

Quantum algorithm

In quantum computing, a **quantum algorithm** is an [algorithm](#) which runs on a realistic model of [quantum computation](#), the most commonly used model being the [quantum circuit](#) model of computation.^{[1][2]} A classical (or non-quantum) algorithm is a finite sequence of instructions, or a step-by-step procedure for solving a problem, where each step or instruction can be performed on a classical [computer](#). Similarly, a quantum algorithm is a step-by-step procedure, where each of the steps can be performed on a [quantum computer](#). Although all classical algorithms can also be performed on a quantum computer,^{[3]:126} the term quantum algorithm is usually used for those algorithms which seem inherently quantum, or use some essential feature of quantum computation such as [quantum superposition](#) or [quantum entanglement](#).

Problems which are [undecidable](#) using classical computers remain undecidable using quantum computers.^{[4]:127} What makes quantum algorithms interesting is that they might be able to solve some problems faster than classical algorithms because the quantum superposition and quantum entanglement that quantum algorithms exploit probably can't be efficiently simulated on classical computers (see [Quantum supremacy](#)).

The most well known algorithms are [Shor's algorithm](#) for factoring, and [Grover's algorithm](#) for searching an unstructured database or an unordered list. Shor's algorithms runs much (almost exponentially) faster than the best known classical algorithm for factoring, the [general number field sieve](#). Grover's algorithm runs quadratically faster than the best possible classical algorithm for the same task, a [linear search](#).

Contents

Overview

Algorithms based on the quantum Fourier transform

- Deutsch–Jozsa algorithm
- Bernstein–Vazirani algorithm
- Simon's algorithm
- Quantum phase estimation algorithm
- Shor's algorithm
- Hidden subgroup problem
- Boson sampling problem
- Estimating Gauss sums
- Fourier fishing and Fourier checking

Algorithms based on amplitude amplification

- Grover's algorithm
- Quantum counting

Algorithms based on quantum walks

- Element distinctness problem
- Triangle-finding problem
- Formula evaluation
- Group commutativity

BQP-complete problems

- Computing knot invariants
- Quantum simulation
- Solving a linear systems of equations

Hybrid quantum/classical algorithms

QAOA

Variational quantum eigensolver

See also

References

External links

Surveys

Overview

Quantum algorithms are usually described, in the commonly used circuit model of quantum computation, by a quantum circuit which acts on some input qubits and terminates with a measurement. A quantum circuit consists of simple quantum gates which act on at most a fixed number of qubits. Quantum algorithms may also be stated in other models of quantum computation, such as the Hamiltonian oracle model.^[5]

Quantum algorithms can be categorized by the main techniques used by the algorithm. Some commonly used techniques/ideas in quantum algorithms include phase kick-back, phase estimation, the quantum Fourier transform, quantum walks, amplitude amplification and topological quantum field theory. Quantum algorithms may also be grouped by the type of problem solved, for instance see the survey on quantum algorithms for algebraic problems.^[6]

Algorithms based on the quantum Fourier transform

The quantum Fourier transform is the quantum analogue of the discrete Fourier transform, and is used in several quantum algorithms. The Hadamard transform is also an example of a quantum Fourier transform over an n-dimensional vector space over the field \mathbb{F}_2 . The quantum Fourier transform can be efficiently implemented on a quantum computer using only a polynomial number of quantum gates.

Deutsch–Jozsa algorithm

The Deutsch–Jozsa algorithm solves a black-box problem which probably requires exponentially many queries to the black box for any deterministic classical computer, but can be done with exactly one query by a quantum computer. If we allow both bounded-error quantum and classical algorithms, then there is no speedup since a classical probabilistic algorithm can solve the problem with a constant number of queries with small probability of error. The algorithm determines whether a function f is either constant (0 on all inputs or 1 on all inputs) or balanced (returns 1 for half of the input domain and 0 for the other half).

Bernstein–Vazirani algorithm

The Bernstein–Vazirani algorithm is the first quantum algorithm that is exponentially more efficient than classical algorithms. It was designed to create an oracle separation between BQP and BPP.

Simon's algorithm

Simon's algorithm solves a black-box problem exponentially faster than any classical algorithm, including bounded-error probabilistic algorithms. This algorithm, which achieves an exponential speedup over all classical algorithms that we consider efficient, was the motivation for Shor's factoring algorithm.

Quantum phase estimation algorithm

The quantum phase estimation algorithm is used to determine the eigenphase of an eigenvector of a unitary gate given a quantum state proportional to the eigenvector and access to the gate. The algorithm is frequently used as a subroutine in other algorithms.

