Fuzzy Systems and Soft Computing
ISSN : 1819-4362

QUANTUM CRYPTOGRAPHY: DESIGN AND IMPLEMENTATION OF GROVER'S ALGORITHM ON 3, 4, AND 5-QUBIT SYSTEMS

Kumari Neha Research Scholar, Department of Physics, Patna University, Patna, Bihar Patna, 800001 singh.neha624@gmail.com
Dr. Amrita, Assistant Professor, Department of Physics, Patna Women's College, Bailey Road, Patna Patna, 800001 amrita.phy@patnawomenscollege.in

Abstract

Quantum Search Algorithm, particularly Grover's Algorithm, has emerged as a promising methodology for enhancing the efficiency of search engines and solving various computational problems. This paper presents a comprehensive study of the design and implementation of Grover's Algorithm on quantum systems with 3, 4, and 5 qubits. Theoretical explanations of Grover's method are provided, elucidating its step-by-step approach for enhanced understanding. Through detailed performance analysis, including gate count, Q-sphere representation, and measurement probability, the study demonstrates the reliability and effectiveness of Grover's Algorithm in searching unsorted databases. Results indicate that with the increase of number of qubits, the required number of quantum gates in the circuit also grows, with 3-qubit systems utilizing 13 gates, 4-qubit systems employing 27 gates, and 5-qubit systems utilizing 34 gates. Despite this increase, the designs exhibit minimal errors or noise, ensuring the accuracy of the outcomes. Visualization of quantum states using the Q-sphere representation offers a global perspective, aiding in the comprehension of algorithm behavior.

Keywords:
Quantum algorithm; Quantum cryptography; Qubit; Quantum circuit; Phase shift; Q-sphere.

1. Introduction

Quantum Search [1], [2] is a methodology that can be employed to enhance the efficiency of several traditional search engines. It conducts a comprehensive search for a viable resolution to a diverse array of issues. The aim is to determine, without previous knowledge about the information structure, if a given number p is a non-trivial factor of a big integer of size N. Finding the non-trivial factors may be accomplished by methodically going over the set until they are found. This problem normally requires around N operations; however, Grover's search strategy requires about $\sqrt{N}$ operations with a probability larger than 1/2.

Grover's Algorithm has the potential to be utilized for searching real databases, however, very few research papers have been published in this specific domain. Researchers regard Oracle as a virtual database for doing searches. The database's size and the quantum hardware's capabilities affect how long searches take on it. As such, using simulation tools to analyze the process of creating a quantum circuit is essential. In order to reduce the amount of quantum gates and design errors, this study uses IBM Quantum Lab, also known as the quantum programming studio, to properly create the quantum circuit for Grover's search algorithm on 3, 4, and 5-qubit systems utilizing various quantum gates.

We have also included a theoretical explanation of Grover's method, outlining the step-by-step approach to enhance understanding of the technique. The paper is organized as follows: The significance and useful applications of the Search Algorithm are described in Section 1. Section 2 analyses existing approaches and their conclusions concerning Grover's search algorithm. Grover's algorithm's operating notion is briefly explained in Section 3. Grover's method is implemented and the results are shown in Section 4 for systems with 3, 4, and 5 qubits. Finally, the discussion and conclusion are presented in Section 5.

2. Literature Survey

The 4-qubit quantum algorithm was implemented successfully on the ibmqx5 architecture in [3], and the results of that implementation were widely publicized. The simulation outcomes were then contrasted with the performance accuracy of their execution. The complexity of circuit design requires

greater hardware support for their implementation performance. A classical quantum search method for locating the shortest route in a given graph was introduced in [4]. It is not possible to use Grover's method to look up an item in a real database. Potential results were obtained by using the QCAD simulator to demonstrate theoretical quantum error correction in [5].
Grover's approach is applicable for searching a singular item within an unorganized database. In [6], an expanded version of Grover's approach was developed to facilitate the search of several objects in a vast and unsorted database. This approach has demonstrated consistent outcomes, similar to Grover's technique for searching a single item. The authors [7] successfully executed Grover's algorithm on an actual ibmqx4 quantum computer with 5 qubits. A Grover's search algorithm implementation with 4-qubit was created [8], and test results indicated that quantum computers are capable of properly solving only simple problems with limited input. In [9], circuit QED was utilized to create an interaction between two transmon qubits, with the aim of enhancing the efficiency of implementing Grover's search algorithm inside the nanosecond timeframe. An implementation of Grover's method was carried out by employing two trapped atomic ion qubits to enhance the speed of the search process. In [10] attempt was made to implement Grover's technique in a scalable system to enhance the likelihood of successfully finding the item being searched. A 3-qubit quantum circuit was designed and constructed [11], whereas the 2-qubit quantum algorithm was implemented [12] with a linear optical device.

