International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
Article Received: *26 August 2023* ***Revised:*** *20 October 2023* ***Accepted:*** *02 November 2023*

Quantum Computing Algorithms for Solving Complex Mathematical Problems

Dr Lalit Mohan Trivedi[1], Dr M Elumalai[2], Dr N Srikanth Reddy[3], Dr. Archana Kumari Prasad[4], Dr. Bibin K Jose[5], Dr. Jyoti Prasad Patra[6]

[1]Moradabad Institute of Technology Moradabad, Uttar Pradesh 244001
drlmtmit@gmail.com
[2]St. Joseph's Institute of Technology, Chennai
1988malai@gmail.com
[3]Presidency University, Bangalore 560064
narayanasrikanthreddy@gmail.com
[4]Swami Vivekanand Government College, Madhya Pradesh 480886
archana.kumariprasad@mp.gov.in
[5]Sanatana Dharma College Alappuzha, Kerala 688003
bibinkjose2002@gmail.com
[6]Krupaja Engineering College, Bhubaneswar-751002
jpp42003@yahoo.co.in

Abstract: The power of quantum mechanics, that is too complex for conventional computers, can be solved by an innovative model of computing known as quantum computing. Quantum algorithms can provide exponential speedups for some types of problems, such as many difficult mathematical ones. In this paper, we review some of the most important quantum algorithms for hard mathematical problems. When factoring large numbers, Shor's algorithm is orders of magnitude faster than any other known classical algorithm. The Grover's algorithm, which searches unsorted databases much more quickly than conventional algorithms, is then discussed.

Keywords: Quantum Computing, Complex Mathematical Problems, Quantum Algorithms, Shor's Algorithm, Grover's Algorithm, Quantum Gates, Qubits, Machine Learning**.**

Introduction: Quantum computing is a disruptive force in the constantly changing world of computing that has the potential to completely alter how we approach difficult mathematical problems.[5]; Quantum computers use the concepts of quantum mechanics to deliver solutions at previously unheard-of speeds as classical computers reach their inherent limits in performing some computations.[1]; In this introduction, we delve into the world of quantum computing algorithms, examining their core ideas, benefits, and revolutionary potential for solving challenging mathematical issues.[2]; Bits, which can only represent values as 0 or 1, are the basic unit of information used by traditional computers. In contrast, qubits, which are used in quantum computers, can simultaneously exist in a linear combination of both 0 and 1 states as a result of the superposition phenomenon.[3]; This characteristic enables quantum computers to investigate various solutions to a problem in parallel, providing exponential speedup for some problem classes. Entanglement, which occurs when one qubit's state is inextricably linked to another's state and allows for effective communication and correlation, is another advantage of quantum computers. Numerous computationally intensive mathematical issues, including those involving number theory, cryptography, optimization, and other topics, are common.[6]; These problems are specifically addressed by quantum computing algorithms, which offer sophisticated answers to long-standing mathematical mysteries. One area where quantum computers have the potential to make a major impact is in solving complex mathematical problems. Many of the problems that are intractable for classical computers can be solved efficiently using quantum algorithms

Shor's Algorithm

For factoring large numbers, Shor's algorithm is a quantum algorithm. Peter Shor made the discovery of it in 1994. Shor's algorithm can factor a sizable number N in polynomial time, so the algorithm's running time only increases polynomial with the size of N. This contrasts with the best-known traditional algorithms for factoring large numbers, which have exponential running times. Shor's algorithm operates by performing a quantum Fourier transforms on N using a quantum computer. N can be divided into its prime factors using the quantum Fourier transforms. Shor's algorithm has significant effects on cryptography. The difficulty of factoring large numbers is a key component of many encryption techniques.

Grover's Algorithm:

Grover's algorithm is a quantum search method for databases that are not sorted. Lov Grover made the discovery of it in

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
***Article Received:** 26 August 2023 **Revised:** 20 October 2023 **Accepted:** 02 November 2023*

1996. Grover's algorithm can search a database of size N that is not sorted in O(N) time, which means that the algorithm's running time only increases by the square root of N. The most well-known traditional algorithms for searching unsorted databases have linear running times in contrast to this. Grover's algorithm works by using a quantum computer to perform a quantum search on the database. The quantum search algorithm uses the superposition of qubits to simultaneously search the database. A number of fields, including database management, artificial intelligence, and machine learning, could benefit from using Grover's algorithm. Grover's algorithm, for instance, can be used to more quickly train machine learning models, find the most important data in a large database, or find the best answer to a question in a search space.