Shor's algorithm

Shor's algorithm solves the discrete logarithm problem and the integer factorization problem in polynomial time,^[7] whereas the best known classical algorithms take super-polynomial time. These problems are not known to be in P or NP-complete. It is also one of the few quantum algorithms that solves a non-black-box problem in polynomial time where the best known classical algorithms run in super-polynomial time.

Hidden subgroup problem

The abelian hidden subgroup problem is a generalization of many problems that can be solved by a quantum computer, such as Simon's problem, solving Pell's equation, testing the principal ideal of a ring R and factoring. There are efficient quantum algorithms known for the Abelian hidden subgroup problem.^[8] The more general hidden subgroup problem, where the group isn't necessarily abelian, is a generalization of the previously mentioned problems and graph isomorphism and certain lattice problems. Efficient quantum algorithms are known for certain non-abelian groups. However, no efficient algorithms are known for the symmetric group, which would give an efficient algorithm for graph isomorphism^[9] and the dihedral group, which would solve certain lattice problems.^[10]

Boson sampling problem

The Boson Sampling Problem in an experimental configuration assumes^[11] an input of bosons (ex. photons of light) of moderate number getting randomly scattered into a large number of output modes constrained by a defined unitarity. The problem is then to produce a fair sample of the probability distribution of the output which is dependent on the input arrangement of bosons and the Unitarity.^[12] Solving this problem with a classical computer algorithm requires computing the permanent of the unitary transform matrix, which may be either impossible or take a prohibitively long time. In 2014, it was proposed^[13] that existing technology and standard probabilistic methods of generating single photon states could be used as input into a suitable quantum computable linear optical network and that sampling of the output probability distribution would be demonstrably superior using quantum algorithms. In 2015, investigation predicted^[14] the sampling problem had similar complexity for inputs other than Fock state photons and identified a transition in computational complexity from classically simulatable to just as hard as the Boson Sampling Problem, dependent on the size of coherent amplitude inputs.

Estimating Gauss sums

A Gauss sum is a type of exponential sum. The best known classical algorithm for estimating these sums takes exponential time. Since the discrete logarithm problem reduces to Gauss sum estimation, an efficient classical algorithm for estimating Gauss sums would imply an efficient classical algorithm for computing discrete logarithms, which is considered unlikely. However, quantum computers can estimate Gauss sums to polynomial precision in polynomial time.^[15]

Fourier fishing and Fourier checking

We have an oracle consisting of n random Boolean functions mapping n -bit strings to a Boolean value. We are required to find n n -bit strings z_1, \dots, z_n such that for the Hadamard-Fourier transform, at least $3/4$ of the strings satisfy

$$|\tilde{f}(z_i)| \geq 1$$

and at least $1/4$ satisfies

$$|\tilde{f}(z_i)| \geq 2.$$

This can be done in Bounded-error Quantum Polynomial time (BQP).^[16]

Algorithms based on amplitude amplification

Amplitude amplification is a technique that allows the amplification of a chosen subspace of a quantum state. Applications of amplitude amplification usually lead to quadratic speedups over the corresponding classical algorithms. It can be considered to be a generalization of Grover's algorithm.

Grover's algorithm

Grover's algorithm searches an unstructured database (or an unordered list) with N entries, for a marked entry, using only $O(\sqrt{N})$ queries instead of the $O(N)$ queries required classically.^[17] Classically, $O(N)$ queries are required, even if we allow bounded-error probabilistic algorithms.

Theorists have considered a hypothetical generalization of a standard quantum computer that could access the histories of the hidden variables in Bohmian mechanics. (Such a computer is completely hypothetical and would *not* be a standard quantum computer, or even possible under the standard theory of quantum mechanics.) Such a hypothetical computer could implement a search of an N -item database at most in $O(\sqrt[3]{N})$ steps. This is slightly faster than the $O(\sqrt{N})$ steps taken by Grover's algorithm. Neither search method would either model of quantum computer to solve NP-complete problems in polynomial time.^[18]

Quantum counting

Quantum counting solves a generalization of the search problem. It solves the problem of counting the number of marked entries in an unordered list, instead of just detecting if one exists. Specifically, it counts the number of marked entries in an N -element list, with error ϵ making only $\Theta\left(\frac{1}{\epsilon}\sqrt{\frac{N}{k}}\right)$ queries, where k is the number of marked elements in the list.^{[19][20]} More precisely, the algorithm outputs an estimate k' for k , the number of marked entries, with the following accuracy: $|k - k'| \leq \epsilon k$.

