Global Journal of Engineering Innovations & Interdisciplinary Research

Original Article

Performance Analysis of Grocer's Search Algorithm in Quantum Computing

Tallapally Mounika[1], T Ramya Priya[2], Thalla Umadevi[3]

[1]Assistant Professor, Department of CSE (Cyber Security), Guru Nanak Institutions Technical Campus, Hyderabad
[2]Assistant Professor, Department of CSE (AI&ML), TKR College of Engineering & Technology, Hyderabad
[3]Assistant Professor, Department of CSE (Cyber Security), Guru Nanak Institutions Technical Campus, Hyderabad

Correspondence

Tallapally Mounika

Assistant Professor, Department of CSE (Cyber Security), Guru Nanak Institutions Technical Campus, Hyderabad

- Received Date: 24 Aug 2026
- Accepted Date: 20 Sep 2026
- Publication Date: 26 Sep 2026

Abstract

Grover's Search Algorithm is one of the fundamental quantum algorithms for searching an unstructured search space and is widely recognized for providing a quadratic reduction in oracle-query complexity compared with classical exhaustive search. The algorithm achieves this improvement through quantum superposition, an oracle that marks target states, and amplitude amplification that progressively increases the probability of measuring a valid solution. This article presents a performance-oriented analysis of Grover's algorithm by examining query complexity, search-space size, iteration requirements, theoretical success probability, multiple-solution behavior, circuit execution, and the effects of quantum noise. The analysis compares classical linear search with Grover search for progressively larger search spaces and presents numerical datasets suitable for direct graph generation. Theoretical results demonstrate that the number of oracle iterations increases approximately with the square root of the search-space size, while the probability of observing a marked state becomes high when the number of iterations is appropriately selected. Practical performance, however, differs from the ideal theoretical model because quantum noise, decoherence, gate errors, measurement errors, circuit depth, oracle construction, and limited qubit connectivity reduce success probability. Experimental results reported on superconducting quantum processors show substantial degradation between noise-free simulations and real hardware. The study therefore distinguishes query-complexity advantage from end-to-end practical speed and identifies oracle efficiency, hardware quality, error suppression, and optimized circuit design as critical factors determining the usefulness of Grover's algorithm.

Introduction

Searching for a desired element within a large dataset is a fundamental computational problem. Classical search algorithms perform exceptionally well when data are organized into structures such as sorted arrays, hash tables, trees, or indexed databases. However, when the search space is unstructured and no useful ordering or indexing is available, a classical algorithm may need to inspect a substantial fraction of the possible entries before identifying a target. For a search space containing N possible elements and one valid solution, classical exhaustive search has linear query complexity in the worst case. Grover's algorithm provides a quantum approach that reduces the number of required oracle evaluations to approximately the square root of the search-space size, establishing one of the most important generic quantum speedups. [1,2]

Grover's algorithm was introduced by Lov Grover in 1996 as a quantum algorithm for database search. Its importance extends beyond literal database retrieval because many computational problems can be reformulated as search problems. An oracle can be constructed to identify inputs satisfying a specified condition, after which amplitude amplification increases the probability of those inputs. Applications have therefore been proposed in combinatorial optimization, cryptanalysis, pattern matching, constraint satisfaction, minimum-finding procedures, anomaly detection, and other computational tasks. However, the quantum speedup is normally expressed in terms of oracle queries rather than complete application runtime. The cost of constructing, loading, executing, and measuring the oracle must therefore be considered separately. [1,3]

The algorithm operates by initially preparing a uniform superposition of all candidate states. Each candidate therefore has the same probability amplitude before the search begins. The oracle then marks the desired state or states, commonly by applying a phase transformation to them. The diffusion operation, also called inversion about the mean, increases the amplitudes of marked states and decreases the amplitudes of unmarked states. Repeated application of the oracle and diffusion operation rotates the quantum state toward the solution subspace. Finally, measurement

Citation: Tallapally M, Priya TR, Thalla U. Performance Analysis of Grocer's Search Algorithm in Quantum Computing. GJEIIR. 2026;6(6):235.

