International Journal for Multidisciplinary Research (IJFMR)

E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

A Comparative Analysis of Classical and Quantum Algorithms through the Environmental Impact Lens

Nemi Botadara

Raigad Military School, New Link Road, Oshiwara, Jogeshwari (W), Mumbai – 400102

Abstract

This paper evaluates eight computational algorithms across sorting, searching, and factoring tasks to analyze how their operation counts impact environmental sustainability. Classical algorithms (Bubble Sort, Insertion Sort, Merge Sort, Timsort, Quicksort, Linear Search) are compared against quantum algorithms (Grover's Search, Shor's Factoring) using algorithmic time complexity as a proxy for processing energy consumption. The analysis establishes a distinct efficiency hierarchy determined by problem scale and mathematical scaling behavior. For quadratic speedups like Grover's Search, a high problem-size threshold exists before quantum algorithms become more energy-efficient than classical alternatives. For exponential speedups like Shor's Algorithm, the quantum framework crosses this threshold almost immediately due to the impossibly large operation counts required by classical alternatives at cryptographic scales. These findings demonstrate that achieving sustainable quantum computing requires targeted deployment at exponential problem domains.

Keywords: Quantum Computing, Grover's Algorithm, Shor's Algorithm, Classical Algorithms, Environmental Sustainability, Energy Efficiency, Algorithmic Complexity, NISQ

1. Introduction

1.1 Quantum Computing: Foundations and Emergence

Quantum computing represents a paradigm shift in how information is processed, moving beyond the binary logic of classical computers to exploit the principles of quantum mechanics — superposition, entanglement, and interference. A classical computer encodes information in bits, each holding a value of 0 or 1. A quantum computer uses quantum bits, or qubits, which can exist in a superposition of both 0 and 1 simultaneously. This capability enables a quantum processor to explore an exponentially large solution space in parallel, offering computational advantages that are physically impossible for classical machines on certain classes of problems.

The conceptual origins of quantum computing trace back to 1981, when physicist Richard Feynman proposed that simulating quantum systems required a computer that itself operates on quantum principles [1]. David Deutsch formalized the notion of a universal quantum computer in 1985 [2], and the field gained substantial momentum in 1994 when Peter Shor published a polynomial-time quantum algorithm for factoring large integers [3]. Lov Grover followed in 1996 with a quantum algorithm capable of searching an unsorted database of N items in $O(\sqrt{N})$ operations, compared to the $O(N)$ required classically [4].

Hardware progress, demonstrated by IBM's 1,121-qubit Condor processor (2023) [5] and Google's Sycamore [6] quantum supremacy demonstration, has made experimentation tangible. Commercial quantum-as-a-service offerings from IBM (IBM Quantum), Google (Quantum AI), Amazon (Braket), and Microsoft (Azure Quantum) have democratized access, enabling researchers to test algorithms on real quantum hardware via cloud interfaces.

Algorithmic research in quantum computing has advanced rapidly across several fronts. Variational Quantum Eigensolver (VQE) and Quantum Approximate Optimization Algorithm (QAOA) represent near-term hybrid algorithms designed for noisy intermediate-scale quantum (NISQ) devices. These algorithms interleave quantum and classical computation, making them runnable on today's imperfect hardware.

Grover's search algorithm remains the canonical benchmark for quantum database search [4, 7]. In its standard formulation, the algorithm uses a two-step repeating cycle. First, it uses a selection process (called an oracle) where it simultaneously looks at all possible answers and puts a marker on the correct one. Then, it applies an amplitude amplification step (known as a diffusion operator) where it mathematically shrinks down the importance of all the unmarked (incorrect) answers and boosts the importance of the marked ones. By repetition, incorrect answers are faded out and the correct answer becomes prominent. Extensions of Grover's framework, including amplitude amplification, quantum walk-based search, and generalized multi-target Grover, have broadened applicability [8].

The most consequential current and near-future applications span cryptography, drug discovery and molecular simulation, financial portfolio optimization, logistics and route optimization, and machine learning. In the cryptographic domain, Shor's algorithm presents a critical challenge to global data infrastructure by providing an exponential quantum speedup capable of breaking standard classical encryption. To protect communications, physical hardware applications like Quantum Key Distribution (which uses quantum physics for encryption and data transmission, making it impossible for hackers to go undetected) are currently being deployed in limited networks in China, Switzerland, and the US.