Algorithms based on quantum walks

A quantum walk is the quantum analogue of a classical random walk, which can be described by a probability distribution over some states. A quantum walk can be described by a quantum superposition over states. Quantum walks are known to give exponential speedups for some black-box problems.^{[21][22]} They also provide polynomial speedups for many problems. A framework for the creation of quantum walk algorithms exists and is quite a versatile tool.^[23]

Element distinctness problem

The element distinctness problem is the problem of determining whether all the elements of a list are distinct. Classically, $\Omega(N)$ queries are required for a list of size N , since this problem is harder than the search problem which requires $\Omega(N)$ queries. However, it can be solved in $\Theta(N^{2/3})$ queries on a quantum computer. The optimal algorithm is by Andris Ambainis.^[24] Yaoyun Shi first proved a tight lower bound when the size of the range is sufficiently large.^[25] Ambainis^[26] and Kutin^[27] independently (and via different proofs) extended his work to obtain the lower bound for all functions.

Triangle-finding problem

The triangle-finding problem is the problem of determining whether a given graph contains a triangle (a clique of size 3). The best-known lower bound for quantum algorithms is $\Omega(N)$, but the best algorithm known requires $O(N^{1.297})$ queries,^[28] an improvement over the previous best $O(N^{1.3})$ queries.^{[23][29]}

Formula evaluation

A formula is a tree with a gate at each internal node and an input bit at each leaf node. The problem is to evaluate the formula, which is the output of the root node, given oracle access to the input.

A well studied formula is the balanced binary tree with only NAND gates.^[30] This type of formula requires $\Theta(N^c)$ queries using randomness,^[31] where $c = \log_2(1 + \sqrt{33})/4 \approx 0.754$. With a quantum algorithm however, it can be solved in $\Theta(N^{0.5})$ queries. No better quantum algorithm for this case was known until one was found for the unconventional Hamiltonian oracle model.^[5] The same result for the standard setting soon followed.^[32]

Fast quantum algorithms for more complicated formulas are also known.^[33]

Group commutativity

The problem is to determine if a black box group, given by k generators, is commutative. A black box group is a group with an oracle function, which must be used to perform the group operations (multiplication, inversion, and comparison with identity). We are interested in the query complexity, which is the number of oracle calls needed to solve the problem. The deterministic and randomized query complexities are $\Theta(k^2)$ and $\Theta(k)$ respectively.^[34] A quantum algorithm requires $\Omega(k^{2/3})$ queries but the best known algorithm uses $O(k^{2/3} \log k)$ queries.^[35]

BQP-complete problems

Computing knot invariants

Witten had shown that the Chern-Simons topological quantum field theory (TQFT) can be solved in terms of Jones polynomials. A quantum computer can simulate a TQFT, and thereby approximate the Jones polynomial,^[36] which as far as we know, is hard to compute classically in the worst-case scenario.

Quantum simulation

The idea that quantum computers might be more powerful than classical computers originated in Richard Feynman's observation that classical computers seem to require exponential time to simulate many-particle quantum systems.^[37] Since then, the idea that quantum computers can simulate quantum physical processes exponentially faster than classical computers has been greatly fleshed out and elaborated. Efficient (that is, polynomial-time) quantum algorithms have been developed for simulating both Bosonic and Fermionic systems^[38] and in particular, the simulation of chemical reactions beyond the capabilities of current classical supercomputers requires only a few hundred qubits.^[39] Quantum computers can also efficiently simulate topological quantum field theories.^[40] In addition to its intrinsic interest, this result has led to efficient quantum algorithms for estimating quantum topological invariants such as Jones^[41] and HOMFLY polynomials,^[42] and the Turaev-Viro invariant of three-dimensional manifolds.^[43]

Solving a linear systems of equations

In 2009 Aram Harrow, Avinatan Hassidim, and Seth Lloyd, formulated a quantum algorithm for solving linear systems. The algorithm estimates the result of a scalar measurement on the solution vector to a given linear system of equations.^[44]

Provided the linear system is a sparse and has a low condition number κ , and that the user is interested in the result of a scalar measurement on the solution vector, instead of the values of the solution vector itself, then the algorithm has a runtime of $O(\log(N)\kappa^2)$, where N is the number of variables in the linear system. This offers an exponential speedup over the fastest classical algorithm, which runs in $O(N\kappa)$ (or $O(N\sqrt{\kappa})$ for positive semidefinite matrices).

Hybrid quantum/classical algorithms

Hybrid Quantum/Classical Algorithms combine quantum state preparation and measurement with classical optimization.^[45] These algorithms generally aim to determine the ground state eigenvector and eigenvalue of a Hermitian Operator.