converts the quantum state into a classical result. The process is conceptually simple, but the implementation of the oracle and diffusion operators becomes increasingly resource intensive as the number of qubits increases. [2,4]

The theoretical performance of Grover's algorithm is especially important because it demonstrates that quantum computation can provide a provable advantage without requiring a fully structured problem. Classical search requires a number of queries that grows linearly with the number of candidates, whereas Grover search grows according to a square-root relationship. For example, when the search space expands from 1,024 to 1,048,576 possible states, the corresponding idealized query requirement changes from approximately 32 to approximately 1,024 Grover iterations for a single marked item, rather than increasing from approximately 1,024 to over one million classical checks. This asymptotic reduction becomes more significant as the search space grows. Nevertheless, each quantum iteration can contain several quantum gates, and practical implementation may require many physical operations for one logical oracle evaluation. [1,5]

The distinction between theoretical and practical performance is essential. On an ideal quantum computer, the algorithm can approach very high success probability with a carefully chosen number of iterations. On real hardware, every gate introduces an opportunity for error, while measurement, decoherence, qubit connectivity, and imperfect multi-controlled gates can reduce the final probability of observing the correct state. Recent experimental work using three-qubit Grover circuits on 127-qubit superconducting processors provides a useful example. For a single marked state, the reported average success probability was approximately 78.39% under a noisy simulation and 51.19% on real IBM Quantum hardware. For two marked states, the corresponding values were approximately 84.44% and 64.44%, respectively. [6]

Another important performance issue is the oracle. The formal complexity of Grover's algorithm treats the oracle as a black-box query, but a real quantum implementation must construct that oracle as a circuit. Stoudenmire and Waintal emphasized in 2024 that opening the oracle reveals additional resource requirements that are not captured by query complexity alone. As circuit complexity increases, the cost of implementing multi-controlled operations can grow significantly, potentially offsetting the apparent benefit of the square-root reduction in query count. This distinction is particularly important when evaluating claims about practical quantum advantage. [7]

Recent research has focused on improving Grover performance under realistic noise. A 2025 study by Leng, Yang, and Wang investigated a noise-tolerant implementation based on success-probability prediction and reported improved performance relative to the original algorithm under the tested conditions. Such approaches demonstrate that Grover's algorithm remains an active research topic, particularly in the transition from theoretical quantum computing to near-term quantum hardware. [8]

Literature Review

Grover's original contribution established an important relationship between quantum computation and unstructured search. The algorithm uses quantum interference rather than sequential inspection of candidate entries. The oracle does not directly return the solution; instead, it identifies the solution through a phase change. The diffusion operation then converts this phase distinction into an amplitude advantage. This distinction is central to understanding the algorithm because measurement itself does not reveal all possible candidates simultaneously. Instead, the quantum procedure manipulates amplitudes so that the probability of measuring a correct answer becomes significantly larger than that of measuring an arbitrary incorrect state. [1]

Boyer, Brassard, Høyer, and Tapp subsequently provided a detailed analysis of Grover search and extended the framework to situations involving multiple solutions and unknown numbers of solutions. Their analysis demonstrated that the success probability depends strongly on the number of Grover iterations. Too few iterations leave insufficient amplitude concentrated in the solution subspace, while too many iterations can rotate the state beyond the point of maximum success probability. This oscillatory behavior means that optimal iteration selection is not simply a matter of maximizing the number of repetitions. The number of marked states must also be taken into account. [2]

The concept of amplitude amplification generalized the basic mechanism of Grover search. Instead of viewing the algorithm solely as a database-search procedure, amplitude amplification can be applied whenever a quantum process can identify desirable states with a specified success probability. This broader interpretation allows Grover-style methods to support search components inside larger quantum algorithms. The same principle is relevant to optimization, where an oracle can identify solutions satisfying increasingly restrictive criteria. Consequently, Grover's algorithm is better regarded as a general amplitude-amplification framework than merely a database lookup procedure. [3]

