, Brayden Ware, Guifre Vidal, and Aaron Szasz, arXiv, April 2026 (37 pages).
- [2202] [Superconducting pairing correlations on a trapped-ion quantum computer](#), by Etienne Granet, Henrik Dreyer, et al., arXiv, November 2025 (63 pages).
- [2203] [Observation of Improved Accuracy over Classical Sparse Ground-State Solvers using a Quantum Computer](#), by William Kirby, Antonio Mezzacapo, et al., arXiv, March 2026 (40 pages).
- [2204] [Quantum advantage for computations with limited space](#), by Dmitri Maslov, Jin-Sung Kim, Sergey Bravyi, Theodore J. Yoder, and Sarah Sheldon, Nature Physics, June 2021 (preprint on arXiv).
- [2205] [Peaked quantum advantage using error correction](#), by Abhinav Deshpande, Bill Fefferman, Soumik Ghosh, Michael Gullans, and Dominik Hangleiter, arXiv, October 2025 (30 pages).
- [2206] [Quantum Computing 2022](#), by James D. Whitfield, Jun Yang, Weishi Wang, Joshua T. Heath, and Brent Harrison, arXiv, June 2022 (14 pages).
- [2207] [Efficient approximation of experimental Gaussian boson sampling](#), by Benjamin Villalonga, Murphy Yuezhen Niu, Li Li, Hartmut Neven, John C. Platt, Vadim N. Smelyanskiy, and Sergio Boixo, arXiv, February 2022 (17 pages).
- [2208] [The boundary for quantum advantage in Gaussian boson sampling](#), by Jacob F.F. Bulmer, 2022 (8 pages).
- [2209] [Gaussian Boson Sampling with Pseudo-Photon-Number Resolving Detectors and Quantum Computational Advantage](#), by Yu-Hao Deng, Jian-Wei Pan, et al., arXiv, September 2023 (7 pages).
- [2210] [Gaussian boson sampling: Benchmarking quantum advantage](#), by Ned Goodman, Alexander S. Delliios, Margaret D. Reid, and Peter D. Drummond, arXiv, April 2026 (15 pages).
- [2211] [Quantum Advantage of Threshold Changeable Secret Sharing Scheme](#), by Xiaogang Cheng, Ren Guo, and Changli Zhou, arXiv, September 2022 (11 pages).
- [2212] [Quantum advantages for transportation tasks: projectiles, rockets and quantum backflow](#), by David Trillo, Thinh P. Le, and Miguel Navascues, arXiv, September 2023 (19 pages).

- [2213] [Absence of quantum advantage for approximate spin glass optimization](#), by Dries Sels and Flaviano Morone, arXiv, July 2026 (4 pages).
- [2214] [Hybrid Quantum Annealing Approach for High-Dimensional and Multi-Criteria Constrained Quadratic Optimization in Arctic Ship Routing](#), by Tara Kit, Kimsay Pov, Myeongseong Go, Leanghok Hour, Arim Ryou, Kiwoong Kim, Tae-Kyung Kim, and Youngsun Han, arXiv, December 2025 (19 pages).
- [2215] [Experimental demonstration of quantum advantage for NP verification with limited information](#), by Federico Centrone, Niraj Kumar, Eleni Diamanti, and Iordanis Kerenidis, Nature, February 2021 (11 pages).
- [2216] [Quantum superiority for verifying NP-complete problems with linear optics](#), by Juan Miguel Arrazola, Eleni Diamanti, and Iordanis Kerenidis, npj Quantum Information, November 2018.
- [2217] [Exponential speedups in fault-tolerant processing of quantum experiments](#), by Ishaan Kannan, Harald Putterman, and Jordan Cotler, arXiv, May 2026 (48 pages).
- [2218] [What is quantum utility?](#), by Robert Davis, November 2023.
- [2219] [The Grand Challenge of Quantum Applications](#), by Ryan Babbush, Jarrod R. McClean, Thomas O’Brien, Nicholas C. Rubin, et al., arXiv, November 2025 (22 pages).
- [2220] [The vast world of quantum advantage](#), by Hsin-Yuan Huang, Soonwon Choi, Jarrod R. McClean, and John Preskill, Physical Review X, July 2026 (29 pages).
- [2221] [A Framework for Quantum Advantage](#), by Olivia Lanes, Mourad Beji, Antonio D. Corcoles, Constantin Dalyac, Jay M. Gambetta, Loic Henriet, Ali Javadi-Abhari, Abhinav Kandala, Antonio Mezzacapo, Christopher Porter, Sarah Sheldon, John Watrous, Christa Zoufal, Alexandre Dauphin, and Borja Peropadre, arXiv, June 2025 (9 pages).
- [2222] [Three Steps \(and nine milestones\) to Commercial Reality in Quantum Computing](#), by Yuval Boger, QuEra website, September 2025 (4 pages).
- [2223] [Quantum utility – definition and assessment of a practical quantum advantage](#), by Nils Herrmann, Daanish Arya, Marcus W. Doherty, Angus Mingare, Jason C. Pillay, Florian Preis, and Stefan Prestel, arXiv, June 2023 (15 pages).
- [2224] [Quantum-Enhanced Data Classification with a Variational Entangled Sensor Network](#), by Yi Xia, June 2021 (17 pages).
- [2225] [Quantum supremacy using a programmable superconducting processor](#), by Frank Arute, October 2019 (12 pages).
- [2226] [Quantum computational advantage using photons](#), by Han-Sen Zhong, Jian-Wei Pan, et al., Science, December 2020 (23 pages)  (preprint on [arXiv](#)).
- [2227] [Classical models may be a better explanation of the Jiuzhang 1.0 Gaussian Boson Sampler than its targeted squeezed light model](#), by Javier Martinez-Cifuentes, K. M. Fonseca-Romero, and Nicolas Quesada, Quantum, August 2023 (20 pages).
- [2228] [Strong quantum computational advantage using a superconducting quantum processor](#), by Yulin Wu, Jian-Wei Pan, et al., arXiv, June 2021 (22 pages).
- [2229] [Quantum Computational Advantage via 60-Qubit 24-Cycle Random Circuit Sampling](#), by Qingling Zhu, Jian-Wei Pan, et al., arXiv, September 2021 (15 pages).
- [2230] [Phase-Programmable Gaussian Boson Sampling Using Stimulated Squeezed Light](#), by Han-Sen Zhong, Jian-Wei Pan, et al., Physical Review Letters, October 2021 (23 pages).
- [2231] [Quantum computational advantage with a programmable photonic processor](#), by Lars S. Madsen, Jonathan Lavoie, et al., Nature, June 2022.
- [2232] [Towards Quantum Advantage on Noisy Quantum Computers](#), by Ismail Yunus Akhalwaya, Abhinav Deshpande, Tomas Babej, Seth Lloyd, Daniel C. McKay, and Prineha Narang, arXiv, 2022 (34 pages).
- [2233] [Analyzing Prospects for Quantum Advantage in Topological Data Analysis](#), by Dominic W. Berry, Nathan Wiebe, Vedran Dunjko, Ryan Babbush, et al., arXiv, September 2023 (54 pages).
- [2234] [Evidence for the utility of quantum computing before fault tolerance](#), by Youngseok Kim, Abhinav Kandala, et al., Nature, June 2023.
- [2235] [Efficient Tensor Network Simulation of IBM’s Eagle Kicked Ising Experiment](#), by Joseph Tindall, Matthew Fishman, Edwin Miles Stoudenmire, and Dries Sels, PRX Quantum, June 2023.
- [2236] [Fast and converged classical simulations of evidence for the utility of quantum computing before fault tolerance](#), by Tomislav Begusic, Johnnie Gray, and Garnet Kin-Lic Chan, Science Advances, January 2024.
- [2237] [Simulation of IBM’s kicked Ising experiment with Projected Entangled Pair Operator](#), by Hai-Jun Liao, Kang Wang, Zong-Sheng Zhou, Pan Zhang, and Tao Xiang, arXiv, August 2023 (8 pages).
- [2238] [Phase transition in Random Circuit Sampling](#), by A. Morvan et al., arXiv, April 2023 (43 pages).
- [2239] [Meet Willow, our state-of-the-art quantum chip](#), by Hartmut Neven, Google news, December 2024 Google’s Random Circuit Sampling benchmark via Cross Entropy Benchmarking was not detailed in a scientific publication.
- [2240] [Beyond-classical computation in quantum simulation](#), by Andrew D. King et al., Science, March 2025 (15 pages)  (preprint on [ruj.uj.edu.pl](#)).
- [2241] [Dynamics of disordered quantum systems with two- and three-dimensional tensor networks](#), by Joseph Tindall, Antonio Mello, Matt Fishman, Miles Stoudenmire, and Dries Sels, Science, May 2026 (36 pages)  (preprint on [arXiv](#)).
- [2242] [Challenging the Quantum Advantage Frontier with Large-Scale Classical Simulations of Annealing Dynamics](#), by Linda Mauron and Giuseppe Carleo, arXiv, March 2025 (10 pages).
- [2243] [Numerical simulation of D-Wave’s quantum advantage experiment with time-dependent variational Monte Carlo](#), by Roeland Wiersema, arXiv, September 2026.
- [2244] [Comment on: “Dynamics of disordered quantum systems with two- and three-dimensional tensor networks”](#), by Andrew D. King, Mohammad H. Amin, et al., arXiv, March 2025 (4 pages).
- [2245] [Robust quantum computational advantage with programmable 3050-photon Gaussian boson sampling](#), by Hua-Liang Liu, Jian-Wei Pan, et al., arXiv, August 2025 (8 pages).
- [2246] [Heuristic Quantum Advantage with Peaked Circuits](#), by Hrant Gharibyan, Mohammed Zuhair Mullath, Nicholas E. Sherman, Vincent P. Su, Hayk Tepanyan, and Yuxuan Zhang, arXiv, October 2025 (24 pages).
- [2247] [Efficient Classical Simulation of Heuristic Peaked Quantum Circuits](#), by David Kremer and Nicolas Dupuis, arXiv, April 2026 (10 pages).
- [2248] [Fermionic dynamics on a trapped-ion quantum computer beyond exact classical simulation](#), by Faisal Alam, Gethin Wright, et al., arXiv, October 2025 (54 pages).
- [2249] [Programmable digital quantum simulation of 2D Fermi-Hubbard dynamics using 72 superconducting qubits](#), by Faisal Alam, Gethin Wright, et al., arXiv, October 2025 (98 pages).

This **Benchmarking** subsection contains 232 bibliographical references and notes containing at least 4,332 pages.

2.6 Cloud quantum computing

A large share of quantum computers is intended to be offered through cloud services. It even got a specific nickname: **QCaaS** (Quantum Computing as a Service). There are various estimates for the cloud quantum computing market including a very optimistic one of \$26B by 2030 by The Quantum Daily [2250, 2251]. Being now about 4 years from now as of 2026, it seems quite optimistic. See [2252] for a broader academic overview of the cloud quantum computing landscape, its open problems and future directions.

2.6.1 Cloud rationale

There are a few technical and economic reasons behind the ineluctable cloudification of quantum computing access. Most quantum computers are relatively expensive devices, costing at least a couple of \$Ms. D-Wave systems are sold at a price of about \$14M and IBM QPUs at over \$40M while other NISQ vendors are around \$8M to \$13M. These are rapidly changing devices with one generation cancelling the previous one ever year.

Also, all quantum algorithms are hybrid in nature with some classical workload involved in pre-processing and post-processing, including code compiling and optimizing tasks. They require a nearby classical computer, if not an HPC or supercomputer, depending on the workload. These classical requirements may grow with FTQC platforms. This mandates some mutualization. Customers in the early stages of quantum computing adoption are better off spending money in opex (cloud access) than capex (QPU acquisitions).

When would customers prefer to acquire QPUs and run them on-premises in their own datacenters? It would make sense when economies of scale are significant within the customer or if he handles very sensitive data and processes. The first on-premises installations will probably be deployed for military and intelligence use cases with governments and/or defense contractors with significant security or sovereignty constraints.

But even when quantum computing matures with upcoming FTQC QPUs, the cloud use rationale will still be valid for most users.

On the technical side, making a quantum computer accessible in the cloud requires putting in place a software infrastructure reminiscent of the old mainframe days. Indeed, QPUs are not “multitasking” machines. They are fed by classical computers through a queueing system in “batch mode”. The batches first compile the code, execute it several times, usually a couple thousand times (at least, in the NISQ regime), then the result is sent in asynchronous mode to the user [2253]. SLURM is one of the most popular open-source frameworks to implement such job scheduling systems.

Work is being done to craft heterogeneous classical/quantum cloud platforms [2254–2256] and optimize resource

allocation models in distributed architectures [2257–2265]. This may use the technique of “job splitting” which will distribute several circuit shots of a NISQ circuit on multiple QPUs having different qubit fidelities [2266].

Recent research on scheduling starts to quantify these strategies rather than just describing them conceptually. Comparing time-based multiplexing, dynamic resource management and workflow decomposition on production HPC clusters and real quantum hardware, one study reports classical resource consumption cuts of up to 45.7% and 64% respectively depending on the quantum-to-classical workload ratio [2267]. Other schedulers jointly optimize queue time and circuit fidelity across heterogeneous cloud providers by modeling hybrid workloads as dynamic DAGs (Directed Acyclic Graphs) with explicit quantum semantics such as entanglement dependencies and no-cloning constraints [2268], or apply evolutionary multi-objective optimization to minimize completion time, execution error rate and load imbalance across noisy quantum processors with varying calibration data [2265].

Another proposal for creating Quantum Data Centers deals with sharing QRAM, T gates and oracle resources [2269].

Q Multi-tenant quantum computing

There are even proposals to enable the simultaneous execution of multiple circuits from different users or multiple instances of the same circuit on the same quantum processor, which bears its own security issues [1297].

Three 2026 multi-tenant quantum processor proposals were made in 2026:

- DynQ derives dynamic, topology-agnostic virtual execution regions directly from live calibration data, using quality-weighted community detection to maximize internal gate quality while minimizing inter-region coupling [2270].
- HALO evaluated on the 133-qubit IBM Torino processor, combining a hardware-aware helper-qubit placement algorithm with a shot-adaptive scheduler so that several user processes can safely co-execute on the same chip [2271].
- DoE and Cascade Quantum (USA) proposed and tested in September 2026 the execution of multiple instances of the same circuit on the same quantum processor [2272], with a small number of qubits (5 to 16).

The quantum computing cloud infrastructure relies a lot on Jupyter notebooks and tools like Kubernetes that au-

tomate the deployment, management, and orchestration of containerized applications [2273]. Jupyter notebooks package the configuration of interactive Python (or other languages) code development with all the related libraries and tools. They are embedded in containers package applications containing all their dependencies that are managed as Kubernetes, which provide the tools for automated rollouts and rollbacks of computing jobs as well as self-healing, security management and monitoring. Kubernetes abstract the underlying infrastructure so that applications can be deployed consistently whether they are running on-premises, in a public cloud, or in a hybrid cloud environment [2274]. There is even a proposal for Qubernetes which are adapted to hybrid classical and quantum workloads [2275]. As quantum platforms will mature, new cloud orchestration mechanisms will be put in place to dynamically allocate QPU resources based on user requirements and other infrastructure constraints [2276].

A related shift in software engineering philosophy, the Service-Oriented Quantum (SOQ) paradigm, proposes treating quantum capabilities as autonomous, composable and interoperable services rather than as auxiliary components bolted onto classical systems, in contrast with the earlier Quantum Service-Oriented Computing (QSOC) approach [2277].

In the FTQC regime, things might get complicated as some algorithms will have very long execution times, potentially in the month domain. Cloud resource allocation will require specific flexible architectures like with the distribution of quantum processing on several QPUs connected through entanglement resource sharing using photonic links. This distribution will have to be dynamic with flexible routing like on the Internet so that virtual QPU provisioning can be done on an ad-hoc basis.

Cloud quantum solutions usually also provide access to emulation solutions. This can be found at IBM, Microsoft, Google, and OVHcloud, to name a few. If the cloud is to run a hybrid algorithm, it must also provision classical datacenters or HPC resources and synchronize their availability with the related QPU (and right now, we're using only one QPU at a time). It will use specific software stacks and middleware tools [2278].

2.6.2 Cloud offerings

Quantum cloud offerings started to be launched around 2016, first in North America then in Europe and in China. They usually mix the access to quantum code emulators, NISQ gate-based quantum computers and analog quantum computers (annealers, cold-atoms based analog simulators).

Beyond vendor announcements, one of the few systematic views of what actually happens between submitting code and retrieving a result comes from a three-month study by ARLIS from the University of Maryland. Run-

ning the same algorithm on multiple cloud platforms and hardware backends, the study logs connection metrics, execution times, the effect of varying qubit counts, comparisons against simulators, and costs. It provides data points on the practical, hardware-dependent friction of using today's quantum cloud toolchains rather than new algorithms or optimizations [2279].

Figure 153 provides a summary view of these cloud-based quantum computing offerings, distinguishing between emulating quantum code on classical computers and executing quantum code on quantum computers and focusing on the Western world offerings.

North America

The first company to launch a quantum cloud offering was **IBM**, which started to make its first QPUs online in 2016. It proposed cloud access to multiple QPUs with 120 to 156 superconducting qubits as of September 2026.

D-Wave launched its Leap cloud solution in 2018 with its quantum annealers.

Rigetti followed with its cloud service launched in 2019.

Quantum cloud offering also started to be proposed in 2020 by cloud vendors selling access to third-party quantum computers, mixed with quantum emulation resources on conventional servers. This is what Amazon and Microsoft announced almost simultaneously at the end of 2019 and made available in 2020. Google followed on in 2021.

Microsoft announced in November 2019 that it was integrating a quantum computing offering into the Azure cloud, and relying on **IonQ** and **Honeywell** (trapped ions qubits) as well as **QCI** (superconducting qubits). As of spring 2022, no QCI QPU was available online, seemingly since their qubits are not yet operational.

Rigetti (superconducting qubits) was added to Azure Quantum in December 2021 (in "private previews" as of March 2022) and **Pasqal**'s quantum simulator addition was announced in May 2022. Microsoft is associated among others with **1QBit** (Canada) to propose quantum software application layers. They promote quantum inspired algorithms that rely on traditional cloud resources, as in this case study of MRI scanner optimization at Case Western Reserve University [2280]. Microsoft and **KPMG** announced a partnership related to the development of quantum inspired algorithms in December 2022.

Microsoft Azure Quantum also supports Qiskit and Cirq Python-based quantum code since October 2021. You can test Azure Quantum for free with a credit of \$500 and even win a credit of \$10K after submitting your applications. Microsoft is also selling the access to **Toshiba**'s Simulated Quantum Bifurcation Machine+ (SQBM+), a classical Ising model solver using quantum inspired models [2281]. In March 2023, Microsoft

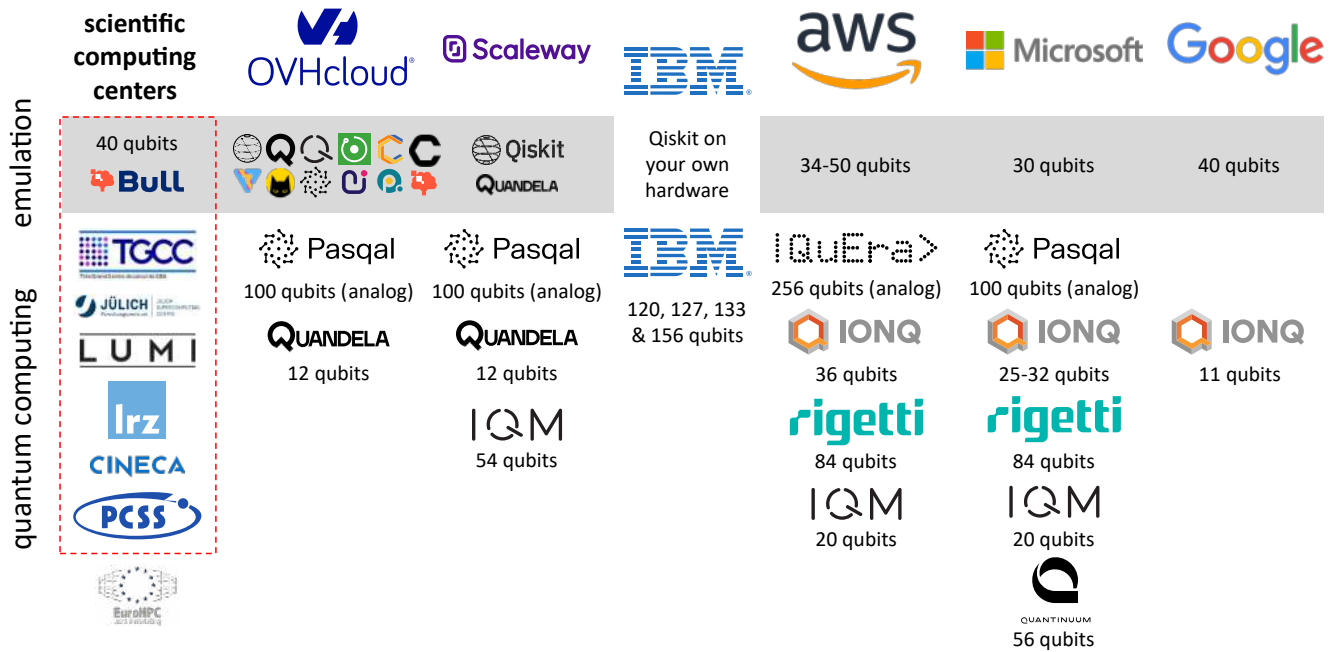


Figure 153: Main quantum cloud emulation and QPU offerings in the USA and Europe. IBM discontinued its online Qiskit emulator in May 2024. It can be used on customer’s laptops or with other cloud vendors like OVHcloud. © Olivier Ezratty, 2021-September 2026.

launched its hybrid quantum computing platform starting with working with Quantinuum [2282].

Amazon made its entrance in the quantum cloud market at the end of 2019 with the announcement of three components: Amazon Braket cloud services, AWS Center for Quantum Computing at Caltech University, and Amazon Quantum Solutions Lab, a customer evangelization program reminiscent of IBM’s Q initiative. Amazon also uses the brand Quantum Compute Cloud (QC2) for its offering. Braket is their whole quantum software infrastructure.

It provides access to quantum computers from D-Wave (the 2000Q and Advantage annealers with respectively 2,048 qubits and 5,000 qubits), IonQ (it started with 11 qubits), Rigetti (16Q Aspen-4 with 16 superconducting qubits and their 31 qubits Aspen-9 version, and later, their 80 qubits system) and OQC (with 8 coaxmon qubits, in 2022). IonQ is thus proposed by both Microsoft and Amazon. Amazon announced in November 2021 that QuEra’s cold atoms systems running in quantum simulation mode (*aka* “Hamiltonian simulation”) would be added to AWS Braket in 2022. Late 2021, Amazon Braket Hybrid Jobs was added to their software portfolio with the capacity to run hybrid algorithms associating classical and quantum computation [2283, 2284]. Amazon also proposes software emulation of quantum algorithms on classical servers.

Amazon Braket is associated with an in-house SDK based on the classic Python language. Development is supported in the integrated open-source environment Jupyter. It also includes support for the OCL (Object Constraint Language) constraint programming language.

Like Microsoft, Amazon is also a partner of quantum software vendors. We find almost the same players with Xanadu, Zapata Computing (dead since 2024), Rahko, QC Ware, 1Qbit and Qsimulate.

Google does not offer any public access to its own Sycamore quantum computers in its cloud offering. Its Sycamore QPUs are only available to a handful of academic institutions in a sort of “private cloud” access. Google still has a digital emulation offering supporting up to 40 qubits. In June 2021, it also announced the integration of IonQ’s 11 qubits processor in its Cloud Marketplace. IonQ becomes de-facto the most distributed solution in the cloud with Amazon, Google and Microsoft.

Equinix (USA) is another cloud player involved in quantum computing. It provides colocation data centers on a global basis for public and industry customers. They got into the quantum computing field in 2023 through two announcements: the installment of a 32-qubit OQC QPU in their Japan datacenter and their partnership with Alice&Bob. Equinix customers will access their software emulator and their future QPUs through their Equinix Metal and Equinix Fabric infrastructure offering.

Likewise, their competitor **Cyxtera Technologies** (USA) launched in 2023 its AI + Quantum Center of Excellence (COE), a hub for companies exploring how to leverage AI and quantum computing located at their LHR3 data center in Reading, UK [2285]. In quantum computing, they are partnering with OQC with whom they had already installed their first QPU in the UK cloud in 2022.

Sygaldry Technologies is a startup launched by Chad Rigetti, Idalia Friedson and AI scientist Michael Keiser which is based in Michigan. It plans to deploy quantum-accelerated AI servers to bring exponentially speed up training and inference. They even hint that they are building quantum computers meeting the specific requirements for AI. The company provided no technical details as of Summer 2026. We know only that the company plans to use multiple qubit modalities. One key concern is not just the hardware but machine learning algorithms and their practical speedup for real enterprise use cases.

In August 2026, they published a preprint describing a method to handle flow matching and score-based diffusion for generative AI model inferences. It associates tensor networks methods and a quantum circuit which could bring some quadratic speedup over Monte Carlo sampling methods. It could enable rare-event sampling via amplitude amplification across complex domains like computer vision, time-series forecasting, and computational biology and also help prepare the initial ansatz of variational circuits. [2286].

Europe

France-based OVHcloud entered the quantum computing field in 2022, initially focusing on cloud-based quantum simulation and emulation environments. The company began making quantum development tools available through dedicated cloud notebooks [2287, 2288]. As of September 2026, OVHcloud provides 18 preconfigured quantum software environments through its quantum emulators offering. These environments span quantum SDKs, frameworks, simulators and emulators rather than representing hardware emulators in the strict sense. They enable developers, researchers and students to design, simulate, benchmark and optimize quantum algorithms using managed cloud notebooks and classical CPU or GPU resources [2289, 2290]. In March 2023, OVHcloud purchased its first quantum computer, Quandela's MosaiQ photonic system. The machine was delivered to an OVHcloud datacenter at the end of 2023 and officially inaugurated in Croix, France, in March 2024. The QPU was used for R&D, experimentation and education around practical [2288, 2291]. The main use case focused on quantum-generated randomness to issue SSL certificates for websites hosted on its infrastructure. In September 2025, OVHcloud announced the rollout of the technology to its web-hosting customers [2292]. OVHcloud launched its quantum platform in November 2025, introducing a Quantum-as-a-Service (QaaS) model in which physical quantum processors can be accessed remotely and billed on a pay-as-you-go basis. Pasqal's 100-qubit Orion Beta was the first system made commercially available through the platform. In April 2026, the platform expanded with Quandela's Belenos, a 12-qubit

photonic, gate-based quantum computer. OVHcloud stated that the platform will provide access to at least eight quantum computers, with a roadmap focused on European vendors [2293–2295]. In June 2026, OVHcloud also announced a research collaboration with Weling aimed at designing the quantum datacenters of the future. The collaboration focuses on the interconnection and orchestration of quantum computers based on different qubit modalities, together with classical computing resources. This initiative extends OVHcloud's quantum strategy beyond access to individual QPUs towards the development of distributed and interconnected quantum computing infrastructures [2296].

Also in France, **Scaleway** is proposing a paid “quantum as a service” with the support of two emulators, one for Qiskit and one for Perceval from Quandela, that runs on Nvidia Tesla P100 and H100 GPUs. It also provides access to various quantum computers from Pasqal, Quandela, IQM and AQT.

In Germany, cloud provider and web hoster **IONIS** announced in November 2022 a partnership with QMWare, the University of Stuttgart and Fraunhofer FOKUS to create the **SeQenC** cloud platform for the German industry with over €10 three-year project funding by the German government (Germany's Ministry of Economic Affairs and Climate Action, BMWK) [2297].

Q Accessing quantum computers for free

On top of using free and open-source quantum circuit emulators like Quirk, quantum developers can also access multiple quantum computers for free across the world. What's free exactly? Usually, you register and get some usage credits in \$s or computing time. Some of them are one-time credit when you create the account (AWS, Microsoft Azure, Scaleway), and others are offered per time slot (month, day, ...).

In his Quantum Benchmark Zoo website, Valentin Gilbert provides an inventory of free access quantum computers [2298].

Otherwise, regular paid access to quantum processors on the cloud is charged for about \$1K to about \$7K per computing hour depending on the system and its capability. These are NISQ pricing, which may later change for FTQC systems. An inventory of the offerings is provided in a table in the quantum engineering section / economics part from this book.

Always in the European union, **Quantum Inspire** in the Netherlands proposes an access to small superconducting and silicon spin qubit platforms and a code emulator. Developed by QuTech, it enables quantum algorithms to be run on conventional computers in em-

ulation mode. It provides a graphical programming interface in the QASM language. The code can then be executed in emulation mode in the cloud on a classic machine, the Dutch national supercomputer Cartesius, with 5, 26 and 32 qubits, depending on the chosen package. Cartesius is equipped with thousands of Intel Xeon and Xeon Phi CPUs and a few dozen Nvidia Tesla K40m GPUs with 130 TB of memory delivering 1.84 PFLOPS. The equipment comes from Atos. Quantum Inspire also provides cloud access to QuTech 5 superconducting qubits and 2 electron spin qubits since April 2020. It added its 17 superconducting qubit system in 2026, named Tuna-17.

China

In May 2023, an academic quantum computing cloud platform was announced resulting from the collaboration between the Beijing Academy of Quantum Information Sciences (BAQIS) and its Quafu Group, the Institute of Physics (IOP) from Chinese Academy of Sciences, and Tsinghua University [2299]. The platform that was launched in November 2022 to early beta testers is providing access to three superconducting qubit QPUs with respectively 10, 18 and 136 qubits [2300, 2301]. **QuantumCTek** is also providing some interconnect infrastructure between these three QPUs that are in different places. As of 2026, the platform hosts 9 systems, Baihua (156 physical qubits but only 111 usable) [2302], Dongling (130 physical and 105 usable) [1125] and Haituo (117 physical and 105 usable) are then, six smaller chips with 3 to 21 qubits.

China Mobile created its own cloud quantum platform in 2023, Wuyue QuantumAI. It provides access to multiple QPUs from China: a photonic system from QBoson with 100 qubits, superconducting QPUs from USTC with up to 110 qubits, from CAS with 21 qubits, from Dianke System with 20 qubits (the company is affiliated with China Electronics Technology Group Corporation (CETC), a state-owned enterprise involved in defense and advanced technologies), and a 20-qubit trapped ions QPU from Guokai Ke Quantum Technology (a mysterious company with no trails online).

China Telecom launched in December 2024 a cloud quantum computing platform named Tianyan with several superconducting QPUs including the Tianyan-24 developed by China Telecom, the Tianyan-176 corresponding to the Zuchongzhi 2.0 with 66 qubits and 110 tunable couplers from Hefei-USTC, Tianyan-176-II, an improved version of the former, and Tianyan-504, corresponding to the Xiaohong-504 QPU developed by China Telecom Quantum Group (CTQG), in collaboration with CAS and QuantumCTek (but with no support for two-qubit gates) [2303, 2304]. In December 2025, they published a preprint on the performance of the Tianyan-287 (the number standing for 105 qubits and 182 couplers), the commercial version of the Zuchongzhi with 105 qubits on which they ran a quantum supremacy

experiment similar to the one from Google Willow published in 2024 but with only 74 qubits [2305].

Baidu promoted its Quantum Leaf cloud offering that was provided with **Paddle Quantum**, a quantum machine learning development toolkit based on the PaddlePaddle programming language, **QCompute**, a Python-based open-source SDK and **Quanlse**, a machine-level programming tools controlling the pulse sent to emulated superconducting qubits. **Alibaba** had also some cloud quantum emulation offering. But both Alibaba and Baidu withdrew from the quantum computing market in 2023.

South-Korea

QAI

South-Korea, 2025



QAI is a Seoul-based quantum-AI data center operator led by Seman Im (CEO) and Mark Hwang (director in charge of quantum business strategy). The company is an infrastructure aggregation. It buys quantum hardware and packages it with GPU capacity inside urban edge data centers it plans, designs, and operates. It acquired SDT's 20-qubit superconducting system KREO SC-20 and its QubeStack hybrid quantum cloud software to install it in its data centers. They also plan to deploy a Qudora (Germany) trapped ion system in 2027. They source software from external vendors including Classiq. Among their customers are BioNSystems, LG Electronics, Devcra, and Garam IDC.

2.6.3 HPC-quantum hybrid computing

HPC-quantum hybrid computing is a special case of cloud infrastructures we just dealt with. The differences are manifold: first, HPC centers are frequently managed by academic organizations and are shared resources that are accessible on a per-project and RFP basis. Second, the related HPC infrastructure is heavier than the clusters from most cloud vendors. Third, these centers are mostly focused on scientific computing with a significant workload that could be accelerated or extended by quantum computers [1790, 2306] (Figure 156).

A broader architectural vision along these lines is Quantum Integrated High-Performance Computing (QHPC), which proposes unifying CPUs, GPUs, FPGAs and QPUs as first-class heterogeneous resources under a layered system spanning resource management, quantum-aware scheduling, hybrid workflow orchestration, middleware abstraction and interconnect technologies [2307].

Most of these centers are working on the integration of quantum computers at various levels like classical interconnection, developing the software infrastructure and designing some hybrid classical-quantum algorithms. A function requires close integration of QPUs and HPCs, in the case of quantum error correction. The classical

Japan		China				
national research infrastructure		 Fujitsu Hybrid QC Platform with RIKEN	 Tianyan	 Wuyue QuantumAI	 Origin Quantum Cloud	 Quafu Cloud Quantum Computing Platform
emulation		quantum simulator on Fugaku (classical)	classical HPC co-processing	info N/A	full/partial/single-amplitude & noisy simulators	info N/A
quantum computing		RIKEN Wako Fugaku and Quantinuum Reimei	Tianyan-504 504 qubits (Xiaohong-504)	Qboson 100 qubits	Wukong-180 180 qubits (2026)	Baihua 119 qubits
		AIST / G-QuAT Tsukuba — QuEra 37 logical qubits (analog, 2025)	Tianyan-176 and 176II 66 qubits (Zuchongzhi 2.0, 2.1)	USTC 110 qubits	Wukong 72 qubits (2024)	Dongling 105 qubits
		Osaka U. (QIQB) Fujitsu chip, 64 qubits (2025)	Tianyan-287 105 qubits (Zuchongzhi 3.0)	CAS 20 qubits		Haituo 105 qubits
		U. Tokyo IBM Quantum System One 127 qubits (Eagle)		Dianke Systems / CETC 21 qubits		+ 6 smaller chips with 3 to 21 qubits

Figure 154: The various HPC and commercial or academic quantum computing available in Japan and China as of 2026. Notice that China Telecom and China Mobile rebrand quantum processors provided by academic labs. © Olivier Ezratty, September 2026.

systems handling these tasks won't be supercomputers but still HPC, using servers like the Nvidia DGX appliances. It led Nvidia and scientists from the DoE and other labs to propose in October 2025 the NVQlink integration proposal between HPC and QPUs using regular Infiniband links, connecting HPCs to the electronic devices handling qubit control across multiple qubit types [1708].

In the European Union, hybrid classical-quantum offerings are launched that target the academic and industry communities. These are associating QPUs and supercomputers. The European project **HPC-QS** (for quantum simulation) is deployed in three sites: in Finland (CSC LUMI), Germany (Munich at the Jülich Supercomputing Centre) and France (Bruyères-le-Châtel at CEA, as part of the France HQI initiative). It started with attaching an Atos QLM (Qaptiva) emulator appliance to these sites' supercomputers, themselves all running classical CPUs and GPGPUs from Nvidia. Pasqal analog quantum simulators were first installed in Jülich and at the CEA TGCC in 2024 and are operational as of 2025. Another one is to be installed at Cineca in Italy.

As part of the EuroHPC Joint Undertaking, 10 QPUs have been deployed or planned to be deployed in 7 European countries. In Finland, IQM are installed at LUMI. In Germany, a 20-qubit IQM system was deployed with an Eviden QLM at the Leibniz Supercomputing center (LRZ) in Garching near Munich, as part of the project Q-EXA. LRZ also hosts a 20 trapped-ions qubit AQT QPU. These hybrid computing centers mainly serve the needs of academic researchers and large companies. You have also an AQT QPU at the Poznan Supercomputing and Networking Center in Poland (Figure 155 lists all projects and vendors).

Estonia launched in 2024 NordIQuEst, a hybrid HPC-QC collaboration with Nordic countries [2308].

In the USA, the main supercomputing centers are managed by the DoE, like the Oak Ridge National Laboratory and its Frontier exascale classical supercomputer.

In Japan, the JHPC-quantum project funded by NEDO hosts the IBM Kobe System 2 QPU and REIMEI's trapped ions QPU along with Fujitsu's Fugaku supercomputer. RIKEN's Fujitsu Fugaku is also used jointly with Quantinuum QPUs. AIST also hosts G-QuAT, home to ABCI-Q, a research supercomputer dedicated to quantum computing, powered by 2,020 NVIDIA H100 GPUs. It is connected to multiple QPU quantum processor types including superconducting (Fujitsu), neutral atom (QuEra), and photonic qubits (OptQC).

HPC-quantum computer integration and co-location will bring new constraints, questions, practices and middleware [2309]. At this point in time, it is still unclear whether such co-location is mandated from a communications bandwidth point of view. It depends on the workloads [2310, 2311]. Most academic papers being published on that topic are very conceptual and lack resource estimates [2312]. A 2026 model jointly developed by NVIDIA, AWS, NERSC and NASA teams starts to close this gap. It decomposes workflow execution into quantum compute, classical compute and communication costs, deriving a communication-to-computation ratio to judge whether a workload is compute-bound or communication-bound, and a feasibility constraint tied to the reaction time set by the logical clock speed of fault-tolerant computation, to determine whether close co-location is required at all [2313].

On the middleware side, two community efforts are proposing ways to standardize interfaces between HPC

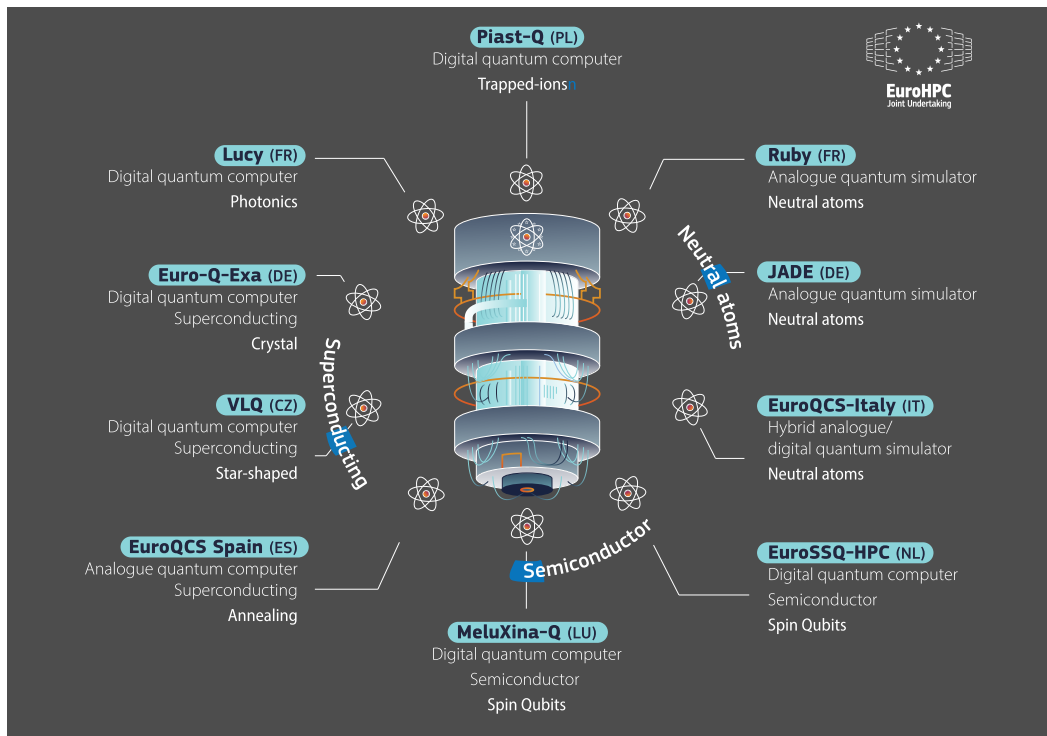


Figure 155: The quantum computers deployed or to be deployed as part of the EuroHPC program. Here are the corresponding QPUs: Piast-Q in Poland: AQT 20 qubits, Ruby in France: Pasqal, >100 cold atoms in analog mode, Jade in Germany: Pasqal with the same configuration, EuroQCS-Italy: Pasqal, EuroSSQ-HPC in the Netherlands: SURF project based on TU Delft work, MeluxinaQ in Luxembourg: 10 spin qubits of origin to be decided, EuroQCS Spain: Qilimanjaro, VLQ in Czechia: IQM, Euro-Q-Exa in Germany: IQM with 54 and then 150 qubits, Lucy in France: Quandela with at least 12 qubits. Source: EuroHPC, September 2025.

Q Should HPC and quantum computers be colocated?

The colocation of quantum computers and HPCs will be practically limited by different physical constraints that we already described in the part on systems engineering. Quantum computers usually need a low noise environment, a very stable temperature, and low ambient electromagnetic fields [2314]. In data-centers, quantum computers will be located in different rooms than regular data center and HPC resources. They will be connected through high-speed fiber links which can operate at short and long distances.

The classical computing resources that need to be close to the quantum processor are the compiler and scheduler, the classical optimizer for NISQ variational

circuits, and error syndrome detection decoders for FTQC systems.

What can be remote is the classical software handling circuit generation or problem embedding, provided the data sent to the computer are of reasonable size to accommodate for regular data center fiber links. These operate at bandwidths in the hundreds of GB per second which could support large FTQC circuits for a while. Likewise, HPCs may need to receive circuit shots data for handling their post-processing. Given these are raw classical bits, the generated data volume should be reasonable as well.

At small scale though, a NISQ variational classical optimizer can be run on an HPC as described in [2315].

schedulers and quantum computers. The Quantum Resource Management Interface (QRMI) is a Rust-based, vendor-neutral middleware layer supporting Rust, C and Python that exposes quantum computers as schedulable resources alongside classical CPUs and GPUs. Initially validated with Slurm, it has since been extended to PBS, LSF, Grid Engine, Kubernetes and the Flux Framework [2316]. The Quantum Device Management

Interface (QDMI) takes a lower-level, quantum hardware agnostic approach, defining a C-level standard for device discovery, capability queries and job lifecycle management, demonstrated end-to-end with a Slurm-based execution path and a Qiskit-facing frontend on IQM superconducting hardware [2317]. In 2026, a survey of nine production quantum-HPC software stacks led by ORNL, NERSC and TUM researchers analyzed these

and other approaches to propose openQSE, a layered reference architecture meant to let different runtime, resource-management, orchestration and execution implementations interoperate for both NISQ and future FTQC workloads without changes to upper-layer application interfaces [2318]. At a lower simulation level, a topology-aware discrete-event simulator, HybridQC, models tightly coupled hybrid compute units as configurable graphs of classical processors, memory, controllers and quantum annealing or gate-based quantum computers, to analyze controller bottlenecks and the diminishing returns of added quantum computing capacity [2319].

Reproducibility will also need to be addressed in specific benchmarking techniques. Can we reproduce the result of some quantum algorithm when executing it several times and on different platforms? Also, systems resiliency to faults will have to be understood beyond the scope of quantum error correction, in cryogenics, control electronics and the compiled code driving the whole. In scientific computing, this is going to be important for cases where algorithms run-time are extremely long. One 2026 paper argues that resilience must become an *a priori* design constraint of HPC-quantum computer integration rather than an afterthought, and proposes borrowing quantitative resilience models and metrics from civil engineering to estimate how vulnerabilities propagate up the classical-quantum computing stack [2320].

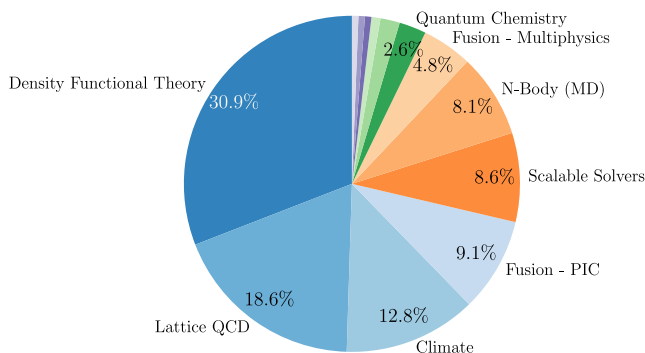


Figure 156: What are the classical applications run on the DoE supercomputers. Source: [1790].

2.6.4 Space as the new cloud?

With Elon Musk hinting that it would make sense to install quantum computers in lunar craters to benefit from vacuum and a low ambient temperature, and Google’s plan to send datacenters in space as part of their Sun-catcher project [2321], one can wonder whether it would make sense to send a quantum computer in space or farther.

There are heterogeneous reasons why this may or may not make sense. We have vacuum, cold temperatures and microgravity [2322]. The 2.7 K figure usually quoted is the cosmic microwave background radiation temperature, not the equilibrium temperature of an actual lunar or orbital installation, which is set by solar illumination, shielding and the heat-rejection budget. Either way it is far too hot for superconducting qubits, which still need their own dilution refrigerator, and rejecting that heat is harder in vacuum than on the ground. You are isolated from various sources of noise like vibrations but you are exposed to a wealth of cosmic rays that could damage both quantum circuits and their surrounding classical electronics, unless highly protected. While satellite communications could be used for QKD interconnect, it would be too slow for high-bandwidth and low-latency communication between QPUs and their surrounding HPCs. Evidently, the cost would be prohibitive given the mass and volume constraints of QPUs, particularly FTQC ones. And how about maintenance and upgrade? These fragile systems would also have to tolerate the vibration of a rocket launch. Launching such devices in space would also add to existing complicated space qualification processes. All power and cooling management would have to be revised for all components. In the end, the cost would quickly offset the benefits, whatever the qubit type. There may be some viability with lightweight photonics-based QPUs that could be used for distributed quantum computing networks using satellite relays.

And the moon? You’d have the same problems, and even more complicated and costly. It may be relevant to think about all of this when quantum computers are way more mature than they are today.

2.6.5 Bibliography and notes

These are the bibliographical references and notes for the **Cloud quantum computing** subsection.

- [1125] [Hierarchical Quantum Optimization via Backbone-Driven Problem Decomposition: Integrating Tabu-Search with QAOA](#), by Minhui Gou, Wei-Feng Zhuang, et al., arXiv, April 2025 (10 pages).
- [1297] [Enabling Multi-programming Mechanism for Quantum Computing in the NISQ Era](#), by Siyuan Niu and Aida Todri-Sanial, arXiv, February 2023 (34 pages).

- [1708] [Platform Architecture for Tight Coupling of High-Performance Computing with Quantum Processors](#), by Shane A. Caldwell, Travis Humble, et al., arXiv, October 2025 (43 pages).
- [1790] [Quantum Computing Technology Roadmaps and Capability Assessment for Scientific Computing - An analysis of use cases from the NERSC workload](#), by Daan Camps, Nicholas J. Wright, et al., arXiv, September 2025 (69 pages).
- [2250] [Report: Quantum Computing as a Service Market to Hit \\$26 Billion by End of Decade](#), by The Quantum Daily, August 2021.

- [2251] [Quantum Computing as a Service Market Sizing - How We Did It](#), by The Quantum Daily, August 2021.
- [2252] [Quantum fault tolerance with constant-space and logarithmic-time overheads](#), by Quynh T. Nguyen and Christopher A. Pattison, arXiv, November 2024 (119 pages).
- [2253] [Quantum Computing in the Cloud: Analyzing job and machine characteristics](#), by Gokul Subramanian Ravi, Frederic T. Chong, et al., arXiv, March 2022 (13 pages).
- [2254] [Architectural Vision for Quantum Computing in the Edge-Cloud Continuum](#), by Alireza Furutanpey, Felix Truger, et al., arXiv, May 2023 (16 pages).
- [2255] [Enabling Multi-threading in Heterogeneous Quantum-Classical Programming Models](#), by Akihiro Hayashi, Thomas M. Conte, et al., arXiv, March 2023 (8 pages).
- [2256] [The Munich Quantum Software Stack: Connecting End Users, Integrating Diverse Quantum Technologies, Accelerating HPC](#), by Lukas Burgholzer, Martin Schulz, Robert Wille, et al., arXiv, September 2025 (18 pages).
- [2257] [Service Differentiation and Fair Sharing in Distributed Quantum Computing](#), by Claudio Cicconetti, Marco Conti, and Andrea Passarella, arXiv, January 2023 (30 pages).
- [2258] [Hypergraphic partitioning of quantum circuits for distributed quantum computing](#), by Waldemir Cambiucci, Regina Melo Silveira, and Wilson Vicente Ruggiero, arXiv, February 2023 (14 pages).
- [2259] [Stochastic Qubit Resource Allocation for Quantum Cloud Computing](#), by Rakpong Kaewpuang, Jiawen Kang, et al., arXiv, January 2023 (5 pages).
- [2260] [Qunicorn: A Middleware for the Unified Execution Across Heterogeneous Quantum Cloud Offerings](#), by Benjamin Weder, Lavinia Stiliadou, et al., arXiv, November 2024 (8 pages).
- [2261] [QSRA: A QPU Scheduling and Resource Allocation Approach for Cloud-Based Quantum Computing](#), by Binhuan Lu, Zhaoyun Chen, and Yuchun Wu, arXiv, November 2024 (4 pages).
- [2262] [Pilot-Quantum: A Quantum-HPC Middleware for Resource, Workload and Task Management](#), by Pradeep Mantha, Florian J. Kiwit, Nishant Saurabh, Shantenu Jha, and Andre Luckow, arXiv, May 2025 (11 pages).
- [2263] [CONQUIRE: A Co-Execution Environment for Quantum and Classical Resources](#), by Atulya Mahesh, Swastik Mittal, and Frank Mueller, arXiv, June 2025 (11 pages).
- [2264] [Building a Software Stack for Quantum-HPC Integration](#), by Amir Shehata, Peter Groszkowski, Thomas Naughton, Murali Gopalakrishnan Meena, Elaine Wong, Daniel Claudino, Rafael Ferreira da Silva, and Thomas Beck, arXiv, March 2025 (14 pages).
- [2265] [QCOEM: Quantum Cloud Orchestration with Evolutionary Multi-Objective Optimization](#), by Tam N. Pham, Hoa T. Nguyen, and Quan Le-Trung, arXiv, July 2026 (7 pages).
- [2266] [QuSplit: Achieving Both High Fidelity and Throughput via Job Splitting on Noisy Quantum Computers](#), by Jinyang Li, Travis Humble, Weiwen Jiang, et al., arXiv, March 2025 (11 pages).
- [2267] [Three ways to share a QPU: Scheduling strategies for hybrid Quantum-HPC applications](#), by Marco Cipollini, Daniele Gregori, et al., Future Generation Computer Systems, April 2026 (18 pages)  (preprint on arXiv).
- [2268] [Qurator: Scheduling Hybrid Quantum-Classical Workflows Across Heterogeneous Cloud Providers](#), by Sinan Pehlivanoglu, Ulrik de Muelenaere, Peter Kogge, and Amr Sabry, arXiv, April 2026 (28 pages).
- [2269] [Quantum Data Center: Perspectives](#), by Junyu Liu and Liang Jiang, arXiv, September 2023 (9 pages).
- [2270] [DynQ: A Dynamic Topology-Agnostic Quantum Virtual Machine via Quality-Weighted Community Detection](#), by Shusen Liu, Pascal Jahan Elahi, and Ugo Varetto, arXiv, January 2026 (35 pages).
- [2271] [HALO: A Fine-Grained Resource Sharing Quantum Operating System](#), by John Zhuoyang Ye, Jiyuan Wang, Yifan Qiao, and Jens Palsberg, arXiv, February 2026 (15 pages).
- [2272] [Parallel Circuit Execution for Scalable Quantum Computation](#), by Avimita Chatterjee, W. Michael Brown, Siyuan Niu, Wibe Albert de Jong, and Thomas Lubinski, arXiv, September 2026 (22 pages).
- [2273] [Improving Quantum Developer Experience with Kubernetes and Jupyter Notebooks](#), by Otso Kinanen, Andrés D. Muñoz-Moller, Vlad Stirbu, and Tommi Mikkonen, arXiv, August 2024 (6 pages).
- [2274] [Kubernetes-Orchestrated Hybrid Quantum-Classical Workflows](#), by Mar Tejedor, Michele Grossi, Cenk Tüysüz, Ricardo Rocha, and Sofia Vallecorsa, arXiv, March 2026 (8 pages).
- [2275] [Qubernetes: Towards a Unified Cloud-Native Execution Platform for Hybrid Classic-Quantum Computing](#), by Vlad Stirbu, Otso Kinanen, Majid Haghighparast, and Tommi Mikkonen, arXiv, July 2024 (31 pages).
- [2276] [QFOR: A Fidelity-aware Orchestrator for Quantum Computing Environments using Deep Reinforcement Learning](#), by Hoa T. Nguyen, Muhammad Usman, and Rajkumar Buyya, arXiv, August 2025 (20 pages).
- [2277] [Rethinking Services in the Quantum Age: The SOQ Paradigm](#), by Jose Garcia-Alonso, Antonio Brogi, et al., arXiv, October 2025 (39 pages).
- [2278] [A Conceptual Architecture for a Quantum-HPC Middleware](#), by Nishant Saurabh, Shantenu Jha, and Andre Luckow, arXiv, August 2023 (12 pages).
- [2279] [Three Months in the Life of Cloud Quantum Computing](#), by Darrell Teegarden, Allison Casey, F. Gino Serpa, Patrick Becker, Asmita Brahme, Saanvi Kataria, and Paul Lopata, arXiv, January 2026 (64 pages).
- [2280] [How the quest for a scalable quantum computer is helping fight cancer](#), by Jennifer Langston, July 2019.
- [2281] [Toshiba launches new SQBM+ quantum-inspired optimization provider](#), by Toshiba, June 2022.
- [2282] [Azure Quantum unlocks the next generation of Hybrid Quantum Computing](#), by Fabrice Frachon, March 2023.
- [2283] [Introducing Amazon Braket Hybrid Jobs – Set Up, Monitor, and Efficiently Run Hybrid Quantum-Classical Workloads](#), by AWS, November 2021.
- [2284] [PennyLane on Braket + Progress Toward Fault-Tolerant Quantum Computing + Tensor Network Simulator](#), by Jeff Barr, December 2020.
- [2285] [Cyxtera Launches AI + Quantum Center of Excellence to Empower Customers to Explore New Technologies](#), by Matt Swayne, The Quantum Insider, December 2023.
- [2286] [Scalable quantum simulation of continuous-time generative models via tensor networks](#), by Nathan X. Kodama, L. Andrew Wray, Sam Cochran, Chad Rigetti, Shravan Veerapaneni, and Michael J. Keiser, arXiv, August 2026 (33 pages).
- [2287] [Atos et OVHcloud s'associent pour accélérer le développement de l'informatique quantique](#), by OVHcloud and Atos, OVHcloud Newsroom, June 2022.
- [2288] [OVHcloud purchases its first Quandela quantum computer to spur innovation in the quantum ecosystem](#), by OVHcloud, OVHcloud Newsroom, March 2023.
- [2289] [Émulateur et simulateur quantique dans le cloud](#), by OVHcloud, OVHcloud Public Cloud, September 2026.
- [2290] [Quantum Emulators catalogue interface screenshots](#), by OVHcloud, OVHcloud Emulators catalogue; two screenshots supplied by the author, dated 24 September 2026, September 2026.
- [2291] [OVHcloud inaugurates quantum computer and announces new educational program to support the European quantum ecosystem](#), by OVHcloud, OVHcloud Newsroom, March 2024.

- [2292] [OVHcloud, the first global player to improve website access security with a quantum computer](#), by OVHcloud, OVHcloud Newsroom, September 2025.
- [2293] [OVHcloud dévoile sa Quantum Platform : la première offre européenne d'ordinateurs quantiques accessibles dans le cloud](#), by OVHcloud, OVHcloud Newsroom, November 2025.
- [2294] [OVHcloud accelerates the European quantum industry with Quandela's Belenos](#), by OVHcloud, OVHcloud Newsroom, April 2026.
- [2295] [QPU as a Service: your algorithms on the quantum cloud](#), by OVHcloud, OVHcloud Public Cloud, September 2026.
- [2296] [OVHcloud accelerates its quantum R&D and strengthens its Quantum Platform at France Quantum 2026](#), by OVHcloud, OVHcloud Newsroom, June 2026.
- [2297] [Germany to Create Its First Quantum Computing Business Cloud](#), by Germany Trade & Invest, November 2022.
- [2298] [Hardware pricing and access](#), by Valentin Gilbert, Quantum Benchmark Zoo, July 2026.
- [2299] [China Unveils New Quantum Computing Cloud Platform](#), by Jia Liu, May 2023.
- [2300] [Quafu-RL: The Cloud Quantum Computers based Quantum Reinforcement Learning](#), by BAQIS Quafu Group, arXiv, March 2024 (7 pages).
- [2301] [Quafu-Qcover: Explore Combinatorial Optimization Problems on Cloud-based Quantum Computers](#), by BAQIS Quafu Group, arXiv, May 2023 (14 pages).
- [2302] [Baihua: A 100-qubit scale, high-performance, and open-access quantum cloud platform](#), by Pei Liu, Wen-Gang Zhang, Jia-Jia Tian, Yi-Bin Guo, Hai-Feng Yu, Yi-Rong Jin, et al., Superconductivity, October 2025 (3 pages).
- [2303] [China Telecom Tianyan cloud platform](#), by China Telecom.
- [2304] [China's Largest Superconducting Quantum Computer "Tianyan-504" Launched](#), State Owned Assets Supervision and Administration Commission of the State Council, December 2024.
- [2305] [Tianyan: Cloud services with quantum advantage](#), by Tianyan Quantum Group, arXiv, December 2025 (9 pages).
- [2306] [New Report by Hyperion Research and Alice & Bob Recommends Near-Term Strategies for HPC Centers to Prepare for Quantum Integration](#), by Alice & Bob and Hyperion, Alice & Bob news, September 2025.
- [2307] [Quantum Integrated High-Performance Computing: Foundations, Architectural Elements and Future Directions](#), by Suman Raj, Siva Sai, Yogesh Simmhan, Kyle Chard, and Rajkumar Buyya, arXiv, April 2026 (30 pages).
- [2308] [NordIQuest: the Nordic-Estonian Quantum Computing e-Infrastructure Quest](#), by Costantino Carugno, Jake Muff, Mikael P. Johansson, Sven Karlsson, and Alberto Lanzanova, arXiv, June 2024 (15 pages).
- [2309] [Dependable Classical-Quantum Computer Systems Engineering](#), by Edoardo Giusto and Travis S. Humble), arXiv, August 2024 (7 pages).
- [2310] [Advantages of Co-locating Quantum-HPC Platforms: A Survey for Near-Future Industrial Applications](#), by Daigo Honda, Nobuyasu Ito, et al., arXiv, August 2025 (20 pages).
- [2311] [Dynamic Solutions for Hybrid Quantum-HPC Resource Allocation](#), by Roberto Rocco, Daniele Gregori, et al., arXiv, August 2025 (7 pages).
- [2312] [Interfacing Quantum Computing Systems with High-Performance Computing Systems: An Overview](#), by Konstantinos Rallis, Panagiotis Dimitrakis, et al., arXiv, September 2025 (28 pages).
- [2313] [Performance Model for Hybrid Quantum-Classical Workflows](#), by Pooja Rao, Elica Kyoseva, et al., arXiv, July 2026.
- [2314] [White paper: integrating quantum processing units into data center infrastructure](#), by Jonathan Burnett, Boniface Yogendran, et al., Open Compute Project, April 2026.
- [2315] [Enabling Hybrid HPCQC Workflows with a Heterogeneous Software Stack](#), by Muhammad Nufail Farooqi, Minh Chung, Burak Mete, Eric Mansfield, Bernd Hoffmann, Teemu Mattsson, Laura Schulz, and Jorge Echavarria, arXiv, August 2026 (9 pages).
- [2316] [Examining QRMI as a Unified Interface for Quantum-HPC Integration](#), by Thomas Badts, Aleksander Wennersteen, et al., arXiv, July 2026 (16 pages).
- [2317] [Practical HPCQC Integration with QDMI: A Real-Hardware Case Study with IQM Systems](#), by Lukas Burgholzer, Marcel Walter, Patrick Hopf, Álvaro Caride-Tabarés Sánchez, Teemu Mattsson, Bernd Hoffmann, Noora Färkkilä, Daniel Bulmash, Robert Wille, and Eric Mansfield, arXiv, April 2026 (11 pages).
- [2318] [Quantum-HPC Software Stacks and the openQSE Reference Architecture: A Survey](#), by Amir Shehata, Travis Humble, Martin Schulz, et al., arXiv, April 2026 (23 pages).
- [2319] [HybridQC: Hardware-Grounded Simulation of Tightly Integrated Hybrid Quantum-Classical Systems](#), by Panayiotis Christou, Shuwen Kan, and Ying Mao, arXiv, July 2026 (27 pages).
- [2320] [The Need for Quantitative Resilience Models and Metrics in Classical-Quantum Computing Systems](#), by Santiago Núñez-Corrales, arXiv, March 2026 (16 pages).
- [2321] [Orbital Compute Landscape: Edge to Infrastructure](#), by Match33, Match33 website, January 2026.
- [2322] [Microgravity and Near-Absolute Zero: A New Frontier in Quantum Computing Hardware](#), by Denis Saklakov, arXiv, December 2025 (27 pages).

This **Cloud quantum computing** subsection contains 77 bibliographical references and notes containing at least 1,054 pages.

2.7 Quantum software development tools key takeaways

- Gate-based programming involves either graphical circuit design (mostly for training purposes) and (usually) Python based programming when qubit-gate structures must be designed in an automated way. Key components of the software development toolset are compilers and optimizers as well as resource estimators, which help figure out the hardware configuration required to run some quantum circuit.
- Python-based programming relies on libraries like IBM's Qiskit or Google's Cirq. There are however many development tools coming from universities and research labs like Quipper. Some tools like ZX Calculus are highly specialized and used to create quantum error correction codes or low-level systems.
- Current NISQ quantum computing commonly relies on repeated circuit executions, often thousands or more, to estimate output probabilities or expectation values. In variational algorithms, a classical optimizer repeatedly updates circuit parameters between groups of quantum executions. A single run yields one probabilistic measurement outcome, while statistical estimates converge as the number of shots increases. VQE may require many circuit executions to estimate the expectation values of Hamiltonian terms with sufficient precision while a classical optimizer updates the ansatz parameters. Large circuits on fault-tolerant quantum systems may also require repeated executions depending on the algorithm and desired success probability, including algorithms using quantum phase estimation (QPE).
- Most quantum computers are used in the cloud, through offerings coming from the computer vendors themselves like IBM or D-Wave or from cloud service providers like Amazon, Microsoft, Google, OVHcloud and Scaleway. Similar offerings are available in China and Japan. In the long term, cloud quantum computing services architectures will need to be distributed and require dynamic routing of photonic entanglement resources to enable the creation of large virtual QPUs.
- Quantum emulators are very useful to learn programming, and test circuits until the emulators reach the limits of classical emulation. They support up to 20 qubits on a laptop, 40 on a server appliance and 50-55 on an HPC in state vector mode. It can exceed 100 qubits with using tensor-network-based emulators for shallow circuits. Emulators can help debug small-scale quantum algorithms. When these emulators include physical simulators of the underlying qubit physics as with Bosonic Qiskit and Quandelà Perceval, they help create algorithms that are error-resilient and design new quantum error correction codes. Quantum emulation is an indispensable part of any quantum cloud offering. Emulator are often called "simulators" which is a misnomer creating confusion with quantum simulators which are classical and quantum systems that can simulate quantum systems in general.
- Gate-based circuit debugging is a significant challenge as it is difficult to implement equivalents of classical-code breaking points. As a result, quantum code certification and verification is a new key discipline, particularly for distributed computing architectures such as the ones relying on the concept of blind quantum computing. Another key discipline is resource estimation, with tools that help figure out the hardware configuration necessary to run a given algorithm. It also provides indications on the related computing time which may be prohibitive and require future optimization techniques.
- Benchmarking quantum computers remains an unsettled field with many competing approaches, including methods used to assess claims of quantum supremacy or quantum advantage. Such claims must specify the computational task, the best classical baseline, the cost metric, the hardware and error-mitigation assumptions, and whether the result has been peer reviewed or independently reproduced. By 2026, several experiments had reported computational or scientific utility on NISQ hardware, especially for quantum-physics problems, but this does not by itself establish an industry-relevant quantum advantage. The book therefore distinguishes claimed advantage, peer-reviewed evidence and independent validation and advances in dequantization and quantum inspired classical techniques.

3 Quantum computing applications

Most algorithms mentioned in the previous parts are generally very low-level. How about assembling them into business solutions, market by market? In this section, we will explore the wealth of potential use cases and existing case studies in multiple user segments. Beforehand, we will propose a semantic guide and an evaluation framework of use cases and case studies. We then look at the potential market size for quantum computing according to analysts, which indirectly cast the relevance of quantum computing per industry segment. Finally, this part also contains an inventory of software, tools and service vendors. The quantum software industry is already surprisingly burgeoning despite quantum computers capabilities being still rather limited at this stage. NISQ computers are not ready yet for prime time deployment for most industry use cases. That is where we have a few case studies, usually related to fundamental physics simulations. Use cases relate most of the time to prospective applications requiring fault-tolerant quantum computers (FTQC). Their prospects depend on the industry vendor hardware roadmaps and algorithmic progress.

3.1 Case studies and use cases evaluation

You can discover here and there a lot of so-called use cases and case studies, coming from various sources: quantum computer industry vendors, their customers or software partners, and academic researchers. It creates a strong discrepancy between the generated perception of a business-ready quantum computing scene and the harsh reality. Most of the time, these case studies are small-scale proofs of concept and software prototypes. They have not yet reached production-grade scale nor brought any practical benefit compared to classical computing due to the limitations of existing quantum hardware.

These are useful as prospective tools. They correspond to a learning phase for research, startups, and industry. It is part of a readiness process that will speed things up if and when hardware will ramp up. And this ramp-up may happen progressively. Let's say you discover an analyst or IT vendor white paper or an academic review paper showcasing how quantum computing will transform your industry. It is usually written using the present tense, showcasing many practical case studies, sometimes with their scientific references.

Meanwhile, as of 2026, no known corporation has deployed these solutions in production. What explains this discrepancy? How can you assess these case studies' readiness for business-grade applications? It is a challenge that is dealt with in this section. We look first at the conditions that could enable quantum computing to add some competitive value compared with classical computing and then, at the ways to classify and assess case studies. Figure 158 and the part below present a framework of questions to ask when reviewing case studies and color codes showing potential red flags making the case study moot and green flag making it relevant [2323].

But beforehand, let's play a little bit with semantics and define the terms used in case studies and use cases wonderland.

Case studies are retrospective and empirical analyses of the deployment or usage of a (quantum) computing system in a real-world context. They investigate how a system performs, how it is adopted, or how users interact with it. Case studies help evaluate a solution's effectiveness, uncover lessons learned, or guide future developments. They provide quantitative and qualitative data about the deployed applications. Not just that it "*accelerated some business*" and provided some "*competitive advantage*".

Use cases present how some upcoming technology is intended to be used. They are prospective and structured descriptions of some system functionality and potential benefits from the point of view of an end user. They should include resource estimations based on real user production volume, data and operational constraints. In quantum computing, these are frequently FTQC cases. The use cases compilation by NQCC published in 2025 is a good example [2324].

Pipelines cover the content of use cases and case studies. In many situations, an end-user solution is not based on a single algorithm or quantum circuit, but is an assembly of many classical and quantum software components. For example, a good chemical simulation case study to develop some new material or new drug will combine several quantum algorithms to simulate different aspects: ground state energy, excited states, molecular pathways, molecular docking, spectroscopy, spectra in condensed phases, solvation energies, catalysis, etc.

Now, we segment existing and future cases relative to quantum computing applications, as shown in Figure 157. It starts with defining three potentially overlapping situations with:

- **Industry relevance** which corresponds to applications that are useful for some end-user. It works with

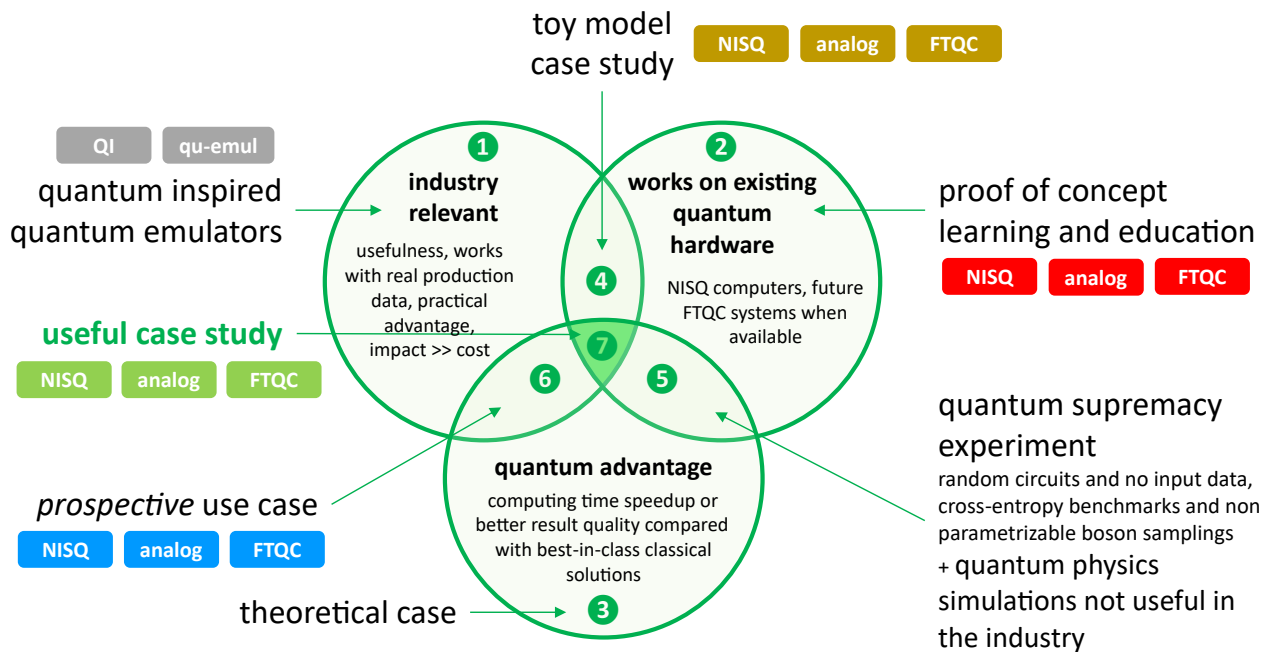


Figure 157: Semantics segmentation between three zones (industry value, runs on quantum hardware, provides some quantum advantage) and their potential overlap. It creates seven zones with distinct scenarios, which are detailed in the main text. Badges color codes: red for no industry value and no quantum advantage, orange for industry value without a quantum advantage, blue for future systems and green for quantum advantage on existing hardware and industry value. © Olivier Ezratty, June 2026.

real production data and not just a toy model. It brings a practical computational advantage versus legacy classical applications. Finally, its economic or societal value is superior to its economic cost and competitive with classical solutions.

- **Works on existing quantum computing hardware** which currently concerns gate-based and analog NISQ quantum computers. When available, future FTQC systems will sit there.
- **Quantum advantage** is about bringing some benefit vs legacy and best-in-class classical applications using the latest generation classical hardware (Nvidia GPU and DGX servers, or Cerebras wafer-scale systems as examples). It can be about providing some computing time speedup or better result quality.

This leads to seven possible existing or future scenarios considering the case when these three scenarios overlap with each other:

1. **Quantum inspired or emulated** cases correspond to quantum algorithms and applications tested or operational in production on classical hardware. They may provide some benefit versus existing legacy classical applications. But they can't be considered as being quantum applications per se. These are still classical applications running on classical hardware. They are industry relevant per the circle definition but do not provide a quantum advantage nor run on a quantum computer. Whenever useful, the cases will be labeled with **QI** and **qu-emul** badges.
2. **Proof of concept** with an end-user application being tested or operational. This can include existing gate-based and analog existing quantum computers and even early-stage small-scale FTQC systems that are either deployed on-premises or accessed through the cloud. They do not use real-life end-user datasets. These are for example the QML and VQE/QAOA case studies tested with a couple dozen qubits, whose circuits usually remain classically simulable at a much lower hardware cost than a quantum computer. Whenever useful, the cases will be labeled with **NISQ**, **analog** and **FTQC** badges.
3. **Theoretical cases** correspond to use cases with some potential theoretical quantum advantage, but that doesn't run on an existing quantum computer and has no industry relevance. This relates to algorithms with a proven theoretical advantage but which have not yet been attached to some industry relevant practical use case. One example is the forrelation algorithm from Scott Aaronson [263].
4. **Toy model case studies** correspond to the many case studies that have a potential interest for the industry and run on an existing NISQ and analog quantum computer but do not bring any practical quantum advantage even when they are using real-life end-user datasets. These are used to validate a solution before full-scale implementation. It may validate some algorithm in principle. A good proof of concept ends with some resource estimates for a solution delivering production-grade quantum advantage, and some cost-benefit analysis for validation. It may be

determined that a quantum advantage would require an analog quantum computer with more qubits, a NISQ system with better quality qubits or an FTQC system. Whenever useful, the cases will be labeled with **NISQ**, **analog** and **FTQC** badges.

5. **Quantum supremacy experiments** correspond to the various randomized circuits experiments run by Google and others which bring a computational advantage vs the classical simulation of the circuit, but do not use any input data. They sample from deliberately chosen circuit distributions used to test quantum versus classical computational performance and are not equivalent to practical certified random number generators. As a result, they are not relevant for any industry application. We also count here the non-parameterized boson sampling experiments run by China academic teams and other teams. To some extent, quantum physics simulations bringing a quantum advantage which are not relevant for the industry may sit there as well.
6. **Prospective use cases** correspond to industry relevant use cases that could provide a practical and theoretical quantum advantage but require some future quantum hardware, mostly FTQC-based, but also potentially NISQ with better qubit fidelities and analog quantum computers with a larger number of higher quality qubits. Whenever useful, the cases will be labeled with **NISQ**, **analog** and **FTQC** badges.
7. **Useful case study** is the sought-after holy grail with an industry relevant application bringing a practical quantum advantage and running on some existing quantum computing hardware. One variant is a benchmark study with a comparison between a deployed solution against competing systems or previous legacy architectures. Whenever useful, the cases will be labeled with **NISQ**, **analog** and **FTQC** badges.

More than two red flags should turn a “case study” into something labelled differently like a “new algorithm”.

Based on this seven-layer framework, we could segment available case studies in a simpler manner in four categories (Figure 159). The first are small-scale NISQ or emulator-based case studies that do not bring any quantum advantage. Even when the result brings some improvement over the tested baseline, these circuits can generally be simulated classically without a QPU and at a much lower cost [2325].

The second case corresponds to analog computing case studies that are not far from a quantum advantage and may even land in that field. The third and fourth cases correspond to future FTQC cases studies with about a thousand logical qubits and then tens of thousands logical qubits. These studies are interesting since they usually provide resource estimates for the supporting QPUs. It is commonly believed that useful chemical simulations will land into case 3 and case 4.

In a May 2024 study, the analyst firm GQI scrubbed 172 quantum computing use cases and used a more detailed segmentation, distinguishing toy demonstrations, classical simulations, problem formulation, theoretical approaches, resource estimations, in deployment and speculation. You can see in Figure 160 how resource estimations are poorly represented, and that actual deployments are not visible.

Let’s mention at last the model card published in 2025 by M.J. Everitt from Quantalytics (UK) that lists all the details to provide in a documented case study or use case with information on the user organization, the vendors, the project performance metrics, evaluation criteria, ethical considerations, assurability, etc. [2326].

Q “if” and “when”

Analysts like to say that the overarching question is more “when” than “if”. You have seen that nuance may be required here. The “if” is still an open question from a pure scientific standpoint. We have no idea about the behavior of large quantum systems, how we can control entanglement at scale, about new sources of correlated noise and other things. We also lack evidence of broad theoretical and practical exponential quantum speedups for the known classes of algorithms corresponding to industry business needs like with optimizations.

Nothing settles the “if” today. The two questions are not independent either: the weaker the evidence on the feasibility of large-scale fault-tolerant systems, the wider the spread on any timeline, and a timeline spanning several decades turns the “if” back into a practical question. A balanced assessment should therefore treat both the “if” and the “when” as unknowns. It is also a matter of humility when facing scientific and technology developments. You can encounter a mix of unpleasant surprises as well as find new amazing solutions to the problems that are currently identified.

3.1.1 Problem sizing

One first key aspect of a case study is the sizing of the problem. Is the conducted experiment toying with real-life data and matching usual business needs or just a small-scale prototype? How about the scale of the problems businesses are trying to solve more efficiently with quantum computers? It deals with the size of the data and the number of parameters or variables of a given problem, like a number of assets in a financial portfolio. Unfortunately, most existing case studies are using small-scale data and should be complemented with some indications on the real size quantitative scenario needs [2328].

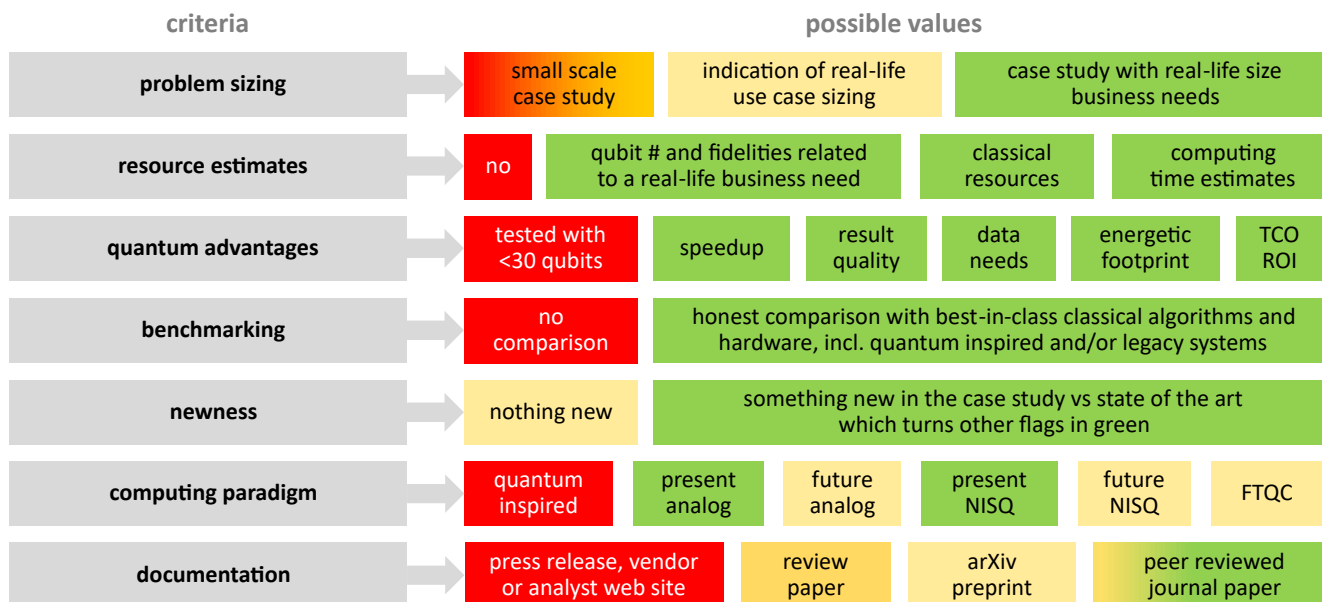


Figure 158: How to evaluate a quantum computing case study or use case. Red flags indicate key missing points in the case study making it irrelevant to show any advantage for quantum computing for the given business problem. Orange flags are intermediate situations but not real showstoppers. Green flags are indications of a serious use case. Quantum inspired use cases are not using quantum computing and therefore should not be presented as “quantum computing use cases”. TCO means total cost of ownership, an economic concept coming from the classical information technology world and ROI, return on investment, all being compared to 100% classical solutions. © Olivier Ezratty, 2023-2025.

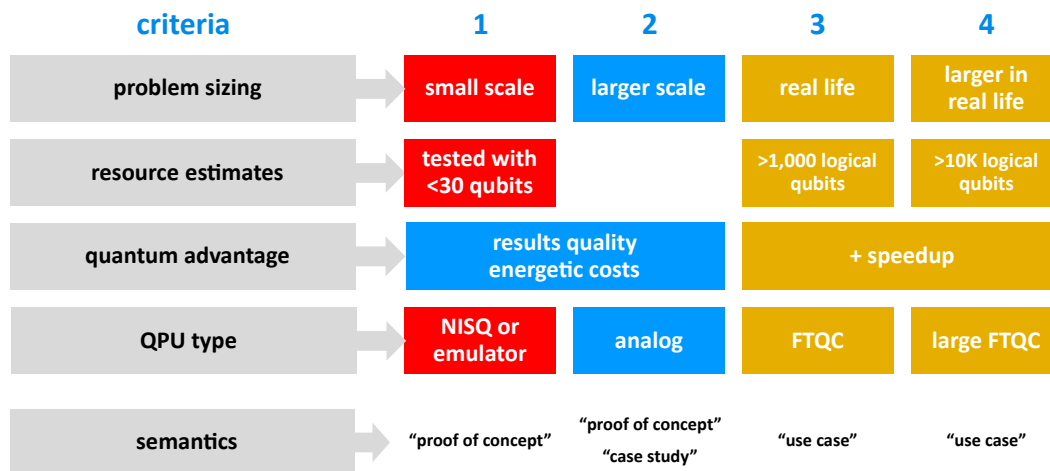


Figure 159: How to simply segment existing quantum computing case studies in four categories. © Olivier Ezratty, 2024.

3.1.2 Resource estimates

A good case study presents estimates of the classical and quantum computing resources needed to solve a given real-world problem and its relation to the problem sizing defined above. It is even more important when the solution was only tested with a small problem. You need to have some clues on the size and type of quantum computers which could be able to solve a real-life problem.

When looking at the details:

- **NISQ case studies** should evaluate the algorithm depth and breadth, the number of shots and the cost of classical computing for all variational algorithms.

Are the qubit counts and fidelities stated for the case study and for its extension to a real business-need sizing? In many cases, NISQ case studies relate to an algorithm that could provide some quantum advantage if run on an early-stage FTQC QPU due to the required number and quality of qubits.

- **FTQC case studies** should report the T count, or whichever non-Clifford resource the target architecture consumes, since magic state production often dominates the cost in surface code architectures. Clifford gates, routing, measurements, decoding and the overall space-time cost should be reported as well, together with the logical qubit fidelity target. Coupled with the physical qubit quality and connectivity, it leads to assessing the number of physical qubits.

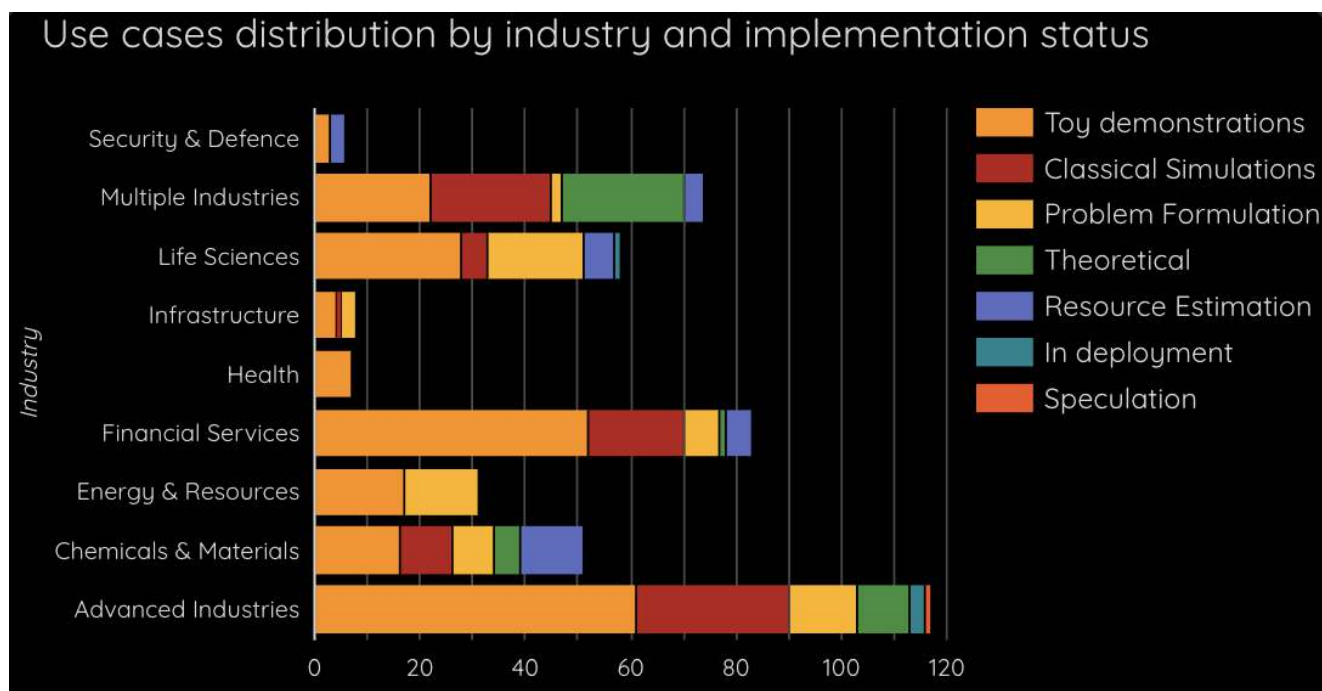


Figure 160: Chart created in 2024 showing how use cases are distributed among various verticals. It shows the usual strength of the financial services, life sciences and energy-chemistry sectors. Source: [2327].

It may be different if the target platform uses the MBQC paradigm, using photonic cluster states.

- **Analog quantum computing case studies** should also be associated with some data on the hardware requirement to obtain some quantum advantage like a number of qubits and their topologies for a quantum annealer and a quantum simulator.

Finally, gate-model circuits using a couple dozen qubits are usually classically simulable, which is the case for most of the use cases presented in this part. But width alone settles nothing: circuit depth, gate types, analog dynamics, noise, mid-circuit operations, the requested observables and the simulator method all matter. A matched runtime and cost comparison is required before claiming that a classical implementation is unchanged, faster or cheaper.

3.1.3 Quantum advantage nature

What is a quantum advantage? For a long time, it has been defined as a situation where a computing speedup can be observed between a quantum hardware and software combination, and a best-in-class classical solution. And preferably, this speedup was to be exponential, leaving in the dust of uselessness any classical computing solution for solving large scale problems. That is the speed advantage. But a speed advantage must be practical and not just theoretical. Do not confuse speedup and actual computing time. Complexity theory classifications are interesting for theoretical purposes but is useless if the actual computing time to solve a given problem exceeds a human lifetime!

And any comparison should include all the classical computing involved in any quantum computing solution [2127].

The quantum computing speed advantage is also usually confused with the exponential space advantage of quantum computing. It is generated by the large quantum state space of N qubits: a pure state is described by 2^N complex amplitudes, constrained by normalization and defined up to a global phase. These amplitudes are not an equivalent amount of usable data handling capacity.

As we have learned, this large “data handling space” is not sufficient to generate a polynomial or exponential speedup even though, when $N > 55$, this data space cannot be fully emulated on a classical computer due to memory constraints. A case study tested with a couple dozen qubits usually stays within reach of a classical simulation, but the absence of a speed advantage should be established with a matched runtime comparison rather than inferred from the qubit count.

the strongest relevant classical baseline, and removed the 20-qubit laptop rule. A quantum computing advantage can also be qualitative when the generated result is better than the strongest relevant classical baseline, without necessarily having been produced faster. This can happen for example with machine learning when it needs less training data or produces classification and prediction models with fewer errors. The term quantum advantage should be reserved for such a specified comparison. Absent a matched classical search, the honest wording is an improved result relative to the tested baseline. You should also make sure that the qualitative result is not accessible to classical computers, starting

with a classical simulation of the very circuit that was run, which is the common situation with small circuits. It would then be a red flag.

You may also trade quality for speedup, typically with the typical heuristic based approaches used in quantum computing, particularly in NISQ regime with variational algorithms. Comparisons may be touchy when you compare apples and oranges, like the best solution on one hand and one non optimal solution on the other hand.

Then, we have the energetics of the solution. It is becoming impossible to avoid considering this aspect when deploying new technologies. A quantum computing solution may provide some energetic advantage compared to classical solutions, but it remains to be proven on a case-by-case basis.

Some additional criteria for a use case or case study are proposed in [2220]:

- **Predictability:** can we foresee the advantage before we run it? This is based on accurate resources and figures of merit estimates, with some extrapolations needed.
- **Typicality:** is it common across problem instances or just a rare case? This is the genericity that we need to create a sizeable market.
- **Robustness:** does it survive noise and imperfections? It should be associated with the cost of quantum error correction and logical qubits and the related resource estimations.
- **Verifiability:** can we confirm the results without exponential classical effort? This is similar to the “*correctness of the output*” seen in another quantum advantage definition proposal from IBM and Pasqal, in [2221].
- **Usefulness:** does it matter for a relevant, real-world task? Preferably, one that adds significant value.

Finally, you can evaluate the **economic advantage** of the solution. It starts by looking at the total cost of ownership (TCO) of the new quantum solution. It includes all incurred expenses to deploy a solution: classical and quantum hardware, software, cloud, services, and training related costs. Studies are not yet available at this point given the immaturity of the technology. Sometimes, you find out that some quantum computers have ridiculously high catalog prices up to tens of millions of dollars. In that case, you are better off using cloud computing resources. You can then mix TCO with ROI (return on investment) considerations, given all cost and value ratios must be compared to classical legacy solutions. As explained in [2354], a “*Quantum economic advantage (QEA) is achieved if there exists a quantum computer that can solve a problem instance for X and all existing classical systems with a computation budget X or less would solve the problem in more time*”.

A paper from HQS gets into more details with many examples on what a quantum advantage would look like [2355]. It insists indirectly on the need for the quantum solution to be *significantly* better than state of the art classical solutions. They insist justly on the need to also invest as much effort as possible to solve the given problem with state of the art classical computers and software as into the research on quantum computing. They call this criteria “*Get the Job Done Without the Quantum Computer*”.

An economic benefit evaluation can be fine tuned with integrating various potential economies of scale. The first would be on the end-user side with a reduced marginal cost for subsequent projects based on the same tools and methodologies. The second would be on the supply side with economies of scales obtained by industry vendors as the market grows, the cost of the hardware decreases thanks to volumes and better technology integration and innovation.

3.1.4 Classical comparisons

Is there an honest and up-to-date comparison between the proposed quantum solution and classical equivalent solutions? It must rely on using the state-of-the-art hardware (HPC, GPUs, ...) and software (algorithms, heuristics, tensor networks, ...). The comparison must not mix apples and oranges between exact solutions and heuristic approximate solutions that would favor one side of the comparison. In the case where the quantum computing solution can be easily emulated on a classical system or was only emulated, it will mean there is no quantum advantage.

Various tricks can be used, and not necessarily voluntarily like claiming superior performance without mentioning computing runtimes, or avoiding including classical computing runtimes needed to run the quantum solution, forgetting to mention the tuning time needed to optimize the provided results, claiming faster runtimes in comparison to quantum solutions running on non-existent platforms and cherry-picking without justification and qualification the problem that enables some quantum advantage like having sparse data and the like [2356].

3.1.5 Algorithm advancement

Was there something new in the case study vs the state of the art in both classical and quantum computing? Where was the progress? Does it have some business consequences?

3.1.6 Quantum computer type

Quantum case studies belong usually to one of these categories from being implementable in the short term to the very long term:

Q Checklist for evaluating case study and use case claims

Here is a list of common weaknesses to look for as a customer, and to avoid when publishing a case study or a use case:

- Comparing two solutions with changing many parameters including the classical part, making comparisons difficult [2329]. The increased performance may come from the new classical part of the solution [2330].
- Describing a complicated method that is tested with fewer than 20 qubits and provides no quantum advantage, without estimating how to obtain one [2331–2339]. Or don't indicate the number of qubits and/or how the algorithm was tested [2340–2342].
- Putting aside classical data preparation and loading quantum circuits from the algorithm. It happens in many QML cases [2343].
- Not using dataset sizes corresponding to usual industry needs [2344] and not considering how often the solution would be used per day.
- Comparing the evolution between two quantum solutions and hiding the comparison with best-in-class classical solutions [2345–2347], or not using the best possible classical solution for the comparison [2348].
- Unfairly comparing classical exact solutions and quantum heuristics which happens with many D-Wave use cases [2349, 2350].
- Not providing resource estimates to obtain some real quantum advantage in either a NISQ, FTQC regime or analog regime [2351, 2352].
- Confusing pure academic use cases with industry-grade use cases [2240].
- Creating poorly captioned charts that obfuscate the comparison with classical solutions [2343].
- Extrapolating some NISQ case study to FTQC scale without considering the incremental cost of QEC and other artifacts (barren plateau, QEM scaling).
- Use overly optimistic hardware capabilities like qubit fidelities (99.99%) and gate speeds.
- Forgetting basic economic reasoning, like when comparing a \$2K or \$25K PC with a \$12M QPU [1142, 2329, 2353].
- **Classical emulation** (not shown in the graph) does not demonstrate any quantum advantage whatsoever if it is not complemented with resource and computing time estimations for a real quantum computer execution. It is also an indication, if this comes out of some hardware vendor, of the immaturity of their hardware platform.
- **Quantum inspired solutions.** These solutions may improve the state of the art of existing classical solutions, but they are not quantum per se. They benefit from advances in tensor network algorithms (MPS, DMRG) and from CPU/GPU and HPC hardware performance improvements.
- **Present Analog quantum computing solutions** which can run on existing analog systems like a D-Wave quantum annealer or a Pasqal/QuEra quantum simulator. At this point in time, these are the most powerful available solutions, but none have yet reached some quantum advantage.
- **Future Analog quantum computing solutions** which require a future analog computer with a larger number of qubits and/or better connectivity. These hardware offerings are presumed to show up in the short term of less than 5 years to support over 300 workable qubits.
- **Present NISQ solutions** which are being tested on an existing quantum computer and can usually be also tested in emulation mode on a classical computer. Their business value is usually minimal, and they usually showcase no quantum advantage in whatever category. Cautiousness is required.
- **Future NISQ solution for a future quantum computer** which may bring some quantum advantage. The related hardware may show up in the short term (IBM's Heron with 133 qubits with 99.9% fidelities) or in a longer term (higher fidelities and even more qubits).
- **FTQC solutions** with the capacity to support from a hundred to thousands of logical qubits and the related fidelities matching the algorithm breadth and depth. These QPUs may be available some day in the longer term, probably beyond 10 to 20 years. It is also important to have an idea of the required FTQC hardware with documented resource estimates. There's a big gap between having 100 and 4,000 logical qubits, particularly given the required physical resources are scaling nearly polynomially here to ensure adequate logical qubits fidelities.

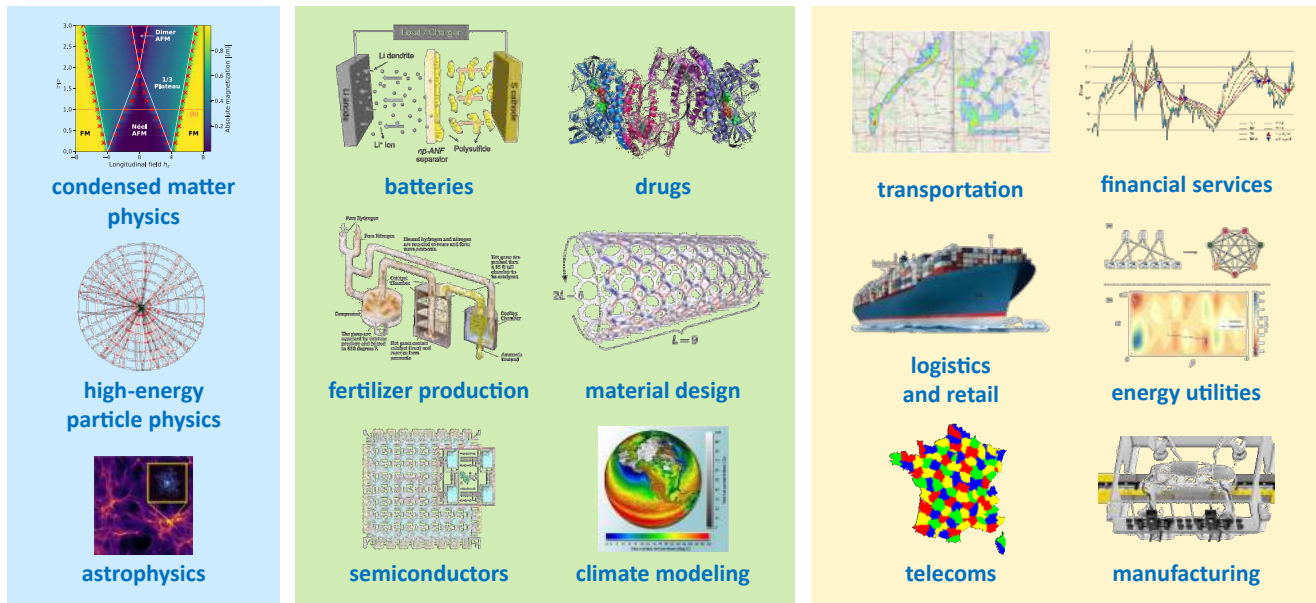


Figure 161: Map of the main solution domains for quantum computing applications separating research oriented use cases and operation level use cases. The horizontal arrows encode the author’s qualitative assessment and not a measurement: our confidence in finding use cases with some quantum advantage is higher on the research side of the map. Separately, and as a market hypothesis rather than a scientific claim, the larger market is probably with business operation applications, where it is quite hard to find algorithms that may bring some quantum advantage. © Olivier Ezratty, 2022-2025, with some image sources: [2357–2365].

3.1.7 Documentation

The last criterion for validating a case study is its documentation.

Is it provided only with a vendor or customer press release (examples: [2366] or [2367] which mentions a paper that tested some algorithm on IBM Hanoi’s QPU with 27 qubits) and website page (red flag), with an arXiv preprint (better) or with a peer-reviewed journal paper (much better, although not necessarily safe, read the paper conclusions carefully)? The more scientific details and data you have on the case study, the better. It may take time from the vendor/customer press release being published and some related scientific publication. So, patience will be a virtue here.

3.1.8 Example

We will use an example to assess a case study from Quantinuum (Figure 162). It was presented in The Quantum Insider in September 2023 as showing “*early stage quantum advantage*” [2370]. It is not clear what that means exactly. When looking at the corresponding arXiv paper, we clearly see that we are not yet in any form of quantum advantage (Figure 163). Indeed, the tests of this new multipurpose Monte Carlo engine were implemented with 6 physical qubits, far from any quantum advantage since it can be emulated on your smartphone and laptop, probably faster than on any quantum computer. The engine contains a resource

estimation tool which shows that an early quantum advantage would require at least 100 qubits with 99.99% two-qubit gates fidelities, far from what is available today.

3.1.9 Market

You could add a last parameter for the use case, its targeted market or domain. As shown in Figure 161, you can separate these use cases in two broad categories: research use cases and operations use cases. Research use cases correspond to fundamental academic research like in condensed matter, high-energy physics at CERN or astrophysics, and applied research is R&D work to design new materials, processes and products which, once industrialized, are running with classical physical resources.

These use cases will demand a smaller number of quantum computers than the operations categories which corresponds to uses cases running on a continuous basis and with potentially some scale and volume. Chemical simulation use cases will sit in the first category while most optimization problems solving will be in the second part.

All this leads to adding a few necessary items to the framework. Any use case should contain an identity card with a customer name, or customer segment, and optionally, an industry partner (quantum hardware or software) and optionally some academic research partner and finally, a date.

Quantinuum's Quantum Monte Carlo Integration Engine Shows Early Stage Quantum Advantage

Quantum Computing Business, Research Matt Swayne • September 13, 2023



A MODULAR ENGINE FOR QUANTUM MONTE CARLO INTEGRATION

ISMAIL YUNUS AKHALWAYA ^{1,2}, ADAM CONNOLLY ¹, ROLAND GUICHARD ^{1,†}, STEVEN HERBERT ^{1,2}, CAIHT KARGI ¹, ALEXANDRE KRAJENBRINK ¹, MICHAEL LUBASCH ², CONOR MC KEEVER ², JULIEN SORCI ¹, MICHAEL SPRANGER ¹, IFAN WILLIAMS ²

¹Quantinuum, Terrington House, 13-15 Hills Road, Cambridge CB2 1NL, United Kingdom

²Quantinuum, Partnership House, Carlisle Place, London SW1P 1BX, United Kingdom

[†]School of Computer Science and Applied Mathematics, University of the Witwatersrand, Johannesburg, South Africa

[‡]Department of Computer Science and Technology, University of Cambridge, United Kingdom

ABSTRACT. We present the Quantum Monte Carlo Integration (QMCI) engine developed by Quantinuum. It is a quantum computational tool for evaluating multi-dimensional integrals that arise in various fields of science and engineering such as finance. This white paper presents a detailed description of the architecture of the QMCI engine, including a variety of distribution-loading methods, a novel quantum amplitude estimation method that improves the statistical robustness of QMCI calculations, and a library of statistical quantities that can be estimated. The QMCI engine is designed with modularity in mind, allowing for the continuous development of new quantum algorithms tailored in particular to financial applications. Additionally, the engine features a resource mode, which provides a precise resource quantification for the quantum circuits generated. The paper also includes extensive benchmarks that showcase the engine's performance, with a focus on the evaluation of various financial instruments.

<https://arxiv.org/abs/2308.06081>

"The new white paper sets out the areas that stand to benefit from the development of QMCI, beyond finance, including achieving efficiencies in supply chain and logistics, energy production and transmission, and data-intensive fields of science such as solving the high-dimensional integrals in high-energy physics. **It concludes that use cases such as estimation and forecasting can benefit from the new QMCI engine in its current form**".

tested with 6 qubits!

"Accordingly, it is entirely reasonable to speculate that a future quantum computer with **~100 qubits and two-qubit gate fidelity ~99.99%** should be capable of running some simple, but **not trivial financial QMCI calculations**. However, whilst such a putative future quantum computer may be able to obtain an advantage in sample complexity for a non-trivial financial Monte Carlo integral – which would itself constitute a valuable outcome – **it is doubtful that it would make practical sense to price such an option on a quantum computer**, as we discuss in more detail in Section 12".

Figure 162: Quantinuum Quantum Monte Carlo Integration Engine as presented in a misleading way in The Quantum Daily, mentioning "early stage quantum advantage". When looking at the actual arXiv paper, there is no such early stage quantum advantage given it would require at least 100 qubits with 99.99% fidelities. Source: [2368].