QAOA

The quantum approximate optimization algorithm is a toy model of quantum annealing which can be used to solve problems in graph theory.^[46] The algorithm makes use of classical optimization of quantum operations to maximize an objective function.

Variational quantum eigensolver

The VQE algorithm applies classical optimization to minimize the energy expectation of an ansatz state to find the ground state energy of a molecule.^[47] This can also be extended to find excited energies of molecules.^[48]

See also

- Quantum optimization algorithms
- Quantum sort
- Primality test

References

1. Nielsen, Michael A.; Chuang, Isaac L. (2000). *Quantum Computation and Quantum Information*. Cambridge University Press. ISBN 978-0-521-63503-5.
2. Mosca, M. (2008). "Quantum Algorithms". *arXiv:0808.0369* (<https://arxiv.org/abs/0808.0369>) [quant-ph (<https://arxiv.org/archive/quant-ph>)].
3. Lanzagorta, Marco; Uhlmann, Jeffrey K. (1 January 2009). *Quantum Computer Science* (<https://books.google.com/?id=-wkJluw0YRsC&pg=PA23&lpg=PA23&dq=quantum%2520computer%2520equivalent%2520classical%2520computer#v=onepage&q=quantum%2520computer%2520equivalent%2520classical%2520computer&f=false>). Morgan & Claypool Publishers. ISBN 9781598297324.
4. Nielsen, Michael A.; Chuang, Isaac L. (2010). *Quantum Computation and Quantum Information* (<https://books.google.com/books?id=-s4DEy7o-a0C&printsec=frontcover#v=onepage&q=undecidable%20OR%20undecidability&f=false>) (2nd ed.). Cambridge: Cambridge University Press. ISBN 978-1-107-00217-3.
5. Farhi, E.; Goldstone, J.; Gutmann, S. (2007). "A Quantum Algorithm for the Hamiltonian NAND Tree". *arXiv:quant-ph/0702144* (<https://arxiv.org/abs/quant-ph/0702144>).
6. Childs, Andrew M.; van Dam, W. (2010). "Quantum algorithms for algebraic problems". *Reviews of Modern Physics*. **82** (1): 1–52. *arXiv:0812.0380* (<https://arxiv.org/abs/0812.0380>). Bibcode:2010RvMP...82....1C (<https://ui.adsabs.harvard.edu/abs/2010RvMP...82....1C>). doi:10.1103/RevModPhys.82.1 (<https://doi.org/10.1103/RevModPhys.82.1>).
7. Shor, P. W. (1997). "Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer". *SIAM Journal on Scientific and Statistical Computing*. **26** (5): 1484–1509. *arXiv:quant-ph/9508027* (<https://arxiv.org/abs/quant-ph/9508027>). Bibcode:1995quant.ph..8027S (<https://ui.adsabs.harvard.edu/abs/1995quant.ph..8027S>). doi:10.1137/s0097539795293172 (<https://doi.org/10.1137/s0097539795293172>).