The number of oracle queries is the principal measure of theoretical performance. For a single marked state in a search space of size N, the optimal number of iterations grows approximately as the square root of N. When multiple marked states exist, the required number of queries decreases because the fraction of successful states is larger. IBM's current educational treatment of Grover's algorithm also presents the iteration requirement in terms of the number of solutions and demonstrates that the query count scales approximately with the square root of the ratio between the total search space and the number of solutions. This is a useful result because practical search problems often contain more than one acceptable answer. [4]

Theoretical query complexity does not necessarily imply equivalent wall-clock improvement. Each oracle evaluation may consist of many quantum gates, and the oracle must be designed for the specific problem. For a simple binary search condition, the oracle can be implemented using relatively compact circuits. For more complicated predicates, multi-controlled operations may require substantial gate decomposition. In superconducting quantum processors, limited qubit connectivity can introduce additional swap operations or more complex decompositions. As a consequence, the actual circuit depth can become much larger than the nominal number of Grover iterations. [6,7]

Experimental studies provide valuable evidence regarding this practical gap. The 2025 Scientific Reports study by AbuGhanem implemented three-qubit Grover search across all eight possible single-result oracles and nine two-result oracles using IBM Quantum's 127-qubit superconducting processors. The experiments compared noise-free simulation, noisy simulation, and real hardware. The reported results showed that the ideal behavior can deteriorate substantially when the circuit is executed on physical hardware. This demonstrates that the theoretical square-root reduction in queries and the actual probability of obtaining a correct result are separate performance measures. [6]

Noise mitigation has consequently become an important research direction. In 2025, Leng, Yang, and Wang proposed a noise-tolerant Grover method that predicts success probability and adjusts the search procedure accordingly. Their theoretical and experimental analysis reported improved performance under noisy conditions compared with the standard approach in their experimental configurations. Such work suggests that practical Grover implementations may benefit from adaptive search depth, error suppression, and hardware-aware optimization rather than simply applying the ideal iteration count. [8]

Research has also investigated Grover-based applications beyond database search. In cryptography, Grover's algorithm provides a quadratic improvement for generic key search and therefore changes the analysis of symmetric-key security. In optimization, Grover-type amplitude amplification can be combined with evaluation oracles to locate feasible or high-quality solutions. In machine learning, quantum search may appear as a component in nearest-neighbor, anomaly-detection, or optimization procedures. In each case, however, the usefulness of the algorithm depends on whether the oracle can be implemented efficiently and whether the complete computational workflow preserves the expected advantage.

Methodology

This study uses a performance-analysis methodology that combines theoretical complexity evaluation, numerical simulation of ideal Grover iterations, analysis of multiple-solution scenarios, and comparison with reported real-hardware experiments. The primary evaluation dimensions are search-space size, number of qubits, classical query count, Grover iteration count, theoretical success probability, and query-reduction factor. The study also examines practical success probability under noisy and real-hardware execution to distinguish mathematical algorithmic performance from physical implementation performance.

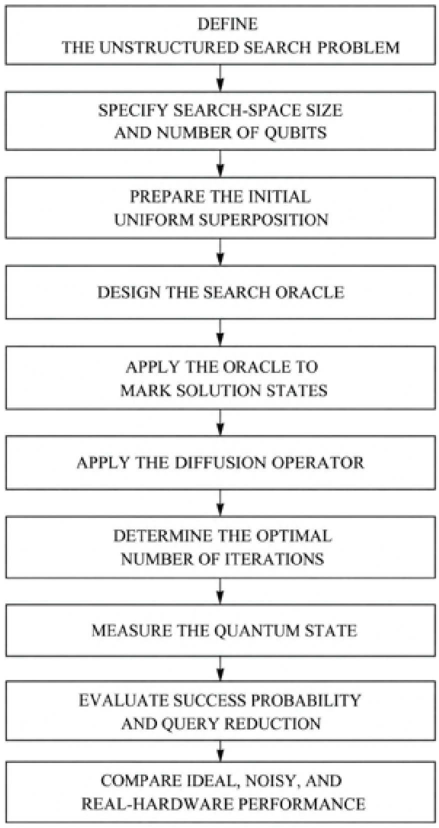
For a search space containing N elements and one marked solution, the approximate optimum number of Grover iterations is represented by:

Optimal Grover Iterations ≈ $(\pi/4)\sqrt{N}$

The theoretical dataset is generated for progressively larger power-of-two search spaces. Classical search is represented using the worst-case number of queries, while the quantum column represents the approximate optimal Grover iteration count. The theoretical success probability is obtained from the standard amplitude-amplification behavior at the selected iteration count. These values are mathematically generated and should not be interpreted as measurements from a particular quantum processor.