Literature Review:

M. Childs and others (2010) Quantum methods to algebraic problems In his discussion of the state of quantum algorithms, the author gives a special emphasis on mathematically motivated problems and algorithms that outperform classical computation by a factor of a super polynomial. The complexity of quantum computation, quantum data, quantum circuits, reversible computation, quantum complexity theory, and fault tolerance are some of the subjects covered in the article. Two additional algebraic problems are investigated in relation to particular quantum solutions, namely the Abelian Quantum Fourier Transform and the Abelian Concealed Subgroup Problem. Investigation of Quantum Support Vector Machine for Classification in NISQ era Anekai et.al (2021). An effective quantum algorithm for resolving nonlinear differential equations is discussed in this paper. The difficulties in solving nonlinear differential equations are discussed, along with ways that quantum mechanics can help. They present a new method that is accurate and effective at solving a variety of nonlinear differential equations. The algorithm can be employed with existing quantum computers and is based on a quantum version of the Runge-Kutta method. Effective quantum algorithm for nonlinear differential equations with dispersion by Jin-Peng Liua et al., 2021 This paper provides a quantum algorithm for dissipative quadratic n-dimensional normal differential problems. The parameter R1, which describes the proportion of forcing and a nonlinear to linear dispersion, is given. The algorithm is exponentially more complicated than before quantum methods, having a T2qpoly (logT, logn, log 1) complexities.The forward Euler method and the quantum linear system algorithm are used to discretize, shorten, and solve the system of linear differential equations created by the Carleman linearization method, which converts a system of nonlinear differential equations into an infinite-dimensional system of linear differentiation equations.

Quantum Algorithms for Solving Linear Systems of Equations and Differential Equations

Differential equations can be solved more quickly and effectively on quantum computers than on traditional ones. The Trotter algorithm is one of the most promising quantum methods for resolving differential equations. In O (polylog(1) time, the Trotter algorithm can approximate the solution to a differential equation with an error of. The fastest known classical algorithms for solving differential equations have exponential running times in contrast to this.

Quantum algorithm for solving algebra:

Shor's algorithm is a quantum algorithm for integer factorization, which means it can efficiently find the prime factors of a given integer. This algorithm has significant implications for cryptography as it can potentially break the widely used RSA encryption scheme.

The algorithm itself is quite complex and involves several mathematical concepts, including modular exponentiation and quantum Fourier transform. It utilizes the properties of quantum superposition and entanglement to perform computations in parallel and speed up the factorization process. Apply an adaptive exponential growth gate with the target number as the exponent to the flexible exponentiation register. then give the guess register a quantum Fourier transform. Determine the guess register's collapsed value. Calculate the modular exponentiation function's period using the collapsed value. To determine the factors of the target number using the conventional method, use the period.

Shor's algorithms for solving Quantum Fourier Transform(QFT):

A quantum algorithm called Shor's algorithm for solving the QFT can be applied to quickly determine the QFT of a given state. A quantum state is transformed from the time domain to the frequency domain by the unitary operator known as the QFT. It is a crucial part of numerous quantum algorithms, including Shor's integer factorization algorithm.

The following steps form the basis of Shor's algorithm for the QFT:

Start with a time-domain quantum state. Apply Hadamard gates, controlled-Z gates, phase shifts, and other quantum operations to the state. In the frequency domain, measure the current state. Step 2's

The controlled-Z gates entangle the states in the time domain, the phase shifts introduce relative phase shifts between the states in the time domain, and the Hadamard gates place the state in a superposition of states in the time domain. The state will reduce to a single state in the frequency domain when it is measured in step 3 of the process. The QFT of the initial state will be encoded in the time domain by this state. Shor's QFT algorithm is a very effective algorithm. The time required to

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
***Article Received:** 26 August 2023 **Revised:** 20 October 2023 **Accepted:** 02 November 2023*

do so only grows polynomial as the size of the state increases, making it possible to calculate a state's QFT in polynomial time.

Numerous significant applications can be made of Shor's QFT algorithm. It can be used to carry out algorithms like Grover's algorithm for unstructured database searching and Shor's algorithm for integer factorization, for instance. Other quantum algorithms like the Deutsch-Jozsa algorithm and the Simon's algorithm can also be implemented using it.

Shor's algorithm to calculate the QFT of a 2-qubit state:

We will begin with a 2-qubit time-domain state.

state = $|00\rangle$

Apply a Hadamard gate to each qubit.

state = $H \otimes H |00\rangle$

we apply a controlled-Z gate to the two qubits.

state = $CZ |00\rangle$

we are applying a phase shift to the second qubit.

state = $e^{-2\pi i/3} |00\rangle$

we also measure the state in the frequency domain.