1.2 Literature Review

Boyer, Brassard, Høyer, and Tapp (1998) [7] established the theoretical optimality of Grover's algorithm, proving via adversarial lower-bound arguments that no quantum algorithm can solve the unstructured search problem in fewer than $O(\sqrt{N})$ oracle queries. Shor's original 1994 paper and its 1997 journal formalization [3] demonstrated that the quantum Fourier transform enables polynomial-time factoring. Viamontes, Markov, and Hayes (2005) [9] used a classical computer to simulate a quantum computer and literally counted every single basic quantum operation (gate-level counts) needed to run Grover's algorithm and compared it to classical searching. Bernstein and Vazirani's oracle complexity work (1997) [10] demonstrated mathematically that for certain problems, a quantum computer could solve things exponentially faster than a classical one by asking a smart black-box question (an oracle query).

Babbush, McClean, Newman, et al. (2021) [11], however, presented a critical counterpoint: they argued that for a "quadratic" speedup, the energy and time spent fixing errors might completely swallow up the quantum advantage in real life. More recently, Jaschke and Montangero (2023) [12] estimated conditions under which quantum algorithms achieve a measurable energy-efficiency advantage over classical supercomputers, finding the advantage holds above a problem-size threshold — providing direct quantitative support for the environmental framing of this paper. AbuGhanem and Eleuch (2024) [13] implemented and characterized Grover's search on IBM's 127-qubit Eagle processors, benchmarking success probabilities under realistic noise, the most current empirical evidence of Grover's hardware performance.

1.3 Environmental Angle: Computation and Energy

The computational efficiency of an algorithm has a direct, if often overlooked, environmental dimension. Data centres consumed an estimated 200–250 TWh of electricity globally in 2022, representing roughly 1% of worldwide electricity demand [14]. Each arithmetic or logical operation executed on a processor dissipates a finite amount of energy through transistor switching. Algorithms requiring fewer operations cause fewer transistor transitions per unit computation, directly reducing dynamic power dissipation for a fixed workload. Moreover, classical encryption systems require massive computational resources and energy expenditure to process large cryptographic keys. Quantum frameworks can transition this intractable, energy-heavy task into an efficient, polynomial-time quantum operation. The World Economic Forum (2026) [15] has noted that quantum algorithms handle information in a fundamentally reversible manner, allowing intermediate states to be uncomputed rather than erased, in principle enabling exponential reductions in minimum energy use for computationally hard problems. Auffèves (2022) [16] argued that even with error correction overhead, quantum computers retain an energy advantage for problems above a certain scale threshold, providing comparative energy consumption data across problem sizes.

1.4 Research Question

This paper investigates the environmental and computational implications of quantum algorithmic speedups by evaluating processing consumption, measured as the total number of operations required to reach a solution as a function of input size N. Specifically, the evaluation of classical linear search against Grover's quantum search is discussed along with the evaluation of classical General Number Field Sieve (GNFS) against Shor's algorithm. To contextualize these systems within a broader computational framework, a secondary evaluation benchmark is established against standard classical sorting paradigms (Bubble Sort, Insertion Sort, Merge Sort, Timsort, and Quicksort) to map a comprehensive operational efficiency hierarchy at scale.

2. Grover's Algorithm

Grover's algorithm, published by Lov K. Grover in 1996 [4], is a quantum algorithm that searches an unsorted database of N items for a target element with a query complexity of $O(\sqrt{N})$, achieving a quadratic speedup over the classical lower bound of O(N) for unstructured search. It is provably optimal [7] — no quantum algorithm can solve the unstructured search problem in asymptotically fewer queries — and represents one of the most broadly applicable quantum speedups known.

2.1 Formal Setup

Let $f : \{0, 1, ..., N-1\} \rightarrow \{0, 1\}$ be an oracle function such that $f(x) = 1$ if and only if $x = x^*$, the target element. The search space has $N = 2^n$ elements indexed by n-bit strings. The algorithm operates on an n-qubit register. The goal is to identify x^* with high probability using as few evaluations of 'f' as possible.