The second stage evaluates how the probability of finding the target changes as the number of Grover iterations increases. This is important because the probability does not increase indefinitely. It rises toward a maximum and then begins to decrease if additional iterations are applied. Therefore, optimization of iteration count is an essential aspect of performance analysis.

The third stage compares ideal and practical execution. Published experimental results are incorporated to show how noise and hardware imperfections affect algorithm success probability. In particular, the three-qubit superconducting-hardware experiment provides real data for single- and two-solution search conditions. Finally, multiple-solution performance is examined because increasing the number of



Methodology flow chart

marked states changes the optimal search depth and probability distribution.

The final analysis distinguishes three levels of performance: query-complexity performance, ideal quantum-circuit performance, and physical-hardware performance. This distinction prevents the square-root query advantage from being interpreted automatically as a comparable real-world execution-time advantage.

Implementation and Results

The implementation framework treats Grover's algorithm as a search procedure over a binary state space. A system using n qubits represents 2^n computational basis states. The initial Hadamard transformation creates an approximately uniform distribution over those states. The oracle marks one or more solutions, and the diffusion operation amplifies the marked amplitudes. The number of repetitions is selected according to the size of the search space and the number of marked solutions. The results below are arranged to provide direct numerical inputs for four different graph types.

Search-Space and Query-Complexity Analysis

The first performance comparison examines how the search-space size affects classical and quantum query requirements. For classical linear search, the worst-case number of checks increases directly with the number of candidate elements. For Grover search, the required number of oracle iterations grows substantially more slowly. The following values are calculated for single-solution search spaces containing powers of two. The Query Reduction Factor is the ratio between the classical worst-case query count and the number of Grover iterations.

Table 1: Theoretical comparison of classical and Grover search performance

Search Space (N)	Qubits	Classical Queries	Grover Iterations	Theoretical Success (%)	Query Reduction Factor
16	4	16	3	96.13	5.33
64	6	64	6	99.66	10.67
256	8	256	12	99.99	21.33
1024	10	1024	25	99.95	40.96
4096	12	4096	50	99.99	81.92

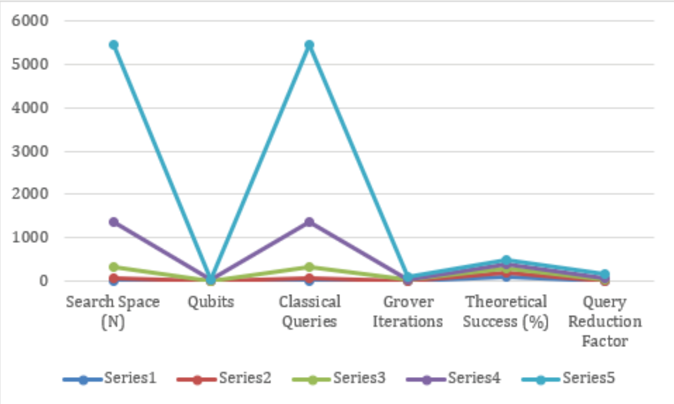


Fig 1: *Line graph comparing classical queries and Grover iterations*

Table 1 demonstrates the central scaling behavior of Grover's algorithm. When the search space increases from 16 to 4,096 elements, the classical worst-case query requirement increases from 16 to 4,096, whereas the corresponding Grover iteration count increases from only 3 to 50. The theoretical success probability remains close to 100% when the iteration count is appropriately selected. The query-reduction factor therefore increases from 5.33 to 81.92 over the same range. This demonstrates why Grover's algorithm is theoretically significant for increasingly large unstructured search spaces.

