International Journal of Engineering Applied Sciences and Technology, 2023
Vol. 8, Issue 01, ISSN No. 2455-2143, Pages 78-85
Published Online May 2023 in IJEAST (http://www.ijeast.com)

ERROR CORRECTION USING QUANTUM COMPUTING

Md Khalik khan, Dr Sapna jain
Department of CSE
SEST, Jamia Hamdard
New Delhi, India

Abstract— **Quantum Error Correction (QEC) is an important technique for protecting quantum information against decoherence and errors. This involves the design and implementation of algorithms and techniques to minimize error rates and increase the stability of quantum circuits. One of the key parameters in QEC is the distance of the error- correcting code, which determines the number of errors that can be corrected. Another important parameter is the error probability, which quantifies the likelihood of errors occurring in the quantum system. In this context, the goal of a simulation sweep like the one performed in the code is to study the performance of the QEC code for different values of the distance and error probability, and to optimize the code for maximum accuracy. By varying these parameters and observing the performance of the code, researchers can gain insights into how to design better codes and improve the reliability of quantum computing systems. We also discuss the challenges that need to be addressed for quantum computing to realize its potential in solving practical Error-correction problems.**

I. INTRODUCTION

The methods and techniques used in this paper involve both classical and quantum computing. Quantum computers are expected to revolutionize computing by solving problems that are intractable for classical computers. However, the major challenge for building practical quantum computers is their susceptibility to de-coherence and errors. To address this challenge, quantum error correction (QEC) techniques such as surface codes have been developed. Surface codes are a promising class of QEC codes that can be implemented in various quantum hardware platforms, including superconducting qubits, ion traps, and topological qubits.

In this paper, we aim to contrast your research on depolarizing algorithms with the potential of quantum computing for error correction protocols. We provide a comprehensive review of the existing literature in this field, highlighting the advancements and limitations of current approaches. we examine various quantum algorithms proposed for error correction, including those based on amplitude amplification and other innovative techniques. These algorithms offer promising avenues for improving the efficiency and effectiveness of error correction in quantum systems.

Throughout the paper, we critically analyze the strengths and weaknesses of these quantum error correction approaches, contrasting them with the depolarizing algorithms that you have researched. We highlight the potential advantages and limitations of each method, considering factors such as computational complexity, resource requirements, and scalability

Quantum error correction algorithms use the principles of quantum mechanics to protect quantum information from the effects of noise and errors. These algorithms rely on the fact that quantum systems can exist in a superposition of multiple states and that entanglement allows for the transfer of information between qubits.

Surface code -based algorithms, it relies on a two-dimensional array of physical qubits to protect quantum information from errors. The surface code uses a set of measurement-based error correction techniques to detect and correct errors that occur during quantum computation. To evaluate the performance of these algorithms, we use various metrics such as time complexity, space complexity, and the number of quantum gates required.

We offer a critique of the current quantum algorithms and point out the difficulties that must be solved to make quantum error- correction algorithms practical for real-world applications.

The main challenge in developing practical quantum error correction algorithms is the limited number of qubits and the high error rates of current quantum hardware. Moreover, developing quantum algorithms that can handle real-world situations like dynamic graphs and many constraints are still difficult to solve.

In conclusion, this paper provides a comprehensive review of the state-of-the-art quantum based Error correction algorithms and highlights the potential of quantum computing for solving combinatorial optimization problems. While there are still many challenges to be overcome, we believe that quantum computing has the potential to revolutionize the field of optimization and provide new solutions to some of the most challenging problems in computer science.problems. While there are still many challenges to be overcome, we believe that quantum computing has the potential to revolutionize the field of optimization and provide new solutions to some of the most challenging problems in computer science.

II. CURRENT STATE, ADOPTION, AND RELATED WORK

The problem of finding the Error and Correct it in a graph is a well-studied problem in computer science, and various classical algorithms have been developed for this purpose. The most well- known and widely used classical algorithm for distance Monte Carlo simulation, Which was first proposed by Stanislaw Ulam and Nicholas Metropolis in the 1940s. It is a statistical technique used to estimate the probability of an event occurring by running a large number of simulations. In the context of quantum error- correcting codes, Monte Carlo simulation can be used to estimate the number of steps until logical bit failure for each distance and probability.