2.2 Algorithmic Steps

The algorithm proceeds in three phases:

Initialization: Apply the n-qubit Hadamard transform $H^{\otimes n}$ to the initial state $|0\rangle^{\otimes n}$, producing a uniform superposition over all N basis states: $|\psi\rangle = (1/\sqrt{N}) \Sigma_x |x\rangle$. Every candidate solution is represented simultaneously.

Grover Iteration (Oracle + Diffusion): Repeat the composite operator $G = (2|\psi\rangle\langle\psi| - I) \cdot U_f$ approximately $\lfloor(\pi/4)\sqrt{N}\rfloor$ times. The oracle (Phase Kickback) $U_f|x\rangle = (-1)^{f(x)}|x\rangle$ flips the sign of the amplitude of the target state without revealing it. The Diffusion Operator $(2|\psi\rangle\langle\psi| - I)$ performs an inversion about the mean amplitude, amplifying the amplitude of the marked state. After k iterations, the amplitude of $|x^*\rangle$ grows to $\sin((2k+1)\theta)$, where $\sin(\theta) = 1/\sqrt{N}$.

Measurement: The quantum register is measured in the computational basis. With probability approaching $1 - O(1/N)$, the measurement yields x^*, the target element.

2.3 Complexity Analysis

The total number of oracle calls is $T = O(\sqrt{N})$. For classical brute-force search, the expected number of queries is $N/2 = O(N)$. The speedup ratio is $O(N)/O(\sqrt{N}) = O(\sqrt{N})$, meaning the quantum advantage grows with the square root of the problem size. At $N = 10^6$, this is a speedup of 1,000×; at $N = 10^9$, the speedup reaches approximately 31,623× [7].

Multi-target extensions generalize the algorithm: if there are k solutions among N items, the required iterations fall to $O(\sqrt{N/k})$, yielding further speedup when multiple targets exist [8]. Recent work by AbuGhanem and Eleuch (2024) [13] has empirically validated these complexity predictions on IBM's 127-qubit hardware, observing success probabilities consistent with theoretical expectations across all eight single-result oracles.

3. Impact of Quantum Computing on the Environment

The environmental impact of computing is increasingly understood through the lens of energy consumption per unit of useful computation. This paper scopes its environmental analysis to algorithm-level processing consumption: the number of logical operations required to compute a solution, which is the primary determinant of dynamic processing power usage under fixed hardware parameters.

The relationship between operation count and processing energy is grounded in the physics of computation. Each arithmetic or logical operation executed on a processor dissipates a finite amount of energy through transistor switching (dynamic power: $P_d = \alpha CVf$, where α is activity factor, C is capacitance, V is supply voltage, and f is clock frequency) [17]. Algorithms requiring fewer operations cause fewer transistor transitions per unit computation, directly reducing dynamic power dissipation for a fixed workload.

Data centres globally consumed an estimated 200–250 TWh in 2022 [14]. For a search operation over $N = 10^6$ items, a classical linear scan requires 10^6 comparisons on average, while Grover's quantum search requires approximately 785 oracle applications ($\lfloor(\pi/4)\times10^3\rfloor$). Each oracle call translates to $O(\log N) = O(20)$ elementary gate operations, yielding a total gate count of approximately 15,700 elementary operations — compared to 10^6 classical comparisons. The ratio is approximately 63.7:1 in favour of Grover's [9].

The World Economic Forum (2026) [15] has noted that quantum algorithms' reversibility fundamentally changes the energy scaling: where classical algorithms must erase intermediate bits (dissipating energy irreversibly), quantum algorithms can "uncompute" intermediate states, allowing minimum energy use to grow more slowly than classical counterparts for hard problems. Pasqal's NISQ devices have demonstrated ~18 kW total energy consumption — orders of magnitude below classical supercomputers consuming 10–20 MW for equivalent problem classes [18].