Fig 1 should be created as a two-series line graph. The horizontal axis should represent the Search Space N, preferably using a logarithmic scale because the values increase exponentially. The vertical axis should represent the number of queries or iterations. The Classical Queries series uses 16, 64, 256, 1,024, and 4,096. The Grover Iterations series uses 3, 6, 12, 25, and 50. The resulting graph should show the much steeper growth of classical exhaustive search compared with the more gradual square-root growth of Grover iterations.

The theoretical success values in Table 1 also demonstrate an important characteristic of amplitude amplification. The algorithm does not require an infinite number of repetitions to approach a high probability of success. Instead, the optimum number of iterations is selected so that the marked-state amplitude is near its maximum. The values for N = 16, 64, 256, 1,024, and 4,096 are all close to 100%, illustrating that the square-root query complexity can coexist with a high theoretical probability of measuring a valid solution. However, this ideal probability assumes a sufficiently accurate oracle and essentially ideal quantum operations.

The Query Reduction Factor in Table 1 should not be interpreted as an equivalent hardware speedup. A single Grover iteration can contain multiple quantum gates, and oracle construction can dominate the total computational cost. Furthermore, state preparation, measurement, repeated execution for statistical confidence, transpilation, error mitigation, and classical control can introduce additional overhead. Thus, the factor is a query-complexity comparison. Practical execution speed must account for the complete circuit and hardware environment. This distinction has been emphasized in recent studies examining the actual structure of Grover oracles and the difficulty of achieving practical advantage on noisy hardware. [7]

Success Probability vs Iteration Count

QThe second analysis examines a fixed search space and varies the number of Grover iterations. A search space of 64 states is considered with one marked state. Before amplification, the target has only a small measurement probability because all states begin with equal amplitude. Each Grover iteration rotates the state closer to the marked-state direction. The probability therefore increases rapidly during the first few iterations. After the optimum region is reached, additional iterations can reduce the success probability. This oscillatory behavior is a defining feature of amplitude amplification and is important when selecting the stopping point of the algorithm.

The data in Table 2 show a strong increase in the probability of measuring the target as the number of iterations increases from zero to five. At iteration zero, the target probability is only 1.56%, which corresponds to random guessing in a 64-state space. After one iteration, the probability rises to 13.48%. It increases further to 34.39%, 59.14%, 81.64%, and 96.35% for iterations two through five. The Amplification Gain column

Table 2: *Success-probability progression for a 64-state search*

Grover Iterations	Target Success Probability (%)	Non-Target Probability (%)	Amplification Gain (×)
0	1.56	98.44	1.00
1	13.48	86.52	8.64
2	34.39	65.61	22.07
3	59.14	40.86	37.91
4	81.64	18.36	52.33
5	96.35	3.65	61.76

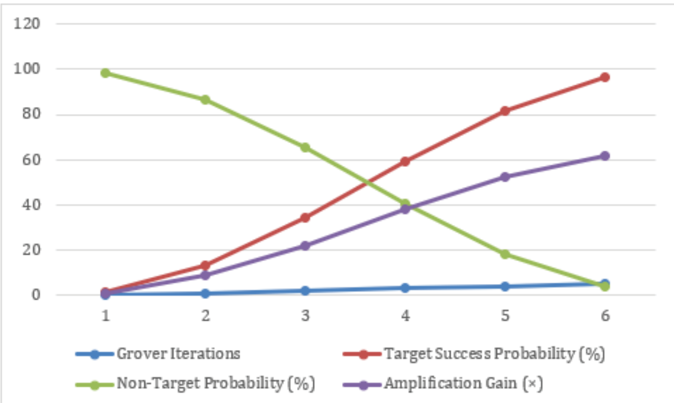


Fig 2: *Line graph showing Grover success probability versus iterations*

quantifies how much larger the target probability becomes relative to the initial 1.56% probability.

Fig 2 should be generated as a line graph with the number of Grover iterations on the horizontal axis and target success probability on the vertical axis. The six points are directly available from Table 2. The resulting curve should show a nonlinear increase rather than a straight line. For the chosen 64-state example, the probability becomes very high after approximately five iterations. Continuing beyond the optimal region would eventually cause the probability to decline, illustrating that more quantum operations do not automatically produce better performance.