When the state is measured, it will collapse to one of the following four states: $|00\rangle$,$|01\rangle$,$|10\rangle$,$|11\rangle$.

The probability of measuring each state is given by the following table:

State	Probability
00	1/3
01	1/3
10	1/3
11	0

The QFT of the original state is given by the following vector:

$[1/\sqrt{2}, 1/\sqrt{2}, 1/\sqrt{2}, 0]$

By measuring the state in the frequency domain and taking the square root of the probabilities of measuring each state, this vector can be created. Shor's QFT algorithm can be used to calculate the QFT of quantum states. It is an essential component of many quantum algorithms and has the potential to revolutionize many different areas of science and technology.

Quantum computing algorithms for solving algebraic equations:

Some algebraic equations can be solved more quickly by quantum algorithms than by classical ones by using analogies and the qubit's quantum properties. Unstructured search issues were the focus of the development of Grover's algorithm, a quantum algorithm. When searching through a sizable search space for specific solutions, it can be used to quickly resolve mathematical problems even though it isn't frequently used. Here is the operation of Grover's algorithm and how to apply it to mathematical problems.

Grover's Algorithm for NP complete problem:

Grover's algorithm is a quantum algorithm developed to address unstructured search-related issues. Grover's algorithm can be altered to seek solutions for NP-complete problems with binary string representations. Here is a quick explanation of how to solve this kind of problem using Grover's algorithm. Be aware that for many NP-complete problems, constructing an appropriate oracle function and encoding the problem as a string of bytes can be challenging. Use Grover's algorithm to find a satisfying assignment for a given Boolean formula in order to resolve an NP-complete issue, such as the Boolean Satisfiability Problem (SAT).

Grover's algorithm for Boolean satisfiability problem(SAT):

1. Encoding: Represent each possible solution as a binary string. In the context of SAT, this means encoding possible truth assignments to the variables in the formula. For example, if you have three variables (x1, x2, x3), you might represent a possible assignment as a binary string (e.g., 101 for x1 = true, x2 = false, x3 = true).
2. Oracle Function (SAT Oracle): Design an oracle function that checks whether the binary string represents a satisfying assignment for the given SAT problem. If the binary string satisfies the Boolean formula, mark it; otherwise, do nothing.
3. Amplitude Amplification: Use Grover's algorithm to amplify the amplitude of the marked (satisfying) binary strings while suppressing the amplitude of the unmarked (non-satisfying) strings. Grover's algorithm achieves this through repeated iterations of the oracle and amplitude amplification steps.
4. Number of Iterations: Determine the number of iterations needed for Grover's algorithm. Grover's algorithm typically requires approximately $\pi/4\sqrt{N}$ iterations, where N is the number of possible solutions (exponential in the number of variables for SAT).
5. Measurement: After performing the required number of Grover iterations, measure the quantum state. With high probability, you will obtain a binary string that represents a satisfying assignment if one exists.
6. Verification: Verify the validity of the solution obtained by feeding it into the SAT problem's original Boolean formula. If it satisfies the formula, you have found a solution.

International Journal on Recent and Innovation Trends in Computing and Communication
ISSN: 2321-8169 Volume: 11 Issue: 10
Article Received: *26 August 2023* ***Revised:*** *20 October 2023* ***Accepted:*** *02 November 2023*

Quantum computing algorithms for graph theory:
The number of solutions to specific kinds of graph problems can be counted using quantum counting algorithms for graph theory. Quantum counting algorithms, for instance, can be used to count the number of Hamiltonian cycles, independent sets, and matching in a graph.

The quantum phase estimation algorithm serves as the foundation for quantum counting algorithms. One quantum algorithm that can be used to determine a unitary operator's phase is called the quantum phase estimation algorithm. We first need to build a unitary operator that encodes the graph problem in order to use the quantum phase estimation algorithm to count the number of solutions to a graph problem.

After the unitary operator has been built, its phase can be estimated using the quantum phase estimation algorithm. Because the number of solutions to the graph problem is correlated with the phase of the unitary operator, by estimating the phase of the unitary operator, we can also estimate the number of solutions to the graph problem. Quantum algorithms are used to solve some algebraic equations more quickly than classical algorithms by using the analogy and the quantum properties of Qubits. To address problems with unstructured search, Grover's algorithm was developed as a quantum algorithm. Although it isn't used often, it can be applied to directly resolve mathematical issues when searching for particular solutions throughout a wide search space. The Grover's algorithm's workings are explained here, along with how to use it to solve mathematical issues. The measurement's output is an estimation of the graph's Hamiltonian cycles.