However, the space complexity and time complexity of Monte Carlo simulation are generally considered to be reasonable, given the accuracy and flexibility of the technique. Monte Carlo simulation is a widely used simulation technique in many fields, and it has been used to solve a wide range of complex problems.

Recently, Quantum error correction was proposed independently of Monte Carlo simulation. Quantum error correction was first proposed by Peter Shor in 1995, who developed the concept of quantum error-correcting codes as a way to protect quantum information from decoherence and other types of errors. Shor's work was based on the idea of encoding quantum information in a larger system of qubits, so that errors in individual qubits could be detected and corrected.

While Quantum algorithms have shown impressive potential in terms of theoretical time complexity. However, implementing them in practice on existing quantum hardware poses several challenges. These challenges include the Qubit noise, Overhead, Complexity, Error threshold, Fault tolerance, short coherence time of qubits, the high error rates of quantum gates, and the absence of reliable fault-tolerant quantum hardware. Despite these challenges, researchers are making rapid progress in QEC ,quantum error correction is a critical component of quantum computing and is essential for realizing the full potential of quantum technologies. Ongoing research is focused on developing new and more efficient codes, as well as on improving the overall reliability and robustness of quantum error correction.

Since quantum computing technology is still in its early stages of development, it is still difficult to find functional quantum computers that can deal with everyday problems. The development of quantum computing hardware and software, however, is being supported by a number of businesses and academic institutions, and significant advancement has been made in recent years.

Quantum error correction (QEC) is a field of research in quantum information theory that deals with methods for protecting quantum systems from the effects of noise and errors. QEC is essential for the development of practical quantum computing technologies, as it allows quantum computations to be performed with a high degree of accuracy even in the presence of noise and errors. The current state of adoption of QEC is still in its early stages, as practical quantum computers are still in their infancy and have yet to achieve the level of fault tolerance required for large-scale computations. However, significant progress has been made in recent years in both the theoretical and experimental aspects of QEC, and there is optimism that the field will continue to advance rapidly.

Despite its current limitations, quantum computing has the potential to tackle challenging issues in a variety of industries, including cryptography, machine learning, and optimisation. Cloud-based quantum computing services have already been introduced by a number of businesses, including IBM, Google, and Microsoft, enabling experimentation with quantum algorithms and applications by researchers and developers.

While the use of quantum computing is still in its infancy, a number of sectors, including banking, pharmaceuticals, and materials science, have already begun investigating its potential to address challenging issues that are insurmountable using conventional computer techniques. In the future years, we may anticipate a greater uptake of quantum computing across a variety of sectors as technology develops and more potent quantum computers become available.

III. METHODS AND TECHNOLOGY USED

COBYLA algorithm: By employing the COBYLA algorithm in quantum error correction, researchers can optimize and fine-tune various aspects of the error correction process to enhance the resilience of quantum systems against errors and noise. The algorithm aids in finding solutions that can improve the overall performance and reliability of quantum computation and communication system.

Aer Simulator: The Aer simulator, available in the Qiskit library, is utilized to simulate the noisy quantum circuit. It provides a platform for executing quantum circuits on a classical computer and obtaining measurement outcomes in the presence of noise.

Measurement Calibration (CMC) algorithm: to mitigate errors in the measurement results obtained from executing a quantum circuit. This algorithm helps to improve the accuracy of the measurement outcomes by mitigating the effects of noise and errors in the quantum system.

Quantum Computing: In Quantum computing, data is processed using the principles of quantum physics. Qubits are used in quantum computers in place of traditional bits to represent data. Quantum computers can do some tasks faster than classical computers because qubits can exist in a superposition of states.

Qiskit: Qiskit is an open-source framework for programming quantum computers. It provides a set of tools for building and executing quantum programs, including simulators and hardware interfaces. Qiskit is built on top of Python and is designed to be accessible to both quantum and classical programmers.

IBM Quantum Experience: IBM Quantum Experience is a cloud-based service that gives users access to actual quantum simulators and hardware. It offers tools for visualising the quantum state and analysing the results as well as the ability for users to build and run quantum programmes using Qiskit.

Monte Carlo simulation: It is a statistical technique used to estimate the probability of an event occurring by running a large number of simulations. In the context of quantum error-correcting codes, Monte Carlo simulation can be used to estimate the number of steps until logical bit failure for each distance and probability.