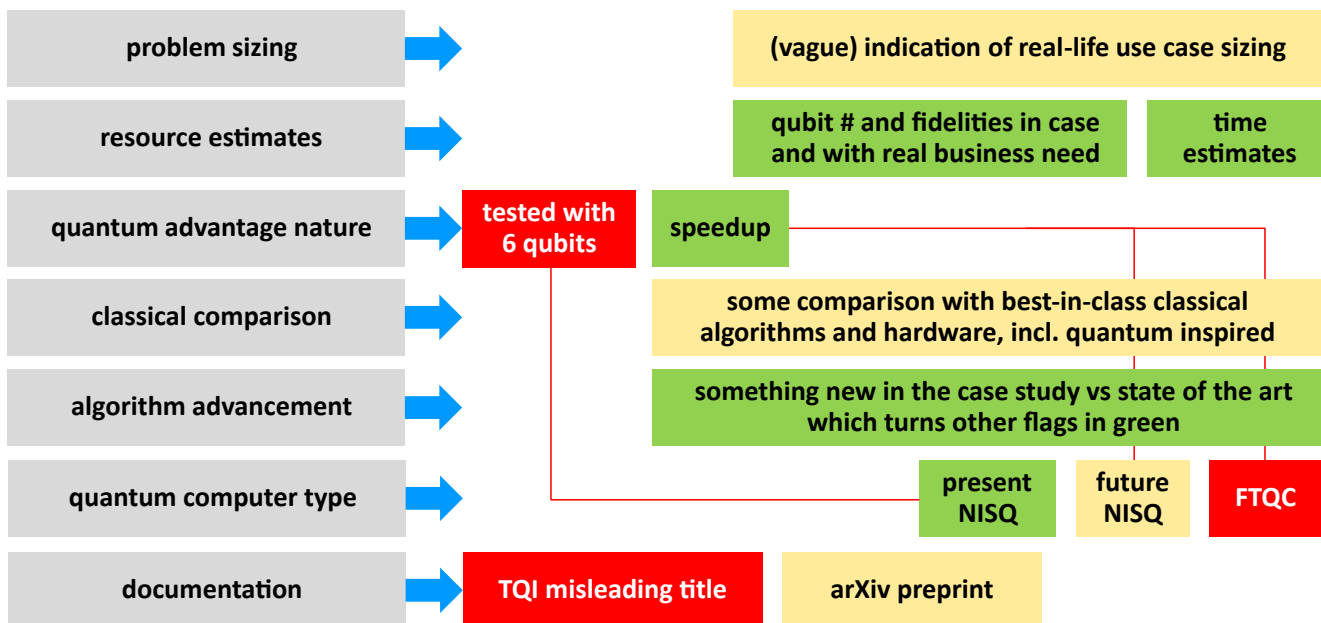


Figure 163: Assessing the Quantinuum QMCI engine paper using our case studies evaluation framework. All in all, this is a new engine that may provide some quantum advantage but in future NISQ and FTQC QPUs, not those that are available as of 2023. © Olivier Ezratty, 2023. It is much better than [2369] which is a 20-qubit use case, with no real quantum advantage.



3.1.10 Bibliography and notes

These are the bibliographical references and notes for the **Case studies and use cases evaluation** subsection.

- [263] [Forrelation: A Problem that Optimally Separates Quantum from Classical Computing](#), by Scott Aaronson and Andris Ambainis, arXiv, November 2014 (60 pages).
- [1142] [The Quest for Quantum Advantage in Combinatorial Optimization: End-to-end Benchmarking of Quantum Solvers vs. Multi-core Classical Solvers](#), by Pranav Chandarana, Alejandro Gomez Cadavid, Enrique Solano, Thorsten Koch, Stefan Woerner, and Narendra N. Hegade, arXiv, March 2026 (17 pages).
- [2127] [Milestones on the Quantum Utility Highway](#), by Catherine C. McGeoch and Pau Farre, arXiv, May 2023 (34 pages).
- [2220] [The vast world of quantum advantage](#), by Hsin-Yuan Huang, Soonwon Choi, Jarrod R. McClean, and John Preskill, Physical Review X, July 2026 (29 pages).
- [2221] [A Framework for Quantum Advantage](#), by Olivia Lanes, Mourad Beji, Antonio D. Corcoles, Constantin Dalyac, Jay M. Gambetta, Loic Henriet, Ali Javadi-Abhari, Abhinav Kandala, Antonio Mezzacapo, Christopher Porter, Sarah Sheldon, John Watrous, Christa Zoufal, Alexandre Dauphin, and Borja Peropadre, arXiv, June 2025 (9 pages).
- [2240] [Beyond-classical computation in quantum simulation](#), by Andrew D. King et al., Science, March 2025 (15 pages) 📄 (preprint on ruj.uj.edu.pl).
- [2323] [Biology and medicine in the landscape of quantum advantages](#), by Benjamin A. Cordier, Nicolas P. D. Sawaya, Gian Giacomo Guerreschi, and Shannon K. McWeene, Journal of The Royal Society Interface, November 2022 (31 pages).
- [2324] [Quantum Computing Use Case Compendium](#), by NQCC, NQCC, 2025 (27 pages).
- [2325] [Calculating the energy profile of an enzymatic reaction on a quantum computer](#), by Patrick Ettenhuber, Nikolaj Thomas Zinner, et al., arXiv, May 2025 (37 pages).
- [2326] [Model Cards for Quantum Technologies Reporting](#), by Mark J Everitt and Siyuan Ji, arXiv, December 2024 (7 pages).
- [2327] [Realizing Quantum Potential: Study of Quantum Computing Use Cases Developed Worldwide](#), by Michael Baczyk, May 2024.
- [2328] [Quantum Algorithm for Maximum Biclique Problem](#), by Xiaofan Li, Wolfgang Nejdl, et al., arXiv, September 2023 (14 pages).
- [2329] [Accelerating large-scale linear algebra using variational quantum imaginary time evolution](#), by Willie Aboumrاد, Martin Roetteler, et al., arXiv, April 2025 (14 pages).
- [2330] [Quantum-enhanced satellite image classification](#), by Qi Zhang, Anton Simen, Carlos Flores-Garrigós, Gabriel Alvarado Barrios, Paolo A. Erdman, Enrique Solano, Aaron C. Kemp, Vincent Beltrani, Vedangi Pathak, and Hamed Mohammadbagherpoor, arXiv, February 2026 (7 pages).
- [2331] [Quantum-Classical Computational Molecular Design of Deuterated High-Efficiency OLED Emitters](#), by Qi Gao et al., June 2023 (10 pages).
- [2332] [A variational quantum eigensolver tailored to multi-band tight-binding simulations of electronic structures](#), by Dongkeun Lee and Hoon Ryu, arXiv, May 2025 (12 pages).
- [2333] [Identifying Protein Co-regulatory Network Logic by Solving B-SAT Problems through Gate-based Quantum Computing](#), by Aspen Erlandsson Brisebois, Gordon Broderick, et al., arXiv, April 2025 (9 pages).
- [2334] [Compare Similarities Between DNA Sequences Using Permutation-Invariant Quantum Kernel](#), by Chenyu Shi, Gabriele Leoni, Mauro Petrillo, Antonio Puertas Gallardo, and Hao Wang, arXiv, March 2025 (11 pages).
- [2335] [Quantum Machine Learning Applied to the Sinking of the Titanic](#), by Luiz Henrique Prudencio dos Santos, Roberto Baginski Batista Santos, et al., arXiv, August 2025 (23 pages).
- [2336] [A brain-inspired paradigm for scalable quantum vision](#), by Chenghua Duan, Xia Wu, et al., arXiv, September 2025 (12 pages).
- [2337] [Quantum Computing Based Design of Multivariate Porous Materials](#), by Shinyoung Kang, Younghun Kim, and Jihan Kim, ACS Central Science, August 2025 (12 pages).
- [2338] [HIV-1 protease cleavage sites detection with a Quantum convolutional neural network algorithm](#), by Junggu Choi, Junho Lee, Kyle L. Jung, and Jae U. Jung, arXiv, October 2025 (45 pages).
- [2339] [Quantum simulation of actinide chemistry: towards scalable algorithms on trapped ion quantum computers](#), by Kesha Sorathia, David Munoz Ramo, et al., arXiv, October 2025 (40 pages).
- [2340] [Quantum maximum power point tracking \(QMPPT\) for optimal solar energy extraction](#), by Habib Feraoun et al., December 2024 (17 pages).
- [2341] [Can a Quantum Computer Simulate Nuclear Magnetic Resonance Spectra Better than a Classical One?](#), by Keith R. Fratus, Nicklas Enenkel, Sebastian Zanker, Jan-Michael Reiner, Michael Marthaler, and Peter Schmitteckert, arXiv, August 2025 (11 pages).
- [2342] [A joint optimization approach of parameterized quantum circuits with a tensor network](#), by Clara Ferreira Cores, Kaur Kristjuhan, and Mark Nicholas Jones, arXiv, February 2024 (29 pages).
- [2343] [Quantum Probabilistic Label Refining: Enhancing Label Quality for Robust Image Classification](#), by Fang Qi, Lu Peng, and Zhengming Ding, arXiv, October 2025 (11 pages).
- [2344] [Solving Currency Arbitrage Problems using D-Wave Advantage2 Quantum Annealer](#), by Lorenzo Mazzei, Giada Bec-cari, Mirko Laruina, and Marco Cococcioni, arXiv, September 2025 (6 pages).
- [2345] [Benchmarking Quantum Solvers in Noisy Digital Simulations for Financial Portfolio Optimization](#), by Ruizhe Shen, Zichang Hao, and Ching Hua Lee, arXiv, August 2025 (17 pages).
- [2346] [Demonstration of a Compatibility-Based Childcare Support Service using Quantum Annealing](#), by Yuuma Matsumoto, Taisei Takabayashi, Rima Sato, Rumiko Honda, and Masayuki Ohzeki, arXiv, September 2025 (17 pages).
- [2347] [PACE-QAOA: Physics-Constrained Quantum Optimization for Qubit-Efficient Power System Islanding](#), by Yuqi Jiang, Zhiding Liang, Qiang Guan, Yan Li, and Ganesh Kumar Venayagamoorthy, arXiv, August 2026 (43 pages).
- [2348] [Quantum optical reservoir computing powered by boson sampling](#), by Akitada Sakurai, Aoi Hayashi, William John Munro, and Kae Nemoto, Optica Quantum, May 2025 (8 pages).
- [2349] [Decomposition-Based QAOA for Maximum Coverage Location Problem in Satellite Constellation Design](#), by Divya Sisodiya, Amiratabak Bahengam, Hang Woon Lee, and Hao Chen, arXiv, July 2026 (21 pages).
- [2350] [Quantum Annealing-Based Algorithm for Efficient Coalition Formation Among LEO Satellites](#), by Supreeth Mysore Venkatesh, Antonio Macaluso, Marlon Nuske, Matthias Klusch, and Andreas Dengel, arXiv, August 2024 (6 pages).
- [2351] [Hybrid Quantum-Classical Scheduling with Problem-Aware Calibration on a Quantum Annealer](#), by Krzysztof Giergiel, Y. Sam Yang, and Anthony B. Murphy, arXiv, September 2025 (10 pages).
- [2352] [How Can Quantum Deep Learning Improve Large Language Models?](#), by Emily Jimin Roh, Hyojun Ahn, Samuel Yen-Chi Chen, Soohyun Park, and Joongheon Kim, arXiv, September 2025 (5 pages).
- [2353] [Combinatorial optimization enhanced by shallow quantum circuits with 104 superconducting qubits](#), by Xuhao Zhu,

- Man-Hong Yung, H. Wang, et al., arXiv, September 2025 (29 pages).
- [2354] [The Quantum Tortoise and the Classical Hare: A simple framework for understanding which problems quantum computing will accelerate \(and which it will not\)](#), by William S. Moses Sukwoong Choi and Neil Thompson, arXiv, October 2023 (14 pages).
 - [2355] [What is a good use case for quantum computers?](#), by Michael Marthaler, Peter Pinski, Pascal Stadler, Vladimir Rybkin, and Marina Walt, arXiv, June 2025 (23 pages).
 - [2356] [How NOT to Fool the Masses When Giving Performance Results for Quantum Computers](#), by Catherine McGeoch, arXiv, November 2024 (14 pages).
 - [2357] [Quantum computing for data analysis in high energy physics](#), by Andrea Delgado, Jorge J. Martinez de Lejarza, et al., arXiv, December 2022 (23 pages).
 - [2358] [Quantum computing enables simulations to unravel mysteries of magnetic materials](#), by Elizabeth Rosenthal, February 2021.
 - [2359] [Towards Cosmological Simulations of Dark Matter on Quantum Computers](#), by Philip Mocz and Aaron Szasz, arXiv, February 2021 (10 pages).
 - [2360] [Multifactorial engineering of biomimetic membranes for batteries with multiple high-performance parameters](#), by Mingqiang Wang, Nicholas A. Kotov, et al., Nature Communications, January 2022.
 - [2361] [Semi-supervised Learning Predicts Approximately One Third of the Alternative Splicing Isoforms as Functional Proteins](#), by Yanqi Hao et al., Cell Reports, July 2015 (8 pages).
 - [2362] [Scaling advantage in quantum simulation of geometrically frustrated magnets](#), by Andrew D. King, Mohammad H. Amin, et al., arXiv, November 2019 (7 pages).
 - [2363] [Comparative Analysis and Enhancement of FPGA and ASIC Performance in Targeted Applications](#), by Haozhe Lou, SpringerLink, February 2025 (13 pages) .
 - [2364] [Quantum computing with neutral atoms](#), by Loic Henriet, Lucas Beguin, Adrien Signoles, Thierry Lahaye, Antoine Browaeys, Georges-Olivier Reymond, and Christophe Jurczak, arXiv, June 2020 (41 pages).
 - [2365] [Traffic flow optimization using a quantum annealer](#), by Florian Neukart, Gabriele Compostella, Christian Seidel, David von Dollen, Sheir Yarkoni, and Bob Parney, arXiv, August 2017 (17 pages).
 - [2366] [Japan Tobacco Inc. and D-Wave Announce Collaboration Aimed at Accelerating Innovative Drug Discovery with Quantum AI](#), by D-Wave, D-Wave News, October 2024.
 - [2367] [Ecosystem Innovation: IonQ's Life Sciences Application Workflow Accelerates the Drug Development Process](#), by IonQ staff, IonQ news, June 2025.
 - [2368] [A Modular Engine for Quantum Monte Carlo Integration](#), by Ismail Yunus Akhalwaya, Ifan Williams, et al., arXiv, August 2023 (87 pages).
 - [2369] [Characterizing a non-equilibrium phase transition on a quantum computer](#), by Eli Chertkov, Michael Foss-Feig, et al., arXiv, November 2022 (7 pages).
 - [2370] [Quantinuum's Quantum Monte Carlo Integration Engine Shows Early Stage Quantum Advantage](#), by Matt Swayne, The Quantum Insider, September 2023.

This **Case studies and use cases evaluation** subsection contains 54 bibliographical references and notes containing at least 969 pages.

3.2 Market forecasts

Any new technology wave brings its market forecasts data born out of analysts and market survey companies. They have a very traditional closed-loop system in place: vendors want to get some ideas of customer demand or positive confirmation of their own biases, analysts poll large customers to get some understanding about their plans, and *voilà*, you get your nice market predictions. It often looks like linear or simple nonlinear regressions. These predictions can become either self-fulfilling prophecies or total failures. The Gartner Group has turned its simplistic hype curve into a kind of Schrödinger's time and topic-independent wave equation of technology trends. But nobody really checked it, particularly when this curve was highly dependent on complicated scientific and technological challenges. It is more about probabilities than simplistic curves.

3.2.1 Quantitative assessments

So, how could we predict the size and shape of the quantum hardware and software markets, vertical per vertical, when you have no idea of when actual useful quantum computers will show up? Will it follow an exponential market growth rate worthy of those of the microcomputer and smartphones industries? Let's look at what we have in store. All the figures quoted below are nominal dollars as published, with no currency-year normalization, and each should be read with its forecast vintage (the year the forecast was issued), its target year, its scope (hardware only, hardware plus software and services, or full quantum technologies), its geography and, most importantly, whether it describes a vendor market or a downstream value creation, two quantities that differ by an order of magnitude.

BCG's quantum computing growth forecasts illustrate this strong uncertainty. They showcase predictions with an optimistic scenario, which starts seeing growth around 2030, and a very conservative one, which only takes off after 2040 [2371]. In both cases, the quantum computing market grows linearly.

Predicting the size of the quantum computing market is indeed highly probabilistic. It was supposed to reach \$553M in 2023 (in 2017) and \$4.4B by 2028 (in 2023 [2372]) according to **Markets and Markets**, then \$1.9B in 2023 for CIR and \$2.64B in 2022 for **Market Research Future** (in 2018). A **Global Market Insights** report from April 2023 estimated that the quantum computing market size was of \$15B in 2022 (probably in some parallel fringe world) and would reach \$65B by 2032 [2373]. This survey was sold for \$5,150!

In 2021, **Hyperion Research**, a company created by former analysts from the Gartner Group, forecasted a \$830M market for 2024 [2374, 2375], and, in 2022, upgraded it to \$964M for the same year, close to the observed outcome, and to \$1.2B in 2028 [2376] without mentioning the usefulness of these systems. At the

end of 2022 at the Q2B conference in Santa Clara, Bob Sorensen from Hyperion said wisely that it was too early to make long-term projections beyond 2025 [2377]. It doesn't prevent some analysts from publishing quantum computing forecasts per vertical market.

GRD Survey predicted in April 2023 that the health-care quantum computing market could reach \$25B by 2032 and Markets&Markets did a similar prediction in September 2023 of \$5.2B for the automotive industry by 2035.

Then we reached \$8.45B in 2024 for **Homeland Security** (in 2018, and we were around \$1B for 2024), \$10B in 2028 for **Morgan Stanley** (as of 2017), \$15B by 2028 for **ABI Research** (2018) and \$64B by 2030 for **P&S Intelligence** (in 2020). **ResearchAndMarkets** predicted in May 2021 that the global quantum technology market would even reach \$31.57B by 2026, including \$14.25B for quantum computing [2378] (this didn't happen, we are way below \$2B). **IDC** planned for a 6-year compound annual growth rate (CAGR) of 50.9% over the 2021-2027 period with a market reaching \$8.6B in 2027 [2379]. In August 2023, it reduced its forecast from \$1.1B in 2022 to \$7.6B in 2027 with a 5Y CAGR (Compound Annual Growth Rate) of 48.1% [2380]. Why such a drop? Their estimates were negatively impacted by slower than expected advances in quantum hardware development, generative AI and macroeconomic factors. But \$7.6B may still be overly optimistic at the speed of progress in the domain.

The Quantum Insider has its own predictions with a total quantum computing market between \$300M and \$1.3B in 2021 that could grow to between \$3.5B and \$10B by 2025 (it didn't happen) and between \$18B and \$65B by 2030 with a CAGR of between 70% and 80% from 2021 to 2025, to slow down between 39% to 45% between 2025 to 2030 [2382].

Some forecasts can reach other crazy heights. For **Bank of America**, quantum technologies will be as important as smartphones. The main reason? Its potential applications in healthcare. To make sure, the point is made, Haim Israel from this bank also touted that quantum computing will be more important than the invention of fire, which is a stretch and hyperbolic [2383].

The only problem: many analyses behind these predictions become confused between big data and quantum computing [2384]. Some are based on vendors' expectations, others on customers' uncertain plans to adopt quantum computing, given nobody has a real clue of when and how it will work.

As of early 2020, **McKinsey** even predicted that quantum computing would be worth \$1T by 2035 [2385]. It is easy to identify the forecast bias used here. It is based on a trick that was used a few years ago to evaluate the size of Internet of things and artificial intelligence markets. It is not the market estimation for quantum technologies as such, but the incremental revenue it could generate for businesses, such as in pharmaceu-

Exhibit 2 - BCG's 2021 Forecast by Use Case Category

1 Quantum-advantaged mathematical function	Sparse matrix math			
	Simulation	Optimization	Machine learning	Cryptography
4 Computational problem types	 (\$175 billion–\$330 billion)	 (\$100 billion–\$220 billion)	 (\$95 billion–\$250 billion)	 (\$40 billion–\$80 billion)
100+ High-value industry use cases (Sizing at tech maturity)	Pharma: Drug discovery \$40 billion–\$80 billion	Finance: Portfolio optimization \$20 billion–\$50 billion	Automotive: AV AI algorithms Up to \$10 billion	Government: Encryption and decryption \$20 billion–\$40 billion
	Aerospace: CFD \$10 billion–\$20 billion	Insurance: Risk management \$10 billion–\$20 billion	Finance: AML and anti-fraud \$20 billion–\$30 billion	
	Chemistry: Catalyst design \$20 billion–\$50 billion	Logistics: Network optimization \$50 billion–\$100 billion	Tech: Search/advertising optimization \$50 billion–\$100 billion	Corporate: Encryption and decryption \$20 billion–\$40 billion
	Energy: Solar conversion \$10 billion–\$30 billion	Aerospace: Route optimization \$20 billion–\$50 billion	Other use cases \$25 billion–\$110 billion	
	Finance: Market simulation (e.g., derivatives pricing) \$20 billion–\$35 billion			
	Other use cases \$75 billion–\$115 billion			

Sources: Industry interviews; BCG analysis.

Note: AML = anti-money laundering; AV AI = audiovisual artificial intelligence; CFD = computational fluid dynamics.

Figure 164: BCG 2021 forecast of quantum computing potential value by use case category and vertical market. Sources are industry interviews and BCG analysis. AML=anti-money laundering, AV, AI=audiovisual artificial intelligence and CFD=computational fluid dynamics. Historical forecast, issued in 2021 and confirmed by BCG in July 2024. It describes downstream value creation, not a vendor market, and no observed market revenue has validated it so far. Source: [2381].

ticals, financial services, or transportation. It is a bit like evaluating the software market (which was around \$593B in 2021, including \$237B in enterprise software, source [Statista](#)) by summing up the total revenue of the companies that use some software! This would be quite a large number and a significant share of worldwide GDP. In 2022, they reused the same methodology to forecast that the Metaverse would create \$5T of value by 2030 [2386]. It mentioned “\$120B invested in 2022” but this included M&As like the \$69B acquisition of Activision by Microsoft and only \$6B to \$8B real investments in startups through venture capital. It failed miserably in about a year after the Metaverse was entirely overshadowed by LLMs (large language models à la ChatGPT). In 2023, McKinsey published an updated version of its market estimates with the same bias [2387]. They forecasted a quantum computing market size of \$9B to \$93B in 2040, not far from what a random number generator would create. They upgraded their “value creation” to \$1.3T in 2035.

In its 2024 iteration of its Quantum Technology Monitor, McKinsey updated again its potential economic value prediction coming from quantum computing by 2035 from \$900B to \$2T. The report was plagued with many errors and misinterpretations of scientific, technological and economical news. For example, it mentioned an

OQC funding of \$100M that was not closed, the famous China’s misleading \$15B investment mark, and Germany double booking with \$5B of public investments when the reality is below \$3B. They also did show a bad understanding of logical qubits advancements (slide 32) confusing demonstrations, blueprints and theoretical papers. On top of that, the report did not mention algorithms and speedups, qRAM and data loading challenges. In the best case scenario, we may then have around 100 logical qubits at this moment. Most case studies show that strong value requires thousands of logical qubits, particularly in quantum chemistry. Also, creating digital twins of whatever chemical process cannot lead to instant business value creation. Any new product or industrial process must be launched, which takes time. In drug design, you will still need to do clinical tests which can last up to 10 years. A successful design pre-clinical test may contribute to growing the pharmaceutical company stock value by 2035, but definitely not its revenue or bottom line! So, any trillion-dollar range value creation may need two instead of one decade to show up. It is just common sense!

In their 2026 Quantum Technology Monitor, McKinsey sustained its forecasts [2388]. They updated their value creation, expected to be between \$1.3T and \$2.7T by

2035, and a market size in the \$25B-\$34B range in 2030, and \$43B-\$71B range in 2035.

As if BCG and McKinsey were not enough, The Quantum Insider published its own prediction for an economic impact of quantum computing of \$1T by 2035 [2389]. They forecasted in 2024 that quantum computing vendors would generate \$50B in revenue and create 840,000 new jobs by 2035, with 250,000 jobs by 2030.

Market predictions should focus on IT products, software and services and should be compared with existing reference markets. For example, the 2020 worldwide servers market size was \$85.7B according to IDC market estimates: Q1, Q2, Q3 and Q4 with respectively \$18.6B, \$18.7B, \$22.6B and \$25.8B. The HPC market size was estimated to be \$43.6B in 2023 by GlobalData, with an expected future compound growth rate of 7.5% [2390]. Expecting that at some point, the quantum computing market would exceed one of these two markets is preposterous, even considering the discrepancies between hardware and hardware plus software and services markets accounting.

The market value fallacy has a side effect in the quantum hype. Some vendors play on the confusion when they are sizing their market when talking to investors. The fallacy propagates easily like in the innocuous paper on EDAs [2391] that estimates that the quantum computing market could reach \$700B as early as 2035, showing a clear confusion between a market and value creation.

In a 2021 publication [2392], BCG estimated the size of the quantum computing market as 20% of its estimated generated value with customers of \$850B, ending with a \$90B to \$170B market captured by technology providers, including software and services... some day after 2040, and a more reasonable \$1B to \$2B before 2030 and \$15B to \$30B after 2030. So, we have here an uncertainty based on an unknown estimated with some fuzzy technology capability predictions. In August 2025, The Quantum Insider estimated the value capture share of QPU vendors at a more modest 6% [2393]. They explained this by a value migration shift to applications creation and a lot of mutualization of the hardware using it.

The problem is some vendors take these data for a market size in their investor pitch presentations, not as a generated value size [2394, 2395]. BCG updated its views and forecasts in July 2024 [2381], being more cautious on the potential value creation in the NISQ era and highlighting the role of public procurement to sustain the quantum computing ecosystem. Still, they maintained their value creation forecasts from 2021 (Figure 164).

Market forecasts are even more troubling when they focus on a given use case category, vertical market and country, and mixing all these things. Each additional segmentation criterion narrows the sample the estimate rests on and multiplies correlated sources of uncertainty.

Scope sensitivity and error propagation then grow faster than the apparent precision, so the more granular the forecast, the wider its real confidence interval. One example is a market prediction for quantum machine learning published by Inside Quantum Technology in December 2024 [2396]. It predicted that revenues from neural-network quantum machine learning would reach \$1.1B in 2030 and \$7B in 2034.

3.2.2 Qualitative assessments

BCG created in 2018 a good inventory of the current potential qualitative use cases of quantum computing per vertical market, in Figure 165.

This covers both case studies coming from D-Wave and prospective applications devised by research labs and with large industry companies including the usual suspects from the aerospace, chemistry, energy, pharmaceuticals and financial sectors. In another chart, in Figure 166, BCG positions these vertical markets along two dimensions: business value and expected time of quantum advantage. This is more gut feeling than rigorous rationale since there are too many variables to have any idea of where each of these industries sit in this fancy chart. We are currently at a too early stage of the quantum computing innovation cycle to make such predictions.

3.2.3 Market variables

Since we have no real idea of when scalable quantum computing will really work, let's try another exercise to determine the critical factors enabling some sort of technology commoditization for quantum computing:

Technology. The first factor is a mix of where and when we'll have first, useful quantum annealers and analog quantum computers, then useful NISQ systems, and then scalable **universal quantum computers** with more than a hundred logical qubits.

Genericity and platform effects. Most successful digital platforms from the last 70 years were market successes thanks to their genericity and platform effects. The genericity comes from the ability to be used in a large number of situations and industries. It was the case with the IBM 360 mainframes which covered well the needs of business and scientific computing when IBM's competitors were focusing on one of these markets (Burroughs and NCR in business applications and Control Data and Digital Equipment on scientific computing). The PC market was a generic one while Unix workstations finally disappeared, as high-end PC running either Windows or Linux took over. And on and on. The platform effect comes from the creation of a critical mass of solutions in multiple markets that support one hardware and software platform. It comes with the platform genericity and versatility first, and with the ability of its promoter to attract developers, software vendors



Figure 165: Quantum computing use-case scenarios per vertical. Historical chart, 2018 vintage. Source: BCG in 2018, [2397].

and other stakeholders, to develop applications on the platform. Generating a platform effect is much more powerful than finding a so-called “killer application”. Most successful business and consumer digital platforms like the web, your laptop or smartphones, have multiple killer applications, not a single one.

Software Tools. Another factor may be the consolidation of software development tools. These tools will continue to mature, and may raise their level of abstraction, and adapt to hardware evolutions. Libraries adapted to the needs of specific markets will consolidate, as in molecular simulation or finance. As the market matures, there will be some consolidation in this market.

Skills. One critical path to market growth as it has been the case for most previous major technology wave will be the availability of skilled workforce, particularly with developers. They will have, at least at the beginning, to handle abstractions levels that have nothing to do with the different forms of programming techniques that dominate today’s computing, even in its event-driven programming variants that are common in

the creation of websites and graphic applications. It is more an extension of the existing scientific computing community. A new generation of algorithm designers and developers will emerge. These will probably be young professionals who will have been able to digest new quantum computing concepts with a clean state mind.

Startups. The market will rely mostly on the fabric of startups, probably slightly ahead of traditional software publishers and IT services companies that may not necessarily venture first into this new world of quantum.

Experience. The first feedback from pilot projects, already underway, particularly with D-Wave, will be important. Most recent projects bring interesting learnings on the actual accelerations that quantum computing can provide. We will have to learn to make objective comparisons between quantum algorithms, quantum hardware architectures and their equivalents running (or not) on supercomputers. It will also be necessary to sort out into “proof of concepts” and projects actually deployed.

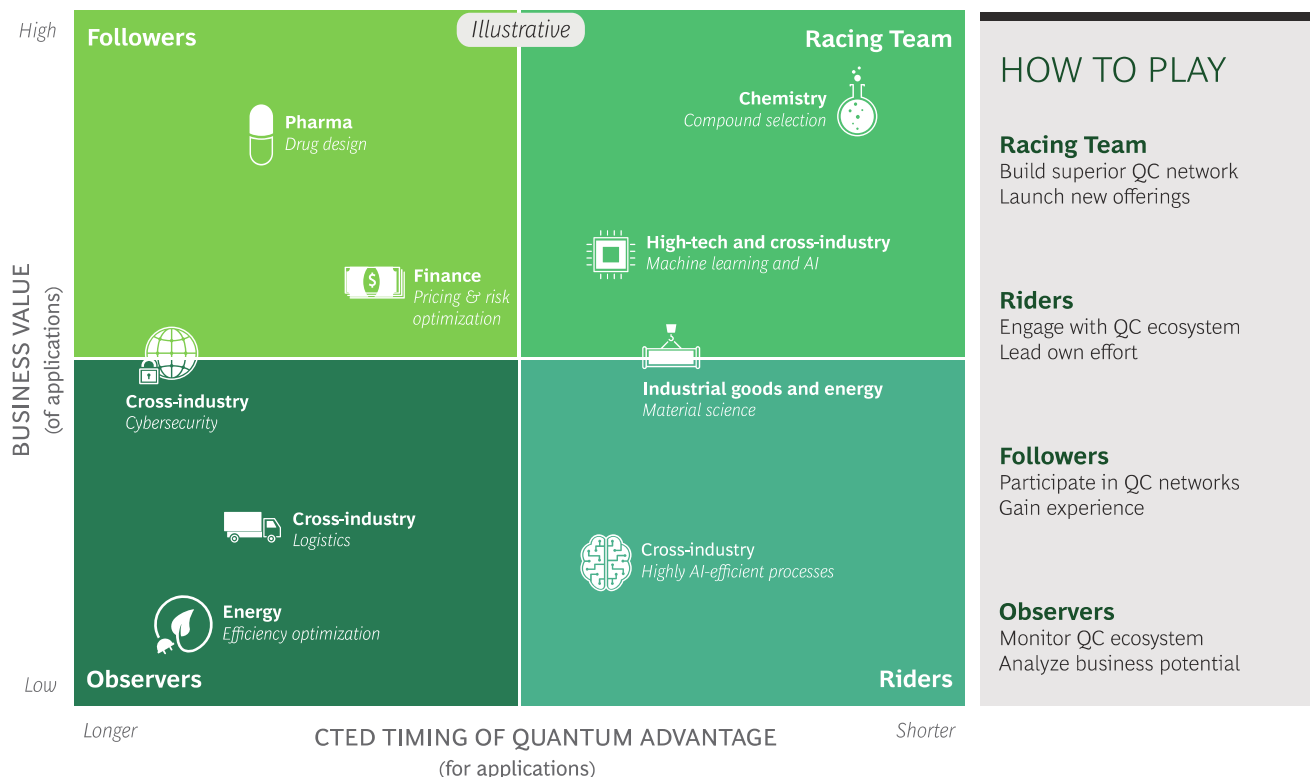
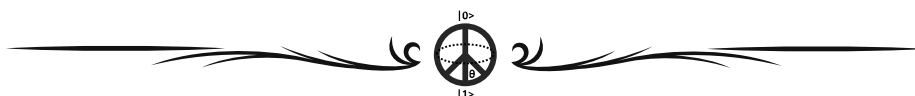


Figure 166: Correlation between use cases business value and expected timing for a quantum advantage. Historical chart, 2018 vintage. Its broad ordering is still consistent with what has been observed since, but no quantitative validation method has been applied to it. Source: [The Next Decade in Quantum Computing and How to Play](#) by Philippe Gerbert and Frank Ruess, BCG, 2018 (30 pages).

Mass impact. Finally, quantum computing commoditization will depend on the potential emergence of solutions that will have an impact on our daily lives. So, mostly consumer applications. It could come from

healthcare and transportation. Uses cases may shift gradually from the research community to the corporate world, and then to consumer applications.



3.2.4 Bibliography and notes

These are the bibliographical references and notes for the **Market forecasts** subsection.

- [2371] [The coming quantum leap in computing](#), by Anant Thaker and Suhare Adam, BCG, May 2018 (19 pages).
- [2372] [Quantum Computing Market Size, Share and Industry Growth Analysis Report by Offering, Deployment \(on-Premises and Cloud\), Application \(Optimization, Simulation, Machine Learning\), Technology \(Trapped Ions, Quantum Annealing, Superconducting Qubits\), End User and Region- Global Growth Driver and Industry Forecast to 2028](#), by Market and Markets, February 2023.
- [2373] [Global Market Insights](#), April 2023.
- [2374] [Quantum Computer Market Headed to \\$830M in 2024](#), by John Russell, September 2021.
- [2375] [New Study Estimates More Than 20 Percent Annual Growth of Global Quantum Computing Marketplace Through 2024](#), by Hyperion Research, February 2022.
- [2376] [Hyperion: QC Market Headed to \\$1.2B in 2025; Lots of Momentum but also Uncertainty](#), by John Russell et al., December 2022.
- [2377] [Global Quantum Computing Market Size and Forecast](#), by Bob Sorensen, December 2022.
- [2378] [The Worldwide Quantum Technology Industry will Reach \\$31.57 Billion by 2026 - North America to be the Biggest Region](#), by Research and Markets, May 2021.
- [2379] [Quantum computing market landscape](#), by TBR Research, October 2019.
- [2380] [IDC Forecasts Worldwide Quantum Computing Market to Grow to \\$7.6 Billion in 2027](#), by IDC, August 2023.
- [2381] [The Long-Term Forecast for Quantum Computing Still Looks Bright](#), by Jean-François Bobier, July 2024.
- [2382] [Quantum Computing Market Expects Double-Digit Growth \[2022\]](#), by Matt Swayne, The Quantum Insider, December 2021.
- [2383] [Quantum Computing Will Be Bigger Than the Discovery of Fire!](#), by Luke Lango, August 2022.
- [2384] [Quantum computing will be the smartphone of the 2020s, says Bank of America strategist](#), by Chris Matthews, December 2019.
- [2385] [McKinsey Forecasts Quantum Computing Market Could Reach \\$1 trillion by 2035](#), by McKinsey, April 2020.
- [2386] [On the road to change Value creation in the metaverse](#), by McKinsey, June 2022 (77 slides).

- [2387] [Quantum Technology Monitor April 2023](#), by McKinsey Quantum-Black team, McKinsey, April 2023.
- [2388] [Quantum Technology Monitor](#), by Henning Soller et al., McKinsey, April 2026 (103 slides).
- [2389] [The Quantum Insider Projects \\$1 Trillion in Economic Impact From Quantum Computing by 2035](#), by Resonance, The Quantum Insider, September 2024.
- [2390] [High Performance Computing \(HPC\) Market Size, Share and Trend Analysis by Region, Component \(Server, Storage, Network, Software, Services, Cloud\), Deployment, Application and Segment Forecast to 2026](#), by GlobalData, February 2023.
- [2391] [EDA Tools For Quantum Chips](#), by Marie C. Baca, SemiEngineering, November 2022.
- [2392] [What Happens When ‘If’ Turns to ‘When’ in Quantum Computing?](#), by Jean-François Bobier, Matt Langione, Edward Tao, and Antoine Gourévitch, July 2021.
- [2393] [Where’s the Real Money in Quantum? Quantum Vendors to Capture Just 6% of Total Projected Impact](#), by Cierra Choucair, The Quantum Insider, August 2025.
- [2394] [Q2 Rigetti Quarterly report](#), by Rigetti, August 2022.
- [2395] [Atlantic Quantum Emerges from MIT’s Engineering Quantum Systems Lab, Raises \\$9M Seed Funding to Make Large-Scale Quantum Computing a Reality](#), by James Dargan, The Quantum Insider, July 2022.
- [2396] [Machine Learning and Deep Learning in the Quantum Era 2024: A Market Forecast and Technology Assessment - Inside Quantum Technology](#), by Inside Quantum Technology, Inside Quantum Technology, December 2024.
- [2397] [The Next Decade in Quantum Computing and How to Play](#), by Philippe Gerbert and Frank Ruess, Boston Consulting Group, 2018 (30 pages).

This **Market forecasts** subsection contains 27 bibliographical references and notes containing at least 49 pages.

3.3 Chemistry and material design

Quantum computing could help improve the energy efficiency of some chemical engineering processes, including ammonia and fertilizer production. It could also contribute to battery chemistry, carbon-capture materials and the design of new materials by addressing electronic-structure, dynamics and optimization problems in molecules, solids and crystalline structures [2398].

Several of these applications have been prototyped at small scale but are not yet production-grade. The associated research is nevertheless useful to estimate the logical-qubit, gate-count and runtime requirements of relevant problems [2399, 2400].

Quantum algorithms have also been proposed for fluid dynamics, including air, water and other fluid flows and their turbulence, through discretizations of equations such as Navier-Stokes [2401]. Potential applications include aerospace and the optimization of turbines for energy production. These proposals generally inherit important caveats from quantum linear-system and state-readout algorithms, so an end-to-end advantage cannot be inferred from the complexity of the internal linear-algebra step alone.

Open-system methods are important when a molecule or material interacts non-negligibly with a solvent, phonon bath, electromagnetic environment or other degrees of freedom [2402]. The usual modeling step separates the degrees of freedom into a system and an environment or bath. In the weak-coupling Born-Markov regime, the bath correlation time is much shorter than the characteristic system relaxation time, so environmental memory can be neglected. Non-Markovian descriptions are required when memory effects remain relevant on the system timescale. Such open quantum-system models are important for chemical reactions in solvents, biology and materials science [2403, 2404].

3.3.1 Quantum chemistry 101

Quantum chemistry studies the electronic structure, properties and reactions of atoms, molecules and condensed-matter systems from quantum mechanics, including strongly correlated systems. It relies primarily on the Schrödinger equation and approximations derived from it, and in some regimes on relativistic Hamiltonians derived from the Dirac equation. It is an exceptionally broad field that I summarize here with selected bibliographical references.

Domains

Quantum chemistry spans several levels of description and several classes of problems:

Exploring electronic structure and bonding by characterizing chemical bonds, molecular orbitals, electron-density distributions and electronic energy lev-

els. Even a small molecule such as H_2O illustrates how orbital structure determines bonding and geometry.

Understanding molecular structure and properties by predicting molecular geometries and their electronic, vibrational, rotational and spectroscopic properties. Examples include the bent geometry of H_2O and the planar aromatic structure of benzene.

Determining energy levels and reactivity by computing electronic, vibrational and rotational energy levels, chemical reactivity, reaction rates and mechanisms. The total molecular energy includes electronic and nuclear contributions, and the vibrational and rotational spectra are quantized.

Modeling and optimizing chemical reactions by studying reaction thermodynamics, kinetics, pathways and catalysis. A typical example is understanding biological nitrogen fixation involving the FeMoCo complex. A realistic reaction model is not only a zero-temperature ground-state calculation. As emphasized by Chan, experimentally relevant observables can require free energies that include solvent and entropic effects [2405]. QPE-based electronic ground-state algorithms usually target eigenenergies of a fixed Hamiltonian, whereas finite-temperature and free-energy calculations require additional sampling or thermal-state machinery. Quantum algorithms for free energies have been proposed, using for example Hamiltonian block encoding and QSVT variants, with potentially large asymptotic speedups under their stated input and conditioning assumptions [2406–2408].

Simulating solute-solvent interactions is important in many inorganic and organic chemistry domains. It helps understand how molecules and ions behave in solvents such as water [2409].

Designing and optimizing new materials by modeling electronic structure and correlations in materials with desired properties. Examples include metals, semiconductors, superconducting materials, magnetic materials and catalysts [2410].

Simulating complex systems such as proteins, other biological macromolecules and their reactions, as discussed later in the healthcare and drug-design section.

Simulating out-of-equilibrium quantum matter, including driven and transient regimes relevant to superconducting and other correlated materials [2411].

While many quantum algorithms for chemistry focus on molecular ground-state energies, quantum chemistry also studies electronic structure, vibrational and rotational spectra, molecular geometry, excited and out-of-equilibrium states and molecular dynamics [2412, 2413]. Computational cost generally grows rapidly with the number of basis functions or active orbitals. Entanglement is not a simple ground-state versus excited-state distinction. Ground states of strongly correlated systems can themselves be highly entangled, and excited-state calculations are often additionally complicated by dense

spectra, near-degeneracies and multireference character [2414].

Industrial users do not necessarily formulate their needs in terms of individual algorithms. Quantum-chemistry applications are usually embedded in broader simulation and experimental pipelines [2415], combining classical electronic-structure calculations, possibly quantum-computing subroutines and laboratory measurements. Near-term deployments are therefore more likely to appear first as hybrid workflows [2416].

Classical quantum chemistry

A common starting point in electronic-structure theory is the Born-Oppenheimer approximation, followed by a hierarchy of mean-field and correlated methods. Full Configuration Interaction (FCI) is exact only within a chosen finite one-electron basis and Hamiltonian, but its determinant space grows combinatorially. Conventional exact FCI therefore remains restricted to relatively small active spaces, while approximate or selected methods are used for larger systems.

One of the oldest electronic-structure methods is the **Hartree-Fock method**. Hartree developed self-consistent-field methods in the late 1920s, while Fock and Slater introduced the antisymmetrized many-electron formulation around 1930 [2417]. Hartree-Fock approximates the many-electron wavefunction by a single Slater determinant and treats each electron in the mean field of the others. It includes exchange exactly within that determinant but neglects dynamical electron correlation. This limitation motivated the development of post-Hartree-Fock methods that explicitly recover correlation effects and are used to predict chemical bonding, molecular properties and electronic structure (Figure 168).

DFT is widely used in quantum chemistry and condensed-matter physics to study molecules, solids and materials. In Kohn-Sham DFT, the basic quantity is the electron density rather than the interacting many-electron wavefunction, and exchange and correlation are represented through an approximate exchange-correlation functional. DFT is therefore not a post-Hartree-Fock method. One widely used DFT package is **Quantum ESPRESSO**, an open-source suite for electronic-structure and materials calculations based largely on plane waves and pseudopotentials [2418]. Post-Hartree-Fock methods instead improve on a Hartree-Fock reference through explicit wavefunction correlation. Common examples include:

FCI (Full Configuration Interaction), which expands the wavefunction over every Slater determinant allowed by the selected finite one-electron basis, particle number and symmetry sector. FCI is therefore exact within that basis and Hamiltonian, not exact with respect to the continuum molecular problem. Its memory and runtime requirements grow combinatorially. For N_{act} spatial active orbitals with N_{α} spin-up and N_{β} spin-down elec-

trons, the determinant count before additional symmetry reduction is $\binom{N_{\text{act}}}{N_{\alpha}}\binom{N_{\text{act}}}{N_{\beta}}$. Practical limits depend strongly on symmetry, implementation and computing resources. For example, conventional PySCF FCI was reported as limited to about 18 electrons in 18 orbitals, while a distributed 2024 implementation reached 1.31 trillion determinants for $\text{C}_3\text{H}_8/\text{STO-3G}$ [2419].

$$N_{\text{det}} = \binom{N_{\text{act}}}{N_{\alpha}} \binom{N_{\text{act}}}{N_{\beta}}$$

MBPT (Many-Body Perturbation Theory) treats electron correlation perturbatively around a reference solution. Møller-Plesset methods such as MP2 truncate a perturbation expansion at a chosen order. The *GW* approximation belongs to Green's-function many-body perturbation theory and approximates the electronic self-energy as $\Sigma \approx iGW$. These approaches trade systematic or diagrammatic approximations against computational cost and are not interchangeable truncations of one common expansion.

CC (Coupled Cluster) uses the exponential wavefunction ansatz $|\Psi\rangle = e^{\hat{T}}|\Phi_0\rangle$. Truncated coupled-cluster methods are generally not variational. CCSD retains single and double excitation operators, while CCSD(T) adds a non-iterative perturbative treatment of triple excitations. Their canonical computational costs commonly scale as $O(N^6)$ and $O(N^7)$, respectively, with N a measure of the orbital basis size. UCCSD (Unitary CCSD) replaces the cluster operator by a unitary form, which is attractive for quantum circuits but is generally more demanding to implement and is not automatically more accurate for every system.

MCSCF (multi-configuration self-consistent field) methods describe systems where a single Slater determinant is insufficient, for example near-degenerate or strongly correlated configurations. They optimize both orbital coefficients and configuration-interaction coefficients within a chosen active space. A conventional complete active-space treatment grows combinatorially and is usually limited to much smaller spaces than 50 active orbitals, although approximate solvers such as DMRG or selected CI can extend the accessible active-space size.

CAS (Complete Active Space) partitions molecular orbitals into inactive, active and virtual subsets. Selecting a chemically meaningful active space is a central approximation in both classical and quantum simulations because the correlated many-body space grows combinatorially with the number of active orbitals and electrons.

CASSCF (Complete Active Space Self-Consistent Field) performs full CI within the selected active orbitals while simultaneously optimizing the molecular orbitals, with inactive orbitals remaining doubly occupied and external virtual orbitals unoccupied in the CAS expansion.

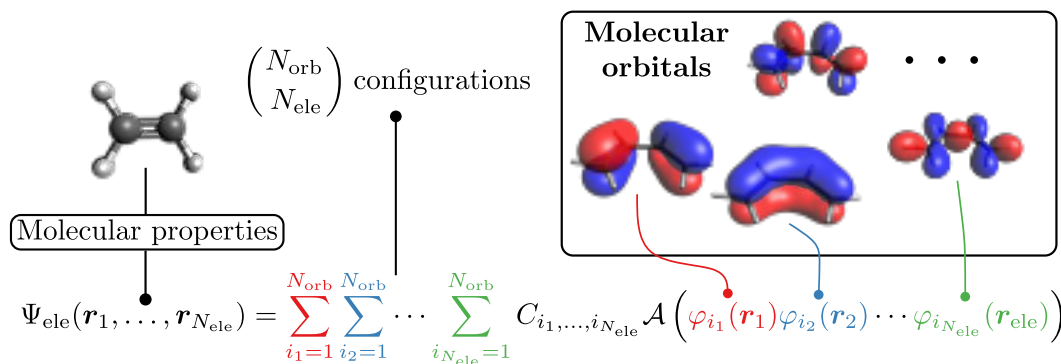


Figure 167: Example of a butadiene electronic wavefunction ψ_{ele} . In full configuration interaction within a chosen orbital basis, it is expanded as a linear combination of Slater determinants built from molecular spin orbitals. Source: [2420]. Added in 2023.

DMRG (Density Matrix Renormalization Group) is a classical variational method based on matrix-product states (MPS). It is especially efficient for low-entanglement states and one-dimensional or quasi-one-dimensional structures, but orbital ordering and tensor-network techniques make it useful for large active spaces in quantum chemistry as well [2421]. A k -local Hamiltonian is a sum of terms, each acting non-trivially on at most k subsystems. Electronic Hamiltonians contain one- and two-body fermionic interactions, but after a fermion-to-qubit mapping these interactions can become Pauli strings that are nonlocal in the qubit index. The worst-case local-Hamiltonian ground-state problem is QMA-complete, which does not imply that every chemically relevant Hamiltonian is hard in that worst-case sense. The practical goal is instead to identify chemically relevant regimes where classical methods such as coupled cluster, selected CI, DMRG or other tensor networks become costly while a quantum algorithm remains implementable. Molecular symmetries and locality can reduce both classical and quantum resources.

The Born-Oppenheimer approximation assumes that electronic and nuclear motion can be separated because nuclei are much heavier and usually move more slowly than electrons. Beyond-Born-Oppenheimer methods relax this separation and include nonadiabatic electron-nuclear couplings, which are particularly important for vibronic phenomena and photochemical dynamics.

Trotter decomposition (or trotterization) approximates a time-evolution operator such as e^{-iHt} by a product of exponentials of simpler Hamiltonian terms. It is one possible Hamiltonian-simulation method for QPE and real-time dynamics, and it can also appear in circuit implementations of unitary coupled-cluster ansätze. It is not intrinsic to VQE or QPE, since alternative methods include qubitization, linear-combination-of-unitaries and quantum signal processing [2422]. Trotterization can also be combined with basis-reduction or qubit-recycling techniques [2423]. Adiabatic state preparation is a different approach used to prepare correlated states [2424].

Linear Response (LR) describes the first-order response of a system to a sufficiently weak time-dependent or static perturbation. In quantum chemistry it underlies calculations of polarizabilities, susceptibilities, excitation spectra and other response properties under electric, magnetic or optical perturbations. Classical implementations include coupled-perturbed Hartree-Fock and time-dependent or response DFT. Quantum linear-response algorithms have also been proposed [2426, 2427].

Accuracy. Energy errors are often reported in milli-Hartree (mHa). The conventional “chemical accuracy” scale is about 1 kcal/mol, corresponding to approximately 1.59 mHa, rather than exactly 1 mHa [2428]. At room temperature, $k_B T \approx 0.94$ mHa, or about 0.59 kcal/mol. These two numerical scales are similar but chemical accuracy is a convention for useful energy differences, not a threshold derived directly from $k_B T$.

These classical algorithms are implemented in many commercial and open-source frameworks [2429]. There is no universal exact-simulation cutoff such as 22 electrons in 22 orbitals. The limit depends on symmetries, basis choice, memory, algorithm and hardware. Conventional determinant-based FCI is commonly restricted to active spaces in the high teens or low twenties, while selected-CI, FCIQMC and distributed implementations can handle much larger determinant spaces at the price of additional approximations or substantial HPC resources [2419].

Quantum computing for quantum chemistry

Quantum computing is being explored as a complement to classical electronic-structure methods, particularly for strongly correlated problems whose wavefunctions span very large Hilbert spaces. Quantum algorithms can estimate energies, response properties, reduced density matrices and other observables, but useful chemistry generally requires an end-to-end workflow that also includes basis construction, active-space selection, state preparation, error control and classical post-processing. Most algorithms first target ground- or excited-state energies, which can then support calculations of reaction

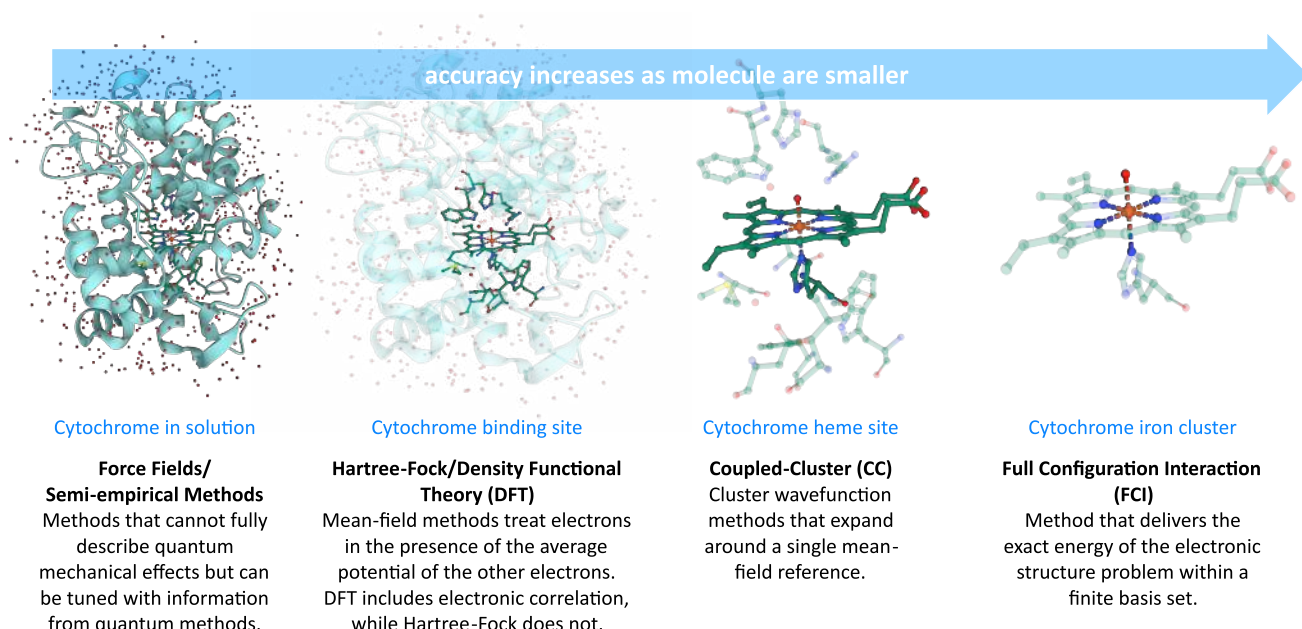


Figure 168: Common electronic-structure methods employed on classical computers. From left to right, the modeled subsystem is reduced around Compound I of cytochrome c peroxidase, allowing progressively more expensive correlated methods to become tractable. Inspired by [2425].

pathways, binding energies, spectra and reaction rates [918, 2430].

Elfving et al. distinguish **chemical accuracy**, meaning agreement with the experimentally relevant quantity, from **chemical precision**, meaning agreement with a sufficiently accurate computational reference such as FCI within a specified finite basis and active space [990]. They also emphasize that a formal quantum speedup can be industrially irrelevant if conventional computations or experiments are already sufficiently accurate, if the full real-world workflow dominates the cost, or if the output has little practical value. Their examples illustrate how a nominally attractive quantum calculation can remain much slower than a mature classical approximation for the same finite-basis target.

Basis sets are finite collections of one-electron functions used to represent molecular orbitals and approximate the continuous electronic wavefunction. FCI is not associated with a uniquely “complete” basis: it is exact only within whichever finite basis is chosen. Common families include minimal basis sets such as STO-3G, split-valence basis sets, polarized and diffuse basis sets, correlation-consistent Gaussian basis sets such as cc-pVDZ and cc-pVTZ, and plane-wave bases for periodic systems [2431].

cc-pVDZ and cc-pVTZ are members of Dunning’s correlation-consistent polarized valence basis-set hierarchy [2432, 2433]. “Double-zeta” and “triple-zeta” refer to using approximately two or three independent radial functions, respectively, to describe each valence atomic orbital, not to the physical width of an orbital. The “p” indicates added polarization functions, and diffuse functions are added separately in the augmented aug-

cc-pVXZ families. Increasing the cardinal number X in cc-pVXZ systematically enlarges the basis and improves convergence toward the complete-basis-set limit for correlated calculations.

In cc-pVDZ and cc-pVTZ, “double-zeta” and “triple-zeta” characterize the valence radial flexibility, while progressively larger sets of polarization functions are added in a correlation-consistent pattern [2432]. They are not minimal basis sets. Moving from cc-pVDZ to cc-pVTZ and beyond usually improves basis-set convergence but increases both classical and quantum resources. Quantum resource requirements depend on the active-space definition and orbital representation, not on the basis-set label alone [2434].

A useful first distinction is between NISQ-oriented algorithms, particularly VQE and sampling-based variants, and fault-tolerant algorithms such as QPE, qubitization and quantum signal processing. Current industrial-scale resource estimates often involve hundreds to thousands of logical qubits and billions or more non-Clifford gates. Some authors also discuss an early-FTQC “utility zone” with tens to roughly one hundred logical qubits and relatively short logical circuits. Such a regime could be scientifically useful without necessarily providing a commercial quantum advantage [2435].

VQE (Variational Quantum Eigensolver) estimates the ground-state energy of a many-body Hamiltonian from a parameterized trial state [2436]. Its measurement overhead can be severe. For a second-quantized electronic Hamiltonian with $O(N^4)$ Pauli terms, naive independent sampling to additive precision ϵ motivates an $O(N^4/\epsilon^2)$ measurement-count estimate under simplifying assumptions, but this is not a universal VQE runtime law [2437].

Q Basis sets and active spaces

In quantum chemistry, the one-electron basis and the correlated active space introduce distinct and complementary truncations of the electronic-structure problem.

A **basis set** is a finite collection of one-electron functions used to expand spatial molecular orbitals. In a full-rank LCAO Hartree–Fock calculation with N_{basis} basis functions, one obtains N_{basis} spatial molecular orbitals before any subsequent orbital truncation. The basis choice controls the discretization error and the size of the one-electron representation.

An **active space** partitions the molecular orbitals into inactive orbitals, which remain doubly occupied, active orbitals (N_{act}), in which the active electrons are allowed to redistribute, and external virtual orbitals, which remain empty in the CAS expansion. A Full CI calculation is performed only within the active subset. Methods such as Complete Active Space Self-Consistent Field (CASSCF) optimize both the active-spatial configuration weights and the molecular orbitals.

Increasing the one-electron basis improves spatial resolution but also increases the cost of integral generation, transformation and correlated solvers. Four-index two-electron integral transformations have a formal $O(N_{\text{basis}}^5)$ cost in their straightforward form, although modern density fitting, Cholesky and factorization techniques reduce practical scaling and storage. Increasing the active space captures more static correlation but expands the many-body determinant space combinatorially.

$$N_{\text{det}} = \binom{N_{\text{act}}}{N_{\alpha}^{\text{act}}} \binom{N_{\text{act}}}{N_{\beta}^{\text{act}}}$$

By defining an active space within a larger basis, one can retain a relatively rich spatial description while restricting the expensive many-body correlation treatment to the orbitals expected to matter most. The price is an active-space truncation error that must be controlled or corrected by embedding and dynamical-correlation methods.

Measurement grouping, low-rank factorization, adaptive ansätze such as ADAPT-VQE [833, 999, 2438, 2439], hybrid methods such as HI-VQE [2440] and Monte Carlo strategies can reduce parts of the cost. Practical VQE also requires circuit fidelities and shot rates beyond what many chemically relevant instances tolerate [2441]. Classical machine-learning tools have been explored for initialization and optimization, including LSTM networks [2442], LLM-based strategies [2443, 2444] and reservoir computing [2445]. VQE can also be combined with Monte Carlo sampling [2446, 2447] **NISQ**.

Most VQE chemistry demonstrations remain small enough that no computational quantum advantage has been established [2448, 2449]. For example, Boeing and IBM studied corrosion-related chemistry with a VQE variant using fewer than 20 qubits, without identifying an advantage regime [2450]. IBM and partners have pursued larger NISQ “quantum utility” workflows using circuits containing thousands of gates together with substantial classical HPC support [2451]. These experiments probe useful algorithmic and hardware regimes, but “utility” doesn’t mean some end-to-end quantum advantage.

QSCI (Quantum-Selected Configuration Interaction) is a hybrid quantum-classical eigensolver in which a quantum device samples an approximate wavefunction to identify important electron configurations. The Hamiltonian is then projected into the selected determinant subspace and diagonalized classically. This avoids direct estimation of every Hamiltonian expectation value and the full VQE optimization loop. It can be relatively

tolerant of some device noise because the quantum output is used for subspace selection, but its accuracy still depends on whether sampling discovers a sufficiently representative determinant set. QSCI was introduced in 2023 by QunaSys researchers [2452] **NISQ**.

SQD (Sample-based Quantum Diagonalization) is closely related to QSCI and was developed by IBM as a sampling-plus-classical-diagonalization workflow [2453], with subsequent extensions including Krylov and drift-based variants [2454, 2455]. Noisy samples can be post-processed to restore particle-number or spin constraints and to expand the selected subspace before classical diagonalization. The sampling circuit often uses a Local Unitary Cluster Jastrow (LUCJ) ansatz designed for shallow hardware-compatible state preparation [2456, 2457]. SQD has enabled electronic-structure demonstrations on tens to around one hundred qubits, but these should be described by their specific system size and accuracy rather than as the “largest to date”, since that status is changing rapidly. In short, SQD uses the QPU to find important configurations and a classical solver to diagonalize the resulting subspace **NISQ**.

EWf (Embedded Wave Function) is an embedding framework that partitions a large molecular system into localized fragments coupled to an effective bath. Related constructions build on density-matrix embedding ideas and use classical mean-field or perturbative information to define and enlarge the embedded problem. Weakly correlated regions can remain classical, while selected strongly correlated fragments are treated with more expensive solvers, including SQD on a quantum

processor. Recent IBM-led work explores this type of quantum-centric embedding workflow [2458, 2459].

QPE (Quantum Phase Estimation) can estimate an eigenenergy of a molecular Hamiltonian when supplied with a state having non-negligible overlap with the target eigenstate. In chemistry it is usually associated with FTQC because accurate Hamiltonian simulation requires deep coherent circuits [2460]. Gate counts can be prohibitive [2461, 2462], and state preparation is itself a major algorithmic cost rather than a purely classical preprocessing step [2463]. QPE and related fault-tolerant algorithms can also support spectroscopy and response calculations [2464–2468]. Resource estimates are highly basis-, active-space- and algorithm-dependent, with chemically challenging examples often requiring hundreds to thousands of logical qubits. Compared with VQE, QPE shifts the dominant burden from repeated shallow expectation-value measurements and classical parameter optimization toward deep Hamiltonian simulation, fault tolerance and high-quality initial-state preparation [2469] **FTQC**.

Quantum linear solvers (QLS), including HHL-type algorithms, can in principle accelerate linear systems that arise in some response, Green’s-function and property calculations. Their strongest asymptotic speedups require favorable sparsity, condition number, efficient state preparation and an output that can remain encoded as a quantum state. State loading and observable readout can significantly reduce or eliminate the end-to-end speedup. It is not a generic ground-state solver [2415] **FTQC**.

Qumodes on continuous-variable photonic processors can also be used in quantum-chemistry algorithms, although fermionic electronic-structure Hamiltonians require an encoding into the available bosonic degrees of freedom [2470]. Continuous-variable processors have also been proposed for molecular dynamics and reaction-related simulations [2471] **NISQ**.

Boson-sampling-inspired and variational photonic methods have also been proposed for selected quantum-chemistry tasks, including approximate electronic-structure problems [2472] **NISQ**. These proposals should be distinguished from standard Gaussian boson sampling, whose most established chemistry connection is to vibronic spectra rather than generic electronic ground-state estimation.

Vibrational energies can be targeted with variational NISQ algorithms or fault-tolerant phase-estimation and Hamiltonian-simulation methods [2473–2476]. Resource estimates can be extremely demanding. Majland et al. report fault-tolerant instances ranging from roughly 10^{10} to 5×10^{15} Toffoli gates and from about 430 to 10^4 logical qubits for molecules including formaldehyde, benzene and naphthalene derivatives [2477] **NISQ** **FTQC**.

Scattering calculations for chemical reactions model collisions between atoms and molecules and can support combustion and gas-phase reaction studies.

Variational quantum approaches have been proposed and demonstrated at small scale [2478] **NISQ**.

Quantum Machine Learning has also been explored for learning molecular states, energies or surrogate models when suitable training data are available [2479]. No generic quantum advantage has been established for these chemistry applications, and data loading, trainability and classical baselines remain important constraints [2480] **NISQ**.

Jordan-Wigner transformations map fermionic creation and annihilation operators onto qubit Pauli operators. This is a standard encoding for gate-based quantum simulation of fermionic systems. Because the mapped Pauli strings can be nonlocal, alternatives such as Bravyi-Kitaev mappings are sometimes preferred. Annealers can encode selected fermionic problems only after some additional transformations to an Ising/QUBO form [2483]. It can for example be used in VQE calculations of ground-state energies or molecular spectra [2484] **NISQ**.

Sampling methods cover ground-state, finite-temperature and real-time problems and include quantum-assisted variants of classical Monte Carlo techniques [2485]. Quantum walks and amplitude-amplification methods can provide theoretical quadratic improvements in sampling or search complexity when the required coherent oracles are available [2486] **FTQC**. Quantum-Assisted Variational Monte Carlo (QA-VMC) combines quantum-generated information with classical variational Monte Carlo or neural-network states [2487, 2488] **NISQ**. Quantum-assisted auxiliary-field quantum Monte Carlo (QC-AFQMC) instead uses trial wavefunctions informed by quantum hardware to reduce bias in a classical AFQMC calculation [2489] **NISQ** **FTQC**.

Chen and Chan proposed in 2025, with the work accepted in Physical Review Letters in September 2026, a framework for robust quantum speedups in correlated electronic structure based on an effective correlation length L_c rather than simply the total system size [2490]. In the analyzed regime, low excitation rank, finite correlation range, screened long-range interactions and inexpensive state preparation can yield polynomial separations in L_c . For $m = 3$ and $D = 3$, the storage avoided by the quantum representation leads to illustrative separations involving powers up to L_c^{18} . The paper also emphasizes that concrete gate counts, circuit depths and fault-tolerance overheads are still required before these asymptotic separations can be converted into practical advantage claims.

Development tools

Many software tools already support quantum-chemistry workflows, including **OpenFermion**, originally developed at Google with contributions from the broader community, and **myQLM-fermion** from Eviden [2491]. Other examples include **TenCirChem** [2492] and **ChemiQ** [2493], both developed in China.

Q How ground state preparation cost scales?

A QPE calculation needs an initial trial state with non-negligible overlap with the target eigenstate. If $S = |\langle \Phi | \Psi_0 \rangle|$ is the ground-state amplitude overlap, an ideal coherent QPE measurement samples the ground-state eigenphase with probability S^2 , so without amplitude amplification the expected repetition cost scales as $1/S^2$ [2481]. The quality of this overlap depends on the electronic structure and the state-preparation method. Strong correlation and increasing system size can make simple Hartree–Fock-like initial states inadequate, but a small spectral gap does not by itself mathematically imply an exponentially small overlap.

Adiabatic state preparation (ASP) provides one route to improve the initial state by evolving from an easy Hamiltonian toward the target Hamiltonian. Standard adiabatic bounds often contain an inverse-square dependence on the minimum spectral gap along the chosen interpolation, approximately $1/\Delta_{\min}^2$, together with matrix-element and schedule dependence. This becomes exponentially costly only when the relevant gap itself closes exponentially with problem size. Conversely, accepting a trial state with exponentially small S makes unamplified QPE ex-

ponentially expensive through the $1/S^2$ repetition factor. These are serious state-preparation bottlenecks, but they do not establish that every strongly correlated molecule loses quantum advantage.

A credible QPE advantage therefore requires an end-to-end regime in which the target problem is difficult for the best classical methods while the initial-state preparation, Hamiltonian simulation, repetitions and error-correction overhead remain manageable. The relevant quantities are not only system size and the spectral gap, but also the achievable overlap, the structure of the Hamiltonian and the quality of classical competing methods [2481].

Ground-state energy estimation is often treated as the baseline quantum-chemistry task. A recent resource analysis identifies state preparation and logical clock speed as important bottlenecks. It mentions instances around of a few hundred spin orbitals, but this is not a universal 250-spin-orbital limit [2482]. Excited-state, real-time dynamics and finite-temperature free-energy calculations usually add further overhead through denser spectra, longer coherent evolution, state preparation or thermal sampling.

A number of quantum-hardware and software companies also develop chemistry and materials-simulation frameworks. Examples discussed later in the book include **HQS** (Germany), **Quantinuum** and its InQuanto platform originating at CQC [2494], **Pasqal** (France, following its 2021 acquisition of Qu&Co), **Good Chemistry** (Canada), **Menten AI** (USA), **Q1t** (the Netherlands), **QSimulate** (USA), **QunaSys** (Japan) and **Riverlane** (UK).

Distributed quantum computing

Large quantum-chemistry calculations can be demanding in qubit count, circuit depth and memory. Distributed quantum computing is therefore one possible future scaling route, provided that the algorithm can be partitioned across interconnected QPUs without excessive entanglement-generation, communication and synchronization overhead. This architecture is discussed in the Quantum Communications part of the book.

In a 2025 paper, US researchers designed a distributed quantum computing architecture named distributed unitary selective coupled cluster (dUSCC), simulating a three-cluster $(H_4)_3$ chain distributed across three cold-atom QPUs with photonic interconnects. Their proof of principle uses a compilation scheme exploiting pseudo-commutativity while accounting for slower inter-module interactions [2495] **NISQ**. The variational NISQ concept could in principle be extended to FTQC schemes.

Sufficient inter-module connectivity and entangling capacity are important because a weakly coupled partition can become amenable to classical decomposition. However, the fact that each individual module is classically simulable does not by itself imply that the globally entangled distributed computation is classically easy.

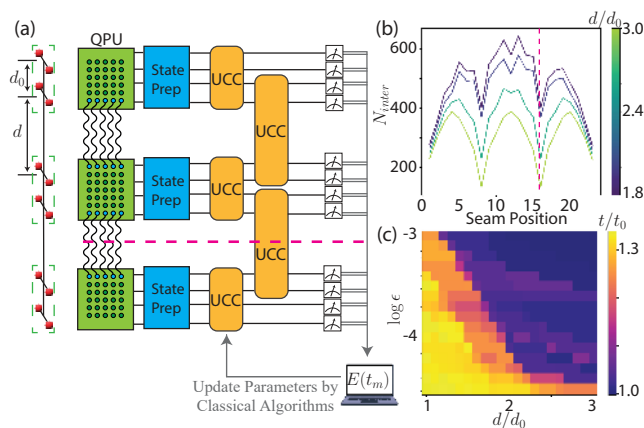


Figure 169: A concept for distributed quantum-chemistry computation across three cold-atom QPUs interconnected by several photonic qubit-to-qubit links. Source: [2495].

Likewise, an HPE team analyzed in 2026 the communication and partitioning costs of distributing electronic-structure simulations across several QPUs compared with classical circuit knitting [2496].

3.3.2 Chemical engineering

Most quantum-chemistry use cases begin with electronic ground-state calculations and then extend to excited states, response properties or reaction dynamics. Current QPUs can reproduce small active-space calculations for few-atom molecules such as H_2 , LiH or H_2O , but without demonstrated quantum advantage [2497]. Larger demonstrations may use tens of spatial orbitals, but the corresponding number of atoms is strongly basis- and active-space-dependent. Realistic photodissociation, nonadiabatic dynamics and docking or binding calculations can require much deeper circuits and, in current resource estimates, generally move toward the FTQC regime.

As shown visually with the badge colors below, most published chemistry demonstrations remain in the no-demonstrated-advantage NISQ or analog regimes. Many VQE demonstrations use fewer than a few tens of active qubits and shallow circuits that remain accessible to classical simulation. Larger sampling-based demonstrations can reach roughly 50 to 100 qubits.

NISQ In September 2017, **IBM** demonstrated a hardware-efficient VQE for small molecules up to BeH_2 , using up to six qubits on a seven-transmon superconducting processor, not 16 computational qubits [2498]. TotalEnergies and Jean-Philip Piquemal later studied benzene under spatial deformation with VQE in classical quantum-circuit simulation [968, 2499]. In the minimal STO-3G basis, the full benzene representation contains 36 spatial orbitals or 72 spin orbitals, hence 72 qubits under a direct spin-orbital mapping. Their in-house simulator could emulate up to 35 qubits, so they relied on active-space reductions. Full-Hamiltonian VQE estimates such as 330,816 Pauli terms illustrate the severe measurement burden, although grouping, factorization and alternative measurement strategies can reduce the number of separately measured terms.

NISQ In 2026, IBM published a preprint on the simulation of a half-Möbius molecule using up to 100 qubits to represent 50 spatial orbitals on an IBM Heron r3 processor [2500]. They scaled the non-parametric SqDRIFT algorithm [2454] and used shallow fixed circuits to sample bitstrings and construct a classical diagonalization subspace of roughly ten million configurations, with quantum error mitigation and extensive classical post-processing. No quantum advantage was demonstrated because classical HCI-based calculations remained more accurate at this scale. The approach is an interesting hybrid blueprint that avoids a variational optimizer and can access observables such as Dyson orbitals. The related peer-reviewed Science work experimentally created the C_{13}Cl_2 half-Möbius molecule and mapped its Dyson orbital using quantum-assisted multireference calculations [2501].

analog Some progress has been reported with quantum annealing in chemical optimization. A 2023 paper from Mizuno et al. at Hokkaido University studied

chemical reaction pathway optimization using D-Wave annealers [2502]. Its physical-qubit overhead depends on the logical problem graph and the minor-embedding required by the hardware topology, so a quadratic physical-to-logical scaling is not universal. The study mainly explored scaling and penalty-parameter effects rather than demonstrating advantage. A 2023 chemical-space virtual-screening study used a 512-qubit D-Wave annealer and reported useful results but did not benchmark them against state-of-the-art classical alternatives [2503].

FTQC An AIST team in Japan presented in 2026 a hybrid architecture for molecular total-energy calculations that divides the workload between early FTQC and classical HPC [2504]. The quantum processor prepares a Complete Active Space (CAS) state to capture difficult static correlation. A Quantum Dominant Orbital Selection (QDOS) stage then identifies the most important active orbitals, and an HPC executes Subspace Dynamical Correlation (SDC) to recover missing dynamical correlation. The proposed compression is intended to avoid a costly full QPE calculation and could be relevant to strongly correlated transition-metal complexes and photochemical materials problems.

NISQ Modeling chemical reactions and molecular dynamics has been demonstrated at small scale on NISQ QPUs [2505], usually with major simplifications [2506, 2507], including gas-phase models [2508]. Chemically accurate ab-initio dynamics at industrially relevant scales is generally expected to require fault-tolerant computation in current resource studies, using QPE, qubitization, QSP or related Hamiltonian-simulation algorithms rather than QPE alone [2509–2511].

NISQ More recent work on quantum dynamics includes trapped-ion simulations of nonadiabatic chemical processes [2512] and studies of frustrated Ising fullerene models using D-Wave annealers [2513]. **FTQC** Resource estimates for photonic fault-tolerant catalysis simulations remain extremely large, reaching approximately 10^{11} to 10^{14} Toffoli gates per femtosecond of simulated evolution in the studied cases [2514].

NISQ Cleveland Clinic, RIKEN and IBM Quantum reported in 2026 a fragmented treatment of two protein-ligand complexes containing 11,608 and 12,635 atoms. Quantum embedding and SQD sampling used up to 94 qubits on 156-qubit IBM Heron processors, with 9,200 circuits and about 1.3×10^9 shots over more than 100 hours. The dominant fragment calculations ran on the Fugaku and Miyabi-G HPC systems and represented about 99% of the reported computational load. No quantum advantage was demonstrated because the QPU acted as a configuration sampler feeding a classical diagonalizer and the quantum-treated fragments remained accessible to classical selected-CI methods [2515, 2516]. Cleveland Clinic researchers also extended the SQD toolchain to molecular-geometry optimization through embedded energies and gradients, demonstrated for menthone and benzidine with up to 70 qubits [2517].

FTQC Boeing researchers studied fullerene-encapsulated cyclic ozone, $(12\text{ O}_3)@\text{C}_{180}$, as a candidate high-energy-density propellant and estimated FTQC resources for QPE-based electronic-structure calculations [2518]. Their largest active-space instance exceeds 4,000 logical qubits and about 2.6×10^{13} T gates. Under their idealized superconducting surface-code architecture, this maps to roughly 6×10^6 physical qubits and a runtime slightly above six years.

FTQC One important chemistry research area is corrosion. Researchers from the USA and Australia estimated FTQC resources for characterizing magnesium-alloy corrosion in aqueous environments and niobium-rich refractory alloys at high temperature. Their pyLIQTR-based estimates use qubitized QPE and span roughly 8,700 to 725,000 logical qubits for the studied instances [2519]. The work also illustrates how tightly classical electronic-structure preprocessing, quantum simulation and classical post-processing are coupled (Figure 170). A separate Airbus-Capgemini study reported different resource estimates based on a trapped-ion fault-tolerance model [2520].

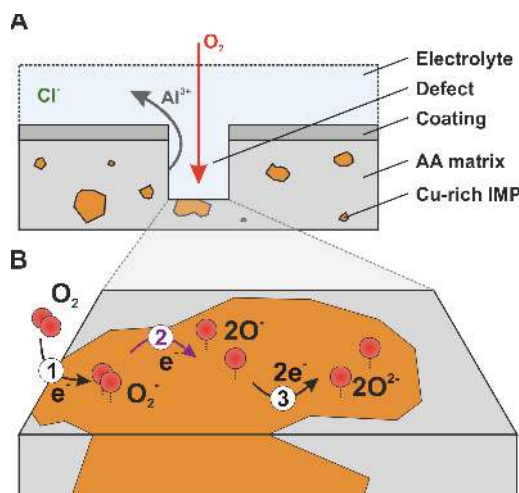


Figure 170: How corrosion affects materials and what researchers are trying to simulate. Source: [2520].

Several chemical companies have investigated quantum-computing use cases. **Dow Chemical** has worked with **1QBit** on quantum-chemistry pilots since 2017. **Mitsubishi Chemical** and Materials Magic, formerly associated with Hitachi Metals, have also explored quantum computing with IBM.

NISQ In 2023, Mitsubishi published a study of OLED-related molecular design using VQE and QAOA on approximately six qubits [2331]. In 2024, Mitsubishi Chemical, Deloitte Tohmatu and Classiq reported circuit-depth reductions for QAOA- and QPE-based variants, but the demonstrated problem instances involved only 16 and 11 qubits, respectively, far below a regime where quantum advantage was established [2521].

NISQ **NISQ** Mitsubishi Electric Research Laboratories proposed Geo-ADAPT-VQE in 2026, using a quantum-information-geometric metric to select ansatz

operators. The method improves circuit construction for ground-state estimation of small molecules such as H_2O and BeH_2 , but the reported results do not establish quantum advantage [2522].

NISQ Capgemini, IBM and Q-CTRL reported in 2026 an SQD hardware study of an open-shell 3d transition-metal complex and extended the workflow to an implicit-solvent self-consistent reaction-field loop [2523]. Their test system, $[\text{Co}(\text{H}_2\text{O})_5\text{CO}_2]^{2+/3+}$, was studied across oxidation states, spin multiplicities and a Co-O(CO_2) dissociation coordinate using 50 qubits on an IBM Heron r3 processor and roughly 2×10^5 to 5×10^5 shots per data point, with Q-CTRL Fire Opal error suppression. No quantum advantage was demonstrated. The classical HCI reference remained more accurate with fewer determinants.

NISQ **NISQ** **JSR Corporation**, IBM and the University of Chicago also applied SQD to iron porphyrin, a chemically relevant motif in catalysis, and reached similarly cautious conclusions about present-day advantage [2524] (see Figure 5 in the paper).

NISQ **NISQ** Fujitsu reported in 2026 a quantum lattice-Boltzmann solver for three-dimensional advection-diffusion transport on trapped-ion hardware. It remains a small fluid-dynamics demonstration and does not establish computational advantage [2525].

Covestro
Germany, 2021



NISQ **Covestro** and **QC Ware** began working together in 2021 and announced a five-year collaboration in 2022 on quantum algorithms for materials and catalyst discovery. Their work includes methods to reduce quantum resources for molecular simulation and energy-gradient calculations [2526, 2527]. A regularized compressed double-factorization (RC-DF) method was used to obtain compact molecular-Hamiltonian representations for 12- to 20-qubit experiments [2528]. These demonstrations do not establish quantum advantage. In parallel, the collaboration has studied fault-tolerant resource requirements for larger chemistry problems [2529, 2530].

FTQC Google Quantum AI and collaborators presented in 2026 an end-to-end FTQC resource model for real-time chemical dynamics and photodissociation. Their estimates include 3.21×10^{13} Toffoli gates and 3,774 logical qubits for CH_4 , rising to 3.07×10^{16} Toffoli gates and 7,698 logical qubits for BrCH_2CHO when estimating a quantum yield with QAE [2531] (Table 14). These counts imply extremely long runtimes unless the fault-tolerant architecture sustains very high effective non-Clifford throughput and parallelism. For scale, 3×10^{16} strictly sequential operations at 1 MHz would take about 10^3 years, although a real architecture can parallelize parts of the computation and the relevant circuit depth is not identical to the raw gate count.

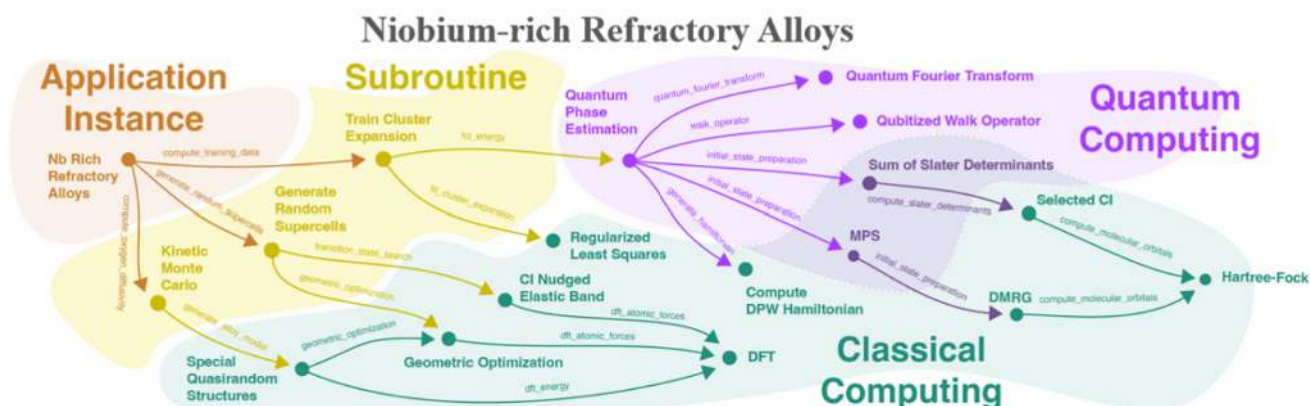


Figure 171: End-to-end hybrid classical-quantum workflow proposed for digitally simulating how niobium-rich refractory alloys could reduce corrosion. Source: [1796].

FTQC Xanadu and Toronto developed in 2026 a quantum algorithm for pre-Born-Oppenheimer chemical dynamics that couples electronic and nuclear motion using a quantum signal processing (QSP)-based framework [2532].

Several lower-level chemistry and physics demonstrations also have no obvious direct industrial use case:

NISQ Simulating NMR spectroscopy on NISQ hardware with error mitigation has been demonstrated on IBM and IonQ systems using up to 34 physical qubits and circuits with roughly 4,600 gates [2533]. A 2026 VQE-based molecular spectroscopy study used only 24 qubits [2534]. These are useful algorithmic demonstrations but do not establish quantum advantage.

NISQ Various techniques improve the accuracy or size consistency of variational circuits without yet demonstrating computational quantum advantage [2535, 2536].

FTQC Quantum algorithms have also been proposed for heat conduction and anomalous transport using a β -Fermi–Pasta–Ulam–Tsingou (β -FPUT) chain, with one study estimating up to 576 logical qubits and circuit depths of 10^7 to 10^8 for its target instances [2537].

3.3.3 Fertilizer production

Nitrogen fertilizers are used worldwide to increase agricultural productivity. Industrial ammonia is the starting point for essentially all mineral nitrogen fertilizers.

A frequently touted quantum-computing application is the discovery of lower-energy routes to ammonia and fertilizer production. We will dissect this story. The direct path from simulating FeMoCo to replacing Haber-Bosch is, at least in the short to medium term, largely overstated.

Ammonia industry production

Using the 2010 global reactive-nitrogen budget compiled by Fowler et al., Haber-Bosch production accounted for about 120 Tg N/year out of a total fixation of roughly

413 Tg N/year, or about 29% [2538]. Modern ammonia production combines atmospheric N_2 with H_2 , most commonly obtained from natural-gas steam methane reforming or, especially in China, coal gasification.

Before Haber-Bosch became dominant, other industrial nitrogen-fixation routes were used. The Birkeland-Eyde process oxidized atmospheric N_2 at very high temperature to NO and subsequently NO_2 , which could be converted to nitric acid and nitrates. The Frank-Caro process fixed nitrogen as calcium cyanamide, $CaCN_2$, by reacting N_2 with calcium carbide at roughly 1,000 °C. These historical routes were displaced mainly because Haber-Bosch became more economical and energy-efficient at large scale, rather than because acid rain or cyanide release is an intrinsic defining limitation of each route.

The Haber-Bosch synthesis process uses iron catalysts and operates at several hundred degrees Celsius and pressures of order 10^2 bar (Figure 175). The dominant energy and carbon burden of conventional natural-gas ammonia comes from producing H_2 by steam methane reforming, not from a carbon-capture benefit of SMR itself. The IEA reports a global average ammonia energy intensity of about 41 GJ/t NH_3 , compared with best-available-technology values around 28 GJ/t for natural-gas plants and 36 GJ/t for coal-based plants [2539]. More than 70% of production is natural-gas-based and most of the remainder is coal-based. Near-zero-emission routes based on electrolytic hydrogen, methane pyrolysis or fossil hydrogen with high-rate CCS remain a small fraction of current production.

Around 70% of industrial ammonia is used to make fertilizers, with the remainder used in plastics, synthetic fibers, explosives and other chemicals [2539]. The IEA estimates about 450 Mt CO_2 /year of direct emissions from ammonia production and roughly 170 Mt CO_2 /year of indirect emissions from electricity and downstream urea use. This corresponds to about 1.3% of energy-system CO_2 emissions, not 6% of global CO_2 emissions

Molecule	η	Logical qubits			Toffoli gates (C_{Toff})			
		State storage (C_{data})	Ancilla qubits (C_{anc})	Total	Nuclear state prep. (10^9)	Electronic state prep. (10^9)	Time evolution (10^{12})	Quantum yield using QAE (10^{12})
CH ₄	15	585	3,189	3,774	2.79	0.10	1.89	32.1
CH ₂ OO	29	1,131	3,385	4,516	3.32	0.42	18.9	322
C ₄ H ₆ O	49	2,058	3,811	5,869	7.39	0.98	60.1	1,002
HNO ₄	46	1,794	3,542	5,336	3.84	0.56	64.0	1,009
CF ₃ CO ₂ H	64	2,688	3,904	6,592	4.88	1.69	191	3,240
C ₅ -HPALD	78	3,510	4,289	7,799	9.87	2.19	239	4,070
HCFC-132b	74	3,330	4,243	7,573	4.72	3.43	742	12,600
CH ₃ OBr	58	2,496	4,512	7,296	4.12	1.25	1,350	22,900
BrCH ₂ CHO	65	3,120	4,578	7,698	4.10	1.70	1,810	30,700

Table 14: resource estimates for simulating photodissociation yields in molecules whose full quantum dynamics is classically intractable. η is the total number of particles (electrons and nuclei). It provides space and time resources for the algorithm, with the qubit counts divided into qubits to store the state (C_{data}) and the ancillas (C_{anc}). The Toffoli gate counts for time evolution indicate one dynamics run for simulation time $t = 30$ fs, and the QAE counts give the cost of calculating the quantum yield to accuracy $\epsilon = 0.095$ (involving multiple simulations of the dynamics). Source: Google AI in [2531].

[2539]. Other major plant nutrients include phosphorus- and potassium-based fertilizers.

Understanding nature

In nature, biological nitrogen fixation is performed by diazotrophic microorganisms using nitrogenase enzymes. Some are free-living, while others live symbiotically, notably in legume root nodules. In the Fowler et al. 2010 budget, natural terrestrial biological fixation contributes about 58 Tg N/year and marine biological fixation about 140 Tg N/year, together roughly 48% of the 413 Tg N/year global fixation estimate [2538]. Agricultural biological fixation contributes another roughly 60 Tg N/year. Lightning fixes about 5 Tg N/year, approximately 1%, primarily by producing NO_x, not ammonia. Fossil-fuel and biomass combustion add roughly 30 to 40 Tg N/year of anthropogenic fixed nitrogen, again mainly as NO_x, not as an ammonia source [2538]. Recycling of organic nitrogen from dead biomass is part of the nitrogen cycle but is not new fixation of atmospheric N₂.

The motivation is therefore to understand biological nitrogen fixation and use biomimicry to inspire lower-temperature or lower-pressure ammonia chemistry. This is where FeMoCo often enters quantum-computing discussions. However, the catalytic mechanism and the surrounding biochemical machinery are substantially more complicated than the simplified narrative of “simulate FeMoCo and obtain a new Haber–Bosch process.”

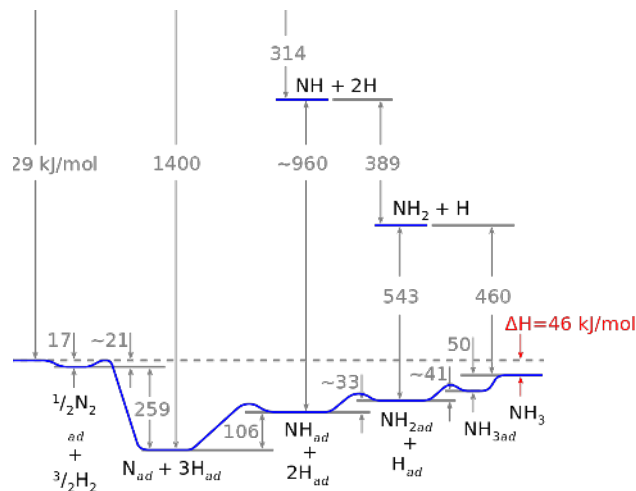


Figure 172: The Haber–Bosch chemical process. Source: [2540]

Nitrogen occurs in oxidation states ranging from highly oxidized nitrate, NO₃[−], to highly reduced ammonia/ammonium, NH₃/NH₄⁺, with intermediates including NO₂[−], NO, N₂O, N₂ and NH₂OH depending on the pathway. Excess nitrate is a major form of reactive-nitrogen pollution in intensive agriculture. The nitrogen cycle complements the carbon and oxygen cycles illustrated by photosynthesis and respiration. Photosynthesis uses CO₂, water and light to synthesize carbohydrates while releasing O₂. Cellular respiration consumes reduced carbon compounds and usually O₂ to generate ATP while releasing CO₂ and water (Figure 173).

Many constituents of living cells, including amino acids, proteins and nucleic acids, contain reduced nitrogen. The nitrogen cycle includes nitrogen fixation, converting N₂ to NH₃/NH₄⁺, nitrification, oxidizing ammonia/am-

monium to nitrite and nitrate, assimilation, incorporating inorganic nitrogen into biomass, ammonification, releasing ammonium from organic nitrogen, and denitrification, reducing nitrate and nitrite through intermediates such as NO and N₂O to N₂. Denitrification is not the conversion of ammonia to N₂, which is a different set of pathways.

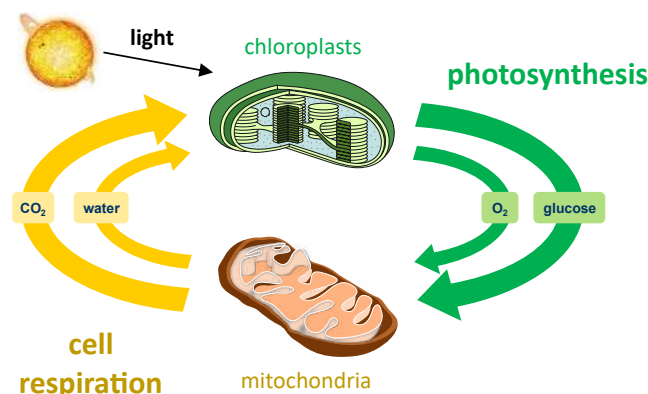


Figure 173: Photosynthesis and cellular respiration are two central biochemical cycles in life. Nitrogen fixation is another fundamental biogeochemical process, performed by diazotrophic microorganisms and supplying reactive nitrogen to ecosystems. Source: Olivier Ezratty reconstitution using open-source schematics. 2023.

Biological nitrogen fixation requires coordinated electron transfer, ATP hydrolysis and protection of nitrogenase from O₂. The immediate electron donors to nitrogenase are typically ferredoxin or flavodoxin, while the metabolic source of reducing equivalents depends on the organism. Nitrogenase also requires metal-sulfur cofactors, MgATP and dedicated cofactor-assembly machinery. In heterocyst-forming cyanobacteria, ATP and reductant are generated through photosynthetic and respiratory electron-transport pathways in the heterocyst, not in mitochondria, since bacteria and cyanobacteria do not contain mitochondria [2541, 2542].

Some filamentous cyanobacteria differentiate specialized microoxic cells called heterocysts for nitrogen fixation, while many other nitrogen-fixing cyanobacteria use different spatial or temporal strategies [2541, 2542]. Molybdenum nitrogenase consists of two component proteins: the Fe protein NifH, a homodimer containing a [4Fe-4S] cluster, and the MoFe protein NifDK, an $\alpha_2\beta_2$ heterotetramer containing the P-clusters and FeMoCo catalytic cofactors. Nitrogenase reduces N₂ to NH₃, which is predominantly protonated to NH₄⁺ at physiological pH. Hydrogenases may recycle the H₂ obligatorily produced by nitrogenase in some organisms, but they are not the universal direct electron donor to nitrogenase.

The overall stoichiometry of molybdenum nitrogenase is commonly written $\text{N}_2 + 8\text{H}^+ + 8\text{e}^- + 16\text{MgATP} \rightarrow 2\text{NH}_3 + \text{H}_2 + 16\text{MgADP} + 16\text{P}_i$ [2541, 2542]. Electrons are transferred from ferredoxin or flavodoxin to the [4Fe-4S] cluster of NifH and then, coupled to ATP binding and hydrolysis, through the P-cluster of NifDK toward FeMoCo. Two ATP molecules are hydrolyzed per transferred electron in this mini-

mum stoichiometry. A schematic overview is shown in Figure 174 [2543].

The FeMoCo cofactor contains seven Fe atoms, one Mo atom, nine sulfides, one interstitial carbon atom and a homocitrate ligand. The widely used 39-atom FeMoCo model $\text{C}_7\text{H}_9\text{Fe}_7\text{MoN}_2\text{O}_3\text{S}_{10}$ contains 254 electrons in a frozen-core treatment, but this is different from the active-space size or the number of basis orbitals. Elfving et al. report 1,365 spatial orbitals for a triple-zeta basis and 2,367 spatial orbitals for a quadruple-zeta basis for that model, while prominent quantum resource studies use much smaller selected active spaces such as 54 electrons in 54 orbitals or 113 electrons in 76 orbitals [230, 990].

What could quantum computing contribute? A central target is the strongly correlated electronic structure of FeMoCo and related iron-sulfur clusters, which may help discriminate reaction intermediates and electron-transfer pathways. The quantum calculation would still model a selected active space embedded in a much larger protein and solvent environment, not the full biological nitrogen-fixation process.

Simulating nature

FTQC Fault-tolerant resource estimates for FeMoCo remain large. For the 54-electron, 54-orbital Reiher active-space Hamiltonian, Lee et al. estimated about 2,142 logical qubits and 5.3×10^9 Toffoli gates using tensor hypercontraction. For the larger 113-electron, 76-orbital Li Hamiltonian, their estimate was about 2,196 logical qubits and 3.2×10^{10} Toffoli gates [230]. Under a specific surface-code model with a 1 μs cycle and 0.1% physical error rate, the smaller instance mapped to roughly four million physical qubits and under four days of runtime. A later peer-reviewed Google-led work introduced spectral amplification and reported factors of about 4 to 195 reduction in fault-tolerant cost across studied electronic-structure instances, including iron-sulfur systems [231] (Figure 176).

These calculations estimate ground-state energies of active-space FeMoCo models, not the complete nitrogenase catalytic cycle. The precise N₂ binding geometry and sequence of intermediates remain active research questions [2549]. Chemically accurate treatment of Mo also requires relativistic effects, which can be incorporated through relativistic Hamiltonians or effective-core potentials, rather than being categorically absent from quantum algorithms. The [4Fe-4S] and P-cluster electronic-structure problems are smaller but can still require fault-tolerant resources at chemically demanding accuracy [2550, 2551].

QI Classical progress is also rapid. A Chinese team reported a GPU-accelerated DMRG calculation of a 73-orbital P-cluster active space [2552]. In January 2026, a Caltech team led by Garnet Kin-Lic Chan presented a tensor-network-based classical calculation of a 76-orbital FeMoCo active space [233]. These results are an impor-

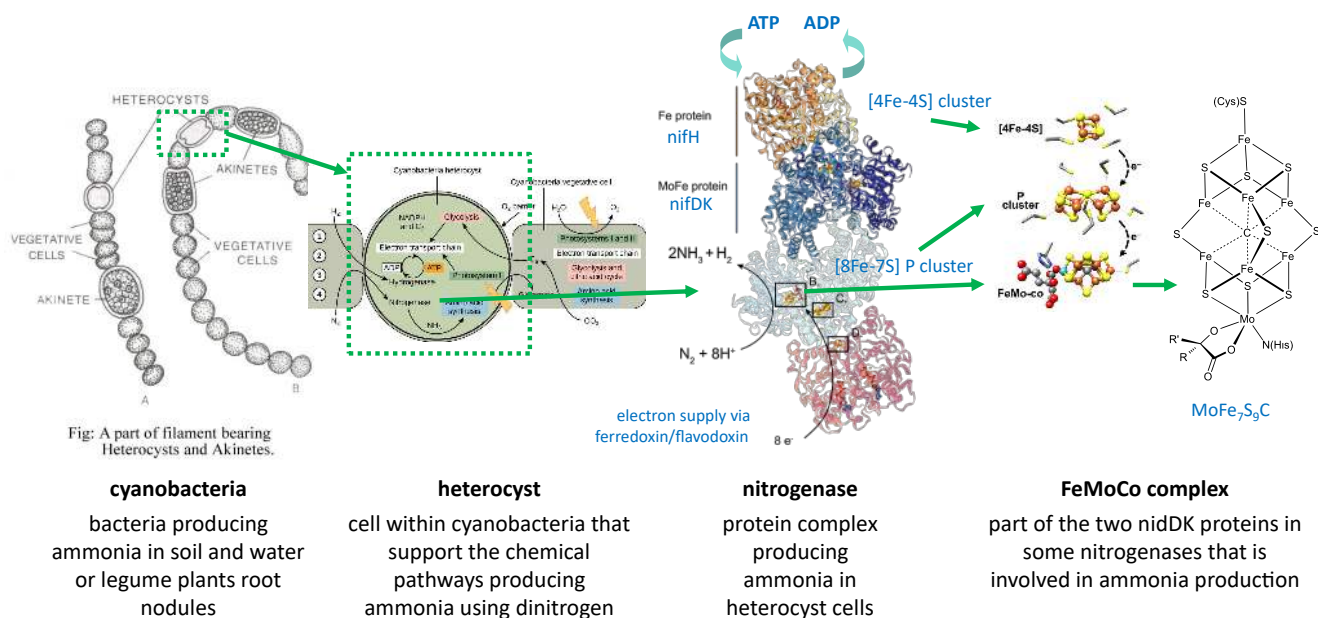


Figure 174: Where FeMoCo stands in the cyanobacteria ammonia production cycle. Schematics assembled by the author using various sources: [2544–2548]. Added in 2023. Corrected in September 2026.

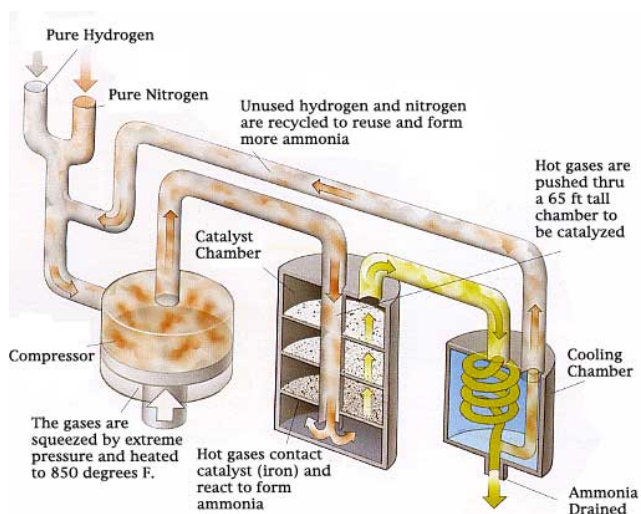


Figure 175: The usual Haber-Bosch process. Source: [2540].

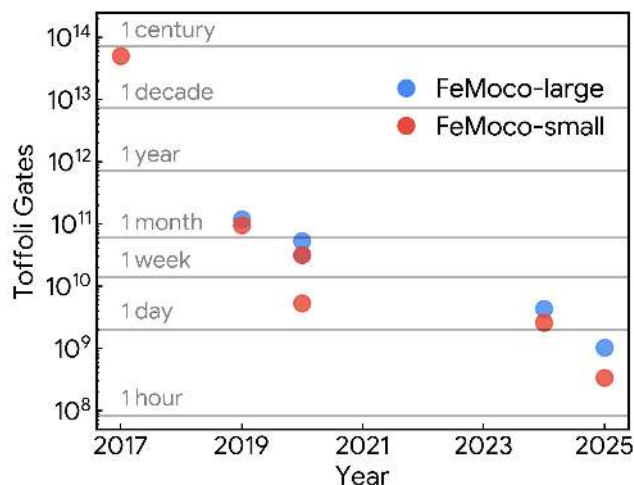


Figure 176: How Google's computing time estimates have progressed over time for computing the ground state of FeMoCo. Source: [231].

tant reminder that quantum resource estimates must be compared with improving classical algorithms and hardware, not with a frozen classical baseline.

The biosynthesis of nitrogenase cofactors is complex. Major assembly proteins and pathways, including NifB and NifEN in FeMoCo biosynthesis, have been identified, although mechanistic details and organism-specific variants remain active areas of research [2543, 2553].

Many sources estimate the FTQC resources for FeMoCo, but fewer explain how a ground-state calculation would translate into an industrial catalyst or process [228]. Would it be sufficient to resolve the full nitrogenase mechanism and electron-transfer pathway? Would it determine how N_2 binds and is activated? Would

those results enable an efficient biomimetic process, and would the full reaction network itself remain tractable? The gap between understanding one electronic-structure problem and engineering an industrial process can be very large.

A possible long-term objective is to develop new catalysts or biological production routes that operate under milder conditions than Haber-Bosch. But comparing the two processes by a “yield” of less than 5% versus more than 95% is not meaningful. The relevant metric is closer to energy intensity and full lifecycle resource cost. The IEA reports a global average of roughly 41 GJ/t NH_3 and a best-available natural-gas plant value near 28 GJ/t NH_3 [2539]. Chan estimates the minimum

physiological ATP hydrolysis implied by nitrogenase stoichiometry at about 24 GJ/t NH₃, but explicitly notes that this excludes cellular maintenance, enzyme synthesis and other biological overheads [2405].