Grover's Search Algorithm

The Grover's Search method, proposed by Lov Grover in 1996, [13] enables the quickest quantum search for an unsorted database. It realizes a temporal complexity of $O(\sqrt{n})$ and a special complexity of O(logN). For a list containing 107 entries, the number of iterations needed to search for a specific item is approximately 10000. The result will have a probability close to 1, and it will be extremely accurate.
Despite providing merely a quadratic speedup, Grover's technique is significant when N is big. Table 1 displays Grover's method, whereas Figure 1 demonstrates the circuit representation of Grover's method.

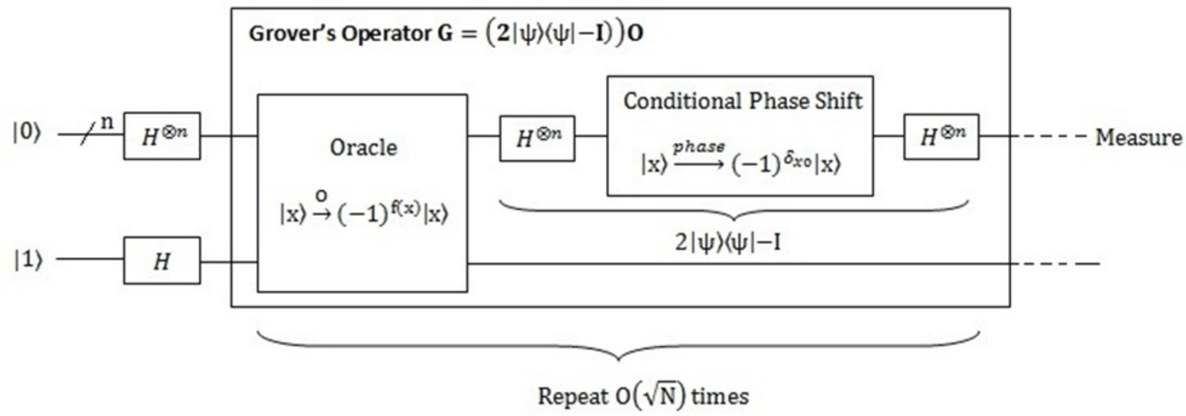

Figure 1: Grover's search algorithm method

Let's assume we have an unsorted database with N entries, which may be represented as a set $X = \{x0 , x1 , . . . , xN-1 \}$[14]. We are also given a Boolean function $f : X \rightarrow \{0, 1\}$. The objective is to locate an element x in X for which f(x) = 1. A better way to describe unstructured search is as a database search problem, in which we are given a database and we need to find an item that meets specific requirements [15]. For example, if we are given a database with N names in it, we could have to figure out where your name exactly is in the database. The term "unstructured" refers to the fact that the database's structure is not guaranteed. We can use binary search to find an element in logarithmic time given a sorted database. On the contrary, we lack any previous information on the contents of the database. The efficiency of doing a linear number of queries to find the target element cannot be increased while using classical circuits. Index numbers ranging from 0 to N − 1 are allocated to the N items, where N is equal to 2 raised to the power of n. The problem of searching may be modeled by a function f that takes an integer x as input from the set of indexes 0 to N−1.
The function f(x) is defined such that it returns a value of 1 if x is located during a search, and a value of 0 if x is not discovered during a search. The function f(x) may be defined as follows:

$$f(x) = \begin{cases} 0 \; if \; x \neq u \\ 1 \; if \; x = u \end{cases}$$

where the element that has to be sought is denoted by u.

Table 1: Grover's Algorithm

Input:

i) An Oracle O that transforms $|x\rangle|q\rangle O \rightarrow |x\rangle|q \oplus f(x)\rangle$

where f(x) = 0 for all $0 \leq x < N - 1$ and f(x) = 1 for x = u.

ii) State $|0\rangle$ with n + 1 qubits.

Output: A matching item was located. that is, u

Runtime: Has an O(1) chance of success and requires O(√N) operations.

Procedure:

Step 1: Set the starting state of the algorithm to $|0\rangle \otimes n$.

Step 2: On the first n qubits, apply the Hadamard transform; on the last qubit, apply the Hadamard and X gate as follows:

1 √N ∑ |x⟩[|0⟩−|1⟩ √2] N−1 x=0

Step 3: The next step is to execute Grover's Operation for R ≤ ⌈ π 4 √ 2N M ⌉ repetitions.

Step3.1: Use Oracle O to apply the following such that |x⟩ O → (−1) f(x) |x⟩

Step 3.2: Take the state that was acquired in Step 3.1 and apply the Hadamard transform H ⊗n to it.

Step 3.3: Apply the conditional phase shift operation using the equation |x⟩ phase → (−1) δx0|x⟩, to all states except the |0⟩ state, which receives a phase shift of −1.