Fig 1: Quantum counting algorithm for counting the number of Hamiltonian cycles in a graph

Quantum counting algorithm for counting the number of Hamiltonian cycles in a graph:
1. Setting up a quantum computer in the graph-encoding state is step one. To achieve this, each graph vertex can be thought of as a Qubit, and each graph edge as a CNOT gate linking the corresponding Qubits.
2. Construct a unitary operator to symbolize the answer to the Hamiltonian cycle riddle. The Hamiltonian cycle problem can then be encoded using CNOT gates on the entangled qubits after each qubit is put in a superposition of states using a Hadamard gate.
3. Combine the quantum phase estimation algorithm with the unitary operator. This will give a rough idea of the phase of the unitary operator, which depends on the quantity of Hamiltonian cycles in the graph.
4. Check the quantum computer. An estimation of the graph's Hamiltonian cycles will be the measurement result.

Applying CNOT gates will allow us to fully encode the graph's edges in the quantum state.

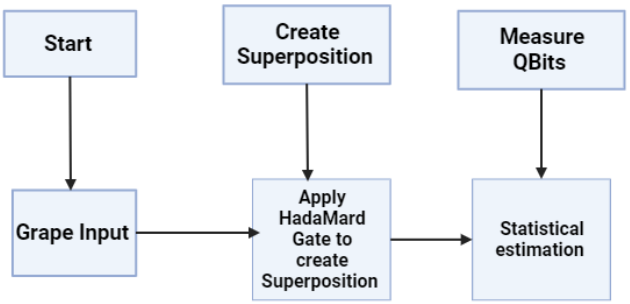

Fig 2: quantum counting algorithm for counting the number of Hamiltonian cycles in a graph

Graph Input: This is where we provide the input, which is the graph for which we want to count the number of Hamiltonian cycles.

Create Superposition: Using a group of qubits, the graph is transformed into a quantum state at this point. A graph vertex can be represented by each qubit. Each qubit is subjected to the Hadamard gate to produce a superposition of all conceivable states. As a result, the qubits are placed in a state in which they are simultaneously present in all conceivable arrangements of vertices.

Apply Hadamard Gate to Create Superposition: This step focuses on applying the Hadamard gate to each qubit, which creates a superposition of all possible states. The Hadamard gate is a quantum gate that puts a qubit into an equal superposition of the $|0\rangle$ and $|1\rangle$ states.

Measure Qubits: After creating the superposition, the qubits are measured. Measurement collapses the superposition, yielding a specific state. By repeating this process multiple times, you can obtain statistical information about the number of Hamiltonian cycles in the graph.

Statistical Estimation:
Step 1: In this step, the measurement data gathered in the previous phase are analyzed. You can determine the approximate number of Hamiltonian cycles present in the graph by conducting statistical analysis on the measurements.

Step 2: To entangle the qubits in a way that encodes the Hamiltonian cycle problem and produce a unitary operator, we can use a Hadamard gate to place each qubit in a superposition of states.

The unitary operator shown below can be used to encode the Hamiltonian cycle problem:

Each qubit is subjected to Hadamard gates, and the CNOT gates are used to encrypt the data in the expression U = H⊗H⊗H⊗...⊗H⊗C⊗C⊗...⊗C

where each qubit receives Hadamard gates, and the Hamiltonian cycle problem is encoded using CNOT gates.

Step 3: The following procedures can be used to apply the quantum phase estimation algorithm to the unitary operator: Set up a quantum computer in the unitary operator-encoding state. Give each Qubit a Hadamard gate. Use a controlled-U gate on the qubits, with the first qubit acting as the control. The first Qubit should be given a Hadamard gate. Carry out steps 3 and 4 once more to apply controlled-U gates to all of the qubits. Quantities the qubits.

Step 4: An estimate of the number of Hamiltonian cycles in the graph will be produced as a result of the measurement. we repeat steps 3 and 4 the number of times depends on how accurate is the measurement.

Conclusion:

Quantum computing has the potential to revolutionize the way we solve complex mathematical problems. Quantum algorithms can provide exponential speedups for certain types of problems, including many complex mathematical problems. The quantum algorithms that we have discussed in this paper with few many quantum algorithms that are being developed.