Garnet Chan therefore cautions against presenting biological nitrogen fixation as an obviously more energy-efficient replacement for Haber–Bosch [2405]. Nitrogenase is slow and the chemical interest is at least as much in understanding how it activates the N≡N triple bond under ambient conditions and in deriving principles for new catalysts. Moreover, an FeMoCo active-space ground state is only one component of the problem. The surrounding protein residues, solvent, proton/electron delivery and multiple catalytic intermediates can require models containing hundreds of atoms [2554]. This substantially weakens the common short-term narrative that quantum simulation of FeMoCo alone will directly improve fertilizer production.

Discovering new catalysts

A more direct path is the computational discovery of improved catalysts for nitrogen reduction and ammonia synthesis. The fertilizer industry seeks catalysts and process designs that can reduce temperature, pressure, hydrogen-production emissions or total energy intensity. Quantum computers could eventually contribute to strongly correlated catalyst models that are expensive for classical electronic-structure methods (Figure 177).

FTQC One study considered homogeneous nitrogen-fixation catalyst discovery and estimated fault-tolerant resources for several active spaces [1782]. For its highest-utility example, the reported double-factorized phase-estimation runtime was about 139,000 QPU-hours under the paper’s conservative superconducting assumptions, comparable to roughly 16 years of one continuously running QPU before parallelism or architectural improvements. The paper therefore supports feasibility research, not a near-term catalyst-discovery workflow.

Other routes target the dominant upstream burden directly, for example by producing H₂ from low-carbon electricity or by reducing emissions from fossil-based hydrogen production [2555].

There are many routes to decarbonize ammonia production that can be deployed before fault-tolerant quantum computers reach the resources needed for large FeMoCo or catalyst simulations. These include low-carbon hydrogen, carbon capture for fossil-based hydrogen, process integration and more efficient fertilizer use. Ammonia is also being studied as a low-carbon energy carrier, particularly for shipping, although its climate benefit depends strongly on how the hydrogen and ammonia are produced [2539].

3.3.4 Cement production

Cement production is also responsible for substantial CO₂ emissions, but it is less prominent in quantum-

computing use-case literature. A plausible quantum-computing contribution would be electronic-structure or materials modeling of novel clinker phases, binders or additives that reduce process emissions or embodied carbon. This remains more speculative than the quantum-chemistry use cases discussed above, and any claimed advantage would need comparison with mature classical atomistic and materials-simulation methods.

3.3.5 Material sciences and design

Materials science studies the structure, properties, processing, performance and applications of materials. It covers mechanical, thermal, electrical, magnetic, optical and chemical behavior in metals, ceramics, polymers, composites, semiconductors and superconductors. Applications span aerospace, automotive engineering, construction and energy technologies, including photovoltaics, wind-turbine materials and advanced batteries.

Environmental applications include materials for pollution control, waste management and water purification [2557]. Materials science is also central to the development of quantum technologies themselves.

New materials research can use Fermi-Hubbard and related lattice models as simplified descriptions of strongly correlated electrons, notably in regimes relevant to unconventional superconductivity. These models are not generic models of all crystals, semiconductors or superconductors. Difficult doped two-dimensional Hubbard regimes can be challenging for classical methods [2558]. A VQE-based experiment on Google Sycamore used only 16 qubits and was classically reproducible [2559]. **NISQ** In 2025, an IBM experiment studied Fermi-Hubbard dynamics with more than 100 physical qubits and reported a “quantum utility” regime [2560], which should be distinguished from a demonstrated end-to-end computational advantage over the best classical methods.

Quantum algorithms have also been proposed for representative volume element (RVE) problems in computational homogenization. A 2024 peer-reviewed formulation reports polylogarithmic dependence on the microscopic discretization size for its quantum subroutines compared with polynomial classical scaling [2561]. This asymptotic statement depends on efficient quantum state preparation, implementation of the required polynomial/QFT operations and extracting only suitably compressed observables. It should therefore be presented as an algorithmic complexity result rather than an established end-to-end exponential advantage. RVE methods predict macroscopic properties of heterogeneous materials from microscopic structure and are used for composites, polycrystals, porous materials, damage, multiscale modeling and thermal analysis [2562].

NISQ Panasonic and Pasqal are studying hybrid QML, QAOA and classical approaches for organic-

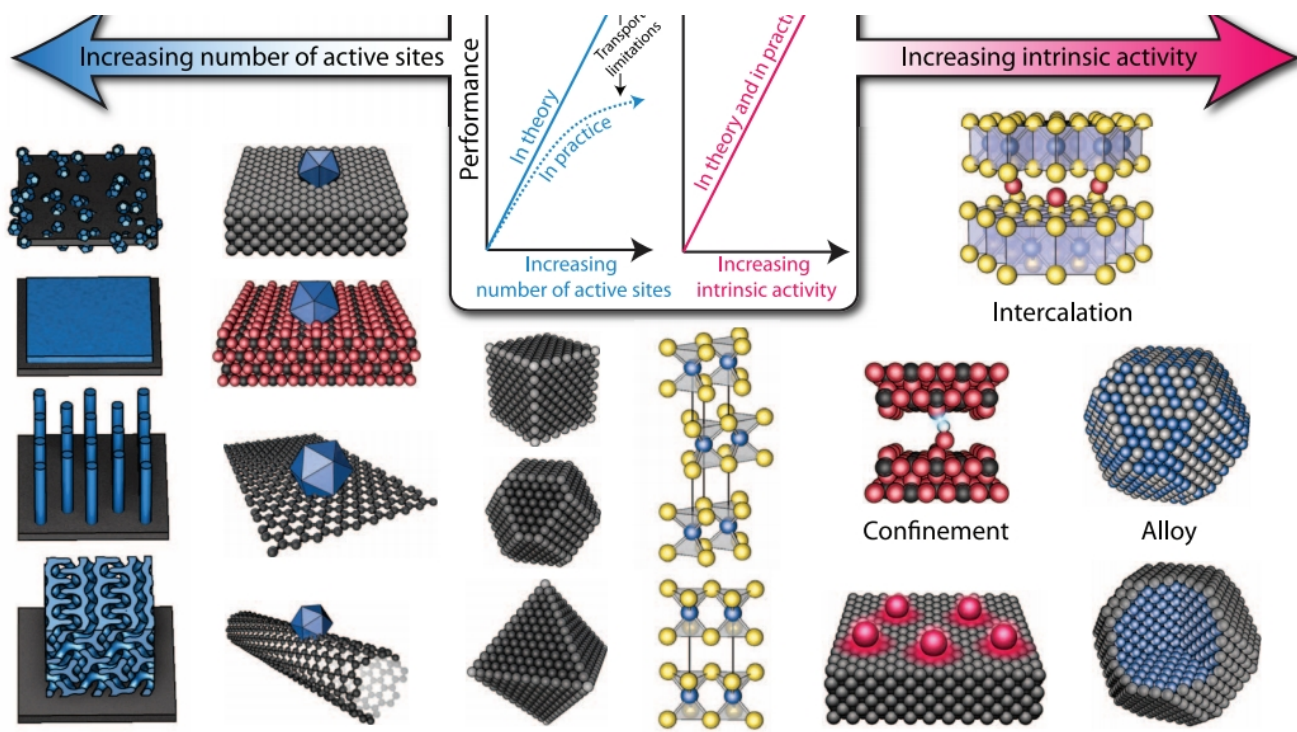


Figure 177: Quantum computing may be useful to simulate new catalysts. Source: [2556] which shows the general computational needs related to these use cases.

semiconductor design. Demonstrations with 22 qubits do not establish quantum advantage [2563]. A KAIST team reported a VQE-based workflow for multicomponent porous materials using IBM hardware, but the active quantum calculation used only 12 qubits. The paper discusses future requirements of roughly 100 to 400 logical qubits [2337, 2564].

3.3.6 Carbon capture

Carbon capture is another potential application domain. Quantum-computing studies usually target the electronic structure and binding chemistry of sorbents such as amines, chelates and metal-organic frameworks rather than a single generic “biomimetic” mechanism.

NISQ BASF has explored electronic-structure calculations for iron-based chelate complexes, first with classical HPC and then with small quantum demonstrations. A 2024 VQE study used 10 trapped-ion qubits for a five-orbital active space, far from a demonstrated advantage regime [2565]. Metal chelates bind a metal ion through multidentate ligands containing donor atoms such as oxygen or nitrogen. In 2026, BASF-related work also explored Fe(III)-NTA with SQD and Krylov-style diagonalization plus substantial classical post-processing [2566].

NISQ BASF announced a partnership with **SEEQC** in 2023 to explore quantum computing for industrial chemical reactions [2567]. The scientific point is that machines in the few-dozen-noisy-qubit regime cannot by qubit count alone be expected to solve chemically

relevant electronic-structure problems beyond classical methods. BASF has also explored quantum applications with IBM, Pasqal and Zapata [2568].

analog Since 2017, **Dow Chemical** has collaborated with the Canadian quantum-software company **1QBit** on molecular-design and optimization studies, including work using D-Wave annealers.

NISQ **TotalEnergies** has worked with CQC, now part of Quantinuum, on VQE-based modeling of carbon capture in metal-organic frameworks. Demonstrations using up to 16 qubits do not establish quantum computational advantage [2569, 2570]. TotalEnergies has also worked with Capgemini in this domain [2571].

NISQ Researchers from the **National Energy Technology Laboratory** and the **University of Kentucky** tested quantum algorithms for amine-related carbon-capture chemistry, including vibrational energies and simplified reaction pathways. Their demonstrations used approximately 8 to 30 qubits with error mitigation, so no computational quantum advantage was established [2572]. **Capgemini** is also working on carbon-capture applications with DARPA funding [2573].

NISQ In 2025, **QC Ware** and **POSCO Holdings** studied CO₂ adsorption in Fe-MOF-74 using active-space reduction and variational/sample-based methods [2574]. Their largest reported quantum instances used up to 28 qubits. A four-layer quantum-number-preserving ansatz reached roughly 4,200 gate or idle operations, illustrating the depth pressure on present noisy devices. This is a useful resource datapoint, but

describing it as a universal “edge” of error mitigation would be too hardware- and technique-dependent.

3.3.7 Batteries

Battery research seeks improvements in energy density, charging rate, cycle life, safety and cost [2575, 2576]. Atomistic and electronic-structure simulations are used to understand reactions at electrodes and electrolytes, intercalation in crystalline materials, degradation mechanisms and candidate chemistries.

Several companies are pursuing battery-related quantum chemistry. **IBM** and **Mitsubishi Chemical** simulated early lithium-oxygen chemistry on superconducting hardware [2577]. In 2025, IBM researchers extended this direction to oxygen-reduction reactions at lithium electrode surfaces using local embedding and SQD with active spaces up to roughly 76 qubits [2578]. The method improved on CCSD for some embedded active-space energies and was validated where possible against CASCI or HCI. This is evidence of improved accuracy over one classical approximation, not of quantum computational advantage over the best available classical workflow. **NISQ**

NISQ **Ford** worked with **Quantinuum** on chemical modeling of EV battery materials [2579]. Their 2022 study used VQE-style methods for LiCoO₂-related gas-phase models such as Li₂Co₂O₄ and Co₂O₄. The reported calculations used a 20-qubit state-vector emulator representing an active space of 10 electrons in 20 spin orbitals. This is therefore a small algorithmic benchmark, not a hardware demonstration or quantum-advantage result.

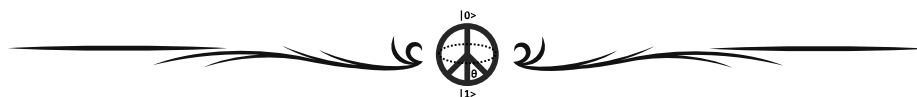
FTQC **Mercedes-Benz** worked with IBM on lithium-sulfur battery chemistry [2580] and later with PsiQuantum

on fault-tolerant resource estimates for electrolyte molecules. Some studied instances require of order 10⁴ logical qubits [2581]. Converting this logical count to a fixed 160 million physical qubits by assuming exactly 10,000 physical qubits per logical qubit is not robust: the physical-to-logical overhead depends strongly on the error-correcting code, physical error rate, target logical error rate, algorithm size and architecture. The logical resource estimate is the more portable number.

FTQC **Hyundai** and **IonQ** have explored quantum algorithms for small lithium-oxide models using trapped-ion hardware [2582]. These experiments involve very small molecules and do not demonstrate quantum advantage. Claims that an industrially useful version would require around 2,000 logical qubits should be read as prospective resource targets under particular modeling assumptions, not as a universal threshold or a reliable date prediction for a vendor roadmap.

FTQC **Volkswagen** has also studied lithium-ion battery simulations with 1QBit and Xanadu. One fault-tolerant resource study estimated 2,375 to 6,652 logical qubits for Li₂FeSiO₄ models containing 156 electrons [2583]. A 2024 study proposed time-domain X-ray absorption spectroscopy with about 900 logical qubits and 10⁸ Toffoli gates per circuit sample [2584]. In 2026, Xanadu proposed a QPE/QSP method for resonant inelastic X-ray scattering of battery cathode models. For one 20-orbital active space, the estimate was about 2.0 × 10¹⁰ Toffoli gates and 414 logical qubits [2585].

Finally, quantum sensing may also contribute to battery R&D and manufacturing diagnostics. **AMTE Power** (UK) led the three-year Project Quantum, funded with £5.4M from Innovate UK, to investigate quantum-sensing methods for lithium-based battery manufacturing. The project included NV-center magnetic sensing to characterize battery processes and aging [2586].



3.3.8 Bibliography and notes

These are the bibliographical references and notes for the **Chemistry and materials** subsection.

- [228] [Elucidating Reaction Mechanisms on Quantum Computers](#), by Markus Reiher, Nathan Wiebe, Krysta M Svore, Dave Wecker, and Matthias Troyer, arXiv, May 2016 (28 pages).
- [230] [Even More Efficient Quantum Computations of Chemistry Through Tensor Hypercontraction](#), by Joonho Lee et al., PRX Quantum, July 2021 (62 pages).
- [231] [Fast Quantum Simulation of Electronic Structure by Spectral Amplification](#), by Guang Hao Low, Robbie King, Dominic W. Berry, Qiushi Han, A. Eugene DePrince III, Alec F. White, Ryan Babbush, Rolando D. Somma, and Nicholas C. Rubin, Physical Review X, October 2025 (41 pages).
- [233] [Classical solution of the FeMo-cofactor model to chemical accuracy and its implications](#), by Huanchen Zhai, Chenghan Li, Xing Zhang, Zhendong Li, Seunghoon Lee, and Garnet Kin-Lic Chan, arXiv, January 2026 (89 pages).
- [833] [An adaptive variational algorithm for exact molecular simulations on a quantum computer](#), by Harper R. Grimsley, Sophia E. Economou, Edwin Barnes, and Nicholas J. Mayhall, Nature Communications, July 2019.
- [918] [Quantum Chemistry in the Age of Quantum Computing](#), by Yudong Cao, Alán Aspuru-Guzik, et al., arXiv, December 2018 (194 pages).
- [968] [Calculating the ground state energy of benzene under spatial deformations with noisy quantum computing](#), by Wasil Sennane, Jean-Philip Piquemal, and Marko J. Rančić, arXiv, November 2022 (14 pages).
- [990] [How will quantum computers provide an industrially relevant computational advantage in quantum chemistry?](#), by V.E. Elfving, A. Bochevarov, et al., arXiv, September 2020 (20 pages).
- [999] [A self-consistent field approach for the variational quantum eigensolver: orbital optimization goes adaptive](#), by Aaron Fitzpatrick, Stefan Knecht, et al., arXiv, December 2022 (21 pages).

- [1782] [Feasibility of accelerating homogeneous catalyst discovery with fault-tolerant quantum computers](#), by Nicole Bellonzi, Alán Aspuru-Guzik, et al., arXiv, June 2024 (27 pages).
- [1796] [Quantum Computing for Corrosion-Resistant Materials and Anti-Corrosive Coatings Design](#), by Nam Nguyen, Christopher N. Beall, Thomas S. Humble, and Nicholas P. Lubbers, arXiv, June 2024 (50 pages).
- [2331] [Quantum-Classical Computational Molecular Design of Deuterated High-Efficiency OLED Emitters](#), by Qi Gao et al., June 2023 (10 pages).
- [2337] [Quantum Computing Based Design of Multivariate Porous Materials](#), by Shinyoung Kang, Younghun Kim, and Jihan Kim, ACS Central Science, August 2025 (12 pages).
- [2398] [Quantum hardware calculations of periodic systems with partition-measurement symmetry verification: simplified models of hydrogen chain and iron crystals](#), by Kentaro Yamamoto, David Muñoz Ramo, et al., arXiv, May 2022 (15 pages).
- [2399] [Enabling the quantum leap Quantum algorithms for chemistry and materials Report](#), by NSF, January 2019 (115 pages).
- [2400] [Quantum Computing: Fundamentals, Trends and Perspectives for Chemical and Biochemical Engineers](#), by Amirhossein Nourbakhsh, Seyed Soheil Mansouri, et al., arXiv, January 2022 (28 pages).
- [2401] [Quantum Navier-Stokes equations](#), by Pina Milisic, Progress in Industrial Mathematics at ECMI 2010 Conference papers, January 2012 (12 pages) .
- [2402] [Beyond Unitary Quantum Simulation: Open-System Approaches to Quantum Chemistry toward Quantum Advantage](#), by Michael Marthaler, Elias Zapusek, and Florentin Reiter, arXiv, May 2026 (9 pages).
- [2403] [Quantum Algorithms and Applications for Open Quantum Systems](#), by Luis H. Delgado-Granados, David A. Mazziotti, et al., arXiv, June 2024 (47 pages).
- [2404] [Quantum-centric supercomputing for materials science: A perspective on challenges and future directions](#), by Yuri Alexeev, Liang Jiang, et al., Algorithmiq, December 2023 (45 pages).
- [2405] [Spiers Memorial Lecture: Quantum chemistry, classical heuristics, and quantum advantage](#), by Garnet Kin-Lic Chan, Faraday Discussions, July 2024 (42 pages).
- [2406] [Quantum Algorithm for Estimating Gibbs Free Energy and Entropy via Energy Derivatives](#), by Shangjie Guo, Corneliu Buda, and Nathan Wiebe, arXiv, November 2025 (15 pages).
- [2407] [Fullqubit alchemist: Quantum algorithm for alchemical free energy calculations](#), by Po-Wei Huang, Bálint Koczor, et al., arXiv, August 2025 (26 pages).
- [2408] [Capturing Protein Free Energy Landscape using Efficient Quantum Encoding](#), by Ashwini Kannan, Sanjib Senapati, et al., arXiv, October 2025 (20 pages).
- [2409] [Implicit Solvent Sample-Based Quantum Diagonalization](#), by Danil Kaliakin, Fangchun Liang, et al., The Journal of Physical Chemistry B, May 2025 (9 pages).
- [2410] [Fault-tolerant simulation of the electronic structure using Projector Augmented-Waves and Bloch orbitals](#), by Rishabh Bhardwaj, Alexander Reed Muñoz, Travis E. Jones, and John Golden, arXiv, April 2026 (14 pages).
- [2411] [Probing Non-Equilibrium Topological Order on a Quantum Processor](#), by M. Will, F. Pollmann, et al., Nature, September 2025 (21 pages).
- [2412] [Quantum Computing Beyond Ground State Electronic Structure: A Review of Progress Toward Quantum Chemistry Out of the Ground State](#), by Alan Bidart, Prateek Vaish, Tilas Kabengele, Yaoqi Pang, Yuan Liu, and Brenda M. Rubenstein, arXiv, September 2025 (45 pages).
- [2413] [Implementation Possibility of Quantum Simulation for Quantum Molecular Dynamics](#), by Xingyu Zhang, Weijia Guo, Jinke Yu, and Qingyong Meng, arXiv, August 2026 (42 pages).
- [2414] [Colloquium: Area laws for the entanglement entropy](#), by J. Eisert, Reviews of Modern Physics, February 2010  (preprint on arXiv).
- [2415] [Quantum linear solvers for quantum chemistry: prospects of exponential quantum advantage](#), by Peniel Bertrand Tsemo, Kenji Sugisaki, Ishita Bhattacharjee, and V. S. Prasanna, arXiv, July 2026 (32 pages).
- [2416] [End-to-End Quantum Simulation of a Chemical System](#), by Wim van Dam, Krysta Svore, Matthias Troyer, et al., arXiv, September 2024 (15 pages).
- [2417] [The Hartree-Fock method: from self-consistency to correct symmetry](#), by Jean-Philippe Martinez, Annalen der Physik, January 2017.
- [2418] [Two-dimensional materials from high-throughput computational exfoliation of experimentally known compounds](#), by Nicolas Mounet, Marco Gibertini, Philippe Schwaller, Davide Campi, Andrius Merkys, Antimo Marrazzo, Thibault Sohier, Ivano Eligio Castelli, Andrea Cepellotti, Giovanni Pizzi, and Nicola Marzari, Nature Nanotechnology, February 2018  (preprint on arXiv).
- [2419] [Distributed Implementation of Full Configuration Interaction for One Trillion Determinants](#), by Hong Gao, Satoshi Imamura, Akihiko Kasagi, and Eiji Yoshida, Journal of Chemical Theory and Computation, February 2024 (8 pages).
- [2420] [Quantum Computing for Molecular Biology](#), by Alberto Baiardi, Matthias Christandl, and Markus Reiher, arXiv, June 2023 (76 pages).
- [2421] [Quantum Information reveals that orbital-wise correlation is essentially classical in Natural Orbitals](#), by Davide Matrià, Leonardo Guidoni, et al., arXiv, April 2024 (18 pages).
- [2422] [Evaluating higher-order product formulae for molecular ground-state energy estimation](#), by Hiromu Abe, Keita Kanno, Ryosuke Kimura, Masahiko Kamoshita, and Kosuke Mitarai, arXiv, May 2026 (12 pages).
- [2423] [Efficient simulation of quantum chemistry problems in an enlarged basis set](#), by Maxine Luo and J. Ignacio Cirac, arXiv, February 2025 (12 pages).
- [2424] [Practicality of quantum adiabatic algorithm for chemistry applications](#), by Etienne Granet, Khaldoun Ghanem, and Henrik Dreyer, arXiv, July 2024 (17 pages).
- [2425] [Drug design on quantum computers](#), by Raffaele Santagati, Alan Aspuru-Guzik, Ryan Babbush, Matthias Degroote, Leticia Gonzalez, Elica Kyoseva, Nikolaj Moll, Markus Oppel, Robert M. Parrish, Nicholas C. Rubin, Michael Streif, Christofer S. Tautermann, Horst Weiss, Nathan Wiebe, and Clemens Utschig-Utschig, Nature Physics, March 2024 (9 pages)  (preprint on arXiv).
- [2426] [Which options exist for NISQ-friendly linear response formulations?](#), by Karl Michael Ziem, Sonia Coriani, et al., arXiv, December 2023 (50 pages).
- [2427] [Understanding and mitigating noise in molecular quantum linear response for spectroscopic properties on quantum computers](#), by Karl Michael Ziem, Sonia Coriani, et al., arXiv, August 2024 (54 pages).
- [2428] [Microcanonical and finite-temperature ab initio molecular dynamics simulations on quantum computers](#), by Igor O. Sokolov, Panagiotis Kl. Barkoutsos, Lukas Moeller, Philippe Suchsland, Guglielmo Mazzola, and Ivano Tavernelli, Physical Review Research, February 2021 (17 pages).
- [2429] [Modern Software for Computer Modeling in Quantum Chemistry and Molecular Dynamics](#), by Marina V. Malysheva et al., 2021 (11 pages).
- [2430] [Molecular Docking with Quantum Circuit Evolution](#), by Gleydson Fernandes de Jesus, Bruno Oziel Fernandez, and Marcelo A. Moret, arXiv, July 2026 (6 pages).
- [2431] [Shortcut to chemically accurate quantum computing via density-based basis-set correction](#), by Diata Traore, Olivier

- Adjoua, César Feniou, Ioanna-Maria Lygatsika, Yvon Ma-day, Evgeny Posenitskiy, Kerstin Hammernik, Alberto Peruzzo, Julien Toulouse, Emmanuel Giner, and Jean-Philip Piquemal, *Communications Chemistry*, November 2024.
- [2432] [Gaussian basis sets for use in correlated molecular calculations. I. The atoms boron through neon and hydrogen](#), by Thom H. Dunning Jr., *The Journal of Chemical Physics*, January 1989 (17 pages) .
- [2433] [Basis Set Exchange](#), by Basis Set Exchange, 2020.
- [2434] [Fault-tolerant resource estimate for quantum chemical simulations: Case study on Li-ion battery electrolyte molecules](#), by Isaac H. Kim, Sam Roberts, et al., *arXiv*, May 2023 (26 pages).
- [2435] [A Perspective on Quantum Computing Applications in Quantum Chemistry using 25–100 Logical Qubits](#), by Yuri Alexeev, Linghua Zhu, et al., *arXiv*, June 2025 (21 pages).
- [2436] [An Introduction to Variational Quantum Eigensolver Applied to Chemistry](#), by Matheus da S. Fonseca, Celso J. Villas-Boas, et al., *arXiv*, June 2025 (39 pages).
- [2437] [Benchmarking the Variational Quantum Eigensolver through Simulation of the Ground State Energy of Prebiotic Molecules on High-Performance Computers](#), by P. Lolur, G. Wendin, et al., *arXiv*, January 2021 (9 pages).
- [2438] [Reducing the Resources Required by ADAPT-VQE Using Coupled Exchange Operators and Improved Subroutines](#), by Mafalda Ramôa, Sophia E. Economou, et al., *arXiv*, May 2025 (31 pages).
- [2439] [Shallow Quantum Circuits for Deep Chemistry via Valence Bond Embeddings](#), by Francisco Javier del Arco Santos and Jakob S. Kottmann, *arXiv*, June 2026 (11 pages).
- [2440] [Hybrid Quantum Algorithms for Computational Chemistry: Application to the Pyridine-Li ion Complex](#), by Fatemeh Ghasemi, Yousung Kang, Yukio Kawashima, and Kyungsun Moon, *arXiv*, January 2026 (23 pages).
- [2441] [Quantifying the effect of gate errors on variational quantum eigensolvers for quantum chemistry](#), by Kieran Dalton, David R. M. Arvidsson-Shukur, et al., *arXiv*, February 2024 (25 pages).
- [2442] [Accelerating Parameter Initialization in Quantum Chemical Simulations via LSTM-FC-VQE](#), by Ran-Yu Chang, En-Jui Kuo, et al., *arXiv*, May 2025 (8 pages).
- [2443] [Automated near-term quantum algorithm discovery for molecular ground states](#), by Fabian Finger, Konstantinos Meichanetzidis, et al., *arXiv*, March 2026 (17 pages).
- [2444] [Quantum-centric machine learning for molecular dynamics](#), by Yanxian Tao, Lingyun Wan, Xiongzhi Zeng, Yingdi Jin, Jie Liu, Zhenyu Li, and Jinlong Yang, *arXiv*, November 2025 (34 pages).
- [2445] [Classical reservoir approach for efficient molecular ground state preparation](#), by Zekun He, Dominika Zgid, A. F. Kemper, and J. K. Freericks, *arXiv*, December 2025 (9 pages).
- [2446] [Quantum computing quantum Monte Carlo with hybrid tensor network for electronic structure calculations](#), by Shu Kanno, Qi Gao, et al., *arXiv*, June 2024 (32 pages).
- [2447] [Quantum-enhanced quantum Monte Carlo: an industrial view](#), by Maximilian Amsler, Christopher Wever, et al., *arXiv*, January 2023 (16 pages).
- [2448] [Ab Initio Transcorrelated Method enabling accurate Quantum Chemistry on near-term Quantum Hardware](#), by Werner Dobrautz, Ivano Tavernelli, et al., *arXiv*, April 2024 (19 pages).
- [2449] [Experimental quantum computational chemistry with optimised unitary coupled cluster ansatz](#), by Shaojun Guo, Jian-Wei Pan, et al., *arXiv*, June 2024 (11 pages).
- [2450] [Quantum computation of reactions on surfaces using local embedding](#), by Tanvi P. Gujarati, Kristen Williams, et al., *npj Quantum Information*, September 2023.
- [2451] [Chemistry Beyond the Scale of Exact Diagonalization on a Quantum-Centric Supercomputer](#), by Javier Robledo Moreno, Antonio Mezzacapo, et al., *arXiv*, July 2025 (65 pages).
- [2452] [Quantum-Selected Configuration Interaction: classical diagonalization of Hamiltonians in subspaces selected by quantum computers](#), by Keita Kanno, Masaya Kohda, Ryosuke Imai, Sho Koh, Kosuke Mitarai, Wataru Mizukami, and Yuya O. Nakagawa, *arXiv*, February 2023 (28 pages).
- [2453] [Chemistry beyond the scale of exact diagonalization on a quantum-centric supercomputer](#), by Javier Robledo-Moreno, Simon Martiel, et al., *Science Advances*, June 2025 (14 pages).
- [2454] [Quantum chemistry with provable convergence via randomized sample-based Krylov quantum diagonalization](#), by Samuele Piccinelli, Ivano Tavernelli, et al., *arXiv*, January 2026 (22 pages).
- [2455] [Orthogonal Quantum Krylov Diagonalisation](#), by Hadi Rammal, Alexandre Perrin, Oumaya Ladhari, Clément Dutreix, Jérémie Messud, and Matthieu Saubaniere, *arXiv*, July 2026 (16 pages).
- [2456] [Bridging physical intuition and hardware efficiency for correlated electronic states: the local unitary cluster Jastrow ansatz for electronic structure†](#), by Mario Motta et al., *Chemical Science*, September 2023.
- [2457] [A Quantum Multi-Programming Framework to Maximize Quantum Resources for the LUCJ Ansatz](#), by Milana Baza-yeva, Abigail McClain Gomez, and Kenneth M. Merz Jr, *arXiv*, May 2026 (25 pages).
- [2458] [Systematic Improvability in Quantum Embedding for Real Materials](#), by Akhil Shajan et al., *Physical Review X*, June 2026.
- [2459] [Molecular Quantum Computations on a Protein](#), by Akhil Shajan, Danil Kaliakin, Fangchun Liang, Thaddeus Pellegrini, Hakan Doga, Subhamoy Bhowmik, Susanta Das, Antonio Mezzacapo, Mario Motta, and Kenneth M. Merz Jr, *arXiv*, December 2026 (40 pages).
- [2460] [Localized Quantum Chemistry on Quantum Computers](#), by Matthew Otten, 2022 (13 pages).
- [2461] [Fault-tolerant quantum computation of molecular observables](#), by Mark Steudtner, Michael Streif, et al., *arXiv*, October 2023 (36 pages).
- [2462] [Fault-tolerant quantum simulation of materials using Bloch orbitals](#), by Nicholas C. Rubin et al., *PRX Quantum*, October 2023 (52 pages).
- [2463] [A Transferable Machine Learning Approach to Predict Optimized Orbitals for Electronic Structure Problems](#), by Lucas van der Horst, Maniraman Periyasamy, Abhishek Y. Dubey, Davide Bincoletto, Jakob S. Kottmann, and Daniel D. Scherer, *arXiv*, May 2026 (13 pages).
- [2464] [Simulating near-infrared spectroscopy on a quantum computer for enhanced chemical detection](#), by Ignacio Loaiza, Juan Miguel Arrazola, et al., *arXiv*, April 2025 (28 pages).
- [2465] [Early Fault-Tolerant Quantum Algorithms in Practice: Application to Ground-State Energy Estimation](#), by Oriol Kiss, Juan Miguel Arrazola, et al., *arXiv*, March 2025 (22 pages).
- [2466] [Nonlinear Spectroscopy via Generalized Quantum Phase Estimation](#), by Ignacio Loaiza, Juan Miguel Arrazola, et al., *arXiv*, April 2025 (19 pages).
- [2467] [Universal Quantum Computational Spectroscopy on a Quantum Chip](#), by Chonghao Zhai, Jianwei Wang, et al., *arXiv*, June 2025 (11 pages).
- [2468] [Any type of spectroscopy can be efficiently simulated on a quantum computer](#), by Liam P. Flew and Ivan Kassal, *arXiv*, October 2025 (10 pages).
- [2469] [The cost of quantum algorithms for biochemistry: A case study in metaphosphate hydrolysis](#), by Ryan LaRose, Norm M. Tubman, et al., *arXiv*, February 2026 (15 pages).
- [2470] [Simulating electronic structure on bosonic quantum computers](#), by Rishab Dutta, Victor S. Batista, et al., *arXiv*, February 2025 (83 pages).

- [2471] [Simulating Chemistry on Bosonic Quantum Devices](#), by Rishab Dutta, Victor S. Batista, et al., arXiv, July 2024 (40 pages).
- [2472] [Boson sampling enhanced quantum chemistry](#), by Zhong-Xia Shang, Jian-Wei Pan, Ming-Cheng Chen, et al., arXiv, April 2024 (15 pages).
- [2473] [Near- and long-term quantum algorithmic approaches for vibrational spectroscopy](#), by Nicolas P. D. Sawaya, Francesco Paesani, and Daniel P. Tabor, arXiv, February 2021 (20 pages).
- [2474] [Optimizing the number of measurements for vibrational structure on quantum computers: coordinates and measurement schemes](#), by Marco Majland, Ove Christiansen, et al., arXiv, September 2024 (11 pages).
- [2475] [Vibrational ADAPT-VQE: Critical points leads to problematic convergence](#), by Marco Majland, Ove Christiansen, et al., arXiv, May 2025 (13 pages).
- [2476] [Accurate harmonic vibrational frequencies for diatomic molecules via quantum computing](#), by Shih-Kai Chou, Hsi-Sheng Goan, et al., arXiv, December 2023 (17 pages).
- [2477] [Fault-tolerant quantum computations of vibrational wave functions](#), by Marco Majland, Ove Christiansen, et al., arXiv, August 2025 (25 pages).
- [2478] [A hybrid quantum-classical algorithm for multichannel quantum scattering of atoms and molecules](#), by Xiaodong Xing, Timur V. Tscherbul, et al., arXiv, April 2023 (8 pages).
- [2479] [A real neural network state for quantum chemistry](#), by Yangjun Wu, Honghui Shang, et al., arXiv, January 2023 (6 pages).
- [2480] [Quantum machine learning for chemistry and physics](#), by Manas Sajjan, March 2022 (99 pages).
- [2481] [Evaluating the evidence for exponential quantum advantage in ground-state quantum chemistry](#), by Seunghoon Lee, Ryan Babbush, John Preskill, David R. Reichman, Earl T. Campbell, Edward F. Valeev, Lin Lin, Garnet Kin-Lic Chan, et al., Nature Communications, April 2023 (7 pages).
- [2482] [Quantum Advantage in Computational Chemistry?](#), by Hans Gundlach, Keeper Sharkey, Jayson Lynch, Victoria Hazoglou, Kung-Chuan Hsu, Carl Dukat, Eleanor Crane, Karin Walczyk, Marcin Bodziak, Johannes Galatsanos-Dueck, and Neil Thompson, 2025 IEEE International Conference on Quantum Computing and Engineering (QCE), September 2025 (11 pages) (preprint on [arXiv](#)).
- [2483] [Putting fermions onto a digital quantum computer](#), by Riley W. Chien, James D. Whitfield, et al., arXiv, February 2026 (20 pages).
- [2484] [Low-depth simulations of fermionic systems on square-grid quantum hardware](#), by Manuel G. Algaba, Fedor Šimkovic IV, et al., arXiv, April 2024 (36 pages).
- [2485] [Walking through Hilbert Space with Quantum Computers](#), by Tong Jiang, Joonho Lee, et al., arXiv, February 2025 (40 pages).
- [2486] [Monte Carlo particle transport on quantum computers](#), by Noé Olivier and Michel Nowak, arXiv, October 2024 (7 pages).
- [2487] [Polynomially efficient quantum enabled variational Monte Carlo for training neural-network quantum states for physico-chemical applications](#), by Manas Sajjan, Vinit Singh, and Sabre Kais, arXiv, December 2024 (35 pages).
- [2488] [Quantum-assisted variational Monte Carlo](#), by Longfei Chang, Zhenlong Li, and Wei-Hai Fang, arXiv, June 2025 (37 pages).
- [2489] [Contextual Subspace Auxiliary-Field Quantum Monte Carlo: Improved bias with reduced quantum resources](#), by Matthew Kiser, Matthias Beuerle, and Fedor Simkovic IV, arXiv, September 2024 (16 pages).
- [2490] [Framework for robust quantum speedups in practical correlated electronic structure and dynamics](#), by Jielun Chen and Garnet Kin-Lic Chan, Physical Review Letters, September 2026 (preprint on [arXiv](#)).
- [2491] [Open source variational quantum eigensolver extension of the quantum learning machine for quantum chemistry](#), by Mohammad Haidar et al., March 2023 (26 pages).
- [2492] [TenCirChem: An Efficient Quantum Computational Chemistry Package for the NISQ Era](#), by Weitang Li, Shengyu Zhang, et al., arXiv, June 2023 (76 pages).
- [2493] [ChemIQ: A Chemistry Simulator for Quantum Computer](#), by Qingchun Wang, Guo-Ping Guo, et al., arXiv, December 2022 (7 pages).
- [2494] [Quantinuum launches InQuanto, a state-of-the-art quantum computational chemistry software platform using quantum computers](#), by Quantinuum, 2022.
- [2495] [Efficient algorithms for quantum chemistry on modular quantum processors](#), by Tian Xue, Jacob P. Covey, and Matthew Otten, arXiv, June 2025 (11 pages).
- [2496] [Distribution Complexity of Electronic Structure Simulations on Quantum Supercomputers](#), by Jason Necaie, Nimit Anand, Gaurav Gyawali, K. Grace Johnson, James D. Whitfield, and Masoud Mohseni, arXiv, June 2026 (21 pages).
- [2497] [Is there evidence for exponential quantum advantage in quantum chemistry?](#), by Seunghoon Lee, arXiv, August 2022 (81 pages).
- [2498] [Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets](#), by Abhinav Kandala, Antonio Mezzacapo, Kristan Temme, Maika Takita, Markus Brink, Jerry M. Chow, and Jay M. Gambetta, Nature, September 2017 (5 pages) (preprint on [arXiv](#)).
- [2499] [Open Source Variational Quantum Eigensolver Extension of the Quantum Learning Machine \(QLM\) for Quantum Chemistry](#), by Mohammad Haidar, Marko J. Rančić, Thomas Ayrat, Yvon Maday, and Jean-Philip Piquemal, arXiv, November 2022 (26 pages).
- [2500] [Exploring pathways towards quantum advantage in quantum chemistry: the case of a molecule with half-Möbius topology](#), by Samuele Piccinelli, Ivano Tavernelli, et al., arXiv, March 2026 (4 pages).
- [2501] [A molecule with half-Möbius topology](#), by Igor Rončević, Leo Gross, et al., Science, March 2026 (preprint on [arXiv](#)).
- [2502] [Finding Optimal Pathways in Chemical Reaction Networks Using Ising Machines](#), by Yuta Mizuno and Tamiki Komatsuzaki, arXiv, December 2023 (27 pages).
- [2503] [Virtual Screening of Chemical Space based on Quantum Annealing](#), by Takuro Tanaka, Masayuki Ohzeki, et al., arXiv, July 2023 (11 pages).
- [2504] [Correction Scheme for Molecular Total Energies From Quantum Phase Estimation Under Limited Qubit Resources](#), by Nobuki Inoue and Hisao Nakamura, Journal of Computational Chemistry, May 2026 (10 pages).
- [2505] [Excited-state molecular dynamics simulation based on variational quantum algorithms](#), by Hirotooshi Hirai, arXiv, November 2022 (25 pages).
- [2506] [Accurate Chemical Reaction Modeling on Noisy Intermediate-Scale Quantum Computers Using a Noise-Resilient Wavefunction Ansatz](#), by Xiongzi Zeng, Jinlong Yang, et al., arXiv, April 2024 (23 pages).
- [2507] [Chemically Motivated Simulation Problems are Efficiently Solvable by a Quantum Computer](#), by Philipp Schleich, Joe Fitzsimons, Alán Aspuru-Guzik, et al., arXiv, April 2025 (32 pages).
- [2508] [Simulation of Chemical Reactions on a Quantum Computer](#), by Sumit Suresh Kale and Sabre Kais, arXiv, April 2024 (8 pages).
- [2509] [Modeling Stochastic Chemical Kinetics on Quantum Computers](#), by Tilas Kabengele, Yash M. Lokare, J. B. Marston, and Brenda M. Rubenstein, arXiv, April 2024 (22 pages).

- [2510] [Quantum-classical crossover in fault-tolerant quantum dynamics simulation](#), by Jinzhao Sun, Pan Zhang, Xiao Yuan, et al., arXiv, July 2026 (45 pages).
- [2511] [Feasibility of first principles molecular dynamics in fault-tolerant quantum computer by quantum phase estimation](#), by Ichio Kikuchi and Akihito Kikuchi, arXiv, April 2025 (22 pages).
- [2512] [Experimental Quantum Simulation of Chemical Dynamics](#), by Tomas Navickas et al., Journal of the American Chemical Society, May 2025  (preprint on dora.lib4ri.ch).
- [2513] [Quantum dynamics in frustrated Ising fullerenes](#), by Alejandro Lopez-Bezanilla, Andrew D. King, et al., arXiv, May 2025 (10 pages).
- [2514] [A comprehensive framework to simulate real-time chemical dynamics on a fault-tolerant quantum computer](#), by Felipe H. da Jornada, Karthik I. Seetharam, et al., arXiv, April 2025 (86 pages).
- [2515] [Crossing the 12,000-atom barrier with heterogeneous quantum-classical supercomputing: quantum chemistry of protein-ligand complexes](#), by Kenneth M. Merz Jr., Mario Motta, et al., arXiv, May 2026 (12 pages).
- [2516] [Closed-loop calculations of electronic structure on a quantum processor and a classical supercomputer at full scale](#), by Tomonori Shirakawa, Antonio Mezzacapo, et al., arXiv, November 2025 (13 pages).
- [2517] [Quantum-Centric Geometry Optimization with Wave-Function-Based Embedding](#), by Danil Kaliakin, Akhil Shajan, Fangchun Liang, Zhen Li, and Kenneth M. Merz Jr, arXiv, July 2026 (32 pages).
- [2518] [Fullerene-encapsulated Cyclic Ozone for the Next Generation of Nano-sized Propellants via Quantum Computation](#), by Thomas W. Watts, Adam Holmes, et al., arXiv, August 2024 (20 pages).
- [2519] [Quantum computing for corrosion-resistant materials and anti-corrosive coatings design](#), by Nam Nguyen, Adam Holmes, et al., arXiv, April 2025 (36 pages).
- [2520] [Quantum Computing in Corrosion Modeling: Bridging Research and Industry](#), by Juan Manuel Aguiar Hualde, Julian van Velzen, Jasper Krauser, Emanuele Marsili, et al., arXiv, February 2025 (40 pages).
- [2521] [Mitsubishi Chemical Group, Deloitte Tohmatsu, and Classiq Compress Quantum Circuits by Up to 97%](#), by Mitsubishi Chemical, Mitsubishi Chemical news, December 2024.
- [2522] [Geo-ADAPT-VQE: Quantum Information Metric-Aware Circuit Optimization for Quantum Chemistry](#), by Mohammad Aamir Sohail and Toshiaki Koike-Akino, arXiv, March 2026 (22 pages).
- [2523] [Sample-based quantum diagonalization approach for open-shell transition-metal complexes in gas and implicit-solvent](#), by David David, Vedangi Pathak, Marek Kowalik, Hamed Mohammadbagherpoor, Vincent Beltrani, Kara Maller, Niall Moroney, and Phalgun Lolur, arXiv, July 2026 (24 pages).
- [2524] [Localized sample-based quantum diagonalization for strongly correlated chemistry](#), by Qiaohong Wang et al., Proceedings of the National Academy of Sciences, June 2026 (12 pages).
- [2525] [Quantum Lattice Boltzmann Solutions for Transport under 3D Spatially Varying Advection on Trapped Ion Hardware](#), by Sayonee Ray, Jezer Jojo, Jason Iaconis, Abeynaya Gnanasekaran, Apurva Tiwari, Martin Roetteler, Chris Hill, and Jay Pathak, arXiv, April 2026 (11 pages).
- [2526] [Local, Expressive, Quantum-Number-Preserving VQE Ansatzes for Fermionic Systems](#), by Gian-Luca R. Anselmetti, David Wierichs, Christian Gogolin, and Robert M. Parrish, arXiv, May 2021 (26 pages).
- [2527] [Analytical Ground- and Excited-State Gradients for Molecular Electronic Structure Theory from Hybrid Quantum/Classical Methods](#), by Robert M. Parrish, Gian-Luca R. Anselmetti, and Christian Gogolin, arXiv, October 2021 (23 pages).
- [2528] [Accelerating Quantum Computations of Chemistry Through Regularized Compressed Double Factorization](#), by Oumarou Oumarou, Christian Gogolin, et al., arXiv, June 2024 (25 pages).
- [2529] [Tailored and Externally Corrected Coupled Cluster with Quantum Inputs](#), by Maximilian Scheurer, Nicholas C. Rubin, et al., arXiv, January 2024 (36 pages).
- [2530] [Quantum simulation of realistic materials in first quantization using non-local pseudopotentials](#), by Dominic W. Berry, Nicholas C. Rubin, Ryan Babbush, et al., npj Quantum Information, December 2024.
- [2531] [End-to-End Simulation of Chemical Dynamics on a Quantum Computer](#), by Elliot C. Eklund, Arkin Tikku, Patrick Sinnott, William J. Huggins, Guang Hao Low, Dominic W. Berry, and Ivan Kassal, arXiv, March 2026 (69 pages).
- [2532] [Efficient Simulation of Pre-Born-Oppenheimer Dynamics on a Quantum Computer](#), by Matthew Pocrnic, Ignacio Loaiza, Juan Miguel Arrazola, Nathan Wiebe, and Danial Motlagh, arXiv, February 2026 (54 pages).
- [2533] [Large circuit execution for NMR spectroscopy simulation on NISQ quantum hardware](#), by Artemiy Burov, Julien Baglio, and Clément Javerzac-Galy, arXiv, December 2025 (17 pages).
- [2534] [Quantum Computing for Electronic Circular Dichroism Spectrum Prediction of Chiral Molecules](#), by Amandeep Singh Bhatia and Sabre Kais, arXiv, February 2026 (18 pages).
- [2535] [Size-Consistent Quantum Chemistry on Quantum Computers](#), by Noah Garrett, Michael Rose, and David A. Mazziotti, arXiv, December 2025 (20 pages).
- [2536] [Symmetry Dilemmas in Quantum Computing for Chemistry: A Comprehensive Analysis](#), by Ilias Magoulas, Muhan Zhang, and Francesco A. Evangelista, arXiv, December 2025 (31 pages).
- [2537] [Digital Quantum Simulation of the quantum-FPUT Lattice: Formulation and Resource Estimation](#), by Kiratholly Nandakumar Madhav Sharma, Juan Manuel Aguiar Hualde, Julian van Velzen, and Phalgun Lolur, arXiv, May 2026 (9 pages).
- [2538] [The global nitrogen cycle in the twenty-first century](#), by David Fowler, The Royal Society Publishing, July 2013 (13 pages)  (preprint on ncbi.nlm.nih.gov).
- [2539] [Ammonia Technology Roadmap – Analysis](#), by IEA, IEA, October 2021 (168 pages).
- [2540] [Catalysis How Dirt and Sand Catalyze Some of the Most Important Transformations](#), by Justin J. Teesdale, Harvard Energy Journal Club, September 2017.
- [2541] [Nitrogen Fixation and Hydrogen Metabolism in Cyanobacteria](#), by Hermann Bothe, Oliver Schmitz, M. Geoffrey Yates, and William E. Newton, Microbiology and Molecular Biology Reviews, December 2010 (23 pages)  (preprint on ncbi.nlm.nih.gov).
- [2542] [Heterocyst Thylakoid Bioenergetics](#), by Ann Magnuson, Life, January 2019.
- [2543] [Biosynthesis of Nitrogenase Cofactors](#), by Stefan Buren, 2020 (48 pages).
- [2544] [Anabaena – structure of the vegetative body and reproduction](#), by Prema, December 2018 Accessed: 2026-01-10.
- [2545] [Science Practice Challenge Questions - Prokaryotic Diversity](#), by Texas Gateway, 2019.
- [2546] [Biosynthesis of Nitrogenase Cofactors](#), by Stefan Buren, Luis M. Rubio, et al., Chemical Reviews, January 2020 (48 pages).
- [2547] [Natural and Engineered Electron Transfer of Nitrogenase](#), by Wenyu Gu and Ross D. Milton, Stats, April 2020 (24 pages).
- [2548] [Strong Electron Correlation in Nitrogenase Cofactor, FeMoco](#), by Jason M. Montgomery and David A. Mazziotti,

- The Journal of Physical Chemistry A, May 2018 (28 pages)  (preprint on [arXiv](#)).
- [2549] [Understanding the Electronic Structure Basis for N₂ A Systematic Quantum MechanicsMolecular Mechanics Investigation](#), by Yunjie Pang and Ragnar Bjornsson, Inorganic Chemistry, March 2023.
- [2550] [Simulating Models of Challenging Correlated Molecules and Materials on the Sycamore Quantum Processor](#), by Ruslan N. Tazhigulov et al., Caltech, November 2022 (11 pages).
- [2551] [Simulations Using a Quantum Computer Show the Technology's Current Limits](#), by Philip Ball, November 2022.
- [2552] [A distributed multi-GPU ab initio density matrix renormalization group algorithm with applications to the P-cluster of nitrogenase](#), by Chunyang Xiang, Weile Jia, Wei-Hai Fang, and Zhendong Li, arXiv, December 2023 (34 pages).
- [2553] [Biosynthesis of Nitrogenase FeMoco](#), by Yilin Hu et al., 2011 (14 pages)  (preprint on [ncbi.nlm.nih.gov](#)).
- [2554] [Protonation of Homocitrate and the E1 State of Fe-Nitrogenase Studied by QM/MM Calculations](#), by Hao Jiang et al., Inorganic Chemistry, November 2023.
- [2555] [Towards sustainable agriculture: Fossil-free ammonia](#), by Peter H. Pfromm et al., 2017 (12 pages)  (preprint on [osti.gov](#)).
- [2556] [Combining theory and experiment in electrocatalysis: Insights into materials design](#), by Zhi Wei Seh, Thomas F. Jaramillo, et al., Science, January 2017 (33 pages)  (preprint on [osti.gov](#)).
- [2557] [Quantum-centric Supercomputing for Materials Science: A Perspective on Challenges and Future Directions](#), by Yuri Alexeev, Maximilian Amsler, et al., arXiv, September 2024 (65 pages).
- [2558] [Experimental realization of an extended Fermi-Hubbard model using a 2D lattice of dopant-based quantum dots](#), by Xiqiao Wang, Richard Silver, et al., Nature Communications, November 2022.
- [2559] [Observing ground-state properties of the Fermi-Hubbard model using a scalable algorithm on a quantum computer](#), by Stasja Stanisic, Ashley Montanaro, et al., Nature Communications, October 2022.
- [2560] [Quantum Utility in Simulating the Real-time Dynamics of the Fermi-Hubbard Model using Superconducting Quantum Computers](#), by Talal Ahmed Chowdhury, Vladimir Korepin, Vincent R. Pascuzzi, and Kwangmin Yu, arXiv, September 2025 (18 pages).
- [2561] [Towards Quantum Computational Mechanics](#), by Burigede Liu, Michael Ortiz, and Fehmi Cirak, arXiv, October 2024 (38 pages).
- [2562] [Quantum Multiscale Modeling: A Hierarchy of Algorithms for Complex Chemical Systems](#), by Seenivasan Hariharan, Kareljan Schoutens, Sachin Kinge, and Lucas Visscher, arXiv, July 2026 (30 pages).
- [2563] [Materials Discovery With Quantum-Enhanced Machine Learning Algorithms](#), by Ignacio F. Graña, Savvas Varsamopoulos, Tatsuhito Ando, Hiroyuki Maeshima, and Nobuyuki N. Matsuzawa, arXiv, March 2025 (19 pages).
- [2564] [KAIST Researchers Use Quantum Computers to Unlock New Energy Materials](#), by Matt Swayne, The Quantum Insider, September 2025.
- [2565] [Solving an Industrially Relevant Quantum Chemistry Problem on Quantum Hardware](#), by Ludwig Nützel, Michael J. Hartmann, et al., arXiv, August 2024 (12 pages).
- [2566] [A New Angle on Quantum Subspace Diagonalization for Quantum Chemistry](#), by Xeno De Vriendt, Jacob Bringe-watt, Nik O. Gjonbalaj, Stefan Ostermann, Davide Vodola, Johannes Borregaard, Michael Kühn, and Susanne F. Yelin, arXiv, February 2026 (18 pages).
- [2567] [SEEQC Partners With BASF To Explore Applications of Quantum Computing in Chemical Reactions for Industrial Use](#), by Matt Swayne, The Quantum Insider, February 2023.
- [2568] [Modeling the Performance of Early Fault-Tolerant Quantum Algorithms](#), by Qiyao Liang et al., Physical Review Research, December 2023.
- [2569] [Modelling Carbon Capture on Metal-Organic Frameworks with Quantum Computing](#), by Gabriel Greene-Diniz, Marko J. Rančić, David Muñoz Ramo, et al., arXiv, January 2023 (21 pages).
- [2570] [CQC and Total Announce Multi-Year Collaboration to Develop Quantum Algorithms for Carbon Capture, Utilization and Storage \(CCUS\)](#), by CQC, April 2020.
- [2571] [A Multi-Scale Quantum Framework for Evaluating Metal-Organic Frameworks in Carbon Capture](#), by Tom W. A. Montgomery, Adrian Varela-Alvarez, Sam Genway, Philip Llewellyn, and Phalgun Lolur, arXiv, June 2025 (11 pages).
- [2572] [Description of reaction and vibrational energetics of CO₂-NH₃ interaction using quantum computing algorithms](#), by Manh Tien Nguyen, 2023 (10 pages)  (preprint on [osti.gov](#)).
- [2573] [Capgemini announces new project with the Defense Advanced Research Projects Agency on quantum computing for energy transition](#), by Capgemini, May 2024.
- [2574] [Quantum simulation of carbon capture in periodic metal-organic frameworks](#), by Dario Rocca, Hong Woo Lee, et al., arXiv, October 2025 (13 pages).
- [2575] [The Promise and Challenges of Quantum Computing for Energy Storage](#), by Alan Ho et al., Joule, May 2018 (4 pages).
- [2576] [Can quantum computing fuel a leap in battery research? Not any time soon, but automakers and quantum companies are exploring it right now](#), by Grace Donnelly, April 2022.
- [2577] [Computational Investigations of the Lithium Superoxide Dimer Rearrangement on Noisy Quantum Devices](#), by Qi Gao, Hajime Nakamura, Tanvi P. Gujarati, Gavin O. Jones, Julia E. Rice, Stephen P. Wood, Marco Pistoia, Jeannette M. Garcia, and Naoki Yamamoto, arXiv, August 2019 (6 pages).
- [2578] [Surface Reaction Simulations for Battery Materials through Sample-Based Quantum Diagonalization and Local Embedding](#), by Marco Antonio Barroca, Mathias Steiner, et al., arXiv, March 2025 (25 pages).
- [2579] [Towards the simulation of transition-metal oxides of the cathode battery materials using VQE methods](#), by Marwa H. Farag and Joydip Ghosh, arXiv, August 2022 (32 pages).
- [2580] [Quantum Chemistry Simulations of Dominant Products in Lithium-Sulfur Batteries](#), by Julia E. Rice, Tanvi P. Gujarati, Tyler Y. Takeshita, Joe Latone, Mario Motta, Andreas Hintennach, and Jeannette M. Garcia, arXiv, December 2021 (7 pages).
- [2581] [Fault-tolerant resource estimate for quantum chemical simulations: Case study on Li-ion battery electrolyte molecules](#), by Isaac H. Kim, April 2022 (27 pages).
- [2582] [IonQ and Hyundai Motor Company Expand Quantum Computing Partnership, Continuing Pursuit of Automotive Innovation](#), by IonQ, December 2022.
- [2583] [Simulating key properties of lithium-ion batteries with a fault-tolerant quantum computer](#), by Alain Delgado, Juan Miguel Arrazola, et al., arXiv, February 2023 (31 pages).
- [2584] [Simulating X-ray absorption spectroscopy of battery materials on a quantum computer](#), by Stepan Fomichev, Juan Miguel Arrazola, et al., arXiv, May 2024 (12 pages).
- [2585] [Quantum algorithm for simulating resonant inelastic X-ray scattering in battery materials](#), by Ignacio Loaiza, Alain Delgado, et al., arXiv, March 2026 (18 pages).
- [2586] [AMTE Power's Project Quantum Signals New Era of British Battery Manufacturing](#), by AMTE, September 2020.

This **Chemistry and materials** subsection contains 202 bibliographical references and notes containing at least 4,734 pages.

3.4 Life sciences

The healthcare market is one of the most sought after by quantum computing players with the largest number of dedicated startups. It covers of course drug discovery and pharmaceutical industries but also diagnostics, all sorts of treatments optimizations like in radiotherapy and various healthcare systems operations.

With financial services, the life sciences domain and market are perceived as being the most promising ones. It may also be oversold given the technology challenges to overcome in order to deliver the many promises put forward, like “*we’ll develop new drugs more efficiently*”. This is well described in a systematic review published in npj Digital Medicine in 2025 that screened 4,915 quantum machine learning studies in digital health [2587]. Only 169 were eligible, 123 of those were then discarded for insufficient rigor, and out of the 46 survivors, only 16 considered realistic operating conditions, meaning quantum hardware runs or noisy simulations. Most papers were using some form of QML with poorly addressed data encoding scalability concerns (examples in [2588, 2589], and in [2590] with only 7 tested qubits, with no resource estimates for a quantum advantage).

Another source highlights the discrepancy between the growth in size of the molecules simulated with quantum computers and the related growth in number of QPU physical qubits [2591] (Figure 178).

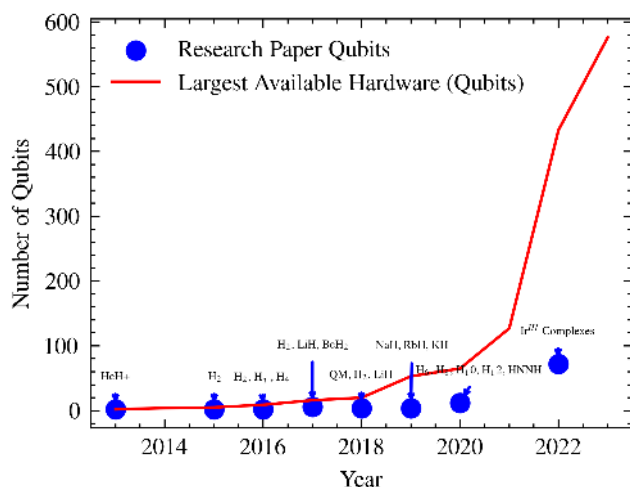


Figure 178: How the simulated molecule size grows along with QPU qubit number growth, most of these being based on VQE. The reason seems to be the poor quality of the available qubits. But this trend is the same in other use cases as highlighted in this part of the book. The chart stops in 2023. The molecular and biomolecular experiments described further in this section now reach 112 to 156 qubits. Source: [2591].

We’ll look at these various fields in this section, showcasing review papers and various case studies, and their related applicability [2592].

Molecular simulations

Molecular simulations are used to determine various properties of organic molecules based on their atomic description (at the lowest level), nucleotide structure (for DNA and RNA) and amino acid sequences (for enzymes, polypeptides, and proteins). These properties include their energetic ground state, 3D structures and potential interactions [2420].

Molecular simulations rely on the broad field of computational chemistry. The description of the nature of chemical bonds by **Linus Pauling** in 1928 really launched the vast field of quantum chemistry. Chemical bonds describe the way electrons of covalent liaisons are shared between atoms and the shape of their related orbitals. Pauling’s work came just after the creation of the **Born-Oppenheimer** approximation in 1927 which simplified Schrödinger’s equation for a molecule by separating the nuclei of the atoms from their electrons. The same year, **Llewellyn Thomas** (1903-1992, English) and **Enrico Fermi** (1901-1954, Italian American) created the later-called **Thomas-Fermi model** which describes the electronic structure of multi-atoms systems. The **Hartree method** (1928) and **Hartree-Fock method** (1930-1935) are also used to approximate the Schrödinger’s wave function and the energy of a quantum many-body system in a ground state.

Density functional theory came much later, in 1964, with the creation of the two Hohenberg-Kohn theorems by **Walter Kohn** (1923-2016, Austrian then American) and **Pierre Hohenberg** (1934-2017, French American). This was closely followed by **Kohn-Sham**’s equations from **Lu Jeu Sham** (1938, Chinese) in 1965. They are the basis of **DFT** (Density Functional Theory), a method that determines the ground-state properties of a many-electron system from its electron density rather than its many-body wavefunction, using an approximate exchange-correlation functional, which is far simpler than solving Schrödinger’s equation with its many variables. Walter Kohn was awarded the Nobel Prize in Chemistry in 1998 for this work, along with **John Pople** (1925-2004, English) who had contributed to the modeling of electronic orbitals in molecules.

Separately, multiscale modeling of complex chemical systems was developed by **Martin Karplus** (1930-2024, Austrian American), **Michael Levitt** (1947, South-African born British-American-Israeli) and **Arieh Warshel** (1940, Israeli-American) who combined quantum and classical mechanical descriptions to model chemical reactions in the 1970s.

They were awarded the Nobel Prize in Chemistry in 2013 for their work. The DFT model was also simplified by **Axel Becke** (1953, Canadian) in 1993 with the hybrid DFT. Some quantum versions of DFT are developed for quantum computing and with some progress, even for NISQ QPUs [2593].

Molecular simulation must also account for quantum nuclear motion, the continuous vibrations of molecules in

their aqueous medium. Chemical bond vibrations have femtosecond periods and collective atomic motions picosecond timescales. On the other hand, more complex chemical processes such as the production and folding of proteins occur on scales ranging from micro-seconds to seconds.

The first small-scale tests of molecular simulation using quantum algorithms were done in 2010 on photonic and NMR platforms with the ground-state energy of H_2 [2594], then on D-Wave annealers and superconducting qubits QPUs, using various breeds of variational algorithms. Analog quantum simulators are also machines suitable for simulating the interaction of atoms within molecules.

One approach relies on generic frameworks that can be distributed over classical computing in massively parallel architecture, and then progressively over quantum computing. This is the case of the **Tinker-HP** framework co-created by Jean-Philip Piquemal, cofounder of **Qubit Pharmaceuticals** and that the company plans to extend with hybrid quantum algorithms [2595, 2596]. Most other quantum startups like **ApexQubit**, **HQS Quantum Simulations**, **MentenAI**, **ProteinQure** and **Qulab** are indeed adopting hybrid computing models, if only to have something practical to market [2597].

NISQ The most common hybrid method is the VQE (Variational Quantum Eigensolver) that is using NISQ quantum hardware, but with various scalability challenges and long execution times [2598]. At this point in time, however, the priority is with quantum inspired algorithms [2599], *aka* classical algorithms based on quantum algorithms [2600].

QI Quantum and quantum inspired computation complete the vast field of machine learning which is already very common in the discovery of therapeutic molecules [2601]. Today, most molecular simulation calculations using various forms of tensor networks are carried out using algorithms running on classical supercomputers, increasingly using GPGPUs such as those from Nvidia or Google's TPUs.

FTQC Molecular simulation can start with simple organic molecules like cholesterol up to protein folding which is many orders of magnitude more complex [2602, 2603]. This last feat is therefore bound to be a very long-term one. Today, we can simulate peptides with about ten amino acids [2604]. The reference resource-efficient encoding needs a qubit count growing as $O(N_a^2)$, N_a being the number of amino acids, while it is the number of Hamiltonian terms that grows as $O(N_a^4)$ [839]. The underlying lattice protein folding optimization problem is itself NP-complete, as seen in the section dedicated to complexity theories, which bounds what any exact method can achieve.

Docking

Docking is a method used to predict how one molecule will bind to another one and form a stable complex. It is

used to predict the function of a drug on a target. It can be used for de-novo drug design or for drug repurposing, when existing drugs are tested on various targets.

78% of the search for therapies is focused on light molecules of less than 900 Daltons, i.e. about a hundred atoms. Its function is to associate itself with a target in the cells, often a specific protein that controls a metabolism that we want to attenuate or amplify [2605] (Figure 179). The discovery of small molecules of a few dozen atoms could be within the scope of the NISQ quantum computers within a few years. The first molecular simulation experiments were carried out on D-Wave annealers. It can perform better than classical simulations in some narrow circumstances [2606]. They work with the search for energy minima, which can be suitable in theory for the simulation of the organization of molecules.

A collaboration was launched in June 2017 between **Bio-gen**, the Canadian quantum software company **1QBit**, and **Accenture** for the creation of new molecules. **Bio-gen** (1978, founded in Geneva, now headquartered in Cambridge, Massachusetts) is a mid-size biotech company with about 7,500 employees specialized in the treatment of neurological and neurodegenerative diseases, its oncology franchise being a legacy activity.

Their use of quantum computing was aimed at retargeting therapeutic molecules, looking for matching between existing treatments and therapeutic targets in neurodegenerative or inflammatory diseases. **Amgen** is also active in the search for new therapies and is working since 2020 with **QSimulate** (2018, USA, \$4M).

A similar project was launched in Spain with the consortium "QHealth: Quantum Pharmacogenomics applied to aging" launched in August 2020 with **aQuantum** (alhambraIT, Prologue Group) with **Gloin**, **Madrija** and various Spanish Universities. Its goals are to find correlations between physiological and genetic variables, drug usage history, side effects and/or potential lack of response of new drugs to fight aging. They plan to do simulations using quantum algorithms. The project totals €5.1M including a grant of €3.7M from CDTI awarded in November 2022 by the **Center for the Development of Industrial Technology** (CDTI) of the Ministry of Science and Innovation of Spain.

NISQ Molecular docking was tested in gate-based computing with digitized-counterdiabatic QAOA which maps a molecular docking problem onto the maximum vertex weight clique problem, a typical optimization problem. This was however only tested on 6, 8 and 12 qubits, way under any quantum advantage or scalability proof threshold [2607]. Pfizer conducted another digitized-counterdiabatic QAOA docking experiment on 14-node and 17-node instances, but these were emulated on a GPU cluster and not run on a QPU, with a future plan to "stress" the solution to 20 and 34 qubits, which in all cases wouldn't reach any quantum advantage [2608]. Likewise, in Singapore, a molecular docking problem was

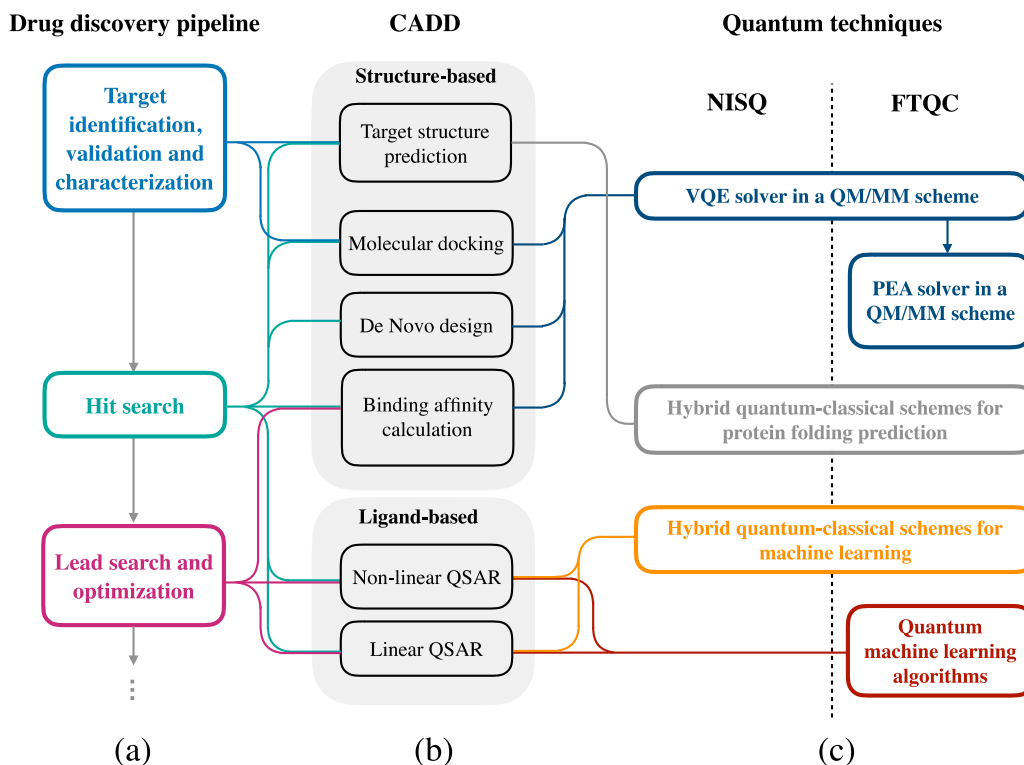


Figure 179: Drug discovery Process flow. CADD = Computer Aided Drug Design. Source: [2605].

reformulated into a maximum vertex-weighted clique problem (MVWCP) problem, but tested with only 5 and 6 qubits, far away from any potential quantum advantage [2609].

NISQ A hybrid quantum/classical machine learning algorithm was used for drug retargeting using Ligand Based Virtual Screening (LB-VS). It was developed in 2022 by **IBM** and **The Hartree Center** in the UK [2610].

NISQ Chugai Pharmaceutical (Japan) and Amazon tested different methods to solve peptide-protein docking in 2024. They run a QUBO problem formulation run on a D-Wave hybrid solver which could handle 6 peptide residues and 34 target protein residues but had trouble scaling beyond this problem size. They compared it with a novel classical constraint programming (CP) approach that seemed to perform much better and could support 13 peptides residues [2611].

At last, quantum-inspired methods are also tested, outside the QPU realm [2612].

Genomics

In genomics, the first potential application of quantum computing is with DNA sequencing and sequences alignment. It is about reconstructing a giant puzzle with small parts of DNA sequences which come out of sequencing [2613] **NISQ**.

The current sequence alignment quantum algorithms are not yet in the realm of current QPU solutions. They need QFT, qRAM for fast data loading, and others

mention the need to use Grover-derived algorithms that won't scale well [2614, 2615]. Or they are tested at very small scales like in Sappington et al. who compared the mutational robustness of RNA folding, spin glasses and random quantum circuits, the quantum part running on a 7-qubit IBM machine that can be emulated faster on a simple laptop [2616], and Varsamis et al. which ran a five-qubit DNA sequence alignment demonstration [2617, 2618] **NISQ**.

New solutions are proposed to optimize data loading for these use cases [2619]. Quantum computing could also be used to process measurement data coming from electrodes that detect nucleotides during sequencing. But no qubit count was provided in the related paper, which is probably not in the quantum advantage domain, and works in emulation mode with being probably too slow compared to classical computing [2620].

Genomics also involves the study of gene regulatory networks (GRNs) from single-cell transcriptomic data which defines the ensemble of regulatory relationships between genes in a biological system [2621].

As shown in [2622], many challenges must be overcome to drive some quantum advantage with all these use cases.

The **DNA-Seq Alliance** combines the startup DNA-Seq and D-Wave, which also does molecular retargeting by combining genomics, protein kinase crystallography, quantum computing and the search for effective cancer treatments.

Proteins and RNA folding

Protein folding and structure prediction are the gold standards of molecular simulation. It is part of structural biology, the branch of molecular biology, biochemistry, and biophysics that studies the molecular structure of biological macromolecules like proteins and DNA/RNA strands. It helps determine how these structures influence the functions of the macromolecules within cells. Classical experimental techniques use X-ray crystallography, NMR spectroscopy, cryo-Electron Microscopy (cryo-EM) and Small-Angle X-ray Scattering (SAXS). Then, digital simulations help determine these structures in-silico.

Computing protein folding helps predict the 3D structure and function of proteins based on their amino acid sequence, itself determined by the nucleotide sequence of their active RNA and DNA sequences. Future quantum solutions will compete with AlphaFold, the classical solution from Google DeepMind which predicts structures for larger proteins but with some limitations, particularly for entirely de-novo designs [2623]. Protein structure is described at four levels (Figure 180): the primary structure is the amino acid sequence, the secondary structure covers local backbone conformations such as α -helices and β -sheets, the tertiary structure is the overall three-dimensional fold of a single polypeptide chain, and the quaternary structure is the organization of several chains into a multi-subunit assembly. These are structural levels, not folding mechanisms.

NISQ One protein folding algorithm was able to simulate the secondary folding of the 10 amino acid Angiotensin on 22 qubits, in emulation, and to run a 7 amino acid neuropeptide on 9 qubits of an IBM 20-qubit quantum computer, in 2019 [839]. This hybrid protein folding algorithm published in 2021 combined a CVaR variational quantum algorithm with an evolutionary optimizer. Its model Hamiltonian contains $O(N^4)$ terms, N being the number of monomers in a tetrahedral lattice. The paper did not mention the performance of the classical DeepMind AlphaFold, whose first version dates back to CASP13 in 2018 and whose AlphaFold 2 breakthrough was published in 2021 [2624], nor any results from the various CASP (Critical Assessment of Structure Prediction) programs that serve as a benchmark in this field. Also, since the circuit depth grew with the size of the proteins to fold, it quickly exceeded the capacities of noisy qubit systems as the number of amino acids grew.

To provide some classical comparison, AlphaFold predicts tertiary structures for proteins with up to 2,700 amino acids, way above what the NISQ algorithm mentioned above could implement. And NISQ solutions still have some limitations like with mutated variants of proteins and proteins with many intrinsically disordered regions (IDRs) [2625]. AlphaFold is currently in its version 3 [2626]. DeepMind and EMBL-EBI catalogued all known proteins in the AlphaFold Protein Structure

Database (AlphaFold DB), with over 214 million documented structures [2627]. Demis Hassabis and John Jumper were awarded the Nobel Prize in Chemistry in 2024 for this work, along with David Baker. AlphaFold 3 now predicts the 3D structure of a larger spectrum of biomolecules including DNA, RNA, and ligands. It even can predict protein-ligand interactions [2626].

NISQ A protein with 12 amino acids was folded on IonQ Forte trapped-ion qubits with Kipu's BF-DCQO algorithm in 2025 [2628], then 14 to 16 amino acids mapped onto 46 to 61 qubits of a fully connected 64-qubit barium system of the IonQ Tempo class, still with BF-DCQO, in 2026 [2629]. A Chalmers, Cleveland Clinic and IBM team reached 26 amino acids using 46 qubits of the 156-qubit `ibm_kingston` in 2025 [2630]. A record of 60 nucleotides for an mRNA secondary structure prediction was then reached the same year using an IBM 156-qubit quantum processor [2631]. In 2025, Cleveland Clinic improved the secondary structure prediction of proteins with the latest IBM Heron QPU [2632]. It performed very well against AlphaFold2 but only for 5 to 26 amino acids. In December 2025, they implemented a NISQ workflow to predict the relative energies of two conformers of the 300-atom Trp-cage miniprotein using SQD and a local unitary cluster Jastrow (LUCJ) ansatz (we described this technique in 3.3.1) [2633].

Other classical approaches use LLM (large language models), with good results. One competing project is **ESMfold** developed by Meta. Its underlying ESM-2 language model family scales up to 15 billion parameters, and it produced the ESM Metagenomic Atlas and its 617 million predicted structures [2634].

Many projects have been launched to improve quantum-based protein folding and they remain theoretical since no existing quantum computer is powerful enough to execute them [2635].

NISQ One common algorithm used here is QAOA but its applicability is questioned [2636, 2637], likewise with QUBO on D-Wave annealers [2638, 2639], which doesn't scale well with current hardware [2640].

FTQC Researchers from Chalmers University of Technology estimated in 2023 the resources needed to fold proteins using a simplified coarse-grained model. A coarse-grained protein folding problem uses simplified modeling and simulation of protein folding by reducing the level of detail in the protein's structure. It groups atoms into larger units or "beads", representing clusters of atoms up to entire amino acids. They end up with estimates in the thousand qubits and over 10^8 two-qubit gates which mandates some FTQC architecture with millions of physical qubits at best [2641]. A CVaR-VQE turn-based encoding published in 2024 went much further on hardware, using up to 114 qubits of the 127-qubit `ibm_cusco` for a 20-bead HP-model chain, but with very shallow ansatzes and no quantum advantage claim [2642].

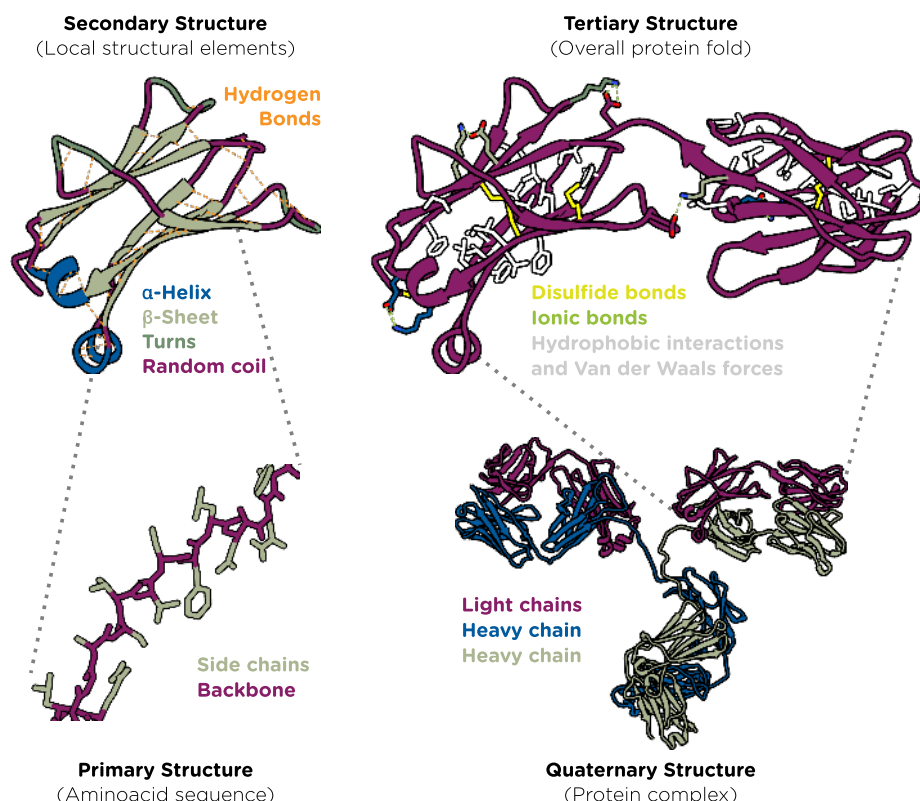


Figure 180: Primary, secondary, tertiary and quaternary structure of a polypeptide or protein. Source: [FAQS Higher Order Structure and HOS Comparisons Explained](#). Added in 2023.

NISQ FTQC In September 2026, researchers from Cleveland Clinic proposed a VQE-based algorithm to reduce the cost of predicting protein folding using a “quantum torsion folder” (QTF) framework. Its qubit requirement for N torsional degrees of freedom scales as $O(\log_2 N)$, which is good [2643]. The paper did some tests on IBM systems using only 6 qubits due to the related circuit depth and size. To obtain a quantum advantage vs some classical circuit emulation, you would need about 20 qubits, enabling one million torsions estimations. But due to the circuit depth, it would require an FTQC platform, and the number of circuit shots would skyrocket, generating computing times above 13,000 years [2644].

analog Another test was published in 2021 by GSK [2645]. It was about solving a mRNA codon optimization problem. Each amino acid in a protein sequence can be encoded by as many as six different codons, these series of three DNA/RNA bases encoding one amino acid. The goal was to find the right combination of these codons. The codon selection in mRNA impacts protein folding and functions. The main task is to balance G and C bases in mRNA to optimize gene expression. This was one of the first published case studies using D-Wave Advantage annealer and its 5,000 qubits and their Leap Hybrid Solver. It worked well with 30 amino acids and could scale up to 1,000 amino acids. The codon optimization problem is formulated as a Binary Quadratic Model that is itself close to an Ising model adapted to D-Wave annealers. It did fare well when compared

to genetic and machine learning algorithms running on classical computers.

NISQ In 2024, a study from Koen Bertels et al. also looked at the way to solve mRNA codon optimization in a NISQ regime with a VQE variant. It was tested with between 8 and 21 qubits, with resources estimations in a NISQ regime of about 10K gates with >80 qubits [2646].

FTQC Khatami et al. propose a protein design algorithm based on a FTQC Grover algorithm, but “*the results of real quantum computers indicate the need for devices with much lower noise to implement our circuits*” [2647]. Again, what is required are FTQC computers. Chandarana et al. propose a hybrid analog-gate-based algorithm but tested only with 9 amino acids, using up to 17 qubits on Quantinuum quantum hardware [2648].

NISQ Malone et al. tested a VQE protein-ligand docking algorithm but only with 8 qubits, a 19 depth circuit and with no resource estimation for a quantum advantage regime [2649]. Domingo et al. tested a hybrid neural network using some quantum error mitigation to improve molecular protein binding affinity predictions, showing only a 40% gain in the training process with 9 qubits. It would require 99.99% gate fidelities and circuits with 300 gates [2650].

NISQ How about doing something simpler like computing the ground state energy of some protein structures. In 2025, IBM and STFC (UK) researchers used a QUBO reformulation of the problem, which was then mapped

on an Ising model and solved with QAOA. It helped reduce the computational cost vs classical simulated annealing techniques. It used 28 qubits on the Qiskit Aer emulator and was therefore not in a quantum advantage regime [2651].

FTQC RNA folding is also a field of research for quantum algorithms. At this point it covers only the secondary structure and could be implemented in FTQC platforms [2652].

The Holy Grail would be to understand how the assembly and then operations of ribosomes work. The human 80S ribosome is made of 80 proteins and 4 ribosomal RNAs, the bacterial 70S of about 55 proteins and 3 rRNAs. Ribosomes produce all proteins in our cells using messenger RNA code, which is itself synthesized from DNA through an also amazing biochemical process involving many complex molecules. Millions of ribosomes operate in a typical human cell, and each of them weighs 4.3 MDa, against 2.3 MDa for its bacterial counterpart. The number of 2.5 or 4.3 million atoms is often mentioned, but this is not true. These are “Daltons” which are equivalent to one twelfth of the mass of carbon 12, or about the mass of a hydrogen atom. However, these organic molecules contain, in addition to hydrogen, a lot of carbon, nitrogen, phosphorus and oxygen. The latter contribute to a large part of the mass of the molecule, hence the fact that the number of Daltons must be divided by about 9 to obtain the number of atoms of an organic molecule, and by about 13 if you only count non-hydrogen atoms. A human ribosome therefore contains on the order of 470,000 atoms.

NISQ Otherwise, protein and DNA related algorithms are published from time to time, but with ridiculous small hardware resources which makes it difficult to assess their relevance (10 qubits in [2333] and 8 qubits in [2334]).

3.4.1 Protein dynamics

Once a protein is folded, it interacts with its environment, usually within a cell. Its folding created a stable 3D structure that was conditioned by various molecular interactions, within the protein and between the protein and its aqueous environment: non-polar amino acids cluster in the interior, away from water while hydrogen bonds form between backbone atoms, creating secondary structures like alpha-helices and beta-sheets, and between amino acid side chains. There are also ionic bonds, disulfide bonds, thermodynamic and kinetic considerations which influence folding.

When proteins become operational in a cell, they are not standing still, they are neither rigid or static entities but dynamic machines constantly undergoing internal motions on a wide range of timescales while interacting with their environment and delivering their function. Protein dynamics refers to the continuous movements and fluctuations that occur within a protein’s structure

after it has folded into its native state. These fluctuations happen in various timescales: femtoseconds related to vibrations of individual atoms, picoseconds to nanoseconds for local side-chain rotations and loop fluctuations, and at last microseconds (μ s) to milliseconds with domain movements, conformational changes involved in enzyme catalysis or ligand binding. Protein dynamics are not random noise. They participate to their function. These various motions allow them to bind with other molecules (ligands, DNA, other proteins), catalyze reactions (enzymes), transmit signals, undergo large conformational changes (for muscle contraction, and molecular motors like the topoisomerase II which unknots entangled DNA molecules and the rotating ATP synthase that handles the process of ATP creation in mitochondria).

Studying protein dynamics can use classical experimental techniques like NMR spectroscopy, X-ray crystallography, and cryo-electron microscopy. Computational molecular dynamics simulations come in. You can guess that it quickly reaches limits in the classical computing realm. As a result, the use of quantum computing is obviously studied in the domain. It starts with picking quantum algorithms and providing resource estimates, as done well in [2653].

Beyond protein dynamics sit the field of biomolecular networks which conditions at a higher abstraction level protein–protein interactions, gene–gene associations, and cell–cell interactions. Algorithmiq is studying the use of quantum walks to simulate these interactions, although with no resource estimates [2654].

FTQC The first quantum algorithms are also proposed to compute biomolecules free energy [2655]. Free energy, particularly Gibbs free energy, is a thermodynamic quantity that tells how much energy is available to do useful work during a chemical or physical transformation. It helps predict chemical reaction feasibility without external input, explain how nutrients are converted into ATP, proteins, and other essential molecules, and how protein folding and binding interact with minimizing free energy.

Simulating how a whole biological cell works is another challenge. This is the field of whole-cell simulations. This could be potentially done at different scales with whole-cell spatial modeling, whole-cell network modeling and atomistic-molecular modeling, each slice using a variety of quantum NISQ and FTQC algorithms [2656]. It is however a field in its very early stage.

3.4.2 Diagnosis

Using quantum machine learning algorithms to do medical image analysis is also investigated. It may help train these systems with less data. However, existing algorithms are currently just on par with their classical peers or with serious limitations [2657, 2658]. Most of them have been tested in variational circuits support-

ing only a few NISQ physical qubits, and for cancer detection in multiple medical imaging sources, including smartphone pictures [2659].

Let's start with medical imaging applications:

NISQ Landman et al. proposed in 2022 to accelerate the training of classical neural networks for medical image classification with a NISQ algorithm with a similar quality than with classical algorithms and no indication of time and cost [608].

NISQ Kiani et al. showcased in 2020 a tomography enhanced image reconstruction QFT-based algorithm with an exponential speedup, provided the imaging data is provided as a quantum state, which is not the case in most if not all situations [2660].

NISQ Deep learning methods were tested for clinical data classification by Moradi et al. in 2022 and for prostate cancer detection by El Maouaki et al. in 2024, using QSVCs and quantum kernel Gaussian process methods and transfer learning based QNNs, but only with 11 and 5 qubits with no indication of the conditions to obtain some quantum advantage [2661, 2662]. Likewise, in 2026, another “toy” test was made using 4 qubits [2663].

NISQ A similar test was done in 2024 to improve MRI (Magnetic-Resonance Imaging) image recognition with only 16 qubits [2664]. Another one done in China in 2024 was done with a GPU-based emulator, so, not demonstrating any quantum advantage [2665]. Even in 2025, another brain image detection algorithm was tested with dividing the problem in chunks of 5 qubits [2666]. One can wonder why in these conditions you'd need a quantum computer.

NISQ Another method of cancer detection proposed in 2022 by Azevedo et al. uses angle embedding which is simpler than a full vector state amplitudes embedding and provides better accuracy than classical methods at small scale. The speed was divided in two with only 7 qubits which means it could be even faster with a classical quantum code emulator. And there's no word in the paper on image resolution [2667].

NISQ A 2023 paper, from Majumdar et al. is using a variety of QML algorithms for histopathology classification with tests done on a PennyLane emulator with up to 7 qubits. Of course, then, with no quantum advantage [2668]. Another test was achieved in 2026, involving only 6 qubits [2669].

NISQ In 2025, a US team developed a hybrid classical-quantum Generative Adversarial Network (GAN) for skin disease image. The quantum process was using a 127-qubit IBM QPU but with circuits chunks of 10 qubits, way below any quantum relevance and with average results [2670].

NISQ Turkish scientists used D-Wave and IBM QPUs in 2020 to classify CT scans images related to COVID diagnosis [2671]. Their quantum software generated 94% to 100% successful classifications while its classi-

cal counterpart did achieve 90% successful results. It was using only 4 qubits from 5-qubits systems (IBM Q-Rome and Q-London). It was a hybrid computing using the quantum transfer learning method. Quantum computing was used only for the classification part at the end of a convolutional network, not for the convolutions that remain classical, using 224×224 pixels versions of the CT scans, using a training set made of 2658 lung CT images with 1,296 COVID-19 and 1,362 Normal CT images. They also tried emulators running with PennyLane, Qiskit and Cirq. The classification layer compressed 512 vectors into 4 vectors with a linear transformation.

analog Another experiment in UK and Germany published in 2025 was using D-Wave annealers to analyze image datasets of 28×28 pixels, but the paper did cast some doubt on the ability to expand this to real-life scenarios “*given the current limitations of quantum computing hardware*” [2672]. In 2026, another team extended this kind of application to images with 256×256 pixels using a “superpixel” problem reduction technique which was tested on mammography images, but without any claim of quantum advantage [2673].

NISQ In the neurologic diseases field, quantum machine learning could be used to detect Alzheimer's disease using handwriting analysis, although not yet in a quantum advantage regime [2674] as well as brain tumors [2675]. A 2025 paper did the same with a hybrid MRI image classification. All the convolutional layers were implemented classically. The quantum part was just a binary classifier using the output of the last convolution running with 3 qubits [2676] (Figure 181). QML-based image classification was also tested in 2025 to help detect dementia on an IonQ QPU. But no matched classical baseline is reported. At that circuit width, the classifier remains classically reproducible [2677]. It was not better with a QAE (Quantum Autoencoder Method) done in 2026 that used only 4 qubits [2678] nor with another method that was using 15 emulated qubits [2679].

NISQ Analyzing ECG (electro cardiograms) is also a field of interest. But research is far from bringing any quantum advantage like with this 6-qubit version [2680].

And other diagnosis applications:

NISQ Alz-QNet uses a Quantum Regression Network (QGRN) to decode how some crucial regulatory or switching genes in the AD Amyloid Beta Precursor Protein (APP) behave, which can help understand the pathogenesis of Alzheimer's disease, and find potential gene inhibitors or regulators for diagnosis and treatment. It runs on a 9-qubit classical emulator, with no matched classical baseline and no resource estimate for reaching a quantum advantage [2681].

NISQ Developed in China in 2022, **DeepQuantum** is a hybrid quantum deep learning algorithm that was used to make predictions on mutant COVID-19 variants [2682]. It was trained using a database of available mutated SARS-CoV-2 RNA sequences. But it used a

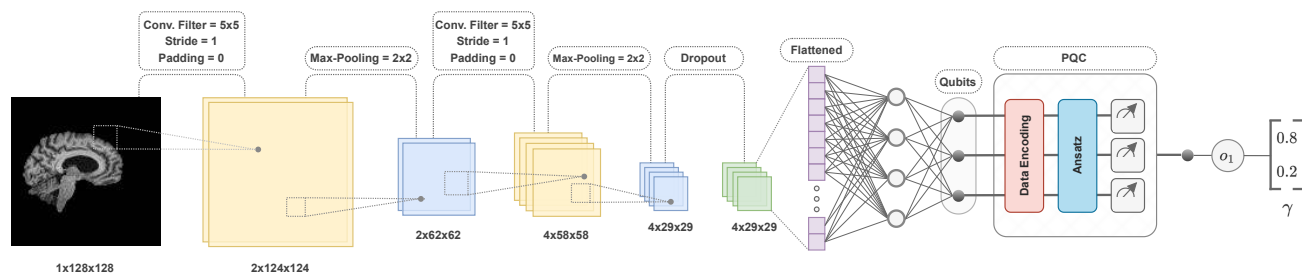


Figure 181: A hybrid classical-quantum detection of Alzheimer's disease with 128×128 pixel MRI images. The classical part is a typical multi-layer convolutional neural network. The quantum part is the last part of a fully-connected classifier at the end of the process. It is using only 3 qubits and yields a binary response. Considering this as being "quantum computing" is a stretch. Source: [2676].

mere 10 qubits way out of the scope of any quantum advantage.

FTQC A 2023 paper, from Kazdaghli et al. from **QcWare** and **AstraZeneca** showcases some improvements with the analysis of clinical data with "competitive results" obtained with 10 qubits on an IBM QPU. The quantum algorithm time is scaling linearly with the number of features while its best classical equivalent scales in cubic time which can indicate some potential quantum advantage [2683]. Using a real-life data set, the quantum algorithm would require 140 qubits and a computing depth of 400. It ends up corresponding to physical qubit fidelities of 99.998%, meaning some form of FTQC with at least 100 physical qubits per logical qubits. A similar conclusion came out of [2684] in 2026.

NISQ A classical PCA-based segmentation was used to feed a quantum machine learning model created in 2026 to predict heat-related physiological events using USA data, although with inconclusive results [2685].

3.4.3 Drug discovery and retargeting

Most major pharmaceutical companies have been exploring and evaluating the potential of quantum computing for a few years, starting by conducting pilot projects with D-Wave quantum annealers and, later on, gate-based quantum computers: AbbVie, Amgen, AstraZeneca, Bayer, Biogen, Bristol-Myers Squibb, Johnson&Johnson, Merck, Roche, Sanofi and Takeda.

Their aspiration is to extend the capabilities of current supercomputers to simulate living organisms' molecules "in silico", mainly to create or discover new treatments. This is the field of "*in-silico drug discovery*" [2686].

This quest is linked to the pharmaceuticals industry worrisome situation, that is discovering fewer new treatments and seeing diminished portfolio of commercial patented drugs. The drugs development cycle from discovery to market is becoming increasingly expensive, particularly during clinical trials. It costs up to \$1B, if not more, and failure rates are numerous. About 60% of oncology phase III trials fail, and only 3% to 5% of oncology candidates entering phase I eventually obtain FDA approval. If we could better digitally simulate

the effects of new treatments before clinical trials, we might be able to increase these success rates. Also, quantum computing could be a critical tool to create digital twins of molecular complexes, used to find the right combinations and optimize their efficiency.

On top of new drug discovery, pharmaceutical companies are also trying to leverage their existing portfolio with drug repurposing. It can speed up clinical trials since most of their adverse effects are already known [2687]. Even though this has not prevented the long controversy surrounding hydroxychloroquine in 2020! In any case, pharmaceutical players need simulation tools and in particular molecular simulation tools: to create molecules, from the simplest (peptides) to the most complicated (proteins, antibodies, vaccines), to model them in 3D, to analyze their interactions between their active sites and targets like cell surface proteins (transmembrane glycoproteins), to study protein-protein interactions (PPI) [2688], and to identify contraindications or detect potential toxicity [2689]. Such treatments can be created ex-nihilo, but most often, they are derived from existing ones (known protein, enzyme, bio-inspiration, ...). We could try to digitally simulate an entire cell with all its organelles. This would become quite complicated since a living cell comprises about 100 trillion atoms! So, forget it!

Drug design could leverage quantum computing in the future in various areas like with molecular simulations, molecular docking, and genomics (Figure 182). One critical field is of course oncology, with its diversity of cancers and treatments [2690].

Many review papers are available on drug discovery using quantum computers, which can use a mix of quantum simulations and quantum machine learning [2691, 2692]. Some point to the enormous time scales for computing even small molecules using either NISQ or FTQC platforms, both in the future tense [2693].

Santagati et al. explain that NISQ algorithms are way too slow due to the number of required shots and measurements, and thus focus only on FTQC architectures [2425]. A RAND Europe report funded by the Novo Nordisk Foundation reaches comparable conclu-

sions [2694]. Baiardi et al. mentions the capability to compute vibrational structure [2420] **NISQ**.

Blunt et al. describe the large number of physical qubits required for molecular simulations and provide good estimates of VQE and QPE qubit and shot requirements for a given example using large-size complete active space configuration interaction (CASSI) calculations. The example relates to the drug Ibrutinib bound covalently to Bruton's tyrosine kinase [2686]. The paper mentions Johnson et al. [2695] which propose an optimization method to run VQE for simple molecules. The results range from 1,300 to 634,915 years of computing, provided you have between 120,000 and 352,000 physical qubits. With some optimization, computing time is reduced to 7.6 days with 3.6 million physical qubits having 99.99% fidelities which is not currently available technology **FTQC**.

In 2022, a photonic-based Boson Gaussian sampler from China was used to implement discovery with molecular docking and RNA folding but “*unequivocal quantum computational advantage has not been realized in our experiments due to photon loss*” [2696] **NISQ**.

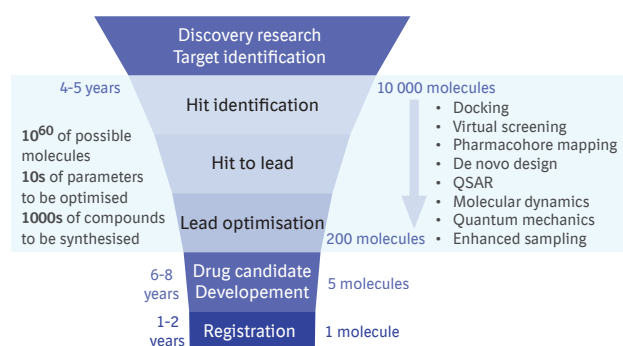


Figure 182: Drug design timeline. Source: [2425].

Po-Yu Kao, Alan Aspuru-Guzik et al. tried to demonstrate in 2023 a quantum advantage on small drug discovery task using a Quantum GAN running on 4 and 8 qubits. The circuits are small enough to be classically simulated, and the paper does not report a matched runtime comparison with the hardware. There is no mention of the impact of physical qubit noise on the model. In other words, they found a classical algorithm that beats another classical algorithm and at a very small scale. It is not bad per se but there's no practical scale and no quantum advantage was proven in any way since it is entirely theoretical. The paper doesn't contain any resource estimation for reaching such a practical quantum advantage either in a NISQ or FTQC regime. The paper does not analyze how the model size or the data-loading cost scales with the qubit count. All of this is misleading on the obtained quantum computing advantage [2697].

Likewise, a Nature Biotechnology paper published in January 2025 explained how to combine quantum and classical computing to enhance the design of potential

drugs targeting KRAS, a protein implicated in various cancers, including lung, colorectal, and pancreatic cancers. Fine. But its quantum part was only using 16 qubits [2698] **NISQ**.

Another technique, Quantitative structure-activity relationship (QSAR), is used in drug discovery to obtain insights into the biological interactions of potential drug candidates. Quantum machine learning solutions are being tested there, like in [2699]. But it was tested with only 3 qubits in 2025, with no matched classical baseline and no scaling analysis, so no conclusion can be drawn **NISQ**.

NISQ FTQC Qubit Pharmaceuticals performed in December 2025 three-dimensional protein pocket hydration-site prediction using a QUBO formulation and a Q-CTRL solver with their Fire Opal error suppression tool, running on up to 123 qubits (IBM Heron), with some potential to reach a quantum advantage with a larger number of qubits and megaquop-scale operation counts, in the eFTQC realm [2700].

NISQ Moderna is working with IBM on the prediction of the secondary structure of mRNAs on existing 127-qubit QPUs using CVaR-VQE, a variation of VQE, with up to 60 nucleotides [2701]. This is encouraging but still below what is needed, for example for the development of mRNA vaccines. As a comparison, the Pfizer-BioNTech and Moderna mRNA vaccines encode the SARS-CoV-2 spike protein with its 1,273 amino acids and classical mRNA secondary structure engines (ViennaRNA 2, MFold, RNAstructure) can do that for mRNA with thousands of nucleotides. In 2025, they co-optimized mRNA sequences for codon optimality and secondary structure with 30 nucleotides, on the same IBM Eagle QPU and they used up to 112 qubits, which is more serious although seemingly not yet in a quantum advantage regime [2702].

analog In 2023, a team in Russia built a compact discrete variational autoencoder (DVAE) with a Restricted Boltzmann Machine (RBM) of reduced size in its latent layer to implement generative chemistry and drug design on a D-Wave Advantage annealer, generating 2,331 novel chemical structures with potential drug function [2703].

NISQ In June 2019, **Merck** announced a three-year partnership with the Karlsruhe, Germany-based startup **HQS Quantum Simulations** for the development of quantum algorithms for chemical simulation. As already mentioned in the part related to SEEQC, **Merck** and SEEQC created together a consortium in 2021 to build a “commercially scalable application-specific quantum computer designed to tackle prohibitively high costs within pharmaceutical drug development”. The project was due for completion “in 18 months”! Meanwhile, Merck collaborated with QuEra to use a quantum reservoir computing (QRC, a variation of machine learning) method for molecular property prediction, based on the Aquila analog QPU [2704]. It did run with 18 qubits

given, for another reservoir computing use case, up to 108 qubits were tested on the same hardware [2705].

In September 2022, **Novo Nordisk** (Denmark) made an announcement that was unique in shape and form. It is launching the Novo Nordisk Foundation Quantum Computing Programme with the goal to build Denmark's generic quantum computer in 2034 with a funding of \$200M, and obviously to become a platform for chemical simulations. It will fund the Niels Bohr Institute but also embed partners from the USA, the Netherlands and Canada. They will invest first in three different qubit platforms and select the best one after a first 7-year period. Also, in 2023, **Moderna** and **IBM** had a partnership agreement announced in April 2023 also for molecular design, but we'll have to wait some years before IBM quantum computers are useful for such tasks.

The pharmaceutical industry is organized in a couple consortia which help it share best practices, including in quantum computing adoption. The **Pistoia Alliance** was conceived in Pistoia, Italy, in 2007 by AstraZeneca, GSK, Novartis and Pfizer, and formally incorporated in 2009. It entertains a quantum computing community. **QuPharm** is a consortium of 17 pharmaceutical companies, including AbbVie, Bayer, GSK, Takeda, and Pfizer that is dedicated to quantum computing.

SandboxAQ announced in June 2023 that it was creating its Bio-Pharma Molecular Simulation Division that will work on drug discovery using a mix of AI and quantum solutions.

In practice, conventional HPCs helped molecule screening for therapies and to create 3D models of the COVID-19 virus and, in particular, of its glycoproteins that cling to the membranes of human cells in order to attack them and enable the virus to reproduce within cells [2706]. In 2022, new SARS-CoV-2 main protease (Mpro) inhibitors were discovered using a classical GPU-based HPC using Atlas's platform from Qubit Pharmaceuticals [2707]. In the more or less distant future, quantum computing may have its say in similar pandemics [2708].

As with any new technology, quantum computing specialists must learn to interact with bioinformatics specialists. Fortunately, bioinformaticians are already bridging the gap between molecular biology and computer science and are well positioned to learn quantum methods [2709].

At last, clinical trial operations such as site selection, patient recruitment scheduling and treatment allocation can be formulated as optimization problems and addressed with quantum algorithms, although no quantum speedup has been established for these tasks [2710, 2711].

3.4.4 Treatments

Network medicine analyzes disease pathogenesis, coupling information coming from Omics databases (protein-protein interaction, genomics data) and environmental factors with Bayesian networks and machine learning tools. The idea is to identify novel disease genes and pathways, their diagnostics, and therapeutics.