Critically, Cornell University researchers demonstrated that a quantum-classical hybrid framework for AI data centres could reduce energy consumption by up to 12.5% and carbon emissions by up to 9.8%

compared to all-classical approaches [19] — a near-term, practical demonstration of quantum energy advantage even before full fault-tolerant hardware is available.

4. Analysis: Algorithm Comparison

4.1 Complexity as a Proxy for Processing Consumption

Under the premise that algorithmic time complexity determines the number of elementary processor operations required for a task, and that each operation has a constant associated energy cost on a given hardware platform, time complexity functions as a reliable proxy for per-solution processing consumption [9]. Table 1 and 2 analyze sorting and searching algorithms together to observe how different algorithmic classes scale from a high-level processing perspective.

Table 1: Algorithmic Complexity Comparison

Algorithm	Best Case	Average Case	Worst Case	Space	Type
Bubble Sort	$O(n)$	$O(n^2)$	$O(n^2)$	$O(1)$	Classical
Insertion Sort	$O(n)$	$O(n^2)$	$O(n^2)$	$O(1)$	Classical
Merge Sort	$O(n \log n)$	$O(n \log n)$	$O(n \log n)$	$O(n)$	Classical
Timsort	$O(n)$	$O(n \log n)$	$O(n \log n)$	$O(n)$	Classical
Quicksort	$O(n \log n)$	$O(n \log n)$	$O(n^2)$	$O(\log n)$	Classical
Classical Linear Search	$O(1)$	$O(N)$	$O(N)$	$O(1)$	Classical
Grover's Search	$O(\sqrt{N})$	$O(\sqrt{N})$	$O(\sqrt{N})$	$O(n)$	Quantum
Shor's Factoring	$O((\log N)^2(\log \log N))$	$O((\log N)^3)$	$O((\log N)^3)$	$O(n^2)$	Quantum

Note: While sorting and searching represent fundamentally different computational tasks, they are grouped here on a unified baseline to establish a high-level operational efficiency hierarchy. Classical linear search has been added alongside standard classical sorting paradigms to provide a direct classical benchmark against Grover's quantum search and Shor's quantum factoring.

For the average and worst-case scenarios, classical linear search scales linearly at $O(N)$ showing that its workload is proportional to database size. Whereas, Grover's Search breaks past this classical barrier by scaling at $O(\sqrt{N})$ and Shor's algorithm provides a more dramatic superpolynomial shift by dropping integer factorization down to the third power of log N. By establishing these theoretical bounds in Table 1, the physical processing operations required as input sizes grow can be accurately modelled.

Table 2: Processing Consumption Per Solution (N = 1,000)

Algorithm	Time Complexity	Ops (~N=1,000)	Relative Power Use	Efficiency Note	Rating
Bubble Sort	$O(n^2)$	$\sim 10^6$	Very High	Baseline — worst	Poor

International Journal for Multidisciplinary Research (IJFMR)

E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

Algorithm	Time Complexity	Ops (~N=1,000)	Relative Power Use	Efficiency Note	Rating
Insertion Sort	$O(n^2)$	$\sim 10^6$	Very High	≈ Bubble Sort	Poor
Merge Sort	O(n log n)	~9,966	Medium	2–3 times better	Moderate
Timsort	O(n log n)	~9,966	Low–Med	Best for real data	Good
Quicksort	O(n log n)	~9,966	Low	Cache-efficient	Good
Classical Linear Search	O(N)	~1,000	Medium	Baseline Classical Search	Moderate
Grover's Search	$O(\sqrt{N})$	~25	Minimal	Quadratic speedup	Best
Shor's Factoring	$O((\log N)^3)$	Poly vs Exp	Very Low	Exp. advantage	Best

Relative processing consumption (operation count) per solution at a milestone of N = 1,000 items.

From Table 2, it is clear that Grover's Search requires mere ~25 oracle operations to isolate the exact same target item as compared to classical linear search. This represents a massive reduction in processing overhead and provides the first quantitative hint of the immense energy-saving potential of quantum frameworks at larger scales.