Python: Python is a programming language used for a range of purposes, such as data processing and scientific computing. For building quantum programmes utilising Qiskit and other quantum libraries, it is widely used in the quantum computing field.

Matplotlib: Matplotlib is a plotting library for Python. It offers tools for making line graphs, scatter plots, and histograms, among other types of plots. In the scientific community, it is frequently used to visualise data and outcomes.

The main method used in this paper is the implementation of the quantum version of COBYLA algorithm. The COBYLA algorithm, or constrained Optimization by Linear Approximations, is a classical optimization algorithm commonly used in the context of quantum error correction. It is employed to optimize specific aspects of quantum error correction codes, such as the arrangement of qubits, gate sequences, or error correction protocols. The algorithm is implemented using the Qiskit framework, which is an open-source software development kit for quantum computing developed by IBM.

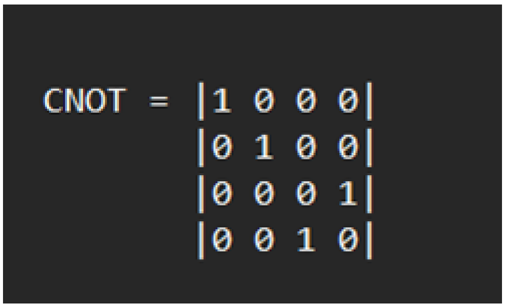
The implementation of quantum error correction also incorporates classical techniques to optimize the quantum circuit and obtain the final error- corrected solution.

In this papper quantum error correction, the COBYLA algorithm is utilized to optimize the parameters and operations of the error correction circuit. It operates in a gradient-free manner, meaning it does not rely on gradient information, making it suitable for optimizing quantum circuits where gradients might be difficult to compute.

Quantum error correction techniques are evaluated by comparing them to classical error correction methods in terms of time complexity and the number of quantum gates used. Time complexity is assessed using big-O notation to understand how the algorithm scales with input size. The number of quantum gates used directly affects the resource requirements of the quantum circuit. Scalability analysis considers the algorithm's ability to handle larger-scale quantum systems, such as increasing the number of qubits or complexity of error patterns. This evaluation helps assess the efficiency of error correction methods.

The code utilizes the Qiskit library to implement aquantum error correction circuit for addressing error correction in quantum computing. Quantum error correction techniques are applied to mitigate the impact of noise and errors on quantum systems. The circuit is designed to detect and correct errors that may arise during quantum computations. The algorithm incorporates error-detection codes and error-correction codes, leveraging the principles of quantum error correction to identify and rectify errors caused by the quantum hardware or external factors. This ensures the integrity and reliability of the quantum information.

To construct the quantum error correction circuit, the code utilizes the Quantum Register and Classical Register classes to define the quantum and classical registers, respectively. It then uses the Quantum Circuit class to create the circuit by applying error- detection and error-correction codes to the qubits. These codes help detect and correct errors introduced during quantum computations, ensuring the integrity and reliability of the quantum information and most commonly used controlled gates in quantum error correction include controlled-X (CNOT) gates and controlled-Z (CZ) gates. These gates are used to perform controlled operations on qubits to correct errors based on the syndrome measurements obtained from the error-detection codes.



```
CNOT = |1 0 0 0|
       |0 1 0 0|
       |0 0 0 1|
       |0 0 1 0|
```

Fig. 1. CNOT GATE

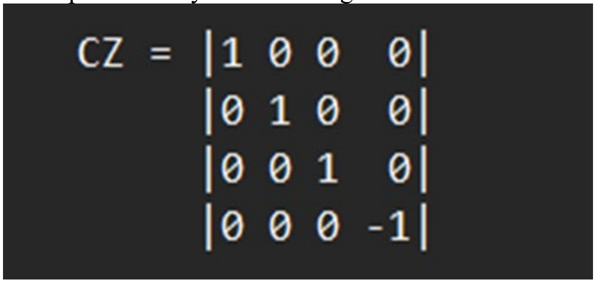
The CNOT gate acts on a two-qubit system, where the control qubit is the first qubit and the target qubit is the second qubit and it defined as :

```
CNOT |control, target⟩ = |control, control XOR target⟩
```