Step 3.4: Use the H ⊗n Hadamard transform. Grover's operation will ultimately yield the following state:

[(2|ψ⟩⟨ψ| − I)O] R 1 √N ∑ |x⟩[|0⟩−|1⟩ √2] N−1 x=0 ≈ u [|0⟩−|1⟩ √2]

Step 4: Assess the initial n qubits in step four. The matching item u will be the outcome.

Grover's method is explained in a step-by-step manner as follows: The state |0⟩ ⊗n is where the algorithm starts. To obtain an equal superposition state, apply the Pauli X transform and Hadamard transform (HX) on the final qubit after performing the Hadamard transform H ⊗n on the first n qubits. This process may be expressed as follows [16]:

H ⊗n |0⟩ ⊗n = |ψ⟩ = 1 √N ∑ |x⟩ N−1 x=0 (2) HX|1⟩ = [|0⟩−|1⟩ √2]

The expression |ψ⟩ = 1/√N ∑ |i⟩ represents the superposition of all base states, where i ranges from 1 to N. Run the Grover's operator now. There are four separate stages in the Grover iteration quantum circuit. There are four steps involved in this process: the Oracle function, Hadamard transform, conditional phase shift operation on all computational basis states (except |0>, which is phase-shifted by -1), and Hadamard transform. The following operation is performed by the oracle, a unitary operator represented by the letter O:

|x⟩|q⟩ O → |x⟩|q ⊕ f(x)⟩

All items from 0 to N-1 have their indexes stored in the input register |x⟩. The symbol ⊕ indicates the addition modulo 2. In case f(x) = 1, the oracle qubit |q⟩ flips, and in all other cases it stays unchanged. The oracle qubit |q⟩ might be referred to as a "ancilla" or "ancillary qubit", which serves the purpose of incorporating additional qubits into the algorithm. If the value of x cannot be located, the application of the oracle to the state will not result in any alteration to the state. However, in the event that x is discovered, the final state is changed to a new state. Grover's approach makes extremely simple use of an oracle. The output that is produced when the oracle is applied to the inputs |x⟩|q⟩ is |x⟩|q ⊕ f(x)⟩. The initial two qubits remain unchanged when x takes on the values of 0 or 1. If the first two qubits are both 1, then the third qubit is inverted. Therefore, we obtain,

|x⟩ (|0⟩−|1⟩ √2) O → (−1) f(x) |x⟩ (|0⟩−|1⟩ √2)

Above equation can be written as [17],

|x⟩ O → (−1) f(x) |x⟩

Steps 2, 3, and 4 may be expressed as the tensor product of n Hadamard gates applied to the state vector (2|0⟩⟨0|). The expression H ⊗n = 2|ψ⟩⟨ψ| − I represents the tensor product of the Hadamard operator H with itself n times. The Grover's operator, designated as G, may be expressed as G =

(2|ψ⟩⟨ψ|−I). The user's text is a single letter, "O". The Grover's method may be described as a rotation of R times inside an angle $\theta/2$, where θ is a real value between 0 and $\pi/2$. This rotation occurs around the state |β⟩ and is limited to an angle of $\pi/4$.
|ψ⟩ ≡ √ N−M N |α⟩+ √ 1 N |β⟩
The state |α⟩ is defined as the sum of all |x⟩ states, except the state |u⟩, divided by the square root of N minus M. The state |β⟩ is simply equal to the state |u⟩. The rotation may be represented as R < ⌈ π 4 √ 2N M ⌉, implying that R has an upper bound of ⌈ π 4 √ 2N M ⌉. Consequently, R can be approximated as O(√N/M). If the value of M is equal to 1, meaning that the solution is discovered in a single iteration of Grover's method, then the complexity of R is O(√N).

3. Utilization and Performance Examination

A thorough examination of the system design, implementation, and performance results for three-, four-, and five-qubit quantum systems is given in this section. The program was ran on Quantum Programming Studio, an online simulator and integrated development environment (IDE) for building, debugging, and simulating quantum algorithms on platforms including Google Cirq, IBM Q, and Rigetti. An open-source simulator based on JavaScript is called Quantum-circuit. More than 20 qubits may be simulated by a quantum circuit utilizing node.js on a server or a web browser. It is possible to import quantum circuits from Quil and Open QASM. After designing the quantum circuit in the IDE, it may be exported to a number of quantum programming languages, including pyQuil and Open QASM, Quil, Qiskit, Cirq, Tensor Flow Quantum, QSharp, and QuEST. This allows for seamless translation between different quantum programming languages. It has been noted that the application of each quantum gate introduces a little inaccuracy to the quantum state. This phenomenon arises from the specific arrangement of circuits and the interplay between individual qubits, as well as their interaction with other qubits and gates.