4.2 Grover's Speedup at Varying N

The quadratic nature of Grover's speedup means its advantage compounds with dataset size. Table 3 quantifies this across six representative dataset sizes:

Table 3: Grover's Speedup at Varying N

N (Items)	Classical Linear Search Ops O(N)	Grover Ops ($\sqrt{N}$)	Speedup Factor	Implication
100	100	10	10×	~90 ops saved
1,000	1,000	~32	~31.6×	~968 ops saved
10,000	10,000	~100	100×	~9,900 ops saved
1,000,000	10^6	~1,000	1,000×	999,000 ops saved
10^9	10^9	~31,623	~31,623×	Massive energy saving
10^{12}	10^{12}	$\sim 10^6$	$\sim 10^6\times$	Practically intractable classically

Grover's algorithm operation count vs. classical linear search at varying dataset sizes. Speedup grows as $\sqrt{N}$; at $N = 10^{12}$, quantum search is a million times more operation-efficient.

As demonstrated in Table 3, the operational gap between classical and quantum search widens dramatically as the problem scale reaches big-data thresholds.

4.3 Shor's Algorithm vs. Classical Factoring

While the previous sections focus on unstructured database searching, this section examines integer factorization which is the mathematical process of breaking a large composite number down into its original prime components. This is the foundation for global security systems like RSA encryption, because classical computers find it incredibly difficult and energy-expensive to factor massive numbers. By evaluating this, the contrast between how a classical system scales against Shor's quantum algorithm becomes clear. Shor's algorithm achieves an exponential speedup over the best classical algorithm (General Number Field Sieve, GNFS) [3, 20]. Notably, in 2023, NYU computer scientist Oded Regev published a multidimensional variant of Shor's algorithm that reduces the gate complexity to $O((\log N)^2(\log\log N))$ improving on Shor's original $O((\log N)^3)$ [21]. This demonstrates that even canonical quantum algorithms remain subject to ongoing refinement. The operation count differential at practical RSA key sizes is shown in Table 4.

Table 4: Shor's Algorithm vs. Classical GNFS (Key Size Comparison)

Key Size (bits)	Classical Ops (GNFS)	Shor's Ops	Speedup	Practical Implication
512	$\sim 10^{23}$	$\sim 10^{6}$	$\sim 10^{17}\times$	Fully classically tractable; quantum trivial
1024	$\sim 10^{36}$	$\sim 10^{9}$	$\sim 10^{27}\times$	Classical borderline; quantum easily solvable
2048	$\sim 10^{63}$	$\sim 10^{10}$	$\sim 10^{53}\times$	Classical infeasible; RSA-2048 standard today
4096	$\sim 10^{100}+$	$\sim 10^{11}$	$\sim 10^{89}\times$	Extreme classical burden; quantum polynomial

Operation count comparison between Shor's algorithm and classical General Number Field Sieve for RSA key sizes. The exponential gap renders classical approaches computationally infeasible at cryptographic scales. Classical GNFS ops computed via exp(1.9 × (n × ln 2)^(1/3) × (ln(n × ln 2))^(2/3)) for an n-bit key [22, 23]. Shor's ops estimated as n^3 under $O((\log N)^3)$ complexity [3].

4.4 Hardware Landscape

The physical realization of Grover's and Shor's algorithms depends on the underlying quantum hardware architecture. Translating the logical speedups into real-world energy savings requires stabilizing physical qubits under distinct engineering constraints. Table 5 maps the dominant quantum computing platforms currently in development by key players in the quantum industry. While qubit counts are increasing rapidly, fault-tolerant operation remains the primary near-term milestone.

Table 5: Current Quantum Hardware Landscape

System	Technology	Qubits	Era / Status	Relevance to This Paper
IBM Eagle / Heron	Superconducting	127–133	NISQ (2022–24)	Leading commercial NISQ platform; Eagle used for Grover benchmarking (AbuGhanem 2024)

System	Technology	Qubits	Era / Status	Relevance to This Paper
Google Sycamore	Superconducting	53–70	Supremacy demo (2019)	Claimed quantum supremacy on random circuit sampling; 200 s vs 10,000 years classical [6]
D-Wave Advantage	Quantum Annealing	5,000+	Optimization (commercial)	Deployed in logistics/supply-chain; not universal gate-model; suited for QUBO problems
IonQ Aria	Trapped Ion	25 (logical)	NISQ (2023)	Higher gate fidelity than superconducting; lower qubit count; suitable for Grover at small N
Quantinuum H2	Trapped Ion	56	NISQ (2024)	Record gate fidelity (>99.9%); demonstrated fully fault-tolerant logical qubit operations
IBM Condor (2023)	Superconducting	1,121	NISQ/near fault-tolerant	First 1,000+ qubit processor; marks transition toward fault-tolerant quantum computation

Representative quantum hardware platforms as of 2023–2024.