8. Boneh, D.; Lipton, R. J. (1995). "Quantum cryptanalysis of hidden linear functions". In Coppersmith, D. (ed.). *Proceedings of the 15th Annual International Cryptology Conference on Advances in Cryptology*. Springer-Verlag. pp. 424–437. ISBN 3-540-60221-6.
9. Moore, C.; Russell, A.; Schulman, L. J. (2005). "The Symmetric Group Defies Strong Fourier Sampling: Part I". [arXiv:quant-ph/0501056](https://arxiv.org/abs/quant-ph/0501056) (<https://arxiv.org/abs/quant-ph/0501056>).
10. Regev, O. (2003). "Quantum Computation and Lattice Problems". [arXiv:cs/0304005](https://arxiv.org/abs/cs/0304005) (<https://arxiv.org/abs/cs/0304005>).
11. Ralph, T.C. "Figure 1: The boson-sampling problem" (http://www.nature.com/nphoton/journal/v7/n7/fig_tab/nphoton.2013.175_F1.html). *Nature Photonics*. Nature. Retrieved 12 September 2014.
12. Lund, A.P.; Laing, A.; Rahimi-Keshari, S.; Rudolph, T.; O'Brien, J.L.; Ralph, T.C. (5 September 2014). "Boson Sampling from Gaussian States". *Phys. Rev. Lett.* **113** (10): 100502. [arXiv:1305.4346](https://arxiv.org/abs/1305.4346) (<https://arxiv.org/abs/1305.4346>). Bibcode:2014PhRvL.113j0502L (<https://ui.adsabs.harvard.edu/abs/2014PhRvL.113j0502L>). doi:10.1103/PhysRevLett.113.100502 (<https://doi.org/10.1103%2FPhysRevLett.113.100502>). PMID 25238340 (<https://www.ncbi.nlm.nih.gov/pubmed/25238340>).
13. "The quantum revolution is a step closer" (<http://phys.org/news/2014-09-quantum-revolution-closer.html>). *Phys.org*. Omicron Technology Limited. Retrieved 12 September 2014.
14. Seshadreesan, Kaushik P.; Olson, Jonathan P.; Motes, Keith R.; Rohde, Peter P.; Dowling, Jonathan P. (2015). "Boson sampling with displaced single-photon Fock states versus single-photon-added coherent states: The quantum-classical divide and computational-complexity transitions in linear optics". *Physical Review A*. **91** (2): 022334. [arXiv:1402.0531](https://arxiv.org/abs/1402.0531) (<https://arxiv.org/abs/1402.0531>). Bibcode:2015PhRvA..91b2334S (<https://ui.adsabs.harvard.edu/abs/2015PhRvA..91b2334S>). doi:10.1103/PhysRevA.91.022334 (<https://doi.org/10.1103%2FPhysRevA.91.022334>).
15. van Dam, W.; Seroussi, G. (2002). "Efficient Quantum Algorithms for Estimating Gauss Sums". [arXiv:quant-ph/0207131](https://arxiv.org/abs/quant-ph/0207131) (<https://arxiv.org/abs/quant-ph/0207131>).
16. Aaronson, S. (2009). "BQP and the Polynomial Hierarchy". [arXiv:0910.4698](https://arxiv.org/abs/0910.4698) (<https://arxiv.org/abs/0910.4698>) [[quant-ph](https://arxiv.org/archive/quant-ph) (<https://arxiv.org/archive/quant-ph>)].
17. Grover, Lov K. (1996). "A fast quantum mechanical algorithm for database search". [arXiv:quant-ph/9605043](https://arxiv.org/abs/quant-ph/9605043) (<https://arxiv.org/abs/quant-ph/9605043>).
18. Aaronson, Scott. "Quantum Computing and Hidden Variables" (<https://www.scottaaronson.com/papers/qchvpra.pdf>) (PDF).
19. Brassard, G.; Hoyer, P.; Tapp, A. (1998). *Quantum Counting*. Lecture Notes in Computer Science. **1443**. pp. 820–831. [arXiv:quant-ph/9805082](https://arxiv.org/abs/quant-ph/9805082) (<https://arxiv.org/abs/quant-ph/9805082>). doi:10.1007/BFb0055105 (<https://doi.org/10.1007%2FBFb0055105>). ISBN 978-3-540-64781-2.
20. Brassard, G.; Hoyer, P.; Mosca, M.; Tapp, A. (2002). "Quantum Amplitude Amplification and Estimation". In Samuel J. Lomonaco, Jr. (ed.). *Quantum Computation and Quantum Information*. AMS Contemporary Mathematics. **305**. pp. 53–74. [arXiv:quant-ph/0005055](https://arxiv.org/abs/quant-ph/0005055) (<https://arxiv.org/abs/quant-ph/0005055>). Bibcode:2000quant.ph..5055B (<https://ui.adsabs.harvard.edu/abs/2000quant.ph..5055B>).
21. Childs, A. M.; Cleve, R.; Deotto, E.; Farhi, E.; Gutmann, S.; Spielman, D. A. (2003). "Exponential algorithmic speedup by quantum walk". *Proceedings of the 35th Symposium on Theory of Computing*. Association for Computing Machinery. pp. 59–68. [arXiv:quant-ph/0209131](https://arxiv.org/abs/quant-ph/0209131) (<https://arxiv.org/abs/quant-ph/0209131>). doi:10.1145/780542.780552 (<https://doi.org/10.1145%2F780542.780552>). ISBN 1-58113-674-9.
22. Childs, A. M.; Schulman, L. J.; Vazirani, U. V. (2007). "Quantum Algorithms for Hidden Nonlinear Structures". *Proceedings of the 48th Annual IEEE Symposium on Foundations of Computer Science*. IEEE. pp. 395–404. [arXiv:0705.2784](https://arxiv.org/abs/0705.2784) (<https://arxiv.org/abs/0705.2784>). doi:10.1109/FOCS.2007.18 (<https://doi.org/10.1109%2FFOCS.2007.18>). ISBN 0-7695-3010-9.
23. Magniez, F.; Nayak, A.; Roland, J.; Santha, M. (2007). "Search via quantum walk". *Proceedings of the 39th Annual ACM Symposium on Theory of Computing*. Association for Computing Machinery. pp. 575–584. [arXiv:quant-ph/0608026](https://arxiv.org/abs/quant-ph/0608026) (<https://arxiv.org/abs/quant-ph/0608026>). doi:10.1145/1250790.1250874 (<https://doi.org/10.1145%2F1250790.1250874>). ISBN 978-1-59593-631-8.