4. Implementation with 3-Qubits

Figure 2 depicts the circuit diagram of a 3-qubit system using 13 gates. A 3-qubit quantum circuit exhibits no significant error rate. Figure 3 displays the Q-sphere with phase angle representation of the 3-qubit state, while Figure 4 shows the measurement probability of the marked state. It is essential to emphasize that, even in the case of a single qubit, the q-sphere and the Bloch sphere are not equivalent. In fact, the Bloch sphere provides a localized picture of the quantum state, similar to the phase disk, in which each qubit is seen independently. Examining the quantum state as a whole from a global perspective can provide greater insight into how registers of qubits, or multi-qubit states, behave when quantum circuits are applied. This global perspective and the quantum state are represented visually by the q-sphere. Therefore, the q-sphere should be the main visualization tool while investigating quantum algorithms and applications on modest numbers of qubits.
When n (number of qubits used) equals 3, there are 8 potential states ($N = 2^3$). Assuming that the answer is located at index 4 in the state |100⟩, we obtain the following:
|ψ⟩ O → 1 /2 (|000⟩+ |001⟩ + |010⟩ + |100⟩ + |011⟩+ |101⟩ + |110⟩+ |111⟩)
Therefore, when the oracle operator acts on the state |x⟩, it alters the amplitude of that state. Now, do the Hadamard transform H ⊗n on these basis states, resulting in:
|ψ⟩ H⊗ 2 → 1/ 2 (|000⟩ + |001⟩ + |010⟩ − |100⟩+ |011⟩ + |101⟩+ |110⟩+ |111⟩)
Following a conditional phase shift, we obtain 2, or 3 distinct superposition states as,
|ψ⟩ phase → 1 /2 (|000⟩ − |001⟩ − |010⟩ + |100⟩⟩− |011⟩− |101⟩− |110⟩− |111⟩)
After the acquired states are subjected to the Hadamard transform H ⊗n, the matched element on an index is achieved and is provided as,
|ψ⟩ → |100⟩
Therefore, Grover's technique requires two repetitions to reach the required state for n = 3.

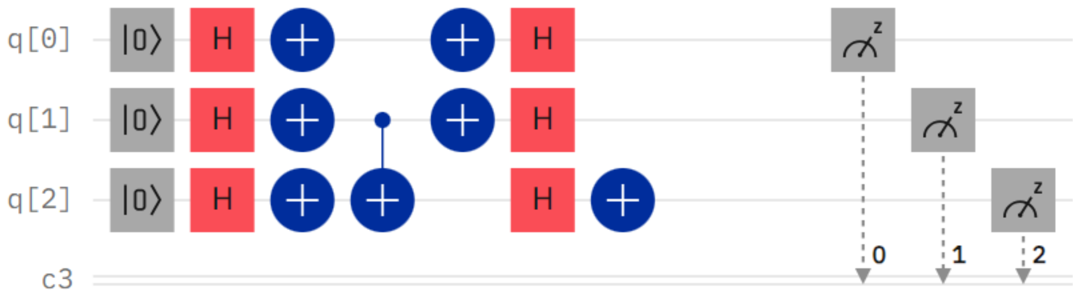

Figure 1: Circuit Diagram for the state |100⟩

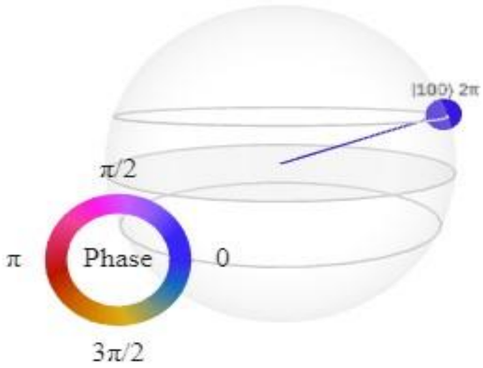

Figure 2: Q- Sphere of 3-qubit system representation for the state |100⟩

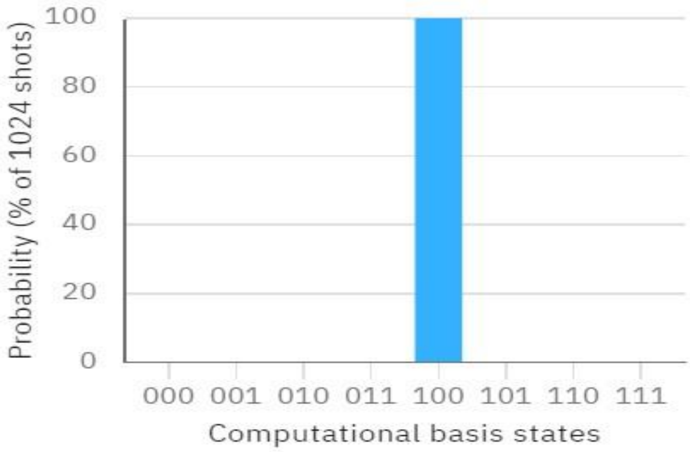

Figure 4: Probability Measurement for a 3-qubit system in the state |100⟩

5. Fourth-Qubit Implementation

Figure 5 depicts the circuit diagram for a 4-qubit system consisting of 27 gates. The design of the system is free from any noise or errors. Figure 6 displays the Q-sphere with phase angle representation of the 4-qubit state, whereas Figure 7 shows the measurement probability. When n is equal to 4, there are 16 alternative states, represented by $N = 2^4$. Assuming the answer is located at index 4, which is represented by |0100⟩, we have the following:

|ψ⟩ O → 1/2(|0000⟩ + |0001⟩ + |0010⟩ + |0100⟩ + |0011⟩ + |0101⟩ + |0110⟩ + |0111⟩ + |1000⟩ + |1001⟩ + |1010⟩ + |1011⟩ + |1100⟩ + |1101⟩ + |1110⟩ + |1111⟩)