5. Results and Discussion

The mathematical speedups established in Section 4 do not automatically translate into instantaneous environmental savings. A holistic comparative analysis requires balancing theoretical operational reductions against the real-world physical power footprints of the hardware architectures. Unlike classical data center servers, where power consumption scales dynamically with CPU/GPU utilization, dominant quantum platforms demand a massive power baseline. Consequently, for small problem sizes (N), a classical server running an O(N log N) Quicksort or a multi-threaded Linear Search will consume vastly less total energy (measured in Joules) than a quantum system, because the classical operation finishes in milliseconds on a machine drawing only a few hundred watts.

For unstructured search tasks, Grover's algorithm introduces a quadratic speedup ($O(\sqrt{N})$) over the linear classical workload ($O(N)$). This advantage is intrinsic to the algorithm and does not depend on specific hardware parameters [4, 7]. As database sizes scale into the petabyte and exabyte ranges, the classical energy profile escalates dramatically due to the sheer volume of transistor switching cycles across thousands of distributed server nodes. The environmental implication is that deploying Grover's algorithm for large-scale unstructured search, once fault-tolerant quantum hardware is accessible, would reduce the processing component of energy consumption by multiple orders of magnitude. However, quantum environmental advantage is only realized at scales where the classical processing workload becomes large enough to surpass the static power requirements of quantum infrastructure. This problem-size threshold must be established because quantum searching only yields net carbon savings at extreme, enterprise-scale data operations.

The environmental equation alters completely when analyzing the integer factorization case. Because Shor's algorithm introduces an exponential speedup over the classical GNFS framework, the classical energy requirement scales exponentially as cryptographic keys grow from 1024 to 4096 bits. As demonstrated by the operations workload in Table 4, computing 10^{63} classical operations to breach a 2048-bit key demands an energy expenditure that would exhaust modern planetary power grids. In this domain, the sheer exponential reduction in logical operations allows the quantum framework to bypass the problem-size threshold almost immediately, making the high electricity cost of cooling the quantum computer completely worth it compared to the massive power a classical data center would burn.

The critical caveat, raised by Babbush et al. (2021) [11] and acknowledged throughout this analysis, is that quantum algorithms require quantum hardware. Current NISQ-era quantum devices are too noisy and insufficiently error-corrected to execute Grover's algorithm faithfully at large N. However, as Jaschke and Montangero (2023) [12] demonstrated, a quantum energy advantage is already measurable above a problem-size threshold even with current NISQ hardware, and this threshold will fall as hardware matures. Therefore, true quantum environmental sustainability is a strict function of problem scale and algorithmic scaling behavior. Quadratic speedups require massive datasets to justify their physical infrastructure costs, whereas exponential speedups achieve environmental viability at highly accessible operational thresholds.

6. Conclusion

This paper evaluated eight computational algorithms across sorting, searching, and factoring tasks to analyze how their operation counts impact environmental sustainability. The analysis confirms a distinct efficiency hierarchy determined entirely by problem scale and mathematical scaling behavior. For quadratic speedups like Grover's Search, a high problem-size threshold exists; classical sorting and searching remain more energy-efficient for small datasets, meaning quantum searching only yields net carbon savings at massive, enterprise scales. Conversely, for exponential speedups like Shor's Algorithm, the quantum framework crosses this threshold almost immediately because the massive operations required by classical alternatives quickly become impossible to sustain. Fewer operations directly translate into reduced processor cycles and lower dynamic processing energy per solution. Ultimately, achieving sustainable quantum computing requires a targeted deployment strategy that caters to exponential problem domains rather than small-scale data.