24. Ambainis, A. (2007). "Quantum Walk Algorithm for Element Distinctness". *SIAM Journal on Computing*. **37** (1): 210–239. [arXiv:quant-ph/0311001](https://arxiv.org/abs/quant-ph/0311001) (<https://arxiv.org/abs/quant-ph/0311001>). doi:10.1137/S0097539705447311 (<https://doi.org/10.1137%2FS0097539705447311>).
25. Shi, Y. (2002). *Quantum lower bounds for the collision and the element distinctness problems*. Proceedings of the 43rd Symposium on Foundations of Computer Science. pp. 513–519. [arXiv:quant-ph/0112086](https://arxiv.org/abs/quant-ph/0112086) (<https://arxiv.org/abs/quant-ph/0112086>). doi:10.1109/SFCS.2002.1181975 (<https://doi.org/10.1109%2FSFCS.2002.1181975>).
26. Ambainis, A. (2005). "Polynomial Degree and Lower Bounds in Quantum Complexity: Collision and Element Distinctness with Small Range". *Theory of Computing*. **1** (1): 37–46. doi:10.4086/toc.2005.v001a003 (<https://doi.org/10.4086%2Ftoc.2005.v001a003>).
27. Kutin, S. (2005). "Quantum Lower Bound for the Collision Problem with Small Range". *Theory of Computing*. **1** (1): 29–36. doi:10.4086/toc.2005.v001a002 (<https://doi.org/10.4086%2Ftoc.2005.v001a002>).
28. Aleksandrs Belovs (2011). "Span Programs for Functions with Constant-Sized 1-certificates". [arXiv:1105.4024](https://arxiv.org/abs/1105.4024) (<https://arxiv.org/abs/1105.4024>) [quant-ph (<https://arxiv.org/archive/quant-ph>)].
29. Magniez, F.; Santha, M.; Szegedy, M. (2007). "Quantum Algorithms for the Triangle Problem". *SIAM Journal on Computing*. **37** (2): 413–424. [arXiv:quant-ph/0310134](https://arxiv.org/abs/quant-ph/0310134) (<https://arxiv.org/abs/quant-ph/0310134>). doi:10.1137/050643684 (<https://doi.org/10.1137%2F050643684>).
30. Aaronson, S. (3 February 2007). "NAND now for something completely different" (<http://scottaaronson.com/blog/?p=207>). *Shtetl-Optimized*. Retrieved 17 December 2009.
31. Saks, M.E.; Wigderson, A. (1986). "Probabilistic Boolean Decision Trees and the Complexity of Evaluating Game Trees" (<http://www.math.ias.edu/~avi/PUBLICATIONS/MYPAPERS/SW86/SW86.pdf>) (PDF). *Proceedings of the 27th Annual Symposium on Foundations of Computer Science*. IEEE. pp. 29–38. doi:10.1109/SFCS.1986.44 (<https://doi.org/10.1109%2FSFCS.1986.44>). ISBN 0-8186-0740-8.
32. Ambainis, A. (2007). "A nearly optimal discrete query quantum algorithm for evaluating NAND formulas". [arXiv:0704.3628](https://arxiv.org/abs/0704.3628) (<https://arxiv.org/abs/0704.3628>) [quant-ph (<https://arxiv.org/archive/quant-ph>)].
33. Reichardt, B. W.; Spalek, R. (2008). "Span-program-based quantum algorithm for evaluating formulas". *Proceedings of the 40th Annual ACM symposium on Theory of Computing*. Association for Computing Machinery. pp. 103–112. [arXiv:0710.2630](https://arxiv.org/abs/0710.2630) (<https://arxiv.org/abs/0710.2630>). doi:10.1145/1374376.1374394 (<https://doi.org/10.1145%2F1374376.1374394>). ISBN 978-1-60558-047-0.
34. Pak, Igor (2012). "Testing commutativity of a group and the power of randomization". *LMS Journal of Computation and Mathematics*. **15**: 38–43. doi:10.1112/S1461157012000046 (<https://doi.org/10.1112%2FS1461157012000046>).
35. Magniez, F.; Nayak, A. (2007). "Quantum Complexity of Testing Group Commutativity". *Algorithmica*. **48** (3): 221–232. [arXiv:quant-ph/0506265](https://arxiv.org/abs/quant-ph/0506265) (<https://arxiv.org/abs/quant-ph/0506265>). doi:10.1007/s00453-007-0057-8 (<https://doi.org/10.1007%2FS00453-007-0057-8>).
36. Aharonov, D.; Jones, V.; Landau, Z. (2006). "A polynomial quantum algorithm for approximating the Jones polynomial". *Proceedings of the 38th Annual ACM symposium on Theory of Computing*. Association for Computing Machinery. pp. 427–436. [arXiv:quant-ph/0511096](https://arxiv.org/abs/quant-ph/0511096) (<https://arxiv.org/abs/quant-ph/0511096>). doi:10.1145/1132516.1132579 (<https://doi.org/10.1145%2F1132516.1132579>).
37. Feynman, R. P. (1982). "Simulating physics with computers". *International Journal of Theoretical Physics*. **21** (6–7): 467–488. Bibcode:1982IJTP...21..467F (<https://ui.adsabs.harvard.edu/abs/1982IJTP...21..467F>). CiteSeerX 10.1.1.45.9310 (<https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.45.9310>). doi:10.1007/BF02650179 (<https://doi.org/10.1007%2FBF02650179>).
38. Abrams, D. S.; Lloyd, S. (1997). "Simulation of many-body Fermi systems on a universal quantum computer". *Physical Review Letters*. **79** (13): 2586–2589. [arXiv:quant-ph/9703054](https://arxiv.org/abs/quant-ph/9703054) (<https://arxiv.org/abs/quant-ph/9703054>). Bibcode:1997PhRvL..79.2586A (<https://ui.adsabs.harvard.edu/abs/1997PhRvL..79.2586A>). doi:10.1103/PhysRevLett.79.2586 (<https://doi.org/10.1103%2FPhysRevLett.79.2586>).