NISQ A 2022 paper from Algorithmiq and a Finland research team outlined how quantum computing could help there. Unfortunately, it didn't contain sizing recommendations for hardware and the problems it could solve [2712]. Another paper from 2025 used QSVM (quantum support vector machines) to identify key biomarkers with COVID-19 patients with up to 16 qubits. At that width the model remains classically reproducible, and the paper provides no matched classical baseline, noise model or scaling analysis [2713]. See the review in [2714] that describes how quantum computing can help discover new biomarkers.

NISQ In 2023, IBM researchers published a good paper listing various use cases in cell therapeutics mostly centered around quantum machine learning, as shown in Figure 183 [2715].

NISQ In 2024, Terra Quantum presented a case study of transplant optimization using a hybrid classical-quantum algorithm, the latter working with 5 qubits [2716]. At that width, the quantum part is classically simulable and the workflow is effectively classical.

David Sahner, who created his own consulting firm **Eigenmed**, is one of the promoters of precision medicine based on predictive machine learning techniques using D-Wave annealers [2717].

3.4.5 Healthcare

Quantum computing could also potentially help improve the efficiency of healthcare systems: hospitals, their emergency departments, and also insurers. It can be about optimizing resource allocations and scheduling work hours for medical staff. You can also undertake various data analyses to find correlations.

The UK NQCC published a white paper on this in 2025 [2718] showing how extensive these potential use cases could be. It still falls in a typical trap with highlighting use cases which are only "quantum inspired" like one on radiotherapy planning optimization [2719].

NISQ Likewise, quantum computing could potentially help predict severe post-surgical complications [2720] and help run longitudinal biomedical studies that track disease progression, treatment response, and the emergence of resistance mechanisms [2721]. However, these two proofs of concept used fewer than 5 qubits and didn't demonstrate any quantum advantage. There's a long way from the promises to actual realizations!

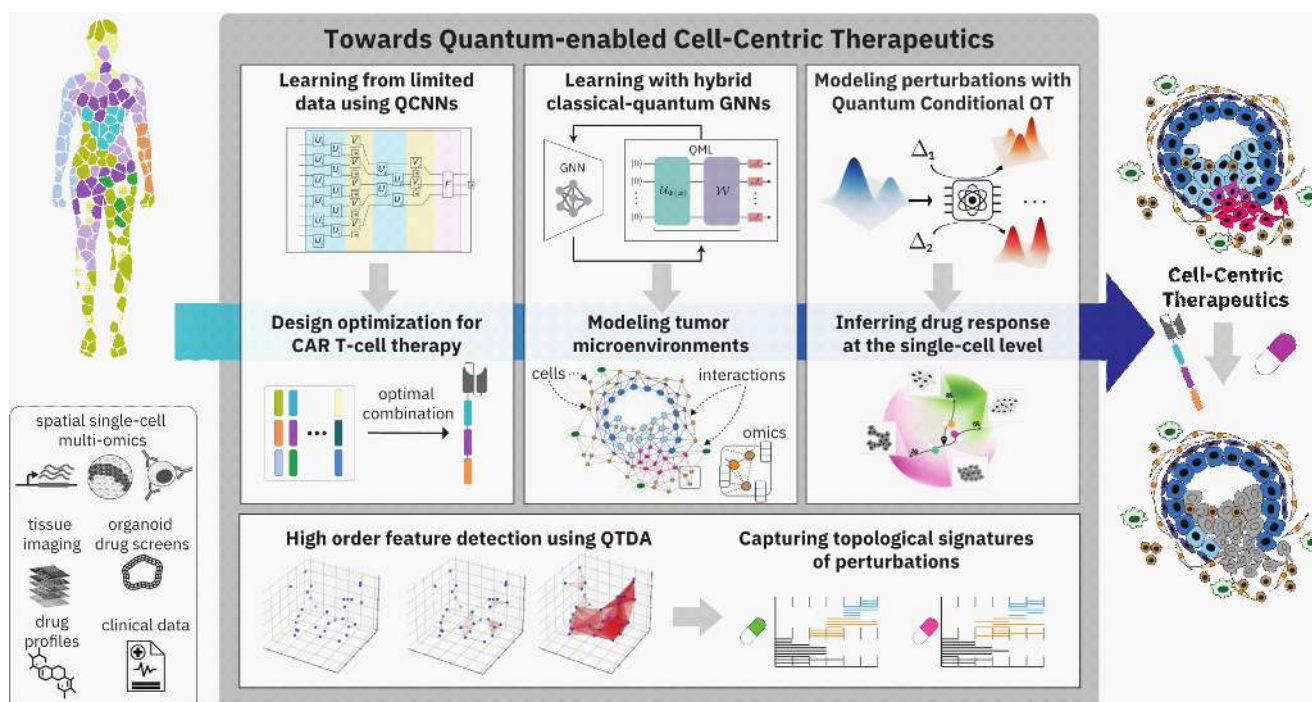
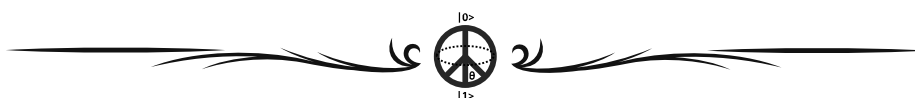


Figure 183: Overview of the various cell therapeutics applications of quantum computing. Source: [2715].



3.4.6 Bibliography and notes

These are the bibliographical references and notes for the **Life sciences** subsection.

- [608] [Quantum Methods for Neural Networks and Application to Medical Image Classification](#), by Jonas Landman, Natansh Mathur, Yun Yvonna Li, Martin Strahm, Skander Kazdaghli, Anupam Prakash, and Iordanis Kerenidis, arXiv, December 2022 (30 pages).
- [839] [Resource-efficient quantum algorithm for protein folding](#), by Anton Robert, Panagiotis Kl. Barkoutsos, Stefan Woerner, and Ivano Tavernelli, npj Quantum Information, February 2021 (5 pages).
- [2333] [Identifying Protein Co-regulatory Network Logic by Solving B-SAT Problems through Gate-based Quantum Computing](#), by Aspen Erlandsson Brisebois, Gordon Broderick, et al., arXiv, April 2025 (9 pages).
- [2334] [Compare Similarities Between DNA Sequences Using Permutation-Invariant Quantum Kernel](#), by Chenyu Shi, Gabriele Leoni, Mauro Petrillo, Antonio Puertas Gallardo, and Hao Wang, arXiv, March 2025 (11 pages).
- [2420] [Quantum Computing for Molecular Biology](#), by Alberto Baiardi, Matthias Christandl, and Markus Reiher, arXiv, June 2023 (76 pages).
- [2425] [Drug design on quantum computers](#), by Raffaele Santagati, Alan Aspuru-Guzik, Ryan Babbush, Matthias Degroote, Leticia Gonzalez, Elica Kyoseva, Nikolaj Moll, Markus Opel, Robert M. Parrish, Nicholas C. Rubin, Michael Streif, Christofer S. Tautermann, Horst Weiss, Nathan Wiebe, and Clemens Utschig-Utschig, Nature Physics, March 2024 (9 pages) (preprint on [arXiv](#)).
- [2587] [A systematic review of quantum machine learning for digital health](#), by Riddhi S. Gupta, Sally Shrapnel, et al., npj Digital Medicine, May 2025.
- [2588] [Quantum QSAR for drug discovery](#), by Alejandro Giraldo, Daniel Ruiz, Mariano Caruso, and Guido Bellomo, arXiv, May 2025 (9 pages).
- [2589] [Q2SAR: A Quantum Multiple Kernel Learning Approach for Drug Discovery](#), by Alejandro Giraldo, Daniel Ruiz, Mariano Caruso, Javier Mancilla, and Guido Bellomo, arXiv, June 2025 (5 pages).
- [2590] [QCS-ADME: Quantum Circuit Search for Drug Property Prediction with Imbalanced Data and Regression Adaptation](#), by Kangyu Zheng, Tianfan Fu, and Zhiding Liang, arXiv, March 2025 (5 pages).
- [2591] [Quantum Embedding of Non-local Quantum Many-body Interactions in Prototypal Anti-tumor Vaccine Metalloprotein on Near Term Quantum Computing Hardware](#), by Elena Chachkarova, Cedric Weber, et al., arXiv, November 2024 (14 pages).
- [2592] [The state of quantum computing applications in health and medicine](#), by Frederik F. Flöther, arXiv, August 2023 (10 pages).
- [2593] [Toward Density Functional Theory on Quantum Computers?](#), by Bruno Senjean, Saad Yalouz, and Matthieu Saubanière, arXiv, November 2022 (19 pages).
- [2594] [Towards quantum chemistry on a quantum computer](#), by B. P. Lanyon, J. D. Whitfield, G. G. Gillett, M. E. Goggin, M. P. Almeida, I. Kassal, J. D. Biamonte, M. Mohseni, B. J. Powell, M. Barbieri, Alan Aspuru-Guzik, and A. G. White, Nature Chemistry, January 2010 (6 pages) (preprint on [arXiv](#)).
- [2595] [Computational Drug Design & Molecular Dynamics: an HPC perspective](#), by Jean-Philip Piquemal, April 2020 (28 slides).
- [2596] [Tinker-HP: a massively parallel molecular dynamics package for multiscale simulations of large complex systems with advanced point dipole polarizable force fields](#), by Louis Lagardère, Jean-Philip Piquemal, et al., 2018 (17 pages).

- [2597] [Can Quantum Computing Play a Role in Drug Discovery? At least one Startup Thinks so](#), by James Dargan, 2020.
- [2598] [The Variational Quantum Eigensolver: A review of methods and best practices](#), by Jules Tilly, Hongxiang Chen, Shuxiang Cao, Dario Picozzi, Kanav Setia, Ying Li, Edward Grant, Leonard Wossnig, Ivan Rungger, George H. Booth, and Jonathan Tennyson, *Physics Reports*, November 2022 (128 pages).
- [2599] [Development of the Quantum Inspired SIBFA Many-Body Polarizable Force Field: Enabling Condensed Phase Molecular Dynamics Simulations](#), by Sehr Naseem-Khan, Jean-Philip Piquemal, et al., arXiv, May 2022 (50 pages).
- [2600] [Quantum and Quantum-inspired Methods for de novo Discovery of Altered Cancer Pathways](#), by Hedayat Alghassi, 2019 (27 pages).
- [2601] [Concepts of Artificial Intelligence for Computer-Assisted Drug Discovery](#), by Xin Yang, 2019 (75 pages).
- [2602] [Designing Peptides on a Quantum Computer](#), by Vikram Khipple Mulligan, September 2019 (20 pages).
- [2603] [Peptide Binding Classification on Quantum Computers](#), by Charles London, Konstantinos Meichanetzidis, et al., arXiv, November 2023 (17 pages).
- [2604] [Interface-Driven Peptide Folding: Quantum Computations on Simulated Membrane Surfaces](#), by Daniel Conde-Torres, Mariamo Mussa-Juane, Daniel Faílde, Andrés Gómez, Rebeca García-Fandiño, and Ángel Piñeiro, arXiv, January 2024 (9 pages).
- [2605] [Potential of quantum computing for drug discovery](#), by Y. Cao, J. Romero, and Alan Aspuru-Guzik, *IBM Journal of Research and Development*, November 2018 (18 pages) 📄.
- [2606] [Molecular Docking via Weighted Subgraph Isomorphism on Quantum Annealers](#), by Emanuele Triuzzi, Gianluca Palermo, et al., arXiv, February 2025 (19 pages).
- [2607] [Molecular docking via quantum approximate optimization algorithm](#), by Qi-Ming Ding, Yi-Ming Huang, and Xiao Yuan, arXiv, May 2024 (23 pages).
- [2608] [Quantum Approximate Optimization Algorithms for Molecular Docking](#), by Christos Papalitsas, Yanfei Guan, Shreyas Waghe, Athanasios Liakos, Ioannis Balatsos, and Vassilios Pantazopoulos, arXiv, March 2025 (15 pages).
- [2609] [Resource-Efficient Bio-Molecular Docking on a NISQ-era Digital Quantum Computer](#), by Tianqi Chen, Adrian M. Mak, Jianguo Li, Jian Feng Kong, Chandra Verma, and Sebastian Maurer-Stroh, arXiv, August 2026 (22 pages).
- [2610] [Quantum Machine Learning Framework for Virtual Screening in Drug Discovery: a Prospective Quantum Advantage](#), by Stefano Mensa, Emre Sahin, Francesco Tacchino, Panagiotis Kl. Barkoutsos, and Ivano Tavernelli, arXiv, April 2022 (16 pages).
- [2611] [Quadratic unconstrained binary optimization and constraint programming approaches for lattice-based cyclic peptide docking](#), by J. Kyle Brubaker, Helmut G. Katzgraber, et al., arXiv, December 2024 (30 pages).
- [2612] [Quantum-Inspired Machine Learning for Molecular Docking](#), by Runqiu Shu, Bowen Liu, Zhaoping Xiong, Xiaopeng Cui, Yunting Li, Wei Cui, Man-Hong Yung, and Nan Qiao, arXiv, February 2024 (8 pages).
- [2613] [QuASeR – Quantum Accelerated De Novo DNA Sequence Reconstruction](#), by Aritra Sarkar, Zaid Al-Ars, and Koen Bertels, arXiv, April 2020 (24 pages).
- [2614] [A Quantum Dot Plot Generation Algorithm for Pairwise Sequence Alignment](#), by Joseph Clapis, arXiv, July 2021 (29 pages).
- [2615] [Classical-to-Quantum Sequence Encoding in Genomics](#), by Nouhaila Innan and Muhammad Al-Zafar Khan, arXiv, April 2023 (58 pages).
- [2616] [Probabilistic Genotype-Phenotype Maps Reveal Mutational Robustness of RNA Folding, Spin Glasses, and Quantum Circuits](#), by Anna Sappington and Vaibhav Mohanty, arXiv, January 2023 (28 pages).
- [2617] [Quantum gate algorithm for reference-guided DNA sequence alignment](#), by G. D. Varsamis, J. Wong, et al., arXiv, August 2023 (19 pages).
- [2618] [Implementation of a quantum sequence alignment algorithm for quantum bioinformatics](#), by Floyd M. Creevey, Mingrui Jing, and Lloyd C. L. Hollenberg, arXiv, June 2025 (11 pages).
- [2619] [Quantum-parallel vectorized data encodings and computations on trapped-ions and transmons QPUs](#), by Jan Balewski, Mercy G. Amankwah, Roel Van Beeumen, E. Wes Bethel, Talita Perciano, and Daan Camps, arXiv, January 2023 (27 pages).
- [2620] [Single-Molecule Identification of Nucleotides Using a Quantum Computer](#), by Masateru Taniguchi, Takahito Ohshiro, and Tomofumi Tada, *J. Phys. Chem. B*, July 2023 📄.
- [2621] [Quantum gene regulatory networks](#), by Cristhian Roman-Vicharra and James J. Cai, *npj Quantum Information*, July 2023 📄 (preprint on [doi.org](#)).
- [2622] [Quantum computing for genomics: conceptual challenges and practical perspectives](#), by Aurora Maurizio and Guglielmo Mazzola, arXiv, July 2025 (14 pages).
- [2623] [Hype versus reality: What you can't do with DeepMind's AlphaFold in drug discovery](#), by Katyanna Quach, September 2022.
- [2624] [Highly accurate protein structure prediction with AlphaFold](#), by John Jumper, Demis Hassabis, et al., *Nature*, July 2021.
- [2625] [A perspective on protein structure prediction using quantum computers](#), by Hakan Doga et al., *IBM Research*, December 2023 (20 pages).
- [2626] [Accurate structure prediction of biomolecular interactions with AlphaFold 3](#), by Josh Abramson, Demis Hassabis, John M. Jumper, et al., *Nature*, May 2024.
- [2627] [AlphaFold Protein Structure Database in 2024: providing structure coverage for over 214 million protein sequences](#), by Mihaly Varadi, Damian Bertoni, Paulyna Magana, Urmila Paramval, Ivanna Pidruchna, Malarvizhi Radhakrishnan, Maxim Tsenkov, Sreenath Nair, Milot Mirdita, Jingi Yeo, Oleg Kovalevskiy, Kathryn Tunyasuvunakool, Agata Laydon, Augustin Židek, Hamish Tomlinson, Dhavanthi Hariharan, Josh Abrahamson, Tim Green, John Jumper, Ewan Birney, Martin Steinegger, Demis Hassabis, and Sameer Velankar, *Nucleic Acids Research*, November 2023 (8 pages).
- [2628] [Protein folding with an all-to-all trapped-ion quantum computer](#), by Sebastián V. Romero, Martin Roetteler, et al., arXiv, June 2025 (12 pages).
- [2629] [Protein folding on a 64 qubit trapped-ion hardware via counterdiabatic quantum optimization](#), by Alejandro Gomez Cadavid, Martin Roetteler, et al., arXiv, April 2026 (7 pages).
- [2630] [Efficient Quantum Protein Structure Prediction with Problem-Agnostic Ansatzes](#), by Hanna Linn, Rui-Hao Li, Alexander Holden, Abdullah Ash Saki, Frank DiFilippo, Tomas Radivoyevitch, Daniel Blankenberg, Laura García-Álvarez, and Göran Johansson, arXiv, September 2025 (18 pages).
- [2631] [Towards secondary structure prediction of longer mRNA sequences using a quantum-centric optimization scheme](#), by Vaibhaw Kumar, Alexey Galda, et al., arXiv, May 2025 (12 pages).
- [2632] [Quantum Algorithm for Protein Structure Prediction Using the Face-Centered Cubic Lattice](#), by Rui-Hao Li, Daniel Blankenberg, et al., arXiv, July 2025 (35 pages).
- [2633] [Molecular Quantum Computations on a Protein](#), by Akhil Shajan, Danil Kaliakin, Fangchun Liang, Thaddeus Pellegri, Hakan Doga, Subhamoy Bhowmik, Susanta Das, Antonio Mezzacapo, Mario Motta, and Kenneth M. Merz Jr, arXiv, December 2025 (40 pages).
- [2634] [Evolutionary-scale prediction of atomic level protein structure with a language model](#), by Zeming Lin et al., March 2023 (28 pages).

- [2635] [QFold: Quantum Walks and Deep Learning to Solve Protein Folding](#), by P A M Casares, Roberto Campos, and M A Martin-Delgado, arXiv, March 2022 (22 pages).
- [2636] [Peptide conformational sampling using the Quantum Approximate Optimization Algorithm](#), by Sami Boulebnane, Xavier Lucas, Agnes Meyder, Stanislaw Adaszewski, and Ashley Montanaro, arXiv, April 2022 (30 pages).
- [2637] [Protein Folding Neural Networks Are Not Robust](#), by Sumit Kumar Jha, Arvind Ramanathan, Rickard Ewetz, Alvaro Velasquez, and Susmit Jha, arXiv, September 2021 (8 pages).
- [2638] [Folding lattice proteins with quantum annealing](#), by Anders Irbäck, Lucas Knuthson, Sandipan Mohanty, and Carsten Peterson, arXiv, October 2022 (25 pages).
- [2639] [Using quantum annealing to design lattice proteins](#), by Anders Irbäck, Lucas Knuthson, Sandipan Mohanty, and Carsten Peterson, arXiv, February 2024 (33 pages).
- [2640] [Exploring Quantum Annealing for Coarse-Grained Protein Folding](#), by Timon Scheiber, Matthias Heller, and Andreas Giebel, arXiv, August 2025 (32 pages).
- [2641] [Resource analysis of quantum algorithms for coarse-grained protein folding models](#), by Hanna Linn, Isak Brundin, Laura García-Álvarez, and Göran Johansson, arXiv, December 2023 (18 pages).
- [2642] [Protein Structure Prediction with High Degrees of Freedom in a Gate-Based Quantum Computer](#), by Jaya Vasavi Pamidimukkala, Soham Bopardikar, Avinash Dakshinamoorthy, Ashwini Kannan, Kalyan Dasgupta, and Sanjib Senapati, Journal of Chemical Theory and Computation, November 2024 (12 pages)  (preprint on arXiv).
- [2643] [Logarithmic-scale variational quantum eigensolver for off-lattice protein structure prediction in continuous torsional angle space](#), by Fabio Cumbo, Bryan Raubenolt, Varun Puram, Natalie Katzenmeyer, Jayadev Joshi, and Daniel Blankenberg, arXiv, September 2026 (50 pages).
- [2644] We should estimate the resource to run this algorithm in a classical emulation intractable regime. Let's be generous and consider the use of 20 logical qubits. Using the QTF's scaling law ($n = \lceil \log_2 N \rceil$), 20 qubits would encode roughly 1,000,000 torsions which is good. Synthesizing the resulting 2.1 million arbitrary rotations into discrete gate sets generates a logical depth of ≈ 84 million cycles. With a $1 \mu\text{s}$ surface code cycle time, a single shot would take 84 seconds. Because the empirical CDF decoder must resolve 2^{20} basis states, an iteration would require at least 10^7 circuit shots, taking ≈ 26.6 years per energy evaluation. A 500-step optimization trajectory would demand over 13,000 years of continuous QPU runtime.
- [2645] [mRNA codon optimization with quantum computers](#), by Dillion M. Fox, Kim M. Branson, and Ross C. Walker, PLOS ONE, October 2021.
- [2646] [A resource-efficient variational quantum algorithm for mRNA codon optimization](#), by Hongfeng Zhang, Aritra Sarkar, and Koen Bertels, arXiv, May 2024 (11 pages).
- [2647] [Gate-based Quantum Computing for Protein Design](#), by Mohammad Hassan Khatami, Udson C. Mendes, Nathan Wiebe, and Philip M. Kim, arXiv, November 2022 (45 pages).
- [2648] [Digitized-Counterdiabatic Quantum Algorithm for Protein Folding](#), by Pranav Chandarana, Narendra N. Hegade, Iraitz Montalban, Enrique Solano, and Xi Chen, arXiv, December 2022 (14 pages).
- [2649] [Towards the simulation of large scale protein–ligand interactions on NISQ-era quantum computers](#), by Fionn D. Malone, March 2022 (16 pages).
- [2650] [Hybrid quantum-classical convolutional neural networks to improve molecular protein binding affinity predictions](#), by L. Domingo, M. Djukic, C. Johnson, and F. Borondo, arXiv, January 2023 (14 pages).
- [2651] [Quantum Algorithm for Protein Side-Chain Optimisation: Comparing Quantum to Classical Methods](#), by Anastasia Agathangelou, Dilhan Manawadu, Ivano Tavernelli IBM Quantum, IBM Research Europe - Zurich, The Hartree Centre, STFC, and Sci-Tech), arXiv, July 2025 (15 pages).
- [2652] [Predicting RNA Secondary Structure on Universal Quantum Computer](#), by Ji Jiang, Guo-Ping Guo, et al., arXiv, May 2023 (19 pages).
- [2653] [Toward end-to-end quantum simulation for protein dynamics](#), by Zhenning Liu, Xiantao Li, Chunhao Wang, and Jin-Peng Liu, arXiv, April 2025 (42 pages).
- [2654] [On Quantum Random Walks in Biomolecular Networks](#), by Viacheslav Dubovitskii, Aritra Bose, Filippo Utro, and Laxmi Pardia, arXiv, June 2025 (7 pages).
- [2655] [How to use quantum computers for biomolecular free energies](#), by Jakob Günther, Matthias Christandl, et al., arXiv, June 2025 (26 pages).
- [2656] [Exploring the use of quantum computing for facilitating spatially and temporally resolved models of a biological cell](#), by Muralikrishnan Gopalakrishnan Meena, Dileep Kishore, Jerry M. Parks, Luke Bertels, Dilipkumar N. Asthagiri, Travis Humble, Thomas L. Beck, and Mitchel J. Doktycz, arXiv, July 2026 (25 pages).
- [2657] [Quantum-classical convolutional neural networks in radiological image classification](#), by Andrea Matic, Maureen Monnet, Jeanette Miriam Lorenz, Balthasar Schachtner, and Thomas Messerer, arXiv, August 2022 (11 pages).
- [2658] [Quantum Biotechnology](#), by Nicolas P. Mauranyapin, Alex Terrason, and Warwick P. Bowen, arXiv, November 2021 (34 pages).
- [2659] [Parameter-Efficient Continuous-Variable Photonic Quantum Neural Networks for Edge Quantum AI: Demonstration in Oral Cancer Detection](#), by Akshay Bhagwan Sonawane, Sophie Choe, and Lakshman Tamil, arXiv, June 2026 (27 pages).
- [2660] [Quantum Medical Imaging Algorithms](#), by Bobak Toussi Kiani, Agnes Villanyi, and Seth Lloyd, arXiv, April 2020 (12 pages).
- [2661] [Error mitigation for a clinical data classification on the IonQ and the IBM quantum computers](#), by Sasan Moradi, 2022 (15 pages).
- [2662] [Quantum Support Vector Machine for Prostate Cancer Detection: A Performance Analysis](#), by Walid El Maouaki, Taoufik Said, and Mohamed Bennai, arXiv, March 2024 (14 pages).
- [2663] [Hybrid Quantum Neural Networks for Enhanced Breast Cancer Thermographic Classification: A Novel Quantum-Classical Integration Approach](#), by Riza Alaudin Syah, Irwan Alnarus Kautsar, Gunawan Witjaksono, and Haza Nuzly bin Abdull Hamed, arXiv, April 2026 (6 pages).
- [2664] [Hierarchical Quantum Control Gates for Functional MRI Understanding](#), by Xuan-Bac Nguyen, Hoang-Quan Nguyen, Hugh Churchill, Samee U. Khan, and Khoa Luu, arXiv, September 2024 (6 pages).
- [2665] [Quantum classical hybrid convolutional neural networks for breast cancer diagnosis](#), by Qiuyu Xiang, Xiaoyu Hua, et al., Scientific Reports, October 2024.
- [2666] [Towards a Utility-Scale Quantum Edge Detection for Real-World Medical Image Data](#), by Emmanuel Billias and Nikos Chrisochoides, arXiv, July 2025 (9 pages).
- [2667] [Quantum transfer learning for breast cancer detection](#), by Vanda Azevedo, February 2022 (14 pages).
- [2668] [Histopathological Cancer Detection Using Hybrid Quantum Computing](#), by Reek Majumdar, Biswaraj Baral, Bhavika Bhalgamiya, and Taposh Dutta Roy, arXiv, February 2023 (6 pages).
- [2669] [Configurable Algorithms for Histopathologic Cancer Detection on Quantum Hardware](#), by Nandika Goyal, Glen Uehara, and Andreas Spanias, arXiv, June 2026 (8 pages).
- [2670] [HybridQ: Hybrid Classical-Quantum Generative Adversarial Network for Skin Disease Image Generation](#), by Qingyue

- Jiao, Kangyu Zheng, Yiyu Shi, and Zhiding Liang, arXiv, June 2025 (10 pages).
- [2671] [COVID-19 detection on IBM quantum computer with classical-quantum transfer learning](#), by Erdi Acar et al., November 2020 (16 pages).
- [2672] [Quantum Annealing Feature Selection on Light-weight Medical Image Datasets](#), by Merlin A. Nau, Andreas K. Maier, et al., arXiv, February 2025 (10 pages).
- [2673] [Superpixel-Based QUBO for Scalable Quantum-Enhanced Medical Image Segmentation](#), by Mohammad Chalhoub, Mahdi Chehimi, Laia Domingo, Omar Alhusein, Ahmed Farouk, and Saif Al-Kuwari, arXiv, July 2026 (7 pages).
- [2674] [Quantum AI for Alzheimer's disease early screening](#), by Giacomo Cappiello and Filippo Caruso, arXiv, December 2024 (20 pages).
- [2675] [Brain Tumor Diagnosis Using Quantum Convolutional Neural Networks](#), by Muhammad Al-Zafar Khan, Abdullah Al Omar Galib, Nouhaila Innan, and Mohamed Bennai, arXiv, May 2025 (10 pages).
- [2676] [CQ CNN: A Hybrid Classical Quantum Convolutional Neural Network for Alzheimer's Disease Detection Using Diffusion Generated and U Net Segmented 3D MRI](#), by Mominul Islam, Mohammad Junayed Hasan, and M.R.C. Mahdy, arXiv, March 2025 (18 pages).
- [2677] [Quantum Transfer Learning to Boost Dementia Detection](#), by Sounak Bhowmik, Talita Perciano, and Himanshu Thapliyal, arXiv, July 2025 (5 pages).
- [2678] [Compression-Driven Anomaly Detection in Brain MRI Using an Interpretable Quantum Autoencoder](#), by Santanu Ganguly, Xing Liang, and Dimitrios Makris, arXiv, June 2026 (42 pages).
- [2679] [Quantum Model Parallelism for MRI-Based Classification of Alzheimer's Disease Stages](#), by Emine Akpınar and Murat Oduncuoglu, arXiv, January 2026 (45 pages).
- [2680] [From Foundation ECG Models to NISQ Learners: Distilling ECGFounder into a VQC Student](#), by Giovanni dos Santos Franco, Felipe Mahlow, Ellison Fernando Cardoso, and Felipe Fanchini, arXiv, March 2026 (16 pages).
- [2681] [Alz-QNet: A Quantum Regression Network for Studying Alzheimer's Gene Interactions](#), by Debanjan Konar, Neerav Sreekumar, Richard Jiang, and Vaneet Aggarwal, arXiv, August 2025 (13 pages).
- [2682] [Quantum Deep Learning for Mutant COVID-19 Strain Prediction](#), by Yu-Xin Jin, Xian-Min Jin, et al., arXiv, March 2022 (34 pages).
- [2683] [Improved clinical data imputation via classical and quantum determinantal point processes](#), by Skander Kazdaghi, Iordanis Kerenidis, Jens Kieckbusch, and Philip Teare, arXiv, December 2023 (14 pages).
- [2684] [Scalable On-Hardware Training of Quantum Neural Networks and Application to Clinical Data Imputation](#), by Natansh Mathur, Panagiotis K. Barkoutsos, Masako Yamada, Martin Roetteler, and Iordanis Kerenidis, arXiv, June 2026 (13 pages).
- [2685] [Classical and Quantum Machine Learning for Population-Level Prediction of Heat-Related Physiological Events](#), by Saul Gonzalez-Bermejo, Tommaso Albrigi, Borja Vazquez-Morado, Urko Regueiro-Ramos, Daniel Casado-Fauli, Sergi Consul-Pacareu, and Parfait Atchade-Adelomou, arXiv, April 2026 (9 pages).
- [2686] [Perspective on the Current State-of-the-Art of Quantum Computing for Drug Discovery Applications](#), by Nick S. Blunt et al., November 2022 (23 pages).
- [2687] [Towards quantum computing for clinical trial design and optimization: A perspective on new opportunities and challenges](#), by Hakan Doga, Daniel Blankenberg, et al., arXiv, April 2024 (22 pages).
- [2688] [Link prediction with continuous-time classical and quantum walks](#), by Mark Goldsmith, Sabrina Maniscalco, et al., arXiv, August 2022 (9 pages).
- [2689] [Quantum to Classical Neural Network Transfer Learning Applied to Drug Toxicity Prediction](#), by Anthony M. Smaldone and Victor S. Batista, arXiv, May 2024 (35 pages).
- [2690] [Quantum computing for oncology](#), by Siddhi Ramesh, Teague Tomesh, Samantha J. Riesenfeld, Frederic T. Chong, and Alexander T. Pearson, Nature Cancer, May 2024 (6 pages) 📄.
- [2691] [Quantum-machine-assisted Drug Discovery: Survey and Perspective](#), by Yidong Zhou, Zhiding Liang, et al., arXiv, February 2025 (17 pages).
- [2692] [Quantum Machine Learning in Drug Discovery: Applications in Academia and Pharmaceutical Industries](#), by Anthony M. Smaldone, Victor S. Batista, et al., arXiv, September 2024 (70 pages).
- [2693] [A hybrid quantum computing pipeline for real world drug discovery](#), by Weitang Li, Shengyu Zhang, et al., Scientific Reports, July 2024.
- [2694] [Using Quantum Computers and Simulators in the Life Sciences, Current Trends and Future Prospects](#), by Salil Gunashekar et al., RAND Europe, funded by Novo Nordisk Foundation, 2022 (132 pages).
- [2695] [Reducing the cost of energy estimation in the variational quantum eigensolver algorithm with robust amplitude estimation](#), by Peter D. Johnson, Jhonathan Romero, et al., arXiv, March 2022 (8 pages).
- [2696] [A universal programmable Gaussian Boson Sampler for drug discovery](#), by Shang Yu, Guang-Can Guo, et al., arXiv, March 2024 (11 pages).
- [2697] [Exploring the Advantages of Quantum Generative Adversarial Networks in Generative Chemistry](#), by Po-Yu Kao et al., May 2023 (12 pages).
- [2698] [Quantum-computing-enhanced algorithm unveils potential KRAS inhibitors](#), by Mohammad Ghazi Vakili, Alán Aspuru-Guzik, Alex Zhavoronkov, et al., Nature Biotechnology, January 2025.
- [2699] [Enhancing Drug Discovery: Quantum Machine Learning for QSAR Prediction with Incomplete Data](#), by Wei-Yin Chiang, Alex Zhavoronkov, et al., arXiv, January 2025 (12 pages).
- [2700] [Practical protein-pocket hydration-site prediction for drug discovery on a quantum computer](#), by Daniele Loco, Kisa Barkemeyer, Andre R. R. Carvalho, and Jean-Philip Piquemal, Physical Review Research, December 2025 (16 pages).
- [2701] [mRNA secondary structure prediction using utility-scale quantum computers](#), by Dimitris Alevras, Alexey Galda, et al., arXiv, May 2024 (12 pages).
- [2702] [Co-optimization of codon usage and mRNA secondary structure using quantum computing](#), by Dimitris Alevras, Alexey Galda, et al., arXiv, July 2025 (6 pages).
- [2703] [Hybrid quantum-classical machine learning for generative chemistry and drug design](#), by A.I. Gircha, A.S. Boev, K. Avchaciov, P.O. Fedichev, and A.K. Fedorov, arXiv, August 2023 (8 pages).
- [2704] [Robust Quantum Reservoir Computing for Molecular Property Prediction](#), by Daniel Beaulieu, Thomas Ehmer, Sheng-Tao Wang, Anh Pham, et al., arXiv, December 2024 (11 pages).
- [2705] [Large-scale quantum reservoir learning with an analog quantum computer](#), by Milan Kornjača, Sheng-Tao Wang, et al., arXiv, July 2024 (24 pages).
- [2706] [TACC Supercomputers Run Simulations Illuminating COVID-19, DNA Replication](#), by HPCWire, HPCwire, March 2020.
- [2707] [Computationally driven discovery of SARS-CoV-2 Mpro inhibitors: from design to experimental validation](#), by Léa El Khoury, 2022 (14 pages).
- [2708] [Covid-19: Quantum computing could someday find cures for coronaviruses and other diseases](#), by Todd R. Weiss, April 2020.

- [2709] [Thirteen tips for engaging with physicists, as told by a biologist](#), by Ken Kosik, Nature, January 2020.
- [2710] [How can quantum computing be applied in clinical trial design and optimization?](#), by Hakan Doga et al., Trends in Pharmacological Science, 2024 (12 pages).
- [2711] [Quantum-enhanced optimization for patient stratification in clinical trials](#), by Laia Domingo and Christine Johnson, arXiv, January 2026 (14 pages).
- [2712] [Quantum network medicine: rethinking medicine with network science and quantum algorithms](#), by Sabrina Maniscalco, Guillermo García-Pérez, et al., arXiv, June 2022 (15 pages).
- [2713] [Can a Quantum Support Vector Machine algorithm be utilized to identify Key Biomarkers from Multi-Omics data of COVID19 patients?](#), by Junggu Choi, Jae U. Jung, et al., arXiv, April 2025 (70 pages).
- [2714] [How quantum computing can enhance biomarker discovery](#), by Frederik F. Flöther, Filippo Utro, et al., arXiv, February 2025 (35 pages).
- [2715] [Towards quantum-enabled cell-centric therapeutics](#), by Saugata Basu, Laxmi Parida, et al., arXiv, August 2023 (29 pages).
- [2716] [Quantum Algorithm Outperforms Current Method of Identifying Healthy Livers for Transplant](#), by Terra Quantum, August 2024.
- [2717] [Predictive Health Analytics](#), by David Sahner, 2018 (54 slides).
- [2718] [Quantum computing for healthcare and pharmaceuticals - NQCC](#), by Sonali Mohapatra, NQCC, March 2025 (18 pages).
- [2719] [Quantum-inspired algorithm for radiotherapy planning optimization - PubMed](#), by Julia M Pakela, PubMed, January 2020 (14 pages).
- [2720] [Predicting Post-Surgical Complications with Quantum Neural Networks: A Clinical Study on Anastomotic Leak](#), by Vojtěch Novák, Vladimír Benčurik, et al., arXiv, June 2025 (50 pages).
- [2721] [Quantum machine learning framework for longitudinal biomedical studies](#), by Maria Demidik, Laxmi Parida, et al., arXiv, April 2025 (5 pages).

This **Life sciences** subsection contains 141 bibliographical references and notes containing at least 2,710 pages.

3.5 Energy

We cover here energy utility industries from exploration, production to distribution.

3.5.1 Power grid

There are many quantum methods to optimize power grids, based on either NISQ algorithms, quantum annealers or future FTQC QPUs [2722, 2723]. They compete with classical methods, often based on Mixed Integer Programming (MIP) techniques. The typical optimization problem is about finding an optimal power flow (OPF), based on IEEE test systems (like a 57-bus, a 73-bus, or a 300-bus case, the integer corresponding to the number of interconnected nodes), which are the reference graphs for benchmarking these solutions [864]. It can be experimented on both gate-based and quantum annealing QPUs [2724–2728]. Grid optimization is particularly important when the weight of intermittent renewable energy is high in the primary energy mix [2729, 2730]. There are many identified conditions to obtain a quantum advantage in solving power flow problems against the classical Newton-Raphson Load Flow (NRLF) algorithm [2731], particularly regarding quantum run times which must be in line with the grid constraints.

NISQ In the NISQ regime, IonQ and the DoE Oak Ridge National Laboratory did also prototype a unit commitment problem solving solution with 3, 10, and 26 power generating units. The quantum part used a QUBO-based variational approach and was tested with an emulator and with IonQ’s 36-qubit Forte system [2732].

NISQ **Aquarios** (Germany) tested in 2025 a variation of QAOA to solve a grid optimization problem [2733]. It is about solving the unit commitment problem which “*determines which power generators should operate within a power grid to meet a given estimated load at the lowest possible cost*”. “*It involves calculating the power flow through each transmission line, $\vec{p}$, based on the power input and output at each node (referred to as buses) in the power grid, p_i . The costs of this power flow problem can be expressed as $|\langle \vec{c}, \vec{p}_x \rangle|$, where $\vec{c}$ represents the linear cost factor for transmitting power over each transmission line*”. Although the unit commitment problem includes many additional cost factors and constraints in practical settings, the paper focused on a simplified version to serve as a proof of concept. It used a Quantum Simulation-based Optimization (QuSO) subset of Mixed-Integer Non-Linear Programming (MINLP), a very broad class of optimization problems. The algorithm was favorably compared to simulated annealing. It was tested with 7 power generating units and 57 buses (nodal point in the grid connecting various components of the power system), 20 qubits and 1024 QAOA layers, which is curiously high,

but is possible due to the usage of a classical state-vector emulator which has no limit in computing depth.

NISQ In another trial, the DoE tested a unit commitment solution on a D-Wave Advantage annealer and found satisfactory results, using 8 thermal generators, 27 renewable generators, and 24 buses [2734].

NISQ E.ON, IBM and academic partners in Canada applied a hardware-efficient copula-QAOA method to solve a 1D knapsack problem derived from a simplified Unit Commitment model. It was run on IBM Heron processors using up to 150 physical qubits using Q-CTRL error suppression. The algorithm utilized a classical lazy greedy warm-start to bias the initial quantum state toward feasible solutions. Despite successfully running circuits up to a depth of $p = 5$, the authors do not claim a practical quantum advantage, noting that natural problem instances of this size remain easily tractable for classical solvers like Gurobi. On a narrow set of artificially generated instances selected for their classical hardness, the reported solution quality was marginally better than the Gurobi and classical baselines. The comparison metric, solver settings, time budget, feasibility criterion and instance selection procedure are not fully specified, so no performance advantage can be inferred from it. The tested instances were limited to 150 items, while this problem can be easy to solve classically with up to 400 units [2735].

NISQ A China 2026 case was about optimizing the islanding challenge which partitions a stressed network to limit disrupted power transfer, while maintaining the operational integrity of every island. It demonstrated an algorithmic and resource advantage over prior quantum algorithms, allowing larger networks to be processed on NISQ systems using QAOA. It was a quantum-to-quantum advantage and run with IEEE benchmarks ranging from 9 to 89 buses using shallow circuits and practical sampling budgets, with power-flow validation confirming the operational feasibility of the resulting islands [2347]. A similar work was done in the USA and published in August 2026 [2736].

In March 2025, Inflection received \$6.2 million in funding from the US DoE Advanced Research Projects Agency-Energy (ARPA-E) agency to develop quantum-enhanced solutions for energy grid optimization [2737].

Other power grid and multi-energy optimization algorithms have been tested previously at low scales and on quantum annealing as well as gate-based QPUs, including with variational algorithms [2738–2744].

EDF is studying quantum computing very closely with various proof of concept projects. It covers quantum new materials simulations, material aging simulations particularly under radiations, safety statistics, combinatorial optimization for smart grids and electric vehicle battery charging optimization (teaming up with Pasqal in [2745]), job scheduling optimization [2746], and customer segmentation using Quantum Machine Learning. These projects are experimental and not yet

reaching any production-grade requirements. EDF and the University of Leiden also launched the Quantum probabilistic safety assessment (QPSA) project as part of the EU Flagship NE|AS|QC project [2747]. It is about the assessment of the safety and reliability of various safety-critical systems, such as nuclear plants, and is based on various tree search and quantum walks algorithms. Algorithms were tested on a 32-qubit IBM quantum emulator showing interesting results with more solution paths than classical solutions.

NISQ DTU tested in Denmark a grid optimization using an HHL algorithm on IBM QPUs with up to 7 qubits. They found scalability issues [2748]! A study from the **Los Alamos National Laboratory** also found that a quantum power flow algorithm based on the HHL algorithm would be slower than its best classical counterpart, when analyzed end-to-end, mainly due to the algorithm readout process [2749]. Other studies suggest that QAOA could scale well in some circumstances [2750].

Repsol (Spain) announced in 2024 the launch of its QCDI project to implement quantum technologies in data acquisition, enhancement, and utilization. It will create digital twins for various industrial applications in sustainable energy and other markets. The company has a revenue of \$3B.

analog **Iberdrola** (Spain) and Multiverse Computing publicized in July 2024 a pilot project in northern Spain aimed at optimizing the installation of grid-scale batteries. It was mixing quantum annealing and quantum inspired algorithms. We can suspect that the whole thing is classical, and not yet in production [2751].

Finally, let's mention a couple of initiatives in the field of power grid optimization. One is **QuEnergy** from QED-C [2752] and the other is a project run by **Phasecraft** and funded by the UK government [2753].

Outside of quantum computing, QKD could be used to protect key energy production and distribution infrastructures as is being investigated for hydropower and dams in the USA [2754].

3.5.2 Renewable energies

These use cases and proofs of concept deal with ways to optimize the production of renewable energies, with solar PV cells, hydrogen fuel cells operations, and windfarm operations.

Solar PV cells. In 2023, the DoE simulated the linear H_4 molecule using 20 trapped ion qubits which can absorb a single photon to produce two excited states [2755]. They used a quantum solver based on the Peeters-Devreese-Soldatov approach, which was developed at Pacific Northwest National Laboratory, another DoE lab. This molecule could help new photovoltaic cells with a better yield. Another study from 2025 studies the feasibility of computing band-gap energies of metal-halide-perovskite supercells used in photovoltaic solar

cells using VQE [2332], with no more than 10 qubits and no matched classical baseline. A team from Canada, Norway and Algeria published in 2024 a case study of a Quantum particle swarm optimization (QPSO) algorithm to optimize the capture of solar energy with photovoltaic panels. It was emulated classically, with an unspecified number of qubits [2340] **NISQ**.

Fuel cells. On the same Quantinuum hardware, another experiment dealt with the optimization of fuel cells membranes for the oxygen reduction reaction using In-Quanto Quantinuum H1-1 trapped-ion QPU. [2756]. In both cases, we were not in a quantum advantage regime and could thus consider these as classical computing advances **NISQ**.

Windfarms. In another documented case study related to the optimization of windfarm turbines layout, the classical Gurobi solution outperforms a gate-based VQE algorithm using 16 emulated qubits which can itself be more efficiently emulated classically than run on a NISQ quantum computer [2757]. A QUBO-based method was also tested in 2026 in the UK [2758] **NISQ**.

3.5.3 Oil exploration

Oil companies are still trying to optimize oil exploration, extraction and refinery.

BP is working on the design of algorithms for optimizing oil exploration. This involves using data from various sensors, particularly seismic sensors, to consolidate models for simulating geological layers under the ground. They joined in 2020 the **IBM Quantum Network** as an industry partner to get access to IBM's 65-qubit quantum computer, software and experts. It follows **ExxonMobil** who entered the program in 2019 as one of the major companies associated with IBM.

TotalEnergies also wants to optimize oil prospection and reserves evaluation using seismic probes. They plan to address complex optimization problems such as **MINLP** (Mixed Integer NonLinear Programming [2760]) to optimize refining, planning, production and transportation.

Quantum computing could also be used to detect oil spills using satellite imaging data with quantum Bayesian networks, using a 127-qubit IBM QPU [2761]. Oil spills are releases of oil into the environment, especially in the sea, due to human activity and oil infrastructure defects like pipeline leaks.

analog In the refinery realm, a China team devised in 2026 some quantum annealing based solver to optimize crude oil scheduling problems [2762].

3.5.4 Nuclear energy

Quantum computing could be used to simulate both fission [2763] and fusion energy reactions [2764, 2765]. At this point, we are still in the testing domain for the

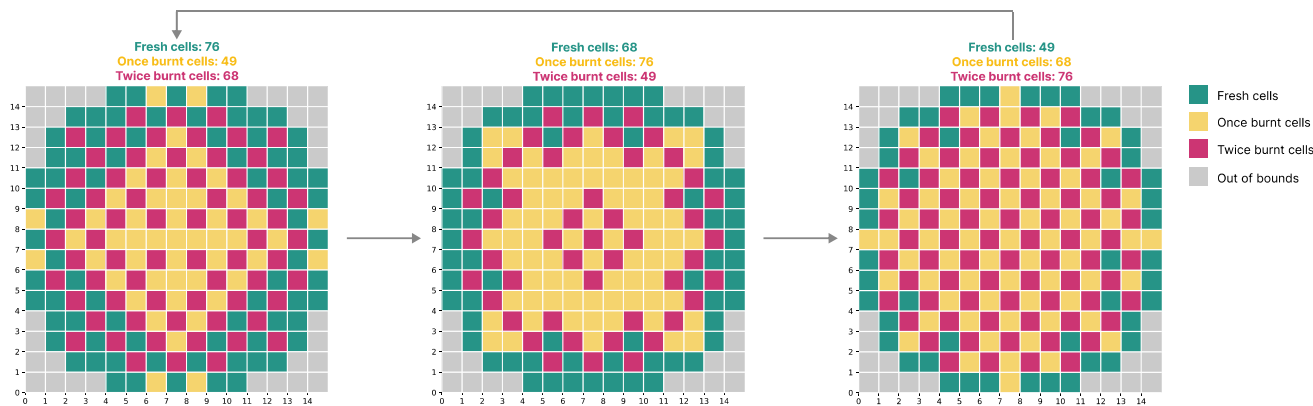


Figure 184: Source: a prototype solution to solve the nuclear plants in-core fuel management problem. It is a QUBO solution made for quantum annealers but is not yet in a quantum advantage regime. Source: [2759].

related use cases. It relates to the capability to simulate complex dynamic systems such as particles behavior in plasmas. Most published use cases show requirements for large scale FTQC systems with thousands of logical qubits [2766].

One 2023 Garcia-Ramos et al. review paper makes a good inventory of what could be achieved with quantum computing and particularly with quantum machine learning in the domain [2767].

- Quantum simulations can be used to model atomic nuclei and various nuclear interactions [2768].

- Quantum algorithms such as VQE (NISQ) and QPE (FTQC) could help solve quantum chemistry problems like with calculating nuclear energy levels or simulating nuclear reactions.
- QUBO can be used to optimize the fuel load problem in nuclear plants (Figure 184).
- QML algorithms or quantum optimization algorithms can extract some insights from nuclear physics experimental and operations data to extract patterns, detect particles in experiments, perform classifications, and make predictions. One optimization use-case is a quantum-inspired bin-packing algorithm to process nuclear waste [2769, 2770].



3.5.5 Bibliography and notes

These are the bibliographical references and notes for the **Energy** subsection.

- [864] [Solving Optimal Power Flow using a Variational Quantum Approach](#), by Thinh Viet Le, Mark M. Wilde, and Vassilis Kekatos, arXiv, August 2025 (22 pages).
- [2332] [A variational quantum eigensolver tailored to multi-band tight-binding simulations of electronic structures](#), by Dongkeun Lee and Hoon Ryu, arXiv, May 2025 (12 pages).
- [2340] [Quantum maximum power point tracking \(QMPPT\) for optimal solar energy extraction](#), by Habib Feraoun et al., December 2024 (17 pages).
- [2347] [PACE-QAOA: Physics-Constrained Quantum Optimization for Qubit-Efficient Power System Islanding](#), by Yuqi Jiang, Zhiding Liang, Qiang Guan, Yan Li, and Ganesh Kumar Venayagamoorthy, arXiv, August 2026 (43 pages).
- [2722] [Quantum Computing for Energy Management: A Semi Non-Technical Guide for Practitioners](#), by Jirawat Tangpanitanon, arXiv, November 2024 (25 pages).
- [2723] [Quantum Computing as a Game Changer on the Path towards a Net-Zero Economy: A Review of the Main Challenges in the Energy Domain](#), by Michela Ricciardi Celsi et al., Energies, February 2024 (22 pages).
- [2724] [Performance Comparison of Gate-Based and Adiabatic Quantum Computing for Power Flow Analysis](#), by Zeynab

Kaseb, Matthias Moller, Peter Palensky, and Pedro P. Vergara, arXiv, October 2025 (7 pages).

- [2725] [Quantum Hardware-in-the-Loop for Optimal Power Flow in Renewable-Integrated Power Systems](#), by Zeynab Kaseb and Pedro P. Vergara, arXiv, May 2025 (10 pages).
- [2726] [Benchmarking Hybrid Quantum-Classical Algorithms for Power Grid Optimization Problems](#), by Igor Gaidai and Rick Mukherjee, arXiv, July 2026 (16 pages).
- [2727] [Quantum Resource Estimation for Minimising Energy Grid Losses](#), by Camille de Valk, Milou van Nederveen, Koen Reerink, and Werner van Westering, arXiv, May 2026 (8 pages).
- [2728] [Quantum Technologies and Edge Devices in Electrical Grids: Opportunities, Challenges, and Future Directions](#), by Marjorie Hoegen, Ferdinanda Ponci, et al., arXiv, March 2026 (19 pages).
- [2729] [Hybrid Quantum-Classical Dispatching for High-Renewable Power Systems: A Noise-Resilient Variational Approach with Real-World Validation](#), by Fu Zhang and Yuming Zhao, arXiv, January 2026 (11 pages).
- [2730] [Quantum Annealing Continuous Optimisation in Renewable Energy](#), by Mansour T.A. Sharabiani, Vibe B. Jakobsen, Martin Jeppesen, and Alireza S. Mahani, arXiv, April 2022 (35 pages).
- [2731] [Conditions for Quantum Advantage in AC Power Flow](#), by Parikshit Pareek, Abhijith Jayakumar, Carleton Coffrin, and Sidhant Misra, arXiv, August 2026 (4 pages).

- [2732] [A New Hybrid Quantum-Classical Algorithm for Solving the Unit Commitment Problem](#), by Willie Aboumrاد, Evgeny Epifanovsky, et al., arXiv, April 2025 (10 pages).
- [2733] [Scaling Quantum Simulation-Based Optimization: Demonstrating Efficient Power Grid Management with Deep QAOA Circuits](#), by Maximilian Adler, Jonas Stein, and Michael Lachner, arXiv, May 2025 (9 pages).
- [2734] [Leveraging Quantum Computing for Accelerated Classical Algorithms in Power Systems Optimization](#), by Rosemary Barrass, Harsha Nagarajan, and Carleton Coffrin, arXiv, March 2025 (16 pages).
- [2735] [Constrained Quantum Optimization at Utility Scale: Application to the Knapsack Problem](#), by Naeimeh Mohseni, Julien-Pierre Houle, Ibrahim Shehzad, Giorgio Cortiana, Corey O'Meara, and Adam Bene Watts, arXiv, February 2026 (12 pages).
- [2736] [SPLIT-Q: A Scalable Sequential Quantum Computing Framework for Coherent Controlled Islanding](#), by Yuqi Jiang, Zhiding Liang, Qiang Guan, Yan Li, and Ganesh Kumar Venayagamoorthy, arXiv, August 2026 (11 pages).
- [2737] [Inflection Secures \\$6.2M ARPA-E Award to Advance Quantum-Powered Energy Grid Optimization](#), by Inflection, Inflection news, March 2025.
- [2738] [Quantum computation: Efficient network partitioning for large scale critical infrastructures](#), by Saikat Ray Majumder, Annarita Giani, Weiwei Shen, Bogdan Neculaes, Daiwei Zhu, and Sonika Johri, arXiv, February 2023 (5 pages).
- [2739] [Towards optimization under uncertainty for fundamental models in energy markets using quantum computers](#), by M.C. Braun, T. Decker, N. Hegemann, S.F. Kerstan, and F. Lorenz, arXiv, January 2023 (16 pages).
- [2740] [Power network optimization: a quantum approach](#), by Giuseppe Colucci, Stan van der Linde, and Frank Phillipson, arXiv, December 2022 (7 pages).
- [2741] [Noise-Resilient Quantum Power Flow](#), by Fei Feng, Yifan Zhou, and Peng Zhang, arXiv, November 2022 (6 pages).
- [2742] [Integrating quantum and classical computing for multi-energy system optimization using Benders decomposition](#), by Ludger Leenders, Martin Sollich, Christiane Reinert, and André Bardow, arXiv, July 2024 (15 pages).
- [2743] [Quantum Optimization for the Future Energy Grid: Summary and Quantum Utility Prospects](#), by Jonas Blenninger, Benedikt Wimmer, et al., arXiv, March 2024 (12 pages).
- [2744] [Opportunities for quantum computing within net-zero power system optimization](#), by Thomas Morstyn et al., June 2024 (22 pages).
- [2745] [Implementing transferable annealing protocols for combinatorial optimisation on neutral atom quantum processors: a case study on smart-charging of electric vehicles](#), by Lucas Leclerc, Constantin Dalyac, Pascale Bendotti, Rodolphe Griset, Joseph Mikael, and Loïc Henriët, arXiv, May 2025 (23 pages).
- [2746] [State-space reduction techniques exploiting specific constraints for quantum search Application to a specific job scheduling problem](#), by Rodolphe Griset, Ioannis Lavdas, and Jiri Guth Jarkovsky, arXiv, January 2025 (31 pages).
- [2747] [Quantum probabilistic safety assessment \(QPSA\)](#), by EDF et al., 2024.
- [2748] [Quantum Computing for Power Flow Algorithms: Testing on real Quantum Computers](#), by Brynjar Sævarsson, Spyros Chatzivasileiadis, Hjörtur Jóhannsson, and Jacob Østergaard, arXiv, July 2022 (8 pages).
- [2749] [Limitations of Fault-Tolerant Quantum Linear System Solvers for Quantum Power Flow](#), by Parikshit Pareek, Abhijith Jayakumar, Carleton Coffrin, and Sidhant Misra, arXiv, September 2025 (10 pages).
- [2750] [Assessing Quantum Computing Performance for Energy Optimization in a Prosumer Community](#), by Carlo Mastroianni, Andrea Vinci, et al., arXiv, November 2023 (14 pages).
- [2751] [Iberdrola and Multiverse Computing Announce Pilot Project Success to Optimize Battery Installation in the Grid](#), by Matt Swayne, The Quantum Insider, July 2024.
- [2752] [QuEnergy: Exploring the role of quantum computing for the electric grid](#), by QED-C, November 2022.
- [2753] [Phasecraft wins £1.2 million UK Government Contract to Develop Quantum Algorithms to Optimise Energy Grids](#), by Matt Swayne, The Quantum Insider, February 2024.
- [2754] [Quantum Key Distribution for Critical Infrastructures: Towards Cyber Physical Security for Hydropower and Dams](#), by Adrien Green, Jeremy Lawrence, George Siopsis, Nicholas Peters, and Ali Passian, arXiv, October 2023 (20 pages).
- [2755] [Modeling Singlet Fission on a Quantum Computer](#), by Daniel Claudino, June 2023 (6 pages).
- [2756] [Platinum-based Catalysts for Oxygen Reduction Reaction simulated with a Quantum Computer](#), by Cono Di Paola, Evgeny Plekhanov, Michal Krompiec, Chandan Kumar, Emanuele Marsili, Fengmin Du, Daniel Weber, Jasper Simon Krauser, Elvira Shishenina, and David Muñoz Ramo, arXiv, April 2024 (29 pages).
- [2757] [Investigating techniques to optimise the layout of turbines in a windfarm using a quantum computer](#), by James Hancock, Matthew J. Craven, Craig McNeile, and Davide Vadacchino, arXiv, February 2024 (26 pages).
- [2758] [Investigating methods to solve large windfarm optimization problems with a minimum number of qubits using circuit-based quantum computers](#), by James Hancock, Matthew Craven, and Craig McNeile, arXiv, December 2025 (43 pages).
- [2759] [Quantum and quantum-inspired optimization for an in-core fuel management problem](#), by Sergey R. Usmanov, Aleksey K. Fedorov, et al., arXiv, August 2023 (3 pages).
- [2760] [Quantum Computing and Non-Linear Integer Optimization](#), by Sridhar Tayur, February 2019 (42 slides).
- [2761] [Quantum Bayesian Networks for Machine Learning in Oil-Spill Detection](#), by Owais Ishtiaq Siddiqui, Nouhaila Innan, Alberto Marchisio, Mohamed Bennai, and Muhammad Shafique, arXiv, April 2025 (8 pages).
- [2762] [Solve Crude Oil Scheduling Problems by Using Quantum-Classical Hybrid Algorithms](#), by Jian Yang, Bohang Wang, Lina Wang, Jiacheng Chen, Gaoxiang Tang, Zihan Deng, Wending Zhao, and Xianfeng Cai, arXiv, April 2026 (38 pages).
- [2763] [A quantum-classical co-processing protocol towards simulating nuclear reactions on contemporary quantum hardware](#), by Francesco Turro, Irfan Siddiqi, et al., arXiv, September 2023 (12 pages).
- [2764] [Quantum Computing for Fusion Energy Science Applications](#), by I. Joseph, Y. Shi, M. D. Porter, A. R. Castelli, V. I. Geyko, F. R. Graziani, S. B. Libby, and J. L. DuBois, arXiv, December 2023 (42 pages).
- [2765] [Quantum Computing Perspective for Electromagnetic Wave Propagation in Cold Magnetized Plasmas](#), by Efstratios Koukoutsis, Kyriakos Hizanidis, George Vahala, Min Soe, Linda Vahala, and Abhay K. Ram, arXiv, November 2023 (13 pages).
- [2766] [Quantum computation of stopping power for inertial fusion target design](#), by Nicholas C. Rubin, Andrew D. Baczewski, et al., arXiv, August 2023 (37 pages).
- [2767] [Nuclear Physics in the Era of Quantum Computing and Quantum Machine Learning](#), by J.E. García-Ramos, A. Sáiz, J.M. Arias, L. Lamata, and P. Pérez-Fernández, arXiv, July 2023 (21 pages).
- [2768] [Quantum nuclear dynamics on a distributed set of ion-trap quantum computing systems](#), by Anurag Dwivedi, A. J. Rasmusson, Philip Richerme, and Srinivasan S. Iyengar, arXiv, June 2024 (44 pages).
- [2769] [Quantum and quantum-inspired optimization for solving the minimum bin packing problem](#), by A. A. Bozhedarov,

A. S. Boev, S. R. Usmanov, G. V. Salahov, E. O. Kiktenko,
and A. K. Fedorov, arXiv, January 2023 (7 pages).

[2770] [Quantum Information Science and Technology for Nuclear
Physics. Input into U.S. Long-Range Planning, 2023](#), by

Douglas Beck, Fanyi Zhao, et al., arXiv, February 2023 (26
pages).

This **Energy** subsection contains 53 bibliographical references
and notes containing at least 853 pages.

3.6 Transportation and logistics

Beyond energy matters mentioned above, transportation industries are mainly interested in algorithms for optimizing complex systems [2771, 2772]. Let's look at what can be done with road, rail, air and maritime transportation.

3.6.1 Automotive

Quantum computing could be implemented in many domains to improve both vehicle production and their operations. See an overview of case studies in [2773] and an economic-impact study of quantum technologies in transportation from a report by the UK government Department for Transport [2774]. It plans for the creation of £8B of economic value by 2035, just in the UK. That is a stretch.

Design

In the product design phase, quantum computing could help optimize vehicle fluid dynamics thanks to solving complex partial differential problems (*aka* Navier Stokes equations), minimizing drag, and improving battery/fuel efficiency. It could also simulate the physics of various materials to improve weight/strengths vehicle ratios and design new batteries (but in the very long term).

Manufacturing

In the manufacturing stage, quantum computing could potentially solve problems like supply-chain optimizations, warehouse robots routing, improving the accuracy of demand forecasting both with end-users and for their suppliers and inventory optimization.

One famous use case is the BPSP, or Binary Paint Shop Problem, a combinatorial optimization problem which could be solved with variants of quantum heuristic approaches using QAOA, after being reduced to weighted MaxCut and Ising models. These are eXpressive QAOA (XQAOA) and Recursive QAOA (RQAOA). Given $2n$ cars, each of n models repeated twice, we must assign one of two colors to each car, ensuring that both instances of each model are painted differently, while minimizing the number of consecutive color changes in the painting sequence. This swap count correlates directly with operational costs and efficiency in the automotive paint shop. The main performance metric is the *paint swap ratio*, defined as the mean number of color-change events per car. Classical approaches, including Recursive Greedy (RG) and Recursive Star Greedy (RSG), achieve paint swap ratios of approximately 0.4 and 0.361, respectively. QAOA could fare better, but with impractical circuit depths.

In [2775], a research team from Singapore benchmarked in 2025 the use of XQAOA against classical methods. It requires n qubits and a shallow circuit of about 10 layers. It could theoretically reach a paint shop swap

ratio of 0.357, beating all classical optimizers. It could exhibit a speedup starting at $n=128$ car models while avoiding the barren plateau side effect. Such paper is however missing key elements like estimating computing time on real existing or future hardware, estimating the cost of the classical optimizer and of the quantum error correction required once you exceed a couple hundred cars. It looks like a clear quantum advantage in speedup and swap ratio would require hundreds to thousands of logical qubits to match industry needs with batches of 2K to 4K cars **FTQC**.

analog Volkswagen Data:Lab published two papers in 2022 coauthored with Terra Quantum related to hybrid computing associated with D-Wave quantum annealers in two areas: the optimization of assembly line work-flow scheduling [2776] and hybrid machine learning to enhance image recognition systems used for car classification [2777]. It is still not clear whether these applications really went into production. Volkswagen also experimented small-scale algorithms with D-Wave for optimizing vehicle recommendations and optimizing cars painting planning.

FTQC Volkswagen added another paper in 2026 along with Kipu Quantum, on the use of HUBO and BF-DCQO, with realistic workloads requiring circuits with a size of 10^{10} , requiring large scale FTQC.

FTQC General Motors made in 2025 a resource estimation for modeling transition metals and a catalyst using a QPE, involving TiO (titanium oxide), MnO (manganese oxide), and FeO (iron oxide), and a Pd zeolite catalyst fragment Z_2Pd , with $Z=Al_2Si_{22}O_{48}$, which could be used for carbon emissions reduction. [2778]. With a physical error rate of 10^{-4} , which is quite stretch whatever the qubit type, 10s of millions of physical qubits would be needed and a reasonable computing time of about 12 hours for superconducting qubits.

Routing

Quantum computing could potentially help optimize cars and truck routing, with traffic flow optimization (TFO) [2779, 2780], minimizing traffic congestion [2781], traffic forecasting [2782] or multimodal fleet operations [2783] and of course, solving the famous Traveling Salesperson Problem [2784]. But considering the state of the art of the related algorithms, it is quite unsure we'll obtain some practical quantum advantage there, and above all, not with solutions solving these problems in milliseconds as mentioned in [2785].

analog Solving the TSP problem without constraints works rather classically. Quantum computing added value could come from solving the TSP problem with additional constraints. In a 2025 case, a D-Wave annealer was used to solve the TSP problem with time constraints, and it didn't perform well [2786].

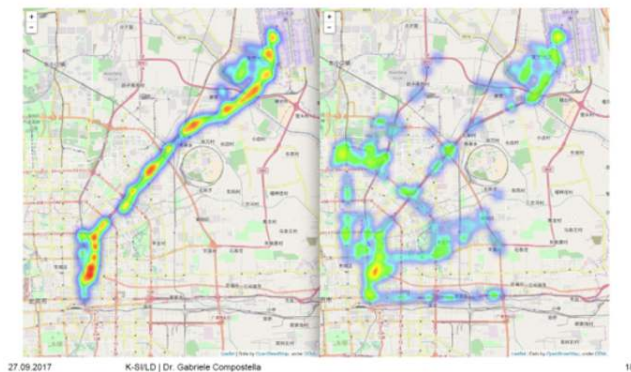


Figure 185: A 2017 proof of concept from Volkswagen and D-Wave about optimizing the route of 418 cabs in Beijing, using training data coming from 10,357 cabs. Source: [2365].

analog An experiment was done in 2017 by **Volkswagen** on D-Wave annealers [2787]. Its goal was to optimize the routes of a (virtual) cab fleet in Beijing. The experiment was using the **T-Driven data set** published by Microsoft in 2008 which describes the routes of 10,357 cabs. The algorithm used was QUBO (Quadratic Unconstrained Binary Optimization). The diagram in Figure 185 shows the result of the optimization of the route of 418 cabs making the journey from the city center to the airport considering the route of 10,357 vehicles [2365]. A less ambitious version of this algorithm was used to optimize the individual travel routes of nine public-transit buses during the November 2019 Web Summit event in Lisbon.

FTQC The most relevant use case is the more generic and complex Capacitated Vehicle Routing Problem (CVRP) where one vehicle must collect goods at various places with its limited capacity [2788–2791] as well as solving irregular strip packing problems [2792, 2793]. One 2025 paper from Germany found that solving a CVRP problem in a useful regime using a HUBO formulation could bring a quantum advantage with at least 7,685 logical qubits and 3×10^{-9} error rates, far off in the FTQC regime [2794].

NISQ In 2023, **Honda** and Terra Quantum published a paper on how to solve the escape routing problem with a hybrid quantum computing solution involving some quantum machine learning [2795]. It was tested on 25 IonQ qubits through AWS.

NISQ In 2023, a team at **TotalEnergies** and LIRMM proposed an algorithm to solve the problem of “Fleet Conversion” which is about optimizing and reducing the carbon emissions and cost of operating a heterogeneous fleet of vehicles combining electric vehicles and traditional ones. It was formulated as a Maximum Weighted Independent Set (MWIS) problem represented in a QUBO variant and implemented in gate-based mode [2796]. The prototype was tested with 7 qubits, far from any quantum advantage.

analog **Toyota** tested in 2025 a small-scale QUBO solution on a D-Wave annealer for optimizing EV routes

and charging locations, but with not much difference with classical simulated annealing [2797].

analog Trucks routing algorithms were explored by **Accenture** and **Denso** using D-Wave annealers. Annealers are so far in a better position to solve these problems than existing superconducting gate-based qubits systems from IBM.

analog In an adjacent domain, **Toyota** worked with D-Wave to optimize traffic lights on a 50×50 road grid, using a D-Wave 2000Q [2798, 2799]. Small scale routing problems were also tested in gate-based mode at CQT in Singapore in 2023 [2800].

NISQ One last and quite original use case is the **Volkswagen** “Windbreaking-as-a-Service” (WaaS) in which larger “windbreaker” vehicles provide aerodynamic shelter for “windsurfer” vehicles through vehicle platooning an advanced driving method where a group of connected vehicles travel closely together in a convoy using wireless communication, automated braking, and acceleration. It is reducing drag and fuel consumption [2801]. The solution was mixing a QAOA quantum algorithm and Gurobi classical optimizer, and was tested with up to 16 qubits in 2025 (Figure 186). In 2026, Volkswagen tested classical heuristics to solve the same problem [2802].



Figure 186: A quantum solution is proposed by Volkswagen and Jülich Supercomputing Center to reduce drag and fuel consumption for vehicles driving behind larger ones. Probably at the expense of driving slower! Source: [2801].

In 2026, multiple preprints were published on various ways to solve the regular TSP problem. The result range from honest negative or baseline results on both gate-based systems and quantum annealers [2803–2806], encoding or resource contributions with no runtime claim not end-to-end resource estimations [2807, 2808], a Grover-style heuristic proof of concept with no accounting for the initialization cost and poor practical Grover scalability [2809], and one provable exact-algorithm speedup using a divide-and-conquer framework requiring an FTQC setup and a QRAM-based oracle to access classical data [2810] **NISQ** **FTQC**.

Likewise, papers were also published on solving CVRP.

One improves QAOA applicability on a VRP problem evaluated against standard QAOA with feasibility rising to about 99.7%. The solution is simulated on small instances with no classical solver comparison. The added value is some incremental QAOA feasibility engineering [2811] **NISQ**.

In 2026, Unisys researchers published an empirical benchmark of using D-Wave annealers to solve CVRP, studying how problem size and constraint density degrade solution quality. It makes no advantage claim

and, instead documents where commercial QA quality breaks down [2812] **analog**.

Another small-scale optimized QAOA doesn't fare better than the best classical optimizers [2813] **NISQ**.

There are many other routing-related problems that researchers are looking to solve with quantum computers: the Bi-Objective Traveling Thief Problem (BI-TTP), a multi-objective optimization problem that simultaneously optimizes travel cost and item profit [2814], predicting the taxi destination within urban road network [2815], predicting travel time using a QAE variant [2816], and selecting vehicles following a single predefined route, such that route-level coverage utility is maximized while redundant spatial overlaps are limited although, not yet with a demonstrable quantum advantage [2817] **NISQ**.

Autonomous vehicles

Machine learning based solutions could under some circumstances benefit from QC, for improving image pattern recognition.

NISQ One less credible use case is autonomous vehicle control whether it is based on some machine learning technique or not [2818, 2819]. It doesn't make much sense for quantum computing due, at least, to the data-loading problem and QPU form factors [2820, 2821]. In the case of "quantum" image recognition, ML training would be quantum and variational. One speculative workflow would train the model variationally, then distill it into a classical surrogate, an emulator or a tensor-network version, to run on the embedded systems in the car, with the goal of obtaining image recognition tools with better accuracy rates. It would still have to be shown that the classical deployment preserves the claimed accuracy and that the quantum training step adds any value in the first place.

NISQ **Hyundai** worked with IonQ to improve 3D object detection with quantum computing. In 2023, IonQ demonstrated a basic image recognition solution of 43 road signs which is a very basic test for a classical convolutional network running on a simple mobile device (smartphones, laptops, whatever) [2822]. It plans to expand it to recognize pedestrians and cyclists. They believe they can do better with quantum computers than with classical NPU's (the neural processing units now embedded in most CPUs and mobile chips).

The deployment of autonomous-vehicle fleets is a nice target application for quantum computers. The more autonomous the vehicles are, the greater the need for automation and route coordination. The problems to be solved will be to determine step-by-step the vehicle fleet routes in order to optimize each of these vehicles and passengers the travel time. There is indeed an insufficient basis for estimating the size of quantum computers needed to practically handle such large-scale problems. How many qubits would be needed to optimize a fleet of hundreds or even millions of autonomous vehicles?

Manufacturers

Let's now look at how various car makers are approaching quantum computing.

Daimler started to work with IBM, with applications for logistics and planning optimization and everything to do with autonomous vehicle routing at the forefront. In 2018, they also launched an initiative with IBM to develop lithium-sulfur batteries, which improve energy density and make it possible to do without metals such as cobalt and nickel. All of this will be achieved through quantum simulation.

NISQ **BMW** is also willing to learn how to use quantum computing in various tasks, one being the optimization of its spare parts supply chain [2823]. They are partnering with Quantinuum to do this, Zapata Computing and Entropica Labs. They started using the Honeywell (now Quantinuum) trapped ions based H0 and H1 with 10 qubits. They used a Recursive Quantum Approximate Optimization Algorithm (R-QAOA) to manage their combinatorial problem. The quality of these trapped-ion qubits made the trial promising although of course not usable at all given the small number of available qubits [2824].

analog In May 2022, BMW started a partnership with Pasqal to solve differential equations to better compute metal forming applications modeling. They also work with QuEra to solve MIS-based optimization problems. It reformulates a quantum-informed recursive optimization (QIRO) using preparation data coming from classical emulations of a QAOA algorithm and running on a QuEra analog quantum simulator [2825].

NISQ **Bosch Automotive Electronics** plant in Madrid and Multiverse announced in 2022 that they were exploring the creation of a digital twin of a factory with quantum computing. They plan to capture data to assess the performance of individual equipment and optimize quality control and overall efficiencies, including energy consumption and waste management. What they call digital twins are actually optimization systems.

analog **BASF** published in 2024 a case study showing how a D-Wave Leap's Hybrid Solver could solve a vehicle routing problem with 150, 000 variables with results on par and with some speedups as compared with a classical Gurobi solver [2826].

In 2021, the German auto industry (BMW, Mercedes-Benz, Volkswagen, and Bosch) and research organizations (DFKI who runs contract research on AI and Forschungszentrum Jülich) even launched the **Q(AI)2** project to test AI applications for quantum computers [2827]. It particularly wants to study the usage of AI to solve various manufacturing optimization problems.

3.6.2 Rail

Most rail use cases deal with solving optimization problems.

analog An experiment was done by **Ferrovie dello stato Italia** to optimize train arrivals in railway stations, also to minimize passengers' connection times, again with D-Wave. In Poland, researchers compared D-Wave annealers and classical methods to optimize train dispatch and found that current NISQ systems do not outperform the classical CPLEX optimizer [2828, 2829].

Another project involving **OQC** and **Q-CTRL** and commissioned by the **UK Department of Transport** is supposed to find ways to make Britain's trains run more efficiently. Thanks to Q-CTRL's quantum error suppression technique, the solution ran 6× faster than existing quantum solutions, but you'd need to wait till 2028 based on IBM's roadmap to obtain some quantum advantage [2830].

AlphaRail

USA, 2000



AlphaRail is a railways software company using machine learning and quantum computing to improve railways operations. They are relying on quantum and quantum-inspired approaches to solve routing and scheduling optimization problems.

3.6.3 Air

With airlines, the current focus is about optimizing aircraft-fleet planning, to maximize the capacity to meet demand while optimizing the aircraft fill rate.

Airport gate assignment

Quantum computing can enable optimizing airport and aircraft-gate management, the so-called flight gate assignment problem [2831], to minimize passenger waiting time, as tested by **DLR** in Germany [2832] (Figure 187). The flight-gate assignment problem could also potentially be solved with a gaussian boson sampler [2833]. In the formulation used here, it is an NP-hard problem that is difficult to deal with using conventional algorithms. Similar small-scale tests have been run elsewhere such as with IBM [2834, 2835] and IonQ [2836]. The University of Hamburg and Lufthansa launched a similar 3-year project in 2024 [2837] **NISQ**.

Flight route optimization

Solving the Tail Assignment Problem (TAP) is another quest. Researchers from **Chalmers University** in Sweden prototyped a promising QAOA hybrid algorithm solving the “tail assignment problem”, which is the task of assigning individual aircraft to a given set of flights, minimizing the overall cost for the airline. It worked with only 2 qubits, to optimize two flights and would scale well as flights are added [2838]. A similar QAOA-based approach was tested in Spain in 2024 [2838] **NISQ**.

These are needs that can be addressed both by machine learning algorithms to consider the past or with quantum optimization algorithms based on a description of the parameters of the problem. The former does prediction and the latter simulation. Simulations avoid the back-mirror bias that can be induced by prediction methods based on past data. A combination of the two methods is possible.

Airline operations

Two other quantum computing cases were tested in 2025, one for optimizing airline crew pairing [2839] and the other, about aircraft loading optimization [2840].

Passenger drones routing

How about designing autonomous quantum flying drones using a quantum computer, a project from Indian researchers [2841]? It makes no sense from a practical standpoint, at least due to real-time constraints. Since the quantum computer can't sit in the drone, a fortiori, one from IBM, there's an inevitable lag in the communication between the drone and the quantum computer which, by the way, should be entirely dedicated to a single drone to work real-time.

In Japan, **Sumitomo** launched in June 2021 a pilot experiment for optimizing flight routes for urban air mobile vehicles (air taxis and unmanned drones). They were to rely on resources from Tohoku University. No precision on the problem sizing and on which quantum system they plan to pilot their solution even if D-Wave is a likely candidate. They published a use case based on a D-Wave quantum annealer in 2024 [2842]. Another case on drone routing was published by researchers from Spain, Greece and Germany in 2025 [2843].

Manufacturers

Airbus has been involved in quantum computing since 2015. They expect to use it in various fields like on quantum chemistry, fighting corrosion [2844], quantum optimization (flight trajectories, satellite mission planning), quantum machine learning (anomaly detection, reinforcement learning) and with quantum solvers for engineering (fluid mechanics simulation, aeroacoustics) [2845].

In 2015, one of their teams based in Newport in the United Kingdom began working on the subject. In 2016, Airbus Ventures invested in the American startup QC Ware. They experimented with the use of a D-Wave for fault tree analysis (FTA), which is used to determine the origin of complex failures with a gain of a factor of 4 compared with traditional methods. This is a difficult NP-difficult combinatorial problem that is potentially easier to solve in quantum programming.

In January 2019, Airbus launched its “Quantum Computing Challenge”, a way to outsource the development of quantum solutions to help them solve their business problems, in fluid mechanics, differential equations,

Passenger Flows

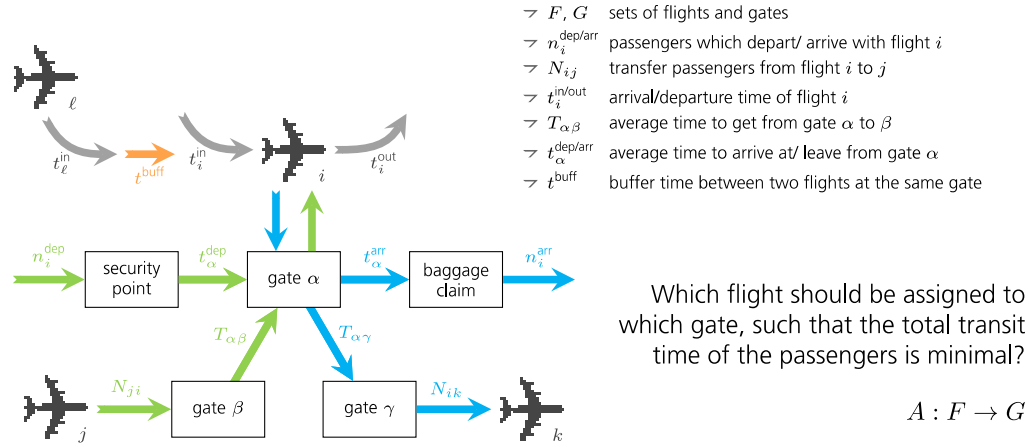


Figure 187: The passenger flow process tested by DLR in 2019. Source: [2832].

flight optimization, wing design, cargo bay filling, etc. [2846].

As of May 2019, 475 teams from 57 countries had competed in this challenge. They came mainly from the USA and India, followed by Europe. They announced the challenge results in October 2020 with five finalists selected out of 36 contestants.

As of 2023, Airbus had conducted some pilot proof of concepts projects:

- Improving catalytic reactions with pure platinum and platinum-capped cobalt in fuel cell stack, developed by Aerostack, a JV been Airbus and ElringKlinger. A VQE-ADAPT based hybrid algorithm was tested on Quantinuum H1-2 QPU and H1-2E classical emulator with 20 qubits. Conclusion: “We show that ORR on Pt/Co catalyst involves strongly correlated species which are good test cases for future demonstration of quantum advantage” and “Practical advantage over classical methods will require improved quantum computing hardware and application of quantum phase estimation techniques”, meaning the need for long-term FTQC QPUs [2756] **FTQC**.
- Optimizing aircraft flight paths, tested by Acubed and qBraid. It is a quantum-inspired algorithm that is not running on any quantum hardware. It is a classical algorithm **QI**.
- Aircraft loading optimization tested on IonQ Aria (25 qubits) and Forte (32 qubits) using a QAOA algorithm using 29 qubits, but with no quantum advantage found [2840].

On top of that, Airbus is a party in the EU funded project Equality with Pasqal, Inria, DLR, Da Vinci Labs, Fraunhofer ENAS, University of Leiden working on aerodynamics, fluids, materials, energy and mission planning. The project coordinated by Capgemini Germany got publicly funded at 100% with €6M. It started

in November 2022 and ended in 2025. Tests were to be run in Europe EuroHPC hybrid centers.

Rolls Royce partnered in 2022 with **Classiq** to develop quantum algorithms to model computation fluid dynamics for the development of its aircraft engines [2847]. They were also teaming up with **Riverlane** and **Xanadu** for the same need [2848] as well avec also with Riverlane, for materials design [2849].

One team from **Capgemini** devised a new hybrid matrix inversion algorithm mixing the famous HHL algorithm and a QSVM (Quantum-enhanced Support Vector Machine (QSVM)). Other teams worked on plane loading optimization problems, on quantum machine learning models and on fluids dynamics applied to aircraft design.

3.6.4 Sea

ExxonMobil and **IBM** are working on finding algorithms to optimize maritime traffic routing. Existing solutions rely on heuristics and simplifications. They were willing to see whether quantum computing could transform these complex optimization problems and solve them more efficiently with quantum computing. Their vision is about container shipments volumes. They formatted their problem as a “vehicle routing problem with time windows” (VRPTW) which is a NP-hard problem. They compared various methods, using a QUBO algorithm that can be transformed on a lower-level VQE or QAOA hybrid algorithm and experimented it with Qiskit on the QasmSimulator IBM quantum emulator backend [2850]. The paper does not provide any clear answers on the gain and on the quantum computer specification that would make it possible to solve a real-life problem. On top of that, it was not at all formulated as a container shipment optimization problem per se but as a simpler truck routing problem **NISQ**.

Maritime container shipping is also a center of interest. **DP World**, the Dubai Port operator, is partnering with **D-Wave** to find ways to use quantum computing to

optimize their port operations. At this stage, it is just exploratory work with no details at all about envisioned applications **analog**.

Likely applications include container loading and off-loading optimization [2851]. Another such project was also undertaken with the **Port of Los Angeles** by SavantX, also using D-Wave annealers and quantum machine learning [2852] **analog**.

Another paper is about optimizing maritime routing in the Arctic sea, accounting for the dynamic nature of sea ice conditions, tested with the D-Wave hybrid solver. It reports better results in a shorter time than classical optimizers, but the comparison is not matched: an approximate hybrid solution is set against exact Gurobi and CPLEX solutions, with no common objective, optimality tolerance, time budget or solution-quality requirement, so no performance advantage follows [2214] **analog**.

3.6.5 Supply chain

Optimizing various supply chains is an interesting field of interest to implement any sort of optimization algorithm. It is also a domain for active research, particularly to enable such algorithms on NISQ platforms [2853–2856]. There are several algorithm papers in that domain [2857–2859].

Sandia Labs (DoE) introduced FALQON in 2022, a Max-Cut QAOA variant performing ansatz optimizations without an expensive classical optimization loop [2860, 2861]. It converges to good approximation ratios and success probabilities with reasonable resources scaling. However, FALQON uses deeper circuits than classical QAOA which turns into a higher requirement regarding physical qubit fidelities.

A team from **ETSISI** in Spain developed use cases for logistics optimizations using D-Wave quantum annealers, focused on solving the vehicle routing problem [2862]. They compared their hybrid solution to classical solutions and found some potential quantum advantage when the randomness of the data is high **NISQ**.

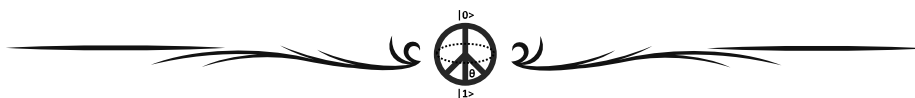
IAG Cargo (resulting from the merger of British Airways World Cargo and Iberia Cargo in April 2011) worked in 2022 with **Quantum-South** to prototype a solution to optimize cargo loading in passenger aircraft using D-Wave quantum annealers with aircraft load optimization and bin packing optimization for a single aircraft unit load device (ULD) [2863]. The solution was ultimately offered in a quantum-inspired format, running on classical computers **QI**.

In Japan, **AISIN** has been testing D-Wave quantum annealers to solve a vehicle routing problem using about 2500 quadratic binary variables [2864] and earlier, on a smallish 8-qubit Rigetti QPU [2865] **analog** **NISQ**.

In Bangladesh, a quantum machine learning based supply chain backorder prediction tool was prototyped in 2023, running on 4 qubits [2866]. It is not easy to understand how it would scale with 4 qubits **NISQ**.

In Italy, a Drone Delivery Packing Problem (DDPP) finding the optimal assignment of deliveries for a limited number of drones transported by a truck was tested on an analog Pasqal system in 2026 with up to 100 deliveries but scalability was difficult beyond 50 atoms [2867] **analog**.

Other logistics problems looking for optimal solutions due to their NP hardness are the Maximal Covering Location Problem (MCLP) that aims to determine the optimal facility placements that maximize total coverage [2868] and the emergency operation center placement in disaster response [2869].



3.6.6 Bibliography and notes

These are the bibliographical references and notes for the **Transportation and logistics** subsection.

- [2214] [Hybrid Quantum Annealing Approach for High-Dimensional and Multi-Criteria Constrained Quadratic Optimization in Arctic Ship Routing](#), by Tara Kit, Kim-say Pov, Myeongseong Go, Leanghok Hour, Arim Ryou, Kiwoong Kim, Tae-Kyung Kim, and Youngsun Han, arXiv, December 2025 (19 pages).
- [2365] [Traffic flow optimization using a quantum annealer](#), by Florian Neukart, Gabriele Compostella, Christian Seidel, David von Dollen, Sheir Yarkoni, and Bob Parney, arXiv, August 2017 (17 pages).
- [2756] [Platinum-based Catalysts for Oxygen Reduction Reaction simulated with a Quantum Computer](#), by Cono Di Paola, Evgeny Plekhanov, Michal Krompiec, Chandan Kumar, Emanuele Marsili, Fengmin Du, Daniel Weber, Jasper Simon Krauser, Elvira Shishenina, and David Muñoz Ramo, arXiv, April 2024 (29 pages).
- [2771] [Quantum Applications Transportation and Manufacturing](#), by Yianni Gamvros, 2017 (20 slides).
- [2772] [Quantum computing for transport research: an introduction, systematic review, and perspective](#), by Lachlan Oberg, Paul Corry, Moji Ghadimi, and Ashish Bhaskar, arXiv, March 2026 (52 pages).
- [2773] [Quantum Computing in Transport Science: A Review](#), by Chence Niu, Elnaz Irannezhad, Casey Myers, and Vinayak Dixit, arXiv, March 2025 (30 pages).
- [2774] [Socio-economic impacts of quantum technologies in transport](#), by UK Department of Transport, GOV, November 2024 (84 pages).
- [2775] [Classical and Quantum Heuristics for the Binary Paint Shop Problem](#), by V Vijendran, Syed M Assad, et al., arXiv, September 2025 (30 pages).
- [2776] [Solving workflow scheduling problems with QUBO modeling](#), by A. I. Pakhomchik, S. Yudin, M. R. Perelshtein, A. Alekseyenko, and S. Yarkoni, arXiv, May 2022 (10 pages).
- [2777] [Hybrid quantum ResNet for car classification and its hyperparameter optimization](#), by Asel Saginalieva, Mo Kordzan-

- ganeh, Andrii Kurkin, Artem Melnikov, Daniil Kuhmistrov, Michael Perelshtein, Alexey Melnikov, Andrea Skolik, and David Von Dollen, arXiv, September 2023 (16 pages).
- [2778] [Validation of Quantum Computing for Transition Metal Oxide-based Automotive Catalysis](#), by Yuntao Gu, Louis Hector Jr, Paolo Giusto, Matthew Titsworth, Alok Warey, Dnyanesh Rajpathak, and Eser Atesoglu, arXiv, December 2025 (6 pages).
- [2779] [Quantum Annealing Approach for the Optimal Real-time Traffic Control using QUBO](#), by Amit Singh, Fang-Pang Lin, et al., arXiv, March 2024 (8 pages).
- [2780] [Quantum Annealing for Realistic Traffic Flow Optimization: Clustering and Data-Driven QUBO](#), by Renáta Rusnáková, Martin Chovanec, and Juraj Gazda, arXiv, October 2025 (29 pages).
- [2781] [Hybrid quantum optimization in the context of minimizing traffic congestion](#), by Jedwin Villanueva, Lloyd C L Hollenberg, et al., arXiv, April 2025 (15 pages).
- [2782] [Data re-uploading in Quantum Machine Learning for time series: application to traffic forecasting](#), by Nikolaos Schetakis, Eleni I. Vlahogianni, et al., arXiv, January 2025 (20 pages).
- [2783] [Quantum computing for transport optimization](#), by Christopher D. B. Bentley, Samuel Marsh, André R. R. Carvalho, Philip Kilby, and Michael J. Biercuk, arXiv, June 2022 (12 pages).
- [2784] [Quantum Approaches to Urban Logistics: From Core QAOA to Clustered Scalability](#), by F. Picariello, A. Noto, et al., arXiv, December 2025 (24 pages).
- [2785] [Quantum computers: How they work, real-world applications, and future impact](#), by Amal Nozieres, March 2025.
- [2786] [Advanced Quantum Annealing Approach to Vehicle Routing Problems with Time Windows](#), by James B. Holliday, Darren Blount, Eneko Osaba, and Khoa Luu, arXiv, March 2025 (9 pages).
- [2787] [Volkswagen takes quantum computing from the lab to the factory](#), by Volkswagen, August 2021.
- [2788] [Recommending Solution Paths for Solving Optimization Problems with Quantum Computing](#), by Benedikt Poggel, Nils Quetschlich, Lukas Burgholzer, Robert Wille, and Jeanette Miriam Lorenz, arXiv, October 2023 (8 pages).
- [2789] [An Advanced Hybrid Quantum Tabu Search Approach to Vehicle Routing Problems](#), by James B. Holliday, Eneko Osaba, and Khoa Luu, arXiv, January 2025 (27 pages).
- [2790] [Solving Large-Scale Vehicle Routing Problems with Hybrid Quantum-Classical Decomposition](#), by Andrew Maciejunes, John Stenger, Dan Gunlycke, and Nikos Chrisochoides, arXiv, July 2025 (9 pages).
- [2791] [Quantum Computing in Intelligent Transportation Systems: A Survey](#), by Yifan Zhuang, Talha Azfar, Yinhai Wang, Wei Sun, Xiaokun Cara Wang, Qianwen Vivian Guo, and Ruimin Ke, arXiv, June 2024 (8 pages).
- [2792] [A heuristic for solving the irregular strip packing problem with quantum optimization](#), by Paul-Amaury Matt and Marco Roth, arXiv, February 2024 (30 pages).
- [2793] [Testing Quantum and Simulated Annealers on the Drone Delivery Packing Problem](#), by Sara Tarquini, Daniele Dragoni, Matteo Vandelli, and Francesco Tudisco, arXiv, June 2024 (16 pages).
- [2794] [Requirements for Early Quantum Advantage and Quantum Utility in the Capacitated Vehicle Routing Problem](#), by Chinonso Onah and Kristel Michielsen, arXiv, September 2025 (13 pages).
- [2795] [A supervised hybrid quantum machine learning solution to the emergency escape routing problem](#), by Nathan Haboury, Mo Kordzanganeh, Sebastian Schmitt, Ayush Joshi, Igor Tokarev, Lukas Abdallah, Andrii Kurkin, Basil Kyriacou, and Alexey Melnikov, arXiv, July 2023 (15 pages).
- [2796] [A Hybrid Quantum-assisted Column Generation Algorithm for the Fleet Conversion Problem](#), by Yagnik Chatterjee, Zaid Allybokus, Marko J. Rančić, and Eric Bourreau, arXiv, March 2024 (20 pages).
- [2797] [Joint Optimization of Electric Vehicle Routes and Charging Locations Learning Charge Constraints Using QUBO Solver](#), by Akihisa Okada, Keisuke Otaki, and Hiroaki Yoshida, arXiv, June 2025 (12 pages).
- [2798] [Toyota Central R&D Labs: A Quantum Approach to Transportation](#), by D-Wave et al., 2021 (2 pages).
- [2799] [Traffic Signal Optimization on a Square Lattice with Quantum Annealing](#), by Daisuke Inoue, Akihisa Okada, Tadayoshi Matsumori, Kazuyuki Aihara, and Hiroaki Yoshida, arXiv, February 2021 (14 pages).
- [2800] [Qubit efficient quantum algorithms for the vehicle routing problem on NISQ processors](#), by Ioannis D. Leonidas, Alexander Dukakis, Benjamin Tan, and Dimitris G. Angelakis, arXiv, September 2023 (9 pages).
- [2801] [QUEST: QUantum-Enhanced Shared Transportation](#), by Chinonso Onah, Neel Miscasci, Carsten Othmer, and Kristel Michielsen, arXiv, May 2025 (11 pages).
- [2802] [Quantum and classical approaches to the optimization of highway platooning: the two-vehicle matching problem](#), by Chinonso Onah, Agneev Guin, Carsten Othmer, J. A. Montañez-Barrera, and Kristel Michielsen, arXiv, March 2026 (30 pages).
- [2803] [Quantum Optimization Methods for the Generalized Traveling Salesman Problem](#), by Maximilian Zorn, Melinda Braun, Michael Ertl, Tommy Kiss, Sara Juarez Oropeza, Claudia Linnhoff-Popien, and Jonas Stein, arXiv, April 2026 (11 pages).
- [2804] [A Hybrid Classical-Quantum Annealing Algorithm for the TSP](#), by Siwei Hu, Victor Lopata, Salvatore Sinno, Shruthi Thuravakkath, and Paolo Zuliani, arXiv, May 2026 (9 pages).
- [2805] [Cutting-plane methodology via quantum optimization for solving the Traveling Salesman Problem](#), by Alessia Ciacco, Luigi Di Puglia Pugliese, and Francesca Guerriero, arXiv, April 2026 (25 pages).
- [2806] [A Quantitative Framework for Comparing Classical and Quantum Algorithms for the Traveling Salesman Problem](#), by Krit Grover and Marcelo Ponce, arXiv, July 2026 (8 pages).
- [2807] [A Quantum Encoding of Traveling Salesperson Tours via Route Generation, Cost Phases, and a Reversible Valid-Permutation Oracle](#), by Alexander Johannes Stasik and Franz Georg Fuchs, arXiv, June 2026 (13 pages).
- [2808] [An edge-based and subspace reduction encoding scheme to solve the traveling salesman problem in quantum computers](#), by Anandu Kalleri Madhu, Chi-Kwong Li, Jami Rönkkö, Mikio Nakahara, and Ray-Kuang Lee, arXiv, December 2025 (14 pages).
- [2809] [Quantum iterative approach to the Traveling Salesman Problem](#), by Arturo Rodríguez-Almazán, Guillermo Rivas, Ricardo S. Alonso, Daniela Falcó, and Mir Amir Hosseini, arXiv, June 2026 (12 pages).
- [2810] [Quantum Divide-and-Conquer for the Traveling Salesman Problem: Surpassing the \$2^n\$ Barrier](#), by Xujun Bai, Yun Shang, and Honghong Lin, arXiv, July 2026 (30 pages).
- [2811] [Improving Feasibility in Quantum Approximate Optimization Algorithm for Vehicle Routing via Constraint-Aware Initialization and Hybrid XY-X Mixing](#), by Yuan-Zheng Lei, Yaobang Gong, Xianfeng Terry Yang, and Nii Attoh-Okine, arXiv, April 2026 (36 pages).
- [2812] [Analyzing performance of commercial quantum annealing solvers for the capacitated vehicle routing problem](#), by Salvatore Sinno, Bhavika Bhalgamiya, et al., AIP Advances, December 2025 (10 pages).
- [2813] [Hierarchical QAOA for the Vehicle Routing Problem via Clustered Decomposition and Local Feasibility Repair](#), by Shreetam Dash, Shreya Banerjee, and Prasanta K. Panigrahi, arXiv, May 2026 (12 pages).

- [2814] [Advanced Quantum Annealing for the Bi-Objective Traveling Thief Problem: An \$\epsilon\$ -Constraint-based Approach](#), by Nguyen Hoang Viet, Nguyen Xuan Tung, Trinh Van Chien, and Won-Joo Hwang, arXiv, March 2026 (14 pages).
- [2815] [A Spatio-Temporal Hybrid Quantum-Classical Graph Convolutional Neural Network Approach for Urban Taxi Destination Prediction](#), by Xiuying Zhang, Qinsheng Zhu, and Xiaodong Xing, arXiv, December 2025 (11 pages).
- [2816] [Quantum Amplitude Estimation for Travel Time Estimation in Stochastic Vehicle Routing Problems](#), by Xingyue Wang and Monika Filipovska, arXiv, August 2026 (11 pages).
- [2817] [Multi-Agent Route Planning as a QUBO Problem](#), by Renáta Rusnáková, Martin Chovanec, and Juraj Gazda, arXiv, June 2026 (8 pages).
- [2818] [Hybrid quantum-classical computation for automatic guided vehicles scheduling](#), by Tomasz Śmierzchalski, Sebastian Deffner, Krzysztof Domino, et al., Scientific Reports, September 2024.
- [2819] [Nav-Q: Quantum Deep Reinforcement Learning for Collision-Free Navigation of Self-Driving Cars](#), by Akash Sinha, Antonio Macaluso, and Matthias Klusch, arXiv, December 2023 (28 pages).
- [2820] [Quantum Artificial Intelligence for Secure Autonomous Vehicle Navigation: An Architectural Proposal](#), by Hemanth Kannamarlapudi and Sowmya Chintalapudi, arXiv, June 2025 (5 pages).
- [2821] [QDCNN: Quantum Deep Learning for Enhancing Safety and Reliability in Autonomous Transportation Systems](#), by Ashtakala Meghanath, Ahmed Farouk, et al., arXiv, March 2025 (11 pages).
- [2822] [Learn Quantum: Machine Learning Image Recognition Application](#), by IonQ, September 2023.
- [2823] [Quantum Computing for Automotive Applications](#), by Carlos A. Riofrío, Johannes Klepsch, Jernej Rudi Finžgar, Florian Kiwit, Leonhard Hölscher, Marvin Erdmann, Lukas Müller, Chandan Kumar, Youssef Achari Berrada, and Andre Luckow, arXiv, March 2025 (25 pages).
- [2824] [How BMW Can Maximize Its Supply Chain Efficiency with Quantum](#), by Honeywell, January 2021.
- [2825] [Quantum-Informed Recursive Optimization Algorithms](#), by Jernej Rudi Finžgar, Aron Kerschbaumer, Martin J. A. Schuetz, Christian B. Mendl, and Helmut G. Katzgraber, arXiv, March 2024 (17 pages).
- [2826] [Real World Application of Quantum-Classical Optimization for Production Scheduling](#), by Abhishek Awasthi, Nico Kraus, Florian Krellner, and David Zambrano, arXiv, August 2024 (6 pages).
- [2827] [Quantum AI For The Auto Industry](#), by DFKI, June 2021.
- [2828] [Solving rescheduling problems in heterogeneous urban railway networks using hybrid quantum-classical approach](#), by Máttyás Koniorczyk, Krzysztof Krawiec, Ludmila Botelho, Nikola Bešinović, and Krzysztof Domino, arXiv, March 2025 (33 pages).
- [2829] [On the Baltimore Light RailLink into the quantum future](#), by Krzysztof Domino, Emery Doucet, Reece Robertson, Bartłomiej Gardas, and Sebastian Deffner, arXiv, July 2025 (22 pages).
- [2830] [Accelerating the schedule for quantum-enhanced rail | Q-CTRL](#), by Q-CTRL, Q-CTRL news, May 2025.
- [2831] [Airline Fleet Assignment Problems with Binary and Integer Programming models: Classical vs Quantum Annealing](#), by Kuntal Adak, Sakshi Kaushik, and Rahul Rana, arXiv, February 2026 (7 pages).
- [2832] [Flight Gate Assignment with a Quantum Annealer](#), by Elisabeth Lobe et al., March 2019 (15 slides).
- [2833] [Gaussian Boson Sampling for binary optimization](#), by Jean Cazalis, Yahui Chai, Karl Jansen, Stefan Kühn, Tirth Shah Q.ANT GmbH, CQTA, Deutsches Elektronen-Synchrotron DESY, and Cyprus Institute), arXiv, December 2023 (3 pages).
- [2834] [Towards Finding an Optimal Flight Gate Assignment on a Digital Quantum Computer](#), by Yahui Chai, Lena Funcke, Tobias Hartung, Karl Jansen, Stefan Kuehn, Paolo Stornati, and Tobias Stollenwerk, arXiv, February 2023 (15 pages).
- [2835] [Exploring Airline Gate-Scheduling Optimization Using Quantum Computers](#), by Hamed Mohammadbagherpoor, Patrick Dreher, Mohannad Ibrahim, Young-Hyun Oh, James Hall, Richard E Stone, and Mirela Stojkovic, arXiv, November 2021 (20 pages).
- [2836] [Simulating the flight gate assignment problem on a trapped ion quantum computer](#), by Yahui Chai, Evgeny Epifanovsky, Karl Jansen, Ananth Kaushik, and Stefan Kühn, arXiv, September 2023 (7 pages).
- [2837] [University of Hamburg, Lufthansa Say Quantum Algorithms for Airport Operations Ready For Takeoff](#), by Matt Swayne, The Quantum Insider, July 2024.
- [2838] [Optimization of Flight Routes: Quantum Approximate Optimization Algorithm for the Tail Assignment Problem](#), by Marta Gili, Paul San Sebastian, and Ane Blázquez-García, arXiv, December 2024 (9 pages).
- [2839] [Variational Quantum Optimization Benchmark Suite for Airline Crew Pairing and More](#), by Dávid Sipos, András Czégel, and Boglárka G.-Tóth, arXiv, April 2025 (13 pages).
- [2840] [Quantum Computing for Optimizing Aircraft Loading](#), by Ananth Kaushik, Richard Ashworth, et al., arXiv, April 2025 (12 pages).
- [2841] [Design and Simulation of an Autonomous Quantum Flying Robot Vehicle: An IBM Quantum Experience](#), by Sudev Pradhan, Anshuman Padhi, and Bikash Kumar Behera, arXiv, June 2022 (7 pages).
- [2842] [Routing and Scheduling Optimization for Urban Air Mobility Fleet Management using Quantum Annealing](#), by Renichiro Haba, Masayuki Ohzeki, et al., arXiv, October 2024 (13 pages).
- [2843] [Solving Drone Routing Problems with Quantum Computing: A Hybrid Approach Combining Quantum Annealing and Gate-Based Paradigms](#), by Eneko Osaba, Michail Alexandros Kourtis, et al., arXiv, March 2025 (8 pages).
- [2844] [Quantum Reservoir Computing for Corrosion Prediction in Aerospace: A Hybrid Approach for Enhanced Material Degradation Forecasting](#), by Akshat Tandon, Jasper Simon Krauser, et al., arXiv, May 2025 (6 pages).
- [2845] [Bringing Aerospace into the Quantum Era](#), by Jasper Krauser, December 2022.
- [2846] [Airbus gets aerodynamic with quantum computing](#), by Michael Feldman, January 2019.
- [2847] [Rolls-Royce Adopts Quantum Hybrid Model for Simulations](#), by Berenice Baker, October 2022.
- [2848] [Rolls-Royce, Riverlane and Xanadu partner to win Canada-UK](#), by Riverlane, March 2024.
- [2849] [NQCC, Rolls-Royce and Riverlane partner to accelerate materials discovery](#), by Riverlane, December 2023.
- [2850] [Formulating and Solving Routing Problems on Quantum Computers](#), by Stuart Harwood, January 2021 (17 pages).
- [2851] [DP World explores quantum computing technology](#), by Arab News, Arab News, April 2021.
- [2852] [SavantX, D-Wave Collaborate on Quantum Algorithms for Busy Port](#), by Matt Swayne, The Quantum Insider, January 2022.
- [2853] [Quantum Computing in Logistics and Supply Chain Management an Overview](#), by Frank Phillipson, arXiv, April 2025 (52 pages).
- [2854] [Navigating the quantum revolution in logistics](#), by Sean Tinney, November 2023.
- [2855] [Parameter-Efficient Quantum-Inspired Fast Weight Programmers for Traffic-Matrix Forecasting](#), by Kuo-Chung Peng, Jiun-Cheng Jiang, Chun-Hua Lin, Tai-Yue Li, Nan-

- Yow Chen, and Samuel Yen-Chi Chen, arXiv, June 2026 (6 pages).
- [2856] [Hybrid Quantum-Classical Optimization for Multi-Objective Supply Chain Logistics](#), by Raoul Heese, Timothée Leleu, Sam Reifenstein, Christian Nietner, and Yoshitsa Yamamoto, arXiv, February 2026 (27 pages).
 - [2857] [Benchmarking of quantum and classical SDP relaxations for QUBO formulations of real-world logistics problems](#), by Birte Ostermann, Timo de Wolff, et al., arXiv, March 2025 (32 pages).
 - [2858] [Quantum Annealing Approaches to Solving the Shipment Rerouting Problems](#), by Fei Li, Arul Rhik Mazumder, and Max Zhao, arXiv, January 2025 (18 pages).
 - [2859] [Optimization Algorithm for Inventory Management on Classical, Quantum and Quantum-Hybrid Hardware](#), by Gabriel P. L. M. Fernandes, Celso J. Villas-Boas, et al., arXiv, November 2024 (12 pages).
 - [2860] [Feedback-based quantum optimization](#), by Alicia B. Magann, Kenneth M. Rudinger, Matthew D. Grace, and Mohan Sarovar, arXiv, January 2023 (7 pages).
 - [2861] [Lyapunov control-inspired strategies for quantum combinatorial optimization](#), by Alicia B. Magann, Kenneth M. Rudinger, Matthew D. Grace, and Mohan Sarovar, arXiv, January 2023 (19 pages).
 - [2862] [Adiabatic Quantum Computing for Logistic Transport Optimization](#), by Juan Francisco Ariño Sales and Raúl Andres Palacios Araos, arXiv, January 2023 (43 pages).
 - [2863] [Quantum-South Explores Quantum Algorithms for Air Cargo Optimization](#), by James Dargan, The Quantum Insider, December 2022.
 - [2864] [Supply chain logistics with quantum and classical annealing algorithms](#), by Sean J. Weinberg, Randall Correll, et al., Scientific Reports, March 2023 (18 pages).
 - [2865] [Short Quantum Circuits in Reinforcement Learning Policies for the Vehicle Routing Problem](#), by Fabio Sanches, Sean Weinberg, Takanori Ide, and Kazumitsu Kamiya, arXiv, September 2021 (15 pages).
 - [2866] [QAmplifyNet: Pushing the Boundaries of Supply Chain Backorder Prediction Using Interpretable Hybrid Quantum-Classical Neural Network](#), by Md Abrar Jahin, Md Sakib Hossain Shovon, Md. Saiful Islam, Jungpil Shin, M. F. Mridha, and Yuichi Okuyama, arXiv, October 2023 (28 pages).
 - [2867] [Drone delivery packing problem on a neutral-atom quantum computer](#), by Sara Tarquini, Matteo Vandelli, Francesco Ferrari, Daniele Dragoni, and Francesco Tudisco, arXiv, February 2026 (10 pages).
 - [2868] [A Hybrid Classical-Quantum Approach for Multi-Constrained Location Optimization Problem](#), by Jorge Saavedra-Benavides, J. Alejandro Montanez-Barrera, Alberto Maldonado-Romo, and Daniel Sierra-Sosa, arXiv, July 2026 (14 pages).
 - [2869] [Emergency hub placement with a neutral-atom quantum computer](#), by Sara Tarquini, Matteo Vandelli, Francesco Ferrari, Daniele Dragoni, and Francesco Tudisco, arXiv, June 2026 (15 pages).