Upon performing the Hadamard transform H ⊗n on |x⟩, the system's state is shown as:

|ψ⟩ H⊗ 2 → 1/2(|0000⟩ + |0001⟩ + |0010⟩ - |0100⟩ + |0011⟩ + |0101⟩ + |0110⟩ + |0111⟩ + |1000⟩ + |1001⟩ + |1010⟩ + |1011⟩ + |1100⟩ + |1101⟩ + |1110⟩ + |1111⟩)

Following a Conditional Phase Shift on all basis states other than |0000⟩, the outcome is displayed as follows:

|Ψ⟩ phase →1/2(|0000⟩ − |0001⟩ − |0010⟩ + |0100⟩ - |0011⟩ − |0101⟩ − |0110⟩ − |0111⟩ − |1000⟩ − |1001⟩ − |1010⟩ − |1011⟩ − |1100⟩ − |1101⟩ − |1110⟩ − |1111⟩

|ψ⟩ → |0100⟩

The matching is discovered when the states are subjected to H ⊗n. Therefore, for n = 4, three Grover's method repetitions are required to reach the target state.

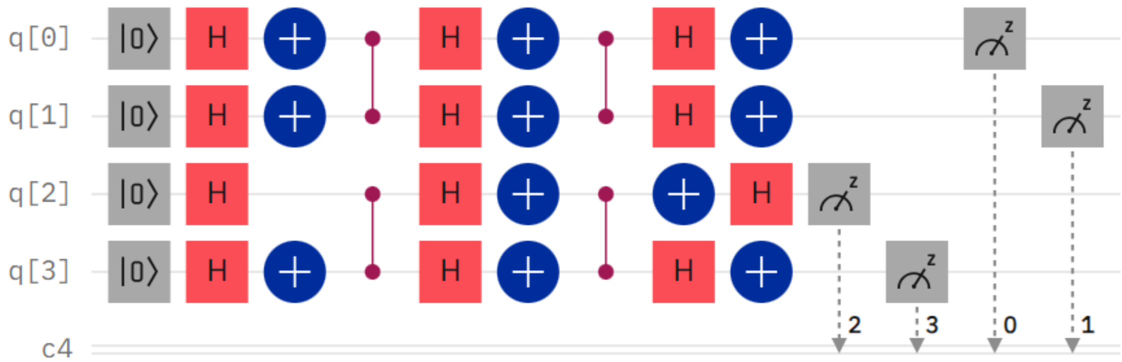

Figure 5: Circuit Diagram for the state |0100⟩

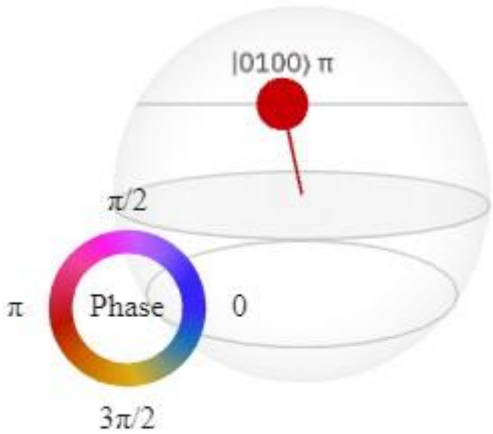

Figure 6: Q- Sphere of 4-qubit system representation for the state |0100⟩

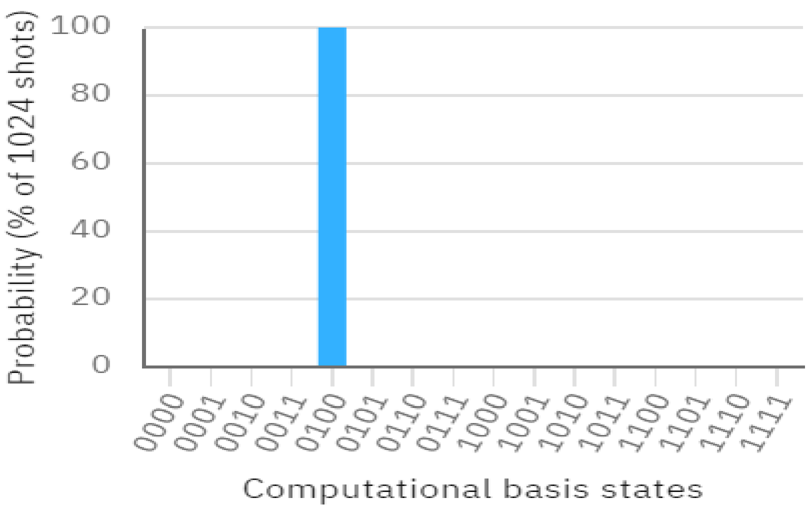

Figure 7: Probability Measurement for a 4-qubit system in the state |0100⟩

6. Using 5-Qubits for Implementation

Figure 8 depicts the circuit diagram for a 5-qubit system consisting of 34 gates. The design of the system is free from any noise or errors. Figure 9 displays the Q-sphere with phase angle representation of the 2-qubit state, whereas Figure 10 shows the measurement probability. When n is equal to 5, there are 32 alternative states, represented by N = 2^5. Assuming the answer is located at index 4, which is represented by |00100⟩, we have the following:

|ψ⟩ O → 1/2(|00000⟩ + |00001⟩ + |00010⟩ + |00100⟩ + |01000⟩ + |10000⟩ + |00011⟩ + |00101⟩ + |01001⟩ + |10001⟩ + |00110⟩ + |01100⟩ + |01010⟩ + |10010⟩ + |11000⟩ + |00111⟩ + |11100⟩ + |01110⟩ + |11001⟩ + |11010⟩ + |01101⟩ + |10110⟩ + |10011⟩ + |01011⟩ + |10101⟩ + |01011⟩ + |01111⟩ + |10111⟩ + |11011⟩ + |11101⟩ + |11110⟩ + |11111⟩)

After the Hadamard transform H ⊗n is applied to |x⟩, the system's state is shown as follows:

|ψ⟩ H⊗ 2 1/2(|00000⟩ + |00001⟩ + |00010⟩ - |00100⟩ + |01000⟩ + |10000⟩ + |00011⟩ + |00101⟩ + |01001⟩ + |10001⟩ + |00110⟩ + |01100⟩ + |01010⟩ + |10010⟩ + |11000⟩ + |00111⟩ + |11100⟩ + |01110⟩ + |11001⟩ + |11010⟩ + |01101⟩ + |10110⟩ + |10011⟩ + |01011⟩ + |10101⟩ + |01011⟩ + |01111⟩ + |10111⟩ + |11011⟩ + |11101⟩ + |11110⟩ + |11111⟩)

Following a Conditional Phase Shift on all basis states other than |00000⟩, the outcome is displayed as follows:

|Ψ⟩ phase →1/21/2(|00000⟩ - |00001⟩ - |00010⟩ + |00100⟩ - |01000⟩ - |10000⟩ - |00011⟩ - |00101⟩ - |01001⟩ - |10001⟩ - |00110⟩ - |01100⟩ - |01010⟩ - |10010⟩ - |11000⟩ - |00111⟩ - |11100⟩ - |01110⟩ - |11001⟩ - |11010⟩ - |01101⟩ - |10110⟩ - |10011⟩ - |01011⟩ - |10101⟩ - |01011⟩ - |01111⟩ - |10111⟩ - |11011⟩ - |11101⟩ - |11110⟩ - |11111⟩)

|ψ⟩ → |00100⟩

The matching is discovered after applying H ⊗n to the states.