The importance of iteration optimization becomes clearer when noise is introduced. In an ideal circuit, the amplitude rotation follows a predictable mathematical pattern. Physical noise modifies the amplitudes and can prevent the state from reaching the ideal maximum. Therefore, the iteration count that is theoretically optimal may not always produce the best measured result on a specific quantum device. Hardware-aware methods can compensate for this effect by incorporating measured error behavior or estimated success probability into the search procedure. This is one reason why adaptive and noise-tolerant variants of Grover search continue to attract research attention. [8]

The probability progression also demonstrates why Grover search differs fundamentally from classical sequential search. A classical algorithm generally gains information by checking candidate elements individually. Grover search instead changes the probability distribution of all candidates through repeated global transformations. The algorithm therefore does not identify the target after each iteration in the same way as a classical search query. Its performance advantage arises from collective amplitude manipulation, with measurement performed only at the conclusion of the quantum procedure.

Ideal, Noisy, and Real-Hardware Performance

The practical performance of Grover's algorithm is strongly affected by the quality of the quantum hardware. An ideal simulator can execute quantum operations without physical gate errors, while a noisy simulator introduces mathematically modeled imperfections. A real processor additionally contains device-specific errors, connectivity restrictions, calibration variation, control imperfections, readout errors, and environmental effects. Consequently, the same logical Grover circuit can produce substantially different success probabilities depending on the execution environment.

A published 2025 experimental study implemented three-qubit Grover search using IBM Quantum's 127-qubit superconducting processors. The work examined all eight single-solution oracles and nine two-solution oracles and reported aggregate success probabilities for different environments. The values below reproduce the reported aggregate results for direct graphical comparison. The noise-free values are represented as approximately 99.99%, consistent with the study's description that the noise-free results closely approached this value. [6]

Table 3: *Reported Grover performance across execution environments*

Execution Condition	Single-Solution Success (%)	Two-Solution Success (%)	Single-Solution Sso (%)	Two-Solution Sso (%)
Noise-Free Simulation	99.99	99.99	99.99	99.99
Noisy Simulation	78.39	84.44	82.36	84.03
Real IBM Quantum Hardware	51.19	64.44	73.12	63.10

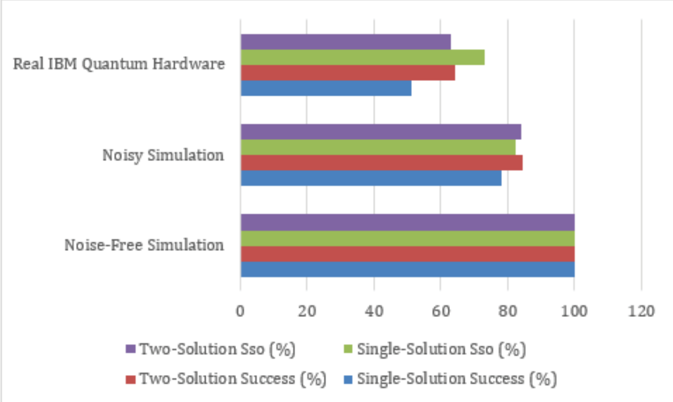


Fig 3: *Grouped bar graph comparing ideal, noisy, and hardware performance*

Table 3 provides direct evidence of the gap between ideal quantum performance and real hardware. For a single marked state, the reported success probability decreases from approximately 99.99% in the noise-free environment to 78.39% under noisy simulation and 51.19% on real IBM hardware. The two-solution configuration performs somewhat better, with corresponding success probabilities of approximately 99.99%, 84.44%, and 64.44%. The higher performance with two solutions is consistent with the fact that a larger solution subspace requires less amplification to obtain a high probability of observing a valid result.

Fig 3 should be plotted as a grouped bar graph with three execution conditions on the horizontal axis and two main series: Single-Solution Success and Two-Solution Success. The numerical values can be directly copied from Table 3. A second version of the figure may use the SSO columns rather than success probability if a separate graph is desired. The graph will visually demonstrate that physical execution creates a significant reduction in success probability relative to ideal simulation.