This **Transportation and logistics** subsection contains 102 bibliographical references and notes containing at least 1,458 pages.

3.7 Financial services

Financial services are another important field for experimenting with quantum technologies, particularly quantum computing [2870–2874]. The sector has substantial forecasting and optimization needs, and many large financial institutions have the scale and resources to invest in emerging technologies.

Financial applications are usually segmented into three categories: simulation, optimization, and machine-learning tasks, with existing classical solutions and proposed quantum counterparts [2875, 2876] (Figure 188 and Figure 189).

Banks have a pressing need to adapt to technological and societal changes. They manipulate large, valuable datasets and optimize many facets of their business, including portfolio returns, risk, regulatory constraints, and fraud detection. Basel III includes capital and liquidity requirements, notably the liquidity coverage ratio (LCR), rather than a “liquid coverage ratio”. We also include here the insurance sector, which is often associated with financial services and banks. There are also interesting mathematical relationships between some key equations in finance and quantum physics. A review paper from Isaiah Hull, Or Sattath, Eleni Diamanti and Göran Wendin describes the wealth of quantum algorithms that could be used in the economic and financial spheres [2877].

Many existing quantum-computing proof-of-concept projects have been tested on relatively small problem instances, often with tens to hundreds of assets, while production workflows may involve much larger universes and many more scenarios or constraints. Analog and gate-based pilots are increasing in scale. However, executing more than 100 physical qubits or a few thousand gates is not by itself evidence that quantum advantage is close. Such a claim requires an end-to-end comparison with strong classical baselines at comparable accuracy, including data loading, measurement, and classical post-processing costs [2878].

3.7.1 Analysts roadmaps

A **BCG** 2020 review on quantum-computing financial use cases proposed a roadmap against the expected progress of quantum hardware over the following 5 to 20 years, as shown in Figure 191 [2879]. The roadmap placed quantum-inspired and analog approaches in the short term, gate-based approaches later, and fault-tolerant quantum computing (FTQC) in the long term. Today, circuits with more than 100 qubits and several thousand gates can be executed, but circuit size alone does not establish application maturity. FTQC resource studies often reach thousands or more logical qubits and millions to billions or more logical operations, depending strongly on the algorithm and accuracy target.

McKinsey is more upbeat and is pushing banks to evaluate as fast as possible the potential use cases of quantum computing [2880].

3.7.2 Bank approaches

Large financial institutions have been studying quantum computing for many years.

JPMorganChase has one of the most visible and prolific in-house quantum research programs in financial services. Marco Pistoia, who previously led its quantum-computing activities, joined IonQ in 2025. The bank investigates quantum algorithms, optimization, fundamental mathematical methods [1799], cybersecurity, quantum communications, and related technologies. Its researchers have also published work on non-convex optimization [2881].

D-Wave has worked with several banks, including **Deutsche Bank**, and created the [Quantumforquants](#) website, dedicated to the uses of quantum in finance. In 2022, they announced an R&D partnership with **Mastercard** to create quantum-hybrid applications for optimizing consumer loyalty and rewards programs, cross-border settlement, and fraud management.

analog **Bank of Canada** and **Multiverse Computing** created in 2022 a prototype quantum simulation of the cryptocurrency market using a D-Wave quantum annealer. It simulated a financial network with 8-10 players [2882].

NISQ With **Xanadu**, the bank also worked on the use of quantum Monte Carlo methods for various use cases, determining that hardware progress was needed to obtain some quantum advantage [2883].

Since 2017, IBM has been highlighting partnerships with **JPMorganChase** [2884, 2885], **Barclays** [2886–2889], **HSBC** [2890–2894] (which also works with Quantinuum) and **Wells Fargo** [2895] to study the uses of quantum computing in trading strategy optimization, investment portfolio optimization, pricing and risk analysis.

Goldman Sachs hired Will Zeng from Rigetti Computing, where he co-developed the Quil quantum instruction language. Zeng founded the Unitary Fund, which promotes open-source quantum software, tools and benchmarks, and became a partner at Quantontation in 2023. Goldman Sachs researchers have worked with IBM on derivative-pricing algorithms [2896]. These studies are not operational production applications, but they help quantify future hardware requirements. Goldman Sachs also worked with Quantum Motion (UK) on lower-level algorithmic improvements for options pricing [108].

Royal Bank of Scotland (RBS) invested in 1Qbit along with Fujitsu and Allianz.

NISQ **Itaú Unibanco** (Brazil) and **QC Ware** (USA) undertook a four-month joint project on customer re-

problem category	use cases	classical solutions	quantum solutions (NISQ)	quantum solutions (FTQC)
simulation	- derivative pricing. - risk analysis (Basel, Solvency). - financial econometrics. - maximum likelihood estimation. - dynamic stochastic general equilibrium modelling (DSGE). - dynamic economic models.	- Monte Carlo integration. - machine learning. - Black-Scholes model.	quantum machine learning (variational methods).	- quantum amplitude estimation in quantum Monte Carlo methods. - quantum machine learning (HHL-based methods).
optimization	- portfolio optimization. - trading optimization. - hedging. - optimal arbitrage. - credit scoring. - financial crash prediction.	- discrete/continuous variables. - branch-and-bound for non-convex cases. - interior-point methods for certain convex cases.	- quantum optimization (QAOA or QUBO based) on analog QPUs. - reverse quantum annealing. - VQE methods.	- quantum walks. - Grover and Simon oracle-based searches.
machine learning	- anomaly and fraud detection. - natural language modeling. - risk clustering. - modeling credit spread. - product recommendation.	- regression, classification, clustering, PCA. - deep learning. - unsupervised cluster analysis.	- quantum machine learning (SVM, PCA, ...), variational. - quantum deep learning, variational. - quantum cluster analysis (analog).	- quantum SVM, PCA. - quantum deep learning (QCNN, QGAN, QGNN, ...)

Figure 188: Source: adapted from “A Survey of Quantum Computing for Finance” by Dylan Herman et al., JP Morgan, Universities of Chicago, Delaware, DoE Argonne National Lab and Menten AI, January 2022 (56 pages). Added in 2023. Updated in September 2024.

Illustration : 14 use cases in financial institutions

	Use cases	Unlock capability / value	Typical industry	FI publicly experimenting
Optimization	1 Portfolio selection, alloc. & optimization	Increase both the scope of assets that can be taken into account and dynamic multi-period scenarios that can be considered	Asset Management, Global Markets	Natwest, BBVA, Commerzbank, Standard Chartered, CBA, Nomura
	2 Optimal execution	Design the best execution strategy for entry, exit and rebalancing	Asset Management, Global Markets	-
	3 Capital allocation	Allow dynamic capital allocation without making over-simplifications (credit risk and insurance risk in particular)	All Financial Institutions	-
	4 Asset Liability Management	Increase the number or detail of assets and run much more detailed scenarios	Universal Banks, Asset Managers	-
	5 Transaction settlement	Ensure that large volume of trades is settled in the most optimized sequence and prioritization.	Global Markets, Transaction Banking, Clearing House	Barclays, Mastercard
	6 Yield curve fitting	Can solve much complex models to improve yield curve fitting accuracy	Global Markets, HFT	-
Machine Learning	7 Credit scoring / clustering	Build more realistic models thanks to the ability to take into account more variables and speed-up the training process	Retail banking	CaixaBank, Crédit Agricole
	8 Default early warnings	Improve detection of changing customer behaviours indicative of financial stress leveraging much more complex datasets	Retail banking	-
	9 Fraud detection / AML	Identify outliers based on a growing number of variables, which will lead to better adjusted models	Retail & Transaction banking, Global Markets	-
	10 Next Best Action / Product	Rely on more consistent clusters defined with a wider range of variables to improve outputs of predictive analytics	Retail and Private Banking, Insurance	-
Simulation	11 Derivative pricing	Improve dramatically the accuracy and efficiency of complex option pricing such as path-dependent and barrier options	Global Markets	JP Morgan, Goldman Sachs, BMO, Scotiabank
	12 Valuation and regulatory ratios	Perform computation on much wider and complex scenarios (VaR, XVA, valuation of credit derivatives, Solvency 2)	Banking, Insurance	HSBC
	13 Risk assessment & tail risk simulations	Compute a wide range of intraday complex stress tests, potentially even on a real time basis	Banking, Insurance	CaixaBank
	14 Multi-factor Interest rate models	Allow to take into account more realistic assumptions	Global Markets	-

Figure 189: Source: [Quantum Market & Industry Applications](#) by Jean-François Bobier, BCG, May 2023 (10 slides).

tention using QML methods for churn prediction. The main outcome was a new method that can also run on classical computers. The reported results improved predictions by 2% and model precision by 6.4% from a 71% baseline, using 180,000 data points [2897].

You can add to this quick review the quantum-software startups entering this market. They use quantum-inspired algorithms or run pilot projects with gate-based or annealing-based quantum computing. Among these are **1Qbit**, **Multiverse Computing**, **ApexQubit**, and **JoS QUANTUM**.

TABLE 5: Algorithms can improve computational efficiency, accuracy, and addressability for defined use case.

Quantum Algorithm	Description	Impact	Needs	Simulation	Optimization	ML
VQE	Variational Quantum Eigensolver	Use energy states to calculate the function of the variables to optimize	Procedure to assign compute-intensive functions to quantum and those of controls to classical		x	
QAOA	Quantum Approximate Optimization	Optimize combinatorial style problems to find solutions with complex constraints	Simplify analysis clauses for constraints and provide robust optimization in complex scenarios		x	
AE	Quantum Amplitude Estimator	Create simulation scenarios by estimating an unknown property, Monte Carlo style	Handle random distributions directly, instead of only sampling, to solve dynamic problems quadratically speeding up simulations	x	x	x
QSVM	Quantum Support Vector Machines	Supervised machine learning for high dimensional problem sets	Map data to quantum-enhanced feature space to enable separation and better separate data points to achieve more accuracy			x
HHL	Harrow, Hassidim, and Lloyd	Estimate the resulting measurement of large linear systems	Solve high dimensional problems speeding up exponentially calculations	x		x
QSDP	Quantum Semidefinite Programming	Optimize a linear objective over a set of positive semidefinite matrices	Estimate quantum system states with less measurements to exponentially speedup in terms of dimension and constraints		x	

TABLE 6: Financial services focus areas and algorithms.

Financial Services	Example Problems	Solution Approach	Quantum Algorithm
Asset Management	Option Pricing Portfolio risk	Simulation	AE
Investment	Portfolio Optimization	Optimization	Combinatorial: VQE, QAOA Continuous: QSDP
Banking	Portfolio Diversification Issuance: Auctions		AE
Retail & Corporate Banking	Financial Forecasting Credit Scoring (e.g. SME Banking) Financial Crimes: Fraud + AML	Machine Learning	QSVM HHL AE

Figure 190: Various algorithms for financial services. Source: [2875].

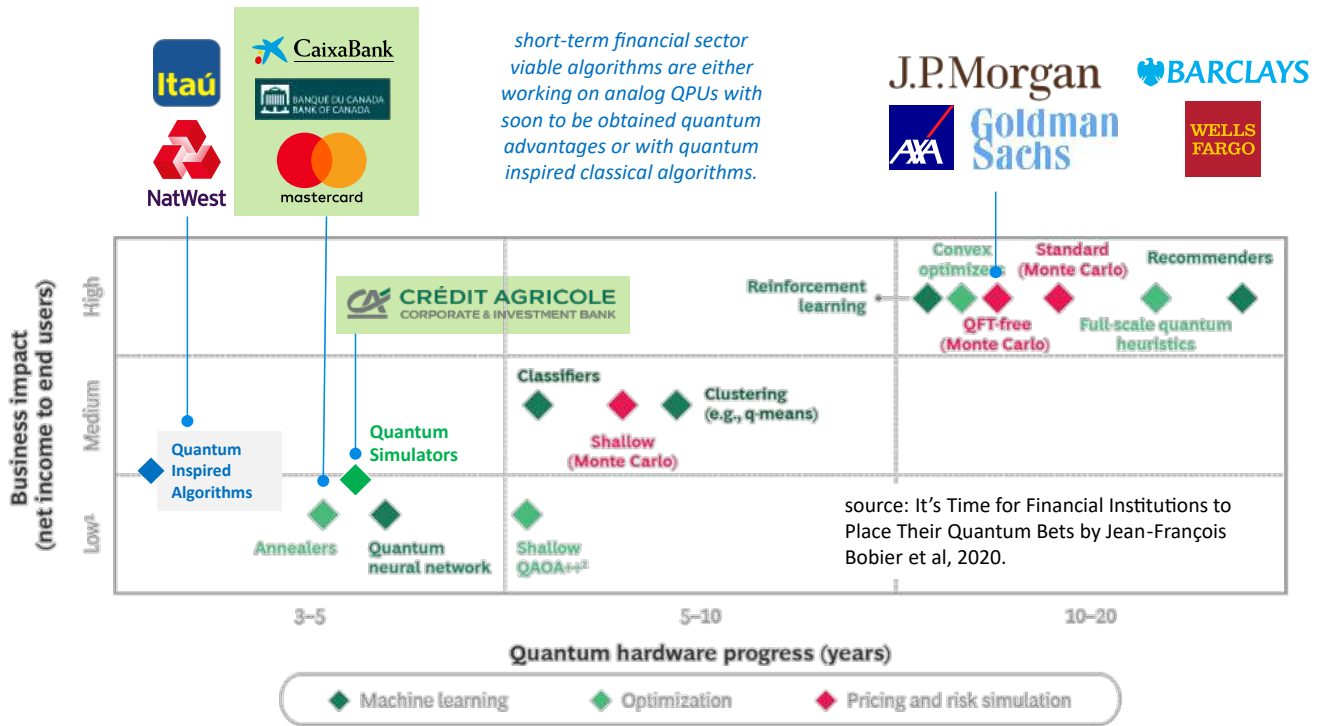


Figure 191: Chart showing the 2020 BCG roadmap for financial applications. It placed quantum-inspired and analog approaches in the short term and most fault-tolerant gate-based algorithms in the longer term. The timing should be interpreted as a 2020 scenario, not as a current prediction of imminent quantum advantage. Added in 2023. Source: [2879] and logos placed by Olivier Ezratty, 2022.

3.7.3 Portfolio optimization

Many prospective case studies and algorithmic research deal with optimizing investment portfolios and minimizing their risks. Financial institutions solve many

portfolio-optimization tasks, including maximizing expected return, controlling risk, and estimating value-at-risk (VaR) and Conditional Value at Risk (CVaR) [2878, 2898–2901]. VaR at confidence level α is a quantile of the loss distribution, not a “maximum expected

Method	(d, T)		Error		T-count		T-depth		# Logical Qubits	
	Auto	TARF	Auto	TARF	Auto	TARF	Auto	TARF	Auto	TARF
Riemann Sum					$\geq 10^{43}$	$\geq 10^{18}$	$\geq 10^{43}$	$\geq 10^{18}$	-	-
Riemann Sum (no-norm)	(3, 20)	(1, 26)	2×10^{-3}		1.6×10^{11}	5.5×10^{10}	1.5×10^8	1.6×10^8	23k	17k
Re-parameterization					1.2×10^{10}	9.8×10^9	5.4×10^7	8.2×10^7	8k	11.5k

Figure 192: For benchmark autocallable and TARF derivative pricing, the re-parameterization method in the source requires about 8,000–11,500 logical qubits, a T-count of order 10^{10} , and T-depth of about 5.4×10^7 – 8.2×10^7 . This figure concerns derivative pricing, not generic credit-risk analysis. Source: [1793].

loss”. For a continuous loss distribution, CVaR, also called expected shortfall, is the mean loss in the tail beyond the VaR threshold. More general definitions handle discrete loss distributions [2902]. Assets are also interdependent, and transaction costs depend on asset type, market conditions, and trading constraints.

You can assess proof-of-concept viability by reading the fine print on the number of portfolio assets, time periods, constraints, and classical baselines. Hardware demonstrations in the cited literature are generally far below the dimensionality and complexity of production-scale portfolio workflows.

NISQ NISQ portfolio-optimization proofs of concept documented in 2024 and 2025 use VQE [2903, 2904], QAOA [2905–2907], and quantum circuit-cutting techniques [2908]. Portfolio models can include higher-order moments such as skewness, which measures asymmetry, and kurtosis, a fourth-moment measure associated with tail heaviness and outliers. Such models can be formulated as HUBO problems (higher-order unconstrained binary optimization) and attacked with QAOA-type variational circuits. A 2025 experiment in Finland went up to 15 qubits and found that “*Unfortunately, solving higher-order portfolio optimization problems with QAOA proved challenging, and only a small subset of problems matched the quality of the solutions obtained with exact methods or classical algorithms*” and “*We suspect that the QAOA’s performance may be limited by the complex optimization landscape of portfolio problems, which often have many local minima. The current results demonstrate that the optimization becomes more challenging as the number of qubits increases*” [2909]. Eric Stopfer and Friedrich Wagner from Fraunhofer IIC reached a similarly negative conclusion in 2025, reporting that mixed-integer programming solved their portfolio instances in seconds and that problem-tailored classical heuristics outperformed the tested quantum annealing and QAOA approaches in solution quality for fixed run-time [2910].

analog The first portfolio-optimization proofs of concept used quantum annealers [2911–2914]. For example, **Raiffeisen Bank International** with Data Reply (Italy) tested a QUBO portfolio optimization on D-Wave Advantage annealers. The dataset came from a real-world production problem. They used two D-Wave hybrid solvers combining quantum annealing and classical methods, together with a purely classical algorithm. The portfolio was structured into three main asset classes: equity (EQ), fixed income (FI), and money

market (MM). A client portfolio typically ranged from 9 to 11 assets [2915]. In another trial, **Data Reply** worked on the Sharpe ratio, which measures excess expected return per unit of return volatility. The ratio is $(R_p - R_f)/\sigma_p$, where R_p is the expected portfolio return, R_f is the risk-free rate, and σ_p is the standard deviation of portfolio returns. For otherwise comparable portfolios, a higher Sharpe ratio indicates a better risk-adjusted return [2916].

Then came NISQ gate-based proofs of concept [2917–2919].

NISQ **JPMorganChase** tested in 2022 a mean-variance portfolio-optimization algorithm with 32 trapped-ion qubits, using XY mixing gates for the QAOA mixer alternating with phase operators. It was further optimized in 2024 [2920, 2921]. Systems of this scale can also be emulated classically on server clusters [2922].

NISQ JPMorganChase researchers also used in 2021 a NISQ-HHL approach, a hybrid adaptation of HHL for small-scale portfolio-optimization problems using 6 and 14 assets from the S&P 500. It used mid-circuit measurement, quantum conditional logic, and qubit reset and reuse in the Quantum Phase Estimation routine for eigenvalue estimation. The technique can reduce the number of ancillary qubits to one and lower connectivity and SWAP-gate requirements. The test was implemented on a Quantinuum System Model H1 with 11 qubits. This is a small, classically tractable proof of concept and does not establish quantum advantage [2923].

NISQ analog **TCS Research India** studied in 2023 how to combine quantum annealing and gate-based quantum computing for larger optimization problems. They improved the Large System Sampling Approximation (LSSA), which divides a large problem into smaller subproblems and combines their partial solutions to approximate the original solution. Tests used real-world Indian stock-market data with up to 64 assets and performed as well as the classical optimization methods tested [2924].

NISQ **CCB Fintech** (China) tested in 2024 another QAOA-based portfolio optimization algorithm with a quantum emulator using 18 qubits, that could fit on a simple laptop [2925].

NISQ A Fraunhofer 2025 study benchmarks quantum annealing and NISQ QAOA against classical methods (MIP, Simulated Annealing, Tabu Search) on 250 re-

alistic portfolio optimization instances. Classical MIP optimally solved all cases in seconds, consistently outperforming quantum heuristics and indicating very limited near-term quantum advantage [2910].

NISQ Kipu Quantum and IonQ researchers proposed in 2026 a hybrid pipeline for cardinality-constrained portfolio optimization on a 250-asset S&P 500 universe. The method decomposes the problem into qubit-compatible subproblems and solves them with bias-field digitized counterdiabatic quantum optimization (BF-DCQO). Hardware experiments included a 64-qubit barium trapped-ion development system. Larger executable subproblems reduced decomposition error and improved final solution quality relative to randomized baselines under identical post-processing. The paper does not demonstrate an end-to-end quantum advantage over state-of-the-art classical portfolio solvers [2926].

NISQ A SquareOne Capital 2026 preprint explored a QAOA ansatz with an XY mixer and Dicke-state initialization for strict-cardinality direct-indexing portfolios. It was tested on only 10 US equities in simulation. The authors reported a backtested Sharpe ratio of 1.81, compared with 1.31 for their Simulated Annealing baseline and 0.98 for Hierarchical Risk Parity. This is a small simulated backtest, not evidence of hardware or scaling advantage [2927].

NISQ A 2026 KDDI Research study evaluated the standard mixer and four XY-family mixers for ternary portfolio optimization on simulated portfolios of 5 and 8 DAX assets. XY mixers were superior in noiseless simulations, but their advantage degraded under depolarizing noise, with the best mixer depending on circuit depth and noise strength. The problem sizes remained too small to establish an advantage over classical solvers [2928].

NISQ BBVA and IBM tested in 2025 a hardware-aware VQE workflow for dynamic portfolio optimization on real Spanish market data, scaling to portfolios of up to 38 assets on IBM Fez and using up to 152 physical qubits. The reported financial performance was on par with the classical methods tested while producing a broader set of high-quality strategies. The study did not demonstrate end-to-end quantum advantage [2929].

NISQ Loughborough University used Quantum Reservoir Computing (QRC) with at most 6 qubits to forecast trading-volume time series for 20 quantum-sector companies and reported trend-classification accuracy above 86% [2930]. The small-scale study does not establish computational advantage. Likewise, a German 2026 study evaluated QRC for revenue time-series forecasting using PennyLane emulation. It reported minor performance gains over classical Reservoir Computing in some multivariable settings, but the experiments remained entirely classically emulable [2931].

NISQ HSBC and Imperial College London mapped in 2026 S&P 500 residual-correlation data into adjacency matrices compatible with Gaussian Boson Sam-

pling (GBS) and used them to cluster comoving assets for market-neutral statistical-arbitrage portfolios over a rolling one-year window. The method compared GBS Boost and a new GBS Roots routine against the classical Spectral and SPONGE algorithms, reporting superior alpha for the quantum-inspired clustering pipeline in large stock universes during high-volatility regimes and little benefit in quiet markets. The GBS component was simulated, so the study does not demonstrate hardware quantum advantage [2932].

NISQ JPMorganChase researchers with Amazon, Quantinuum, and 55 North formulated portfolio diversification as a Maximum Independent Set (MIS) problem on asset-correlation graphs and solved it with qReduMIS, a recursive hybrid algorithm in which QAOA measurements identify frozen nodes that can unblock provably optimal classical graph reductions. It was tested on real data from four major market indices with up to 225 assets and executed on Quantinuum's 98-qubit Helios system, with QAOA kernels using up to 78 qubits and 1,016 two-qubit gates. Standalone QAOA failed to find the optimum on the S&P 100 and Nikkei 225 instances, while qReduMIS reached success probabilities of 0.40 and 0.95, respectively, with average approximation ratios at least 0.96. This is a significant hybrid-hardware result, but it does not establish end-to-end quantum advantage over the best classical portfolio solvers [2933].

analog In Germany and the USA, a 2025 study used a 5,600-qubit D-Wave annealer for financial asset clustering on 170 assets. Despite good processing speed, the study did not establish quantum utility because it did not benchmark against state-of-the-art classical QUBO solvers [2934].

analog Two 2026 preprints questioned the quantum contribution of D-Wave's hybrid solver for portfolio optimization. Their benchmarks against standard classical solvers such as Gurobi indicated that much of the observed performance could be attributed to the classical component of the hybrid workflow [2935]. A second paper by Luis Lozano further analyzed the same issue and reached a similar conclusion [2936].

analog BBVA studied in 2026 multi-period portfolio rebalancing under realistic constraints, including lower and upper weight bounds, class-level exposure limits, and turnover limits. The portfolio-selection problem was formulated as a QUBO and embedded in a CQM solved with D-Wave's hybrid solver. In the reported benchmark, the hybrid quantum solution lagged the classical solvers [2937].

FTQC JPMorganChase and MIT researchers proved in 2026 asymptotic quadratic quantum speedups for derivative pricing beyond Black-Scholes. For the CIR model and a variant of the Heston stochastic-volatility model, the result exploits a property the authors call fast-forwardability. Separately, for certain non-fast-forwardable correlated stochastic processes, they introduce a quantum Milstein sampler that enables quantum

multilevel Monte Carlo. The paper is theoretical and does not provide a complete fault-tolerant resource estimate in thousands of logical qubits [2938].

FTQC An Investec Bank UK and University of A Coruña study outlined in 2026 an end-to-end multi-asset option-pricing pipeline integrating market calibration with Quantum Amplitude Estimation (QAE). In numerical tests, the quantum-amplitude-estimation component required about 10–100 times fewer oracle queries than classical Monte Carlo for comparable precision. This is a query-complexity result obtained in emulation, not a demonstrated wall-clock speedup or a full FTQC resource estimate [2939].

3.7.4 Derivatives and options pricing

Pricing derivatives, risk management, algorithmic trading and portfolio optimization belong to the broad field of quantitative finance [2941].

Derivatives are financial contracts whose value is derived from an underlying asset, such as stocks, bonds, commodities, or market indices.

Derivative pricing determines the fair value of financial contracts from current market data, the dynamics assumed for the underlying risk factors, contract terms, interest rates, volatility, and other parameters. Depending on the product and model, classical pricing uses closed-form formulas, trees and lattices, partial-differential-equation solvers, Fourier methods, or Monte Carlo simulation. Some quantum proposals reformulate selected pricing tasks through linear systems, matrix functions [2942], Markov-chain discretizations [2943], or amplitude-estimation-based Monte Carlo. These are algorithm-specific reformulations, not a universal description of derivative pricing.

Options pricing is a subset of derivative pricing that deals with valuing options contracts. An option gives the holder the right, but not the obligation, to buy or sell an underlying asset at a predetermined price, called the strike price, during an exercise period that ends on the expiration date.

Options can be classified into **call options** (giving the right to buy the underlying asset) and **put options** (giving the right to sell the underlying asset).

Option-pricing models such as the **Black-Scholes-Merton model** estimate option values under explicit assumptions about the stochastic dynamics of the underlying asset [2944]. The standard European-option formula depends on the current underlying price, strike price, time to expiration, risk-free interest rate, volatility, and, when applicable, dividend yield [2945].

NISQ With these variables, algorithms can determine a theoretical option value under a chosen model. Some proposals use quantum generative adversarial networks [2946] or classical Matrix Product States (MPS) to prepare distributions for quantum Monte Carlo [2947]. Ex-

periments using 11–23 qubits do not establish quantum advantage [2948]. The same caution applies to a 10-qubit quantum-reservoir-computing study of S&P 500 volatility [2949].

FTQC Derivative pricing can use quantum Monte Carlo and amplitude-estimation methods, including encodings that support negative prices or payoffs [2950, 2951]. Canonical QAE provides a quadratic reduction in sampling or query complexity relative to Monte Carlo under suitable state-preparation and oracle assumptions, but practical end-to-end speedup also depends on the cost and depth of those operations. Deep coherent implementations are generally associated with FTQC hardware. Derivative pricing can also use QSP (quantum signal processing) methods [2952].

QI Tensor-network classical solutions are also relevant for derivative pricing [2953]. Toshiba also uses its Simulated Bifurcation Machine (SBM), a quantum-inspired classical optimizer, in financial-services studies including trading and asset-management applications [2954].

NISQ Complementary to options pricing is stock-price or market-variable forecasting. Various quantum methods are being studied, including QLSTM (quantum long short-term memory) neural networks. A study using up to 15 physical qubits reported good predictive results, but this scale and result do not establish quantum computational advantage [2955].

FTQC A Tsinghua University group developed in 2026 a fault-tolerant algorithm for derivative pricing under stochastic partial-differential-equation models. It derives a quadratic improvement in sampling complexity over classical Monte Carlo under its assumptions, but does not provide a complete real-world fault-tolerant resource estimate [2956].

FTQC DBS Bank and Classiq created in 2026 a hybrid framework to estimate potential future exposure (PFE) for path-dependent derivatives such as FX Target Redemption Forwards (TARFs). It uses Iterative Quantum Amplitude Estimation (IQAE) in a reduced-order model and was validated with NVIDIA CUDA-Q and Amazon Braket SV1 simulation. The authors reported relative errors of 1–8% against classical benchmarks at 97.5% and 99% confidence levels and estimated that about 300 logical qubits could support a full 52-week exposure calculation. They also note that discretization and approximate-monotonicity assumptions can introduce bias, while deeper circuits are required as the method scales [2957].

3.7.5 Risk analysis

Risk analysis estimates the distribution of possible outcomes for an investment or portfolio under probabilistic models. Classical risk analytics often uses Monte Carlo simulation, which generates many scenarios with pseudorandom or quasi-random sampling. A QRNG can provide physical random bits, but replacing a classical

Q Did IBM and HSBC reach a quantum advantage in 2025 with a bond trading application?

In September 2025, IBM and HSBC published a preprint titled “*Enhanced fill probability estimates in institutional algorithmic bond trading using statistical learning algorithms with quantum computers*” [2894]. The study reported a practically interesting predictive gain, but it did not demonstrate computational quantum advantage.

The task was to estimate the probability that an institutional bond-trading inquiry would be filled. Real production-scale intraday trading data were transformed by circuits executed on IBM Heron processors and, for comparison, by noiseless quantum simulation. The transformed data were then fed into classical machine-learning models. The authors reported up to about 34% relative improvement in out-of-sample test scores for models using hardware-transformed data compared with models using the original data or noiseless-simulation transforms. Importantly, the paper itself states that the empirical results suggest that inherent hardware noise contributes to the effect.

This result shows an empirical feature-engineering effect from a noisy quantum processor, not a computational speedup, classical intractability result, or end-to-end quantum advantage. The downstream

prediction was classical. The main methodological objection stands on its own [2940]: the reported 34% relative predictive gain is an empirical effect of noisy hardware used as a feature transform, and improved predictive quality does not by itself establish quantum computational advantage.

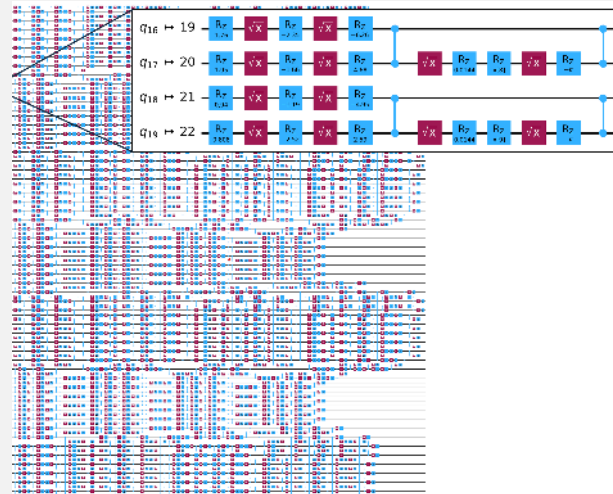


Figure 193: IBM-HSBC circuit. Source: [2894].

pseudorandom generator by a QRNG does not create a quantum computational speedup. Quantum Monte Carlo proposals instead seek speedups through coherent state preparation and amplitude estimation. Quantum walks can appear in specialized state-preparation or sampling algorithms, but they are not a generic replacement for a Monte Carlo random-number generator.

FTQC Matsakos and Nield developed end-to-end quantum Monte Carlo circuits for equity, interest-rate, and credit-risk scenario generation and combined them with QAE [2958]. The work is a resource and algorithmic study, with small proof-of-principle circuits and classical simulation. It does not demonstrate a hardware quantum advantage.

NISQ Many other risk-analysis case studies were also published but were tested only at small scale [2958–2961].

NISQ GE Research and IonQ developed a hybrid copula-based risk-aggregation method and tested it on IonQ Aria [2962, 2963]. The peer-reviewed study trained models with up to 8 qubits and evaluated models with up to 16 qubits for small copula instances. It reported prediction quality comparable to or better than selected classical copula models in some tests, but did not demonstrate a computational speedup over classical simulation.

FTQC CQT Singapore et al. introduced in 2026 a quantum algorithm that discovers and samples rare events

below a probability threshold without first learning which events are rare, motivated by financial crashes and other low-probability triggers. The algorithm achieves optimal quantum scaling for the rare-event primitive, with quadratic speedup in some settings and polynomial speedup for certain stationary stochastic processes. It was not tested on quantum hardware [2964].

3.7.6 Currency

In a master’s thesis at RWTH Aachen University, Cameron Perot studied foreign-exchange applications on D-Wave annealers [2965].

NISQ Czech National Bank compared algorithms for gate-based quantum computers (QAOA, VQE and Grover Adaptive Search), quantum annealers, and classical exact algorithms for currency-related portfolio optimization. The study did not demonstrate an advantage with the available quantum systems [1250].

NISQ IBM tested a VQE-based cryptocurrency-arbitrage formulation using 25 qubits on a 127-qubit QPU, with experiments running for about 12 hours [2966]. The experiment did not demonstrate quantum advantage.

analog An Italian research team tested a currency-arbitrage QUBO on D-Wave Advantage and Advantage2 systems and reported a lower computing time

than the selected classical Tabu Search baseline in 2025 [2344]. This is not sufficient to establish general quantum advantage, particularly because the source and size of the dataset and the number of currencies were not documented.

analog A currency-arbitrage application studied by Korean researchers in 2026 used a resource-efficient QUBO formulation with realistic trading constraints, such as fixed start currencies and transaction fees [2967]. It required fewer qubits than other formulations. The authors derived exact penalty bounds and introduced a gauge transformation that compresses coefficient ranges. The approach was evaluated with D-Wave’s classical simulated annealing rather than on a quantum annealer. An exact classical Held-Karp algorithm outperformed the simulated-annealing model by orders of magnitude in execution speed, so the study demonstrates no current quantum advantage.

3.7.7 Credit scoring

Credit scoring estimates a borrower’s probability of default or ability to repay a loan or credit-card balance. Many proposed quantum credit-scoring applications use quantum machine-learning techniques [2968].

NISQ CACIB (France), Pasqal and Multiverse Computing worked on “fallen angel” detection for credit-risk management. The study used a real-life dataset spanning 2001–2020 with more than 90,000 observations and about 150 features, covering more than 2,000 companies across multiple industries and countries [2969]. The training set contained about 65,000 examples from 2001–2016 and the test set about 26,000 examples from 2016–2020. The authors implemented their quantum-enhanced learning model on a neutral-atom QPU with up to 60 qubits and reported predictive performance competitive with a state-of-the-art random-forest benchmark, with better interpretability and comparable training times [2969]. Tensor-network simulations were used to study larger scales. The paper discusses projected crossovers at hundreds to thousands of qubits under model assumptions, but these are extrapolations rather than a demonstrated quantum advantage.

NISQ In 2025, IBM and American Express tested ensemble models mixing classical and quantum learning models to predict credit-card default using up to 101 physical qubits. The reported quantum-classical ensembles were slightly below the best fully classical models, and the scaling prospects remained undetermined [2970].

NISQ Also in 2025, **Intesa Sanpaolo** in Italy tested a credit-card default prediction method using quantum reservoir computing on a QuEra analog simulator. It reported faster training than the classical deep neural networks used as baselines [573]. In 2026, the bank also studied credit-risk VaR/CVaR models using NISQ variational circuits with minimized gate depth, but the

hardware experiment used only 3 qubits of a 17-qubit QuantWare QPU [2971].

NISQ BBVA built in 2026 a Credit Valuation Adjustment (CVA) workflow that ran on an IBM quantum computer, covering market calibration, discretization, oracle construction, and execution. The test used 7 qubits of a 156-qubit processor. QAE does not “depend on a Grover search” in the ordinary search sense. Amplitude-estimation methods use Grover-like amplitude-amplification operators, and their scalability depends strongly on state preparation, oracle cost, coherent circuit depth, and the number of amplification iterations [2972].

FTQC Fault-tolerant resource estimates vary by orders of magnitude across financial algorithms. For benchmark derivative pricing, Chakrabarti et al. estimate about 8,000 logical qubits and T-depth 5.4×10^7 for their preferred method [1793]. By contrast, Dalzell et al. estimate that a quantum interior-point method for a 100-asset portfolio instance at precision 10^{-7} would require about 8×10^6 logical qubits, roughly 7×10^{29} T gates and T-depth 2×10^{24} [2973]. At a GHz-scale logical clock, such depth would still imply millions of years. Their analysis identifies large constant factors, poor conditioning, tomography, and data-access overheads as major obstacles. These examples are much more informative than comparing the logical-qubit count with Shor factoring.

3.7.8 Fraud detection

NISQ Fraud detection uses various methods including quantum machine learning [2974–2976].

NISQ Many fraud-detection demonstrations remain far below a convincing quantum-advantage regime, including studies using 12 qubits in an emulator [2977], 7 qubits [2978], and 3 qubits with reported accuracy of 82% versus more than 99% for the classical baselines in that study [2979]. Other prototypes map fraud-detection QUBOs onto coherent Ising machines [2980], but such mappings do not by themselves establish quantum advantage.

3.7.9 Other use cases

Here is an inventory of other potential quantum-computing use cases in financial services:

- Solving **dynamic economic models** using interpolation algorithms.
- **Numerical differentiation** algorithms in financial econometrics, structural micro econometrics, maximum likelihood estimation, dynamic stochastic general equilibrium (DSGE) modelling, and large-scale macroeconomic modelling conducted by central banks and government agencies.

- **Macro-economic models** using finite elements methods.
- **Macroeconomics, forecasting** [2981, 2982], **financial market predictions** [2983], **modeling credit spread**, and **pricing financial derivatives** based on quantum machine-learning or quantum numerical methods, including work by Goldman Sachs with IBM [1793, 2984].
- **Simulation of agent choices** over time using Monte Carlo simulations.
- **Model market instability**, also on a quantum annealer [2985].
- **Sentiment analysis** [2986].
- Compute **financial return predictions** [2987].
- Handle **securities transactions settlement** [2988].
- **Quantum games** can help determine correlations between strategic player actions in trading markets, around the notions of Nash equilibria. It could help create more efficient financial strategies [2989].
- **Quantum-Tokens** can store unclonable keys in quantum states in quantum computers that can then be used for personal authentication in a physical device issued by a bank. It is a concept not far from qPUFs (quantum physical unclonable function) that we describe in the quantum communication part of this book [2990].

analog **Satispay** (Italy) announced in October 2023 its intention to deploy a quantum-hybrid application using D-Wave’s CQM hybrid solver for mobile-payment customer rewards. The company announcement claimed a 50% improvement in the optimization metric. This case is documented only by a press release.

FTQC **CQT** created an algorithm solving the market equilibrium problem which is about computing the balance of supply and demand that determines the optimal price and allocation of goods in a market so that the supply equals the demand [2991]. It provides only a theoretical polynomial speedup.

3.7.10 Cybersecurity

Quantum-related cybersecurity in financial services mainly concerns migration to post-quantum cryptography (PQC) and, for selected high-value communication links, quantum key distribution (QKD) [2992]. A quantum Internet is not required for post-quantum security. Various PQC migration projects are ongoing in finance, including Project LEAP led by the BIS Innovation Hub Eurosystem Centre, Banque de France and Deutsche Bundesbank [2993].

3.7.11 Quantum money

The Hull/Diamanti et al. paper [2877] reviews the broad topic of quantum money, an idea developed by **Stephen Wiesner** around 1970 and published in 1983. In Wiesner’s private-key scheme, a banknote contains a public classical serial number and a quantum state made of many independently prepared two-level systems. The issuing bank keeps a secret classical description of the prepared states and bases associated with that serial number. In qubit notation, each system can be prepared in one of four BB84-like states, $|0\rangle$, $|1\rangle$, $|+\rangle$, or $|-\rangle$. With polarization encoding, these correspond for example to four polarizations such as 0° , 90° , 45° , and 135° , not only 0° and 45° . Verification uses the bank’s secret basis information. The no-cloning theorem and measurement disturbance then limit counterfeiting. Many later private-key, public-key, and semi-quantum variants have been proposed, including a 2024 public-key semi-quantum money scheme [2994].

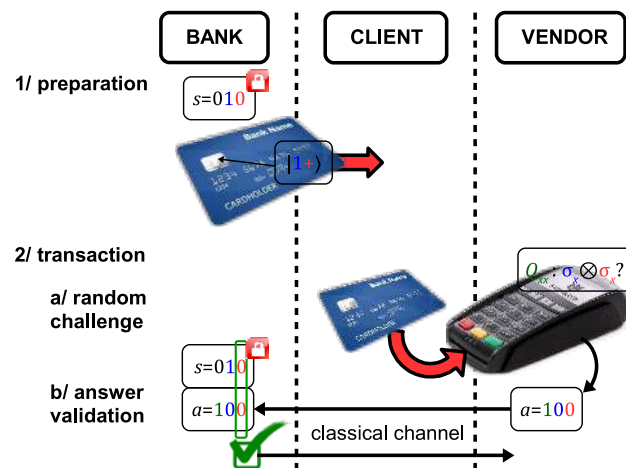


Figure 195: A quantum money scheme. Source: [2877].

Quantum-money proposals include banknote-like private-key schemes, coin or token schemes, and public-key constructions such as Quantum Lightning. The main obstacle is not that quantum memory “does not exist”. Experimental quantum memories do exist, and a quantum-money experiment has operated in a regime compatible with state-of-the-art memories [2995]. The practical challenge is achieving sufficiently long storage time, efficiency, fidelity, compactness, robustness, and tamper resistance for portable money or payment tokens. Quantum Lightning is designed so that it should be computationally infeasible to create two valid quantum “bolts” with the same serial number, rather than simply preventing a bank from issuing duplicates. Some newer token schemes avoid long-lived quantum memories or long-distance quantum communication. Overall, the practicality of quantum money remains highly uncertain.



best analysts report to date: december 2019,
11 pages, 1.5 pages on « solutions »

showcased QC applications domains:

- quantum machine learning to better detect and mitigate fraud.
- risks assessment with actuarial models for enhanced pricing and risk pooling precision.
- portfolio optimization.
- model life expectancy for large populations.
- faster (no) and more secure data transfer (yes).

one Evergreen
Actuarial case study
running on D-Wave,
on optimizing
Solvency matching
adjustment

Insurance- Optimisation of the Solvency II Matching Adjustment

Evergreen Actuarial

Insurance Optimization

An exploration of whether quantum computing can be used to select an optimal subset of assets such that an insurer's Solvency II matching adjustment is maximised. If an insurer has 1,000 assets, such as corporate and government bonds, then there are 2^{1000} permutations of assets that could be selected. Fitting at order of 10^{301} means that it would not be possible for a classical computer to consider each permutation. This work considers a simplified matching adjustment problem and tests whether a quantum computing approach finds the optimal solution.

Reinsurance News

Quantum computing a potential cyber risk for re/insurers: Fitch

12th November 2019 - Author: Charlie Wood

The day when quantum computing power can be applied to real world scenarios is fast approaching, posing a number of important questions around the parameters of cyber risk and security of data encryption.

Fitch Ratings analysts note how quantum computers – estimated to run 100 million times faster than current technology – stand to revolutionise research efforts, new product development and operating efficiency.

Concurrently, Fitch warns of the potential implications should a 'bad actor' be the first to fully develop and make operational a quantum computer.



wrong risk
assessment: we're
far far away from a
quantum computer
breaking RSA codes

Figure 194: Some use cases and constraints for quantum computing in the insurance business. © Olivier Ezratty, 2021.

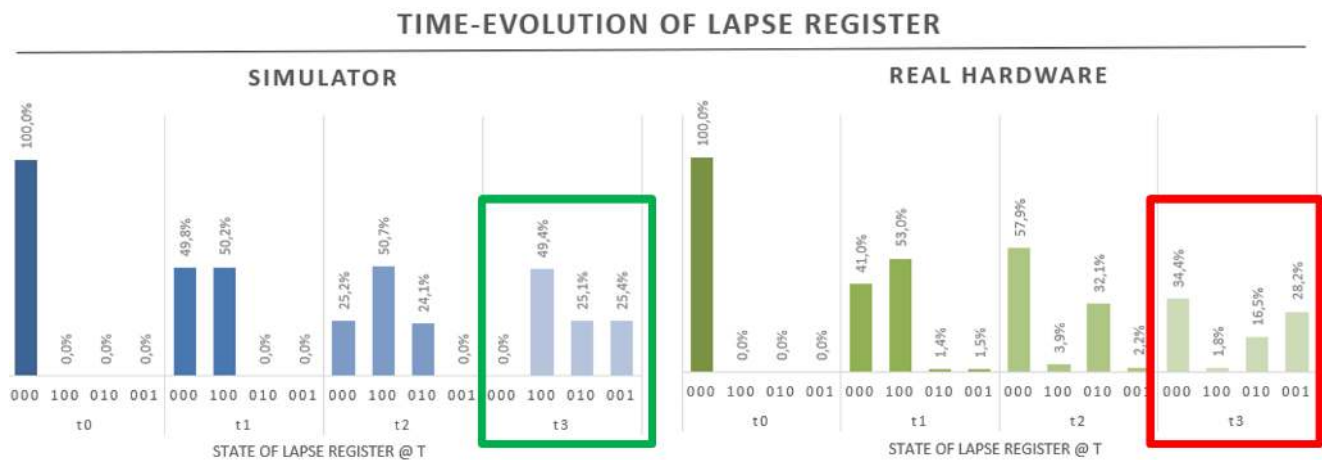


Figure 196: AXA Konzern AG trial of a quantum amplitude-estimation algorithm for insurance-contract valuation, showing in red the detrimental effect of qubit noise with 27 qubits on an IBM QPU. Source: [2996]. Added in 2023.

3.7.12 Insurance

The insurance market also has complex optimization and simulation problems, particularly in risk modelling. The related surveys and review papers I have found are less extensive than those for banking and financial markets [2997].

Some analyst reports use the future Shor-based threat to public-key cryptography as a cybersecurity warning for insurers and their customers [2998].

Some related reports are misleading, for example when they state that quantum-based communications could be “quicker over long distances” as well as more secure [2999]. QKD can provide information-theoretic key-distribution security under protocol and device assumptions, but it does not make ordinary data transmission intrinsically faster.

An 11-page report was published in late 2019 by **Novarica**, a US insurance consulting company [3000]. Besides a generic introduction to quantum computing, it devoted about one and a half pages to insurance use cases. These included risk modelling, portfolio optimization, quantum machine learning for fraud detection, actuarial models for pricing and risk pooling, and life-expectancy modelling for large populations (Figure 194).

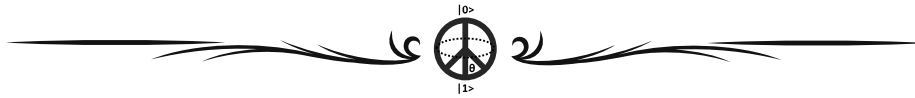
NISQ Another paper published in 2022 by Michael Adam from **AXA Konzern AG** investigated quantum computing for insurance-contract valuation using a phase-estimation-based QAE circuit and QFT components, comparing the approach with classical Monte Carlo [2996]. It was tested on IBM hardware with 27 qubits, where noise strongly degraded the result, as shown below. The particular phase-estimation/QFT-based QAE implementation is deep and noise-sensitive and is more naturally associated with future fault-

tolerant hardware. However, QAE is not intrinsically tied to QPE or an inverse QFT: later variants such as Iterative QAE avoid QPE and reduce qubit and gate requirements [3001]. The AXA paper did not provide a full FTQC resource estimate for quantum advantage (Figure 196). In 2025, AXA and AWS published a blueprint on reinsurance optimization using a quantum branch-and-bound approach, with resource estimates concluding that practical quantum advantage would be difficult to obtain [3002].

analog One risk-modelling algorithm created by **JoS QUANTUM** is documented [3003]. Another case was publicized by **CaixaBank** with D-Wave in 2022. Their vendor case study reported a 90% reduction in time-to-solution for an investment-portfolio hedging and optimization workflow compared with the bank's legacy classical system [3004]. But it relies on D-Wave's hybrid CQM solver so it can't be interpreted as a demonstrated quantum speedup.

NISQ In 2023, researchers from **Antares Global** published a peer-reviewed paper on quantum computing for insurance capital modelling under reinsurance contracts, not on cyber-risk estimation. The quantum circuit represents the surplus process of a cedant and includes initial reserve, insurance and reinsurance premiums, claim amounts, retention levels, and deductibles for two non-proportional reinsurance contracts [3005].

NISQ **Allstate** and IBM created in 2026 a hybrid QAOA-based method for chance-constrained knapsack problems motivated by insurance underwriting. Experiments on IBM Heron processors used circuit depths up to 177, 3,443 gates, and as many as 150 qubits. The authors report high-quality solutions and indications of improvement over the classical optimization schemes they tested, but the study does not establish a general asymptotic or wall-clock quantum advantage [3006].



3.7.13 Bibliography and notes

These are the bibliographical references and notes for the **Financial services and insurance** subsection.

- [108] [Low Depth Phase Oracle Using a Parallel Piecewise Circuit](#), by Zhu Sun, Nikitas Stamatopoulos, et al., arXiv, October 2024 (14 pages).
- [573] [Quantum Reservoir Computing for Credit Card Default Prediction on a Neutral Atom Platform](#), by Giacomo Vitali, Davide Corbelleto, et al., arXiv, October 2025 (21 pages).
- [1250] [Finding the Optimal Currency Composition of Foreign Exchange Reserves with a Quantum Computer](#), by Martin Vesely, arXiv, March 2023 (30 pages).
- [1793] [A Threshold for Quantum Advantage in Derivative Pricing](#), by Shouvanik Chakrabarti, Rajiv Krishnakumar, Guglielmo Mazzola, Nikitas Stamatopoulos, Stefan Woerner, and William J. Zeng, Quantum, June 2021 (41 pages).
- [1799] [End-to-end quantum algorithms for tensor problems](#), by Enrico Fontana, Shouvanik Chakrabarti, et al., arXiv, October 2025 (53 pages).
- [2344] [Solving Currency Arbitrage Problems using D-Wave Advantage2 Quantum Annealer](#), by Lorenzo Mazzei, Giada Becari, Mirko Laruina, and Marco Cococcioni, arXiv, September 2025 (6 pages).
- [2870] [Quantum computing for finance: Overview and prospects](#), by Román Orús, Samuel Mugel, and Enrique Lizaso, Reviews in Physics, November 2019 (12 pages).
- [2871] [A Survey of Quantum Computing for Finance](#), by Dylan Herman, Cody Googin, Xiaoyuan Liu, Alexey Galda, Ilya Safro, Yue Sun, Marco Pistoia, and Yuri Alexeev, arXiv, June 2022 (60 pages).
- [2872] [Quantum computing for finance](#), by Dylan Herman, Marco Pistoia, Yuri Alexeev, et al., Nature Reviews Physics, July 2023 (preprint on arXiv).
- [2873] [Quantum Computing - Moving Quickly From Theory to Reality](#), by Citi, July 2023 (154 pages).
- [2874] [Quantum Technologies: Key Strategies and Opportunities for Financial Services Leaders](#), by Adam Burden et al., World Economic Forum, July 2025 (32 pages).
- [2875] [Quantum Computing for Finance: State-of-the-Art and Future Prospects](#), by Daniel J. Egger, Claudio Gambella, Jakub Marecek, Scott McFaddin, Martin Mevissen, Rudy Raymond, Andrea Simonetto, Stefan Woerner, and Elena Yndurain, IEEE Transactions on Quantum Engineering, October 2020 (24 pages).
- [2876] [A Brief Review of Quantum Machine Learning for Financial Services](#), by Mina Doosti, Petros Wallden, Conor Brian Hamill, Robert Hankache, Oliver Thomson Brown, and Chris Heunen, arXiv, July 2024 (19 pages).
- [2877] [Quantum Technology for Economists](#), by Isaiah Hull, Or Sattath, Eleni Diamanti, and Göran Wendin, arXiv, October 2021 (106 pages).
- [2878] [Portfolio construction using a sampling-based variational quantum scheme](#), by Gabriele Agliardi, Bimal Mehta, et al., arXiv, August 2025 (7 pages).
- [2879] [It's Time for Financial Institutions to Place Their Quantum Bets](#), by Jean-François Bobier, 2020.
- [2880] [How quantum computing could change financial services](#), by Miklos Dietz, December 2020.
- [2881] [Mechanisms for Quantum Advantage in Global Optimization of Nonconvex Functions](#), by Dylan Herman, Shouvanik Chakrabarti, et al., arXiv, October 2025 (85 pages).
- [2882] [Bank of Canada and Multiverse Computing Complete Preliminary Quantum Simulation of Cryptocurrency Market](#), by Multiverse, April 2022.
- [2883] [Quantum Monte Carlo for Economics: Stress Testing and Macroeconomic Deep Learning](#), by Vladimir Skavysh, Sofia Priazhkina, Diego Guala, and Thomas R. Bromley, arXiv, September 2024 (54 pages).
- [2884] [JP Morgan Chase poaches an IBM 'Master Inventor' with 26 patents for quantum computing](#), by Hugh Son, January 2020.
- [2885] [Option Pricing using Quantum Computers](#), by Nikitas Stamatopoulos, Daniel J. Egger, Yue Sun, Christa Zoufal, Raban Iten, Ning Shen, and Stefan Woerner, arXiv, July 2020 (20 pages).
- [2886] [Why banks like Barclays are testing quantum computing](#), by Penny Crossman, American Banker, July 2018.
- [2887] [Barclays demonstrates proof-of-concept quantum clearing algorithm](#), by Cliff Saran, October 2019.

- [2888] [Quantum Algorithms for Mixed Binary Optimization applied to Transaction Settlement](#), by Lee Braine, arXiv, October 2019 (8 pages).
- [2889] [Quantum Machine Learning in Finance: Time Series Forecasting](#), by Dimitrios Emmanoulopoulos and Sofija Dimoska, arXiv, February 2022 (20 pages).
- [2890] [HSBC Working with IBM to Accelerate Quantum Computing Readiness](#), by IBM and HSBC, March 2022.
- [2891] [Unsupervised quantum machine learning for fraud detection](#), by Oleksandr Kyriienko and Einar B. Magnusson, arXiv, August 2022 (7 pages).
- [2892] [Approaching Collateral Optimization for NISQ and Quantum-Inspired Computing](#), by Megan Giron, Georgios Korpas, Waqas Parvaiz, Prashant Malik, and Johannes Aspmann, arXiv, December 2023 (14 pages).
- [2893] [Quantum Multiple Kernel Learning in Financial Classification Tasks](#), by Shungo Miyabe, Jae-Eun Park, et al., arXiv, December 2023 (15 pages).
- [2894] [Enhanced fill probability estimates in institutional algorithmic bond trading using statistical learning algorithms with quantum computers](#), by Axel Ciceri, Kavitha Yograj, et al., arXiv, September 2025 (16 pages).
- [2895] [How Wells Fargo is becoming quantum-ready](#), by Christopher Frers, May 2024.
- [2896] [Quantum Risk Analysis of Financial Derivatives](#), by Nikitas Stamatopoulos, B. David Clader, Stefan Woerner, and William J. Zeng, arXiv, April 2024 (21 pages).
- [2897] [QC Ware Applies Quantum Computing Principles to Increase Customer Retention at Itaú Unibanco](#), by James Dargan, The Quantum Insider, May 2022.
- [2898] [Enhancing Quantum Expectation Values via Exponential Error Suppression and CVaR Optimization](#), by Touheed Anwar Atif, Reuben Blake Tate, and Stephan Eidenbenz, arXiv, January 2025 (17 pages).
- [2899] [In the Fight Against Noisy Quantum Computing, CVaR Proves a Worthy Opponent in the Fight Against Noisy Quantum Computing, CVaR Proves a Worthy Opponent](#), by Cierra Choucair, The Quantum Insider, November 2024.
- [2900] [Provable bounds for noise-free expectation values computed from noisy samples](#), by Samantha V. Barron, Stefan Woerner, et al., Nature Computational Science, November 2024.
- [2901] [Implementing Credit Risk Analysis with Quantum Singular Value Transformation](#), by Davide Veronelli, Francesca Cibrario, Emanuele Dri, Valeria Zaffaroni, Giacomo Ranieri, Davide Corbelleto, and Bartolomeo Montrucchio, arXiv, July 2025 (10 pages).
- [2902] [Conditional value-at-risk for general loss distributions](#), by R. Tyrrell Rockafellar and Stanislav Uryasev, Journal of Banking & Finance, July 2002 (29 pages)  (preprint on doi.org).
- [2903] [Scaling the Variational Quantum Eigensolver for Dynamic Portfolio Optimization](#), by Álvaro Nodar, Inmaculada Posadillo, et al., arXiv, February 2025 (18 pages).
- [2904] [Achieving High-Quality Portfolio Optimization with the Variational Quantum Eigensolver](#), by Anbang Wang, Zhonggang Lv, Zhenyuan Ma, Dunbo Cai, and Zhihong Zhang, arXiv, August 2025 (7 pages).
- [2905] [Quantifying the advantages of applying quantum approximate algorithms to portfolio optimisation](#), by Haomu Yuan, Christopher K. Long, Hugo V. Lepage, and Crispin H. W. Barnes, arXiv, October 2024 (35 pages).
- [2906] [DATEV and IQM Explore Quantum Computing for Portfolio Optimization Using QAOA](#), by Cierra Choucair, The Quantum Insider, January 2025.
- [2907] [Quantum Portfolio Optimization with Expert Analysis Evaluation](#), by Nouhaila Innan, Ayesha Saleem, Alberto Marchisio, and Muhammad Shafique, arXiv, July 2025 (6 pages).
- [2908] [Scaling Portfolio Diversification with Quantum Circuit Cutting Techniques](#), by Vicente P. Soloviev, Antonio Márquez Romero, Josh Kirsopp, and Michal Krompiec, arXiv, June 2025 (17 pages).
- [2909] [Higher-Order Portfolio Optimization with Quantum Approximate Optimization Algorithm](#), by Valter Uotila, Julia Ripatti, and Bo Zhao, arXiv, September 2025 (12 pages).
- [2910] [Quantum Portfolio Optimization: An Extensive Benchmark](#), by Eric Stopfer and Friedrich Wagner, arXiv, September 2025 (17 pages).
- [2911] [Quantum computational finance: quantum algorithm for portfolio optimization](#), by Patrick Rebstroff and Seth Lloyd, arXiv, November 2018 (18 pages).
- [2912] [Benchmarking the performance of portfolio optimization with QAOA](#), by Sebastian Brandhofer, Daniel Braun, Vanessa Dehn, Gerhard Hellstern, Matthias Hüls, Yanjun Ji, Ilia Polian, Amandeep Singh Bhatia, and Thomas Wellens, arXiv, July 2022 (28 pages).
- [2913] [Solving the Optimal Trading Trajectory Problem Using a Quantum Annealer](#), by Gili Rosenberg, Poya Haghnegahdar, Phil Goddard, Peter Carr, Kesheng Wu, and Marcos López de Prado, arXiv, August 2016 (7 pages).
- [2914] [Comparative analysis of diverse methodologies for portfolio optimization leveraging quantum annealing techniques](#), by Zhijie Tang, Alex Lu Dou, and Arit Kumar Bishwas, arXiv, July 2024 (12 pages).
- [2915] [A real world test of Portfolio Optimization with Quantum Annealing](#), by Wolfgang Sakuler, Johannes M. Oberreuter, Riccardo Aiolfi, Luca Asproni, Branislav Roman, and Jürgen Schiefer, arXiv, March 2023 (32 pages).
- [2916] [Diversifying Investments and Maximizing Sharpe Ratio: a novel QUBO formulation](#), by Mirko Mattesi, Luca Asproni, Christian Mattia, Simone Tafano, Giacomo Ranieri, Davide Caputo, and Davide Corbelleto, arXiv, February 2024 (23 pages).
- [2917] [Black-Litterman Portfolio Optimization with Noisy Intermediate-Scale Quantum Computers](#), by Chi-Chun Chen, San-Lin Chung, and Hsi-Sheng Goan, arXiv, December 2023 (11 pages).
- [2918] [NEASQC financial application library v1.0 available](#), by NEASQC, 2022.
- [2919] [On the potential of quantum walks for modeling financial return distributions](#), by Stijn De Backer, Luis E. C. Rocha, Jan Ryckebusch, and Koen Schoors, arXiv, December 2024 (24 pages).
- [2920] [End-to-End Protocol for High-Quality QAOA Parameters with Few Shots](#), by Tianyi Hao, Zichang He, Ruslan Shaydulin, Jeffrey Larson, and Marco Pistoia, arXiv, October 2024 (14 pages).
- [2921] [Decomposition Pipeline for Large-Scale Portfolio Optimization with Applications to Near-Term Quantum Computing](#), by Atithi Acharya, Helmut G. Katzgraber, Marco Pistoia, et al., arXiv, November 2024 (19 pages).
- [2922] [Alignment between Initial State and Mixer Improves QAOA Performance for Constrained Optimization](#), by Zichang He, Marco Pistoia, et al., arXiv, January 2024 (14 pages).
- [2923] [Solving Linear Systems on Quantum Hardware with Hybrid HHL++, by Romina Yalovetzky, Pierre Minssen, Dylan Herman, and Marco Pistoia, arXiv, July 2024 \(18 pages\).](#)
- [2924] [Exploring the synergistic potential of quantum annealing and gate model computing for portfolio optimization](#), by Naman Jain and M Girish Chandra, arXiv, May 2023 (12 pages).
- [2925] [Quasi-binary encoding based quantum alternating operator ansatz](#), by Bingren Chen, Hanqing Wu, Haomu Yuan, Lei Wu, and Xin Li, arXiv, January 2024 (23 pages).
- [2926] [Large-scale portfolio optimization on a trapped-ion quantum computer](#), by Alejandro Gomez Cadavid, Ananth Kaushik, Pranav Chandarana, Miguel Angel Lopez-Ruiz, Gaurav Dev, Willie Aboumrard, Qi Zhang, Claudio Girotto,

- Sebastián V. Romero, Martin Roetteler, Enrique Solano, Marco Pistoia, and Narendra N. Hegade, arXiv, February 2026 (10 pages).
- [2927] [Constrained Portfolio Optimization via Quantum Approximate Optimization Algorithm \(QAOA\) with XY-Mixers and Trotterized Initialization: A Hybrid Approach for Direct Indexing](#), by Javier Mancilla, Theodoros D. Bouloumis, and Frederic Goguikian, arXiv, February 2026 (11 pages).
- [2928] [Performance Comparison of QAOA Mixers for Ternary Portfolio Optimization](#), by Shintaro Yamamura, Satoshi Watanabe, Masaya Kunimi, Kazuhiro Saito, and Tetsuro Nikuni, arXiv, February 2026 (31 pages).
- [2929] [Variational Quantum Eigensolver for Real-World Finance: Scalable Solutions for Dynamic Portfolio Optimization Problems](#), by Irene De León, Álvaro Nodar, et al., arXiv, December 2025 (22 pages).
- [2930] [A Quantum Reservoir Computing Approach to Quantum Stock Movement Forecasting in Quantum-Invested Markets](#), by Wendy Otieno, Alexandre Zagoskin, Alexander G. Balanov, Juan Toterogongora, and Sergey E. Savell'ev, arXiv, May 2026 (16 pages).
- [2931] [Learning Temporal Patterns in Financial Time Series: A Comparative Study of Quantum LSTM and Quantum Reservoir Computing](#), by Danyal Maheshwari, Gerhard Hellstern, Martin Zaefferer, Martin Braun, and Tanja Döhler, arXiv, May 2026 (17 pages).
- [2932] [Gaussian Boson Sampling for Asset Clustering in Statistical Arbitrage Portfolios](#), by Dayne Marcus Lopena, Daniel Buguks, Zhenghao Li, Ewan Mer, Shana H. Winston, Shang Yu, Mihai Cucuringu, Del Rajan, Philip Intallura, and Raj B. Patel, arXiv, July 2026 (23 pages).
- [2933] [Quantum-Informed Portfolio Selection: An End-to-End Pipeline Validated on Trapped-Ion Hardware with Real Market Data](#), by Romina Yalovetzky, Helmut G. Katzgraber, Roger Bongiovanni, Niraj Kumar, Rob Otter, et al., arXiv, July 2026 (29 pages).
- [2934] [Toward Quantum Utility in Finance: A Robust Data-Driven Algorithm for Asset Clustering](#), by Shivam Sharma, Supreeth Mysore Venkatesh, and Pushkin Kachroo, arXiv, September 2025 (9 pages).
- [2935] [A Penalty-Free Pipeline for Direct Quantum-Annealer Portfolio Optimization](#), by Luis Lozano, arXiv, May 2026 (22 pages).
- [2936] [Where the Quantum Lives in D-Wave Hybrid Portfolio Optimization: An Operational Decomposition Audit](#), by Luis Lozano, arXiv, June 2026 (31 pages).
- [2937] [Quantum Annealing for Dynamic Portfolio Optimization under Realistic Transaction Costs](#), by Escolástico Sánchez-Martínez, Senaida Hernandez Santana, Ventura Sarasa Laborda, Pablo Serrano Molinero, Guillermo Botella Juan, and Alberto del Barrio García, arXiv, July 2026 (11 pages).
- [2938] [Quantum Speedups for Derivative Pricing Beyond Black-Scholes](#), by Dylan Herman, Yue Sun, Jin-Peng Liu, Marco Pistoia, Charlie Che, Rob Otter, Shouvanik Chakrabarti, and Aram Harrow, arXiv, February 2026 (105 pages).
- [2939] [Quantum computing for multidimensional option pricing: End-to-end pipeline](#), by Julien Hok and Álvaro Leitao, arXiv, January 2026 (26 pages).
- [2940] [HSBC unleashes yet another “qombie”: a zombie claim of quantum advantage that isn’t](#), by Scott Aaronson, Shtetl-Optimized, September 2025.
- [2941] [Applications of Quantum Machine Learning for Quantitative Finance](#), by Piotr Mironowicz, Akshata Shenoy H., Antonio Mandarino, A. Ege Yilmaz, and Thomas Ankenbrand, arXiv, May 2024 (39 pages).
- [2942] [Quantum computational finance: martingale asset pricing for incomplete markets](#), by Patrick Rebentrost, Alessandro Luongo, Samuel Bosch, and Seth Lloyd, arXiv, September 2022 (31 pages).
- [2943] [Quantum Computational Algorithms for Derivative Pricing and Credit Risk in a Regime Switching Economy](#), by Eric Ghysels, Jack Morgan, and Hamed Mohammadbagherpoor, arXiv, November 2023 (13 pages).
- [2944] [Quantum option pricing via the Karhunen-Loève expansion](#), by Anupam Prakash, Yue Sun, Shouvanik Chakrabarti, Charlie Che, Aditi Dandapani, Dylan Herman, Niraj Kumar, Shree Hari Sureshbabu, Ben Wood, Iordanis Kerenidis, and Marco Pistoia, arXiv, February 2024 (25 pages).
- [2945] [Quantum Algorithm for Local-Volatility Option Pricing via the Kolmogorov Equation](#), by Nikita Guseynov, Javier Gonzalez-Conde, et al., arXiv, November 2025 (20 pages).
- [2946] [Efficient option pricing with unary-based photonic computing chip and generative adversarial learning](#), by Hui Zhang, Lingxiao Wan, Sergi Ramos-Calderer, Yuancheng Zhan, Wai-Keong Mok, Hong Cai, Feng Gao, Xianshu Luo, Guo-Qiang Lo, Leong Chuan Kwek, José Ignacio Latorre, and Ai Qun Liu, arXiv, August 2023 (11 pages).
- [2947] [Time series generation for option pricing on quantum computers using tensor network](#), by Nozomu Kobayashi, Yoshiyuki Suimon, and Koichi Miyamoto, arXiv, February 2024 (15 pages).
- [2948] [Quantum Circuits for the Black-Scholes equations via Schrödingerisation](#), by Shi Jin, Zihao Tang, Xu Yin, and Lei Zhang, arXiv, May 2025 (25 pages).
- [2949] [Quantum Reservoir Computing for Realized Volatility Forecasting](#), by Qingyu Li, Chiranjib Mukhopadhyay, Abolfazl Bayat, and Ali Habibnia, arXiv, May 2025 (18 pages).
- [2950] [Real Option Pricing using Quantum Computers](#), by Alberto Manzano, Gonzalo Ferro, Álvaro Leitao, Carlos Vázquez, and Andrés Gómez, arXiv, July 2024 (32 pages).
- [2951] [Quantum Monte Carlo algorithm for solving Black-Scholes PDEs for high-dimensional option pricing in finance and its complexity analysis](#), by Jianjun Chen, Yongming Li, and Ariel Neufeld, arXiv, January 2023 (58 pages).
- [2952] [Derivative Pricing using Quantum Signal Processing](#), by Nikitas Stamatopoulos and William J. Zeng, Quantum Journal, April 2024 (15 pages).
- [2953] [A highly efficient tensor network algorithm for multi-asset Fourier options pricing](#), by Michael Kastoryano and Nicola Pancotti, arXiv, March 2022 (9 pages).
- [2954] [The world’s first demonstration of systems that execute unprecedented stock trading strategies based on computationally-hard quadratic discrete optimization by using quantum-inspired computer](#), Toshiba’s simulated bifurcation machine, by Toshiba, December 2023.
- [2955] [A Hybrid Quantum-Classical Model for Stock Price Prediction Using Quantum-Enhanced Long Short-Term Memory](#), by Kimleang Kea, Dongmin Kim, et al., Entropy, November 2024 (19 pages).
- [2956] [Quantum Derivative Pricing for SPDEs via BDSDE Representation](#), by Xinmiao Li, Yanqiao Wang, Rundu Lu, Zhengwei Liu, and Jin-Peng Liu, arXiv, June 2026 (70 pages).
- [2957] [Quantum Counterparty Credit Risk: A Study of Path-Dependent Derivatives](#), by Sandeep Jha, Rei Sato, et al., arXiv, June 2026 (11 pages).
- [2958] [Quantum Monte Carlo simulations for financial risk analytics: scenario generation for equity, rate, and credit risk factors](#), by Titos Matsakos and Stuart Nield, Quantum, April 2024.
- [2959] [Quantum Deep Hedging](#), by El Amine Cherrat, Snehal Raj, Iordanis Kerenidis, Abhishek Shekhar, Ben Wood, Jon Dee, Shouvanik Chakrabarti, Richard Chen, Dylan Herman, Shaohan Hu, Pierre Minssen, Ruslan Shaydulin, Yue Sun, Romina Yalovetzky, and Marco Pistoia, arXiv, November 2023 (43 pages).
- [2960] [Towards practical Quantum Credit Risk Analysis](#), by Emanuele Dri, Edoardo Giusto, Antonello Aita, and Bartolomeo Montrucchio, arXiv, December 2022 (12 pages).
- [2961] [Low depth amplitude estimation on a trapped ion quantum computer](#), by Tudor Giurgica-Tiron, Sonika Johri, Iordanis

- Kerenidis, Jason Nguyen, Neal Pisenti, Anupam Prakash, Ksenia Sosnova, Ken Wright, and William Zeng, arXiv, September 2021 (12 pages).
- [2962] [IonQ and GE Research Demonstrate High Potential of Quantum Computing for Risk Aggregations](#), by IonQ and GE Research, June 2022.
- [2963] [Copula-based risk aggregation with trapped ion quantum computers](#), by Daiwei Zhu, Weiwei Shen, Annarita Giani, Saikat Ray-Majumder, Bogdan Neculaes, and Sonika Johri, Scientific Reports, October 2023  (preprint on [dx.doi.org](https://doi.org)).
- [2964] [Quantum enhanced rare event discovery and sampling](#), by Naixu Guo, Po-Wei Huang, Qisheng Wang, Jayne Thompson, Patrick Rebstroff, Mile Gu, and Chengran Yang, arXiv, June 2026 (36 pages).
- [2965] [Quantum Boltzmann Machines: Applications in Quantitative Finance](#), by Cameron Perot, arXiv, January 2023 (62 pages).
- [2966] [Differential Evolution VQE for Crypto-currency Arbitrage. Quantum Optimization with many local minima](#), by Gines Carrascal, Beatriz Roman, Guillermo Botella, and Alberto del Barrio, arXiv, August 2023 (17 pages).
- [2967] [Resource-Efficient QUBO Formulation for Anchored Currency Arbitrage](#), by Eric A. F. Reinhardt and Adam J. Hauser, arXiv, August 2026 (18 pages).
- [2968] [Quantum Machine Learning for Credit Scoring](#), by Nikolaos Schetakis, Davit Aghamalyan, Michael Boguslavsky, Agnieszka Rees, Marc Raktomalala, and Paul Griffin, arXiv, August 2023 (12 pages).
- [2969] [Financial risk management on a neutral atom quantum processor](#), by Lucas Leclerc, Luis Ortiz-Gutiérrez, Sebastián Grijalva, Boris Albrecht, Julia R. K. Cline, Vincent E. Elfving, Adrien Signoles, Loïc Henriët, Gianni Del Bimbo, Usman Ayub Sheikh, Maitree Shah, Luc Andrea, Faysal Ishtiaq, Andoni Duarte, Sam Mugel, Irene Cáceres, Michel Kurek, Roman Orús, Achraf Seddik, Oumaima Hammami, Hacene Isselane, and Didier M'tamon, Physical Review Research, November 2023.
- [2970] [Credit Default Prediction with Projected Quantum Feature Models and Ensembles](#), by Andras Ferenczi, Dagen Wang, Mariya Bessonova, Sutapa Samanta, Todd Hodges, John Hancock, Guillermo Mijares Vilariño, Amol Deshmukh, Mariana LaDue, Girish Pillai, and Hilary Packer, arXiv, October 2025 (10 pages).
- [2971] [Quantum Circuit-Based Adaptation for Credit Risk Analysis](#), by Halima Giovanna Ahmad, Davide Corbelleto, et al., arXiv, May 2026 (27 pages).
- [2972] [A Noise-Aware Quantum Algorithm for Credit Valuation Adjustments on Real Quantum Hardware](#), by Guillem Borràs Espert, Francisco Gómez Casanova, Luis de Pedro Sánchez, Senaida Hernández Santana, and Pablo Serrano Molinero, arXiv, July 2026 (56 pages).
- [2973] [End-To-End Resource Analysis for Quantum Interior-Point Methods and Portfolio Optimization](#), by Alexander M. Dalzell, B. David Clader, Grant Salton, Mario Berta, Cedric Yen-Yu Lin, David A. Bader, Nikitas Stamatopoulos, Martin J. A. Schuetz, Fernando G. S. L. Brandão, Helmut G. Katzgraber, and William J. Zeng, PRX Quantum, November 2023.
- [2974] [Mixed Quantum-Classical Method For Fraud Detection with Quantum Feature Selection](#), by Michele Grossi, Andreas Rudnik, et al., arXiv, August 2022 (11 pages).
- [2975] [Financial Fraud Detection: A Comparative Study of Quantum Machine Learning Models](#), by Nouhaila Innan, Muhammad Al-Zafar Khan, and Mohamed Bennai, arXiv, August 2023 (30 pages).
- [2976] [QFNN-FFD: Quantum Federated Neural Network for Financial Fraud Detection](#), by Nouhaila Innan, Alberto Marchisio, Mohamed Bennai, and Muhammad Shafique, arXiv, July 2025 (9 pages).
- [2977] [Toward Practical Quantum Machine Learning: A Novel Hybrid Quantum LSTM for Fraud Detection](#), by Rushikesh Ubale, Sujan K.K., Sangram Deshpande, and Gregory T. Byrd, arXiv, April 2025 (11 pages).
- [2978] [Comparative Performance Analysis of Quantum Machine Learning Architectures for Credit Card Fraud Detection](#), by Mansour El Alami, Nouhaila Innan, Muhammad Shafique, and Mohamed Bennai, arXiv, January 2025 (12 pages).
- [2979] [QFDNN: A Resource-Efficient Variational Quantum Feature Deep Neural Networks for Fraud Detection and Loan Prediction](#), by Subham Das, Ahmed Farouk, et al., arXiv, April 2025 (12 pages).
- [2980] [Quantum Computing in Community Detection for Anti-Fraud Applications](#), by Yanbo (Justin) Wang et al., Entropy, November 2024 (9 pages).
- [2981] [Improved Financial Forecasting via Quantum Machine Learning](#), by Sohumi Thakkar, Skander Kazdaghi, Natansh Mathur, Iordanis Kerenidis, André J. Ferreira-Martins, Samurai Brito QC Ware Corp, IRIF - Université Paris Cité, CNRS, and Itaú Unibanco, arXiv, April 2024 (33 pages).
- [2982] [Optimal quantum reservoir computing for market forecasting: An application to fight food price crises](#), by L. Domingo, M. Grande, G. Carlo, F. Borondo, and J. Borondo, arXiv, November 2023 (10 pages).
- [2983] [QADQN: Quantum Attention Deep Q-Network for Financial Market Prediction](#), by Siddhant Dutta, Nouhaila Innan, Alberto Marchisio, Sadok Ben Yahia, and Muhammad Shafique, arXiv, August 2024 (6 pages).
- [2984] [Towards Quantum Advantage in Financial Market Risk using Quantum Gradient Algorithms](#), by Nikitas Stamatopoulos, Guglielmo Mazzola, Stefan Woerner, and William J. Zeng, arXiv, July 2022 (25 pages).
- [2985] [Applications of Quantum Annealing in Computational Finance](#), by D-Wave, 2016 (29 slides).
- [2986] [Applying QNLP to sentiment analysis in finance](#), by Jonas Stein, Ivo Christ, Nicolas Kraus, Maximilian Balthasar Mansky, Robert Müller, and Claudia Linnhoff-Popien, arXiv, September 2023 (6 pages).
- [2987] [Characterizing asymmetric and bimodal long-term financial return distributions through quantum walks](#), by Stijn De Backer, Luis E. C. Rocha, Jan Ryckebusch, and Koen Schoors, arXiv, May 2025 (24 pages).
- [2988] [Securities Transaction Settlement Optimization on superconducting quantum devices](#), by Francesco Martini, Luca Dellantonio, et al., arXiv, January 2025 (4 pages).
- [2989] [Quantum Advantage in Trading: A Game-Theoretic Approach](#), by Faisal Shah Khan, Norbert M. Linke, Anton Trong Than, and Dror Baron, arXiv, March 2025 (17 pages).
- [2990] [Ensemble-Based Quantum-Token Protocol Benchmarked on IBM Quantum Processors](#), by Lucas Tsunaki, Bernd Bauerhenne, Malwin Xibraku, Martin E. Garcia, Kilian Singer, and Boris Naydenov, arXiv, February 2025 (14 pages).
- [2991] [Quantum algorithm for large-scale market equilibrium computation](#), by Po-Wei Huang and Patrick Rebstroff, arXiv, February 2025 (8 pages).
- [2992] [Quantum Technology for Securing Financial Messaging](#), by QED-C, July 2024 (44 pages).
- [2993] [Quantum computing and the financial system: opportunities and risks](#), by Raphael Auer et al., October 2024 (33 pages).
- [2994] [A Quantum Vault Scheme for Digital Currency](#), by Anne Broadbent, Raza Ali Kazmi, and Cyrus Minwalla, arXiv, August 2024 (11 pages).
- [2995] [Experimental investigation of practical unforgeable quantum money](#), by Mathieu Bozzio, Adeline Orioux, Luis Trigo Vidarte, Isabelle Zaquine, Iordanis Kerenidis, and Eleni Diamanti, npj Quantum Information, January 2018.
- [2996] [Potential Applications of Quantum Computing for the Insurance Industry](#), by Michael Adam, arXiv, October 2022 (43 pages).

- [2997] [The impacts of quantum computing on insurance - From theory to reality](#), by Lloyds Lab, February 2021 (34 pages).
- [2998] [Quantum computing a potential cyber risk for re/insurers](#), by Charlie Wood, November 2019.
- [2999] [Top 5 insurance quantum computing use cases](#), by Danni Santana, January 2018.
- [3000] [Quantum Computing to Affect Insurer Tech Strategies](#), by Insurance Canada, December 2019.
- [3001] [Iterative quantum amplitude estimation](#), by Dmitry Grinko, Julien Gacon, Christa Zoufal, and Stefan Woerner, npj Quantum Information, March 2021.
- [3002] [Modern Computational Methods in Reinsurance Optimization: From Simulated Annealing to Quantum Branch & Bound](#), by George Woodman, Helmut G. Katzgraber, Marcin Detyniecki, et al., arXiv, April 2025 (23 pages).
- [3003] [A Quantum Algorithm for the Sensitivity Analysis of Business Risks](#), by M. C. Braun, T. Decker, N. Hegemann, S. F. Kerstan, and C. Schäfer, arXiv, March 2021 (21 pages).
- [3004] [CaixaBank Group, D-Wave Collaborate on Innovative New Quantum Applications for Finance Industry](#), by D-Wave, March 2022.
- [3005] [Quantum Computing in Insurance Capital Modelling under Reinsurance Contracts](#), by Muhsin Tamturk and Marco Carenzo, AppliedMath, October 2023 (17 pages).
- [3006] [A Quantum Approach to Stochastic Optimization in Insurance Underwriting](#), by Mitchell Bordelon, Jean Utke, et al., arXiv, June 2026 (40 pages).

This **Financial services and insurance** subsection contains 143 bibliographical references and notes containing at least 2,848 pages.

3.8 Other user segments

We cover here the use cases and case studies of smaller potential markets, at least, considering the number of available cases.

3.8.1 Industry

Industry in the broadest sense of the term is another vertical market for quantum computing. As soon as there is a complex optimization problem for scheduling, logistics or complex system design support, quantum may have its say.

Job Shop Problem. One classic optimization problem in manufacturing and operations research is the job shop problem (JSP) or job shop scheduling problem (JSSP). It is about scheduling a set of jobs on a set of machines, where each job consists of a sequence of tasks that need to be processed in a specific order. Each task must be performed on a specific machine, and no two tasks for the same job can be processed simultaneously. Several quantum algorithms are proposed to solve this problem which is NP-hard, with a lot of variations depending on the problem constraints. These are all heuristic-based and do not find the optimum solution. They operate on quantum annealers [3007] or gate-based QPUs and variational algorithms, all, with still uncertain theoretical speedups [3008–3010]. A 2026 paper from Siemens describes such absence of quantum advantage for both D-Wave and an IBM processor [3011]. All these techniques compete with still-improving classical algorithms using machine learning [3012] **NISQ**.

FTQC In 2024, a team in Germany and the Netherlands produced interesting resource estimates for solving an industrial shift scheduling problem in automotive manufacturing by using the Quantum Industrial Shift Scheduling algorithm. It is an optimization problem related to finding exact solutions to the optimal production schedule for an end-to-end manufacturing system with multiple production sites, complying with various constraints (legal regulations, intermediate storage between production sites, etc.). It is based on a Grover Adaptive Search [3013]. The team found out that a quantum speedup would require physical error rates below 10^{-6} and qubit readouts times below 10 ns, due to the known limitations of Grover based quantum algorithms, which brings only a quadratic speedup with significant prefactors making it difficult to bring any quantum advantage even for industry-scale problems. As a result, the best approach will be to use tensor-network-based classical heuristics and for a very long time (Figure 197). This may explain why FTQC use cases related to solving optimization and decision problems are just starting to appear [3014]. In October 2025, IonQ proposed an FTQC solution using some non-variational Iterative-QAOA [3015].

Cable Routing Optimization Problem (CROP) is a multi-flow routing task used to optimize industrial

layouts and smart manufacturing installations. A team in Germany and from Vinci proposed a QUBO-based quantum algorithm to solve this problem using VQE in 2025. Unfortunately, it was tested with only up to 16 qubits, below any quantum advantage threshold [3016].

Robot planning. A variation of the JSP (job shop problem) and the VRP (vehicle routing problem) is the robot planning problem, where hundreds of robots operate in a single manufacturing plant or logistic warehouse. The goal is to implement efficient load balancing between the robots, with optimal sequencing of individual tasks and robot movements, avoiding traffic jams, within the cycle time of the whole production line [3017, 3018].

One variant it about solving the Multi-vehicle Coverage Path Planning (CPP) problem. A proposal was made in 2024 to solve it using a QAOA-based approach, but tested only with 20 qubits, with no resource estimates to obtain any quantum advantage [3019] **NISQ**.

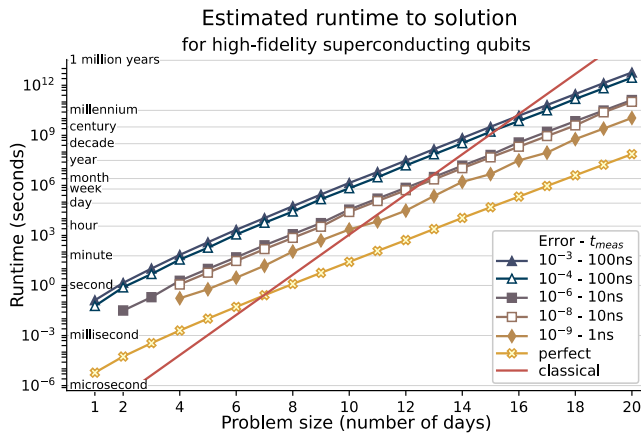
Milling simulation. An evaluation of a HHL-based algorithm was done in Germany in 2024 to compute a simulation of a multi-axis milling of thin-walled aerospace components [3020]. The non-surprising result is that it can't work on existing (IBM) NISQ hardware and will require FTQC QPUs **FTQC**.

Electronic circuit design. The routing of electronic circuits is also a NP-complete problem that could be partly handled by quantum algorithms, provided they have enough qubits. This could be useful for designers of ASIC-type circuits and especially FPGAs, these circuits whose operating logic is dynamically programmable via two key parameters: the decision tables of the processing units and the links between these units [3021]. It seems that quantum computing could also be used within computer-aided design tools [3022]. Classical circuit placement optimization problems could be expressed as a Min- k -Union problem and solved with a QUBO formulation on an IBM QPU and a D-Wave quantum annealer [3023]. It could bring some computational advantage in the future with better QPUs **FTQC**.

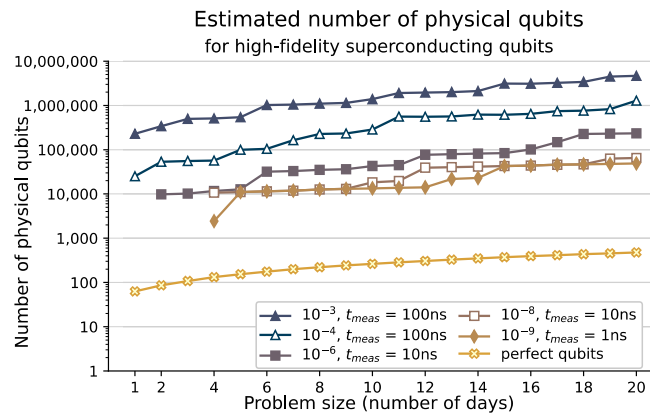
FTQC IBM designed in 2026 an FTQC solution to simulate RLC electronic circuits (resistors, inductors and capacitors). It could extract global values of interest from complex circuits in a much faster time than on a classical computer, like total stored energy or total dissipated power loss [3024].

Finite element method-based simulation. This technique has been the cornerstone of mechanical simulations for at least four decades. A Quantum Finite Element Method (Q-FEM) could improve the process. But it is currently tested at very small scale (7 qubits) and already exhibiting limitations like barren plateau effects [3025]. This category of application will certainly require FTQC QPUs **FTQC**.

Otherwise, various use cases have been tested, but not yet in a quantum advantage regime, since being emulated on a classical system with a small number of qubits



(a) Estimated runtime for arriving at the best solution.



(b) Estimated number of qubits for arriving at the best solution.

Figure 197: Resource estimation of a Grover-based industrial shift scheduling problem in automotive manufacturing could be implemented in the future based on various assumptions related to superconducting qubit quality (physical error rate and readout time). The left charts show that even with reasonable forecasts of 99.9% fidelities at scale and 100 ns readout time, a quantum solution would be more performant than a classical one only after several centuries of computing. You'd need (sci-fi) physical qubit fidelities of 99.9999% and readout times of 10 ns to reduce this threshold to months. Now (right), with physical qubit numbers, we are in the millions. And this doesn't account for the cost and fidelities of interconnect solutions which are mandatory given the limitations of physical qubits per chip. However, the number of required logical qubits is reasonable, sitting in the 300-600 range to obtain a quantum advantage regime. Source: [3013].

like for the prediction of surface roughness in 3D printed specimen [3026], other artificial vision operations and quality control [3027, 3028] and with some multi-agent neural network [3029]. All this is not far from classical quantum inspired methods that are also investigated [3030].

Predictive maintenance. It can be improved, particularly in the aerospace domain, to predict the remaining useful life of engines and other manufacturing sensitive components. It can for example use Quantum Long Short-Term Memory algorithms [3031] **FTQC**.

3.8.2 Civil engineering and construction

Civil engineering could potentially benefit from quantum computing, according to an arXiv review [3032] and a post in The Constructor [3033]. Kay Smarsly, an author of the review, also coauthored [3034] on quantum-inspired use cases in the domain.

The ideas? Balancing material deliveries, workforce schedules, and regulatory constraints in building construction. Create optimal construction sequences “*in seconds, compared to weeks of simulations using conventional computing*”. Reduce planning time for large-scale infrastructure projects by 60–80%, while improving resource utilization by 30–40% compared to traditional methods. The numbers are not sourced and look like fairy tales.

I found in 2025 a case study on building an energy model consumption predictive control (MPC) strategy using a QAOA algorithm [3035]. It optimizes the energy consumption for buildings powered with renewable energy production including photovoltaic panels and battery energy storage. In the case study, the quantum MPC

strategy delivered a 6.8% increase in energy efficiency over deterministic MPC and achieved a 41.2% annual reduction in carbon emissions by efficiently managing battery energy storage and renewable energy sources. The QAOA algorithm used a 127-qubit IBM QPU, seemingly in emulation mode rather than on hardware, and was compared with D-Wave and a PC using a single Intel CPU. The instance size, the number of qubits, the circuit depth, the shot count and the tuning of the classical baseline are not documented, and the compute budgets are not matched. The 6.8% and 41.2% figures can therefore not be attributed to the quantum part of the pipeline.

In 2024, researchers from the University of Bari Aldo Moro in Italy proposed a hybrid quantum-classical architecture for supporting Smart City security using a D-Wave annealer performing some Quantum Machine Learning (QML) tasks. It was used to improve the prediction of security incidents and intrusions. The paper described an improvement in training time of 80% and 71% in inferences with the QBoost method vs a classical Random Forest (RF) but with a slightly lower accuracy (97% vs 99%) [3036].

3.8.3 Retail

The retail sector could benefit from quantum computing generic use cases in transportation and logistics improvements as well as in some marketing optimization techniques. Robotized warehouse operations could also benefit from some quantum-driven optimizations like what **Ocado Group** (UK) is piloting with D-Wave [3037]. Always with D-Wave, **Pattison Food Group**

(Canada) is looking at ways to optimize grocery operations.

Beyond that, consultants and analysts have not much to say about it or they sometimes err by mentioning big-data applications, which are out of scope of quantum computers [3038].

PayPal is also testing a D-Wave quantum annealer to solve some optimization problems and detect fraud using quantum machine learning [3039].

3.8.4 Telecommunications

At this stage, the main quantum involvement of telecom companies is mainly related to quantum telecommunications and cryptography.

Quantum communications

In the USA, **Verizon** launched a QKD trial on three sites in the Washington, D.C., area [3040]. In 2021, they also tested a VPN using a PQC (post-quantum cryptography) solution based on the Saber NIST competition finalist, for connection between two sites in the US and UK. **AT&T** is also investing resources in QKD networks, teaming up with Caltech.

In France, **Orange** participated in a QKD trial in the Nice region with partners including the InPhyNi research lab.

In the UK, **British Telecom** has also tested a QKD setup to demonstrate some secured backup of datacenter resources. In October 2020, they deployed a pilot 6 km long QKD infrastructure in Bristol to connect several industry sites, in partnership with **Toshiba** as part of the **AQuaSec** (Agile Quantum Safe Communications) program, cofunded by UKRI [3041]. In October 2021, BT and Toshiba announced the deployment of a commercial QKD network in London, mixed with PQC for non photonic endpoints.

Quantum computing

Quantum computing can still play a role in solving various optimization problems in the telecom industry, with the following potential case studies:

- Configurations of paths and wavelengths on land-line fiber-optic networks and similar optimization problems for satellite communications [3042] and MIMO antennas optimization [3043–3045]. It could also be used to enhance wireless-network energy efficiency with index modulation, a task that is an NP-hard problem [3046]. It uses a Grover Adaptive Search (GAS) providing a quadratic speedup (not an exponential one) and still requires fault-tolerant large-scale quantum computers that do not exist yet **FTQC**.
- Optimizing resource allocation of wide-area internet networks using a QUBO variant on a D-Wave Advan-

tage, completed by some system parameters using classical machine learning [3047] **analog**.

- Optimizing the tracking of mobile devices across base stations, in a 2024 pilot project from DoCoMo pilot in Japan, using a D-Wave's hybrid-solver. It led to time savings from 27 hours to 40 seconds compared to classical methods, which were, however, not clearly documented [3048] **analog**.
- Predicting text message traffic with qLSTM algorithms (although, with no more than 14 qubits...) [3049] **NISQ**.
- Optimize fiber backbone networks, as tested by **Cinfo**, **Kipu Quantum** and **R** in 2024 [3050]. It used an initial classification of the telecom operator network topology using 180 qubits of a D-Wave Advantage, for network segmentation, followed by an examination of sublattices using a **QuEra** analog simulator using 20 and 46 qubits. A gate-based QAOA-based test was made by Nvidia et al. in 2024, but only tested on a classical 32 qubit emulator with no resource estimate to obtain some quantum advantage [3051] **analog**.
- Placement, power and frequency assignment of overlapping cells in 4G/5G mobile networks [3052].
- Optimizing the setup of 4G/5G radio cells frequencies as tested in Italy with the telecom operator **TIM** using a D-Wave QUBO based algorithm **analog**.
- Solving the Physical Cell Identifier (PCI) assignment problem in cellular networks, as tested by **Pasqal** on its analog quantum simulator with the **LINKS foundation** (Italy) [3053] **analog**.
- Quantum human activity monitoring the home automation space using Wi-Fi in the ultra-wideband range of 60 GHz and a variational quantum circuit (VQC) building a QNN (quantum neural network), tested by researchers from **Mitsubishi** [3054] **NISQ**.
- Finally, quantum computing will probably be relevant in many use cases related to the deployment and operation of existing wireless and upcoming faster 6G mobile networks like with spectrum management optimizations, routing optimizations and the like [3055–3058] (Figure 198).

3.8.5 Content and media

Are you wondering how quantum computing could be used to create some content and art? That's the weird idea some have, at least with regard to music creation. Computers have played a role in music creation and art in general for a while so why not quantum computers? Quantum mechanics is about waves, like music [3059]!

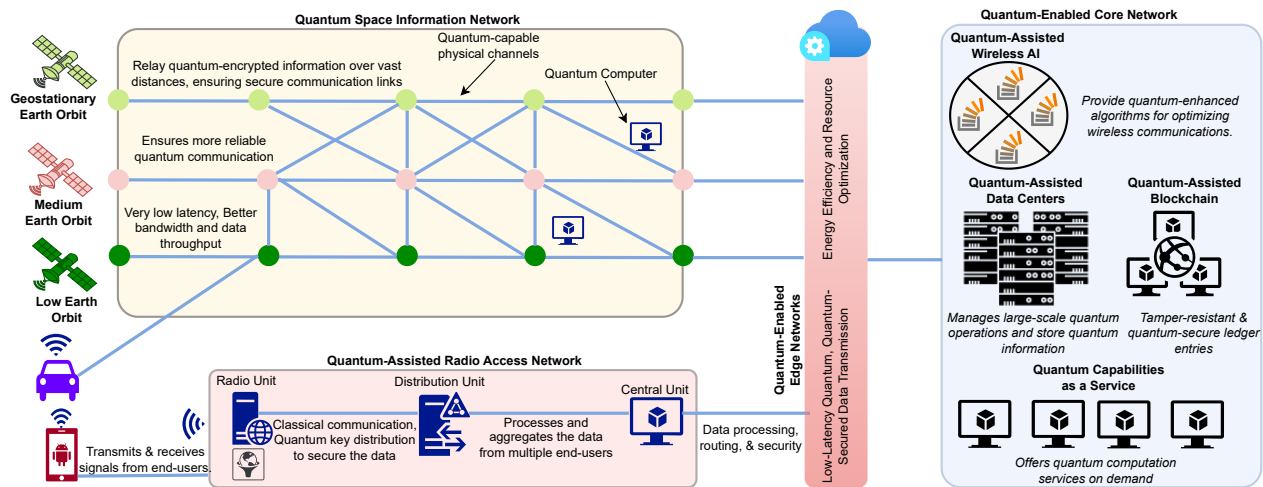


Figure 198: The various places where quantum computing might be useful in the deployment of 6G mobile networks. Source: [3055].

Quantum music

At this point, quantum music is mostly about finding other sources of randomness to create melodies (using quantum walks-based algorithms) and to generate credible synthetic voices. But linear algebra is also used, as in this 2026 paper using a variant of HHL using a coherent Fourier harmonic oracle to generate melodies [3060].

The **Quantum Music** project [3061], was run by Volkmar Putz and Karl Svozil in Austria from 2015 to 2018, laying some foundations on quantum music.

Eduardo Miranda from the Interdisciplinary Centre for Computer Music Research (ICCMR) at the University of Plymouth (UK) works on using quantum computers to create music [3062–3064], particularly as part of the **QuTune** project which he led with Bob Coecke. He authored the “Quantum Computer Music” book published in 2022 [3065], several of its chapters having previously been published on arXiv [3066–3070]. He also coordinated the publication of the collective book “Advances in Quantum Computer Music” in 2025 [3071] and released the “Qubism” music album in 2024, all its 8 tracks being composed using quantum computers [3072]. The University of Plymouth runs a PhD program “PhD Quantum Computing in the Arts, Music and Humanities” [3073].