39. Kassal, I.; Jordan, S. P.; Love, P. J.; Mohseni, M.; Aspuru-Guzik, A. (2008). "Polynomial-time quantum algorithm for the simulation of chemical dynamics" (<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2596249>). *Proceedings of the National Academy of Sciences of the United States of America*. **105** (48): 18681–86. arXiv:0801.2986 (<https://arxiv.org/abs/0801.2986>). Bibcode:2008PNAS..10518681K (<https://ui.adsabs.harvard.edu/abs/2008PNAS..10518681K>). doi:10.1073/pnas.0808245105 (<https://doi.org/10.1073/pnas.0808245105>). PMC 2596249 (<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2596249>). PMID 19033207 (<https://www.ncbi.nlm.nih.gov/pubmed/19033207>).
40. Freedman, M.; Kitaev, A.; Wang, Z. (2002). "Simulation of Topological Field Theories by Quantum Computers". *Communications in Mathematical Physics*. **227** (3): 587–603. arXiv:quant-ph/0001071 (<https://arxiv.org/abs/quant-ph/0001071>). Bibcode:2002CMAPh.227..587F (<https://ui.adsabs.harvard.edu/abs/2002CMAPh.227..587F>). doi:10.1007/s002200200635 (<https://doi.org/10.1007/s002200200635>).
41. Aharonov, D.; Jones, V.; Landau, Z. (2009). "A polynomial quantum algorithm for approximating the Jones polynomial". *Algorithmica*. **55** (3): 395–421. arXiv:quant-ph/0511096 (<https://arxiv.org/abs/quant-ph/0511096>). doi:10.1007/s00453-008-9168-0 (<https://doi.org/10.1007/s00453-008-9168-0>).
42. Wocjan, P.; Yard, J. (2008). "The Jones polynomial: quantum algorithms and applications in quantum complexity theory". *Quantum Information and Computation*. **8** (1): 147–180. arXiv:quant-ph/0603069 (<https://arxiv.org/abs/quant-ph/0603069>). Bibcode:2006quant.ph..3069W (<https://ui.adsabs.harvard.edu/abs/2006quant.ph..3069W>).
43. Alagic, G.; Jordan, S.P.; König, R.; Reichardt, B. W. (2010). "Approximating Turaev-Viro 3-manifold invariants is universal for quantum computation". *Physical Review A*. **82** (4): 040302. arXiv:1003.0923 (<https://arxiv.org/abs/1003.0923>). Bibcode:2010PhRvA..82d0302A (<https://ui.adsabs.harvard.edu/abs/2010PhRvA..82d0302A>). doi:10.1103/PhysRevA.82.040302 (<https://doi.org/10.1103/PhysRevA.82.040302>).
44. Harrow, Aram W; Hassidim, Avinatan; Lloyd, Seth (2008). "Quantum algorithm for solving linear systems of equations". *Physical Review Letters*. **103** (15): 150502. arXiv:0811.3171 (<https://arxiv.org/abs/0811.3171>). Bibcode:2009PhRvL.103o0502H (<https://ui.adsabs.harvard.edu/abs/2009PhRvL.103o0502H>). doi:10.1103/PhysRevLett.103.150502 (<https://doi.org/10.1103/PhysRevLett.103.150502>). PMID 19905613 (<https://www.ncbi.nlm.nih.gov/pubmed/19905613>).
45. Moll, Nikolaj; Barkoutsos, Panagiotis; Bishop, Lev S.; Chow, Jerry M.; Cross, Andrew; Egger, Daniel J.; Filipp, Stefan; Fuhrer, Andreas; Gambetta, Jay M.; Ganzhorn, Marc; Kandala, Abhinav; Mezzacapo, Antonio; Müller, Peter; Riess, Walter; Salis, Gian; Smolin, John; Tavernelli, Ivano; Temme, Kristan (2018). "Quantum optimization using variational algorithms on near-term quantum devices". *Quantum Science and Technology*. **3** (3): 030503. arXiv:1710.01022 (<https://arxiv.org/abs/1710.01022>). doi:10.1088/2058-9565/aab822 (<https://doi.org/10.1088/2058-9565/aab822>).
46. Farhi, Edward; Goldstone, Jeffrey; Gutmann, Sam (14 November 2014). "A Quantum Approximate Optimization Algorithm". arXiv:1411.4028 (<https://arxiv.org/abs/1411.4028>) [quant-ph (<https://arxiv.org/archive/quant-ph>)].
47. Peruzzo, Alberto; McClean, Jarrod; Shadbolt, Peter; Yung, Man-Hong; Zhou, Xiao-Qi; Love, Peter J.; Aspuru-Guzik, Alán; O'Brien, Jeremy L. (23 July 2014). "A variational eigenvalue solver on a photonic quantum processor" (<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4124861>). *Nature Communications*. **5** (1): 4213. arXiv:1304.3061 (<https://arxiv.org/abs/1304.3061>). Bibcode:2014NatCo...5E4213P (<https://ui.adsabs.harvard.edu/abs/2014NatCo...5E4213P>). doi:10.1038/ncomms5213 (<https://doi.org/10.1038/ncomms5213>). ISSN 2041-1723 (<https://www.worldcat.org/issn/2041-1723>). PMC 4124861 (<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4124861>). PMID 25055053 (<https://www.ncbi.nlm.nih.gov/pubmed/25055053>).
48. Higgott, Oscar; Wang, Daochen; Brierley, Stephen (21 May 2018). "Variational Quantum Computation of Excited States". arXiv:1805.08138 (<https://arxiv.org/abs/1805.08138>) [quant-ph (<https://arxiv.org/archive/quant-ph>)].