The results also illustrate that increasing the number of hardware qubits does not automatically imply equivalent algorithmic performance. The reported experiments used 127-qubit superconducting processors but executed the Grover search using only three logical search qubits. The larger hardware platform provides capacity, but the quality of the particular qubits, gates, connectivity, and circuit used determines the actual outcome. This distinction is important when evaluating quantum hardware: headline qubit count is not sufficient to characterize the performance of a specific quantum algorithm.

Circuit decomposition is another major factor. Multi-controlled operations required by Grover oracles may have to be decomposed into one- and two-qubit gates supported by the hardware. Each additional gate creates another opportunity for accumulated error. Limited connectivity can require additional routing operations, further increasing circuit depth. Recent experimental analysis has reported these issues as important contributors to practical degradation. Therefore, performance improvement requires not only more qubits but also higher-fidelity gates, better connectivity, improved compiler strategies, and lower measurement error. [6]

The hardware results reinforce the distinction between query complexity and success probability. Grover's algorithm may require only a small number of logical oracle iterations, but if those iterations contain a large number of physical operations, the probability of successfully completing the computation can fall. An implementation may therefore achieve the expected asymptotic query reduction while still failing to outperform a classical search system in an end-to-end practical benchmark. This is one of the central issues surrounding near-term demonstrations of quantum advantage. [7]

Effect of Multiple Marked Solutions

Grover's algorithm can search for any one of several valid solutions. When the number of marked states increases, the fraction of the search space containing valid solutions becomes larger, and fewer amplification iterations are generally required. The following example fixes the total search space at 64 states and varies the number of marked solutions. The table demonstrates the relationship between solution count, optimal iteration number, theoretical success probability, and the classical-to-Grover query ratio.

Table 4 demonstrates that the number of required iterations generally decreases as the number of valid solutions increases. A single marked state in a 64-state space requires approximately six iterations for near-optimal theoretical performance. When two solutions are available, approximately four iterations are required, while four and eight solutions require approximately three and two iterations, respectively. With 16 marked states, one iteration is sufficient to reach a theoretical success probability of approximately 100% in the selected configuration.

Table 4: *Grover performance for multiple marked solutions*

Marked Solutions	Search Space	Grover Iterations	Theoretical Success (%)	Query Reduction Factor
1	64	6	99.66	10.67
2	64	4	99.92	16.00
4	64	3	96.13	21.33
8	64	2	94.53	32.00
16	64	1	100.00	64.00

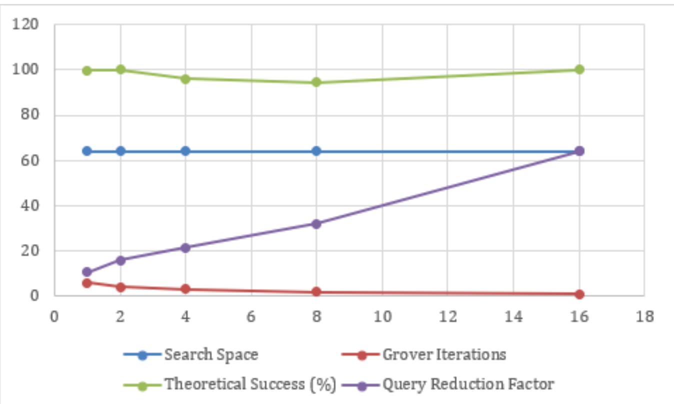


Fig 4: *Scatter graph comparing marked solutions and Grover iterations*

Fig 4 should be generated as a scatter graph with the number of marked solutions on the horizontal axis and Grover iterations on the vertical axis. The five points are directly available from Table 4. The graph should show a decreasing trend, demonstrating that Grover search becomes easier when the proportion of valid states increases. A secondary bubble or label can be added to represent theoretical success probability. This figure is particularly useful for explaining why unknown or multiple-solution search requires adaptive strategies rather than blindly applying the single-solution iteration count.