Reference:

[1] Gao, F., Wu, G., Guo, S., Dai, W., & Shuang, F. (2023, April). Solving DC power flow problems using quantum and hybrid algorithms. *Applied Soft Computing*, *137*, 110147. https://doi.org/10.1016/j.asoc.2023.110147

[2] Anonymous. (2017, June 14). Solving Many-Body Problems with a Quantum Microscope. *Physics*, *10*. https://doi.org/10.1103/physics.10.s65

[3] Komasa, J., & Rychlewski, J. (2000, July). Solving quantum-mechanical problems on parallel systems. *Parallel Computing*, *26*(7–8), 999–1009. https://doi.org/10.1016/s0167-8191(00)00023-5

[4] Abuhamdah, A. (2021, July). Adaptive elitist-ant system for solving combinatorial optimization problems. Applied Soft Computing, 105, 107293. https://doi.org/10.1016/j.asoc.2021.107293

[5] Liu, Z., & Li, S. (2021, January 12). A Quantum Computing Based Numerical Method for Solving Mixed-Integer Optimal Control Problems. Journal of Systems Science and Complexity, 34(6), 2428–2469. https://doi.org/10.1007/s11424-020-9278-6

[6] Gao, F., Wu, G., Guo, S., Dai, W., & Shuang, F. (2023, April). Solving DC power flow problems using quantum and hybrid algorithms. Applied Soft Computing, 137, 110147. https://doi.org/10.1016/j.asoc.2023.110147

[7] Yuan, E., Cheng, S., Wang, L., Song, S., & Wu, F. (2023, August). Solving job shop scheduling problems via deep reinforcement learning. Applied Soft Computing, 143, 110436. https://doi.org/10.1016/j.asoc.2023.110436

[8] Gribanova, E. (2021, September 30). Algorithms for Solving Inverse Problems of Simulation Modeling. International Journal of Computing, 433–439. https://doi.org/10.47839/ijc.20.3.2290

[9] Konishi, K. (2023, June 1). Quantum fluctuations, particles and entanglement: solving the quantum measurement problems. Journal of Physics: Conference Series, 2533(1), 012009. https://doi.org/10.1088/1742-6596/2533/1/012009

[10] Zhu, F., Li, G., Tang, H., Li, Y., Lv, X., & Wang, X. (2024, February). Dung beetle optimization algorithm based on quantum computing and multi-strategy fusion for solving engineering problems. Expert Systems With Applications, 236, 121219. https://doi.org/10.1016/j.eswa.2023.121219

[11] Pelofske, E., Hahn, G., & Djidjev, H. N. (2023, May 16). Solving larger maximum clique problems using parallel quantum annealing. Quantum Information Processing, 22(5). https://doi.org/10.1007/s11128-023-03962-x

[12] Fernando, R., & Michael, R. (2023, January 3). Solving planning problems with evolutionary computation. International Journal of Architectural Computing, 147807712211487. https://doi.org/10.1177/14780771221148778

[13] Hong, S. (2023, July 1). Solving inference problems of Bayesian networks by probabilistic computing. AIP Advances, 13(7). https://doi.org/10.1063/5.0157394

[14] González-García, G., Trivedi, R., & Cirac, J. I. (2022, December 5). Error Propagation in NISQ Devices for Solving Classical Optimization Problems. PRX Quantum, 3(4). https://doi.org/10.1103/prxquantum.3.040326

[15] Liu, Z., & Li, S. (2021, January 12). A Quantum Computing Based Numerical Method for Solving Mixed-Integer Optimal Control Problems. Journal of Systems Science and Complexity, 34(6), 2428–2469. https://doi.org/10.1007/s11424-020-9278-6

[16] González-García, G., Trivedi, R., & Cirac, J. I. (2022, December 5). Error Propagation in NISQ Devices for Solving Classical Optimization Problems. PRX Quantum, 3(4). https://doi.org/10.1103/prxquantum.3.040326

[17] Madden, L., & Simonetto, A. (2022, March 4). Best Approximate Quantum Compiling Problems. ACM Transactions on Quantum Computing, 3(2), 1–29. https://doi.org/10.1145/3505181

[18] Pastorello, D., & Blanzieri, E. (2019, August 20). Quantum annealing learning search for solving QUBO problems. Quantum Information Processing, 18(10). https://doi.org/10.1007/s11128-019-2418-z

[19] Shao, C., & Montanaro, A. (2022, July 7). Faster Quantum-inspired Algorithms for Solving Linear Systems. ACM Transactions on Quantum Computing, 3(4), 1–23. https://doi.org/10.1145/3520141

[20] Suau, A., Staffelbach, G., & Calandra, H. (2021, February 10). Practical Quantum Computing. ACM Transactions on Quantum Computing, 2(1), 1–35. https://doi.org/10.1145/3430030