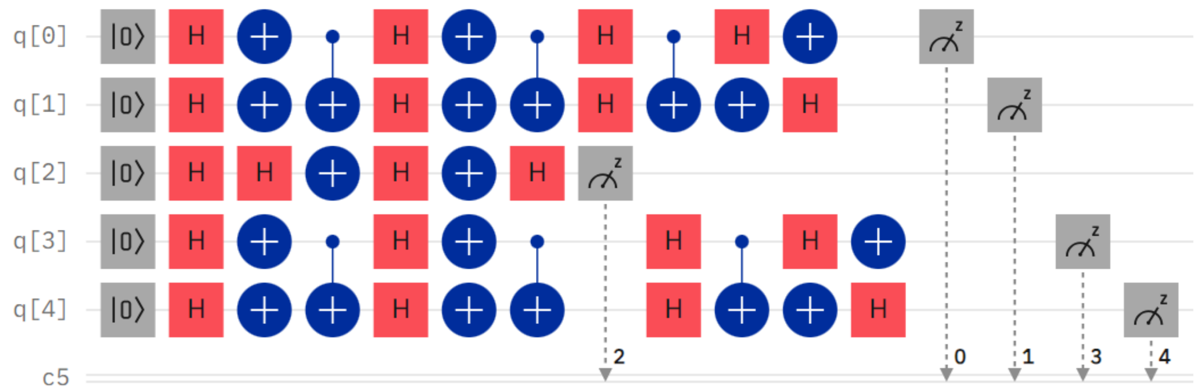

Figure 8: Quantum Circuit Diagram for the state |00100⟩

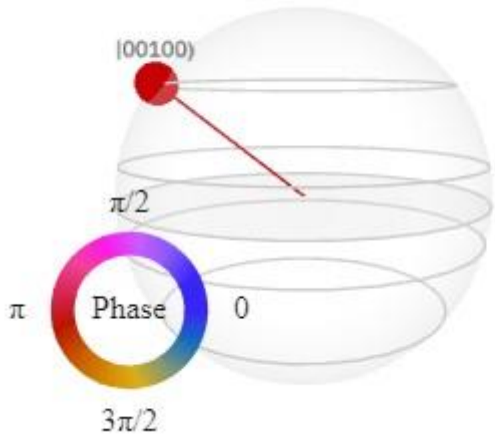

Figure 9: Q-sphere of a 5-qubit system representation for the state |00100⟩

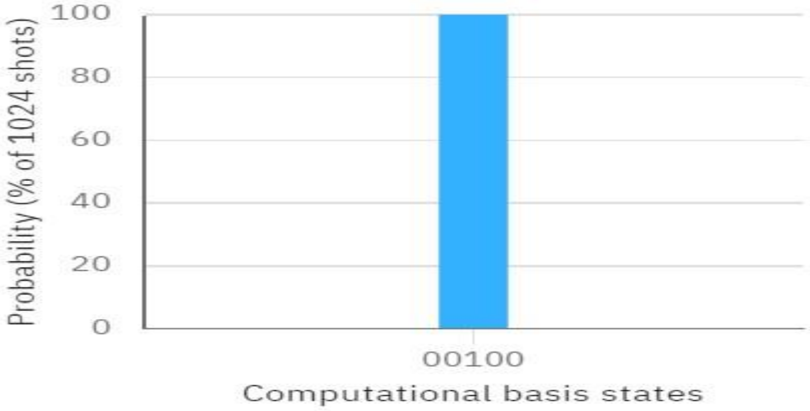

Figure 10: Probability measurement for a 5-qubit system for the state |00100⟩

Consequently, Grover's technique requires 4 iterations to reach the required state for n = 5.

7. Discussion and Conclusion

Quantum search algorithms have potential applications in the following areas: i) efficiently counting the number of solutions to a given search problem; ii) solving NP-complete problems more quickly; iii) finding cryptosystem keys more quickly [18], iv) determining the shortest path between two cities [19], and v) more quickly searching problems or extracting statistics from unstructured databases [20] than with classical computing. Grover's quantum search method was created and put into practice by the author on 3, 4, and 5 qubit systems. Analysis was provided about the quantity of gates utilized, the Q-sphere representation, and the likelihood of measurement.

The findings illustrate that for each qubit increase, the number of gates required in the quantum circuit grows, with 3-qubit systems utilizing 13 gates, 4-qubit systems employing 27 gates, and 5-qubit systems utilizing 34 gates. However, despite the increase in gate count, the designs remain free from significant errors or noise, ensuring the reliability of the results. Moreover, the study showcases the visualization of quantum states using the Q-sphere representation, providing a global perspective of the quantum state and aiding in the comprehension of algorithm behavior on small qubit registers. Overall, the research underscores the potential of Grover's Algorithm in revolutionizing search methodologies, particularly in database searches, and highlights the importance of accurate quantum circuit design and simulation tools in facilitating quantum algorithm development. Through rigorous analysis and simulation, the study contributes to advancing our understanding and utilization of quantum computing techniques for real-world applications.

8. Future Scopes of Grover's Algorithm in Quantum Cryptography

Grover's algorithm represents a transformative tool in quantum cryptography, offering the potential to revolutionize search processes and enhance security protocols. Its future applications span from optimizing brute-force attacks and Quantum Key Distribution to aiding in the development of post-quantum cryptographic protocols. However, realizing these potentials requires overcoming significant challenges in quantum hardware and system integration. As quantum computing continues to advance, Grover's technique is expected to have a significant impact on how cryptography develops in the future, providing strong protection against new quantum attacks.

References:

[1] Kaye P., Laflamme R., Mosca M., "An Introduction to Quantum Computing", Oxford University Press Inc., New York, 2007, pp. 1-276. https://doi.org/10.48550/arXiv.0708.026

[2] Nielsen M. A. and Chuang I. L., "*Quantum Computation and Quantum Information*", Cambridge University Press, New York, 2010, pp. 1-676. ISBN 1139495488.