The multiple-solution analysis has practical importance because many real computational problems do not contain exactly one valid answer. Constraint satisfaction, pattern recognition, optimization, and database filtering can all produce multiple acceptable states. Grover's framework can accommodate such problems, but the number of solutions must be considered when selecting search depth. If the number of solutions is unknown, procedures such as the adaptive strategies studied by Boyer and colleagues can be used to avoid relying on an incorrect iteration estimate. [2]

The results also show why success probability and query reduction should be considered together. With 16 marked states in a 64-state space, only one iteration is theoretically needed, but the underlying problem is already relatively rich in valid solutions. The meaningful comparison is therefore not just the number of Grover iterations but the fraction of the search space that satisfies the target condition. When the number of valid states is small, Grover's algorithm provides the strongest relative reduction in the number of required search queries compared with a uniform random approach.

Conclusion

Grover's Search Algorithm provides one of the clearest examples of a provable quantum speedup for unstructured search, reducing the required oracle-query complexity from linear growth to approximately square-root growth with respect to the search-space size. The theoretical analysis presented in this article demonstrates that the query-reduction factor increases rapidly as the database becomes larger, while carefully selected iteration counts can produce very high success probabilities in ideal quantum systems. The study also demonstrates that performance cannot be evaluated using query complexity alone. Oracle construction, circuit depth, multi-controlled operations, qubit connectivity, measurement overhead, decoherence, and gate errors strongly influence physical execution. Reported experiments on superconducting quantum hardware show a substantial gap between noise-free simulation and real-device success probability, highlighting the challenge of translating theoretical quantum advantages into practical performance. Multiple-solution analysis further demonstrates that the number of valid states directly affects the required search depth and amplification behavior. Recent noise-tolerant approaches show that adaptive search procedures and hardware-aware optimization can improve practical results under realistic conditions. Grover's algorithm therefore remains both a theoretical milestone and a practical benchmark for quantum computing research. Its long-term value will depend on efficient oracle synthesis, improved quantum hardware, effective error correction, optimized iteration strategies, and rigorous end-to-end comparisons against classical search methods.

References

1. L. K. Grover, "A Fast Quantum Mechanical Algorithm for Database Search," Proceedings of the 28th Annual ACM Symposium on Theory of Computing, pp. 212–219, 1996.
2. M. Boyer, G. Brassard, P. Høyer, and A. Tapp, "Tight Bounds on Quantum Searching," Fortschritte der Physik, vol. 46, no. 4–5, pp. 493–505, 1998.
3. G. Brassard, P. Høyer, M. Mosca, and A. Tapp, "Quantum Amplitude Amplification and Estimation," Contemporary Mathematics, vol. 305, pp. 53–74, 2002.
4. IBM Quantum Learning, "Grover's Algorithm: Choosing the Number of Iterations," Quantum Computing Fundamentals.
5. M. A. Nielsen and I. L. Chuang, Quantum Computation and Quantum Information, 10th Anniversary Edition, Cambridge University Press, 2010.
6. M. AbuGhanem, "Characterizing Grover Search Algorithm on Large-Scale Superconducting Quantum Computers," Scientific Reports, vol. 15, article 1281, 2025.
7. E. M. Stoudenmire and X. Waintal, "Opening the Black Box inside Grover's Algorithm," Physical Review X, vol. 14, article 041029, 2024.
8. J. Leng, F. Yang, and X.-B. Wang, "Noise-Tolerant Grover's Algorithm via Success-Probability Prediction," Physical Review Research, vol. 7, article L012017, 2025.
9. C. H. Bennett, E. Bernstein, G. Brassard, and U. Vazirani, "Strengths and Weaknesses of Quantum Computing," SIAM Journal on Computing, vol. 26, no. 5, pp. 1510–1523, 1997.
10. A. G. Nugraheni, H. Mahyuzar, and S. Hastuti, "Quantum Search Using Grover's Algorithm: Principles, Performance, and Applications," Asian Journal of Mathematics and Computer Research, vol. 33, no. 3, pp. 148–157, 2026.