A quantum music online event, was organized in November 2021 by the University of Plymouth (UK) with the sponsoring from IBM and Cambridge Quantum Computing (now Quantinuum) [3074]. The last one was organized in Palermo in October 2025.

In 2019 **Quantum Sound** was the first music created and performed from measurements of superconducting qubits. It was not far from some form of random music generator. The project was run with the financial support of the Yale University Quantum Institute as well as from their top quantum scientists Michel Devoret and

Robert Schoelkopf [3075]. Another similar project was run in 2024 by quantum scientists in Austria with the “*The Sound of Entanglement*” live musical performance. It used real-time measurements of entangled photons in a Bell test to create the BruQner soundtrack, named after Anton Bruckner. Was its randomness generating nice melodies, or “Salieri” quantum music salad [3076]? Good music has usually some structure. I’m wondering whether quantum randomness can create such structure. Even machine learning tools generating artificial music and reusing existing music structure has had a tough time so far to generate emotionally impactful music.

At last, quantum music was generated in a more sophisticated way with a quantum annealer from D-Wave in 2021 by a very international team (India, Poland, Mexico, Estonia) [3077]. In 2022, voice synthesis was studied using quantum algorithms [3078].

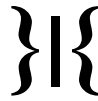
Omar Costa Hamido built a quantum music synthesizer based on the creation of Bell states and simple Grover search [3079], a proposal to use quantum Internet to protocols in the context of network performances [3080], a theory on music inspired by quantum physics [3081] after his thesis work on quantum art [3082]. He later updated his work to improve timbre, harmony and spatialization of the generated music [3083].

Bob Coecke proposed in 2025 the concept of a quantum guitar which connects each string to a qubit, given both are a wave with the ability for the guitarist to transition from classical cords to quantum virtual cords, all being connected to Moth’s emulation software [3084]. So we have here some quantum-inspired quantum guitar and no quantum computer in sight. Now I want something that exceeds Jimi Hendricks and Jimmy Page’s past performances! Bob Coecke is also behind QBE (Quantum Brain Art), a research organization based in Oxford, UK, that works along those lines in quantum art.

One can now wonder whether the beauty of music can work without nonlinearities!

Moth Quantum

UK, 2022, \$3.79M



Moth Quantum is developing quantum computing technology for music, art, and gaming. Moth initially created the Moth synth out of the work from Bob Coecke’s work at ICCMR’s Miranda lab from the University of Plymouth [3085].

In 2024, the company hired James Wooton from IBM as its Chief Scientist. In 2025, he reused a quantum reservoir network designed for the creation of musical scores to create variants of the Super Mario Bros game [3086]. They showcase their Archeo platform that combines generative AI and quantum computing. In 2025, it was used to create some music track, titled RECURSE, by the electronic artist ILĀ. It is using a quantum computing reservoir AI system was originally developed at ICCMR by Miranda’s team [3087].

Moth also developed Quantum Brush, an open-source digital painting tool running on a 20-qubit IQM NISQ QPU with an undetermined number of actual qubits [3088]. The idea was extended with a variational circuit version by Inria in 2026 [3089].

You are left with the need to create a benchmarking method to differentiate brushes created with such tools and more classical brushes. That requires inventing a new definition for a quantum advantage with is purely aesthetics-based.

They believe that quantum computing could help create games by enhancing procedural content generation (PCG), a technique used to algorithmically create game elements like levels, maps, and narratives [3090]. It could enable more complex, varied, and dynamic content to be generated efficiently. It could compute the Jones polynomial to design intricate puzzles or game mechanics that are infeasible with traditional computing. It would still require FTQC. Hopefully, this is only related to game design and not running them. With Bob Coecke, Moth proposed the creation of “Quantum Concept Music” in 2025, a new language and theory for quantum music which is based on ZX-calculus. It uses the Bell-pair analogy to describe correlations in music [3091].

Quantum visual art

Quantum computing has inspired visual artists for the creation of modern art.

Libby Heaney is a quantum physicist from Bristol University turned into a quantum artist in 2019, who also works with IBM quantum computers [3092].

In 2022, a “visual art” was created with the help of a quantum computer. [The Quantum Prophet](#) was cre-

ated with the quantum artists trio [Insight](#) and **Kipu Quantum** (Figure 199).



Figure 199: The Quantum Prophet, created by the trio [Insight](#) in 2022.

With the support of some AI, the author interfaced himself in real time with a quantum computer. Via motion capture, he manipulated qubits to modify aesthetically his animated 3D artwork which is sold with an NFT (non fungible token, a derivative from the blockchain). All-in-all, the creative process was still largely in the hands of some humans!

Joel Russel Huffman, a product designer from the IBM quantum team, is working on quantum fractals done with quantum computing [3093].

Pindar Van Arman creates “quantum portraits” with an unspecified mix of AI and quantum computing codes running on IBM QPUs and which are seemingly sold using crypto currencies [3094].

Refkik Anadol is a visual artist using quantum computer originated randomness. He sells his art with NFTs.

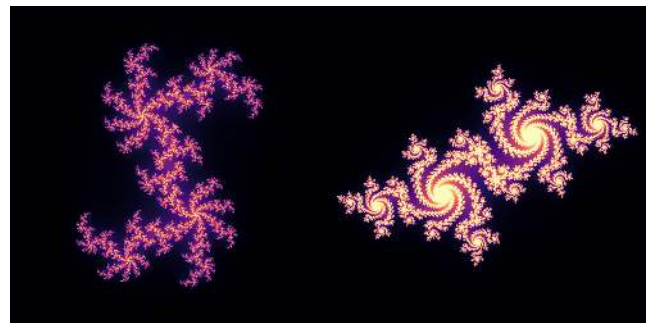


Figure 200: Wiktor Mazin fractal example. Source: [3095].

Wiktor Mazin used IBM Qiskit to create fractal art in 2022 using only one and two qubits (Figure 200). I didn’t get if he really needed a QPU and not just an emulator to run his code.

Roman Lipski designs “Quantum Blur” art, with the help of an IBM Zurich team with Stephan Schneider,

James Wootton, Marcel Pfaffhauser, Mira Wolf-Bauwens and Walter Riess [3096]. The quantum algorithms used seem to add some quantum noise to images.

Serge Aa (horomox) is a digital artist using artificial intelligence, QRNG, Qiskit and even post-quantum cryptography to design his graphical creations around the ecology of consciousness theme [3097, 3098] (Figure 201).



Figure 201: Horomox QRNG based graphical art. Quantum Air NamKhas – Machine Whisperers. 2025. Source: [3099].

Nancy Kawalek runs an innovative art and science lab in the Pritzker School of Molecular Engineering at the University of Chicago. Her work is quantum inspired, in the arts meaning [3100], in the fields of card games [3101], movies and theater plays creation.

Arianna Crippa et al. combined QPU-generated digital images with the artistic practice of oil painting in 2025 [3102]. The goal was to revisit Caravaggio, Magritte and Richter paintings. It used a quantum reordering of the lattice tiles leading to quantum transformed images of the original paintings. It is not sure that the end result benefited from the process from the artistical standpoint compared to some modifications that you could implement with various Photoshop tweaks.

These quantum artists do not necessarily document the technical part of their work like the above IBM team. Like many quantum algorithms, their work is hybrid in nature, mixing classical and quantum tools (quantum computers, QRNG), on top of their own creativity. Others view quantum concepts as aesthetic phenomena that can be used in artistic mediation [3103]. One can wonder how structure can be generated by linear algebra and QRNG. Art indeed is often driven by structure in space in time.

Art explaining quantum

Another form of quantum art without pretensions is to exhibit art to explain what quantum physics and technologies are about, like was done in Switzerland in 2020 [3104]. In a word, quantum computing being used for its own science intermediation related tasks [3105].

Quantum games

Another content-related field being explored is gaming. Proof of concepts of quantum computing based games were proposed in 2023 for Mastermind [3106] and Go [3107], and of games using quantum principles [3108]. There's also some work on game theory going on in Columbia and Singapore [3109].

Text

Quantum computing can also be used to analyze and compare texts. This was proposed in 2026 in the QOuLiPo process by Christophe Jurczak, and with tests on papers from the Renaissance running on a Pasqal analog simulator. It proposes to compute a “rigidity rho”, a metric describing how unique is a book's structure [3110] **analog**.

Media

[3111] inventoried in 2023 some potential use cases which were seemingly based on poor knowledge of quantum computing: “*real-time video editing and rendering*” (quantum computing is not real-time), “*generate more realistic visual effects than classical computers*” (maybe, but not necessarily in an affordable manner), “*enable new forms of interactive media that are impossible with classical computers. This could include virtual reality, augmented reality, and haptic feedback*” given there is no pressing need for a quantum computer to do that.

Only one case is using some quantum approach to detect fake news using the paradigm of quantum entanglement [3112].

3.8.6 Marketing

Marketing is also an area where optimization algorithms for complex systems based on quantum computers could be of interest. This concerns the optimization of the marketing mix, that of media plans, or the maximization of advertising revenues, various areas that are also invested by the AI field.

analog Volkswagen experimented with a vehicle recommendation system in online sales sites, with a D-Wave.

Predictive systems based on the exploitation of past data and simulation based on the knowledge of market operating rules are once again opposed to each other. However, these rules do not fall under the notion of AI expert systems, which manage logical predicates, but more complex causality models [761].

Marketing and opinion analysis can also implement quantum inspired algorithms [3113, 3114] **QI**.

Quantum computing could also be used to simulate opinion dynamics and was experimented at small scale by RIKEN in Japan in 2026 [3115].

3.8.7 Defense and aerospace

The military-industrial complex has always been a big consumer of advanced IT. It is therefore not surprising that it is interested in quantum technologies. This is obviously the case in the USA but also in Europe, with Airbus being one of the first to take an interest in quantum applications, and also China and Russia to name a few others [3116, 3117].

The **US Air Force** has also identified various needs that can be covered by the four categories of quantum technologies with a special mention for quantum sensing in time measurement and navigation [3118]. They are also interested in quantum radars and, finally, in quantum computing applied to optimization problems (Figure 202).

In France, **DGA** funded or co-funded since 2011 about twenty theses on quantum and eight projects for €6.6M. The **Defense Innovation Agency** (reporting to the DGA) planned to launch a call for projects for quantum sensors in 2020 and in 2021 to fund a research project to support PQC in dedicated hardware. In July 2020, it became an ASTRID RFP on sensors, cryptography and quantum communications and on the creation of quantum computing algorithms [3119].

NATO launched its DIANA Quantum Center in Copenhagen, Denmark, in 2024 [3120]. It is an accelerator for startups creating dual-use quantum technologies. It is hosted at the Niels Bohr Institute (NBI) [3121]. It entertains multiple working groups on quantum technologies.

The use of quantum technologies in the military field also gives rise to elucubrations that hybridize the plausible and the offbeat, such as those of the American political scientist **James Der Derian**, director of Project Q at the University of Sydney [3122].

You also have weird plans like the UK military which seems interested to put quantum computers in tanks, which doesn't make much sense, at least for the next 20 years. The reason is they've been lured in this path by Orca which touts ambient temperature quantum computing [3123].

Here are some various published case studies or use cases of quantum use in this vast sector.

Quantum communications

Quantum communications are studied by the military around the world to secure communications [3124], particularly on the battlefield. **NATO** is experimenting it [3125]. Of course, **China** is working on it although with the visible part being only the civil use cases.

Quantum sensing

The **Adequade** project funded by the European Defence Fund since 2022 with €27M over 3 years and with 31 participating organizations focuses on various quan-

tum sensing technology for the defense industry: PNT (position-navigation-timing), RF sensing, and quantum optronics sensing. It is coordinated by Thales. In a different field, navies are interested in quantum sensing and particularly in micro-gravimetry measurement tools used to detect submarines.

Formal proofs

analog **Lockheed Martin** partnered with **Google** and **NASA** to test D-Wave annealers starting in 2014. They developed with it a solution for formal proof of software operation. **NASA** co-founded the Quantum Artificial Intelligence Laboratory (QuAIL) with Google, operating a D-Wave Two. They test quantum optimization algorithms in different directions to optimize spacecraft filling, a variant of the bin-packing problem, on quantum versions of machine learning and deep learning algorithms, on problem decomposition and embedded computing [3126]. **NASA** published a review paper in June 2024 on their multi-year exploration of quantum computing in both quantum annealing and NISQ gate-based computing. It concluded that there is still a long way to go [3127].

NISQ In 2015, **Raytheon** and **IBM** demonstrated the efficiency of a quantum algorithm using a “black box” or “oracle” to reconstruct an unknown bit string, all running on an IBM five-qubit general-purpose quantum computer [3128]. This is obviously far from a real-world use case.

Image analysis

Airbus group created a team based at their Newport site in Wales, which is tackling the uses of quantum computing, particularly in the analysis of aerial images (not obvious...) or for the design of new materials (more obvious). They also want to optimize wings' air flow, a problem that is nowadays dealt with by finite element simulation. They could try to optimize the air conditioning in airplanes, the biggest source of cabin noise, above the plane's engines!

analog In collaboration with **Fraunhofer IAIS**, Thales also prototyped a quantum image alignment solution for satellite images with quantum-based key point extraction and feature matching. It was tested with D-Wave quantum annealers and gate-based quantum computers. Another project with CNES and CS Group dealt with hyper-spectral image classification also on a quantum annealer [3129]. The outcome is always the same: classical systems still deliver superior results, but the proposed methods have the potential to outperform classical systems with future quantum computers [600].

NISQ **Kipu Quantum** et al. presented a technique to improve satellite images classification in 2026, claiming a quantum advantage against a regular ResNet-50. But the comparison was uneven for many respects: artificially limiting the classical feature extractor to match qubit limits, feeding the quantum pipeline with rich

Quantum Technologies and AF Needs

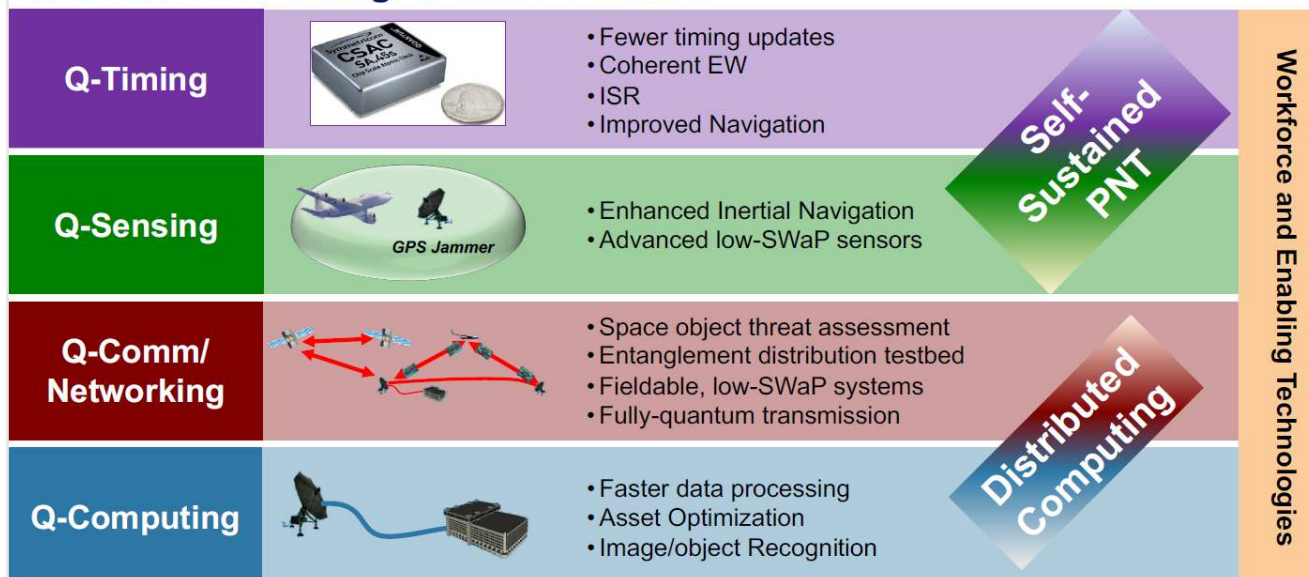


Figure 202: Quantum technology and US Air Force needs. Source: [Quantum Information Science at AFRL](#) by Michael Hayduk, December 2019 (21 slides).

multi-body correlation features that were not present from the classical baseline training data, and evaluating the performance gains on a small test set of only 200 images where a 2.5% improvement represents just the effect of five items [2330].

Radar operations

analog Thales prototyped in 2021 a quantum annealing solution running on D-Wave Advantage (5,000 qubits) to optimize radar configuration (the “Integrated Side Lobe Ratio” or ISLR NP-hard problem, which solution is “*the finding optimal sequences of phase shifts to minimize the mean squared cross-correlation side-lobes of a transmitted radar signal and a mismatched replica*”) [3130, 3131]. Yet, it didn’t reach any quantum computing advantage, but the authors think future evolutions of D-Wave annealers may be promising for this respect. Thales also developed a drone mission planning algorithm in 2024 [3132].

Satellite mission planning

Satellite mission planning is another interesting use case, that can be addressed with variational quantum algorithms [3133].

NISQ Thales Alenia Space with Terra Quantum applied hybrid quantum computing to the optimization of earth observation satellite missions [3134]. The problem consists of maximizing the number of high-priority tasks to be realized by multiple satellites. They used a hybridized quantum-enhanced reinforcement learning agent that can achieve a completion percentage of 98.5% over high-priority tasks, an improvement over the classical baseline greedy methods having a completion rate of 63.6%. Where was this tested? On a QMWare classi-

cal emulator from Terra Quantum, thus not showing a quantum advantage.

analog Similar problems were solved at a small scale on D-Wave annealers [3135]. In Germany, a quantum annealing solution optimized the clustering of low-earth orbit (LEO) satellites to optimize Internet communications in 2024, with a quantum advantage, compared with a classical Gurobi-based solution [2350]. A 2026 US study on solving the maximum coverage problem for satellite constellations was implemented in QAOA with an advertised computing time advantage vs Gurobi [2349]. But, as for the former case, it compared a quantum approximate (heuristic) solver and an exact solution solver, Gurobi. And this was using only 12 qubits on an IBM quantum processor.

analog In 2024, Thales announced it was working on solving a similar satellite mission problem using a **Pasqal** analog quantum simulator [3136]. It led to a preprint published in June 2026 [3137] which didn’t exhibit a computing time advantage due to the 1 Hz duty cycle of current simulations on Pasqal analog systems. With future increased sampling frequency, things could change, although not asymptotically.

NISQ U.S. Naval Research Laboratory published a preprint in 2026 on data flow scheduling for multi-beam satellites which is an NP-hard combinatorial optimization problem [3138]. They developed a hybrid quantum-classical solution that improves scheduling efficiency with a QUBO reformulation, then turned into a variational QAOA solver circuit. The test was working on a small toy model running on an IBM processor, and did not demonstrate quantum advantage since it did not scale to enough expressive ansatzes due to barren plateau effects and gate fidelity limitations.

3.8.8 Intelligence services

The world of intelligence and targeted eavesdropping is obviously interested in the potential usage of quantum technologies. Shor’s algorithm is the main application targeted by organizations managing electronic eavesdropping such as the **NSA** and all its colleagues.

They are firefighters who are eager to decode information intercepted from various targets (embassy communications, economic intelligence, etc.) and to protect the sensitive communications of their own states against this type of decryption. They are therefore investing simultaneously in quantum computing (the “arsonist” dimension) and in quantum keys and post-quantum cryptography (the “firefighter” dimension).

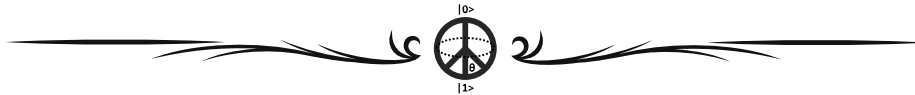
On the other hand, these investments are not very public. The NSA has communicated well for almost ten years on the firefighter dimension but very little on the arson dimension.

They have surely acquired the various generations of D-Wave computers to get their hands on, in conjunction

with **Lockheed Martin** which is one of their major suppliers. NSA also maintains a joint laboratory with NIST and the University of Maryland, **QuICS**, which will be launched in 2014.

One way to lift a veil on these activities is to detect laboratory and startup grants awarded by **IARPA**, the intelligence innovation agency led by the DNI (Director of National Intelligence), who oversees all American intelligence. It consolidates collaborative research funding for all intelligence agencies. It has already launched five programs around quantum technologies: in superconducting qubits (CSQ), logical qubits (LogiQ, with IBM), error correction (MQCO, also with IBM), the creation of development tools (QCS, with Raytheon and GeorgiaTech) and quantum annealing computation (QEO). But it is not clear that this has significantly advanced the state of the art.

Other Western intelligence services may also have acquired D-Wave, notably the British GCHQ. The NSA is also in contact with IBM and Google to explore the path of superconducting quantum general-purpose computers.



3.8.9 Bibliography and notes

These are the bibliographical references and notes for the **Other user segments** subsection.

- [600] [Towards Bundle Adjustment for Satellite Imaging via Quantum Machine Learning](#), by Nico Piatkowski, Thore Gerlach, Romain Hugues, Rafet Sifa, Christian Bauckhage, and Frederic Barbaresco, arXiv, April 2022 (8 pages).
- [761] [A quantum-inspired classical algorithm for recommendation systems](#), by Ewin Tang, arXiv, May 2019 (32 pages).
- [2330] [Quantum-enhanced satellite image classification](#), by Qi Zhang, Anton Simen, Carlos Flores-Garrigós, Gabriel Alvarado Barrios, Paolo A. Erdman, Enrique Solano, Aaron C. Kemp, Vincent Beltrani, Vedangi Pathak, and Hamed Mohammadbagherpoor, arXiv, February 2026 (7 pages).
- [2349] [Decomposition-Based QAOA for Maximum Coverage Location Problem in Satellite Constellation Design](#), by Divya Sisodiya, Amiratabak Bahengam, Hang Woon Lee, and Hao Chen, arXiv, July 2026 (21 pages).
- [2350] [Quantum Annealing-Based Algorithm for Efficient Coalition Formation Among LEO Satellites](#), by Supreeth Mysore Venkatesh, Antonio Macaluso, Marlon Nuske, Matthias Klusch, and Andreas Dengel, arXiv, August 2024 (6 pages).
- [3007] [Solving Distributed Flexible Job Shop Scheduling Problems in the Wool Textile Industry with Quantum Annealing](#), by Lilia Toma, Markus Zajac, and Uta Störl, arXiv, March 2024 (19 pages).
- [3008] [A case study of variational quantum algorithms for a job shop scheduling problem](#), by David Amaro, Matthias Rosenkranz, Nathan Fitzpatrick, Koji Hirano, and Mattia Fiorentini, arXiv, February 2022 (16 pages).
- [3009] [Differentiable Quantum Architecture Search For Job Shop Scheduling Problem](#), by Yize Sun, Jiarui Liu, Yunpu Ma, and Volker Tresp, arXiv, January 2024 (1 pages).
- [3010] [Highly Efficient Encoding for Job-Shop Scheduling Problems and its Application on Quantum Computers](#), by Math-

ias Schmid, Sarah Braun, Rudolf Sollacher, and Michael J. Hartmann, arXiv, January 2024 (17 pages).

- [3011] [A Reality Check on Quantum Optimisation: Evidence from an Industrial Case Study](#), by Hila Safi, Karen Wintersperger, Oliver von Sicard, Christoph Niedermeier, and Wolfgang Mauere, arXiv, July 2026 (12 pages).
- [3012] [Attention-based Reinforcement Learning for Combinatorial Optimization: Application to Job Shop Scheduling Problem](#), by Jaejin Lee, Seho Kee, Mani Janakiram, and George Runger, arXiv, March 2024 (19 pages).
- [3013] [Assessing the Requirements for Industry Relevant Quantum Computation](#), by Anna M. Krol, Marvin Erdmann, Ewan Munro, Andre Luckow, and Zaid Al-Ars, arXiv, August 2024 (19 pages).
- [3014] [Online Job Scheduler for Fault-tolerant Quantum Multi-programming](#), by Shin Nishio et al., arXiv, May 2025 (12 pages).
- [3015] [A Non-Variational Quantum Approach to the Job Shop Scheduling Problem](#), by Miguel Angel Lopez-Ruiz, Martin Roetteler, et al., arXiv, October 2025 (20 pages).
- [3016] [From Cables to Qubits: A Decomposed Variational Quantum Optimization Pipeline](#), by Paul-Niklas Ken Kandora, Steffen Rebennack, et al., arXiv, October 2025 (16 pages).
- [3017] [Optimization of Robot Trajectory Planning with Nature-Inspired and Hybrid Quantum Algorithms](#), by Martin J. A. Schuetz, J. Kyle Brubaker, Henry Montagu, Yannick van Dijk, Johannes Klepsch, Philipp Ross, Andre Luckow, Mauricio G. C. Resende, and Helmut G. Katzgraber, arXiv, June 2022 (17 pages).
- [3018] [Application of quantum annealing for scalable robotic assembly line optimization: a case study](#), by Moritz Willmann, Marcel Albus, Jan Schnabel, and Marco Roth, arXiv, December 2024 (7 pages).
- [3019] [A Quantum Computing Approach for Multi-robot Coverage Path Planning](#), by Poojith U Rao, Florian Speelman, Balwinder Sodhi, and Sachin Kinge, arXiv, July 2024 (8 pages).

- [3020] [A methodical approach to evaluate the potential of Quantum Computing for Manufacturing Simulations](#), by Stefan Schröder, João Felipe, Sven Danz, Pascal Kienast, Alessandro Ciani, Philipp Ganser, and Thomas Bergs, arXiv, August 2024 (8 pages).
- [3021] [FPGA-Placement via Quantum Annealing](#), by Thore Gerlach, Stefan Knipp, David Biesner, Stelios Emmanouilidis, Klaus Hauber, and Nico Piatkowski, arXiv, December 2023 (12 pages).
- [3022] [Computer-Aided Design for Quantum Computation](#), by Robert Wille et al., 2018 (6 pages).
- [3023] [QCEDA: Using Quantum Computers for EDA](#), by Matthias Jung, Sven O. Krumke, Christof Schroth, Elisabeth Lobe, and Wolfgang Mauerer, arXiv, March 2024 (7 pages).
- [3024] [Simulating dynamics of RLC circuits with a quantum differential-algebraic equations solver](#), by Arkopal Dutt, Anirban Chowdhury, Kristan Temme, and Hari Krovi, arXiv, April 2026 (103 pages).
- [3025] [An Implementation of the Finite Element Method in Hybrid Classical/Quantum Computers](#), by Abhishek Arora, Benjamin M. Ward, and Caglar Oskay, arXiv, March 2025 (46 pages).
- [3026] [Quantum Machine Learning Approach for the Prediction of Surface Roughness in Additive Manufactured Specimens](#), by Akshansh Mishra and Vijaykumar S. Jatti, arXiv, April 2023 (18 pages).
- [3027] [Quantum artificial vision for defect detection in manufacturing](#), by Daniel Guijo, Roman Orus, et al., arXiv, January 2024 (11 pages).
- [3028] [Boosting Defect Detection in Manufacturing using Tensor Convolutional Neural Networks](#), by Pablo Martin-Ramiro, Unai Sainz de la Maza, Sukhbinder Singh, Roman Orus, and Samuel Mugel, arXiv, April 2024 (12 pages).
- [3029] [Quantum Multi-Agent Actor-Critic Neural Networks for Internet-Connected Multi-Robot Coordination in Smart Factory Management](#), by Won Joon Yun, Jae Pyoung Kim, Soyi Jung, Jae-Hyun Kim, and Joongheon Kim, arXiv, January 2023 (11 pages).
- [3030] [Quantum Inspired Optimization for Industrial Scale Problems](#), by William P. Banner, Shima Bab Hadiashar, Grzegorz Mazur, Tim Menke, Marcin Ziolkowski, Ken Kennedy, Jhonathan Romero, Yudong Cao, Jeffrey A. Grover, and William D. Oliver, arXiv, May 2023 (10 pages).
- [3031] [Hybrid Quantum Recurrent Neural Network For Remaining Useful Life Prediction](#), by Olga Tsurkan, Leonid Fedichkin, et al., arXiv, April 2025 (11 pages).
- [3032] [Quantum computing in civil engineering: Potentials and Limitations](#), by Joern Ploennigs, Markus Berger, Martin Mevissen, and Kay Smarsly, arXiv, March 2024 (10 pages).
- [3033] [Quantum Computing in Construction: Revolutionizing Complex Planning](#), by The Constructor, The Constructor, March 2025.
- [3034] [Quantum-inspired computing in civil engineering – Opportunities, challenges, and visions](#), by Kay Smarsly and Martin Kliesch, August 2024 (8 pages).
- [3035] [Decarbonization of Building Operations with Adaptive Quantum Computing-Based Model Predictive Control](#), by Akshay Ajagekar et al., Engineering, February 2025 (24 pages).
- [3036] [Hybrid quantum architecture for smart city security](#), by Vita Santa Barletta et al., Journal of Systems and Software, November 2024 (15 pages).
- [3037] [Towards Real Time Multi-robot Routing using Quantum Computing Technologies](#), by James Clark et al., 2019 (10 pages).
- [3038] [Quantum computing in the consumer and retail sectors](#), by Linda Ellett et al., July 2021.
- [3039] [PayPal: Harnessing Quantum Computing in FinTech | D-Wave Qubits 2021](#), December 2021.
- [3040] [This Executive Director Is Leading Verizon Into the Future Through Quantum Computing](#), by Joanna Goodrich, IEEE Spectrum, November 2020.
- [3041] [BT and Toshiba install UK's first quantum-secure industrial network between key UK smart production facilities](#), by British Telecom, October 2020.
- [3042] [Heterogeneous Quantum Computing for Satellite Constellation Optimization: Solving the Weighted K-Clique Problem](#), by Gideon Bass, Casey Tomlin, Vaibhaw Kumar, Pete Rihaczek, and Joseph Dulny III, arXiv, March 2018 (16 pages).
- [3043] [Evaluation of quantum annealer performance via the massive mimo problem](#), by Zsolt I. Tabi, August 2021 (14 pages).
- [3044] [Quantum Approximate Optimization Algorithm for Maximum Likelihood Detection in Massive MIMO](#), by Yuxiang Liu, Zaichen Zhang, et al., arXiv, October 2025 (6 pages).
- [3045] [QMIMO: Circuit Based Quantum MIMO Design with Variational Receiver](#), by Sayeda Bipanchi Ahmed, Bikash K. Behera, Mandar Thatte, and Prasanta K. Panigrahi, arXiv, August 2026 (37 pages).
- [3046] [Quantum Speedup for Index Modulation](#), by Naoki Ishikawa, IEEE, July 2021 (11 pages).
- [3047] [ILP-based Resource Optimization Realized by Quantum Annealing for Optical Wide-area Communication Networks – A Framework for Solving Combinatorial Problems of a Real-world Application by Quantum Annealing](#), by Arthur Witt, Jangho Kim, Christopher Körber, and Thomas Luu, arXiv, May 2024 (22 pages).
- [3048] [Developing Quantum Computing Infrastructure for Service Optimization](#), by D-Wave et al., July 2024 (7 pages).
- [3049] [Benchmarking Quantum and Classical Sequential Models for Urban Telecommunication Forecasting](#), by Chi-Sheng Chen, Samuel Yen-Chi Chen, and Yun-Cheng Tsai, arXiv, August 2025 (6 pages).
- [3050] [R, Cinfo, and Kipu Quantum design a quantum algorithm for analyzing and optimizing telecommunication networks - cinfo](#), by Juan BarcÃ³n, January 2024.
- [3051] [Novel Application of Quantum Computing for Routing and Spectrum Assignment in Flexi-Grid Optical Networks](#), by Oumayma Bouchmal et al., October 2024 (16 pages).
- [3052] [High-Speed Resource Allocation Algorithm Using a Coherent Ising Machine for NOMA Systems](#), by Teppei Otsuka, Aohan Li, Hiroki Takesue, Kensuke Inaba, Kazuyuki Aihara, and Mikio Hasegawa, arXiv, December 2022 (16 pages).
- [3053] [Noise Resilient Quantum Computing method for Cellular Networks](#), by Pasqal, February 2024.
- [3054] [Quantum Transfer Learning for Wi-Fi Sensing](#), by Toshiaki Koike-Akino, Pu Wang, and Ye Wang, arXiv, May 2022 (6 pages).
- [3055] [Quantum Technologies for Beyond 5G and 6G Networks: Applications, Opportunities, and Challenges](#), by Engin Zeydan, Madhusanka Liyanage, et al., arXiv, April 2025 (30 pages).
- [3056] [Quantum Machine Learning-based 6G edge Network: Enabling Adaptive Communication and Model Aggregation](#), by Wenjing Xiao, Jiatai Yan, Chenglong Shi, Shixin Chen, Miaojiang Chen, Min Chen, Saif Al-Kuwari, and Ahmed Farouk, arXiv, May 2026 (9 pages).
- [3057] [Hybrid Classical-Quantum Optimization of Wireless Routing Using QAOA and Quantum Walks](#), by Eric Howard, Hardique Dasore, Hom Nath Dhungana, Radhika Kuttala, Samuel Murphy, Emma Soo, and Shah Haque, arXiv, April 2026 (23 pages).
- [3058] [Recursive QAOA for Interference-Aware Resource Allocation in Wireless Networks](#), by Kuan-Cheng Chen, Hiromichi Matsuyama, Wei-hao Huang, and Yu Yamashiro, arXiv, February 2026 (8 pages).

- [3059] [Quantum music Physics has long looked to harmony to explain the beauty of the Universe. But what if dissonance yields better insights?](#), by Katie McCormick, May 2021.
- [3060] [HHL with a Coherent Fourier Oracle: A Proof-of-Concept Quantum Architecture for Joint Melody-Harmony Generation](#), by Alexis Kirke, arXiv, April 2026 (28 pages).
- [3061] [Quantum music](#), by Volkmar Putz and Karl Svozil, arXiv, August 2015 (5 pages).
- [3062] [Quantum Computer: Hello, Music!](#), by Eduardo R. Miranda, arXiv, June 2020 (32 pages).
- [3063] [Creative Quantum Computing: Inverse FFT, Sound Synthesis, Adaptive Sequencing and Musical Composition](#), by Eduardo R. Miranda, arXiv, December 2021 (33 pages).
- [3064] [The Arrival of Quantum Computer Music](#), by Eduardo R. Miranda, May 2020.
- [3065] [Quantum Computer Music](#), by Eduardo Reck Miranda, SpringerLink, 2022 📖.
- [3066] [Making Music Using Two Quantum Algorithms](#), by Euan J. Allen, Jacob F. F. Bulmer, and Simon D. Small, arXiv, January 2022 (13 pages).
- [3067] [New Directions in Quantum Music: concepts for a quantum keyboard and the sound of the Ising model](#), by Giuseppe Clemente, Arianna Crippa, Karl Jansen, and Cenk Tüysüz, arXiv, April 2022 (14 pages).
- [3068] [QuiKo: A Quantum Beat Generation Application](#), by Scott Oshiro, arXiv, April 2022 (23 pages).
- [3069] [A Quantum Natural Language Processing Approach to Musical Intelligence](#), by Eduardo Reck Miranda, Richie Yeung, Anna Pearson, Konstantinos Meichanetzidis, and Bob Coecke, arXiv, December 2021 (41 pages).
- [3070] [Advancing generative AI for music with photonics](#), by Eduardo Reck Miranda and Brian Siegelwax, International Journal of Parallel, Emergent and Distributed Systems, February 2025 (20 pages).
- [3071] [Advances in Quantum Computer Music](#), by Eduardo Reck Miranda et al., WSPC Book Series in Unconventional Computing: Volume 6, January 2025 (308 pages).
- [3072] [Qubism](#), by Eduardo Reck Miranda, 2024 (8 track music album).
- [3073] [PhD Quantum Computing in the Arts, Music and Humanities](#), University of Plymouth, 2025.
- [3074] [1st International Symposium on Quantum Computing and Musical Creativity](#), by ISQCMC, ISQCMC, 2021.
- [3075] [Superconducting qubits as musical synthesizers for live performance](#), by Spencer Topel, Kyle Serniak, Luke Burkhardt, and Florian Carle, arXiv, March 2022 (17 pages).
- [3076] [BruQner - The Sound Of Entanglement](#), YouTube, June 2025 (18 mn).
- [3077] [Music Composition Using Quantum Annealing](#), by Ashish Arya, Ludmila Botelho, Fabiola Cañete, Dhruvi Kapadia, and Özlem Salehi, arXiv, January 2022 (29 pages).
- [3078] [Teaching Qubits to Sing: Mission Impossible?](#), by Eduardo Reck Miranda and Brian N. Siegelwax, arXiv, August 2022 (31 pages).
- [3079] [Quantum Synth: a Quantum-Computing-based synthesizer](#), by Omar Costa Hamido, Giovanni Amedeo Cirillo, and Edoardo Giusto, Proceedings of the 15th International Audio Mostly Conference, September 2020 (4 pages) 📖.
- [3080] [A Quantum-Classical Network for Beat-Making Performance](#), by Scott Oshiro et al., 2020 (13 pages).
- [3081] [Intro to Quantum Harmony: Chords in Superposition](#), by Christopher Dobrian and Omar Costa Hamido, arXiv, April 2024 (13 pages).
- [3082] [Adventures in quantumland](#), by Oscar Costa Hamido, 2021 (142 pages).
- [3083] [Techniques for Quantum-Computing-Aided Algorithmic Composition: Experiments in Rhythm, Timbre, Harmony, and Space](#), by Christopher Dobrian and Omar Costa Hamido, arXiv, May 2025 (6 pages).
- [3084] [A Quantum Guitar](#), by Bob Coecke, arXiv, September 2025 (5 pages).
- [3085] [Q1Synth: A Quantum Computer Musical Instrument](#), by Eduardo Reck Miranda et al., Applied Sciences, February 2023.
- [3086] [Level Generation with Quantum Reservoir Computing](#), by João S. Ferreira, Pierre Fromholz, Hari Shaji, and James R. Wootton, arXiv, May 2025 (7 pages).
- [3087] [A quantum reservoir computing approach to computer-aided music composition](#), by Eduardo Reck Miranda and Hari Vignesh Shaji, Academia Quantum, May 2025 (16 pages).
- [3088] [Quantum Brush: A quantum computing-based tool for digital painting](#), by João S. Ferreira, Arianna Crippa, Astryd Park, Daniel Bultrini, Pierre Fromholz, Roman Lipski, Karl Jansen, and James R. Wootton, arXiv, September 2025 (10 pages).
- [3089] [Variational Quantum Brushes](#), by Jui-Ting Lu, Henrique Ennes, Chih-Kang Huang, and Ali Abbassi, arXiv, December 2025 (18 pages).
- [3090] [Procedural Generation and Games at the Dawn of Fault Tolerant Quantum Computing](#), by Daniel Bultrini and James Wootton, arXiv, August 2025 (9 pages).
- [3091] [Quantum Concept Music Score from Quantum Pictorialism: Musical Incarnation of a Bell-Pair under Measurements](#), by Rakhat-Bi Abdysagay and Bob Coecke, arXiv, October 2025 (6 pages).
- [3092] [Libby Heaney: Ent-Commissioned by Light Art Space \(LAS\)](#), by Libby Heaney, 2022.
- [3093] [Using Real Quantum Hardware to Make Art Creatives in Quantum](#), by Joel Russel Huffman, 2023.
- [3094] [Open sea quantum portraits](#), by Vanarman.
- [3095] [Creating Fractal Art With Qiskit](#), by Wiktor Mazin, June 2022.
- [3096] [Investigating the usefulness of Quantum Blur](#), by James R. Wootton and Marcel Pfaffhauser, arXiv, October 2023 (11 pages).
- [3097] [horomox](#), by Serge Aa, 2025.
- [3098] Serge Aa, "OArt: Quantum exo-AI shelters for the aesthetics of strategic emptiness", in: July 2026, URL: <https://www.scienceopen.com/hosted-document?doi=10.14236/ewic/EVA2026.8>.
- [3099] [horomox graphic art repository](#), by Serge Aa, 2025.
- [3100] [Scientific process inspires UChicago art/science lab, whose latest play entangles quantum concepts and kung fu](#), by Louise Lerner, December 2019.
- [3101] [Game Design Inspired by Scientific Concepts of Quantum Physics](#), by Sunanda Prabhu Gaunkar, Uri Zvi, et al., arXiv, February 2024 (4 pages).
- [3102] [Quantum computing inspired paintings: reinterpreting classical masterpieces](#), by Arianna Crippa, Yahui Chai, Omar Costa Hamido, Paulo Itaborai, and Karl Jansen, arXiv, May 2025 (10 pages).
- [3103] [Roadmap to Quantum Aesthetics](#), by Ivan C. H. Liu and Hsiao-Yuan Chen, arXiv, February 2026 (7 pages).
- [3104] ["Do you speak quantum?" A quantum physics and art exhibitio](#), by Chiara Decaroli and Maciej Malinowski, June 2022.
- [3105] [From Understanding to Resonance: A Case Study of Quantum Music and Embodied Science Communication](#), by Kenji Noda, Akira Koga, Stefan Heusler, and Koji Hashimoto, arXiv, August 2026 (15 pages).
- [3106] [Playing Mastermind on quantum computers](#), by Lvzhou Li, Jingquan Luo, and Yongzhen Xu, arXiv, February 2023 (33 pages).
- [3107] [Quantum Go: Designing a Proof-of-Concept on Quantum Computer](#), by Shibashankar Sahu, Biswaranjan Panda, Arnab Chowhan, Bikash K. Behera, and Prasanta K. Panigrahi, arXiv, June 2022 (7 pages).

- [3108] [Defining Quantum Games](#), by Laura Piispanen, Marcel Pfaffhauser, James Wootton, Julian Togelius, and An-nakaisa Kultima, arXiv, January 2025 (24 pages).
- [3109] [Experimental Implementation of the Quantum Volunteer's Dilemma on NISQ Hardware: Noise Analysis and Digital-Twin Validation](#), by Germán D. Díaz Agreda, Jhon Alejandro Andrade Hoyos, Carlos A. Durán Paredes, Sebastián Cajas Ordoñez, Noah Dane Hebdon, Siong Thye Goh, and Dax Enshan Koh, arXiv, May 2026 (30 pages).
- [3110] [QOuLiPo: What a quantum computer sees when it reads a book](#), by Christophe Jurczak, arXiv, May 2026 (31 pages).
- [3111] [Revolutionizing Digital Media: The Impact of Quantum Computing](#), by Greg Doig et al., June 2023.
- [3112] [A Quantum Approach to News Verification from the Perspective of a News Aggregator](#), by Theodore Andronikos and Alla Sirokofskich, arXiv, February 2024 (22 pages).
- [3113] [From Qubits to Opinions: Operator and Error Syndrome Measurement in Quantum-Inspired Social Simulations on Transversal Gates](#), by Yasuko Kawahata, arXiv, April 2024 (27 pages).
- [3114] [Discussion Model for Propagation of Social Opinion via Quantum Galois Noise Channels:Entanglement, Super-Spreader](#), by Yasuko Kawahata, arXiv, November 2023 (36 pages).
- [3115] [Quantum Simulations of Opinion Dynamics](#), by Xingyu Guo, Xiaoyang Wang, and Lingxiao Wang, arXiv, March 2026 (9 pages).
- [3116] [Quantum Technology for Military Applications](#), by Michal Krelina, arXiv, November 2021 (52 pages).
- [3117] [An introduction to military quantum technology for policy-makers](#), by Michal Krelina, SIPRI, March 2025 (16 pages).
- [3118] [Introduction to Quantum Computing](#), by William Oliver, MIT, December 2019 (21 slides).
- [3119] [Defense Research and Innovation: launch of a new ASTRID call for projects on quantum technologies](#), by ANR, July 2020.
- [3120] [Defence Innovation Accelerator for the North Atlantic \(DI-ANA\)](#), by NATO, September 2023.
- [3121] [Quantum Technologies and the Science for Peace and Security Programme](#), by NATO, November 2023 (24 pages).
- [3122] [Drones, radars, nuclear: how the quantum will change the war](#), by Vic Castro et al., February 2020.
- [3123] [UK Military Wants to Install Quantum Computers in Tanks for Some Reason](#), by Lonnie Lee Hood, June 2022.
- [3124] [Quantum Communication for Military Applications](#), by Niels M. P. Neumann, Maran P. P. van Heesch, and Patrick de Graaf, arXiv, November 2020 (11 pages).
- [3125] [NATO cybersecurity center finishes tests of quantum-proof network](#), by Jonathan Greig, March 2022.
- [3126] [Quantum Computing at NASA: Current Status](#), by Rupak Biswas, September 2017 (21 slides).
- [3127] [Assessing and Advancing the Potential of Quantum Computing: A NASA Case Study](#), by Eleanor G. Rieffel, Rupak Biswas, et al., arXiv, June 2024 (27 pages).
- [3128] [Demonstration of quantum advantage in machine learning](#), by D. Ristè, Marcus P. da Silva, Colm A. Ryan, Andrew W. Cross, John A. Smolin, Jay M. Gambetta, Jerry M. Chow, and Blake R. Johnson, arXiv, December 2015 (8 pages).
- [3129] [Hyper-Spectral Image Classification Using Adiabatic Quantum Computation](#), by Bartłomiej Gardas et al., IEEE IGARSS 2023, July 2023 (5 pages) 📄.
- [3130] [Phase-coded radar waveform AI-based augmented engineering and optimal design by Quantum Annealing](#), by Timoth   Presles et al., 2021 (9 pages).
- [3131] [Phase-coded Radar Waveform Design with Quantum Annealing](#), by Timothe Presles, Cyrille Enderli, Gilles Burel, and El Houssain Baghious, arXiv, October 2023 (11 pages).
- [3132] [Quantum Algorithms for Drone Mission Planning](#), by Ethan Davies and Pranav Kalidindi, arXiv, September 2024 (14 pages).
- [3133] [A Hybrid Classical Quantum Computing Approach to the Satellite Mission Planning Problem](#), by Nils Quetschlich, Vincent Koch, Lukas Burgholzer, and Robert Wille, arXiv, July 2023 (6 pages).
- [3134] [Quantum algorithms applied to satellite mission planning for Earth observation](#), by Serge Rainjonneau, Igor Tokarev, Sergei Iudin, Saaketh Rayaprolu, Karan Pinto, Daria Lemtiuzhnikova, Miras Koblan, Egor Barashov, Mo Kordzanganeh, Markus Pflitsch, and Alexey Melnikov, arXiv, August 2023 (13 pages).
- [3135] [Optimization of Image Acquisition for Earth Observation Satellites via Quantum Computing](#), by Ant  n Makarov, M  rcio M. Taddei, Eneko Osaba, Giacomo Franceschetto, Esther Villar-Rodr  guez, and Izaskun Oregi, arXiv, July 2023 (12 pages).
- [3136] [Pasqal and Thales Successfully Solve Satellite Planning Challenges with Neutral-Atom Quantum Computing](#), by Pasqal, May 2024.
- [3137] [Satellite Mission Planning with Rydberg Atoms](#), by Michel Nowak, Benjamin Marchand, Yassine Naghmouchi, Serge Rainjonneau, Wesley Coelho, Louis Vignoli, and Louis-Paul Henry, arXiv, June 2026 (10 pages).
- [3138] [Toward Quantum-Optimized Flow Scheduling in Multi-Beam Digital Satellites](#), by Qiben Yan, John P. T. Stenger, and Daniel Gunlycke, arXiv, February 2026 (8 pages).

This **Other user segments** subsection contains 137 bibliographical references and notes containing at least 2,169 pages.

3.9 Science

Fundamental research is starting to test and use quantum computing, particularly in materials development and particle physics research [2357, 3139–3144]. I also put here the broad sector of climate change mitigation.

3.9.1 Climate change mitigation

Climate-change mitigation or containment easily come forward when the societal benefits of quantum computing and other quantum technologies are promoted by various stakeholders in the quantum ecosystem [3145, 3146]. Some vendors like PsiQuantum, Terra Quantum, Zapata and IQM promote related use cases in their communication.

We will see here what categories of quantum solutions and algorithms are being talked about and their credibility. They cover various chemical simulations enabling optimized carbon cycle processes, weather and climate forecasts, transportation, and power grid optimizations.

Most of the proposed solutions are very long term and require large FTQC QPUs [3147]. Overpromises on fixing climate change come for example from **McKinsey** [3148], **PsiQuantum** who has no functional quantum computer yet, **OQC** which created a weather forecasting prototype with 8 qubits [3149, 3150], and on weather modeling from **Pasqal** and **BASF** [3151] **FTQC**.

NISQ **Quantinuum** is a little more credible with working on low-emission refrigerant production which deals with simpler problems although their existing 20 functional qubits are of no real use today [3152].

Chemical and physics simulations

Chemical and material simulations are some of the most amazing potential use cases of quantum computing. It could enable us to better understand how nature works, how to imitate it and how to produce energy or various materials more efficiently [3153].

The typical futuristic quantum computing use cases are about creating:

- **New solvents** for carbon capture and adsorbents or solid sorbents for direct-air capture. [3154] proposed an algorithm based on a VQE to compute how a carbon dioxide molecule reacts with an ammonia molecule. The research involved using a classical supercomputer to simulate the quantum circuit, including the noise levels expected in a NISQ. Obtaining the vibrational energy levels of NH_2COOH proved difficult and would require an FTQC QPU [3155] **NISQ**.
- **Innovative catalysts for fertilizer production**, related to understanding the famous FeMoCo complex natural reaction in nitrogenase. We discussed it in the chemistry part (in 3.3.3) **FTQC**.
- **Higher density batteries**, enabling the electrification of transportation [3156] **FTQC**.
- **More efficient solar panels** even though the impact could be minimal, since the more relevant metric may be the cost per watt produced rather than the panel yield.
- **New zero-carbon cement clinkers**, although I've not yet seen any research paper on this (see 3.3.4).
- **New materials**, replacing cement, aluminum, and steel whose production and recycling are energy intensive. The proposed circuit in [834] requires QEC and FTQC, between 10^{10} and 10^{15} T gates to obtain a good chemical accuracy with the small studied molecules, with runtimes reaching 50 days and a need for 100 million physical qubits **FTQC**.
- **Lower direct combustion emissions** than coal by switching to liquefied natural gas (LNG), subject to lifecycle methane leakage and to the chosen system boundary [3157] **FTQC**.
- **Hydrogen production techniques** with modeling fuel cell membranes, catalysts and electrical currents **FTQC**.

PsiQuantum is pushing this climate change agenda for a reason: it requires quantum error correction and a large number of physical and logical qubits, namely as a starter, a million physical qubits and one hundred logical qubits. The use case the company is putting forward is the improvement of fertilizer production. The caveat is that even PsiQuantum's own literature positions FeMoCo complex quantum simulation in the multiple thousands of logical qubits scale and it would be only part of the solution since, after this simulation, the creation of a new catalyst and production process must be created and simulated as well. So, when the company says that green ammonia could be produced and used in ships by 2030, it is very optimistic.

Weather modeling and climate predictions

On a more operational basis and potentially shorter term, another use case for quantum computing and climate change is weather forecasting and climate modeling [3158]. Weather forecasting could potentially help run simulations and predict scenarios faster and more accurately than with existing solutions, which already work well. Weather modeling uses a wide range of data like temperature, pressure, wind, moisture, and other meteorological variables [3159–3161].

Weather forecasting brings some operational use cases:

- For finding optimum locations for wind and solar farms.
- To make more accurate weather simulations to predict energy production and improve grid balancing and supply predictions.

- For typhoon trajectory forecasting [3162].
- To predict CO₂ emissions peaks due to fossil fuel power plant activities in connection with renewable energy production and optimize the whole grid with better planning.
- To predict water supply or water quality [3163], and create irrigation scheduling [3164].
- To optimize the detection of wildfires with satellites [3165].

Modeling weather with quantum computers probably involves the ingestion of a lot of data which is not where they excel [3166]. It can lead for example to downscale imaging-data resolution [3167]. Also, many weather forecasting equations are nonlinear (Navier-Stokes equation). This is addressed with encoding multiple copies of a quantum state and letting these copies interact with each other via an interaction Hamiltonian [3168]. Related work from researchers from Oxford University shows a need for 755 logical qubits, 10^{29} gates and 10^{20} s of computing time, so billions of years. An error target of 10^{-29} creates an enormous error-correction overhead to create such logical qubits [3169]. The paper reports “the inversion of a matrix of size $N = 2^{17}$ on currently available quantum backends from IBM” where the feat is not inverting a matrix per se but estimating expectation values related to matrix eigenvalues. A full matrix inversion has an exponential cost due to the number of shots necessary for reading out its results. Making weather forecasts requires probably tens of billions of parameters [3170] **FTQC**.

On the other hand, climate modeling may be more macroscopic in nature and require less data, although very diverse in type. It may be more accessible to future NISQ and FTQC QPUs [3171, 3172] **NISQ** **FTQC**.

In 2025, a team from Brazil developed a toy-model of climate change prediction model running on an Eviden classical emulator hardware that they acquired in 2021 named CIMATEC KUATOMU. It was using only between 6 and 8 qubits while the emulator supports up to 30-40 qubits depending on the hardware configuration [3173] **NISQ**.

In 2026, a German team prototyped a quantum machine learning based modeling of cloud cover [3174].

Other less data-hungry methods may rely on QML-based long-term series forecasting [3175] **NISQ**.

Transportation optimizations

This topic was already dealt with in the transportation and logistics section of this part of the book. It is about the famous TSP problem (traveling salesperson problem) and other variants but can also involve the optimization of trucks loads to avoid trucks being driven empty. This also includes route planning with routes offering better fuel efficiency.

Power grid optimizations

Power grid optimizations could potentially get the help of quantum computing when the problem complexity exceeds the capacity of classical systems and algorithms [3176].

The laundry list of potential solutions includes:

- Optimize power generation through simulations and map demand/supply with better accuracy [3177, 3178].
- Identify leaks and gaps in power supply, impacting how it is stored, transmitted, and distributed.
- Preempt forest fires and make flood predictions [3179, 3180].
- Predict extreme weather conditions that can impact power supply and surge in demand.

Some of these potential applications are also inventoried in a Q4Climate initiative publication [3181].

3.9.2 Astrophysics

Quantum applications in astrophysics range from quantum sensing to quantum computing.

Superconducting qubits, or more precisely SQUIDs and microwave cavities, are used to detect dark matter, in the form of axions, a dark matter candidate, or “hidden photons”, which are hypothetical vector bosons which could mix with regular photons via the so-called kinetic mixing process [3182]. Other researchers are also using squeezed states to detect axions [3183].

Quantum computing is tested in multiple domains, either in quantum matter simulations or for analyzing instrument data.

NISQ Simulations were tested for exotic magnetic materials [2358]. In the NISQ regime, researchers are toying with QPU in surprisingly low regimes. Researchers from Taiwan and the US used a federated QT-LSTM framework to analyze gravitational wave signal. They actually used QNNs to generate the weights of classical LSTM models and with a mere 11 qubits [3184]. An Italian team evaluated the use of Quantum Convolutional Neural Networks (QCNNs) for identifying signals resembling Gamma-Ray Bursts (GRBs). They observed better accuracy than classical methods as the number of qubits was increasing from 4 to 12! Again, this is classical computing improving classical computing [3185].

NISQ Black hole simulations were also done with 12 qubits in 2025, at a scale that remains classically reproducible [3186, 3187]. A comparison of black hole mass estimations between classical and quantum deep learning tools yielded worse results with the quantum version, which used 4 to 5 qubits. [3188]. There are also FTQC algorithms proposals for the simulation of

lattice gauge theories in the Kogut-Susskind formulation to explore various physical phenomena, from the fundamental interactions of elementary particles using Quantum Chromodynamics (QCD [3189]) and in astrophysics and cosmology [3190]. In August 2026, a team of UK scientists simulated black hole thermality with three chiral spin chains of 32 qubits running on an IBM processor using out-of-time-ordered correlators (OTOCs) [3191].

NISQ A Spanish team published a paper in 2025 on a neutrino telescope QML-based event classification task, the data coming from the IceCube Neutrino Observatory in the South Pole and the KM3NeT detectors that are in the Mediterranean Sea off the coasts of Sicily and France. It was tested on an IBM 127-qubit QPU. But it used only 40 disjoint pairs of qubits, each implementing a simple QCNN [3192].

NASA and Harvard scientists proposed in 2025 an architecture associating an imager generating coherent light directly converted into so-called pixel-qubits and processed into memory qubits to detect exoplanets, but it was not implemented experimentally [3193].

3.9.3 High-energy physics

High-energy physics (HEP) is a domain where quantum computing can be useful, mostly to analyze the data coming out of particle accelerators but also for various standard model quantum physics simulation tasks [3194–3197].

After having investigated quantum computing for a good number of years with a first workshop organized in November 2018, **CERN** launched a formal **Quantum Technology Initiative** (QTI) in September 2020 [3198]. They want to use quantum computing to analyze the noisy data coming from their ultrasensitive particles detectors and to simulate the behavior of many-body quantum phenomena [3199]. CERN also participates in international quantum computing education and training with its online training resources.

NISQ IBM worked with CERN to select/classify LHC events using QSVM [3200]. Scientists in China are doing similar work to detect the Higgs boson in the future Circular Electron Positron Collider (CEPC) particle accelerator to be deployed in 2030 in China, with its underground 100 km circumference [3201]. It runs with only 6 qubits!

NISQ Back in 2017, **Caltech** used a D-Wave Two X quantum annealer with 1,098 qubits to “rediscover” the Higgs boson using CERN LHC data and a QAML algorithm (quantum annealing machine learning) [3202]. Later in 2020, they improved it with their QAML-Z algorithm, quantum annealing machine learning model zooming in on a region of the analyzed energy surface [3203]. Sometimes, scientists are toying with quantum computers with so few qubits that it doesn’t make any

sense, the underlying mathematics being much simpler to solve with a simple classical computer [3204].

NISQ Recent tests start to bring some interesting results with more qubits (Figure 204), like in this paper from the University of Washington relating the use of a 156-qubit QPU from IBM along with some error mitigation technique to run a variant of VQE to simulate scalar field scattering. It is still in a regime that can be classically simulated with MPS (tensor networks). A crossover would require circuits whose entanglement structure defeats the best tensor-network methods at the target accuracy, with hardware runtime and sampling cost measured against them. Other experiments with similar QPUs were conducted using quantum error mitigation techniques [3205–3207].

NISQ There were also a series of HEP physics simulation runs on trapped-ion qudits from AQT in Austria or in emulation mode, showing the potential of this method to simulate 2D gauge models [3209–3215]. HEP scientists in the UK also proposed in 2025 the use of continuous-variable photonic quantum extreme learning machines (QELM) for selecting collider data, and tested it in a classical simulator [3208] (Figure 203). Likewise, researchers in Spain proposed to use qutrits for LHC data analysis [3216].

FTQC On an even more positive note, a 2025 paper from CERN on quantum anomaly detection demonstrates a theoretical exponential speedup, an asymptotic algorithmic result with no noise-tolerant near-term implementation or hardware resource estimate, for simulating Kogut-Susskind Hamiltonians via orbifold lattice which is related to Yang-Mills theory and QCD [3217, 3218].

NISQ IBM et al. simulated real-time string dynamics of a (2+1)D U(1) quantum link model, a compact and always-confining QED, quenching an initial electric-flux string and following its transverse quantum fluctuations up to thermalization. It used up to 112 qubits with digital Trotter evolution and error mitigation. No quantum advantage was claimed [3226].

NISQ Google with TU Munich studied a (2+1)D Z₂ lattice gauge theory on a 2D array of superconducting qubits, preparing a low-energy state with a shallow variational circuit, creating charge excitations with local gates, and imaging string dynamics under Trotterized time evolution. The array held on the order of a few dozen qubits but with no quantum advantage given a Z₂ theory at these sizes is classically tractable by tensor networks and exact diagonalization [3227].

NISQ IQus University of Washington and PNNL (USA) set up a framework for energy loss and hadronization of a heavy quark crossing a 1+1D SU(2) non-Abelian lattice of light quarks, tracking light-quark energy, local color-charge densities, and multipartite entanglement, with heavy-quark hopping via fermionic SWAPs and state preparation by domain decomposition. They used 18 qubits on an IBM processor with

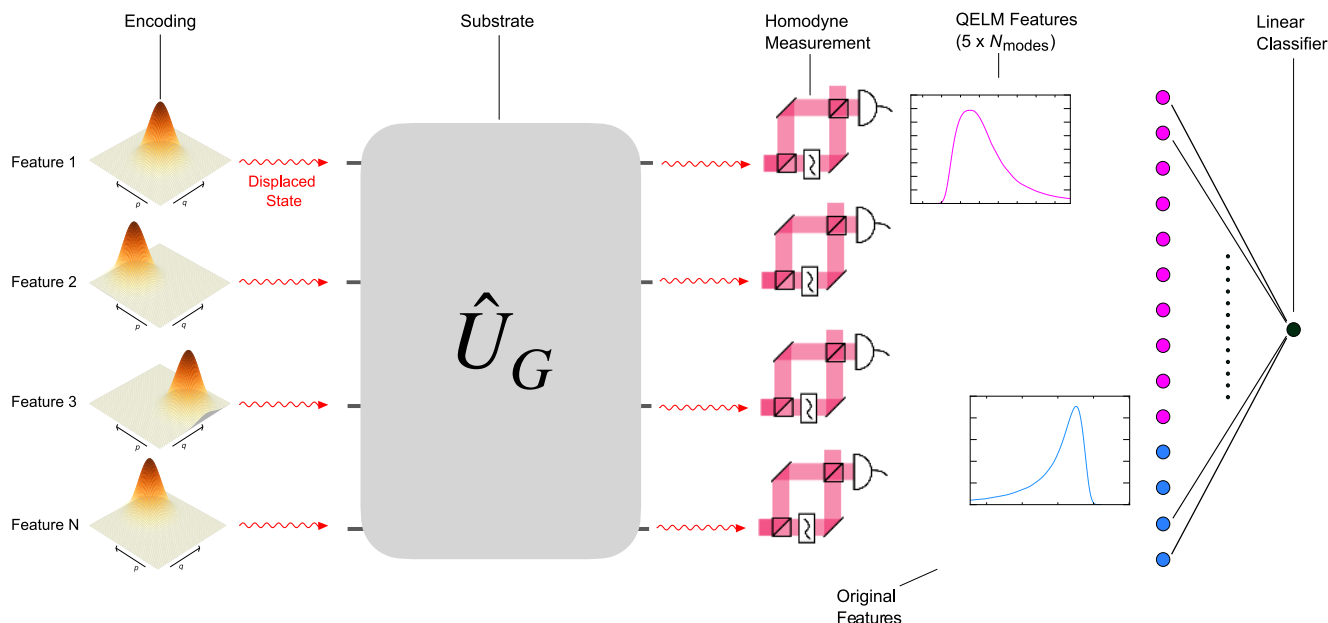


Figure 203: A quantum extreme learning machines (QELM) proposal to analyze data from HEP experiments. Source: [3208].

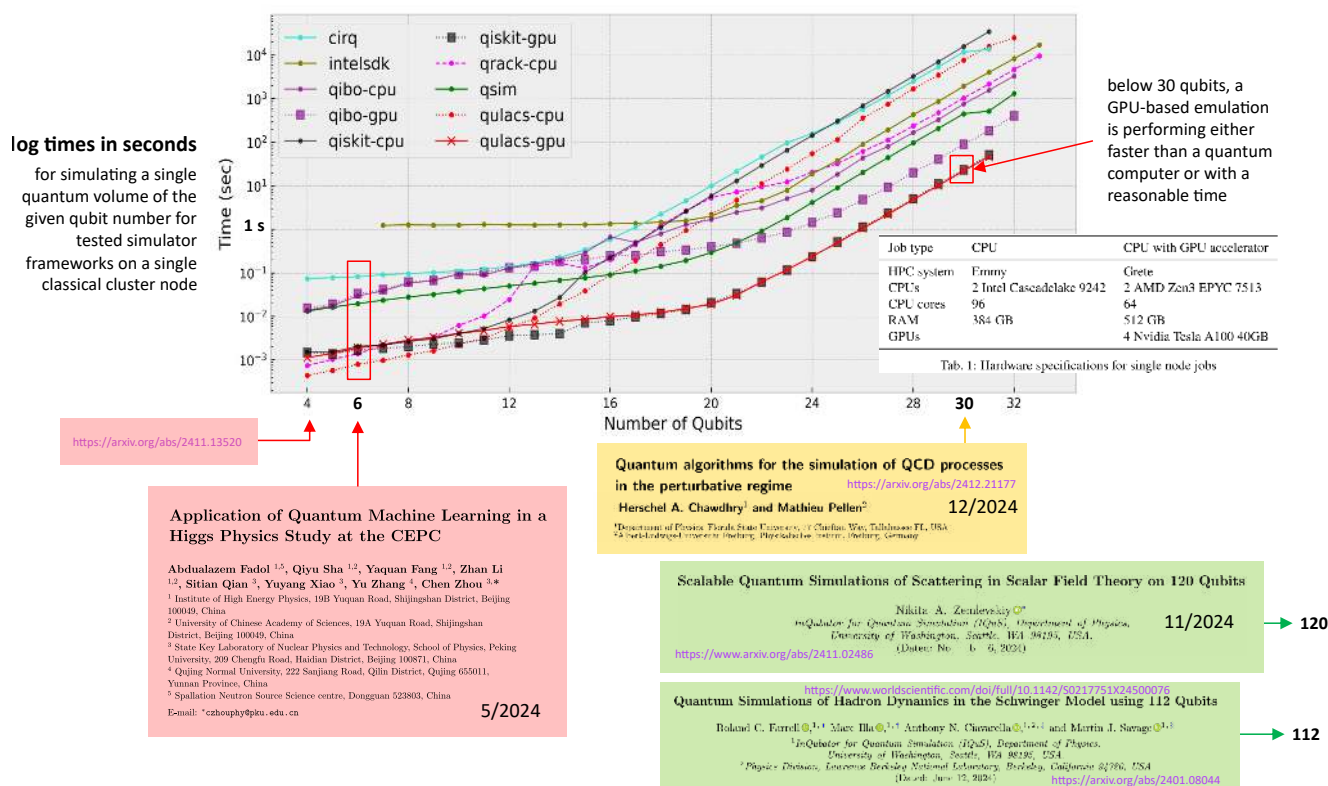


Figure 204: How HEP quantum computing simulations have grown in time to overcome the 100 physical qubit barrier. Sources for 3 qubits [3219, 3220], 4 qubits [3221], 6 qubits [3222], 30 qubits [3223], 112 qubits [3224] and 120 qubits [3225]. The above chart shows the time spent to emulate classically an equivalent circuit with a depth and breadth of N qubits, the best results being obtained with GPUs [1685]. For the circuits and systems measured here, GPU-based simulation compares well with the quantum runs up to about 30 qubits, in relative computing time and in seconds. This is specific to the tested circuit families, fidelities and outputs, and the configurations compared do not have matching prices, latency or amortization. © Consolidation by Olivier Ezratty, January 2025.

no quantum advantage, with results matching classical simulation [3228].

NISQ IQus University of Washington and LBNL (USA) simulated a hadron wavepacket in the 1+1D Schwinger model, a confining toy model of QCD with bound hadrons, on a 56-site lattice mapped to 112 qubits.

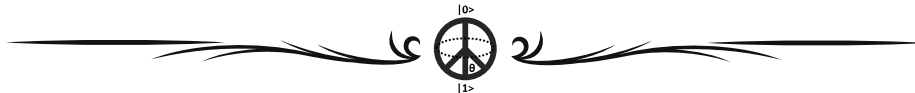
The vacuum came from SC-ADAPT-VQE [3229], the localized hadron was built on top, then propagated with second-order Trotterization and error mitigation. This is among the largest gauge-theory dynamics run to date. Yet no advantage was demonstrated, because the results remained reproducible with matrix product state simulations [3230].

NISQ Keio University, RIKEN iTHEMS, and Yukawa Institute demonstrated thermalization dynamics of a $(2+1)\text{D}$ q -deformed $\text{SU}(2)$ Yang-Mills theory with 18 qubits on a trapped-ion computer with no quantum advantage [3231]. Kavli IPMU, University of Tokyo with IBM proposed the sparsified bosonic SYK model as a target for future quantum advantage and reported first

simulations on both classical simulators and IBM superconducting qubits. SYK is relevant to holography and quantum gravity, hence its high-energy-theory interest. Sparsification deliberately trims the Hamiltonian to stay tractable on near-term hardware while preserving SYK physics, and the hardware runs are small scale. As the title says, no advantage was demonstrated [3232].

3.9.4 Quantum physics

Quantum computing prospects also exist to perform quantum-physics simulations in various fields, including gravitationally induced quantum entanglement [3233] or quantum field theory [3234, 3235].



3.9.5 Bibliography and notes

These are the bibliographical references and notes for the **Science** subsection.

- [834] [Quantum Computation for Periodic Solids in Second Quantization](#), by Aleksei V. Ivanov et al., PRR, 2022 (29 pages).
- [1685] [A comparison of HPC-based quantum computing simulators using Quantum Volume](#), by Lourens van Niekerk, Dhiraaj Kumar, Aasish Kumar Sharma, Tino Meisel, Martin Leandro Paleico, and Christian Boehme, arXiv, December 2024 (11 pages).
- [2357] [Quantum computing for data analysis in high energy physics](#), by Andrea Delgado, Jorge J. Martinez de Lejarza, et al., arXiv, December 2022 (23 pages).
- [2358] [Quantum computing enables simulations to unravel mysteries of magnetic materials](#), by Elizabeth Rosenthal, February 2021.
- [3139] [Quantum Algorithm for High Energy Physics Simulations](#), by Benjamin Nachman, February 2021 (6 pages).
- [3140] [Quantum Simulation for High Energy Physics](#), by Christian W. Bauer, Silvia Zorzetti, et al., arXiv, April 2022 (103 pages).
- [3141] [Simulating Collider Physics on Quantum Computers Using Effective Field Theories](#), by Christian W. Baue, November 2021 (7 pages).
- [3142] [Quantum computing hardware for HEP algorithms and sensing](#), by M. Sohaib Alam, Silvia Zorzetti, et al., arXiv, April 2022 (23 pages).
- [3143] [Snowmass White Paper: Quantum Computing Systems and Software for High-energy Physics Research](#), by Travis S. Humble, Dirk Krücker, et al., arXiv, March 2022 (17 pages).
- [3144] [Potential Applications of Quantum Computing at Los Alamos National Laboratory](#), by Andreas Bärttschi, Marc Vuffray, and Yu Zhang, arXiv, March 2025 (173 pages).
- [3145] [Sustainable Development - How Quantum Technologies Can Help Drive the UN's Sustainable Development Goals](#), by Capgemini, 2022 (15 pages).
- [3146] [What If Quantum Could Crack the Energy Transition's Toughest Problems?](#), by Maurice Berns, Matt Langione, Jens Burchardt, Jean-François Bobier, Endure McTier, Hanl Park, and Eden Cottee-Jones, Boston Consulting Group, September 2026.
- [3147] [How Quantum Computing Can Tackle Climate and Energy Challenges](#), by Annarita Giani et al., October 2022.
- [3148] [Quantum computing just might save the planet](#), by Peter Cooper, May 2022.
- [3149] [Precise Forecasting with hybrid quantum-classical algorithms](#), by Katy Alexander, April 2024.
- [3150] [Hybrid quantum-classical reservoir computing for simulating chaotic systems](#), by Filip Wudarski, Davide Venturelli, et al., arXiv, April 2024 (18 pages).
- [3151] [Pascal, BASF to collaborate on quantum compute-powered weather modeling](#), by Andrew Brust, July 2022.
- [3152] [How Quantum Computing Can Help Keep Things Cool](#), by Honeywell, 2022.
- [3153] [How quantum computing can help tackle global warming](#), by McKinsey, May 2022.
- [3154] [Carbon-capture technology could benefit from quantum computing](#), by Physics World, Physics World, April 2023.
- [3155] [Exploration of Quantum Computing in Materials Discovery for Direct Air Capture Applications](#), by Marco Antonio Barroca, Rodrigo Neumann Barros Ferreira, and Mathias Steiner, arXiv, May 2024 (19 pages).
- [3156] [Quantum computation of dominant products in lithium-sulfur batteries](#), by J.E. Rice, 2021 📄 (preprint on arXiv).
- [3157] [Quantum Computing and Simulations for Energy Applications: Review and Perspective](#), by Hari P. Paudel et al., August 2022 (46 pages).
- [3158] [Will Quantum Technology Be The Silver Bullet For Climate Change?](#), by Markus Pflichtsch, September 2022.
- [3159] [Quantum and climate action - The role of quantum computing in creating a sustainable future](#), by Daryl Pereira, January 2023.
- [3160] [The ESCAPE project: Energy-efficient Scalable Algorithms for Weather Prediction at Exascale](#), by Andreas Müller et al., 2019 (28 pages).
- [3161] [Quantum-Assisted Machine Learning Models for Enhanced Weather Prediction](#), by Saiyam Sakhuja, Shivanshu Siyanwal, Abhishek Tiwari, Britant, and Savita Kashyap, arXiv, March 2025.
- [3162] [Quantum-Enhanced Parameter-Efficient Learning for Typhoon Trajectory Forecasting](#), by Chen-Yu Liu, Yen-Jui Chang, et al., arXiv, May 2025 (11 pages).
- [3163] [Predicting Water Quality using Quantum Machine Learning: The Case of the Umgeni Catchment \(U20A\) Study Region](#), by Muhammad Al-Zafar Khan, Jamal Al-Karaki, and Marwan Omar, arXiv, November 2024 (13 pages).
- [3164] [A Physics-Grounded QUBO Encoding of Irrigation Scheduling for QAOA](#), by Alisher Ortikov and Alisher Ilhamov, arXiv, July 2026 (14 pages).

- [3165] [Partitioned Iterative Quantum Scheduling of Satellites for Urgent Disaster Response: Case study of Wildfire](#), by Lucas T. Braydwood, Taejin Park, Hirofumi Hashimoto, Zoe Gonzalez Izquierdo, Andrew Michaelis, Eleanor Rieffel, and Shon Grabbe, arXiv, June 2026 (11 pages).
- [3166] [Quantum data assimilation: a new approach to solving data assimilation on quantum annealers](#), by Shunji Kotsuki, June 2024 (9 pages).
- [3167] [Hybrid Quantum-Classical Corrective Diffusion Modeling for Meteorological Downscaling](#), by Rui Wang, Edoardo Pasetto, Amer Delilbasic, Morris Riedel, Kristel Michielsen, and Gabriele Cavallaro, arXiv, May 2026 (11 pages).
- [3168] [Variational quantum algorithms for nonlinear problems](#), by M. Lubasch, PRA, 2020 (7 pages).
- [3169] [Quantum Computers for Weather and Climate Prediction: The Good, the Bad and the Noisy](#), by Felix Tennie and Tim Palmer, arXiv, October 2022 (20 pages).
- [3170] [Strong scaling for numerical weather prediction at petascale with the atmospheric model NUMA](#), by Andreas Muller, 2018 (preprint on arXiv).
- [3171] [Opportunities and challenges of quantum computing for climate modelling](#), by Mierk Schwabe, Veronika Eyring, et al., arXiv, February 2025 (20 pages).
- [3172] [Quantum Neural Networks for Cloud Cover Parameterizations in Climate Models](#), by Lorenzo Pastori, Arthur Grundner, Veronika Eyring, and Mierk Schwabe, arXiv, February 2025 (39 pages).
- [3173] [Exploring Quantum Machine Learning for Weather Forecasting](#), by Maria Heloisa F. da Silva, Gleydson F. de Jesus, Christiano M. S. Nascimento, Valéria L. da Silva, and Clebson Cruz, arXiv, September 2025 (15 pages).
- [3174] [Quantum Machine Learning for Climate Modelling](#), by Mierk Schwabe, Lorenzo Pastori, Valentina Sarandrea, and Veronika Eyring, arXiv, December 2025 (6 pages).
- [3175] [QuLTFSF: Long-Term Time Series Forecasting with Quantum Machine Learning](#), by Hari Hara Suthan Chittoor, Paul Robert Griffin, Ariel Neufeld, Jayne Thompson, and Mile Gu, arXiv, March 2025 (6 pages).
- [3176] [Qubits for climate change](#), by Prachi Mishra, August 2022.
- [3177] [Quantum computing opportunities in renewable energy](#), by Annarita Giani and Zachary Eldredge, SN Computer Science, September 2021 (1).
- [3178] [Quantum Computers Can Now Interface With Power Grid Equipment](#), by NREL, July 2023.
- [3179] [Multiverse Computing deploys quantum technology to predict floods](#), by Multiverse Computing, October 2023.
- [3180] [Quantum-Train Long Short-Term Memory: Application on Flood Prediction Problem](#), by Chu-Hsuan Abraham Lin, Chen-Yu Liu, and Kuan-Cheng Chen, arXiv, July 2024 (6 pages).
- [3181] [Quantum technologies for climate change: Preliminary assessment](#), by Casey Berger, Agustin Di Paolo, Tracey Forrest, Stuart Hadfield, Nicolas Sawaya, Michał Stęchły, and Karl Thibault, arXiv, June 2021 (10 pages).
- [3182] [Searching for Dark Matter with a Superconducting Qubit](#), by Akash V. Dixit, April 2021 (7 pages).
- [3183] [A quantum-enhanced search for dark matter axions](#), by K. M. Backes, H. Wang, et al., arXiv, August 2020 (7 pages).
- [3184] [Federated Quantum-Train Long Short-Term Memory for Gravitational Wave Signal](#), by Chen-Yu Liu, Samuel Yen-Chi Chen, Kuan-Cheng Chen, Wei-Jia Huang, and Yen-Jui Chang, arXiv, March 2025 (6 pages).
- [3185] [Benchmarking Quantum Convolutional Neural Networks for Signal Classification in Simulated Gamma-Ray Burst Detection](#), by Farida Farsian, Tiziana Trombetti, et al., arXiv, January 2025 (9 pages).
- [3186] [Capturing the Page Curve and Entanglement Dynamics of Black Holes in Quantum Computers](#), by Talal Ahmed Chowdhury, Kwangmin Yu, Muhammad Asaduzzaman, and Raza Sabbir Sufian, arXiv, March 2025 (23 pages).
- [3187] [Quantum Simulation of Fermions in \$AdS_2\$ Black Hole: Chirality, Entanglement, and Spectral Crossovers](#), by Kazuki Ikeda and Yaron Oz, arXiv, September 2025 (50 pages).
- [3188] [Comparative Analysis of Black Hole Mass Estimation in Type-2 AGNs: Classical vs. Quantum Machine Learning and Deep Learning Approaches](#), by Sathwik Narkedimilli, Aswath Babu, et al., arXiv, February 2025 (29 pages).
- [3189] [Universal framework with exponential speedup for the quantum simulation of quantum field theories including QCD](#), by Jad C. Halimeh, Masanori Hanada, and Shunji Matsuura, arXiv, June 2025 (40 pages).
- [3190] [Exponential improvements in the simulation of lattice gauge theories using near-optimal techniques](#), by Mason L. Rhodes, Michael Kreshchuk, and Shivesh Pathak, arXiv, January 2025 (48 pages).
- [3191] [Simulating Black Hole Thermality and Interior Scrambling on a Superconducting Quantum Processor](#), by Ryan Smith, Ewan Forbes, Iason Sofos Andrew Hallam, and Giannis Pachos, arXiv, August 2026 (18 pages).
- [3192] [Neutrino Telescope Event Classification on Quantum Computers](#), by Pablo Rodriguez-Grasa, Pavel Zhelnin, Carlos A. Argüelles, and Mikel Sanz, arXiv, June 2025 (11 pages).
- [3193] [Enhancing Optical Imaging via Quantum Computation](#), by Aleksandr Mokeev, Babak Saif, Mikhail D. Lukin, and Johannes Borregaard, PRX Quantum, January 2026 (18 pages).
- [3194] [Quantum Computing for High-Energy Physics: State of the Art and Challenges](#), by Alberto Di Meglio et al., PRX Quantum, August 2024.
- [3195] [Quantum Information meets High-Energy Physics: Input to the update of the European Strategy for Particle Physics](#), by Yoav Afik, Knut Zoch, et al., arXiv, April 2025 (10 pages).
- [3196] [Efficient use of quantum computers for collider physics](#), by Christian W. Bauer, arXiv, March 2025 (5 pages).
- [3197] [Quantum Complexity and New Directions in Nuclear Physics and High-Energy Physics Phenomenology](#), by Caroline E. P. Robin and Martin J. Savage, arXiv, April 2026 (64 pages).
- [3198] [CERN Quantum Technology Initiative Strategy and Roadmap](#), by Di Meglio, October 2021 (46 pages).
- [3199] [Particle track reconstruction with noisy intermediate-scale quantum computers](#), by Tim Schwägerl, Cigdem Issever, Karl Jansen, Teng Jian Khoo, Stefan Kühn, Cenk Tüysüz, and Hannsjörg Weber, arXiv, March 2023 (6 pages).
- [3200] [Application of quantum machine learning using the quantum kernel algorithm on high energy physics analysis at the LHC](#), by Sau Lan Wu et al., Physical Review Research, September 2021.
- [3201] [Application of Quantum Machine Learning in a Higgs Physics Study at the CEPC](#), by Abdualazem Fadol, Qiyu Sha, Yaquan Fang, Zhan Li, Sitian Qian, Yuyang Xiao, Yu Zhang, and Chen Zhou, arXiv, March 2024 (14 pages).
- [3202] [Solving a Higgs optimization problem with quantum annealing for machine learning](#), by Alex Mott, Daniel Lidar, et al., Nature, July 2017 (5 pages) (1).
- [3203] [Quantum adiabatic machine learning with zooming](#), by Alexander Zlokapa, Alex Mott, Joshua Job, Jean-Roch Vlimant, Daniel Lidar, and Maria Spiropulu, arXiv, October 2020 (9 pages).
- [3204] [Simulating neutrino oscillations on a superconducting qutrit](#), by Ha C. Nguyen, Bao G. Bach, Tien D. Nguyen, Duc M. Tran, Duy V. Nguyen, and Hung Q. Nguyen, arXiv, February 2023 (26 pages).
- [3205] [Observation of hadron scattering in a lattice gauge theory on a quantum computer](#), by Julian Schuhmacher, Guo-Xian Su, Jesse J. Osborne, Anthony Gandon, Jad C. Halimeh, and Ivano Tavernelli, arXiv, May 2025 (13 pages).

- [3206] [Resource-Efficient Simulations of Particle Scattering on a Digital Quantum Computer](#), by Yahui Chai, Joe Gibbs, Vincent R. Pascuzzi, Zoë Holmes, Stefan Kühn, Francesco Tacchino, and Ivano Tavernelli, arXiv, July 2025 (18 pages).
- [3207] [Real-Time Dynamics in a \(2+1\)-D Gauge Theory: The Stringy Nature on a Superconducting Quantum Simulator](#), by Jesús Cobos, Enrique Rico, et al., arXiv, July 2025 (25 pages).
- [3208] [Continuous-variable photonic quantum extreme learning machines for fast collider-data selection](#), by Benedikt Maier, Michael Spannowsky, and Simon Williams, arXiv, October 2025 (21 pages).
- [3209] [Variational quantum simulation of U\(1\) lattice gauge theories with qudit systems](#), by Pavel P. Popov, Martin Ringbauer, Erez Zohar, Valentin Kasper, et al., arXiv, July 2023 (19 pages).
- [3210] [Digital quantum simulation of a \(1+1\)D SU\(2\) lattice gauge theory with ion qudits](#), by Giuseppe Calajò, Simone Montangero, Pietro Silvi, et al., arXiv, October 2024 (15 pages).
- [3211] [Non-perturbative signatures of fractons in the twisted multi-flavor Schwinger Model](#), by Pavel P. Popov, Valentin Kasper, Maciej Lewenstein, Erez Zohar, Paolo Stornati, and Philipp Hauke, arXiv, April 2024 (14 pages).
- [3212] [Simulating 2D lattice gauge theories on a qudit quantum computer](#), by Michael Meth, Christine Muschik, Martin Ringbauer, et al., arXiv, October 2024 (23 pages).
- [3213] [Quantum Simulation of non-Abelian Lattice Gauge Theories: a variational approach to \$\mathfrak{su}\(D\)\$ s](#), by Emanuele Gaz, Pavel P. Popov, Guy Pardo, Maciej Lewenstein, Philipp Hauke, and Erez Zohar, arXiv, January 2025 (16 pages).
- [3214] [Probing Hadron Scattering in Lattice Gauge Theories on Qudit Quantum Computers](#), by Rohan Joshi, Jan C. Louw, Michael Meth, Jesse J. Osborne, Kevin Mato, Guo-Xian Su, Martin Ringbauer, and Jad C. Halimeh, arXiv, July 2025 (2121 pages).
- [3215] [Efficient Qudit Circuit for Quench Dynamics of \$\mathfrak{su}\(2+1\)\$ D Quantum Link Electrodynamics](#), by Rohan Joshi, Michael Meth, Jan C. Louw, Jesse J. Osborne, Kevin Mato, Martin Ringbauer, and Jad C. Halimeh, arXiv, July 2025 (1919 pages).
- [3216] [Qutrits for physics at the LHC](#), by Miranda Carou Laiño, Veronika Chobanova, and Miriam Lucio Martínez, arXiv, October 2025 (10 pages).
- [3217] [Exponential speedup in quantum simulation of Kogut-Susskind Hamiltonian via orbifold lattice](#), by Georg Bergner and Masanori Hanada, arXiv, May 2025 (20 pages).
- [3218] [Fundamental Physics at the Frontier of Noisy Quantum Computation](#), by Nikita A. Zemlevskiy, arXiv, September 2026 (413 pages).
- [3219] [Jet evolution in a quantum computer: quark and gluon dynamics](#), by Nuno Filipe Castro, José Guilherme Milhano, and Maria Gabriela Jordão Oliveira, arXiv, February 2025 (17 pages).
- [3220] [Quantum error mitigation in optimized circuits for particle-density correlations in real-time dynamics of the Schwinger model](#), by Domenico Pomarico, Elisa Ercolessi, et al., arXiv, January 2025 (19 pages).
- [3221] [Quantum Attention for Vision Transformers in High Energy Physics](#), by Alessandro Tesi, Katia Matcheva, et al., arXiv, November 2024 (9 pages).
- [3222] [Application of quantum machine learning in a Higgs physics study at the CEPC](#), by Qiyu Sha Abdualazem Fadol, International Journal of Modern Physics A, March 2024 (14 pages)  (preprint on [arXiv](#)).
- [3223] [Quantum algorithms for the simulation of QCD processes in the perturbative regime](#), by Herschel A. Chawdhry and Mathieu Pellen, arXiv, December 2024 (9 pages).
- [3224] [Quantum Simulations of Hadron Dynamics in the Schwinger Model using 112 Qubits](#), by Roland C. Farrell, Marc Illa, Anthony N. Ciavarella, and Martin J. Savage, arXiv, June 2024 (54 pages).
- [3225] [Scalable Quantum Simulations of Scattering in Scalar Field Theory on 120 Qubits](#), by Nikita A. Zemlevskiy, arXiv, November 2024 (50 pages).
- [3226] [String dynamics of a \(2+1\)D U\(1\) quantum link model on a digital quantum computer](#), by Anthony Gandon, Alessandro Mariani, Debasish Banerjee, Emilie Huffman, Gurtej Kanwar, Francesco Tacchino, Uwe-Jens Wiese, and Ivano Tavernelli, arXiv, June 2026 (25 pages).
- [3227] [Visualizing dynamics of charges and strings in \(2 + 1\)D lattice gauge theories](#), by T. A. Cochran, P. Roushan, et al., Nature, June 2025.
- [3228] [A Framework for Quantum Simulations of Energy-Loss and Hadronization in Non-Abelian Gauge Theories: SU\(2\) Lattice Gauge Theory in 1+1D](#), by Zhiyao Li, Marc Illa, and Martin J. Savage, arXiv, December 2025 (28 pages).
- [3229] [Scalable Circuits for Preparing Ground States on Digital Quantum Computers: The Schwinger Model Vacuum on 100 Qubits](#), by Roland C. Farrell, PRX Quantum, April 2024.
- [3230] [Quantum simulations of hadron dynamics in the Schwinger model using 112 qubits](#), by Roland C. Farrell, Marc Illa, Anthony N. Ciavarella, and Martin J. Savage, Physical Review D, June 2024  (preprint on [osti.gov](#)).
- [3231] [Onset of thermalization of q-deformed SU\(2\) Yang-Mills theory on a trapped-ion quantum computer](#), by Tomoya Hayata, Yoshimasa Hidaka, and Yuta Kikuchi, arXiv, January 2026 (12 pages).
- [3232] [Towards Quantum Advantage in Sparsified Bosonic SYK Models](#), by Vaibhav Gautam, Atsushi Matsuo, and Masahito Yamazaki, arXiv, December 2025 (10 pages).
- [3233] [Digital quantum simulation of quantum gravitational entanglement with IBM quantum computers](#), by Carlos Sabín, arXiv, February 2023 (11 pages).
- [3234] [Towards a variational Jordan–Lee–Preskill quantum algorithm](#), by Junyu Liu, December 2022 (30 pages).
- [3235] [Optimized Quantum Simulation Algorithms for Scalar Quantum Field Theories](#), by Andrew Hardy, Priyanka Mukhopadhyay, M. Sohaib Alam, Robert Konik, Layla Hormozi, Eleanor Rieffel, Stuart Hadfield, João Barata, Raju Venugopalan, Dmitri E. Kharzeev, and Nathan Wiebe, arXiv, July 2024 (50 pages).

This **Science** subsection contains 101 bibliographical references and notes containing at least 6,152 pages.

3.10 Software and tools vendors

There are already many quantum software and developer-tool startups, particularly with regard to what suitable hardware is available (Figure 205). Initially, many of them were developing software running on D-Wave annealers. Then, as gate-based vendors like IBM started to put their hardware in the cloud, most software vendors adopted it. Many software vendors adopt hybrid approaches that combine business knowledge, associated algorithms and their execution on classical machines and quantum computers, hybrid classical-quantum algorithms, or so-called “quantum inspired” algorithms that run on classical computers. Only a few software vendors have adopted the quantum simulation paradigm which is a pity given these systems may be the most viable in the mid-term.

Even though quantum software won’t solve all business and technical problems, it is time for legacy software vendors to examine the value it could provide [3236]. Funding, headcount, products and partnership in this inventory are dated from September 2026 and may be outdated as you read the book. Funding amounts are cumulative disclosed amounts which sometimes includes public subsidies.

These approaches are essential for survival. Indeed, a startup cannot be exclusively dedicated to quantum computing at the risk of only being able to sell proofs of concept on a very small scale that cannot generally be deployed industrially [3237].

There are real opportunities to position yourself in this emerging market! You will notice that this inventory does not include any Chinese startup. This directory is not exhaustive and is built from publicly visible, mostly English-language sources, so this absence says more about its scope than about the Chinese ecosystem. This ecosystem is still very young. It will evolve in parallel with the development of commercial quantum computers. You see already the startup scene maturing with many vendors adopting platform approaches and developing partnership models all over the place [3238].

1QBit

Canada, 2012-2024, \$35M



1QBit is a multi-sector quantum software company. It was funded among others by Fujitsu, as well as by Accenture and Allianz and was acquired in 2024 by Terra Quantum AG (Switzerland).

They developed various low-level quantum algorithmic components that are hardware neutral. This includes, for example, the graph processing that they apply in several markets, via their consulting activity. They cover financial markets, for the dynamic optimization of investment portfolios or to simplify the allocation of asset classes in a portfolio [2913]. They also developed QEMIST, a library for accelerating innovation in materials science and drug discovery. In addition to being a

long-standing partner of D-Wave, they work with IBM and Fujitsu. Their customers include Dow Chemical (chemicals), Biogen (biotechnology) and Allianz. They are now focused on quantum AI.

1Qbit also developed TopQAD (Topological Quantum Architecture Design), a software toolkit that turns a quantum algorithm into primary quantum gates, schedules operations in a circuit, contains a FTQC emulator and an assembler used to design the quantum modules required for executing the input circuit at the physical level. They also work on quantum FTQC error correction codes and architectures [3239]. Since 2025, TopQAD anchors 1QBit’s contribution to the HPE-led consortium selected by DARPA for stage 1 of its Quantum Benchmarking Initiative (QBI), where 1QBit handles fault-tolerant QEC design, algorithm compilation and automated resource estimation alongside Qolab (superconducting qubits), Synopsys (EDA), Quantum Machines (control and real-time QEC) and the University of Wisconsin (algorithms and benchmarks).

Adaptive Finance Technologies

Canada, 2020



Adaptive Finance Technologies came out of the Creative Destruction Lab. It was created by Roman Lutsiv, Vlad Anisimov and Edward Tang and develops investment and credit risk management software. They used classical machine learning methods and are prototyping quantum machine learning solutions running on D-Wave annealers.

AGI Quantum

Sri Lanka, 2020



AGI Quantum is a company created by Ishara Mahanama that is exploring ideas around how AI and quantum systems could combine to accelerate discovery.

AIQTECH

Canada, 2018



AIQTECH is a machine learning specialist that explores the uses of QML, but currently seems entirely focused on classical AI. They are partners of the IBM Quantum Network. It is a two-person shop created by Farzad Qassemi (CEO) and Felix Motzoi.

Algorithmiq

Italy, 2020, €36M



Algorithmiq is a spin-off from the University of Turku created by Sabrina Maniscalco (CEO), Guillermo García-Pérez (CSO), Matteo Rossi (CTO) and Boris Sokolov, which develops quantum software for life sciences and data science, particularly on drug discovery and genomics, as part of its Aurora platform [3240, 3241]. The company had a headcount of over 40 as of mid-2025. It moved its headquarters from Helsinki to Milan in May



Figure 205: Map of the quantum software industry. Inclusion is based on publicly visible activity in quantum software and tools as of September 2026, and the map is not exhaustive. Companies whose logos are available in vector format (SVG) are reproduced at better quality. © Olivier Ezratty, September 2026.

2026 when it raised €18M led by United Ventures and Cassa Depositi e Prestiti (CDP) with Inventure VC, Italy's largest quantum venture round to date, bringing total funding to €36M. It retains operations in Finland, the UK, Ireland and the USA.

Algorithmiq is a partner of IBM, and also works with Google, Microsoft, AWS and Rigetti, having signed commercial agreements with the last three in 2025. In December 2023, they announced they had run a large quantum error mitigation experiment on IBM's 127-qubit QPUs, using 68 qubits and 98 layers of CNOT gates, with a total of 2,402 CNOT gates. For this, they developed their own tensor-network-based QEM (TEM). They then used 91 qubits in a many-body quantum chaos simulation [3242]. For Cleveland Clinic, they developed custom versions of ADAPT-VQE to simulate the dynamics of boron-dipyrromethene (BODIPY), a class of organic fluorescent dyes that is used in photodynamic therapy (PDT) as a photosensitizer [3243, 3244]. They also partner with Quantum Circuits, also in the field of chemical simulations and drug discovery. They work with AstraZeneca and the Hartree Centre (UK) on the study of proton transfer reactions. They also created a simulation of photon drug interactions in cancer prevention and treatment with the Cleveland Clinic [3245], as part of a \$4.8M project funded by the Wellcome Leap initiative from the Wellcome Trust.

In 2025, they disclosed the Majorana Propagation classical method to simulate fermionic circuits which is inspired by the Pauli propagation method coauthored by Zoe Holmes [3246]. This method seems to be more efficient than tensor-based circuit simulation methods, and is useful for quantum chemistry applications. This has no connection with the Microsoft Majorana qubits. They published in 2026 a paper on the realization of quantum dynamics simulations [3247].

In 2026 they introduced DQI (Digital Quantum Interface), a platform connecting quantum computers with classical systems, which contains a measurement toolbox to reduce computing errors, an error mitigation toolbox and a circuit optimization toolbox.

They also created an online learning platform for quantum science and technology, the QPlayLearn project.

And with IBM, they launched the Quantum Advantage Tracker on GitHub to consolidate quantum advantage claims [3248].

Allosteric Bioscience

USA, 2021, \$920K



Allosteric Bioscience, Inc

Allosteric Bioscience is a company integrating quantum computing and AI with biomedical sciences to create improved treatments for aging and longevity. In February 2022, the company announced it was teaming up with and investing in Polaris Quantum Biotech. They first work on creating an inhibitor of a key protein involved in aging.

Alysophil

France, 2018



Alysophil®

Alysophil is a company creating new molecules for the chemical industry using AI. They are testing quantum machine learning techniques and partnering with Qandela [3249].

Ankh.1

USA, 2018-202X



Ankh .1

Ankh.1 is a company created by Oleg Baranov which developed Anubis Cloud, a virtual machine in the cloud for data scientists that integrates with the open-source solution Jupyter as well as with the TensorFlow and Keras learning machine frameworks. The company seems to have ceased operations.

Anzaetek

South Korea, 2023



Anzaetek is a quantum software company created by Renaud Béchade, which develops QML and quantum inspired solutions for hospitals. Their offer includes QML//QFL Explorer, which combines a multimodal data embedding and a federated learning tool, running in an emulator.

Apply Science

Italy, 2019



Apply Science is an applied mathematics services company that is experimenting with quantum computing in the IBM Quantum Network. They are testing QML algorithms for virtual drug development.

They targeted the two main markets: finance and chemical simulation, in addition to generic optimization problems and predictive analysis. They were developing both classical/quantum hybrid computing and quantum machine learning solutions. The company stopped operating in March 2021.

ApexQubit

Belarus and USA, 2018-2023



APEXQUBIT

ApexQubit was a drug-discovery company that develops quantum software solutions for the pharmaceutical sector targeting rare diseases. They operated in project mode and publish some research papers on their website. It ceased operations in 2023.

Aqemia

France-UK, 2019, \$101M



Aqemia is a software company developing drug discovery and retargeting algorithms using statistical physics, AI and quantum inspired algorithms, packaged in its Qemi platform. The startup is a spin-off from Ecole Normale Supérieure run by Maximilien Levesque (CEO) and Emmanuelle Martiano. In December 2020, they announced

a partnership with Sanofi to discover new treatments for COVID-19. They raised €30M in October 2022, then a further €30M in January 2024 led by Wendel Growth with Bpifrance Large Venture, Eurazeo and Elaia, bringing their Series A to €60M and cumulative funding past \$100M by early 2025, when they opened a London office. This funds their AI-enabled drug discovery pipeline, not their quantum-based algorithms efforts. In December 2023, they signed a Sanofi collaboration worth up to \$140M in upfront and milestone payments, expanded in January 2026 and again in July 2026 with a new therapeutic target. They also run an immuno-oncology discovery collaboration with Servier.

aQuantum

Spain, 2018



aQuantum is a quantum-software engineering services company doing contract research, development and consulting. They develop hybrid classical-quantum computing software and provide quantum software project management expertise, particularly in quantum machine learning.

They also developed $|QuantumPath\rangle$ (aka $Q|Path\rangle$), a quantum software development and lifecycle application platform. It contains all the tools to handle the whole software design and execution lifecycle covering both gate-based, quantum simulators and quantum annealing based computing, supporting Qiskit (IBM), Forest (Rigetti), Ocean (D Wave), ProjectQ and Quantinuum's $|tKet\rangle$. They embed the open-source Quirk graphical tool in their development environment. Since April 2022, QuantumPath is integrated with AWS and Amazon Braket making it possible for developers to access the quantum emulation and quantum hardware platforms supported by Amazon Braket.

Launched in 2023, QAgnostic QAOA extends the software portability from $|QuantumPath\rangle$ to run QAOA algorithms to solve optimization problems. It is possible to launch the same optimization algorithms on QPUs from different providers as well as various quantum computing emulators while supporting gate-based and annealing paradigms.

A*Quantum

Japan, 2018, \$3M



A*Quantum specializes in the development of quantum software solutions for both annealers (including digital annealers from Fujitsu) and gate-based quantum computers (from IBM). Their ambition is to create high-level software libraries for users.

Aqarios Quantum

Germany, 2022



Aqarios Quantum is specialized in quantum machine learning applied to the financial sector and in logistics [3250]. Its AqOPT platform is used to solve com-

plex optimization problems, mainly with quantum annealing, with a tool to transform problems into Ising model/QUBO form, a solver recommendation tool, and a tool to analyze a potential quantum advantage. In November 2024, they released Luna, an application platform with 40 optimization scenarios, >30 specialized algorithms, and a user-friendly interface. The company also works with E.ON on energy grid optimizations [3251]. They partner with Q-CTRL and Quantware. The company is traded on the Dusseldorf stock exchange since July 2026.

Arclight Quantum Computing

China, 2020, \$3M



Arclight Quantum Computing is a quantum software company spun out of the Institute of Software Research of the Chinese Academy of Sciences. With CAS, they developed isQ-Core, a quantum cloud software development platform that can execute gate-based quantum code on both classical emulators and QPU, like for the analysis of strongly correlated materials [3252].

They also created an EDA (electronic design automation) tool for the creation of quantum processors.

Arline

Germany, 2020



Arline was developing a compiler optimizing QML algorithms execution, reducing the number of quantum gates used and taking into account all qubit characteristics such as their connectivity. They also proposed Arline Benchmarks, an automated benchmarking platform for quantum compilers. It compares gate count, circuit depth and compiler runtime. It can also be used to combine compiled circuits and optimization routines coming from different compilers in a custom pipeline to optimize algorithms performance. The company is dormant.

Artificial Brain

India/USA, 2022

ARTIFICIAL BRAIN

Artificial Brain is developing hybrid quantum-classical algorithms targeting energy, aviation, finance and climate change use cases. Its first achievement was an algorithm identifying optimal locations for electric vehicle charging. It is running on D-Wave quantum annealers (probably from the Pegasus generation) using a mix of quantum annealing and genetic algorithm developed on Qbsolv [3253]. It produces a result in 3 seconds for a search space of 8.5×10^{15} combinations. The cardinality of the search space alone does not demonstrate any speedup. Its scalability depends on how D-Wave will expand the capacity of its annealers in the future. The company was founded by Jitesh Lalwani, who worked beforehand in the software industry.

Artiste-qb.net

Canada, 2018



Artiste-qb.net has a business model similar to that of 1Qbit: they develop algorithmic patterns of intermediate levels that they then assemble according to the needs of their customers. The startup was created by an international team including German researchers. They develop a Python-based set of libraries in open-source, available on GitHub. As of 2026, the company appeared to be dormant.

Automatski

India, 2014



Automatski is a software company established in London, India in Bangalore and in California. They do applied contract research to develop quantum algorithms on any form of computer and quantum simulator.

They claim the creation of a 300-qubit classical emulator in full vector state [3254]. They focus on creating biochemistry algorithms and claim to have solved protein folding and to cure diabetes, cancers and 4,000 other diseases. They also claim to have created a quantum annealer with a “quintillion qubit” based on some “new physics”, probably some photonic coherent Ising machine. They combine huge overselling and trade secrets. No accessible paper, benchmark, product documentation, customer validation or reproducible comparison substantiates any of these claims.