[3] Zalka C., "Could Grover's Quantum Algorithm Help in Searching an Actual Database?", Quantum Physics, 1999, pp. 1-7. https://doi.org/10.1103/PhysRevA.62.052305

[4] Aghaei M. R. S., Zukarnain Z. A., Mamat A., Zainuddin H., "A Hybrid Algorithm For Finding Shortest Path In Network Routing", Journal Of Theoretical And Applied Information Technology,

2009, pp. 360-365. https://doi.org/10.24425/ijet.2022.139851
[5] Priya R. P., Baradeswaran A., "An efficient simulation of quantum error correction Codes", Alexandria Engineering Journal, Vol. 57, 2018, pp. 2167–2175. https://doi.org/10.1016/j.aej.2017.06.013
[6] Chen G., Fulling S. A., and Scully M. O., "Grover's Algorithm for Multiobject Search in Quantum Computing", Article in Lecture Notes in Physics, 1999, pp. 1-12. https://doi.org/10.1007/3-540-40894-0_15
[7] Abhijith J., Adedoyin A., Ambrosiano J., Anisimov P., Bärtschi A., Casper W., Chennupati G., Coffrin C., Djidjev H., Gunter D., Karra S., Lemons N., Lin S., Malyzhenkov A., Mascarenas D., Mniszewski S., Nadiga B., O'malley D., Oyen D., Pakin S., Prasad L., Roberts R., Romero P., Santhi N., Sinitsyn N., Swart P. J., Wendelberger J. G., Yoon B., Zamora R., Zhu W., Eidenbenz S., Coles P. J., Vuffray M. and Lokhov A. Y., "Quantum Algorithm Implementations For Beginners", Computer Science Emerging Technologies, 2020, pp. 1-94. https://doi.org/10.48550/arXiv.1804.03719
[8] Mandviwalla A., Ohshiro K., Ji B., "Implementing Grover's Algorithm on the IBM Quantum Computers", in Proc. 2018 IEEE Int. Conference on Big Data, 2018,pp. 2531-2537. https://doi.org/10.1109/BigData.2018.8622457
[9] Said T., Chouikh A., Essammouni K. and Bennai M., "Implementation of Grover quantum search algorithm with two transmon qubits via circuit QED", Quant. Phys. Lett., vol. 6, no. 1, 2017, pp. 29-35. https://doi.org/10.18576/qpl/060105
[10] Brickman K. A., Haljan P. C., Lee P. J., Acton M., Deslauriers L. and Monroe C., "Implementation of Grover's Quantum Search Algorithm in a Scalable System", Physical Review A, vol. 72, no. 5, 2005, , pp. 1-4. https://doi.org/10.1103/PhysRevA.72.050306
[11] Figgatt C., Maslov D., Landsman K. A., Linke N. M., Debnath S. and Monroe C., "Complete 3-Qubit Grover search on a programmable quantum computer", Nature Communications, vol. 8, no. 1918, 2017, pp. 1-9. https://doi.org/10.1038/s41467-017-01904-7
[12] Samsonov E., Kiselev F., Shmelev Y., Egorov V., Goncharov R., Santev A., Pervushin B. and Gleim A., "Modeling two-qubit Grover's algorithm implementation in a linear optical chip", Physica Scripta, vol. 95, no. 4, 2020. https://doi.org/10.1088/1402-4896/ab6523
[13] Luan L., Wang Z., Liu S., "Progress of Grover Quantum Search Algorithm", Energy Procedia, vol. 16, pp. 1701 – 1706, 2012. https://doi.org/10.1016/j.egypro.2012.01.263
[14] Gressling, T. (2020). Data Science in chemistry: artificial intelligence, big data, chemometrics and quantum computing with jupyter. Walter de Gruyter GmbH & Co KG. http://dx.doi.org/10.1515/9783110629453
[15] Amellal, H., Meslouhi, A., & El Allati, A. (2018, October). Effectiveness of quantum algorithms on classical computing complexities. In Proceedings of the 3rd International Conference on Smart City Applications (pp.1-3). https://doi.org/10.1145/3286606.3286811
[16] Portugal, R. (2013). *Quantum walks and search algorithms (Vol. 19)*. New York: Springer. ISBN : 978-1-4614-6335-1.
[17] Dong, X., Dong, B., & Wang, X. (2020). Quantum attacks on some Feistel block ciphers. Designs, Codes and Cryptography, 88(6), 1179-1203. https://doi.org/10.1007/s10623-020-00741-y
[18] Zalka C., "Could Grover's Quantum Algorithm Help in Searching an Actual Database?", Quantum Physics, 1999, pp. 1-7. https://doi.org/10.1103/PhysRevA.62.052305
[19] Hahanov V., Miz V., "Quantum computing approach for shortest route finding", East-West Design & Test Symposium (EWDTS 2013), Rostovon-Don, Russia, 2013, pp. 27-30. https://doi.org/10.1109/EWDTS.2013.6673095
[20] Chen G., Fulling S. A., and Scully M. O., "Grover's Algorithm for Multiobject Search in Quantum Computing", Article in Lecture Notes in Physics, 1999, pp. 1-12. https://doi.org/10.1007/3-540-40894-0_15