References

1. Feynman R.P., "Simulating physics with computers," International Journal of Theoretical Physics, 1982, 21 (6–7), 467–488.
2. Deutsch D., "Quantum theory, the Church–Turing principle and the universal quantum computer," Proceedings of the Royal Society of London A, 1985, 400 (1818), 97–117.
3. Shor P.W., "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer," SIAM Journal on Computing, 1997, 26 (5), 1484–1509. (Original: 1994 FOCS proceedings.)
4. Grover L.K., "A fast quantum mechanical algorithm for database search," Proceedings of the 28th Annual ACM Symposium on Theory of Computing (STOC), 1996, 212–219.
5. IBM Research, "IBM Condor: 1,121-qubit quantum processor," IBM Quantum Blog, 2023. https://research.ibm.com/blog/ibm-quantum-roadmap-2025
6. Arute F., et al. (Google AI), "Quantum supremacy using a programmable superconducting processor," Nature, 2019, 574 (7779), 505–510.
7. Boyer M., Brassard G., Høyer P., Tapp A., "Tight bounds on quantum searching," Fortschritte der Physik, 1998, 46 (4–5), 493–505.

8. Brassard G., Høyer P., Mosca M., Tapp A., "Quantum amplitude amplification and estimation," AMS Contemporary Mathematics Series, 2002, 305, 53–74.
9. Viamontes G.F., Markov I.L., Hayes J.P., "Is quantum search practical?" Computing in Science & Engineering, 2005, 7 (3), 62–70.
10. Bernstein E., Vazirani U., "Quantum complexity theory," SIAM Journal on Computing, 1997, 26 (5), 1411–1473.
11. Babbush R., McClean J.R., Newman M., et al., "Focus beyond quadratic speedups for error-corrected quantum advantage," PRX Quantum, 2021, 2 (1), 010103.
12. Jaschke D., Montangero S., "Is quantum computing green? An estimate for an energy-efficiency quantum advantage," Quantum Science and Technology, 2023, 8 (2), 025001.
13. AbuGhanem M., Eleuch H., "Comprehensive characterization of three-qubit Grover search algorithm on IBM's 127-qubit superconducting quantum computers," Scientific Reports, 2024, 14. https://doi.org/10.1038/s41598-024-80188-6
14. IEA, "Data Centres and Data Transmission Networks," International Energy Agency, 2022. https://www.iea.org/reports/data-centres-and-data-transmission-networks
15. World Economic Forum, "Scaling quantum computing for maximum energy efficiency," WEF Insight Report, 2026. https://www.weforum.org/stories/2026/01/how-can-we-scale-quantum-computing-in-the-most-energy-efficient-way/
16. Auffèves A., "Quantum technologies need a quantum energy initiative," PRX Quantum, 2022, 3 (2), 020101.
17. Landauer R., "Irreversibility and heat generation in the computing process," IBM Journal of Research and Development, 1961, 5 (3), 183–191.
18. Pasqal / de Castro A., "Quantum computing in the roadmap to greener calculations," Pasqal Blog, 2023. https://www.pasqal.com/blog/quantum-computing-rethinking-energy-consumption/
19. Quantum Computing Report, "Can quantum computers address the AI energy problem?" Citing Cornell University study, 2025. https://quantumcomputingreport.com/can-quantum-computers-address-the-ai-energy-problem/
20. Hamdi S.M., Zuhori S.T., "A compare between Shor's quantum factoring algorithm and General Number Field Sieve," IEEE International Conference on Informatics, Electronics & Vision (ICIEV), 2014. https://ieeexplore.ieee.org/document/6919115
21. Regev O., "An efficient quantum factoring algorithm," arXiv:2308.06572 [quant-ph], 2023. (Covered in: Quanta Magazine, October 2023.)
22. Pomerance C., "A tale of two sieves," Notices of the American Mathematical Society, 1996, 43 (12), 1473–1485.
23. Lenstra A.K., Lenstra H.W. Jr. (Eds.), "The Development of the Number Field Sieve," Lecture Notes in Mathematics, Springer-Verlag, 1993, vol. 1554.