Avanetix

Germany, 2019



Avanetix develops hybrid algorithms dedicated to solving supply chain problems. They combine classical optimization methods, machine learning and quantum computing. They target the automotive and logistics markets. The startup is founded and managed by serial entrepreneur Naimah Schütter.

Beit.tec

Poland, 2017, \$10M



Beit.tech is specialized in quantum machine learning. It is mainly a research project funded by the European Union, covering the period 2017-2020. The three founders including Wojtek Burkot are former Google employees. They create algorithms for optimizing complex graphs that run on classical computers, targeting the problem class addressed by D-Wave annealers.

Beyond Limits

USA, 2014



Beyond Limits is an AI software company focused on reasoning applications and autonomous decision-making with a first product targeting refineries in the oil industry. They announced some partnership with IQM to

work on quantum AI technology in the APAC region around Singapore.

Bloq Quantum

India, 2024, \$155K



Bloq Quantum is a startup created by Sreekuttan L S (CEO) and Jay Patel (CTO) which develops a “low code” (or even “no code”) quantum computing platform based on an intuitive interface and as a subscription-based service. It was released in 2026 with a drag-and-drop user interface which connects to different SDKs and quantum computing hardware. The company targets primarily the financial and healthcare sectors. It also sells consulting services for the development of custom quantum solutions.

Blueqat

Japan, 2008, \$2.3M



Blueqat, formerly MDR for Machine learning and Dynamics Research, is creating algorithms integrating AI and chemistry, working, among others, with customers from the cosmetics industry like KOSÉ.

They are working with D-Wave annealers. The startup was founded by Yuichiro Minato and various other alumni of the University of Tokyo.

Blueqat developed an algorithm that analyzes the distribution of cosmetics product features in a multidimensional space. It visualizes existing areas and reveals unknown product areas they were able to open up, to create possibilities for new cosmetic designs that humans never thought of. The solution was patented and thus, is not yet publicly documented.

BlueQubit

USA, 2022, \$10M



BlueQubit released in 2023 a software platform to access both classical emulators of quantum code with up to 34 qubits and quantum processing units, particularly focused on quantum machine learning applications. Positioned as a QSaaS framework, for Quantum Software as a Service, it supports the Qiskit and Cirq frameworks as well as Nvidia classical infrastructures. The San Francisco-based startup was created by alumni from Google, QC Ware, Twitter, Caltech, MIT and Stanford University.

They collaborate with QuEra and won a project with DARPA in October 2023 as part of the IMPAQT program. They also work with Quantum Art in Israel as well as with Honda in Japan for image recognition. In October 2025, they published a preprint about a new quantum advantage experiment using “random peaked circuits” that are easier to verify classically without some exponential cost. It was run on a 56-qubit Quantinuum H2 QPU, using 2,000 qubit gates [2246]. But the algorithm doesn't seem to have a direct practical use case.

In April 2026, IBM researchers demonstrated that these circuits could be classically simulated efficiently [2247].

Boltz.ai

Canada, 2020



Boltz.ai is specialized in the development of AI and quantum software for the agriculture business. They create crop field allocation optimization tools.

Boston Quantum

USA, 2022-2025



Boston Quantum was building quantum enterprise software for the financial industry, with an initial focus in foreign currency exchange and crypto-currencies arbitrage detection. It used quantum and physics-inspired algorithms. It seemed to be actually quantum inspired solutions.

BosonQ Psi

India, 2020



BosonQ Psi (*aka* BQP) created the cloud-based BQ-Phy software suite, a computer-aided engineering (CAE) solution performing complex simulations using hybrid quantum-classical algorithms. It covers structural mechanics, thermal analysis and design optimization. The startup was created by Abhishek Chopra, Rut Lineswala and Jash Minocha. They make various claims of $\times 10$ to $\times 1000$ performance gains with their simulations, but the comparison points are not clear (classical vs classical or else). They work with the AFRL in the USA in the field of simulation using hybrid quantum-classical computing. They are partnering with Artificial Brain, another startup from India. They also work on solving computational fluid dynamics problems [3255] and Poisson's equation [1169]. In September 2026, IBM Ventures invested an undisclosed amount in the startup.

Boxcat

Canada, 2017



Boxcat is a startup created by Ystallonne Alves that develops image and video processing solutions based on hybrid quantum algorithms (Figure 206). They target the media and medical imaging markets. Their algorithms are based on currently available hardware architectures (D-Wave, IBM, Rigetti, Xanadu). The process they present on their site is an image realized on a D-Wave, which could have been realized with Nvidia's latest GPUs.

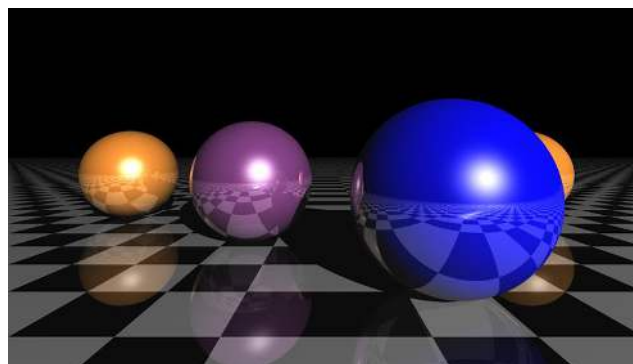


Figure 206: An artificial image generated on a D-Wave 2000Q by Boxcat. Source: Boxcat.

Cambridge Quantum Computing Limited

UK, 2015-2021, \$72.8M



Cambridge Quantum Computing developed the $t|ket\rangle$ quantum operating system and various quantum algorithms including Arrow for machine learning, before merging into Quantinuum in 2021 [3256].

They were partnering with OQC and with IBM which was one of their investors. CQC was also active in post-quantum cryptography. $t|ket\rangle$ is available broadly and for free since February 2021 and in open-source. It covers many quantum computing platforms (IonQ, Honeywell, AQT, IBM Qiskit, Rigetti, Amazon Braket and Azure Quantum) and incorporates circuit optimization and routing. It is interfaced with Python with Pytket. $t|ket\rangle$ is also used by EUMEN, CQC's quantum computational chemistry platform, and the company's QML framework. CQC also partnered with Roche to use quantum algorithms for drug discovery targeting Alzheimer's disease, as announced in January 2021.

In 2021, CQC launched a cloud software random quantum number generator. It is using a classical random generator, a quantum random number generator amplifying the randomness of the first and a Bell test used to check the resulting randomness, then running on an IBM quantum system [3257, 3258].

CQC demonstrated how NLP (Natural Language Processing) could be implemented on NISQ quantum computers. Their researchers explain that the structure of natural language is natively quantum, which could lead to efficient translating, and better understanding of complete sentences and texts. The paper, however, didn't provide much indication on the way data was encoded in the qubits. It lacks supporting data on actual system performance [3259].

In October 2021, this was packaged in an open-source Quantum Natural Language Processing (QNLP) toolkit and library, lambeq, which was then used in 2024 in the QDisCoCirc question-answering model to create interpretable models [3260, 3261].

The company merged with Honeywell Quantum Systems in 2021, to become Quantinuum in December 2021, with a total staff of over 350 employees.

Cascade Quantum

USA, 2023



Cascade Quantum Inc, formerly Coherent Computing Inc, is a software company created in Oregon by Sonika Johri, an alumna of IonQ. She built a full stack hardware independent quantum software platform including some high-level abstraction layer. It targets multiple use cases, such as in life sciences [3262]. She also works on “scalable quantum machine learning” that is tested with up to 16 qubits [3263]. The company published a preprint on benchmarking QML in oncology applications in August 2026 [3264], but again, without exceeding 16 qubits in the approach, which is very limiting. They launched Red Cedar in 2026, a modular QML platform.

ChemAlive

Switzerland, 2014



ChemAlive is a quantum computational chemistry startup and contract research company. It provides simulation tools for getting molecular properties and synthetic-reaction planning using basic 2D chemical syntax.

They deal with reaction mechanism elucidation and optimization, kinetic observed rate modeling, molecular design, virtual screening and drug discovery, molecular and spectroscopic property prediction, materials modeling and design, data and computation driven synthetic planning, experimental execution of chemical synthesis and experimental research on spectroscopic and electronic molecular properties. They developed Construkt, a software tool transforming molecular drawings into 3D structures and energies.

All of this is quantum... but seems to be computed classically. Quantum chemistry doesn't necessarily mean quantum-computed quantum chemistry. Once quantum-computing hardware scales, they will naturally switch to it.

Classiq Technologies

Israel, 2020, \$204M



Classiq Technologies was created by Nir Minerbi (CEO), Amir Naveh (VP-R&D) and Yehuda Naveh (CTO, who spent 20 years at IBM Research in Haifa, including quantum, condensed matter specialist).

The company's product is Qmod, a quantum programming tool providing a higher level of abstraction than classical quantum gate programming [3265]. It uses an equivalent of software gate patterns in classical programming and supports declarative programming. It is on Amazon Braket as announced in 2022. Classiq's compiler is also inspired by techniques used in EDA (Electronic Design

Automation) tools [3266]. For a long time in beta-test, Classiq 1.0 was finally released in 2026.

In May 2025, Classiq raised a \$110M Series C round, at the time the largest round ever for a quantum software company. It was extended in November 2025 with a \$30M, bringing total funding to about \$204M. The company had a headcount of about 130 as of mid-2026.

They have announced various partnerships, including one with Florence Quantum Labs on precision agriculture, using quantum-enhanced biosensors and AI-quantum hybrid algorithms, and another with Sumitomo Corporation and Mizuho-DL Financial Technology to compress quantum circuits used in Monte Carlo simulations for financial risk analysis. Their referenced enterprise customers include BMW, Citi, Deloitte, HSBC, Mizuho, Rolls-Royce and Toshiba.

CogniFrame

Canada, 2016



CogniFrame (2016, Canada) is a software publisher of data analysis platform software exploiting machine learning. They also develop hybrid algorithms for the financial sector based on D-Wave annealers.

One of their first customers is the Canadian investment bank Alterna Savings. The proposed applications are classic in the financial field: credit risk assessment and investment portfolio optimization. Late 2021, they launched **FirstQ Store**, an aggregation platform of quantum computing applications, provided as an application running on Windows, Mac and Linux desktops. It supports QUBO applications for Toshiba's Simulated Bifurcation Machine (a sort of digital annealer) and gate-based algorithms [3267]. In April 2023, the company announced a service partnership with KPMG's Global Quantum Hub.

ColibriTD

France, 2019, €9M



ColibriTD is a quantum software R&D company created by Hacène Goudjil and Laurent Guiraud that develops vertical use case software components with a team of a dozen skilled PhDs/post-docs, focused on gate-based computing models. They were 20 as of August 2025.

They designed H-DES, a hybrid differential equation solver and the Python library MPQP (Multi-Platform Quantum Programming) which makes it easier to design quantum circuits and supports multiple emulators and QPU backends through automatic library code translation. Both tools are integrated in the QUICK platform (Quantum Innovative Computing Kit), which consolidates templates to solve problems in the financial, pharmaceutical, material design and logistic fields.

They also contribute to the advancement of quantum software engineering tools like in functional testing

[3268] and in methods to solve PDE (partial differential equations) including on NISQ systems [3269].

Commutator Studios [commutator.studios]
Germany, 2022, €1.5M

Commutator Studios is a company cofounded by César A. Rodríguez Rosario (formerly chief scientist at Strangeworks, and a specialist in non-Markovian effects [3270–3272]) and Jan M. Knaup that develops a machine-learning driven, quantum error management platform, seemingly implementing some forms of quantum error mitigation in NISQ regime. The company claims that its software generates x20 speedups and improves results accuracy, but the comparison benchmark is not clear. Their website does not provide any technical details on what their software solution is actually doing.

Conductor Quantum
USA, 2024



Conductor Quantum is a company created by Brandon Severin and Joel Pendleton that builds AI systems for operating and scaling quantum computers. They created Coda, a LLM-based language interface for quantum computing that generates a quantum circuit, simulates it on classical hardware, executes it on quantum hardware, and returns the results.

CreativeQuantum CreativeQuantum
Germany, 2010

CreativeQuantum is specialized in quantum physics-based R&D in the chemical and pharmaceutical industries.

They seem, however, to run these many algorithms with classical computers. Which makes sense given quantum computers are not yet powerful enough to run these physics simulation tasks efficiently.

Culgi 
the Netherlands, 2004-2020

Culgi is a computational chemistry company that will someday adopt quantum computing or simulation for its software. It was founded in 1999, changed its name in 2004 and was acquired by Siemens in 2020.

Dark Qore 
the Netherlands, 2024

Dark Qore is building tools to reduce the resource cost of FTQC circuits through hardware-aware compilation and resource estimation. The company was created by Nikiforos Paraskevopoulos (CEO), who was previously doing his PhD in the Sebastian Feld quantum machine learning group at TU Delft [3273]. Dark Qore made some quantitative modeling supporting the scaling case for QuantWare's VIO QPU architecture announcement in 2025 [3274].

Dihuni
USA



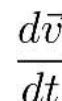
Dihuni is a service company created by Indian developers working in the AI, cloud and Internet of Things domains. It created Qubrid, a QML platform that is supposed to support multiple frameworks and classical plus quantum backends hardware. It supports Qiskit. The company looks like a one-person company run by its CEO Pranay Prakash.

Dirac
USA, 2021



Dirac was developing quantum software and algorithms for robotic applications. That New York City startup was created by Filip Aronshtein. The company name makes search engine optimization difficult. No public paper, benchmark or product documentation is available to assess the technology. It seems to have pivoted out of quantum computing.

dividiti
UK, 2014-2024



dividiti developed quantum algorithms, particularly in machine learning and using hybrid methods. Their solutions were open-source. It was based on a service model, which is rather the standard in this market at the moment.

EdenCode 
USA, 2025, \$1.3M

EdenCode is led by Wanda Hou (CEO), Yi-Zhuang You (CTO), and Hong-Ye Hu. The company develops a real-time neural decoder for QEC, competing with Riverlane. They claim sub-ms decoding latency, a 99.9% error-detection rate, processing up to 10× faster than conventional approaches, and up to a 10× reduction in logical error rate (vs what?). In April 2026, they released a Graph Transformer Decoder, described as a foundational model for QEC in which a single network decodes surface codes across all code distances. They also contributed to a QEC and QEM benchmarking tool creation [3275]. The founder's past work involves classical LLMs [3276, 3277] and the use of machine learning in quantum measurement [3278].

Elyah
Japan/Dubai, 2018



Elyah is developing quantum software to “improve people's lives”. The company is made up of two people, Salman Al Jimeely, based in Dubai, and Sydney Andrew, based in Tokyo. No public technical documentation is available. The company appear to be dormant.

Entanglement

USA, 2017



Entanglement is a quantum software development service company. One of their achievements was to create a quantum inspired software for vaccine distribution optimization in the USA in 2021.

Entropica Labs

Singapore, 2018, \$12M



Entropica Labs was founded by Tommaso Demarie (CEO), Ewan Munro (CTO), joined in 2018 by Joaquin Keller, a former Orange researcher based in France, who left them to later create **exaQ.ai**. It was initially dedicated to the creation of quantum (and non-quantum) algorithms for life sciences and in particular for genomics, based on quantum machine learning. The result is faster development of therapies, in partnership with pharmaceutical companies. It then offered its Entropy Development Framework that manages the workflow of quantum software. They are working with Honeywell/Quantinuum and BMW, and the Singapore Defence Science and Technology Agency to create proof of concepts for supply chain optimization [3279].

They pivoted into providing software tools for FTQC and QEC, with an integrated stack support of the full QEC lifecycle. They are building Quilt, a Fault-Tolerance Operating System (FTOS) that packages fault-tolerant programs into instructions a control stack can execute, along with Loom, a hardware-aware compiler that converts quantum programs into fault-tolerance-ready circuits, and the Entwine and EKA code design and simulation tools. The startup collaborates with the Singapore academic ecosystem, including CQT, notably on real-time error decoding [3280]. Their partners include Xanadu (through PennyLane and Catalyst), Quantum Machines, Yaqumo and qBraid, whose Lab hosts Loom, plus a February 2026 MoU with Quobly (France) on silicon spin qubit fault-tolerance co-design. Chad Rigetti joined their board in November 2024. Their disclosed funding is closer to \$7.4M than the \$12M shown above, including a \$4.7M Series A in November 2023, extended to \$5.5M in September 2024.

exaQ.ai

Singapore, 2020



exaQ.ai is a quantum-machine-learning company created by Joaquin Keller, who formerly co-founded Entropica Labs and was a teacher in France and a R&D lead at Orange.

Their offering is based on the “polyadic QML Library” that implements supervised quantum machine learning for multi-class classification on NISQ architectures [3281]. It was tested on IBM Quantum hardware with accuracy levels similar to classical machine learning, doing a ternary classification of the [Iris flower dataset](#)^w that

contains only 150 objects to classify in three classes. They also provide their ManyQ quantum computer emulator that is optimized for quantum machine learning and supports CPUs and GPUs.

FACCTs

Germany, 2016



FACCTs is a spin-off from Max Planck Institute for Chemical Energy Conversion that develops ORCA, a quantum-chemical software package.

Stafford Computing

USA, 2024



Stafford Computing provides services in quantum machine learning. It also developed the quantum inspired methods based FalconDale Viable Quantum System Kit (VQSK) targeting the financial sector. They published some work on datasets loading in 2026 [3282].

FAR Biotech

USA, 2016



FAR Biotech does drug discovery based on quantum representation of molecular structures done, so far, on classical computing.

finQbit

Poland, 2023, \$27K



finQbit is a quantum software company focused on financial services and risk management.

Fixstars Amplify

Japan, 2023



Fixstars Amplify provides a cloud-based platform that includes an SDK and an execution environment for D-Wave quantum annealers, Toshiba, Fujitsu and Hitachi digital annealers and IBM QPUs. They also partner with Qudora (Germany).

Their SDK is compatible with its own annealing machine, Fixstars Amplify AE, and all available quantum annealing/Ising solvers, mathematical optimization solvers, and gate-based quantum computers like the ones from IBM. On top of that, they developed Fixstars Amplify Benchmark, a framework for benchmarking the performance of solvers for QUBO problems which provides a command-line interface to perform benchmarking and a definition of the benchmark problem. In November 2024, they announced a joint project with RIKEN (Akira Furusawa) and NTT (Hidehiro Yonezawa) for the creation of a cloud accessible general-purpose continuous-variables and time-division multiplexing based optical QPU.

First Quantum

South Korea, 2023



First Quantum is a company created by Suk Whan Chang (CEO) and Doyeol Ahn (CTO) that is developing quantum circuits and using algorithms optimization based on its proprietary and patented quantum Kar-nough Map technique, supporting Qiskit. The technique simplifies circuits, particularly based on Toffoli gates [3283]. They cover QFT, fluid dynamics, quantum chemistry and financial services.

FirstQFM

Sweden, 2025, €1.2M

FirstQFM

FirstQFM was created by Visweswaran Vish R (CEO, also cofounder and CEO of CogniFrame in Canada) and Isaiah Hull (CTO). It is a software company which states that it has a “*patent pending quantum foundation model technology that can help improve performance and scalability across multiple layers of the quantum computing stack*”. The company has not published the model description, the supported task, the metric, the baseline or the patent application number, so the claim cannot be assessed.

Fokus

Canada, 2021

FOCUS

Fokus is a spin-out from University of Waterloo, led by Michele Mosca and Sadegh Raeisi which creates a software solution to improve the performance of MRI and NMR systems with machine learning and quantum computing. Quantonation is one of its investors.

GenMat

USA-Canada, 2021-2024, \$15M


genmat

GenMat aka Quantum Generative Materials is a stealth quantum computing software startup. They received \$15M in funding (plus \$35M following development milestones) from **Comstock Mining** (USA), a gold and silver mining company also invested in lithium-ion battery manufacturing. In October 2024, Comstock acquired the whole company.

In March 2023, they announced ZENO, a generative artificial intelligence that simulates known and new materials more efficiently and computes electrical and thermal conductivity, heat capacity, local and charged density of states, band gap, formation energies, magnetic properties, and more. For the time being, this is still classical computation.

Global Data Quantum

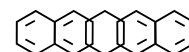
Spain, 2025

Global

Global Data Quantum is a software company working at the crossroads of data, AI and quantum computing. It provides a quantum portfolio optimizer that is available in Qiskit Functions catalog.

Good Chemistry

Canada, 2021



GOOD CHEMISTRY

Good Chemistry is a spin-off from 1Qbit which developed QEMIST Cloud, a cloud developers platform associating classical machine learning and quantum computing to undertake computational chemistry simulations. Accenture is an investor in the company. It was acquired by SandboxAQ in 2024.

Grid

Japan, 2009

GRID INFRASTRUCTURE+
LIFE + INNOVATION

Grid is a specialist in deep learning and learning by reinforcement with their ReNom platform.

They have adapted this library in a quantum version called ReNomQ. And they have been IBM Quantum partners since September 2019. On the other hand, their AI was probably not very efficient to help them find a company name that is easy to be found via search engines.

Groovenauts

Japan, 2012, \$12M

 **Groovenauts**

Groovenauts developed in 2016 a D-Wave based cloud service called Magellan Blocks to solve complex optimization problems. They exploit hybrid algorithms combining machine learning and quantum algorithms. Their first customers include a Japanese retailer who optimizes its planning and Mitsubishi Estate who optimizes household waste collection.

Hafnium Labs

Denmark, 2016

 **Hafnium Labs**

Hafnium Labs develops software that provides physical property data for molecules and mixtures by combining quantum chemistry and AI. So, not yet a quantum software vendor.

Haiqu

USA, 2022, \$15M

HAIQU

Haiqu was cofounded by Mykola Maksymenko (CTO) and Richard Givan (CEO). It develops a platform-agnostic software to extend NISQ quantum hardware capability into applications in finance, chemistry, life sciences, mobility and other domains [3284].

Harmoniqs

USA, 2025, \$200K

 **Harmoniqs**

Harmoniqs is a startup from Chicago created by Andy Goldschmidt, Aaron Trowbridge, Jack Champagne and Gennadi Ryan. They developed a quantum control optimization software for quantum computers and analog quantum simulators. It is a suite of Julia-based packages, including software for quantum optimal control,

efficient trajectory storage, and trajectory optimization [3285].

Horizon Quantum Computing

Singapore, 2018, \$153.33M



Horizon Quantum Computing creates quantum development tools. Their ambition is to compile code from classical development tools such as Matlab, C, C++ and then optimize it and run it on quantum computers, to make quantum computing accessible to traditional developers. In short, they want to democratize quantum software development.

All this is in their Triple Alpha web-based IDE. They also work on quantum Internet software. As such, they announced in 2022 that they will establish a quantum communication on Singapore's National Quantum-Safe Network. The startup was launched by Joe Fitzsimons and Si-Hui Tan, both coming from Singapore's CQT research center [3286].

In November 2022, it opened an office in Dublin, Ireland, the first outside Singapore, which has since become its European headquarters. Not a big surprise given Joe Fitzsimons is Irish. In April 2024, it announced the installation of a Rigetti Novera 9-qubit QPU, coupled to an OPX1000 from Quantum Machines for qubit control, to test their compiler. This became Ember-1, their first superconducting testbed, inaugurated in January 2026. In April 2026, they announced the purchase of a 256-qubit sixth-generation IonQ trapped-ion system, located in Dublin as confirmed in June 2026, giving them two hardware modalities. They also run collaborations with AQT on trapped ions, with Alice & Bob on cat-qubit emulation inside Triple Alpha, and, since July 2026, with Quantum Machines on an embedded continuous calibration framework for Ember-1. Their programming language is named Beryllium.

In February 2025, the company announced its willingness to launch a SPAC-driven IPO at a valuation of \$500M. The business combination with dMY Squared closed in March 2026, generating nearly \$120M of gross proceeds, with the entity renamed Horizon Quantum Holdings Ltd and trading on Nasdaq under the ticker HQ since March 2026. By early August 2026, the exercise of 79% of the public warrants had added about \$28.7M. In August 2026, it released quarterly results reporting \$0 of revenue for the six-month period, consistent with a pre-revenue R&D stage. The operating loss and cash position for that filing period should be quoted from the filing itself.

HQS

Germany, 2017, €15M



HQS is a Karlsruhe Institute of Technology spin-out startup led by Michael Marthaler. They are developing quantum algorithms in the field of organic and inorganic molecular simulation of simple molecules (methane, light

emission in OLEDs, diffusion of molecules in liquids). It mostly targets the chemistry industry [3287].

They announced in July 2018 an open-source porting tool for ProjectQ code (IBM platform) to Cirq (Google platform). They released their Quantum Assisted Design toolbox in 2020 and qoco, a quantum circuit representation library in 2021. They already have BASF and Bosch as customers. In practice, they also develop classical versions of their algorithms, running on datacenters or supercomputers [3288].

Their latest €12M funding round in February 2022 was led by Quantonation, extended by an investment of €1M by Trumpf in June 2023. The company had a staff of 50 as of mid-2024.

IFF Technologies (2020, USA) is developing Polar+, a software tool for precise protein analysis, bundled in a drug discovery platform. It finds binding sites on a protein by comparing binding energies at every atom. It helps in the incorporation of subtler, electromagnetic effects between atoms. They seem to work on gate-based QAOA and also on a GBS (gaussian boson sampler).

Iceberg Quantum

Australia, 2024, \$8M



Iceberg Quantum is a quantum architecture company which designs fault-tolerant architectures based on qLDPC codes that reduce the hardware overhead for quantum error correction [3289]. It was created by Felix Thomsen (CEO), Samuel Smith (CTO), and Larry Cohen (CSO). It announced a technology partnership with Oxford Ionics in January 2025 (IonQ). In February 2026, the company published an qLDPC-based FTQC scheme that would reduce the resource estimates to break an RSA-2048 key with 100K physical qubits, with 99.9% fidelities and bounded non-local connectivity [3290]. They improved it in September 2026 [3291]. They also work with Diraq.

Ingenii

USA, 2021



Ingenii is a startup created by Christine Johnson (CEO) and Marko Djukic (CTO, coming from Purdue University) that wants to simplify the work of scientists using quantum computers.

They started by creating a Python package to submit data science jobs like quantum chemical simulations to quantum hardware like IonQ through Microsoft Azure for a starter. These algorithms are of course limited by the current capacities of existing quantum computers. The startup focuses on life and environmental science use cases, up to working on benchmarking and error mitigation, using machine learning. The startup was created out of the Duality accelerator in Chicago.

Innovatus

Singapore, 2018



Innovatus Q is a spin-off from the Centre for Quantum Technologies in Singapore. They work on hybrid quantum algorithms based on trapped ions and superconductors. It appears to be dormant as of 2026.

Jaynes Computing (2019, Canada) is creating a cloud-based solution based on some quantum machine learning (QML) in the supply chain market. They are supposed to use some NISQ hardware, without any details. The startup was created by German Alfaro and was spun out of the Creative Destruction Labs. It seems focused on classical AI.

InstaDeep

UK, 2015-2023, \$107M



InstaDeep is a machine learning software vendor who started to investigate the field of quantum machine learning in a team led by Tom Barrett. Their first announcement dealt with classical quantum simulations using machine learning [3292]. Still, they hired Jules Tilly in that team, who knows well the inner workings of both VQE and QPE quantum algorithms. The company was acquired by BioNTech in January 2023, the pharmaceutical company at the origin of the Pfizer RNA-based COVID-19 vaccine.

Inspiration-Q

Spain, 2020



Inspiration-Q is providing quantum-inspired software solutions in the areas of optimization, simulation, and machine learning, mostly in the financial and transportation sectors.

Intelligent Quantum Inception

China, 2026



The company cofounded by Liu Qingfeng and Hui Zhai is a joint venture between iFLYTEK and Liangyi Wanxiang with the goal to integrate AI and quantum technology. Liangyi Wanxiang is a spinout of Hui Zhai's atom quantum computing team at Tsinghua. They develop software and algorithms. Zhuifeng is a fast large-scale atom rearrangement algorithm aimed at controlling arrays at the 10K atoms scale [3293], and Bianque is an AI-based QEC decoder. They also develop AI-enabled quantum algorithms.

Jij

Japan, 2018, \$7.4M



Jij was created by researchers from the Tokyo Tech Institute of Technology. It develops software for quantum annealing, including OpenJij, an open-source framework for implementing Ising models to model particle interactions, built on D-Wave's QUBO APIs. They also work on QPINNs [441]. They are also partners of Microsoft Azure and work with Orca Computing.

JoS Quantum

Germany, 2018



JoS Quantum develops quantum software solutions for the financial services industry, particularly in risk management and fraud detection. They also do contract research.

Ketita Labs

Estonia, 2018-2022



Ketita Labs was developing unspecified quantum software for NISQ computers, and for good reason. It was a university spin-off.

Kipu Quantum

Germany, 2021, €13.5M



Kipu Quantum was cofounded by Enrique Solano (co-CEO since 2026 and technology advisor to the board since July 2026), Daniel Volz (CEO until December 2025) and Tobias Grab (Chief Strategist Officer, who became CEO in July 2026). They have teams based in Berlin, Basel, and in Bizkaia (Spain). The startup's goals are to “*design and manufacture of modular and co-designed quantum computers*” tailored to solve specific tasks with NISQ, without waiting for FTQC generations. It is about reducing the footprint of NISQ algorithms, to enable them to deliver better value to customers in various fields. The company is quite prolific in the creation of new or improved quantum programming paradigms.

The created a wealth of computing approaches for that respect:

- NISQA, a paradigm announced in 2021 that merges digitized-counterdiabatic quantum computing (DCQC) and digital-analog QC (DAQC) for NISQ computers. It uses digital and analog compression techniques to reduce the number of physical qubits required to solve specific industry problems (combinatorial and optimization problems, chemistry, QML, ...).
- BF-DCQO (“bias-field digitized counterdiabatic quantum optimization”), a gate-based approach launched in 2024-2025, tested on an IBM Heron 156-qubit QPU that was supposed to fare better than D-Wave annealers and classical Tabu Search for solving a combinatorial problem reformulated into a HUBO problem [3294, 3295]. In September 2025, a D-Wave team countered this and exhibited better results with their recent Advantage 2 4,400 qubits annealer both in terms of results quality and computing time [3296]. Moreover, the D-Wave authors claimed that the contribution quantum part of BF-DCQO was minimal to the performance of the gate-based approach. Kipu updated its views on counterdiabatic quantum computing in 2025, detailing the conditions

of obtain some quantum advantage with their “Iskay” BF-DCQO algorithm on an IBM Heron 156 qubit system [3297], and similarly with DAQC [3298]. A team in Poland led by Quantumz.io questioned BF-DCQO’s runtime advantage in October 2025 [3299]. In April 2025, Kipu published a preprint on the use of BF-DCQO for folding proteins with 14 to 16 amino acids with a 64-qubit IonQ system, far from any quantum advantage regime [2629].

- HSQC (Hybrid Sequential Quantum Computing), a paradigm introduced in 2025 to solve combinatorial optimization problems that associates classical simulated annealing (SA), memetic Tabu search (MTS) and BF-DCQO [3300]. They even proposed to use BF-DCQC to add some reasoning capabilities to LLMs with improved chain of thoughts prompting [3301] and even some energy efficiency improvements, again, using BF-DCQO, to solve HUBO problems used to capture collective, higher-order relationships among reasoning fragments [779]. It doesn’t mean, however, that a quantum algorithm is implementing some form of reasoning nor that the energy efficiency gain is significant in LLM inferences.
- DCOS (digitized counterdiabatic quantum sampling), in October 2025, a hybrid quantum-classical algorithm for efficient sampling from energy-based models, such as low-temperature Boltzmann distributions [3302]. It was tested to compute a higher-order spin-glass Hamiltonian using an IBM Heron 156 qubits processor.
- QE-MTS (quantum-enhanced memetic tabu search), in November 2025, a non-variational hybrid algorithm that solves the low-autocorrelation binary sequence (LABS) problems at the crossroads of combinatorial optimization and signal processing [3303]. The paper shows that seeding a classical memetic tabu search (MTS) with bitstrings from digitized counterdiabatic quantum optimization (DCQO, which is classically simulated) improves the exponential scaling base of the time-to-solution. Experimentally, it is however limited to fewer than 20 qubits, way below any potential quantum advantage.
- Quantum feature extraction which they implemented in October 2025 using a digital counterdiabatic approach (for some molecular toxicity classification and image recognition) [3304] and on an unspecified analog cold-atom computer [3305], and later, in April 2026, on some IonQ hardware [3306]. In February 2026, this became Rimap, a service enhancing classical machine learning model performance.

BASF is one of their customers, working on solving logistics scenarios (optimizing robot scheduling and solving a TSP problem), using DCQC. They also work with Lufthansa Industry Solutions on the QCMobility flight operations planning project, with IonQ on protein folding and optimization records, and with NYU Abu Dhabi.

Otherwise, Kipu acquired PlanQK / Anaqor in July 2024, which enabled them to become sort of a quantum applications marketplace, consolidated in April 2026 into the Kipu Quantum Hub, a platform giving unified access to quantum backends, solvers and AI-based orchestration.

In September 2024, the company published a roadmap on the way to reaching quantum advantage [3307]. It also introduced its Kipu complexity index (KCI) to define the workloads supported by their solutions on NISQ hardware. The KCI depends on how many terms are processed, the locality of the problem corresponding to the polynomial order of the optimization problems and the average number of connections per qubit in the used QPU.

In April and May 2025, the company published two papers claiming a quantum advantage in solving combinatorial problems, one using a branch and bound counterdiabatic optimization and the other with a digital quantum optimization, both running on IBM QPUs with over 100 physical qubits. The first one is said to outperform classical optimizer CPLEX [3308–3311].

In June 2025, Kipu Quantum researchers proposed a multi-QPU hybridized model named Sequential Quantum Computing [3312] with an example combining a first step run on a D-Wave Advantage 2 and a second one running a DF-BCQO on an IBM QPU.

Kuano

UK, 2020, \$3.6M

KUANO

Kuano creates quantum software solutions for the design of molecules and for the inhibition of enzymes, which is used both in pharmaceuticals and to create protective agents in agriculture. They use quantum emulation and quantum algorithms as well as machine learning. The company was founded by former GTN employees, including their CEO Vid Stojevic, who was the CTO of GTN.

Kunfeng Quantum Technology

China



Kunfeng Quantum Technology is a quantum software company based in Shanghai focused on machine learning applications. It developed HQconv and FQconv, two quantum neural network algorithms for image recognition [3313]. It also proposes an EDA software to design quantum chipsets, the Quantum-chip Design Automation Platform (QDAP) that is distributed as a cloud service. It was founded by Xu Hua, who has a PhD from the University of Maryland [3314].

MagiQware

the Netherlands, 2025

MagiQware

MagiQware is creating AI-based solutions for handling magic state distillation.

Menten.ai
USA, 2018, \$4M



Menten.ai develops hybrid algorithms combining machine learning and quantum programming to simulate organic chemistry and design enzymes, peptides and proteins, working with D-Waves annealers.

Miqro-Era
China, 2022



Miqro-Era is a quantum computing application company based in Shanghai that “offers cutting-edge quantum computing technology to solving technical problems”. They developed MiqroForge a workflow platform for quantum-enhanced computational chemistry [3315]. They had a staff of 40 as of mid-2025.

Mi Caves
Australia, 2016



Mi Caves provides “Quantum AI Centres”, sort of custom data centers mixing and matching AI, Quantum Computing, and 6G connectivity. This company’s website is filled with buzzwords but without any specifics on what they actually do in quantum computing.

MolKet SAS
France, 2024



MolKet SAS is a company created by Alain Chancé and Taha Selim that provides cloud-based software with AI services and solutions for molecular modeling and design using quantum and high performance classical computing.

Molecular Quantum Solutions
Denmark, 2019



Molecular Quantum Solutions or MQS, provides computational tools for pharmaceutical, biotech and chemical industries. It is using HPC and quantum computers. In September 2025, they published a paper comparing quantum annealing, digital annealing (both using QUBO solvers), and classical solvers for reaction network pathway analysis and mRNA codon selection [3316]. In most cases, classical solutions outperformed quantum solutions using a D-Wave annealer. They also published some hybrid algorithm mixing VQE and MPO (classical tensor networks) in 2024, although without indicating how many qubit they were using and where [2342] and other work on VQE in 2023 [3317].

Multiverse Computing
Spain, 2019, €726M



Multiverse Computing develops quantum and quantum-inspired software for various markets, starting with financial services, with portfolio optimization, risk analysis, market simulation and fraud detection.

They announced in August 2021 their Singularity Spreadsheet app solution which gives access to some portfolio optimization algorithm running on D-Wave annealers in the cloud, directly from within Microsoft Excel (video). It is now part of their Singularity SDK for Portfolio Optimization. They have then expanded their reach to mobility, energy, and manufacturing verticals. They also use more traditional techniques based on machine learning, digital annealing (with Fujitsu) and quantum inspired solutions using tensor networks. This became the Singularity Machine Learning module, made available in November 2024 on IBM’s Qiskit Functions Catalog.

They published several papers in 2022 on combinatorial problem solving using quantum annealing with ZF in Germany [3318], on artificial vision using both gate-based and quantum annealers with the research institute Ikerlan Technology Center in Basque country, Spain, reporting better results than the classical machine-learning baseline they tested, without a stated metric or uncertainty [3027], on a quantum-inspired tensor neural network to solve partial differential equations, in partnership with CACIB (French investment bank) [3319], and on portfolio optimization with Ally and Protiviti on D-Wave annealers [1249], as well as with Bankia [1247]. This is a good practice and track record for a quantum software vendor.

Their Fair Price quantum solution estimates prices under the specified pricing model for financial institutions and seems to run on IonQ quantum processors [3320]. The solution was licensed as a cloud service for €100K to €200K a year.

In November 2023, the company launched CompactifAI, a software compressor of LLM (large language models) using tensor networks with the aim of reducing energy and compute costs [3321]. CompactifAI is designed to reduce energy requirements at multiple points in the lifecycle of an LLM, including the training phase, general operations and retraining. Multiverse’s new software also reduces the overall footprint of the models, making them more portable and easier to run at the edge in applications such as autonomous vehicles and remote production facilities. Their work on LLM compression is going on [3322]. In June 2025, the company raised \$215M (€189M) to expand its CompactifAI offering, seemingly temporarily putting aside its quantum software business. In July 2026, it announced a Series C of “up to” \$570M (€500M) (meaning, it was not entirely closed) at a \$1.7B pre-money valuation, a five-fold step-up from the Series B, bringing total funding to about \$800M. The company reported annualized revenue growing more than ten-fold since the Series B, and about €100M of ARR as of January 2026. All of this is classical AI business, so I don’t account for this funding in my startup funding consolidation chart present later in the book.

The company is partnering with Xanadu, Microsoft, Fujitsu, IBM, Rigetti, D-Wave, NTT, Strangeworks,

Pasqal (France), IQM (Finland), Objectivity IT (UK, an IT services company), Kinesis Network Inc (on CompactifAI) among others. They have several offices outside Spain, in Toronto (Canada, with one of their cofounders, Sam Mugel), Munich (Germany), Paris (France) and San Francisco (USA, opened in 2024). The EIC (European Innovation Council) provided them with €12.5M comprising a mix of grant and venture funding. They have 22 patents pending in quantum computing algorithms. Multiverse made about €10M of revenue in 2023 vs €2.9M in 2022.

Late 2023, Multiverse announced with Moody's Analytics the launch of the QFStudio SaaS platform and marketplace for quantum-based and quantum-inspired solutions in the finance sector, with web and API access to algorithmic solutions [3323]. They also announced in 2025 a project to develop some quantum-inspired gesture recognition software for Airbus. Their disclosed customers and partners now include Allianz, the Bank of Canada, Bosch, Iberdrola, Indra, PwC (since November 2025) and Telefónica, with CompactifAI distributed through the AWS Marketplace and optimized for Qualcomm's AI200/AI250 datacenter accelerators. They hold over 160 patents.

NetraMark

Canada, 2016-2021



NetraMark develops quantum software solutions for the pharmaceutical industry to define therapeutic targets. They are part of the Quantum Machine Learning program at the Creative Destruction Lab in Toronto. It was acquired by the brain biotech **Nurosene Health** (2019, Canada) in October 2021.

Nexq

USA, 2024



Nexq is a quantum/classical AI software platform for creating applications across multiple industries, and particularly in healthcare.

Novarion

Austria, 2016



Novarion is a server storage and GPU server vendor who announced its intention to create the first hybrid quantum computer, originally targeted for 2025, without ever mentioning what sort of quantum processor it would integrate. Nothing has shipped.

In October 2020, they started a partnership with **Terra Quantum** (Switzerland) to create a Joint Venture to create '<Qa|aS>by QMware'. It supports machine learning and big data analytics. QMware's customers will be able to develop and effectively run completely Hybrid Quantum Applications. Applications built on QMware's Hybrid Quantum Cloud are supposed to run on upcoming native quantum processors when they show up. Meanwhile, it runs on some classical emulators, seem-

ingly a QLM from Atos/Eviden. And it is Gaia-X and GDPR compatible.

ODE L3C

USA, 2018



ODE L3C is an American NGO involved in the creation of chemical simulation algorithms. Its ambition is to solve "difficult NP" problems with quantum computation, which is far from obvious. This sounds more like a service provider than a software publisher. The company was created by Keeper Layne Sharkey, coauthor with Alain Chancé of the book "*Quantum Chemistry and Computing for the Curious*" (2022).

Opacity

Australia, 2020-2024



Opacity offered Quiver, a quantum code optimization software compatible with IBM's Qiskit. Their hardware-agnostic solution mapped processor errors at the global and individual qubit level, including parasitic interactions between qubits. It then allowed the code to be optimized to take into account this duly mapped noise. The tool seemed to be dedicated to developers as well as to quantum computers designers. The company went out of business in 2024.

Orientum

South Korea, 2024



Orientum is a weird startup working in the financial sector. They plan to create the Atom-Q platform that contains a software platform, a quantum cloud service to provide "*ultra-low latency services*" which is a curious promise given how QPU work, and...a quantum venture fund. This is a curious mix of activities [3324].

OTI Lumionics

Canada, 2011, \$5.7M



OTI Lumionics is specialized in the design of new materials and in particular LEDs and OLEDs. They have developed quantum and quantum-inspired molecular simulation algorithms for this purpose, mainly running on Nvidia GPU [3325, 3326].

In particular, this allows them to predict the properties of the created materials like their color when being excited [3327], model chemical relationships and determine geometric structures. They are partners of Microsoft Azure (video) as well as with AWS, with whom they developed a classical emulation of a variant of the VQE algorithm with 80 qubits [3328].

ParityQC

Austria, 2020



ParityQC is a spin-off of the University of Innsbruck created by Wolfgang Lechner and Magdalena Hauser,

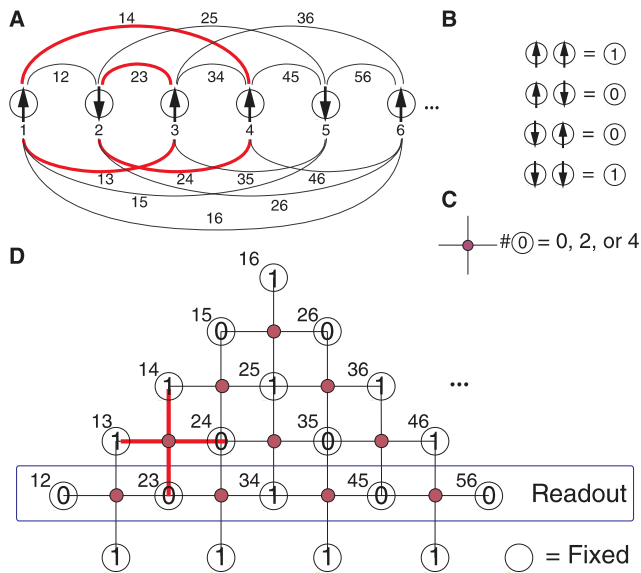


Figure 207: The LHZ architecture which turns a many-to-many spin problem into a nearest neighbor gate-base implementation. Source: [3329].

the first being the scientist and the second, handling business aspects of the venture. By September 2025, the company had about 60 employees on top of 15 researchers working at the University of Innsbruck.

As architects, they connect the dots between hardware and software. They support an architecture called LHZ, created by Wolfgang Lechner and Austrian colleagues Philipp Hauke and Peter Zoller, which is compatible with different hardware quantum platforms with 2D qubit connectivity architectures [3329–3338].

Its principle is to encode all-to-all logical interactions between qubits using only local physical interactions plus additional constraints, so that a problem can run on an architecture where qubits are connected to their nearest neighbors only, as is the case with solid-state quantum processors with superconducting and quantum dot spin qubits [3339] (Figure 207). Their architecture can also be implemented with FTQC systems and includes an in-house error correction scheme [3340].

They also develop software solutions to solve optimization problems (CADD, N-body problems, constraint problems) adapted to digital and analog quantum computers (qubits with universal gates or quantum simulators [3341, 3342]). Their ParityOS software suite optimizes the solution software parameters as well hardware control ones [3343].

They showcased in 2024 a version of their qLDPC derived parity code and FTQC architecture to correct biased noise qubits like cat-qubits [3344]. They also partnered with Quantum Brilliance for the development of a portable QPU for Cyberagentur, to be delivered by 2027.

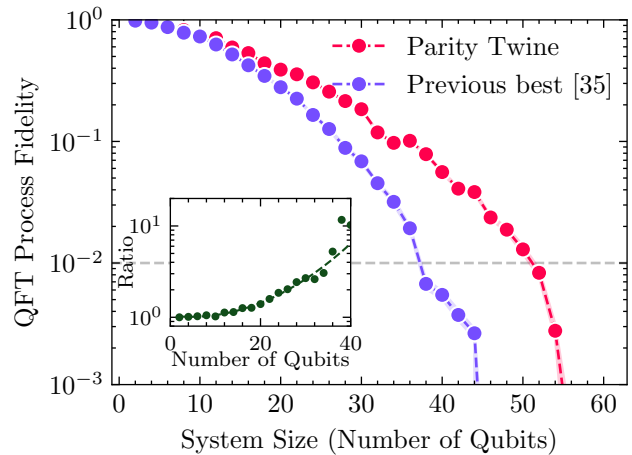


Figure 208: The circuit fidelity of a QFT realized with the Parity Twine encoding using up to 52 qubits. In that case, the post-processed fidelity is below 1%, which makes it unfortunately unpractical. Source: [181].

In 2025, Parity QC proposed its Parity Twine encoding method to optimize circuits gate count and circuit depth, that works with different qubit connectivity schemes: linear, square grids, hexagonal, ladder and all-to-all connected [1606]. It can reduce circuit size by up to 85%. They also proposed the concept of “replacement-type gates” which allow two and single qubit operations with preserving biased noised qubits, which improves the efficiency of quantum error correction codes. They prototyped it with cold atom and silicon spin qubits [3345, 3346].

The Parity Twine architecture was used, as presented in an April 2026 preprint, to run a quantum Fourier transform (QFT) with an IBM Heron r3 system, obtaining a post-processed fidelity of 1% with 52 qubits, a record so far [181] (Figure 208). But this fidelity is way too low to make such QFT practical. The paper’s main claim is a scaling improvement in two-qubit gate count relative to swap-based implementations when using iSWAP gates, reaching about a factor of 3 at the sizes tested. A process fidelity of about 1% is also not a useful algorithmic fidelity. These gates are supported by Google in one of their Willow chips.

They also described how to implement CNOT (fault-tolerantly) and arbitrary rotation gates (no fault-tolerantly) with their parity code used in concatenated mode in December 2025 [3347].

Phase Space Computing

Sweden, 2017

PHASE SPACE
COMPUTING

Phase Space Computing is a spin-off from the University of Linköping that develops training solutions on quantum computing for secondary and higher education.

Phasecraft

UK, 2019, \$56M



PHASECRAFT

PhaseCraft is a quantum software company spun out of University College London (UCL) and the University of Bristol by Toby Cubitt, Ashley Montanaro and John Morton (also a cofounder of Quantum Motion).

They initially planned to exploit quantum computing to create better energy collection and storage systems (batteries, solar PV, ...). The company launched in 2022 a partnership with Oxford PV, a perovskite photovoltaic panel company, to improve PV efficiencies thanks to quantum computer-based simulations. Their researchers developed an optimized version of an algorithm solving the Fermi-Hubbard model to compute some ground state with fewer resources which could be helpful to create high-temperature superconducting materials [2559, 3348].

In October 2025, they published a preprint on a simulation of the time-dynamics of a 2D Fermi-Hubbard fermionic many-body systems running on a 56-qubit Quantinuum H2 QPU, with a potential quantum advantage [2248] and a similar one running with 72 qubits from a Google Willow chip [2249]. In January 2026, they demonstrated that Fermi-Hubbard dynamics could be solved classically but it's a much simpler process than the ones implemented on Quantinuum and Google hardware [3349], thanks to the use of purely imaginary interactions, which allows the system to map onto a dephasing open-system Lindbladian (for example, to simulate decoherence) and scale polynomially on a classical computer.

In 2021, they also published a proposal for how to simulate a “kagome magnet”. While it did not exhibit any quantum advantage and worked only with 20 qubits, it extrapolated that with 50 qubits, the problem could be solved by 200 two-qubit gates, something high-fidelities NISQ qubit system now achieve [3350]. They did a similar thing in 2022 for materials simulations, still landing in a zone beyond the then state of the art NISQ capabilities [3351]. Their work also deals with reducing the circuit overhead of running QPE (quantum phase estimations) [3352].

In February 2024, Phasecraft was awarded a £1.2M contract from the UK government for phase 3 of a project to optimize energy grids. In September 2025, they raised a \$34M Series B, bringing total funding over \$50M including grants. Their hardware partners are Google Quantum AI, IBM, Quantinuum and QuEra, which is consistent with the platforms used in their 2025 and 2026 demonstrations.

In March 2025, they developed THRIFT (Trotter Heuristic Resource Improved Formulas for Time-dynamics), a quantum simulation method that improves simulation

efficiency and lowers simulation costs by about one order of magnitude for the 1D transverse-field Ising model, in comparison with standard Trotter product formulas for the same simulation, not with an end-to-end classical solver. It could have practical applications in materials science and drug discovery. In 2026, they also did show that using local fermionic encoding to qubits, computing the expectation values of quadratic fermionic observables is stable to noise in states with spatially decaying correlations, at the cost of using more ancilla qubits [3353]. This could enable simulations with larger fermionic systems on NISQ platforms.

PiDust

Greece, 2019



PiDust is a startup launched by Vasilis Armaos, Paraskevas Deligiannis and Dimitris Badounas, who are alumni of University of Cambridge, Stanford and the University of Patras. They develop quantum algorithms in chemistry.

Pine.ly

Canada, 2019-2022



Pine.ly is positioned on software to assist in the creation of innovative materials with quantum computing. They aim in particular at the recycling of CO₂ emissions. The startup was created by three women, Nayer Hatefi, Shabnam Safaei and Rachelle Choueiri, all three scientists. The company was seemingly closed in 2022.

POLARISqb

USA, 2020, \$2M



POLARISqb, aka **Polaris Quantum Biotech**, is a startup that wants to use quantum computing to create new therapies. The company was created by Shahar Keinan (CEO) and Bill Shipman (CTO). They are partnering with Fujitsu, probably using their conventional supercomputers and their digital annealing computer. Their idea is to use personalized medicine techniques to create ad-hoc therapies. Shahar Keinan received a PhD in chemistry from the Hebrew University of Jerusalem. She specializes in computational chemistry. In August 2022, they worked with **Auransa** to identify interesting target proteins for curing specific breast cancers using Tachyon and Auransa's SMarTR Engine based on classical AI.

They are mapping their problems into QUBO to run on D-Wave quantum annealers. Their QuADD (Quantum-Aided Drug Design) is a cloud-based Software-as-a-Service platform where researchers submit a defined 3D structure of a target protein binding pocket along with specific ligand requirements [3354]. The system translates the constraints (solubility, toxicity, geometric fit, synthesizability) into an optimization problem sent to a D-Wave annealer which scans a very large virtual

chemical space of molecules, and the solution identifies optimized molecular libraries targeting the specific binding pocket [3355]. They got a grant from DARPA in 2025 as part of the Embedded Entrepreneurship Initiative (EEI) to accelerate the commercialization of its quantum computing platform for drug design after they participated in the “Imagining Practical Applications for a Quantum Tomorrow” (IMPAQT) program, where they used quantum computing to identify small-molecule inhibitors against Ebola-related protein-protein interactions.

Prediq Technology

Belgium, 2024



Prediq Technologies is a CFD software company from Belgium which created Prediq Quantum in France in 2026, cofounded by Marc Tombroff (CEO), Kunal Puri (CTO) and Luke Peffers (advisor) that develops quantum CFD (computational fluid dynamics) algorithms for various fields including weather forecasting. At this stage, these are running on hybrid systems combining classical GPU and quantum circuit emulators.

ProteinQure

Canada, 2017



ProteinQure is a Toronto-based startup that uses various technologies including quantum computing to create and simulate new “*in silico*” therapies. They use quantum algorithms to simulate protein folding.

They are also developing hybrid algorithms using GPUs. They support different hardware architectures including D-Wave computers. In their experiments, they manage to simulate molecules with 6 atoms in universal quantum computers and reach 11 atoms with D-Wave. In practice, however, it would seem that they have put quantum computing on the backburner and are now focused on classical learning machines in the meantime.

Q1t

the Netherlands, 2018



Q1t creates mathematical models and software for classical and quantum computers in the field of quantum chemistry, quantum optics and financial analysis. They have tried these algorithms on quantum simulators and quantum optics.

They created the q1tsim quantum simulation library published on GitHub. It implements new quantum gate types for creating simpler circuits, simulating measurements without affecting qubit states, enabling a circuit to be rerun, starting with the previous quantum state for debugging purposes.

/q99

USA, 2023



The company created in Palo Alto by Argentine serial entrepreneur Facundo Martín Díaz (President) and cybersecurity serial entrepreneur Martin Sajon (CEO) is developing and patenting quantum applications integrated with AI and machine learning. It is Latin America’s first quantum software startup with a global enterprise focus. HYPERSMART is their conceptual framework which combines classical machine learning models with hybrid quantum-classical algorithms and probabilistic modeling. They propose a predictive intelligence engine to capture hidden variables and nonlinear relationships in data sets. The company targets financial services, logistics and energy, healthcare verticals and national security verticals. It seems to rely on Nvidia circuit emulators and on IBM quantum systems.

Q-Alchemy

Germany, 2024



Q-Alchemy is a quantum computing cloud platform developed by data cybernetics ssc, a German deep-tech startup which targets the energy sector. It was created by Carsten Blank from Karlsruhe Institute of Technology (KIT) and Georg Philipp Leser. The company developed a solution to optimize classical data loading in quantum circuits for NISQ and early stage FTQC based on the qTucker algorithm (Tucker Iterative Quantum State Preparation) [3356]. It constructs shallow, quantum circuits that reduce the gate count needed for this. The main use cases are quantum chemistry state preparation, quantum machine learning, partial differential equations (VQE), and Monte Carlo simulations. This is provided as a Python SDK supporting Qiskit and PennyLane.

QAR-lab

Germany, 2021



QAR-lab (2021, Germany) aka “Quantum Applications and Research Laboratory” created the UQO middleware platform for architecture-independent programming and platform evaluations, focused on solving optimization and machine learning problems.

They support D-Wave quantum annealers, Fujitsu digital annealers, and gate-based quantum computers from Rigetti and IBM. The company was created by Claudia Linnhoff-Popien (CEO) and Thomas Gabor who both come from the Institut für Informatik at Ludwig-Maximilians-Universität München (LMU) and had a team of 20 as of April 2023.

QbitLogic

USA, 2014, \$1.5M



QbitLogic is another startup that develops quantum machine learning applications, without more precision in their communication. They also develop an AI based system, CodeAI, to debug software.

Qbraid

USA, 2020



Qbraid provides a quantum online coding platform. Based in Chicago, the company was created by Kanav Setia (CEO) who has a PhD from Dartmouth College in quantum simulation and worked with IBM research.

The platform supports the most popular programming frameworks (Qiskit, TensorFlow Quantum, Q#, Xanadu PennyLane, Braket, IonQ, Rigetti pyQuil, D-Wave, Intel's C++ framework...) and contains a transpiler that optimized the quantum code for the target hardware platforms. They proposed a way to optimize the implementation of the VQE algorithm for chemical simulations [3357].

IBM listed qBraid Lab in 2024 as one of the recommended online notebook environments after IBM Quantum Lab was retired, and it now exposes more than 30 devices from IBM, IonQ, Rigetti, IQM, AQT, OQC, D-Wave, QuEra, AWS Braket and Azure Quantum behind a single API, with cross-framework conversion across more than 18 SDKs. A Rigetti Cepheus-1 108-qubit system was added in early 2026. In July 2026, the company launched QUEST (Quantum University Education and Support Track), a \$200K pooled fund with Rigetti, IonQ, AQT and OQC providing up to \$5,000 of compute credits per course to about 40 universities and more than 1,200 students for the 2026-2027 academic year, with North Carolina A&T as anchor institution. They also partner with Quantum Rings to expose its Open Quantum backends inside qBraid Lab, and publish the qBook courseware.

QBrain

Italy, 2023



QBrain is a company based in Milano created by Marco Maronese (CTO) and Lorenzo Moro. They created Qortex which provides an AI-optimized middleware to optimize quantum algorithms to a given QPU. It includes machine learning models trained to optimize gate and pulse-level quantum compilation, quantum error mitigation, qubit routing and embedding as well as parameters scheduling for analog quantum computers. In a word, it is about classical machine learning helping quantum computing. Qortex is provided as a SaaS solution.

Qcentroid

Spain, 2023



Qcentroid is a company created by Carlos Kuchkovsky and Antonio Peris that provides a quantum as a service

platform with an algorithms marketplace. It covers various verticals (financial services, transportation, energy, pharmaceuticals, retail) and does hardware benchmarking. Its early partners are Artificial Brain, Quintessence Labs, Kipu Quantum and Qulabz. It targets customers launching proof-of-concept projects but does not provide many details on their scope and supported hardware either natively quantum or in emulation mode, or even for designing quantum inspired classical algorithms. The company partners since May 2024 with OpenQuantum, a quantum open-source software initiative sponsored by the Unitary Fund, providing access to OpenQuantum software in the Qcentroid QaaS platform.

Q-CTRL

Australia, 2017, \$128M



Q-CTRL is a startup created by Michael Biercuk, who came from the University of Sydney. It develops a set of enabling software tools to improve the operations and programming of quantum computers around quantum error detection and suppression for NISQ quantum computers and quantum sensors, and, then, for FTQC platforms. In October 2024, it expanded its Series B to \$113M with \$59M of new equity, bringing total funding including grants to about \$128M. Its headcount was around 200 as of early 2026.

Boulder Opal is a quantum control infrastructure software working at the firmware level. It leverages machine learning to improve qubits control pulses and optimize quantum error mitigation. It is a Python toolkit used by quantum computers designers that works with IBM Qiskit, Rigetti and with Quantum Machines pulse generators. They implement error-suppression techniques that increase the likelihood of quantum computing algorithm success between $1,000\times$ and $9,000\times$ on quantum hardware, as measured using the QED-C algorithmic benchmarks, but probably with a very low number of physical qubits (Figure 209). These impressive numbers relate to the impact of optimized quantum error suppression techniques. In November 2023, this solution was proposed to IBM Quantum pay-as-you-go customers.

They also work on optimizing how qubits are used in a given QPU to take into account their characteristics [3358]. They are relying on Google Cloud and TensorFlow to run the classical machine learning algorithms of their solution [3359].

Fire Opal is a set developer tools for quantum algorithm designers while Black Opal is an educational tool for students new to quantum programming. In May 2022, Q-CTRL combined a 5-qubit QEC code (using 9 qubits, due to the additional helper qubits) with Fire Opal and could improve the code's ability to correctly identify errors by 70% on IBM QPUs (meaning: one qubit error detection rate was improved by 70%) [3360]. Fire Opal error suppression techniques can help improve the successful execution of QPE [3361] and QAOA algorithms at small scale [3362]. Starting in 2023, Q-CTRL was behind many of the "quantum utility" case studies

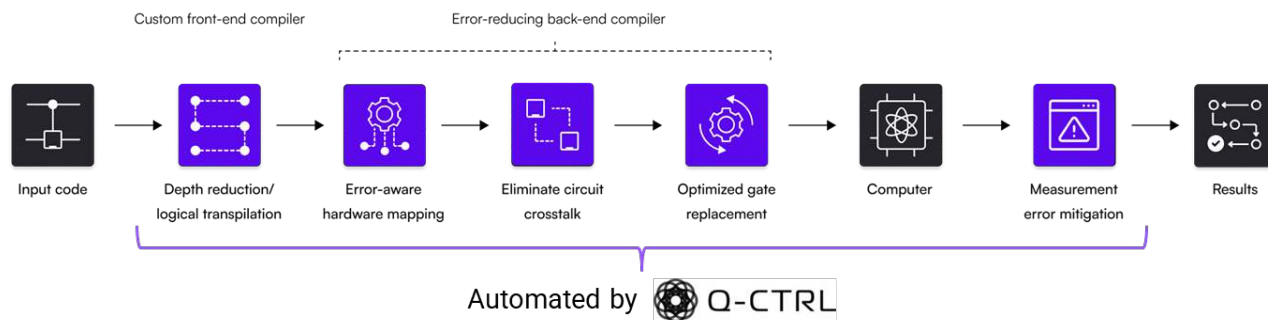


Figure 209: Q-CTRL error reduction techniques. Source: Q-CTRL. December 2023. Added in September 2024.

using recent IBM QPUs. Fire Opal supports QPUs from IBM, Rigetti, OQC, and Diraq.

Q-NEXUS is the name of the full-stack FTQC architecture they published in a preprint in April 2026. It physically separates the computational and information storage workloads using different types of qubits or error correction codes as needed [3363]. They propose to dedicate processors to the creation of the magic states. The architecture can reduce the number of physical qubits by a factor of 138 as compared to Craig Gidney’s 2019 paper for RSA-2048 key breaking with 26 million physical qubits [149], not with the 2025 update that requires only a million physical qubits [387]. The result is not far from the Pinnacle architecture proposed by Iceberg Quantum in 2026. It needs 190,000 physical qubits for a computation done in 10 days, while the Pinnacle architecture needs 151,000 physical qubits with a week computing time, also with superconducting qubits having similar characteristics and the use of non-local qLDPC codes [3290]. They also presented a compiler capable of orchestrating up to 1,000 logical qubits. Some options are stretch, such as the use of qRAM memory.

Quantum advantage? In May 2026, Q-CTRL touted a 3,000 times speedup for solving a problem of commercial relevance in materials science [913, 3364]. It was simulating the non-equilibrium dynamic of a 1D Fermi-Hubbard model. It used a chain of qubits in a 156 qubit IBM Heron processor to simulate a fermionic chain, like a snake which avoids the defective two-qubit gates in the chip. The peak load was the use of 13,800 two-qubit gates with a 452 layers deep circuit (with a 31-site chain and 62 qubits) and 9,057 two-qubit gates over 152 layers (with a 60-site chain on 120 qubits). It could have applications in understanding Mott-insulator transitions, quantum magnetism and strange-metal transport properties and for the design of high-temperature superconductors. 20,000 shots were run with a wall-clock time of less than 3 minutes. It was said to be up to 3000× faster than an optimized classical tensor-network simulation that requires 100 to 160 hours of classical CPU runtime. All this used their Fire Opal platform, a specialized fermion-to-qubit mapping sequence and a layer of fermionic SWAP (fSWAP) gates that compresses the circuit size. There are some caveats like

the post-selection technique with a discard rate reaching 98.3% for large chains. Algorithmiq challenged the performance, showing that their Majorana propagation technique could produce similar results on a standard MacBook Air also in less than 3 minutes [3246, 3365]. Qedma also implemented the simulation classically with an MPO-based Heisenberg evolution [3366]. Multiverse also challenged the claim with its own tensor network implementation in June 2026 [3367]. The debate keeps raging on the subtleties of these various approaches.

Snake layout

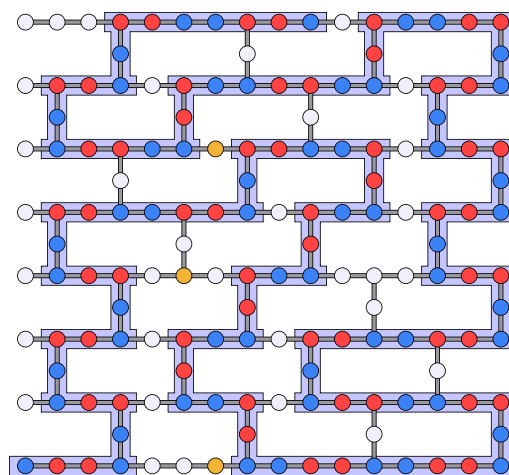


Figure 210: The 120-qubit snake used in Q-CTRL’s 1D Fermi-Hubbard simulation. It avoids the malfunctioning two-qubit gates in the IBM chip, that are continuously characterized. The qubit pairs color is an indication of the spin orientation they are encoding. Source: [913].

Their scientific contributions cover several other improvements. In May 2025, they demonstrated long-range entanglement and high-fidelity quantum operations by combining error suppression and error detection without full logical encoding. They created a 75-qubit entangled GHZ state [3368, 3369]. They also work on dynamic decoupling [3370], and implemented it in 2026 with the IBM Nighthawk 120-qubit processor to reduce the effect of increased crosstalk [3371]. They also proposed in 2025 a method that bypasses gate calibration by using fast tomographic characterization to map the true unitaries

of uncalibrated pulse sequences. Their pulse-aware compiler then tracks and compensates for during circuit synthesis to minimize computing times and improve the circuit execution fidelity [3372].

Q-CTRL is creating prototype algorithms for buses dynamic scheduling in Sydney, Australia (*disclaimer: it can't yet be operational given the power of existing quantum computers*).

Quantum sensing is another Q-CTRL activity. They optimize quantum sensing solutions particularly in the field of acceleration, gravity and magnetic fields measurement in space applications. They are partnering with Advanced Navigation (Australia), which specializes in geolocalization.

QClairvoyance Quantum Labs

India, 2024, \$3.4M



QClairvoyance was created Sai Shankar, a 20-year veteran of the Indian Army Corps of Signals, with research led by Jaiganesh G. The company builds quantum chemistry workflows for drug discovery, combining SQD with the LUCJ ansatz and DMET fragmentation, and runs them on IQM's 24-qubit Sirius processor with up to 16 qubits [3373, 3374] and also on IBM hardware. It also sells PQC consulting to Indian BFSI and government clients and operates a training business claiming over 1,000 students trained, acting as Bloq Quantum's quantum skilling partner since July 2026. They published an arXiv paper in June 2026 related to some vector state simulation on Fujitsu classical hardware, and it was withdrawn [3375].

Qodex Quantum

USA, 2025



Qodex Quantum is a company created by Ethan N. Evans (Chief Research Officer) and Zachary Bradshaw (Chief of Mathematics) that develops quantum machine learning solutions. They published in 2025 a long paper on quantum error correction with stabilizer codes [3376].

Quantum Computing Inc

USA, 2018, \$1.64B



Quantum Computing Inc, aka QCi, is a quantum software, hardware consulting company that started with the creation of Qatalyst, a high-level software development and cloud provisioning tool.

It is mainly enabling developers to solve constrained optimization problems (QAOA, QUBO, graph optimization) with either gate-based accelerators, quantum annealers or classical computers (Figure 211). It is based on using six simple high-level APIs. It is commercially available. They support their home QikStart Program, a marketing initiative to accelerate real-world use cases with their customers.

Qatalyst supports D-Wave, IonQ and Rigetti accelerators through Amazon's Braket cloud services. In March 2021, the company hired a "Chief Revenue Officer" (Dave Morris) and a Marketing VP (Rebel Brown) after having announced in December 2020 that they were filing for a Nasdaq IPO.

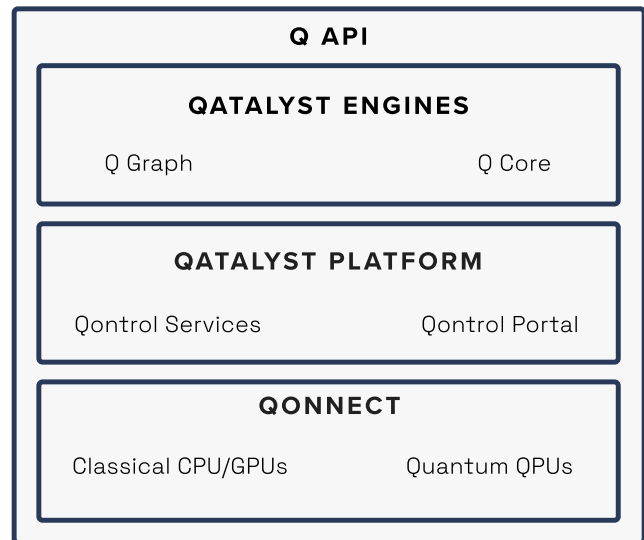


Figure 211: QCi software platform.

In July 2021, it became the first publicly traded pure player quantum computing company. QCi also created QUBT University in August 2021, an online initiative to train students on their Qatalyst tool. It is also a tactic to partner with academic institutions, like Notre Dame University in Indiana. They also have QCi Qonsulting practice, implementing their Path2Quantum (P2Q) methodology, a four-phase framework to planning quantum computing deployments. In March 2022, QCi announced a partnership looking like an acquisition of a stealth company, QPhoton (2020, USA) from New Jersey and created by Yuping Huang (also assistant professor at the Center for Quantum Science and Engineering at Stevens Institute of Technology), and which develops some sort of photonic quantum computer of unspecified nature.

QCi hooked this system to its Qatalyst software. QPhoton was funded by various US federal agencies (DARPA, NSF, NASA, DoD). QCI launched in September 2022 its Dirac 1 Entropy Quantum Computer (EQC) cloud-based subscription based on Qphoton's technology.

In June 2022, QCi announced QAMplify, a solution supposedly increasing the processing power of quantum computers by x20. Of course, it deserves some scrutiny. Their patented process is supposed to increase the number of variables that can be handled in quantum hardware by a factor x5 for gate-based and x20 for quantum annealing. How they are doing this is not yet documented.

Since November 2024, QCi has raised about \$1.64B through successive private placements, including \$200M, \$500M and \$750M tranches, ending Q2 2026 with \$1.3B

of cash and investments. That balance sheet funded a pivot away from software toward vertically integrated thin-film lithium niobate (TFLN) photonics and semiconductor manufacturing: it acquired Luminar Semiconductor for \$110M in February 2026 out of Luminar Technologies' bankruptcy, NuCrypt for \$5M, and NHanced Semiconductors in June 2026, launching a second fab. Q2 2026 revenue reached \$5.6M against \$61K a year earlier, though most of the company's income still comes from interest on its cash. At this point QCi belongs more in the hardware and photonics sections of this book than among software vendors.

Q Deep

Russia, 2025



Q Deep is a service and contract research company on quantum computing and communications that is part of the Tatarstan Quantum Consortium. It was created by Hadi Salloum. It created Qonquester Cloud, a hybrid solver operating on D-Wave to solve QUBO formulated problems. It also created QGPT, a LLM solving optimization problems.

QBaltic

Estonia, 2019



QBaltic develops algorithms for quantum computing, quantum cryptography and artificial intelligence. QBaltic is a contract research spin-off from the University of Latvia, University of Tartu in Estonia and QuBalt, Germany and Latvia.

QbitSoft

France, 2022



QbitSoft is a software as a service (SaaS) vendor solving optimization problems on various industries like insurance, financial services and logistics.

They are developing algorithms using Quantum Annealing capacities. The French insurance company Matmut is one of their customer references. They also work on decarbonating supply chains in France. The company was created by Olivier Pegeon, a former business executive from IBM France. They are a team of 12 as of September 2024 o/w 4 PhD's.

QC Ware

USA, 2014, \$39.7M



The startup was created by Matt Johnson, who has a financial background, and Kin-Joe Sham and Randy Correll, who seem to have gotten late into quantum computing. The team also included Iordanis Kerenidis, based in France, a leading specialist in quantum machine learning. He was overseeing international algorithms development. But he left the company in 2024 to create his own startup, Quantum Signals, as QC Ware was refocusing the entire company on quantum chemistry.

Airbus Ventures is one of their investors on top of the Koch group. They received a \$1M US public funding via the NSF in 2017.

QC Ware develops a platform for cloud-based quantum software development. They create quantum algorithms and software for large companies with two layers: their proprietary Forge platform and open-source libraries for optimization, chemical simulation and machine learning. They provide tools to load training data from learning machine models into memory more quickly. They have also developed an algorithm for calculating the distance between objects, which can be used to train both supervised (classification) and unsupervised (clustering) machine learning models. Their first customers include **Equinor** for oil exploration optimization, Japan's **AISIN** for certification testing of automatic gearbox software, **Airbus** for aircraft flight envelope optimization and **BMW** for autonomous vehicle route optimization. They are also targeting financial markets as well. It supports universal gate quantum computers (IBM, Rigetti), D-Wave quantum annealing computers and software emulators (IBM, Google, Microsoft, Rigetti).

In April 2023, QCWare introduced its Promethium classical software-as-a-service (SaaS) quantum chemistry platform designed to help develop pharmaceutical, chemical, and materials discovery solutions. It runs on NVIDIA H100 and A100 GPGPUs. It can compute various chemical systems with about 100 atoms and up to 2,000 atoms with one to two orders of magnitudes time savings compared to other classical solutions according to them. The SaaS platform runs on AWS.

The "case studies" they are communicating about were created in 2023 with JPMorganChase (deep hedging using quantum reinforcement learning, tested with 16 qubits [2959]), AISIN (supply chain logistics tested on the D-Wave hybrid solver and solving small instance problems [2864] and static code testing) and Boehringer Ingelheim (protein-drug interaction using quantum signal processing using 8 qubits, along with AQT and Covestro [3377]). These were more small scale "proofs of concepts" than real and deployed "case studies".

In 2026, QC Ware ran two hybrid quantum-classical chemistry demonstrations with Promethium. In August, it computed the electrostatic interaction energy of nitric oxide reductase by combining GPU-accelerated modeling with measurements on a 156-qubit IBM Heron processor. In September, it repeated the exercise on IonQ's Forte through Amazon Braket, reducing a 115-atom P450nor active site model with over 1,000 molecular orbitals down to a 4-orbital active space mapped onto 8 qubits, landing within 4% of the reference and inside the 1 kcal/mol chemical accuracy threshold. Both remain technology demonstrations rather than product features, and the qubit counts stay tiny, but the workflow is honest about where the quantum part actually sits. Promethium itself is claimed to run up to 20×

faster than conventional CPU-based DFT platforms on selected workloads.

The startup organizes a reference annual conference on quantum computing, the **Q2B**, in Santa Clara (California, since 2017), Tokyo (since 2022) and in Paris (from 2023 to 2025), and then Copenhagen (2026) and Chicago (2027) [3118].

QDC.ai

USA-Poland, 2022



QDC.ai is a company developing machine learning, HPC, combinatorial optimization, scientific programming, and quantum computing algorithms, mostly using optimized classical hardware platforms, but with an eye on quantum hardware. Their software platform VeloxQ is an HPC engine that automatically selects the appropriate solver to solve various problems (QUBO, MAX-SAT, MAX-CUT, CP-SAT, Simulated Bifurcation and Heuristic algorithms).

QEC Labs

USA, 2025



QEC Labs was created by Nithin Raveendran (CEO) and his PhD advisor Bane Vasić, from the University of Arizona. They both contributed to the development of qLDPC codes, along with Valentin Savin, now at Quobly [3378] and also with Harvard's Mikhail Lukin's team [3379]. Their goal is to build efficient decoders, fault-tolerant logic, and complete QEC systems using qLDPC error correction codes. Their core product supports ultra-fast, hardware-native qLDPC decoders targeting sub-microsecond latency for scalable, FTQC, implemented across GPU, FPGA, and ASIC platforms, probably based on the founders past academic work [3380, 3381].

Qdaria

Norway, 2023



Qdaria was created in Oslo by Daniel Mo Houshmand. It develops quantum data science and quantum machine learning software. It also explores the field of synthetic biology and topological quantum computing based on based on Fibonacci anyons.

QEDma Quantum Computing

Israel, 2020, \$30.7M



QEDma Quantum Computing is a quantum software company cofounded by Dorit Aharonov (Hebrew University), Nathaniel Lindner (Technion) and Asif Sinai.

They create quantum algorithms and software tools positioned as a quantum operating system. It includes a multi-qubit quantum gates characterization tool, a gate optimizer and a fidelity enhancing compiler implement-

ing error suppression and error mitigation techniques [3382], the Quantum Error Suppression and Error Mitigation (QESEM) solution, which is using an unbiased quasi-probabilistic (QP) mitigation framework and is proposed as a pattern in Qiskit Functions [3383].

They published in December 2025 a preprint targeting the transitional between NISQ and FTQC regimes [3384]. They introduce Syndrome-Aware Logical Error Mitigation (SALEM), a framework that applies error mitigation at the logical gate level of a fault-tolerant architecture, improving executable circuit volume through several mechanisms: recycling internal error correction data, reducing runtime costs with inverse-variance weighting, with achieving exponential quantum volume boosts, with introducing a volume-dependent QEC threshold and with practical scaling via coarse-graining and mid-shot rejection (CG-SALEM). They are also associated with one of the three separate quantum advantage claims presented by IBM in June and July 2026, each of which should be assessed on its own task, validation protocol and best disclosed classical comparator [3385]. That preprint, not yet peer reviewed and posted on July 27, 2026 with RIKEN and BlueQubit and promoted by IBM on July 30 as "trusted quantum computation", used QESEM on an IBM Heron r3 to measure magnetization dynamics of a Floquet Ising magnet on a heavy-hex lattice with percent-level precision, resolving long-lived subharmonic prethermal oscillations in systems of up to 74 qubits. Leading tensor-network simulations failed to converge in that regime and sparse Pauli-path simulations stayed strongly truncation dependent despite extensive GPU and Fugaku computations. The claim is corroborated at selected Floquet cycles on Quantinuum System Model H2 and Helios trapped-ion hardware, which is a more careful validation protocol than most quantum advantage claims come with. It remains an empirical claim about where tested classical methods become unreliable, not a complexity-theoretic one. This assessment is dated September 2026 and later classical simulations may revise it.

Among others, the company partners with Quantinuum on quantum error mitigation.

Qindom

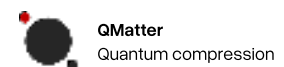
Canada, 2018-2023, \$2M



Qindom was developing quantum machine learning (QML) software running on D-Wave quantum annealers.

Qmatter

UK, 2024, \$1.2M



Qmatter was created by Alexis Ralli, Timothy Weaving, Peter Coveney from UCL, and Peter Love from Tufts University. They reduce the size of quantum problems to enable both classical and near-term quantum systems simulations in the fields of drug discovery and materials research. They also create physics informed data libraries for machine learning companies to train AI

models. Their methods are documented in a couple arXiv papers [3386–3390].

QMill

Finland, 2024, \$4.5M



QMill is a company created by Hannu Kauppinen (CEO), Ville Kotovirta (CTO), Mikko Möttönen (CSO) and Toni Annala (Chief Algorithm Engineer) which is developing NISQ quantum algorithms to solve complex optimization problems.

In January 2026, the company claimed that its latest quantum algorithm could demonstrate verifiable quantum advantage on a 48-qubit system at 99.94% gate accuracy, outperforming the El Capitan supercomputer, a six-fold relaxation in fault-tolerance requirements compared with an earlier estimate that required 200 qubits at 99.99% accuracy [3391]. This is however not documented in a paper and the problem is not specified nor the circuit depth. The 6-fold improvement concerns both the reduction in qubit count and the relaxation in qubit error rates.

Quantum Research Sciences (2024, USA) is a quantum software company that spun out of Purdue University's Quantum Institute. It received an Air Force research grant of \$2.5M to develop some quantum software for optimizing spare parts inventories. It is supposed to achieve a 28% improved accuracy over classical computers. They also work with CERN on parsing the data coming out of the LHC. Given their partnerships, they seem to use both IBM and D-Wave QPUs. The company has a staff of 10 as of October 2024.

Qognitive.ai

USA, 2024



Qognitive.ai is building some sort of quantum and brain inspired machine learning solution for financial markets (Quantum Cognition Machine Learning or QCML). In 2026, they published a paper on the development of a linear Hamiltonian-based QML method with a compact quantum representation of classical data via ground state problems for k-local Hamiltonians, along with IBM [3392].

Qoherent

Canada, 2019, \$173K



Qoherent is developing machine-learning based RF signals analysis tools that helps create adaptive RF communications and sensing systems in the Software-Defined Radio (SDR) realm.

They analyze signals in the telecom ranges (4G, 5G, V2X, Satellite communications...). The company is investigating the usage of quantum machine learning techniques to improve its solutions.

Qoro Quantum

Germany-UK, 2024



Qoro Quantum provides a software platform for distributed quantum computing [3393, 3394]. It implements algorithm processing, partitioning and parallelizing, sending jobs to multiple compute nodes using various techniques (circuit cutting, dynamic circuits). They also provide tools to interconnect QPUs and HPCs, and developed Maestro, a quantum circuit emulator supporting state vector, tensor networks and stabilizer modes, and GPU-based acceleration, that they released in December 2025 [3395]. They were 5 as of July 2025.

In May 2026, they launched Solo, a self-service cloud tool to simulate (or, to me more precise, emulate) hybrid quantum workloads combining CPUs, GPUs, and NISQ quantum computers, and using their Maestro engine.

QPerfect

France, 2023-2026, €2M



QPerfect was a spin-off from UNISTRA (University of Strasbourg) whose first product was MIMIQ-circuits, a quantum code emulator supporting large circuits with hundreds of qubits and using optimized tensor networks.

The company was created by Sébastien Buffechoux (CEO, who left the company as of early 2024 and was replaced by Philippe Blot), Guido Masella (CTO), Guido Pupillo (qubit control lead), Johannes Schachenmayer (Numerics lead) and Shannon Whitlock (Hardware lead). The company had a staff of 11 as of September 2024. In June 2024, QPerfect announced a partnership with QuEra (USA), which will use MIMIQ to model QuEra's quantum error correction.

In January 2025, in an emulator benchmark carried out by researchers from PSI, ETH Zurich and EPFL, MIMIQ performed as one of the top emulators tested with a QFT [1665]. Its speed is due to the use of low level CPU instructions and memory optimization, applicable to the range of 5-20 qubits. Above 20 qubits, MIMIQ switches over to an MPS tensor network engine, which fares well up to hundreds of qubits. In 2025, QPerfect created its own benchmark of quantum emulators [3396]. In 2026, Shannon Whitlock published AtomTwin.js, a digital twin simulation framework for cold atom operations [3397].

In Spring 2025, BTQ announced its willingness to invest \$2M in QPerfect for studying the development of a gate-based cold atom QPU, to help them mine cryptocurrencies. The company is also working on quantum error correction codes [3398]. In November 2025, BTQ acquired QPerfect for both its software and hardware skills in cold-atom gate-based quantum computing. The acquisition was administratively completed in July 2026.

QPurpose

Denmark, 2022



QPurpose develops quantum algorithms and quantum software “*for practical use cases in all businesses*”. The company is a spin-off from the Centre for Quantum Mathematics (QM) at the University of Southern Denmark.

Qrithm

USA, 2018



Qrithm develops quantum algorithms in diverse and rather disparate fields: machine learning, materials science, cryptography and finance.

Qruise

Germany, 2021



Qruise is providing the Qruise Toolkit, positioned as a collection of algorithms (optimizers, simulators, machine learning tools and other software components) designed to accelerate the development of quantum technology and to improve its applications.

The company was cofounded by Shai Machnes, Simone Montangero, Frank Wilhem-Mauch and Tommaso Calarco and had a team of 20 people as of April 2023. Their scientists contributed to over 280 publications at the same date in both quantum physics and quantum computing, with interesting work on machine learning based quantum mitigation error techniques, parametrized quantum machine learning circuits [3399], benchmarking optimization algorithms [3400], and on the energetics of quantum computing [3401]. In 2022, they were awarded a €2.5M grant from the EIC Council Transition program.

Qsimulate

USA, 2018, \$9.3M



Qsimulate develops quantum solutions for molecular simulation for healthcare and chemistry. They are partners of Amazon Braket and Google and are already working with Amgen. They also work on nuclear fusion simulations [3402]. The company was co-founded by Toru Shiozaki and Garnet Chan, both specialized in chemistry.

QSIMPLUS

Korea, 2023



QSIMPLUS is selling QSIMpro, a software quantum code emulator used for quantum communication and optical network design.

Qu&co

the Netherlands, 2017-2022



Qu&co was created by Benno Broer and Vincent Elfving, both quantum physicists who worked at TU Delft.

They developed tailor-made quantum software solutions for large companies, accompanied by benchmark tools, particularly for chemical simulation applications based on DFT. They develop solutions for simulating fluid mechanics with nonlinear differential equations solved with VQE algorithms (hybrid variational quantum eigen-solvers). They even solved Navier-Stokes 1D equations. They partner with IBM, Microsoft and Schrodinger (USA).

In March 2021, they launched a beta release of QUBEC, a chemistry and materials science toolkit. It comprises Q-time, a tool estimating when quantum advantage can be expected for solving chemistry or materials problems. QUBEC workflow manager supports quantum systems from IBM, IonQ and Rigetti. It is available through the IBM Quantum Experience and Amazon Braket platforms. LG Electronics is one of their customers.

In August 2021, they announced a new funding round with Quantonation, Runa Capital and SPInvest, with an undisclosed amount and in January 2022, the company merged with Pasqal (France). Benno Broer became for some time the CCO of Pasqal (Chief Customer Officer). What became the Amsterdam office of Pasqal was closed in Summer 2025.

Quanscient

Finland, 2021, \$21.6M

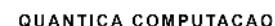


Quanscient was created by Alexandre Halbach, Asser Lähdemäki, Valtteri Lahtinen and Juha Riippi (CEO).

It provides Quantum Simulation-as-a-Service (QSaaS) software. Its multiphysics simulation algorithms use finite element method and partial differential equations solvers. They hint about using hybrid classical/quantum computing but given the state of the art of quantum computers, it is probably relying on classical computing. They help simulate physics in the electromagnetism, mechanics and fluid dynamics fields.

Quantica Computacao

India, 2019



Quantica Computacao is the first Indian quantum startup and a software company working on creating a cloud development environment an QML algorithms. It is incubated in the Indian Institute of Technology Madras from Chennai.

Quantopticon

UK, 2017



Quantopticon develops the Quantillion modeling and simulation software for the design of photonic compo-

nents for quantum communication and computing applications. It is a tool for the design of new materials.

The company was created by a mother-daughter duo: Mirella Koleva (CEO) and Gaby Slavcheva (CSO). Their first customers are Qandela, QuiX and Toshiba.

Quantum Benchmark Inc

Canada, 2017



Quantum Benchmark provides an error-correcting code software solution for general-purpose quantum computers and error evaluation. It was thus apparently a competitor to Q-Ctrl.

They also offer a quantum computer performance validation system. The package is integrated into the True-Q suite, launched in 2018, with True-Q Design, which is used to evaluate the error rate of a quantum computer and to optimize its architecture, and True-Q OS, which helps optimize the accuracy of software solutions. The target market is initially the manufacturers of quantum computers and those who evaluate them. Eventually, it will be that of user customers. Note that they have already tested Google's Cirq framework, having been part of Google's beta test program for this language, and that Google is using their solution, as it did in its 2019 supremacy experiment. They are also partnering with IBM (since 2018, on error characterization, mitigation, correction, and performance validation) as well as with Fujitsu Labs (since 2020, to develop quantum algorithms). The company was acquired by **Keysight Technologies** in May 2021, on top of Labber Quantum in 2020. Among its key people are Joseph Emerson (CEO), Joel Wallman (CTO) and Daniel Gottesman (Senior Scientist, specialized in error correction, then at the Perimeter Institute), Stefanie Beale and Kristine Boone (both researchers).

Quantum Elements

USA, 2023



Quantum Elements is a software company created by Izhar Medalsy, Daniel Lidar (USC) and Amir Jacoby. They position themselves as developers of an AI-based scalable and universal quantum operating system. They cover automated quantum hardware calibration to error mitigation and suppression, including a technique of logical dynamical decoupling which suppresses errors that are not well addressed by error correction codes. They also work on surface codes implementation with test done on IBM hardware in 2026 [3403].

Quantum Flytrap

Poland, 2020-2022



Quantum Flytrap was a quantum computing and cryptography software company. They created a real time browser-based emulator using photonic elements instead of quantum gates, the "Virtual Quantum Lab" that is being used at Stanford University and the University

of Oxford [3404]. It enabled the simulation of an optical table and to run various quantum optical physics experiments. They developed the Pulser Studio IDE for Pasqal. The company has shown no visible commercial activity since, though the Virtual Quantum Lab remains online.

Quantum-South

Uruguay, 2019



Quantum-South is a spin-off from the University of Montevideo who is specialized in developing quantum optimization software, first targeting the cargo shipments in ships and airlines. Their software was released in 2022. The Automatic Package Selection and Ordering (APSO) is proposed as a quantum inspired solution as of 2023. They also target the financial sector which may be more dynamic although more crowded with many existing quantum software startups. In cargo shipments, they are partnering with Quantum Brilliance (Australia) with their prototype (5) NV centers qubits which, of course, is experimental.

Quacocon

USA, 2020



Quacocon is a small startup created by Tina Sebastian and Barbara Dunn that develops software solutions for the food supply chains combining AI and quantum annealing.

Quanmatic

Japan, 2022



Quanmatic is a startup based on the research of Nozomu Togawa (CSO) of Waseda University, with Sumitaka Koga (CEO), Shu Tanaka (CTO) and Yosuke Mukasa (CPO).

They develop algorithms for quantum computing, classical optimizations and AI. In February 2024, they solved an NP-hard optimization problem using quantum or digital annealers. They implement some form of hybrid annealing method that extracts a subQUBO model from a large-sized QUBO model on a classical computer and solves the extracted subQUBO model on an Ising machine (D-Wave or Fujitsu) [3405]. They also partner with OQC, and with Rohm in the field of manufacturing process optimization.

QuantonomiQS

Belgium, 2026



quantonomiQS is specialized in the development of applications in the financial services vertical with a focus on compliance, security, and ownership. Their AiQverse software platform orchestrates classical, AI, hybrid, and quantum models. The dose of quantum here is a typical unknown.

Quantum Thought

USA, 2019



Quantum Thought (2019, USA) develops quantum or quantum inspired algorithms for chemical, AI and security markets. According to their website, it seems to be mainly a service company operating in project mode and doing consulting services. Their CEO is Rebecca Krauthamer.

Quantastica

Finland -Estonia-Serbia, 2019



Quantastica develops hybrid quantum algorithm software tools including Quantum Programming Studio, a graphical web development environment for creating quantum algorithms executable on quantum computers or simulators, including a classical simulator they have developed themselves.

Quantistry

Germany, 2018



Quantistry created a cloud-based simulation platform for material research and development, using quantum simulations, molecular dynamics and machine learning.

Quantumz.io

Poland, 2019, €1 M



Quantumz.io develops the Quantum Simulator Platform (QSP), a quantum program emulation solution running with GPUs. In September 2025, it released VeloxQ 1, a classical optimization algorithm that they say outperforms D-Wave annealers in speed and accuracy. They are also developing a PQC (post-quantum cryptography) solution called banax, including some dedicated hardware to implement it.

Quantopo LLC

USA, 2017-2021



Quantopo LLC was specialized in machine learning algorithms. They focused on biotech, supply chain and logistics. They are part of the Creative Destruction Lab in Canada. But as they don't have an active website, it is not certain that they still exist.

Quantum AI

Italy, 2021



Quantum AI is a company providing AI based financial trading software. It is supposed to use some form of quantum computing that is not disclosed. I suspect it is some form of (classical) quantum inspired software.

Quantum Mads

Spain, 2020



Quantum Mads was created in Bilbao by Eriz Zárata and Alain Mateo Armas and is positioned in the financial market. It offers four software tools with a mix of quantitative/classical/hybrid/quantum-inspired software. With Q-MADS, an investment strategy analysis framework for traders, Q-RETAIL, a framework for retail banks, Q-ALLOCATE, for asset allocation optimization and Q-CRYPTO, a framework for optimal path finding in graphs. The whole is based on the HHL linear algebra algorithm. They also do some fundamental research on quantum learning with academics in Australia and Singapore [3406].

Kausal.ai

USA, 2020



Kausal.ai is providing quantum and classical model insights and governance. Their platform uses large language models as well as causal methods to provide human understandable insights for health care, life sciences and finance industries. It provides classical and quantum machine learning model governance, monitoring, and explainability, identifies when to re-train models, meta insights combining multiple models, model discrimination analysis, population and characteristic stability and explainability.

Quantum open-source Foundation

Slovakia



Quantum open-source Foundation got a \$4K grant from Unitary Fund. It publishes a list of quantum open-source projects on [GitHub](#), with various software development tools and libraries for gate-based quantum as well as and quantum annealing computing. It is a repository of existing open-source projects including IBM Qiskit and PennyLane from Xanadu. NISQ. Provides financial funding for quantum open-source software projects. Organize events. More a community than a foundation like the Apache or Mozilla foundations.

Quantagonia

Germany, 2021-2025, \$4.3M



Quantagonia is a company based in Frankfurt and Berlin who created HybridSolver, a hybrid quantum platform allowing users to run existing code and decision models on both quantum hardware and high performance computers, including on D-Wave quantum annealers [3407].

When you look at the details, this is mostly a way to implement more efficiently mixed integer programming, linear programming and QUBO problems on classical computers, before quantum computers can support it with some computing advantage. I have doubts on their claim to support existing code, but it can make sense for decision models which are code-independent and are implemented as optimization problems which have their specific implementation in classical and quantum

computing. The company was created by Dirk Zecheil (CEO), Sabina Jeschke, Sebastian Pokutta (CTO) and Philipp Hannemann. It was acquired by **Strangeworks** (USA) in August 2025, its HybridSolver orchestration engine becoming part of the Strangeworks stack.

QUANTier

Hong Kong, 2022



QUANTier is a spin-off from HKUST (Hong Kong University of Science and Technology) which was to develop a neutral atoms qubits based quantum computer with some undisclosed funding from ParticleX, an accelerator and investment fund from Hong Kong. They pivoted later to provide some cloud based quantum computing middleware.

Quantum Intelligence Corp

South Korea, 2023



Quantum Intelligence Corp was launched by Hwan ho Choi.

It created the QUEST-ADMET platform, which uses quantum computing and AI to predict compound properties through electronic charge distribution algorithms, efficiently assessing the efficacy and safety of therapeutic drug candidates. It partners with Classiq in pharmacology and with QuEra on drug discovery.

Quantum Signals

France, 2024



Quantum Signals was Yanni Gamvros and Iordanis Kerenidis, formerly at QC Ware. They design AI and quantum based financial services, for modeling the microstructure of financial markets. They predict distributions of mid-price movements and liquidity trends for equities and futures in intraday trading. At this point, it seems at best quantum inspired classical software.

QuantyCat

USA, 2020



QuantyCat created cloud-based APIs for quantum software development, supporting D-Wave, IonQ and Rigetti through Amazon Braket. But they seemed to have exited the quantum business as of 2026.

Qubit Engineering

USA, 2018



Qubit Engineering was founded by University of Tennessee alumni. They develop classical and quantum optimization algorithms suitable for wind turbine design and location optimization. Another very niche market. They are partners of Microsoft Azure.

Qubitera

USA, 2018



Qubitera develops solutions combining AI and quantum.

Qubitra Technologies

UK, 2026



Qubitra Technologies a UK-based joint venture between Fujitsu and the investor fund sventures led by Vishal Shete (CEO), formerly at Terra Quantum, Daniel Wynne (COO) and Kugendran Naidoo (CSO) from J.P. Morgan. They are creating quantum-enabled applications and a marketplace platform for the global quantum ecosystem, focused particularly on financial services (fraud detection, derivatives pricing, trading). It will use a mix of quantum computing and quantum inspired techniques. The application platform will provide access to Fujitsu's digital annealer and proprietary quantum applications.

Qubitl Quantum Technologies

India, 2018



Qubitl Quantum Technologies is a contract research laboratory developing quantum machine learning software. They initially specialized in cybersecurity, having created a QRNG solution (Q-RandCon), a PQC protected healthcare solution (HealthCetra) and a Quantum Differential Phase Shift technology for QKD (Q-Shift).

Qubit Pharmaceuticals

France-USA, 2020, €23M



Qubit Pharmaceuticals is a startup co-founded by Jean-Philip Piquemal, a CNRS professor-researcher at Sorbonne University, a long-time specialist in molecular dynamics simulation that mathematically models the quantum mechanics of organic molecules. It is led by Robert Marino (CEO).

Its algorithms have been in use for a long time. Jean-Philip Piquemal is the co-author of the Tinker molecular simulation library and its Tinker-HP version adapted to supercomputers. It exploits massively parallel CPU-based systems and Nvidia GPU tensors, all with high-precision computation to compute molecular polarizable force field. Their Amoeba software computes molecular interactions and water polarization, and Sirius implements machine learning. Other solutions are dedicated to variational quantum eigensolvers (VQE). At this point in time, they are using quantum inspired and quantum emulation solutions running on classical systems like their own HPC-QC Gaia HPC of 50 PFLOPS installed in the Paris region. It supports the full emulation of up to 40 noiseless qubits [3408]. They partner with various institutions like Institut Curie and the University of Bordeaux in cancer therapy developments.

They develop drugs in three sectors: oncology, immunology and antivirals. What about quantum computing in all this? It could be used to define optimized parameters for classical simulation, in short, within the framework of hybrid algorithms.

They find ways to reduce the footprint of quantum computing [1565, 3409]. They regularly publish interesting research papers which are scattered around in this book, notably along with Pasqal. In 2025, it includes a way to create quantum chemistry synthetic datasets [777], a variant of an adaptive VQE that is resilient to qubit noise [3410] and a classical AI foundation model that simulates dynamic molecular interactions with good accuracy and efficiency [3411].

Qubit Pharmaceuticals also created Hyperion, a GPU-based circuit simulator/emulator adapted to chemical applications [1713]. It uses custom-optimized Sparse Matrix-Sparse Vector (SpMspV) kernels to accelerate exact matrix-vector multiplications, which enables State-Vector (SV) ADAPT-VQE exact simulations with up to 32 qubits and uses a new partitioned emulation technique named the SV-MPS strategy that routes non-interacting terms into an exact sparse SV core and delegates interacting terms to the MPS (matrix product state) engine, supporting up to 40 qubits. They also proposed the MB-ADAPT-VQE variational algorithm that uses of Lie algebras structures to determine efficiently the Pauli strings to use (*aka* “generators”).

Jean-Philip Piquemal’s laboratory at Sorbonne University got a €9M ERC for the development of simulation solutions for organic systems of several million atoms [3412] in 2020. They welcomed the investment fund Quantonation in their capital in June 2020 [3413].

As of mid-2026, they had a staff of 60 spread across Paris, Boston and Sherbrooke.

QubitSolve

France, 2026



QubitSolve is a CFD (computational fluid dynamics) software company created by Madhava Syamlal [3414]. It was awarded a \$1.1M SBIR fund in 2025 to develop CFD software for quantum computers.

QuDot

USA, 2018



QuDot developed software for the simulation of quantum circuits on traditional computers, the QuDot Net. They used techniques based on Bayesian networks to optimize the in-memory representation of qubits. The company seems dormant or closed.

Quminex

Canada, 2024



Quminex is using AI and quantum computing to improve mineral exploration. It was created by Marc Fassbender (CEO). What they exactly do with quantum computing is not yet documented.

QunaSys

Japan, 2018, \$27.2M



QunaSys also develops quantum algorithms in chemistry [2422, 3415–3418] and healthcare [3419]. From the universities of Tokyo, Osaka and Kyoto, they also maintain the Qulacs simulator developed at Kyoto University. In 2021, they created Qamuy, a quantum cloud platform. They also created in 2020 the Japanese consortium QPARC with 50 participating companies investigating various use cases of quantum computing. The company is also participating in the **Pistoia Alliance**. In May 2023, IBM Venture made an investment in the company of an undisclosed amount. It also organizes the Quantaggle challenges. In September 2024, QunaSys launched QURI Parts in IBM’s Qiskit Functions Catalog, providing chemical and material simulation quantum algorithms [3420]. They also work with Foxconn for the integration in QURI Parts of Foxconn’s fermionic simulation quantum code. They are getting ready for both NISQ and early stage FTQC (eFTQC) [3421].

Qunova Computing

South Korea, 2021, \$10M



Qunova Computing is a company created by June-Koo Kevin Rhee (CEO) which develops quantum software solutions for material and drug discovery.

They developed various quantum chemistry algorithms: the Variational Quantum Approximate Spectral Clustering (VQASC) algorithm which approximates the Fiedler vector for spectral clustering using parameterized quantum circuits with fewer parameters than the dataset size, the Statistical Qubit Freezing (SQF) to improve the scalability and accuracy of adiabatic quantum annealers by fixing qubits likely to converge to specific values, fragment molecular orbital (FMO) methods with variational quantum eigensolvers (VQE) to simulate large molecules efficiently and, at last, the HiVQE (for “Handover Iteration VQE”) algorithm that reduces resource requirements by removing Pauli string measurements, achieving chemical accuracy with fewer qubits, which was demonstrated on a 20-qubit IQM QPU to achieve a chemical accuracy below 1.6 millihartrees with three different geometries of lithium sulfide (Li₂S). It became available on IBM Quantum Network in March 2025. The company has a European headquarters based in Copenhagen, Denmark.

Qutwo

Finland, 2026, €25M



Qutwo was cofounded by Kuan Yen Tan (IQM co-founder), Peter Sarlin (who created and sold Silo AI

to AMD for \$665M in 2024) and Kaj-Mikael Björk (co-founder of Silo AI). It created QuTwo OS, an orchestration middleware layer to route workloads across classical, quantum, and hybrid computing architectures, including quantum-inspired classical compute. It is also developing Miles, a classical open-source framework for large-scale post-training reinforcement learning of advanced AI models.

QxBranch

USA, 2014, \$8.5M



QxBranch was created by former Lockheed Martin employees. It offers solutions, probably tailor-made, for the financial, insurance, aerospace and cyber security markets. Based in Washington DC, they already have offices in Hong Kong, London and Adelaide, Australia. They are partners of D-Wave and IBM. The startup was acquired by Rigetti (USA) in July 2019.

Racine AI

France, 2024



Racine AI develops open-source datasets and vision-language models for specialized scientific and technical domains. Its quantum datasets include VDR_Quantum_Papers, which contains about 6,800 samples extracted from scientific papers on quantum circuits, including diagrams, equations and architectures, and VDR_Quantum_Synthetic, containing about 4,000 synthetic quantum-circuit samples designed for training AI models on technical diagrams.

Rahko

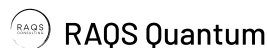
UK, 2018, £1.3M



Rahko is a quantum machine learning and chemical simulation software development company based in London. It was founded by Leonard Wossnig. They are among the first Amazon AWS partners for the use of quantum resources in the cloud, and the first in Europe. In May 2020, they announced that they would work with Merck on “quantum inspired” algorithms, i.e. on conventional computers. In 2021, they announced that they were collaborating with Honeywell and achieved excellent accuracy executing a Discriminative Variational Quantum Eigensolver algorithm on Honeywell’s H0 6-qubit trapped ion system. The company was acquired by **Odysey Therapeutics** in January 2022, a company specialized in drug discovery in the fields of inflammatory diseases and cancers.