External links

- The Quantum Algorithm Zoo (<http://math.nist.gov/quantum/zoo/>): A comprehensive list of quantum algorithms that provide a speedup over the fastest known classical algorithms.
- Andrew Childs' lecture notes on quantum algorithms (<https://www.cs.umd.edu/~amchilds/qa/>)
- The Quantum search algorithm - brute force (<https://bastion.center/the-quantum-search-algorithm/>).

Surveys

- Smith, J.; Mosca, M. (2012). "Algorithms for Quantum Computers". *Handbook of Natural Computing*. p. 1451. doi:[10.1007/978-3-540-92910-9_43](https://doi.org/10.1007/978-3-540-92910-9_43) (https://doi.org/10.1007%2F978-3-540-92910-9_43). ISBN 978-3-540-92909-3.
- Childs, A. M.; Van Dam, W. (2010). "Quantum algorithms for algebraic problems". *Reviews of Modern Physics*. **82**: 1–52. arXiv:[0812.0380](https://arxiv.org/abs/0812.0380) (<https://arxiv.org/abs/0812.0380>). Bibcode:2010RvMP..82....1C (<https://ui.adsabs.harvard.edu/abs/2010RvMP..82....1C>). doi:[10.1103/RevModPhys.82.1](https://doi.org/10.1103/RevModPhys.82.1) (<https://doi.org/10.1103%2FRevModPhys.82.1>).

Retrieved from "https://en.wikipedia.org/w/index.php?title=Quantum_algorithm&oldid=914350362"

This page was last edited on 6 September 2019, at 21:26 (UTC).

Text is available under the [Creative Commons Attribution-ShareAlike License](#); additional terms may apply. By using this site, you agree to the [Terms of Use](#) and [Privacy Policy](#). Wikipedia® is a registered trademark of the [Wikimedia Foundation, Inc.](#), a non-profit organization.