RAQS Quantum

Singapore, 2025



RAQS Quantum is a quantum technology consulting company developing quantum-inspired algorithms [3422, 3423], creating a quantum-inspired edge-processing chip and offering training on quantum AI. It was created

by Raghunath Koduvayur (CEO) along with Rainer Dumke (NTU, known for his work on superconducting qubits) and Dario Poletti (CQT).

ReactiveQ

Canada, 2018-2021



ReactiveQ develops quantum simulation algorithms for the design of innovative materials such as high-temperature superconductors, all on a NISQ quantum computer.

Riverlane

UK, 2016, \$120.7M



Riverlane is a spin-off from the University of Cambridge whose main business is to provide quantum error correction solutions to QPU vendors. The company had a staff of 180 as of July 2026.

Error syndrome decoding. They develop Deltaflow.Decode, a software quantum error-syndrome decoder. It was complemented by a dedicated quantum error correction chip announced in September 2023, which will implement a surface code at the error syndrome decoder level with a Xilinx Ultrascale+ FPGA and a 12 nm ASIC, the DD0A [3424, 3425]. The FPGA and ASIC implementations both decode the surface code syndrome and are compared. The ASIC version is particularly energy frugal, with an expected consumption of only 7.85 mW for a distance 23 surface code. It was demonstrated that it could operate in real-time on a Rigetti QPU in October 2024 with fast gates of 8 to 60 ns and 84 physical qubits [3426], and in a 2025 Nature Communications paper [3427].

They also have Deltaflow.Control, a high speed pulse sequence to control qubits using off-the-shelf hardware. Finally, Aqueduct is an automation and administration environment to run quantum circuit experiments. It seems to be an evolution of Deltaflow.OS which is dedicated to NISQ systems and optimizes access to hardware resources for qubit control. It was deployed in mid-2020 at several sites in the UK and in partnership with SEEQC and, later, CQC (Quantinuum). They even got a £2.1M Horizon Europe grant in 2024 for the expansion of their QEC decoder. They have multiple research [3428] and industry partnerships going on, including one with Alice & Bob that was announced in May 2024. In its roadmap announced in July 2024, Riverlane planned to support MegaQuOp FTQC QPUs by the end of 2026, a target restated in its March 2026 roadmap update (no FTQC hardware supports this as of September 2026). In a 2026 PRX Quantum, they did show that when accounting for their better qubit connectivity and impact on quantum error correction and fault-tolerance time overheads, slow qubit platforms like cold atoms and trapped ions may end-up being nearly as fast as faster qubit platforms, at least with relatively large circuits [3429].

In October 2024, Riverlane published its Quantum Error Correction Report with a good description of how it works and quotes from industry and academic specialists [3430]. It was updated in November 2025 [3431]. It was completed by their technology published in March 2026 which spans a period going till 2033, and targets the support of MegaQuOp and GigaQuOp by 2032 and a TeraQuOp by 2033 inline with some vendors roadmap like IBM [3432]. They reasonably envision the advent of utility-scale FTQC by 2033.

In February 2025, they launched the SurgeonQ project along with IQM and Zurich Instruments to develop the first QPU executing multiple real-time quantum error correction (QEC) operations [3433]. In July 2025, Deltaflow was deployed at the CentreSquare data center alongside OQC hardware as part of the UK DECIDE project, with Qblox, Infleqtion and Rigetti, the first commercial QEC deployment in the UK. In September 2025, Deltaflow 2 was installed at Oak Ridge National Laboratory next to the Frontier supercomputer, the first real-time QEC system integrated with a top-tier HPC. They also released Deltakit, a software platform helping developers learn and adopt QEC, funded partly through a Deltakit Community Fund run with the Unitary Foundation. Their partners include Rigetti, Alice & Bob, QuEra, Infleqtion, Atlantic Quantum, ORNL and the UK NQCC. They opened a Delft office in the Netherlands, on top of Cambridge and Boston.

Meaning, QPU running algorithms with one million gate operations, which translates into logical qubit error rates around 10^{-6} . This will obviously depend on the capability of the QPU vendors to deliver such hardware, Riverlane providing one of the enabling technologies enabling this feat.

Aqueduct. It is a Python SDK-based software platform aimed at running quantum experiments, focused on qubit control.

In 2025, Riverlane proposed a soft-information-based technique using analog-like measurement data to improve quantum error suppression performance and reduce overhead in NISQ devices, with a 10% improvement on error suppression for superconducting qubits and 20% for neutral atoms [3434].

QECi. Riverlane launched its Quantum Error Correction Interface (QECi) in March 2025 [3435]. It is an open-source communication protocol and physical-layer standard designed to connect classical quantum control electronics directly to dedicated real-time QEC decoding architectures. It standardizes low-latency, high-throughput data streaming between qubit read-out electronics and decoding engines such as Riverlane's Deltaflow system. QECi mitigates the decoding backlog problem across diverse qubit types. In September 2026, Riverlane and Altera validated a reference implementation of QECi on Agilix 7 FPGAs. Btw, Altera parted ways with Intel in March 2026.

Quantum algorithms. Meanwhile, the company also provides services in quantum computing and develops new algorithms combining machine learning and quantum in chemistry, even optimizations [3436] and other fields (with Merck, Rolls Royce, ...). They created with dividiti Ltd, a one man shop created by a certain Grigori Fursin, the Quantum Collective Knowledge, a benchmark SDK for quantum hardware and software. In July 2021, Riverlane created a consortium with **As-tex Pharmaceuticals** and **Rigetti UK** to develop quantum drug discovery algorithms running on Rigetti platforms in the cloud and to improve QPU and HPC integration [3437]. This was part of an 18-month feasibility study funded by a grant from UKRI as part of the UK quantum plan [3438]. They work with the **MIT Plasma Science and Fusion Center (PSFC)** for the simulation of fusion energy plasma. In 2026, they also published a paper on the way to optimize gate synthesis for small angle rotation gates [1571].

RQuanTech

Switzerland, 2018



RQuanTech develops RTranscender, a quantum machine learning tool for finance, healthcare, automotive, seismology and cyber security. It supports Fourier transforms, qubit-based arithmetic operations which can help craft oracles (additions, multiplications, divisions, exponentials), factorization, discrete logs, etc.

Sanctuary

Canada, 2021



Sanctuary is a startup created by Geordie Rose, the cofounder and first CTO of D-Wave.

It wants to implement reinforcement learning algorithms with quantum computing. At this point, the company is hiring a bunch of scientists and developers, with no offering yet looming.

SavantX

USA, 2015



SavantX was created by Ed Heinbockel and David Ostby who in the past worked for the FBI, the Department of Defense and the US Intelligence Community including the DIA where they developed OSINT collection tools (open-source intelligence, using unstructured data).

Their core skills are data search, discovery, analysis and visualization. They plan to leverage quantum computing to identify hidden patterns in data and help optimize complex systems. One of their first realizations in that domain was implemented for the Port of Los Angeles in 2020, to optimize cargo flow and container handling using D-Wave quantum annealers and classical machine learning [3439]. Their target markets are the nuclear, healthcare, utilities and defense industries.

Schrödinger

USA, 1990, \$193M



Schrödinger is a digital drug design company, mainly using molecule screening and drug repurposing. It is an established competitor of Qubit Pharmaceuticals (France). They work with Sanofi. The company is listed on NASDAQ. They inevitably became interested in quantum computing and have started a partnership with Qu&Co to ramp up their skills in quantum computing.

Semicyber

USA, 2018



Semicyber develops algorithms in various fields: data analysis (non-quantum), quantum and others for critical applications for the USA defense sector, particularly the US Air Force.

So, they are probably closer to the service company than to the product-oriented startup. The startup is co-founded and managed by Kayla Farrow, an engineer specializing in algorithm creation and signal processing.

SHYN

Bulgaria, 2016



SHYN develops solutions for the visualization of data coming from quantum calculations. So, some quantum dataviz! One use case involves detecting quantitative fake news. It was co-funded by Google's Digital News Information Fund dedicated to the press. This €150M fund distributed funding of a few €100K to more than 400 projects in Europe.

Sigma-i Labs

Japan, 2019, \$3.7M



Sigma-i Labs is a consulting company and private laboratory that grew out of the Tohoku University Quantum Annealing Computing Research Laboratory, based in Sendai and led by their CEO Masayuki Ozeki. They started by doing consulting around the creation of software for D-Wave's annealers [3440], using their cloud Leap platform since 2019 [3441], and particularly in the telecommunications industry [3442] and for manufacturing with some job shop scheduling algorithm [3443].

Snowdrop

Canada, 2024



Snowdrop is a quantum software company created by Geordie Rose, the cofounder and former CTO of D-Wave. He wants to develop practical solutions for D-Wave quantum annealers, involving quantum machine learning.

Softquantus

Estonia, 2024



Softquantus is a company created by Roytman Piccoli that is registered in Estonia but based in France. It develops some software for tracking quantum computing usage in the cloud.

SolidState.AI

Canada, 2017



SolidState.AI develops machine learning solutions for the industry covering yield improvement, production calibration and predictive maintenance. All of this is based on hybrid classical/quantum algorithms. They work with Bosch, Applied Materials, Mercedes-Benz as well as with D-Wave, Rigetti, Microsoft and IBM Q, among others.

Spin Quantum Tech

Colombia, 2018



Spin Quantum Tech develops quantum algorithms in the field of cybersecurity that combine AI and quantum. They seem to create PQC (post-quantum cryptography) that exploits new encryption algorithms. They are also working on chemical simulation, which has nothing to do with it.

SQK

USA, 2024



SQK is a US-Korea startup developing quantum-classical algorithms for medical image reconstruction. Their QMEDIC software solution reduces noise and artifacts, improves contrast and edge fidelity, and recovers clinically meaningful detail. They combine AI reconstruction with physics-aware modelling to enforce consistency with acquisition and scanner characteristics [3444]. They also created 'Quantum Workbench Services' to enhance and orchestrate the workflow of classical computational applications with quantum computation. QUKKOS is a multi-layer transpiler that implements an adaptive backend into existing application environment. It is compatible XACC, OpenQASM and CUDA. IBM invested in the company in February 2026.

SoftwareQ

Canada, 2017



SoftwareQ offers development software for quantum computing including a compiler (staq [3445]), emulator (Quantum++ or qpp [3446]) and various optimization tools to reduce the number of gates like T gates in quantum arithmetic [141, 3447]. The company was co-founded by Michele Mosca (on top of evolutionQ) and Vlad Gheorghiu of the Canadian Institute of Quantum Computing.

Strangeworks

USA, 2018, \$28M



Strangeworks develops quantum software. Like many colleagues, they target the aerospace, energy, finance and healthcare markets. They are at the origin of the creation of a Q&A site on [quantum computing, Quantum Computing Stack Exchange](#). The company was founded by William Hurley aka whurley, and is based in Austin, Texas, with a staff of about 20 as of April 2023.

In 2019, they launched a beta of their multi-platform development environment for quantum applications supporting quantum computers or emulators from Rigetti Computing (Forest), DWave (Leap), Microsoft (Q#), Google (Cirq), IBM (Qiskit) and Inflektion (ColdQuanta) Hilbert gate-based cold atom QPU (as of late 2021).

This environment seems to facilitate collaborative work and sharing of results. In February 2021, it turned into an initiative to “humanize quantum computing”. In August 2025, Strangeworks acquired **Quantagonia** (Germany) and its HybridSolver orchestration engine, adding offices in Frankfurt and Munich. It also expanded into India and Sri Lanka in July 2025, partnered with Tech Mahindra in November 2025 and launched Strangeworks Aura, an enterprise optimization platform, in July 2026. It is the international reseller for NEC’s quantum-inspired vector annealing systems and distributes Toshiba’s SQBM+. Its investors include IBM, Hitachi Ventures, RTX Ventures, Lightspeed Venture Partners and Great Point Ventures. It is based on Strangeworks QS (Quantum Syndicate) which consolidates quantum hardware vendors solutions and software tools, Strangeworks QC (Quantum Computing), a free quantum computing ecosystem to learn quantum code using common quantum programming languages and Strangeworks EQ (Enterprise Quantum), an enterprise infrastructure solution consolidating QC and QS with better security, IP protection, quantum machine access, resource aggregation, custom integrations, private deployments, project management and the like. In January 2022, Strangeworks announced a partnership with **Entangled Networks** (Canada) to support their future Multi-QPU Computers Entangled Networks and the associated MultiQopt code compiler. This scale-out approach to quantum computers is of course to be first validated at the hardware level and with physical (optical) connections between several QPUs before becoming operational.

Stratum.ai

Canada, 2018



Stratum.ai develops a quantum software dedicated to a very specialized market, the optimization of mineral prospecting, particularly in gold.

Super.tech

USA, 2020, \$150K



Super.tech is a startup launched by Pranav Gokhale, Fred Chong and Teague Tomesh that develops a software stack dedicated to the control of quantum computing systems ranging from a hundred to a thousand qubits. The solution is the result of the NSF-funded Practical-Scale Quantum Computation (EPiQC) research project involving five Chicago-area and MIT universities and stars such as Peter Shor and Aram Harrow. It is creating a software infrastructure targeting the development of NISQ solutions. The company was acquired by ColdQuanta in May 2022.

SuperQ Quantum

Canada, 2022, IPO



SuperQ Quantum modestly presents itself as the “*Chat-GPT Equivalent of Quantum Computing and Super-computing*”. They built a quantum computing software platform, Super, that supports quantum annealing, gate-based quantum computing and HPCs. It automates problem solving and orchestrates the distribution of computing tasks across quantum and classical hardware. Users have access to a natural language interface to formalize their problems. Their first customer announced in July 2025 is Verge Agriculture, which works on an agricultural quantum optimization project involving robotic motion planning, running on a D-Wave annealer. The platform was formally launched at the CES 2026 in Las Vegas.

The company was initially a bitcoin mining company, Atco Mining. It changed name in 2025 to SuperQ after acquiring the assets of Staque Computing. The CEO of the SuperQ is Muhammad Ali Khan who was also the founder and CEO of Staque, and the senior VP of innovation at Verge Agriculture. It demonstrates how self-help is the best help!

Terra Quantum

Switzerland, 2019, \$85.8M



Terra Quantum was cofounded by Markus Pflitsch (CEO) who was later joined by Valerii Vinokur (co-CTO) and Florian Neukart (co-CTO). Their \$75M funding helped them expand their R&D effort and also, their international expansion, with the opening of new offices in Munich and in the Silicon Valley with over 200 permanent staff in total (2024). In April 2026, they signed a non-binding letter of intent with Mountain Lake Acquisition Corp. II valuing the company at \$3.25B, but the exclusivity period lapsed without a definitive agreement. On May 2026, they instead signed a definitive business combination agreement with another SPAC, Axiom Intelligence Acquisition Corp 1, valuing Terra Quantum at about \$3.5B, with the combined company

expected to trade on Nasdaq and the deal targeted to close in the second half of 2026.

The company develops quantum software solutions in all possible quantum fields: quantum computing, quantum cryptography and quantum sensing. They modestly position themselves as building the “*European quantum ecosystem*” [3448] and working on the “*development of revolutionary quantum computing applications*”, with supporting the development of more efficient fertilizers, batteries and power grids.

Quantum computing. Their “Quantum Algorithms as a Service” is an algorithms library with the usual optimizations and QML pieces. The company then customizes it according to the needs of their customers, making them a mixed product/services company. They also develop classical and hybrid chemical process algorithms [3449].

In January 2022, Terra Quantum launched an alpha release of its QMware hybrid quantum cloud data center (QMware), in partnership with Novarion Systems (Austria, covered earlier in this section). Their HPC and quantum emulation workloads are powered by Intel Xeon Platinum CPUs and Nvidia A100 GPGPUs and DGX servers. Their CPU-GPU-QPU interconnect is based on CXL (Compute Express Link), a unified in memory communication open standard protocol created in 2019 by Intel for high-speed CPU-to-device and CPU-to-memory connections, designed for HPCs and built on the PCI Express (PCIe) interface. Their hybrid quantum computing approach is described in a white paper where they claim it outperforms classical approaches [3450]. They also propose to implement the platform with some flexible virtual classical setups [3451].

Their architecture is based on a memory-centric compute architecture that supports processing with QPUs, CPUs and GPUs as well as hybrid quantum computing with 12 TB of NVRAM shared by all systems. They also developed a unified information theoretical model for classical and quantum information that allows for efficient QPU emulation via a hardware agnostic intermediate representation of the quantum circuits. In 2022, they launched QUANTON-HGX2, a new generation of servers with Nvidia GPGPU and AMD EPYC CPUs. It turned into a more formal partnership with Nvidia in April 2024 for the development of data analytics solutions. In 2023, they benchmarked their QMware emulator against AWS’s emulators on a QML use case [1148].

In 2026, they proposed Soft-Quantum Algorithms, a method to train a variational ansatz circuit matrix directly and faster, but only for a few qubits (5) [3452].

Quantum inspired. They also propose quantum inspired algorithms design services. For example, with computational fluid dynamics for Evonik [3453] and for derivative options pricing for BBVA [3454]. In 2025, a hybrid quantum/classical algorithm to predict the temperature of steel production blast furnaces was also

nearly entirely classical, given the quantum part was using only 6 physical qubits [3455].

No-code. They launched in April 2025 a “closed” beta of TQ42 Studio, with QAI Hub, a no-code quantum machine learning platform, and Qode Engine, a Python SDK, to support hybrid quantum-classical AI development.

Classical algorithms. In May 2023, Terra Quantum acquired **divis intelligent solutions** (Germany) which brings some knowledge in classical algorithms. In 2024, they expanded their stakes in Asia and particularly in South Korea with a partnership with **MegazoneCloud Corporation** for the development of quantum computing algorithms and applications, and another one with **POSCO Holdings** for the development of steel manufacturing optimization algorithms. They also develop quantum inspired classical algorithms, like in quantum chemistry [3456].

Cryptography. In December 2023, Terra Quantum launched TQ42 Cryptography, an open-source PQC library and set of algorithms and related tools. The library will be expanded with Entropy-as-a-Service and Quantum Keys-as-a-Service [3457]. They already provide a photonic based QRNG creating 1.2 Mb/s of random bits and are also working on developing QKD solutions. They won in 2024 a project to combine PQC and QKD for the US Air Force [3458]. Their “Quantum Security as a Service” offer uses a QKD solution, given the physical architecture and related hardware are not specified.

Hardware. They also have side pet-projects in the development of qubits, like cuprate-based superconducting qubits in the “flowermon” category along with many other research labs like MPI in Germany and Sapienza Rome in Italy. They are also studying with Aalto University in Finland the development of more efficient superconducting qubits based QPUs.

Tinubu Software

France, 2000

The logo for Tinubu Software, featuring the word "tinubu" in a lowercase, bold, sans-serif font. The letters are black, with the 'i' and 'u' having a slight dot and tail respectively.

Tinubu Software is a credit insurance software company that is investigating the usage of quantum computing to extend its offering to improve prediction on time series. They participated in the Classiq 2022 quantum computing coding challenge and were the bronze winners with Thomas Frossard, Ayoub El Qadi, Quoc Viet Nguyen and Marcelin Gallezot.

Tokyo Quantum Computing

Japan, 2017

The logo for Tokyo Quantum Computing, featuring the words "Tokyo Quantum Computing" in a stacked, sans-serif font. The text is colored: "Tokyo" is blue, "Quantum" is orange, and "Computing" is green.

Tokyo Quantum Computing wants to develop quantum annealing computer simulation software like many of the software startups from Japan.

Tradeteq

UK, 2016, \$29.9M



Tradeteq is a financial trading platform that uses AI for risk assessment and portfolio optimization. Their ambition is to use quantum computing to develop their own quantum tools. In April 2020, they announced that they would work in this direction with the Singapore Management University (SMU) and with quantum neural network algorithms.

TraverseQuantum

China, 2025



TraverseQuantum is a Beijing startup, that design architectures for FTQC, real-time QEC and full-stack system engineering. It was created by the former Alibaba DAMO Academy quantum lab along with Tsinghua University researchers. It was created by Jianxin Chen (CEO), Zhu Xing (CTO) and Chen Changming (COO).

Their technology covers five layers: superconducting quantum processor design, quantum design automation, architecture and error correction (instruction set, microarchitecture, real-time QEC), control electronics, and compilers. Their first product is a control-to-GPU low-latency interconnect module linking quantum control electronics to GPU platforms through LVDS/Aurora interfaces and a 100G RDMA bidirectional channel, supporting real-time decoding and control feedback. They developed a parallel neural decoder paper, benchmarked on the Zuchongzhi 3.2 processor [3459]. Their Bunny codes uses native CNOT plus CXSWAP gates to run qLDPC codes on square-grid connectivity [3460].

Turing

USA, 2016



Turing is a company created by Seth Lloyd and Michele Reilly that wants to solve key societal problems with hybrid classical-NISQ software solutions using AI and quantum machine learning techniques.

Seth Lloyd is a famous prolific quantum scientist in quantum computing and Michele Reilly has been working on qRAM.

Type one systems (2022, Denmark) is developing an unspecified quantum sensor which enables scientists to study quantum systems and fields. It does high-precision measurements at the single-quantum level [3461]. Their name reflects their mission which is to build systems that help the transition to a type one civilization in Kardashev's scale. Meaning, capturing all the solar energy received by the Earth.

Unitary Foundation

USA, 2018



Unitary Foundation, previously Unitary Fund, is a kind of equivalent of the Mozilla Foundation for quantum

technologies. It is a non-profit organization creating open-source quantum libraries, tools, hardware and content [3462].

They fund through a microgrant program (\$4K) developers. They sponsored about 20 projects including the error mitigation framework mitiq [3463], Qrack (emulator accelerator on GPUs), OLSQ (Optimal Layout Synthesizer for Quantum Computing, a pre-compiler optimizer reducing the SWAP gates count), a quantum machine training textbook and Pulser, developed by Pasqal. They are supported by IBM, AWS, Pasqal, Cisco, Quandela, the DoE, the NSF and others.

Walrus Computing

Germany, 2025



Walrus Computing

Walrus Computing is a company created by Peter-Jan Derks, Johannes Frank, Paul K. Faehrmann, and Frederik Wilde that develops compilation tools for FTQC hardware. They collaborated with Quandela in 2026 [3464]. They also developed Piper Draw, an open-source web application for building pipe diagrams for topological quantum error correction.

WYW

France, 2025



WYW is a one-man shop created by a quantum physicist, Mohammad Haidar, who covers the fields of PQC, AI and quantum algorithms. He also created the OpenVQA Hub, a community built around an open-source platform that includes OpenVQA, OpenVQE, and OpenHybrid, which support IBM's Qiskit, Bull's MyQLM, and Google's Cirq [3465].

Yiyaniq

Canada, 2025



Yiyaniq is a company based in Toronto created by Behnam Javanparast (CEO) and Estelle Inack (CTO) that develops portfolio optimization solutions with "quantum intelligent algorithms".

Zapata Quantum

USA, 2017, \$82.4M



Zapata Quantum is a quantum software and services company founded by Harvard researchers including Christopher Savoie and Alan Aspuru-Guzik from the University of Toronto who has developed many founding algorithms in chemistry quantum applications.

Their partners included Google, Rigetti, IonQ, D-Wave (on machine learning) and IBM. Also, Bosch (Germany) was one of their corporate investors. Honeywell invested in the company in March 2020. The company established an office in the UK in June 2021. They initially developed a complete quantum operating system serving as a hub between application algorithms and quantum accelerators of all types. In April 2020, this took the

form of Orquestra, a platform for managing quantum application workflows, released in 2021, with:

- **Building code** using their Orquestra Studio. It offered a set of code libraries supporting Cirq (Google), Qiskit (IBM), PennyLane (Xanadu), PyQuil (Rigetti), Q# (Microsoft) as well as pyAQASM (Atos/Eviden).
- **Orchestrate** its deployment across multiple QPU and classical emulation platforms using Zapata Computing Quantum Workflow Language (ZQWL), which is YAML-compatible and supports various quantum hardware architectures (NISQ, quantum annealing) and classical computing (quantum emulators such as those from Atos/Eviden, supercomputers, cloud servers). YAML is a language that dates back to 2001. It is used to create configuration files. It is used in conjunction with Python. Orquestra included tools for managing batch computing. The Orquestra Data Correlation Service (ODCS) collects treatment data in a MongoDB database which is then exported as Excel tables, a Jupyter notebook or for the Table software.
- **Deploying** the Orquestra runtime locally or on the cloud.

Zapata Computing became Zapata AI in 2023 to pivot into developing classical generative AI solutions using tensor networks. In September 2023, the company announced its IPO via a SPAC for a pre-money valuation of \$200M and began trading on April 1st, 2024. It opened at \$13.6, fell to \$5.7 in a day and was a penny stock as of June 2nd, 2024, at 85c, risking delisting if it stayed under the \$1 water for over 6 months. In Q1 2024, their revenue was quite low, at \$1.22M, with a \$5M loss. In October 2024, the company went bankrupt and closed.

In September 2025, it reemerged after obtaining a \$3 million convertible bridge financing part of its debt and re-branded itself to its initial Zapata Quantum name. It got an additional funding of \$15M in April 2026. In May 2026, its co-founders Yudong Cao and Jonathan Olson returned in the company as CTO and VP of Strategy and Operations.

ZebraKet

Canada, 2022-2025



ZebraKet is a supply chain startup created by Alex Khan (CEO) which was incubated at the Creative Destruction Lab (CDL) and University of Maryland Quantum Startup Foundry Pre-Traction program. It develops various algorithms supporting among other platforms D-Wave annealers. The company was acquired by Aligned IT at an undisclosed date.

3.11 Service vendors

On top of software vendors, the service vendors industry is getting structured to work with end-user customers and help them adopt quantum technologies, mostly focused on quantum computing.

Large IT service vendors like **Accenture**, **Capgemini** and **KPMG** [3466] have launched quantum technology practices, often partnering with hardware and software vendors. Accenture teams up with IonQ, Quantinuum, D-Wave and 1QBit. KPMG is with Microsoft, selling “quantum inspired” projects. Capgemini created its Quantum Lab (Q-Lab, led out of Paris by Julian Van Velzen) early in 2022, partnering with IBM and positioned in Germany, Portugal, India and France. It covers all branches of quantum technologies (computing, telecommunications and cryptography, sensing) and focuses on life sciences, financial services, automotive and aerospace verticals [3467]. **McKinsey** has a quantum practice that is part of its QuantumBlack.AI “excellence center”. **EY Global Services Limited** partners with IBM. **T-Systems** (Germany, the service branch of Deutsche Telekom) also has a quantum computing practice, a T-Lab in Berlin dedicated to quantum technologies and particularly quantum telecommunications and is also partnering with IBM.

Specialized head hunting HR companies also exist in the quantum market like **Quantum Futures** (UK), and **Psirch** (USA), which advocates for the creation of Chief Quantum Officers at the board level in Corporations [3468]. It is a bit extreme to say the least. At least, Corporations should have a CTO in their board.

Here are a bunch of more specialized quantum services and consulting companies.

USA

Protiviti (USA) is a 7,000-people consulting company with a quantum practice focused like many other on the potential applications from quantum computing and on securing IT infrastructures against the potential related threats (so, PQC et al). They are partnering with Multiverse Computing.

Psi-Ontic (USA) is a quantum consulting created in Florida by Alan Martin, a management consultant coming from the automotive industry and with a physicist PhD. background. He works with large end-user customers as well as with investment funds and startups. He also creates market studies and quantum strategy assessments for various government organizations.

Quantum Computing Engineering Inc. (QCE) (2019, USA) is a company created by Gonzalo Florez Giraldo that does consulting for putting in place quantum computing solutions in most addressable markets (optimization, chemistry, healthcare, finance, machine learning). It also addresses cybersecurity and cryptography. The company is based in Houston, Texas.

Blanqet (USA, 2024) was created by Liang Jiang, David Awschalom and Scott Aaronson. This seems to be a front-end company for these academic researchers when they do consulting in the industry.

Emergence Quantum (USA, 2025) is a contract research company created by Microsoft's lab in Australia alumni, David Reilly (CEO) and Thomas Ohki [3469]. Their activities cover control and readout cryogenic electronics applied to scaling quantum computers regardless of qubit modality (spins, trapped ions, atoms, photonics, superconductors), energy-efficient and cryogenic classical computing for data centers and edge platforms (including neuromorphic, reservoir, and analog approaches), and scalable cryogenic sensing interfaces with quantum-limited amplification. It works with companies like Diraq, IonQ, Archer Materials, PsiQuantum and Maybell Quantum [3470–3472].

Gladiolus Veritatis Consulting (USA) is a one-person company created by Vicky Choi that is specialized in quantum optimization algorithms.

Safe Quantum Inc (2020, USA) sells PQC services, **Quantalytics Inc** (1989, USA) is focused on networking security services. **StrategicQC** (USA) is specialized in recruiting talent in quantum technologies. **Quantum Computing Engineering** (USA) or QCE, provides generic consulting services on quantum computing, and **Aspen Quantum Consulting** (USA) does due diligence investigation of startups for investors. **Light-house Disruptive Innovation** provides services on many disruptive technologies including quantum technologies.

UK

Applied Quantum Computing (2022, UK) is a service company created by Tim Thomas and Robert Cumming, with a past in the financial industry, as physicist and developer. Their mission is to help customers get a good understanding of the impact of quantum computing on their business, plan their quantum readiness, explore and experiment practical use cases [3473].

QinetiQ (2001, UK) is a defense contractor with 6,000 employees that created a quantum practice, working in the field of quantum sensing and quantum computing. They contributed to the deployment of an Inflection QPU at NQCC in the UK.

Plantagenet Systems (2014, UK) is a consulting service shop run by Roberto Desimone who runs projects for various UK organizations including their Ministry of Defense, InnovateUK, the UK National QT program, the National Quantum Computing Centre (NQCC) and the UK Quantum Hubs.

And **Q&I** (UK) also called Qandi, **Q4B** (UK), **Quant-Gates** (UK), **Caledonian Photonics Limited** (UK, consulting service in lasers and quantum photonics), **Cystel** (UK, security risk assessments), **RayCal** (UK) which was an analyst firm in quantum technologies, and

Inside Quantum Technology (2018, UK) created by Lawrence Gasman with the help of 3DR Holdings.

Germany

Quant-X Security & Coding (2019, Germany) is a quantum service company created by Xenia Bogomolec (CEO) and Peter Nonnenmann (Scientific Advisor and Analyst). They develop quantum algorithms for quantum annealing and gate-based systems. They also integrate PQC for dynamic partner authentication in QKD networks.

Q-iSIM (2016, Germany) aka “Quantum Interdisciplinary Simulations” develops quantum annealing software in physics simulation and fluids dynamics.

D-fine (2002, Germany) has 1,500 consultants and is participating in quantum computing projects related to DLR.

Quantimize (2024, Germany) was created by 5 students from TUM (Munich) who want to focus on sustainability-related applications. They were the winners of the Quantum Climate Challenge 2022 by Deloitte in which they developed a flight trajectories optimization algorithm that minimizes climate effects. They studied several variants with QAOA and QUBO problem reformulation, a quantum genetic algorithm and a quantum neural network. They also were at the second place of the global Qiskit Hackathon 2021 where they worked on vehicle smart charging optimizations.

Brandan Quantum (2025, Germany) is a one-man quantum service and training shop created by Fintan Bolton [3474].

Qonsultancy a small consultancy shop in quantum technologies created by Jens Eisert from Freie Universität Berlin.

Quantag IT Solutions (2014, Germany) is an IT company created by Sergie Salata (CEO) which develops quantum algorithms, propose PQC deployment services and a lot of classical IT services. Salata proposed a new quantum algorithms classification in 2026 [3475].

Israel

QuantyMize (2022, Israel) is a startup created by Daniel Porat (CEO) and Emanuele Dalla Torre (CSO, Bar Ilan University in Ramat Gan). It created ECO, a quantum inspired algorithm used for various optimization tasks, including power grid optimizations using D-Wave quantum annealers. Its first public customer is Maiora Solutions (Spain), a data-science service company for which they developed Resmart, an AI-based solution for catering and foodtech.

Spain

Qureca (Spain) whose name means Quantum Resources & Careers, and which does training, recruitment, business development and events,

Quanvia

Spain, 2022



Quanvia is a quantum consulting, service, research and training company targeting the usual suspect industries (retail, finance, logistics, energy, transport, automotive, pharmaceuticals, chemistry).

The company spun out of the University of the Basque Country, Bilbao, Spain. It is run by Enrique Solano (President, also cofounder of Kipu Quantum), Javier Mancilla Montero (Managing Partners), Dawoon Choi (Quantum Machine Learning Director) and Ariela Strimling (Quantum Computing Tech Lead). The team is spread in Spain, Chile and the USA. The Latin America expansion came from the acquisition of Nimoy Cognitive Computing, a data-science service shop.

France

QuRISK (2021, France) does consulting on quantum computing originated risks on cybersecurity and also develops quantum algorithms and a technology watch AI engine named oQo, **Q. BPO Consulting** (2020-2026, France) seemed to be a generalist quantum computing consulting shop, **QuantumBCS** (2019-2025, France) is doing unspecified quantum software development on top of Blockchain solutions development and **Expleo Group** (France) a consulting firm seemingly working in quantum chemistry [3476]. **Qounselor** (France, 2024) is a one-man service shop led by Valentin Gilbert. Quantum Momentum is a company created by Igor Vujic, who started by creating a newsletter and is offering education services. **Quantsoc** (France, 2025) is a chemical engineering consulting company created by Wassil Sennane [3477].

Rest of European Union

We have also **Quantum Quants** (the Netherlands) provides services in quantum computing.

Unitary Zero Space (2020, Finland) is a quantum consulting and training services company. Their services portfolio includes helping organizations commercialize their own quantum technologies, educating customers on quantum technologies and cybersecurity expertise. They packaged their training offering in a half-day quantum computing workshop format (that's quite short...). The company was created by Topias Uotila (quantum programming), Risto Hakala (cybersecurity) and Juha Muhonen (quantum physics).

Kvantify (2022, Denmark, €33.3M) is a quantum chemistry specialized company [2325].

Qool (2024, Denmark) is a software development company that formulates problems in QUBO to be run either on QPUs or digital annealers.

Quantum Phi (2018, Czech Republic).

Quantum Technologies AB (Sweden) is specialized in quantum computing use cases in material design.

QbitFlow (2022, Austria) is a service company covering multiple fields (quantum sensing, computing, communications, training, ...).

Qubitsok (Poland) was created in 2025 by Piotr Lewandowski. It is a website that contains a quantum job board fed by 14 custom parsers that crawl more than 20 sources weekly, listing on the order of 1,300 to 2,500 open roles depending across a couple hundred companies in about 40 countries. It also contains a daily arXiv feed covering quant-ph and cs.ET and a salary benchmark tool. Revenue comes from employer fees.

Qdeeptech (Austria), created by Angie Qarry provides a wealth of consulting services in deep technology including quantum technologies.

Sonovero R&D (Poland) is involved in the MIROK project to build a quantum computer in Poland.

QuantumNet (2021, Italy) is a contract research company in quantum computing applications.

Quantum Ket (2024, Italy) combines quantum technologies and artificial intelligence for the development of quantum sensing, communication and computing solutions.

Qubitech (2021, Greece) is a quantum consulting company cofounded by Alexis Askitopoulos (CSO). Not to be confused with QubitTech, a scam crypto trading company that “*uses quantum technology to generate a monthly ROI of 25% for its investors and a total of 250% in 10 months*”.

Qubic QC is a quantum computing consultancy shop created by Joel F. Kremer and based in Albania.

Japan

Mercari R4D (Japan) is a weird company that “*strives to circulate our limited resources to build a future where anyone can unleash their potential*” and “*Mercari R4D is a research organization that focuses on the marketplace of the future in order to implement our findings practically and provide new value to the world*”. They have three researchers in quantum information science working in various fields (quantum Internet, quantum computing, quantum annealing) [3478, 3479].

Quemix (2019, Japan) is an IT company specialized in providing quantum computing solutions in quantum chemistry, computer aided engineering and machine learning. Among its customers are AsahiKASEI, Honda [3480], Mitsui Kinzoku, NGK Insulators, and Sumitomo Rubber Industries. They proposed a quantum algorithm for solving a particular differential equation, the advection-diffusion-reaction equation by employing a novel approximate probabilistic imaginary-time evolution (PITE) operator, which seems to be a derivative from the QITE algorithm [3481]. It has applications in fluid dynamics [450]. They also tested a small scale QPE with a Quantinuum NISQ system with a 1D error detection code [450].

India

DN-Quantum Computing (2019, India) aka DishaNishish Technologies is a service company helping corporations develop quantum computing solutions, noticeably in chemistry and with transmon qubits.

AQMATICS (India) proposes IT cybersecurity and quantum computing services for multiple sectors (finance, life sciences).

QuantQurious Technologies (2025) provides training services for quantum professionals.

AmberFlux (2013) provides quantum computing consulting services on top of existing machine learning and data science services. **QRDLab** does about the same, also in India. **LTIMindtree** is a large system integrator also joining the quantum computing bandwagon, partnering with IBM.

Qwatch proposes many IT services offerings including on quantum computing.

Tata Consulting Services has a branch working on quantum computer. TCS established an AI and quantum competence center in France in January 2025 to service the needs of the aerospace sector [3482].

QuEdX provides business services on quantum technologies for labs and education.

Qrishi

India, 2021



Qkrishi develops quantum models, algorithms and kernels for applications in automotive, finance [3483, 3484] (in that latter case, using up to 14 qubits), agriculture, seismology, signal processing and other areas. The Birla Institute of Management Technology has also partnered with them to create a Quantum Computing course that integrates business and technical elements.

Silicofeller (2021, India) develops quantum solutions and advertises doing it in simulation, optimization and quantum machine learning, which is fine, and then, crosses the line with mentioning the metaverse and Blockchain as application domains which are a bit far-fetched.

eQspanse NextGen Technologies (2024, India) is an IT shop specializing in developing products, providing services, and deploying solutions utilizing Quantum and AI technologies that is focused on space and deep technology.

APAC

Other services companies include **Max Kelsen** (Australia, par of Bain Consulting since 2023), **h-bar Quantum Consultants** (Australia) and **AngelQ Quantum Computing** (Singapore),

Rest of the world

SoftServe (1993, Ukraine) is a generic IT service company running quantum computing projects.

Quanta-ly (2020, Libya) is preparing Libyan industries for the adoption of quantum products and services with training, consulting and advanced secure communications. No country, even in war, can escape quantum technologies!

coreDevX

Chile, 2014



coreDevX was created by Javier Urrutia (CEO) and Victor Toledo (CTO) is a diversified IT service and product company. It distributes SpinQ educational QPUs in Chile and develops their own 20-qubit superconducting qubit QPU based on a chip from SpinQ. They also provide GPU-based emulator clusters (QSimulator 100 and 200), QRNG and QKD solutions from IDQ and Falcondale, a third-party QML solution for the financial sector.

Orientom (2013, South Korea) provides quantum computing consulting and software development services in the specific domain of electricity grids optimization with quantum machine learning. It partners with Quandela in France and EQ Tech Energy in Thailand.

Quantific (2022, Switzerland) provides quantum computing consulting and software development services.

Gradiom Sarl (2022, Switzerland) also provides quantum software development services. It is a one-man shop created by Sébastien Perseguers [3485].

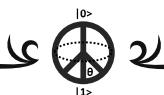
QuGeeks

Switzerland, 2022



QuGeeks is a jobs board and search company specialized in quantum technologies. Their outreach includes “The Quantum Hubs” newsletter and its related social network activity.

Qnow (2024, Brazil-USA) is a consulting company focusing on life sciences, financial services and cybersecurity, with merging AI and quantum computing.



3.11.1 Bibliography and notes

These are the bibliographical references and notes for the **Software and service vendors** subsection.

- [141] [A Polylogarithmic-Depth Quantum Multiplier](#), by Fred Sun and Anton Borisov, arXiv, April 2026 (9 pages).
- [149] [How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits](#), by Craig Gidney and Martin Ekerå, arXiv, April 2021 (31 pages).
- [181] [Demonstrating Record Fidelity for the Quantum Fourier Transform](#), by Philipp Aumann, Michael Fellner, David Alber, Max Cykiert, Christoph Fleckenstein, Roeland ter Hoeven, Leo Stenzel, Riccardo J. Valencia-Tortora, and Wolfgang Lechner, arXiv, April 2026 (6 pages).
- [387] [How to factor 2048 bit RSA integers with less than a million noisy qubits](#), by Craig Gidney, arXiv, May 2025 (40 pages).
- [441] [Geometric Quantum Physics Informed Neural Network](#), by Wai-Hong Tam, Reza Safari, and Hiromichi Matsuyama, arXiv, May 2026 (16 pages).
- [450] [Real and Fourier space readout methods: Comparison of complexity and applications to CFD problems](#), by Xinchu Huang, Hirofumi Nishi, Yoshifumi Kawada, Tomofumi Zushi, and Yu-ichiro Matsushita, arXiv, November 2025 (41 pages).
- [777] [Pushing the Accuracy Limit of Foundation Neural Network Models with Quantum Monte Carlo Forces and Path Integrals](#), by Anouar Benali, Jean-Philip Piquemal, et al., arXiv, April 2025 (16 pages).
- [779] [Quantum Combinatorial Reasoning for Large Language Models](#), by Carlos Flores-Garrigos, Enrique Solano, and Narendra N. Hegade, arXiv, October 2025 (9 pages).
- [913] [Fast, accurate, high-resolution simulation of large-scale Fermi-Hubbard models on a digital quantum processor](#), by Gavin S. Hartnett, Khadijeh Sona Najafi, Aleksei Khindanov, Haoran Liao, Michael Schutzman, Michael R. Hush, Michael J. Biercuk, and Yuval Baum, arXiv, May 2026 (35 pages).
- [1148] [Benchmarking simulated and physical quantum processing units using quantum and hybrid algorithms](#), by Mohammad Kordzanganeh, Markus Pflitsch, Alexey Melnikov, et al., arXiv, June 2023 (21 pages).
- [1169] [Variational Quantum Linear Solver via Block Encoding for the Poisson Equation](#), by Viraj Dsouza, Ayush Singhal, Alex Khan, Rut Lineswala, and Abhishek Chopra, arXiv, August 2026 (14 pages).
- [1247] [Hybrid Quantum Investment Optimization with Minimal Holding Period](#), by Samuel Mugel, Enrique Lizaso, Roman Orus, et al., arXiv, December 2021 (6 pages).
- [1249] [Financial Index Tracking via Quantum Computing with Cardinality Constraints](#), by Samuel Palmer, Roman Orus, Harish Naik, Samuel Mugel, et al., arXiv, August 2022 (8 pages).
- [1565] [Polylogarithmic-depth controlled-NOT gates without ancilla qubits](#), by Baptiste Claudon, Fabrice Debbasch, Alberto Peruzzo, Jean-Philip Piquemal, et al., Nature Communications, July 2024.
- [1571] [More efficient Clifford+T synthesis for small-angle rotations and application to Trotterization](#), by Marius Bothe, Christoph Sünderhauf, Michael J. Witham, Earl T. Campbell, and Nick S. Blunt, arXiv, June 2026 (43 pages).
- [1606] [Connectivity-aware Synthesis of Quantum Algorithms](#), by Florian Dreier, Christoph Fleckenstein, Gregor Aigner, Michael Fellner, Reinhard Stahn, Martin Lanthaler, and Wolfgang Lechner, arXiv, January 2025 (38 pages).
- [1665] [Benchmarking Quantum Computer Simulation Software Packages: State Vector Simulators](#), by Amit Jamadagni, Andreas M. Läuchli, and Cornelius Hempel, arXiv, July 2024 (19 pages).
- [1713] [High Performance Quantum Emulation for Chemistry Applications with Hyperion](#), by Olivier Adjoua, Siwar Badreddine, César Feniou, Igor Chollet, Diata Traore, Guillaume Michel, and Jean-Philip Piquemal, arXiv, April 2026 (15 pages).
- [2246] [Heuristic Quantum Advantage with Peaked Circuits](#), by Hrant Gharibyan, Mohammed Zuhair Mullath, Nicholas E. Sherman, Vincent P. Su, Hayk Tepanyan, and Yuxuan Zhang, arXiv, October 2025 (24 pages).
- [2247] [Efficient Classical Simulation of Heuristic Peaked Quantum Circuits](#), by David Kremer and Nicolas Dupuis, arXiv, April 2026 (10 pages).
- [2248] [Fermionic dynamics on a trapped-ion quantum computer beyond exact classical simulation](#), by Faisal Alam, Gethin Wright, et al., arXiv, October 2025 (54 pages).
- [2249] [Programmable digital quantum simulation of 2D Fermi-Hubbard dynamics using 72 superconducting qubits](#), by Faisal Alam, Gethin Wright, et al., arXiv, October 2025 (98 pages).
- [2325] [Calculating the energy profile of an enzymatic reaction on a quantum computer](#), by Patrick Ettenhuber, Nikolaj Thomas Zinner, et al., arXiv, May 2025 (37 pages).
- [2342] [A joint optimization approach of parameterized quantum circuits with a tensor network](#), by Clara Ferreira Cores, Kaur Kristjuhan, and Mark Nicholas Jones, arXiv, February 2024 (29 pages).
- [2422] [Evaluating higher-order product formulae for molecular ground-state energy estimation](#), by Hiromu Abe, Keita Kanno, Ryosuke Kimura, Masahiko Kamoshita, and Kosuke Mitarai, arXiv, May 2026 (12 pages).
- [2559] [Observing ground-state properties of the Fermi-Hubbard model using a scalable algorithm on a quantum computer](#), by Stasja Stanisic, Ashley Montanaro, et al., Nature Communications, October 2022.
- [2629] [Protein folding on a 64 qubit trapped-ion hardware via counterdiabatic quantum optimization](#), by Alejandro Gomez Cadavid, Martin Roetteler, et al., arXiv, April 2026 (7 pages).
- [2864] [Supply chain logistics with quantum and classical annealing algorithms](#), by Sean J. Weinberg, Randall Correll, et al., Scientific Reports, March 2023 (18 pages).
- [2913] [Solving the Optimal Trading Trajectory Problem Using a Quantum Annealer](#), by Gili Rosenberg, Poya Haghnegahdar, Phil Goddard, Peter Carr, Kesheng Wu, and Marcos López de Prado, arXiv, August 2016 (7 pages).
- [2959] [Quantum Deep Hedging](#), by El Amine Cherrat, Snehal Raj, Iordanis Kerenidis, Abhishek Shekhar, Ben Wood, Jon Dee, Shouvanik Chakrabarti, Richard Chen, Dylan Herman, Shaohan Hu, Pierre Minssen, Ruslan Shaydulin, Yue Sun, Romina Yalovetzky, and Marco Pistoia, arXiv, November 2023 (43 pages).
- [3027] [Quantum artificial vision for defect detection in manufacturing](#), by Daniel Guijo, Roman Orus, et al., arXiv, January 2024 (11 pages).
- [3118] [Introduction to Quantum Computing](#), by William Oliver, MIT, December 2019 (21 slides).
- [3236] [Why quantum software will be eating the world](#), by Yuval Boger, June 2022.
- [3237] [The hard sell of quantum software](#), by Jon Cartwright, 2019.

- [3238] [Collaboration is Dominating Quantum Computing](#), by Russ Fein, The quantum leap, April 2022.
- [3239] [Optimizing Multi-level Magic State Factories for Fault-Tolerant Quantum Architectures](#), by Allyson Silva, Pooya Ronagh, et al., arXiv, March 2025 (25 pages).
- [3240] [Disease Gene Prioritization With Quantum Walks](#), by Harto Saarinen, Mark Goldsmith, Rui-Sheng Wang, Joseph Loscalzo, and Sabrina Maniscalco, arXiv, November 2023 (12 pages).
- [3241] [Quantum Computing for Life Sciences: Revolutionizing Drug Discovery and Healthcare](#), by Jason Gumarang, Algorithmiq news, February 2025.
- [3242] [Dynamical simulations of many-body quantum chaos on a quantum computer](#), by Laurin E. Fischer, Sergey N. Filippov, et al., Nature Physics, January 2026 (32 pages) 📄 (preprint on arXiv).
- [3243] [Practical techniques for high precision measurements on near-term quantum hardware: a Case Study in Molecular Energy Estimation](#), by Keijo Korhonen, Daniel Cavalcanti, et al., arXiv, September 2024 (15 pages).
- [3244] [Photodynamic Therapy as a Driver for Useful Quantum Advantage in Chemistry: Perspective from the Q4Bio program](#), by Q4Bio Team Algorithmiq, Algorithmiq, May 2026 (21 pages).
- [3245] [\$\Delta\$ ADAPT-VQE: Toward Accurate Calculation of Excitation Energies on Quantum Computers for BODIPY Molecules With Application in Photodynamic Therapy](#), by Anton Nykänen, Fabijan Pavošević, et al., arXiv, April 2024 (21 pages).
- [3246] [Simulation of Fermionic circuits using Majorana Propagation](#), by Aaron Miller, Zoë Holmes, Guillermo García-Pérez, et al., arXiv, March 2025 (18 pages).
- [3247] [Loschmidt echo for probing operator hydrodynamics in heterogeneous structures](#), by Algorithmiq team, Algorithmiq website, February 2026 (21 pages).
- [3248] [Quantum Advantage Tracker](#), by Algorithmiq and IBM, GitHub, 2026.
- [3249] [Photonic Quantum Computing For Polymer Classification](#), by Alexandrina Stoyanova, Taha Hammadia, Arno Ricou, and Bogdan Penkovsky, arXiv, November 2022 (22 pages).
- [3250] [Solving a real-world modular logistic scheduling problem with a quantum-classical metaheuristics](#), by Florian Krellner, Abhishek Awasthi, Nico Kraus, Sarah Braun, Michael Poppel, and Daniel Porawski, arXiv, July 2025 (10 pages).
- [3251] [Demonstrating Quantum Scaling Advantage in Approximate Optimization for Energy Coalition Formation with 100+ Agents](#), by Naeimeh Mohseni, Giorgio Cortiana, et al., arXiv, October 2025 (11 pages).
- [3252] [Utilizing Quantum Processor for the Analysis of Strongly Correlated Materials](#), by Hengyue Li, Shenggang Ying, et al., arXiv, August 2024 (15 pages).
- [3253] [Towards an Optimal Hybrid Algorithm for EV Charging Stations Placement using Quantum Annealing and Genetic Algorithms](#), by Aman Chandra, Jitesh Lalwani, and Babita Jajodia, arXiv, April 2022 (6 pages).
- [3254] [300 Logical Qubit Quantum Systems Now Available](#), by Automatski, Automatski news, November 2024.
- [3255] [Demonstration of Scalability and Accuracy of Variational Quantum Linear Solver for Computational Fluid Dynamics](#), by Ferdin Sagai Don Bosco, Dhamotharan S, Rut Lineswala, and Abhishek Chopra, arXiv, September 2024 (9 pages).
- [3256] [t|ket : A Retargetable Compiler for NISQ Devices](#), by Seyon Sivarajah, Silas Dilkes, Alexander Cowtan, Will Simmons, Alec Edgington, and Ross Duncan, arXiv, April 2020 (43 pages).
- [3257] [Quantum-Proof Cryptography with IronBridge, TKET and Amazon Braket](#), by Duncan Jones et al., March 2021.
- [3258] [Practical randomness amplification and privatisation with implementations on quantum computers](#), by Cameron Foreman, Sherilyn Wright, Alec Edgington, Mario Berta, and Florian J. Curchod, arXiv, March 2023 (23 pages).
- [3259] [Foundations for Near-Term Quantum Natural Language Processing](#), by Bob Coecke, Giovanni de Felice, Konstantinos Meichanetzidis, and Alexis Toumi, arXiv, December 2020 (43 pages).
- [3260] [Scalable and interpretable quantum natural language processing: an implementation on trapped ions](#), by Tiffany Duneau, Bob Coecke, et al., arXiv, September 2024 (60 pages).
- [3261] [Efficient Generation of Parameterised Quantum Circuits from Large Texts](#), by Colin Krawchuk, Nikhil Khatri, Neil John Ortega, and Dimitri Kartsaklis, arXiv, May 2025 (26 pages).
- [3262] [A Quantum Platform for Multiomics Data](#), by Michael Kubal and Sonika Johri, arXiv, June 2025 (7 pages).
- [3263] [Bit-bit encoding, optimizer-free training and sub-net initialization: techniques for scalable quantum machine learning](#), by Sonika Johri, arXiv, January 2025 (23 pages).
- [3264] [Benchmarking Quantum and Classical Machine Learning Models on Oncological Data](#), by Sydney Leither, Thomas Lubinski, Michael Kubal, and Sonika Johri, arXiv, August 2026 (11 pages).
- [3265] [Qmod: Expressive High-Level Quantum Modeling](#), by Matan Vax, Peleg Emanuel, Eyal Cornfeld, Israel Reichen- tal, Ori Opher, Ori Roth, Tal Michaeli, Lior Preminger, Lior Gazit, Amir Naveh, and Yehuda Naveh, arXiv, February 2025 (9 pages).
- [3266] [Design and synthesis of scalable quantum programs](#), by Tomer Goldfriend, Yehuda Naveh, et al., arXiv, January 2025 (33 pages).
- [3267] [Benchmark of quantum-inspired heuristic solvers for quadratic unconstrained binary optimization](#), by Hiroki Oshiyama and Masayuki Ohzeki, arXiv, December 2021 (11 pages).
- [3268] [Principles of quantum functional testing](#), by Nadia Milazzo, Olivier Giraud, Giovanni Gramegna, and Daniel Braun, arXiv, September 2022 (13 pages).
- [3269] [Solving nonlinear differential equations on noisy 156-qubit quantum computers](#), by Karla Baumann, Youcef Modheb, Roman Randrianarisoa, Roland Katz, Aoife Boyle, and Frédéric Holweck, arXiv, February 2026 (12 pages).
- [3270] [Non-Markovian Open Quantum Systems](#), by Cesar A. Rodriguez-Rosario and E.C.G. Sudarshan, arXiv, May 2008 (11 pages).
- [3271] [Thermodynamics of quantum coherence](#), by César A. Rodríguez-Rosario, Thomas Frauenheim, and Alán Aspuru-Guzik, arXiv, August 2013 (8 pages).
- [3272] [Non-Markovian quantum processes: Complete framework and efficient characterization](#), by Felix A. Pollock et al., Physical Review A, January 2018.
- [3273] [Near-Term Spin-Qubit Architecture Design via Multipar- tite Maximally Entangled States](#), by N. Paraskevopoulos, PRX Quantum, April 2025 (25 pages).
- [3274] [To Scale Up or To Scale Out: Evaluating Space-Time Costs of Compiled Logical Circuits on Modular Superconducting Quantum Processors](#), by Nikiforos Paraskevopoulos, Sebastian de Bone, Mick Christophersen, Simon Storz, A. Mert Bozkurt, Arno Bargerbos, and Sebastian Feld, arXiv, August 2026 (18 pages).
- [3275] [Opportunities and challenges in scaling quantum error detection on hardware](#), by Yanis Le Fur, Ethan Egger, Hong-Ye Hu, Vincent Russo, William J. Zeng, and Ryan LaRose, arXiv, May 2026 (11 pages).
- [3276] [How Focused Are LLMs? A Quantitative Study via Repet- itive Deterministic Prediction Tasks](#), by Wanda Hou, Leon Zhou, Hong-Ye Hu, Yubei Chen, Yi-Zhuang You, and Xiao- Liang Qi, arXiv, November 2025 (19 pages).

- [3277] [Machine Learning Symmetry Discovery for Integrable Hamiltonian Dynamics](#), by Wanda Hou, Molan Li, and Yi-Zhuang You, arXiv, January 2026 (6 pages).
- [3278] [Machine learning the effects of many quantum measurements](#), by Wanda Hou, Samuel J. Garratt, Norhan M. Eassa, Elliott Rosenberg, Pedram Roushan, Yi-Zhuang You, and Ehud Altman, arXiv, September 2025 (21 pages).
- [3279] [Multi-Objective Optimization and Network Routing with Near-Term Quantum Computers](#), by Shao-Hen Chiew, Chee Wei Nga, et al., arXiv, August 2023 (19 pages).
- [3280] [Information-efficient decoding of surface codes](#), by Long D.H. My, Shao-Hen Chiew, Jing Hao Chai, and Hui Khoon Ng, arXiv, January 2026 (10 pages).
- [3281] [Polyadic Quantum Classifier](#), by William Cappelletti, Rebecca Erbanni, and Joaquín Keller, arXiv, July 2020 (8 pages).
- [3282] [The Fourier Wall: Why Public Tabular Datasets Refuse Quantum Advantage, and a Certified Recipe for Where It Lives](#), by Javier Mancilla and Tomás Tagliani, arXiv, July 2026 (16 pages).
- [3283] [Quantum circuit optimization using quantum Karnaugh map](#), by J.-H. Bae, Paul M. Alsing, Doyeol Ahn, and Warner A. Miller, Scientific Reports, September 2020.
- [3284] [Reducing quantum and classical resources for quantum-centric supercomputing workloads on near-term hardware](#), by Maxence Grandadam, Vladyslav Bohun, Dikshant Dulal, and Maciej Koch-Janusz, arXiv, September 2026 (4 pages).
- [3285] [Universal Dynamics with Globally Controlled Analog Quantum Simulators](#), by Hong-Ye Hu, Susanne F. Yelin, et al., arXiv, August 2025 (12 pages).
- [3286] [Joe Fitzsimons, Horizon Quantum Computing, Abstracting Quantum Computation](#), by Joe Fitzsimons, April 2020.
- [3287] [Post-processing noisy quantum computations utilizing N-representability constraints](#), by Tomislav Piskor, Florian G. Eich, Michael Marthaler, Frank K. Wilhelm, and Jan-Michael Reiner, arXiv, April 2023 (15 pages).
- [3288] [HQS Quantum Simulations: How to survive a Quantum winter](#), by Richard Wordsworth, 2020.
- [3289] [Lifted surgery: Fast processing with QLDPC codes](#), by Lucas Berent, Lawrence Z. Cohen, and Armanda O. Quintavalle, arXiv, September 2026 (38 pages).
- [3290] [The Pinnacle Architecture: Reducing the cost of breaking RSA-2048 to 100 000 physical qubits using quantum LDPC codes](#), by Paul Webster, Lawrence Z. Cohen, et al., arXiv, February 2026 (22 pages).
- [3291] [The Pinnacle Architecture with fixed connectivity of degree eight](#), by Paul Webster, Tom Peham, and Lawrence Z. Cohen, arXiv, September 2026 (8 pages).
- [3292] [Autoregressive neural-network wavefunctions for ab initio quantum chemistry](#), by Thomas D. Barrett, Aleksei Malyshchev, and A. I. Lvovsky, arXiv, January 2022 (8 pages).
- [3293] [An Algorithm for Fast Assembling Large-Scale Defect-Free Atom Arrays](#), by Tao Zhang, Xiaodi Li, Hui Zhai, and Linghui Chen, arXiv, April 2026 (6 pages).
- [3294] [Bias-field digitized counterdiabatic quantum optimization](#), by Alejandro Gomez Cadavid, Physical Review Research, April 2025.
- [3295] [Bias-Field Digitized Counterdiabatic Quantum Algorithm for Higher-Order Binary Optimization](#), by Sebastián V. Romero, Enrique Solano, Narendra N. Hegade, et al., Nature Communications Physics, August 2025 (12 pages).
- [3296] [Comparing Quantum Annealing and BF-DCQO](#), by Pau Farré, Erika Ordog, Kevin Chern, and Catherine C. McGeoch, arXiv, September 2025 (6 pages).
- [3297] [Quantum coherence and counterdiabatic quantum computing](#), by Raziél Huerta-Ruiz, Maximiliano Araya-Gaete, Diego Tancara, Enrique Solano, Nancy Barraza, and Francisco Albarrán-Arriagada, arXiv, April 2025 (8 pages).
- [3298] [Digital-analog quantum computing of fermion-boson models in superconducting circuits](#), by Shubham Kumar, G. Alvarado Barrios, et al., npj Quantum Information, March 2025.
- [3299] [Recent quantum runtime \(dis\)advantages](#), by J. Tuziemski, B. Gardas, et al., arXiv, October 2025 (22 pages).
- [3300] [Hybrid Sequential Quantum Computing](#), by Pranav Chandarana, Sebastián V. Romero, Alejandro Gomez Cadavid, Anton Simen, Enrique Solano, and Narendra N. Hegade, arXiv, October 2025 (14 pages).
- [3301] [Quantum Reasoning for Large-Language Models \(QR-LLM\): Quantum Intelligence](#), by Carlos Flores Garrigos and Narendra Hegade, Kipu Quantum GmbH, October 2025.
- [3302] [Digitized Counterdiabatic Quantum Sampling](#), by Narendra N. Hegade, Enrique Solano, Paolo A. Erdma, et al., arXiv, October 2025 (18 pages).
- [3303] [Scaling advantage with quantum-enhanced memetic tabu search for LABS](#), by Alejandro Gomez Cadavid, Pranav Chandarana, Sebastián V. Romero, Jan Trautmann, Enrique Solano, Taylor Lee Patti, and Narendra N. Hegade, arXiv, November 2025 (9 pages).
- [3304] [Digitized Counterdiabatic Quantum Feature Extraction](#), by Anton Simen, Enrique Solano, Qi Zhang, et al., arXiv, October 2025 (10 pages).
- [3305] [Analog Quantum Feature Selection with Neutral-Atom Quantum Processors](#), by Jose J. Orquin-Marques, Carlos Flores-Garrigos, Alejandro Gomez Cadavid, Anton Simen, Enrique Solano, Narendra N. Hegade, Jose D. Martin-Guerrero, and Yolanda Vives-Gilabert, arXiv, October 2025 (7 pages).
- [3306] [Quantum Feature Selection with Higher-Order Binary Optimization on Trapped-Ion Hardware](#), by Carlos Flores-Garrigós, Anton Simen, Qi Zhang, Enrique Solano, Narendra N. Hegade, Sayonee Ray, Claudio Giroto, Jason Iaconis, and Martin Roetteler, arXiv, April 2026 (7 pages).
- [3307] [Kipu's Roadmap Towards Commercial Quantum Advantage](#), by Kipu Quantum, September 2024.
- [3308] [Runtime Quantum Advantage with Digital Quantum Optimization](#), by Pranav Chandarana, Enrique Solano, Narendra N. Hegade, et al., arXiv, May 2025 (14 pages).
- [3309] [Runtime Quantum Advantage with BF-DCQO](#), by Pranav Chandarana et al., Kipu Quantum blog, April 2025.
- [3310] [Branch-and-bound digitized counterdiabatic quantum optimization](#), by Anton Simen, Sebastián V. Romero, Alejandro Gomez Cadavid, Enrique Solano, and Narendra N. Hegade, arXiv, April 2025 (7 pages).
- [3311] [Branch-and-bound digitized counterdiabatic quantum optimization](#), by (Anton Simen et al), Kipu Quantum blog, April 2025.
- [3312] [Sequential Quantum Computing](#), by Sebastián V. Romero, Alejandro Gomez Cadavid, Enrique Solano, and Narendra N. Hegade, arXiv, June 2025 (4 pages).
- [3313] [RGB Image Classification with Quantum Convolutional Ansatzes](#), by Yu Jing, Hua Xu, et al., arXiv, February 2022 (17 pages).
- [3314] [Release of the world's first "cloud native" quantum chip design service cloud platform](#), by iMedia, iMedia, 2021.
- [3315] [MiqroForge: An Intelligent Workflow Platform for Quantum-Enhanced Computational Chemistry](#), by Jianan Wang, Wenbo Guo, Xin Yue, Minjie Xu, Yueqiang Zheng, Jingxiang Dong, Jiarui Hu, Jian Xia, and Chuixiong Wu, arXiv, August 2025 (34 pages).
- [3316] [Comparative Studies of Quantum Annealing, Digital Annealing, and Classical Solvers for Reaction Network Pathway Analysis and mRNA Codon Selection](#), by Milind Upadhyay and Mark Nicholas Jones, arXiv, September 2025 (56 pages).
- [3317] [Resource efficient method for representation and measurement of constrained electronic structure states with a quantum computer](#), by Kaur Kristjuhan and Mark Nicholas Jones, arXiv, March 2023 (18 pages).

- [3318] [Multi-disk clutch optimization using quantum annealing](#), by John D. Malcolm, Roman Orus, Samuel Mugel, et al., arXiv, August 2024 (11 pages).
- [3319] [Quantum-Inspired Tensor Neural Networks for Partial Differential Equations](#), by Raj Patel, Samuel Mugel, Roman Orus, et al., arXiv, August 2022 (14 pages).
- [3320] [Quantum portfolio value forecasting](#), by Cristina Sanz-Fernandez, Rodrigo Hernandez, Christian D. Marciniak, Ivan Pogorelov, Thomas Monz, Francesco Benfenati, Samuel Mugel, and Roman Orus, arXiv, November 2021 (5 pages).
- [3321] [CompactifAI: Extreme Compression of Large Language Models using Quantum-Inspired Tensor Networks](#), by Andrei Tomut, Samuel Mugel, Roman Orus, et al., arXiv, May 2024 (5 pages).
- [3322] [Block removal for large language models through constrained binary optimization](#), by David Jansen, Roman Rausch, David Montero, and Roman Orus, arXiv, January 2026 (7 pages).
- [3323] [Multiverse Computing and Moody's Analytics Announce QFStudio SaaS Quantum Platform](#), by Multiverse, December 2023.
- [3324] [Comparative Study of Quantum-Circuit Scalability in a Financial Problem](#), by Jaewoong Heo and Moonjoo Lee, arXiv, April 2024 (19 pages).
- [3325] [Parallel iQCC Enables 200 Qubit Scale Quantum Chemistry on Accelerated Computing Platforms Surpassing Classical Benchmarks in Ruthenium Catalysts](#), by Seyyed Mehdi Hosseini Jenab, Brandon Henderson, and Scott N. Genin, arXiv, March 2026 (14 pages).
- [3326] [Towards Quantum Advantage in Chemistry](#), by Scott N. Genin, Ilya G. Ryabinkin, Michael G. Helander, et al., arXiv, March 2026 (27 pages).
- [3327] [Estimating Phosphorescent Emission Energies in Ir\(III\) Complexes using Large-Scale Quantum Computing Simulations](#), by Scott N. Genin, Ilya G. Ryabinkin, Nathan R. Paisley, Sarah O. Whelan, Michael G. Helander, and Zachary M. Hudson, arXiv, November 2021 (32 pages).
- [3328] [Sparse Simulation of VQE Circuits](#), by Damian S. Steiger, Thomas Häner, Scott N. Genin, and Helmut G. Katzgraber, arXiv, February 2025 (10 pages).
- [3329] [A quantum annealing architecture with all-to-all connectivity from local interactions](#), by Wolfgang Lechner et al., October 2015 (5 pages).
- [3330] [Universal Parity Quantum Computing](#), by Michael Fellner, PRL, October 2022 (7 pages).
- [3331] [Applications of universal parity quantum computation](#), by Michael Fellner, PRA, October 2022 (8 pages).
- [3332] [Parity Quantum Optimization: Compiler](#), by Kilian Ender, Roeland ter Hoeven, Benjamin E. Niehoff, Maike Drieb-Schön, and Wolfgang Lechner, arXiv, March 2023 (9 pages).
- [3333] [Parity Quantum Optimization: Encoding Constraints](#), by Maike Drieb-Schön, Kilian Ender, Younes Javanmard, and Wolfgang Lechner, arXiv, March 2023 (24 pages).
- [3334] [Quantum Approximate Optimization with Parallelizable Gates](#), by Wolfgang Lechner, arXiv, March 2018 (5 pages).
- [3335] [Low-depth Circuit Implementation of Parity Constraints for Quantum Optimization](#), by Josua Unger, Anette Messinger, Benjamin E. Niehoff, Michael Fellner, and Wolfgang Lechner, arXiv, July 2024 (11 pages).
- [3336] [Parity Quantum Optimization: Benchmarks](#), by Michael Fellner, Kilian Ender, Roeland ter Hoeven, and Wolfgang Lechner, arXiv, March 2023 (12 pages).
- [3337] [Rapid counter-diabatic sweeps in lattice gauge adiabatic quantum computing](#), by Andreas Hartmann and Wolfgang Lechner, arXiv, September 2019 (11 pages).
- [3338] [A Josephson Parametric Oscillator-Based Ising Machine](#), by Sasan Razmkhah, Mehdi Kamal, Nobuyuki Yoshikawa, and Massoud Pedram, arXiv, December 2023 (9 pages).
- [3339] [From NISQ to Fault-Tolerance: Applications and Algorithmic Benchmarks for Spin Qubits](#), by Frederik Lohof, Florian Ginzel, and Wolfgang Lechner, arXiv, September 2026 (19 pages).
- [3340] [Error correction for encoded quantum annealing](#), by Fernando Pastawski and John Preskill, arXiv, April 2016 (4 pages).
- [3341] [Rydberg blockade based parity quantum optimization](#), by Martin Lanthaler, Clemens Daska, Kilian Ender, and Wolfgang Lechner, arXiv, October 2022 (10 pages).
- [3342] [Quantum optimization with globally driven neutral atom arrays](#), by Martin Lanthaler, Kilian Ender, Clemens Daska, and Wolfgang Lechner, arXiv, October 2024 (11 pages).
- [3343] [Applications of Universal Parity Quantum Computation](#), by Michael Fellner, Anette Messinger, Kilian Ender, and Wolfgang Lechner, arXiv, November 2022 (9 pages).
- [3344] [Fault-tolerant quantum computing with the parity code and noise-biased qubits](#), by Anette Messinger, Valentin Torggler, Berend Klaver, Michael Fellner, and Wolfgang Lechner, arXiv, April 2024 (8 pages).
- [3345] [Replacement-Type Quantum Gates](#), by Florian Ginzel, Javad Kazemi, Valentin Torggler, and Wolfgang Lechner, arXiv, August 2025 (20 pages).
- [3346] [Optimizing Atom Transport, Gate-Count and Depth with Parity Twine](#), by Javad Kazemi, Michael Fellner, Riccardo J. Valencia-Tortora, Michael Schuler, and Wolfgang Lechner, arXiv, September 2026 (19 pages).
- [3347] [Fault-tolerant multi-qubit gates in Parity Codes](#), by Anette Messinger, Christophe Goeller, and Wolfgang Lechner, arXiv, December 2025 (6 pages).
- [3348] [Strategies for solving the Fermi-Hubbard model on near-term quantum computers](#), by Chris Cade, Lana Mineh, Ashley Montanaro, and Stasja Stanisic, arXiv, November 2020 (11 pages).
- [3349] [Efficient classical simulation of time dynamics in Fermi-Hubbard models with imaginary interactions](#), by Raul A. Santos, arXiv, January 2026 (6 pages).
- [3350] [Probing ground state properties of the kagome antiferromagnetic Heisenberg model using the Variational Quantum Eigensolver](#), by Jan Lukas Bosse and Ashley Montanaro, arXiv, October 2021 (1 pages).
- [3351] [Towards near-term quantum simulation of materials](#), by Laura Clinton, Toby Cubitt, Brian Flynn, Filippo Maria Gambetta, Joel Klassen, Ashley Montanaro, Stephen Piddock, Raul A. Santos, and Evan Sheridan, arXiv, November 2022 (94 pages).
- [3352] [Quantum Phase Estimation without Controlled Unitaries](#), by Laura Clinton, Toby S. Cubitt, Raul Garcia-Patron, Ashley Montanaro, Stasja Stanisic, and Maarten Stroeks, arXiv, October 2024 (35 pages).
- [3353] [On the stability to noise of fermion-to-qubit mappings](#), by Guillermo González-García, Filippo Maria Gambetta, and Raul A. Santos, arXiv, March 2026 (31 pages).
- [3354] [Quantum-Aided Drug Design: Find Your Star Molecule Webinar](#), by PolarisQB, YouTube, May 2024 (40 mn video).
- [3355] [A Comparison of Small Molecule Generation Methods in Structure-Based Drug Design: Artificial Intelligence vs Quantum Computing | ChemRxiv](#), by Kendall Byler and Shahar Keinan, ChemRxiv, November 2025 (17 pages).
- [3356] [Low-rank quantum state preparation](#), by Israel F. Araujo, Carsten Blank, Ismael C. S. Araújo, and Adenilton J. da Silva, arXiv, July 2023 (10 pages).
- [3357] [Truncation technique for variational quantum eigensolver for Molecular Hamiltonians](#), by Qidong Xu and Kanav Setia, arXiv, February 2024 (10 pages).
- [3358] [Learning to rank quantum circuits for hardware-optimized performance enhancement](#), by Gavin S. Hartnett, Aaron Barbosa, Pranav S. Mundada, Michael Hush, Michael J. Biercuk, and Yuval Baum, arXiv, November 2024 (18 pages).

- [3359] [Boosting quantum computer hardware performance with TensorFlow](#), by Michael J. Biercuk et al., October 2020.
- [3360] [Experimental benchmarking of an automated deterministic error suppression workflow for quantum algorithms](#), by Pranav S. Mundada, Michael J. Biercuk, Yuval Baum, et al., arXiv, May 2023 (20 pages).
- [3361] [Quantum phase estimation with Fire Opal](#), by Rowen Wu, May 2023.
- [3362] [Optimize hybrid quantum-classical algorithms directly with Fire Opal](#), by Rowen Wu, April 2023.
- [3363] [Heterogeneous architectures enable a 138× reduction in physical qubit requirements for fault-tolerant quantum computing under detailed accounting](#), by Pranav S. Mundada, Aleksei Khindanov, Yulun Wang, Claire L. Edmunds, Paul Coote, Michael J. Biercuk, Yuval Baum, and Michael Hush, arXiv, April 2026 (34 pages).
- [3364] [Q-CTRL Delivers 3,000x Speedup in Materials Discovery for the Energy Sector with Quantum Computing, Demonstrates Evidence of Practical Quantum Advantage](#), by Q-CTRL team, Q-CTRL website, May 2026.
- [3365] [Simulating Q-CTRL's practical quantum advantage experiment on a laptop with Algorithmiq's Majorana Propagation](#), by Guillermo García-Pérez, LinkedIn, May 2026.
- [3366] [Simulating Q-CTRL's experiment with MPO-based classical simulation](#), by Netanel Lindner, LinkedIn, May 2026.
- [3367] [Pushing the Classical Frontier of 1D Fermi-Hubbard Quench Dynamics Beyond Current Quantum Simulations](#), by Roman Rausch, Sukhbinder Singh, Saeed S. Jahromi, Augustine Kshetrimayum, and Roman Orus, arXiv, June 2026 (30 pages).
- [3368] [Q-CTRL Achieves Computational Gains in Generation of Long-Range Entanglement Enhanced by Error Detection](#), by Matt Swayne, The Quantum Insider, May 2025.
- [3369] [Achieving computational gains with quantum error-correction primitives: Generation of long-range entanglement enhanced by error detection](#), by Haoran Liao, Michael J. Biercuk, Yuval Baum, et al., arXiv, May 2025 (8 pages).
- [3370] [Resource-efficient context-aware dynamical decoupling embedding for arbitrary large-scale quantum algorithms](#), by Paul Coote, Roman Dimov, Smarak Maity, Gavin S. Hartnett, Michael J. Biercuk, and Yuval Baum, arXiv, September 2024 (12 pages).
- [3371] [Unlocking peak performance on IBM Quantum Nighthawk devices with crosstalk suppression](#), by Paul Coote, Q-CTRL website, February 2026.
- [3372] [No need to calibrate: characterization and compilation for high-fidelity circuit execution using imperfect gates](#), by Ashish Kakkar, Samuel Marsh, Yulun Wang, Pranav Mundada, Paul Coote, Gavin Hartnett, Michael J. Biercuk, and Yuval Baum, arXiv, November 2025 (16 pages).
- [3373] [Towards Chemically Accurate and Scalable Quantum Simulations on IQM Quantum Hardware: A Quantum-HPC Hybrid Approach](#), by Anurag K. S. V., Ashish Kumar Patra, Manas Mukherjee, Alok Shukla, Sai Shankar P., Ruchika Bhat, Radhika T. S. L., and Jaiganesh G., arXiv, April 2026 (86 pages).
- [3374] [Quantum Simulation of Ligand-like Molecules through Sample-based Quantum Diagonalization in Density Matrix Embedding Framework](#), by Ashish Kumar Patra, Anurag K. S. V., Sai Shankar P., Ruchika Bhat, Raghavendra V., Rahul Maitra, and Jaiganesh G., arXiv, November 2025 (29 pages).
- [3375] [Bridging the NISQ and Fault-Tolerant Regimes: Generative-ML-Assisted Quantum Selected CI for Molecular Simulations](#), by Anurag K. S. V., Ashish Kumar Patra, Manas Mukherjee, Ruchika Bhat, Sai Shankar P., Rahul Maitra, and Jaiganesh G., arXiv, June 2026.
- [3376] [Introduction to Quantum Error Correction with Stabilizer Codes](#), by Zachary P. Bradshaw, Jeffrey J. Dale, and Ethan N. Evans, arXiv, July 2025 (88 pages).
- [3377] [Estimation of electrostatic interaction energies on a trapped-ion quantum computer](#), by Pauline J. Ollitrault, Michael Streif, et al., <https://pubs.acs.org/doi/10.1021/acscentsci.4c00058>, March 2024 (8 pages).
- [3378] [Quantum Low-Density Parity-Check Codes](#), by Bane Vasic, Valentin Savin, Nithin Raveendran, et al., arXiv, October 2025 (40 pages).
- [3379] [Constant-overhead fault-tolerant quantum computation with reconfigurable atom arrays](#), by Qian Xu, J. Pablo Bonilla Ataides, Nithin Raveendran, Bane Vasic, Mikhail D. Lukin, Liang Jiang, Hengyun Zhou, et al., Nature Physics, April 2024 (preprint on arXiv).
- [3380] [Trapping Sets of Quantum LDPC Codes](#), by Nithin Raveendran and Bane Vasić, arXiv, October 2021 (19 pages).
- [3381] [Soft Syndrome Decoding of Quantum LDPC Codes for Joint Correction of Data and Syndrome Errors](#), by Nithin Raveendran, Narayanan Rengaswamy, Asit Kumar Pradhan, and Bane Vasić, arXiv, May 2022 (7 pages).
- [3382] [On the Importance of Error Mitigation for Quantum Computation](#), by Dorit Aharonov, Maor Shutman, et al., arXiv, March 2025 (28 pages).
- [3383] [Reliable high-accuracy error mitigation for utility-scale quantum circuits](#), by Dorit Aharonov, Netanel H. Lindner, et al., arXiv, August 2025 (12 pages).
- [3384] [Syndrome aware mitigation of logical errors](#), by Dorit Aharonov, Yosi Atia, Eyal Bairey, Zvika Brakerski, Itsik Cohen, Omri Golan, Ilya Gurwich, Netanel H. Lindner, and Maor Shutman, arXiv, December 2025 (25 pages).
- [3385] [Resolving Structure in Prethermal Floquet Dynamics with Precision Quantum Computation](#), by Eyal Leviatan, Netanel H. Lindner, et al., arXiv, July 2026 (50 pages).
- [3386] [Ladder Operator Block-Encoding](#), by William A. Simon, Carter M. Gustin, Kamil Serafin, Alexis Ralli, Gary R. Goldstein, and Peter J. Love, arXiv, December 2025 (40 pages).
- [3387] [Extending Quantum Computing through Subspace, Embedding and Classical Molecular Dynamics Techniques](#), by Thomas M. Bickley, Angus Mingare, Tim Weaving, Michael Williams de la Bastida, Shunzhou Wan, Martina Nibbi, Philipp Seitz, Alexis Ralli, Peter J. Love, Minh Chung, Mario Hernández Vera, Laura Schulz, and Peter V. Coveney, arXiv, November 2025 (18 pages).
- [3388] [Bridging Quantum Chemistry and MaxCut: Classical Performance Guarantees and Quantum Algorithms for the Hartree-Fock Method](#), by Alexis Ralli, Tim Weaving, Peter V. Coveney, and Peter J. Love, arXiv, June 2025 (36 pages).
- [3389] [Noncontextual Pauli Hamiltonians](#), by Alexis Ralli, Tim Weaving, and Peter J. Love, arXiv, June 2025 (18 pages).
- [3390] [Towards Compact Wavefunctions from Quantum-Selected Configuration Interaction](#), by Tim Weaving, Angus Mingare, Alexis Ralli, and Peter V. Coveney, arXiv, November 2025 (16 pages).
- [3391] [QMill announces a six-fold leap in reaching quantum advantage](#), by QMill, QMill website, January 2026.
- [3392] [Shallow-circuit Supervised Learning on a Quantum Processor](#), by Luca Candelori, Swarnadeep Majumder, Antonio Mezzacapo, Javier Robledo Moreno, Kharen Musaelian, Santhanam Nagarajan, Sunil Pinnamaneni, Kunal Sharma, and Dario Villani, arXiv, January 2026.
- [3393] [QAOA in Quantum Datacenters: Parallelization, Simulation, and Orchestration](#), by Amana Liaqat, Ahmed Darwish, Adrian Roman, and Stephen DiAdamo, arXiv, March 2025 (11 pages).
- [3394] [Qoro Quantum and CESGA Demonstrate Distributed Quantum Circuit Simulations on HPC Infrastructure](#), by Cierra Choucair, The Quantum Insider, May 2025.
- [3395] [Maestro: Intelligent Execution for Quantum Circuit Simulation](#), by Oriol Bertomeu, Hamzah Ghayas, Adrian Roman, and Stephen DiAdamo, arXiv, December 2025 (11 pages).

- [3396] [Comparative Benchmarking of Utility-Scale Quantum Emulators](#), by Anna Leonteva, Guido Masella, Maxime Outteryck, Asier Piñeiro Orioli, and Shannon Whitlock, arXiv, April 2025 (28 pages).
- [3397] [AtomTwin.jl: a physics-native digital twin framework for neutral-atom quantum processors](#), by Shannon Whitlock, arXiv, April 2026 (32 pages).
- [3398] [Addressable gate-based logical computation with quantum LDPC codes](#), by Laura Pecorari, Francesco Paolo Guerri, Hugo Perrin, and Guido Pupillo, arXiv, November 2025 (12 pages).
- [3399] [Pulse-efficient quantum machine learning](#), by André Melo, Nathan Earnest-Noble, and Francesco Tacchino, arXiv, November 2022 (8 pages).
- [3400] [Benchmarking Optimization Algorithms for Automated Calibration of Quantum Devices](#), by Kevin Pack, Shai Machnes, and Frank K. Wilhelm, arXiv, September 2025 (12 pages).
- [3401] [Is quantum computing green? An estimate for an energy-efficiency quantum advantage](#), by Daniel Jaschke and Simone Montangero, Quantum Science and Technology, January 2023.
- [3402] [Quantum computation of stopping power for inertial fusion target design](#), by Nicholas C. Rubin, Dominic W. Berry, Alina Kononov, Fionn D. Malone, Tanuj Khattar, Alec White, Joonho Lee, Hartmut Neven, Ryan Babbush, and Andrew D. Baczewski, arXiv, August 2023 (37 pages).
- [3403] [Surface code scaling on heavy-hex superconducting quantum processors](#), by Arian Vezvaei, Cesar Benito, Mario Morford-Oberst, Alejandro Bermudez, and Daniel A. Lidar, Nature Communications, July 2026 (11 pages).
- [3404] [Visualizing quantum mechanics in an interactive simulation – Virtual Lab by Quantum Flytrap](#), by Piotr Migdal, Klementyna Jankiewicz, Pawel Grabarz, Chiara Decaroli, and Philippe Cochin, arXiv, May 2022 (29 pages).
- [3405] [Hybrid Annealing Method Based on subQUBO Model Extraction With Multiple Solution Instances](#), by Yuta Atobe et al., IEEE, 2022 (14 pages).
- [3406] [Quantum Hamiltonian Evolution for Coherent Quantum Learning](#), by Ignacio B. Acedo, Javier Gonzalez-Conde, Pablo Rodriguez-Grasa, Barry C. Sanders, and Lirandë Pira, arXiv, September 2026 (18 pages).
- [3407] [Hybrid Quantum Solvers in Production: how to succeed in the NISQ era?](#), by Eneko Osaba, Esther Villar-Rodriguez, Aitor Gomez-Tejedor, and Izaskun Oregi, arXiv, September 2024 (12 pages).
- [3408] [Sparse Quantum State Preparation for Strongly Correlated Systems](#), by CÃ©sar Feniou et al., March 2024 (27 pages)  (preprint on [HAL](#)).
- [3409] [Major breakthrough in quantum algorithms: Qubit Pharma and Sorbonne University drastically reduce the number of qubits needed to simulate molecules](#), by Qubit Pharmaceuticals, July 2024.
- [3410] [Greedy gradient-free adaptive variational quantum algorithms on a noisy intermediate scale quantum computer](#), by CÃ©sar Feniou, Jean-Philip Piquemal, et al., Nature Scientific Reports, May 2025 (18 pages).
- [3411] [A Foundation Model for Accurate Atomistic Simulations in Drug Design](#), by Thomas PlÃ© et al., ChemRxiv, May 2025 (19 pages).
- [3412] [Extreme-scale Mathematically-based Computational Chemistry \(EMC2\)](#), by Cordis, 2020.
- [3413] [Qubit Pharmaceuticals closes a pre-seed round with Quantonation](#), by Quantonation, June 2020.
- [3414] [Computational Fluid Dynamics on Quantum Computers](#), by Madhava Syamlal, Carter Copen, Masashi Takahashi, Benjamin Hall QubitSolve Inc., and Inflection), arXiv, July 2024 (14 pages).
- [3415] [Hardness of classically sampling quantum chemistry circuits](#), by Ayoub Hafid, Masaya Kohda, et al., arXiv, April 2025 (11 pages).
- [3416] [First and second quantized digital quantum simulations of bosonic systems](#), by Mathias Mikkelsen and Hubert Okadome Valencia, arXiv, November 2025 (14 pages).
- [3417] [Quantum computing for accurate large-scale electronic-structure calculations: DFT-embedded, post-processed quantum-selected configuration interaction](#), by Tuan Minh Do, Yuichiro Yoshida, Tomoya Shiota, and Wataru Mizukami, arXiv, June 2026 (14 pages).
- [3418] [Symmetry-Adapted State Preparation for Quantum Chemistry on Fault-Tolerant Quantum Computers](#), by Viktor Khinevich and Wataru Mizukami, arXiv, January 2026 (19 pages).
- [3419] [Fast-forwardability of Jordan-Wigner-transformed Fermion models based on Cartan decomposition](#), by Yuichiro Hidaka, Hirotaka Oshima, et al., arXiv, February 2025 (12 pages).
- [3420] [Explicit block encodings of rate matrices for simulating polymerization kinetics on quantum computers](#), by Yuhei Ikeda, Hokuto Iwakiri, Soichiro Nishio, and Kentaro Matsumoto, arXiv, September 2026 (30 pages).
- [3421] [Contrasting Statistical Phase Estimation with the Variational Quantum Eigensolver in the era of Early Fault Tolerant Quantum Computation](#), by Ming-Zhi Chung, Ryosuke Imai, et al., arXiv, September 2024 (30 pages).
- [3422] [SMT-AD: a scalable quantum-inspired anomaly detection approach](#), by Apimuk Sornsang, Si Min Chan, Wenxuan Zhang, Swee Liang Wong, Joshua Lim, and Dario Poletti, arXiv, April 2026 (11 pages).
- [3423] [Quantum-Inspired Algorithms beyond Unitary Circuits: the Laplace Transform](#), by Noufal Jaseem, Sergi Ramos-Calderer, Gauthameshwar S., Dingzu Wang, JosÃ© Ignacio Latorre, and Dario Poletti, arXiv, March 2026 (9 pages).
- [3424] [A real-time, scalable, fast and resource-efficient decoder for a quantum computer](#), by Ben Barber, Abbas B. Ziad, et al., Nature Electronics, January 2025  (preprint on [arXiv](#)).
- [3425] [Local Clustering Decoder: a fast and adaptive hardware decoder for the surface code](#), by Abbas B. Ziad, Mark L. Turner, et al., arXiv, November 2024 (17 pages).
- [3426] [Demonstrating real-time and low-latency quantum error correction with superconducting qubits](#), by Laura Caune, Alexander D. Hill, et al., arXiv, October 2024 (11 pages).
- [3427] [Local clustering decoder as a fast and adaptive hardware decoder for the surface code](#), by Abbas B. Ziad, Ankit Zawaladiya, Canberk Topal, Joan Camps, GyÃ¶rgy P. Gehr, Matthew P. Stafford, and Mark L. Turner, Nature Communications, December 2025 (15 pages).
- [3428] [Reducing the error rate of a superconducting logical qubit using analog readout information](#), by Hany Ali, Jorge Marques, Ophelia Crawford, Joonas Majaniemi, Marc Serra-Peralta, David Byfield, Boris Varbanov, Barbara M. Terhal, Leonardo DiCarlo, and Earl T. Campbell, arXiv, August 2024 (23 pages).
- [3429] [Scalable Decoding Protocols for Fast Transversal Logic in the Surface Code](#), by Mark L. Turner et al., PRX Quantum, January 2026 (17 pages).
- [3430] [Riverlane Quantum Error Correction Report 2024](#), by Riverlane, October 2024 (67 pages).
- [3431] [Riverlane Quantum Error Correction Report 2025](#), by Riverlane, November 2025.
- [3432] [Riverlane’s QEC technology roadmap to utility scale quantum computing](#), by Riverlane, Riverlane, March 2026 (19 pages).
- [3433] [IQM, Riverlane, and Zurich Instruments join forces to launch world’s first QEC platform](#), by Riverlane, Riverlane News, February 2025.

- [3434] [Reducing quantum error correction overhead using soft information](#), by Joonas Majaniemi and Elisha S. Matekole, arXiv, April 2025 (16 pages).
- [3435] [Quantum Error Correction Interface \(QECi\) Specification](#), by Riverlane, March 2025.
- [3436] [Quantum algorithms to simulate quadratic classical Hamiltonians and optimal control](#), by Hari Krovi, arXiv, April 2024 (52 pages).
- [3437] [Quantum Computation of Electronic Structure with Projector Augmented-Wave Method and Plane Wave Basis Set](#), by Aleksei V. Ivanov, Earl Campbell, Róbert Izsák, et al., arXiv, June 2025 (55 pages).
- [3438] [Riverlane and Astex Pharmaceuticals join forces with Rigetti Computing to drive drug discovery forward](#), by Amy Flower, July 2021.
- [3439] [SavantX: Logistics Optimization at the Port of Los Angeles](#), by D-Wave and SavantX, D-Wave, 2021 (2 pages).
- [3440] [Accelerating Extended Benders Decomposition with Quantum-Classical Hybrid Solver](#), by Takuma Yoshihara and Masayuki Ohzeki, arXiv, October 2025 (4 pages).
- [3441] [Sigma-i and D-Wave Announce Largest-Ever Quantum Cloud-Access Contract |D-Wave Systems](#), by D-Wave, July 2019.
- [3442] [Optimization of Connection Patterns between Mobile Phones and Base Stations using Quantum Annealing](#), by Taisei Takabayashi, Masayuki Ohzeki, et al., arXiv, December 2024 (10 pages).
- [3443] [Quantum-classical hybrid algorithm using quantum annealing for multi-objective job shop scheduling](#), by Kenta Sawamura, Masayuki Ohzeki, et al., arXiv, November 2025 (15 pages).
- [3444] [Beyond the “Black Box”: Guarding Medical AI with 160-Qubit Distributed Quantum Monte Carlo](#), by Lucas Son, Medium, March 2026.
- [3445] [staq - A full-stack quantum processing toolkit](#), by Matthew Amy and Vlad Gheorghiu, arXiv, August 2020 (21 pages).
- [3446] [Quantum++: A modern C++ quantum computing library](#), by Vlad Gheorghiu, arXiv, December 2018 (21 pages).
- [3447] [Polynomial-time T-depth Optimization of Clifford+T circuits via Matroid Partitioning](#), by Matthew Amy, Dmitri Maslov, and Michele Mosca, arXiv, December 2013 (25 pages).
- [3448] [Terra Quantum secures EUR10m to build the European Quantum Ecosystem](#), by James Dargan, April 2020.
- [3449] [Optimization of chemical mixers design via tensor trains and quantum computing](#), by Nikita Belokonev, Artem Melnikov, Maninadh Podapaka, Karan Pinto, Markus Pflictsch, and Michael Perelshtein, arXiv, April 2023 (8 pages).
- [3450] [Practical application-specific advantage through hybrid quantum computing](#), by Michael Perelshtein, Asel Saginalieva, Karan Pinto, Vishal Shete, Alexey Pakhomchik, Artem Melnikov, Florian Neukart, Georg Gesek, Alexey Melnikov, and Valerii Vinokur, arXiv, May 2022 (13 pages).
- [3451] [A Uniform Quantum Computing Model Based on Virtual Quantum Processors](#), by George Gesek, arXiv, February 2023 (9 pages).
- [3452] [Soft-Quantum Algorithms](#), by Basil Kyriacou, Mo Kordzanganeh, Maniraman Periyasamy, and Alexey Melnikov, arXiv, April 2026 (6 pages).
- [3453] [Terra Quantum and Evonik Successfully Complete Geometry Optimization Project](#), by Terra Quantum, Terra Quantum news, February 2025.
- [3454] [Terra Quantum and BBVA Complete Pilot for Advanced Exotic Derivatives Pricing Solution](#), by Terra Quantum, Terra Quantum news, February 2025.
- [3455] [Predictive control of blast furnace temperature in steelmaking with hybrid depth-infused quantum neural networks](#), by Nayoung Lee, Alexey Melnikov, et al., arXiv, April 2025 (13 pages).
- [3456] [Tensor Train Optimization for Conformational Sampling of Organic Molecules](#), by Christopher Zurek et al., September 2024 (33 pages).
- [3457] [Terra Quantum Launches TQ42 Cryptography Library for Secure Data Transmission, Storage and Authentication](#), by Terra Quantum, December 2023.
- [3458] [Terra Quantum Wins Competitive SBIR Funding to Study Feasibility of Long-Range Quantum-Resistant Network for the US Air Force](#), by Terra Quantum, August 2024.
- [3459] [Lifting connectivity bottlenecks in superconducting quantum processors via enriched native two-qubit gates](#), by Hanyi Wang, Jingzhe Guo, Lijun Sun, Zhaohui Yang, Weizhi Tao, Xingye Yuan, Qiankun Wang, Bihao Guo, Chunwang Liu, Rui Yang, Yang Li, Yu Fan, Jiasheng Hu, Junhe Wang, Shuyue Zheng, Shengbin Wang, Xinfang Zhang, Feng Wu, Hantao Sun, and Jianxin Chen, arXiv, August 2026 (36 pages).
- [3460] [Bunny Codes: Broadening Superconducting Quantum Error Correction Capability through Advanced Control Engineering](#), by Runshi Zhou, Xingye Yuan, Linghang Kong, Fang Zhang, Kai Zhang, Zhaohui Yang, and Jianxin Chen, arXiv, August 2026 (20 pages).
- [3461] [Atomistic structure learning](#), by Mathias S. Jørgensen, Henrik L. Mortensen, Søren A. Meldgaard, Esben L. Kolsbjerg, Thomas L. Jacobsen, Knud H. Sørensen, and Bjørk Hammer, arXiv, February 2019 (19 pages).
- [3462] [2021 Annual Report](#), by Unitary Fund, Unitary Fund, 2021 (23 pages).
- [3463] [Mitiq: A software package for error mitigation on noisy quantum computers](#), by Ryan LaRose, Nathan Shammah, William J. Zeng, et al., arXiv, August 2022 (33 pages).
- [3464] [Two Layers, No Swaps: Biplanar SPOQC Architecture Improves Runtime of Fermi-Hubbard Simulation](#), by Boris Bourdoncle, Peter-Jan Derks, Théo Dessertaine, and Johannes Frank, arXiv, May 2026 (44 pages).
- [3465] [OpenVQA Community: Unlocking the Future of Variational Quantum Algorithms](#), by Mohammad Haidar, YouTube, 2025 (8 mn video).
- [3466] [Chemically-Accurate Prediction of the Ionisation Potential of Helium Using a Quantum Processor](#), by Manolo C. Per, Joshua W. Dai, et al., arXiv, April 2025 (11 pages).
- [3467] [Capgemini, IBM Launch Quantum Lab to Promote Quantum Use Cases](#), by Matt Swayne, The Quantum Insider, January 2022.
- [3468] [The Rise of the Chief Quantum Officer and The Prospect of Quantum at The Governance Level - an Interview with Shai Phillips, President of Psirch](#), by James Dargan, The Quantum Insider, October 2022.
- [3469] [Former Microsoft Quantum Team Unites to Form Emergence Quantum And Deliver Missing Infrastructure For Quantum Tech](#), by Matt Swayne, The Quantum Insider, May 2025.
- [3470] [A cryogenic CMOS chip for generating control signals for multiple qubits](#), by S. J. Pauka, K. Das, R. Kalra, A. Moini, Y. Yang, M. Trainer, A. Bousquet, C. Cantaloube, N. Dick, G. C. Gardner, M. J. Manfra, and D. J. Reilly, Nature Electronics, January 2021 (7 pages) 📄.
- [3471] [CMOS compatibility of semiconductor spin qubits](#), by Nard Dumoulin Stuyck et al., arXiv, September 2024 (36 pages).
- [3472] [Spin-qubit control with a milli-kelvin CMOS chip](#), by Samuel K. Bartee, Kohei M. Itoh, Arne Laucht, Andrew S. Dzurak, David J. Reilly, et al., Nature, June 2025 (12 pages).
- [3473] [Using a quantum computer to solve a real-world problem – what can be achieved today?](#), by R.Cumming and T.Thomas, arXiv, November 2022 (62 pages).
- [3474] [Partial oracles quantum algorithm framework – Part I: Analysis of in-place operations](#), by Fintan M. Bolton, arXiv, April 2026 (50 pages).

- [3475] [Quantum Algorithms: A Structural Classification](#), by Serge Salata, ResearchGate, June 2026 (18 pages).
- [3476] [Quantum algorithm for bioinformatics to compute the similarity between proteins](#), by Anthony Chagneau, Yousra Massaoudi, Imene Derbali, and Linda Yahiaoui, arXiv, February 2024 (33 pages).
- [3477] [On the robustness of Quantum Phase Estimation to compute ground properties of many-electron systems](#), by Wasil Sennane and Jérémie Messud, arXiv, January 2026 (30 pages).
- [3478] [Pricing multi-asset derivatives by variational quantum algorithms](#), by Kenji Kubo, Koichi Miyamoto, Kosuke Mitarai, and Keisuke Fujii, arXiv, July 2022 (18 pages).
- [3479] [Performance of Quantum Networks Using Heterogeneous Link Architectures](#), by Kento Samuel Soon, Naphan Benchasattabuse, Michal Hajdušek, Kentaro Teramoto, Shota Nagayama, and Rodney Van Meter, arXiv, May 2024 (10 pages).
- [3480] [Quantum State Readout via Overlap-Based Feature Extraction](#), by Hirofumi Nishi, Taichi Kosugi, Xinchu Huang, Satoshi Hirose, Tatsuya Okayama, and Yu-ichiro Matsushita, arXiv, May 2025 (14 pages).
- [3481] [A probabilistic imaginary-time evolution quantum algorithm for advection-diffusion equation: Explicit gate-level implementation and comparisons to quantum linear system algorithms](#), by Xinchu Huang, Hirofumi Nishi, Taichi Kosugi, Yoshifumi Kawada, and Yu-ichiro Matsushita, arXiv, March 2025 (69 pages).
- [3482] [TCS Turns to AI and Quantum Computing For Aerospace Revolution](#), by Matt Swayne, The Quantum Insider, January 2025.
- [3483] [End-to-End Portfolio Optimization with Quantum Annealing](#), by Sai Nandan Morapakula, Sangram Deshpande, Rakesh Yata, Rushikesh Ubale, Uday Wad, and Kazuki Ikeda, arXiv, April 2025 (9 pages).
- [3484] [Prediction of Stocks Index Price using Quantum GANs](#), by Sangram Deshpande, Gopal Ramesh Dahale, Sai Nandan Morapakula, and Uday Wad, arXiv, September 2025 (10 pages).
- [3485] [Hardness-Dependent Adiabatic Schedules for Analog Quantum Computing](#), by Sébastien Perseguers, arXiv, October 2024 (15 pages).

This **Software and service vendors** subsection contains 282 bibliographical references and notes containing at least 4,603 pages.

3.12 Quantum computing business applications key takeaways

- Most quantum computing market forecasts are highly optimistic and assume an early advent of scalable quantum computers. They also sometimes tweak forecasts by pushing business value numbers instead of an actual market for quantum technologies. These numbers are confusing and sometimes taken as market sizes, which fools some investors.
- There are interesting potential use cases of quantum computing in nearly every vertical market, particularly in energy, chemistry, healthcare, transportation and then finance. Most of them are theoretical or have been evaluated at a very low scale given the capacity of existing quantum computers. Some may be useful with advanced noisy computers (NISQ) while most of them will require highly scalable fault-tolerant quantum computing systems (FTQC). Others may find their way on analog quantum simulators.
- This book contains a framework proposal to assess the interest and relevance of published use cases. It helps separate marketing claims from practical evidence. Under a strict criterion requiring an independently documented production deployment and a demonstrated advantage over the best practical classical baseline, no qualifying corporate quantum-computing application was identified by the book's September 2026 cutoff. Many documented NISQ use cases run on fewer than 30 physical qubits and can therefore often be emulated classically at lower cost and sometimes faster. Without credible resource estimates or benchmarking against strong classical baselines, such case studies are primarily useful for learning, workflow development and algorithm exploration. The book contains a list of items to watch for in documented case studies (retrospective) and use cases (prospective).
- In some cases, the potential use cases are in the overpromising twilight zone such as simulating very complex molecules, improve climate modeling and develop innovations to fixing climate change, curing cancers or optimizing large fleets of autonomous vehicles. All these are dubious long-term promises. Serious case studies contain resource estimations. The use cases bringing significant business value frequently require a large number of logical qubits, in the thousands, and usually, rather long computing times. There is still a lot of room for quantum-algorithms optimizations.
- One purveyor of case studies is D-Wave with its quantum annealer although it has not yet demonstrated a real quantum advantage. IBM is second there, having evangelized a broad number of customers and developers since 2016, particularly around the theme of “quantum utility” which has not yet reached the realm of industry relevant applications even though they switched gears on “quantum advantage” in August 2026 with three case studies that are still far from being industry related. IonQ and Quantinuum are also publicizing case studies, but they are usually below a quantum advantage level, using fewer than 25 qubits. Quantinuum is making progress with its 56 and 98 qubits quantum processors.
- Beyond computing time, a quantum advantage can also come from the system's energetic footprint and/or the precision of the outcome. Specific cases can claim to be in that situation with analog quantum simulations.
- There are already many software vendors in the quantum-computing space despite the limited availability of hardware demonstrating practical quantum advantage. Their activities include pilot projects, software frameworks, hybrid quantum-classical algorithms, and quantum-inspired algorithms running on classical hardware, and engineering or consulting services. This book also covers the growing IT and consulting-services ecosystem around quantum technologies.

Page intentionally left blank.

Page intentionally left blank.

Page intentionally left blank.

Page intentionally left blank.

[illegible]