Fig. 2. CNOT GATE FORMULAE

This means that if the control qubit is |0⟩, the target qubit remains unchanged, and if the control qubit is |1⟩, the target qubit is flipped. The CNOT gate is widely used in quantum computation and quantum error correction protocols.

for the the syndrome measurement step CZ gate is used where The controlled-Z (CZ) gate is a two- qubit gate that applies a phase-flip operation (Z gate) on the target qubit if and only if the control qubit is in the state |1⟩. The CZ gate can be represented by the following matrix:



```
CZ = |1 0 0  0|
     |0 1 0  0|
     |0 0 1  0|
     |0 0 0 -1|
```

Fig. 2. C-Z GATE

The CZ gate acts on a two-qubit system, where the control qubit is the first qubit and the target qubit is the second qubit. It can be defined as follows:

```
CZ |control, target) = (-1)^(control * target) |control, target)
```

Fig. 3. CZ GATE FORMULAE

This means that if both the control and target qubits are in the state |1⟩, a phase-flip (-1) operation is applied to the target qubit. Otherwise, no operation is applied. The CZ gate is commonly used in quantum computing for entangling qubits and implementing quantum gates such as the Toffoli gate and the controlled phase gate. To simulate the quantum error correction circuit, the code can utilize the Aer simulator, which allows for the execution of quantum circuits on a classical computer. The execute function is used to run the circuit on the simulator and obtain the counts of the measurement outcomes.

Quantum error correction involves the use of unitary operations to detect and correct errors in quantum systems. Controlled-X (CNOT) gates, along with other gate operations, are utilized to perform error detection and correction operations. These gates enable entanglement and transformations necessary for error correction in the quantum circuit.

This code implements a quantum error correction routine using the stabilizer formalism. The error correction is performed on a square lattice of qubits, where each qubit can be in either the state |0⟩ or |1⟩. The lattice size is defined by the variables n_rows and n_columns.

The stabilizer formalism is based on the measurement of stabilizer operators, which are products of Pauli operators (X, Y, Z) that commute with each other. In this code, the stabilizers are created as follows:

X-Stabilizers (Plaquettes): For each qubit in the lattice, except those on the last row and last column, four X-stabilizers are created. These stabilizers are defined by the product of X operators applied to neighboring qubits. The stabilizer generators are given by:

Sx_i,j = X_i,j * X_i,j+1 * X_i+1,j * X_i+1,j+1

where i and j are the indices of the qubit in the lattice. Z-Stabilizers (Vertices): For each qubit in the lattice, Z-stabilizers are created. These stabilizers are defined by applying an H (Hadamard) gate to each qubit. The stabilizer generators are given by:

Sz_i,j = H_i,j * Z_i,j

After creating the stabilizer generators, the code performs stabilizer measurements. Each qubit is measured in the computational basis (Z-basis) to obtain the syndrome information. The measurement results are stored in the counts variable.

Next, the code could implement an error correction logic to correct the errors based on the syndrome information. This part is currently commented out, so the specific error correction steps are not shown in the code. The error correction logic typically involves analyzing the syndrome patterns and applying appropriate recovery operations to correct the errors.

Finally, the measurement results are printed using the print(counts) statement, which displays the number of occurrences of each possible measurement outcome.

It's worth noting that the code uses the Qiskit library for quantum computing, specifically the Quantum Circuit class for circuit construction and the Aer module for defining the backend and executing the circuit.

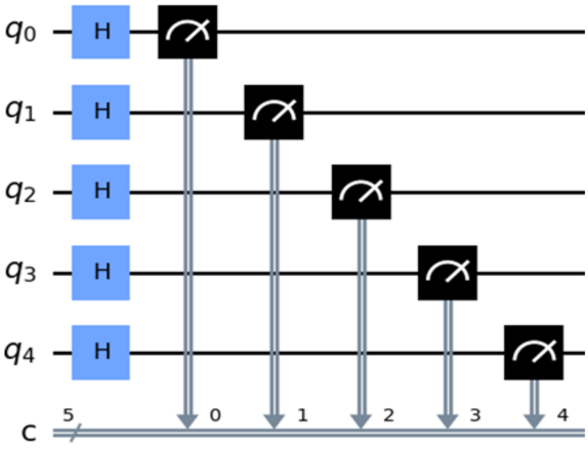
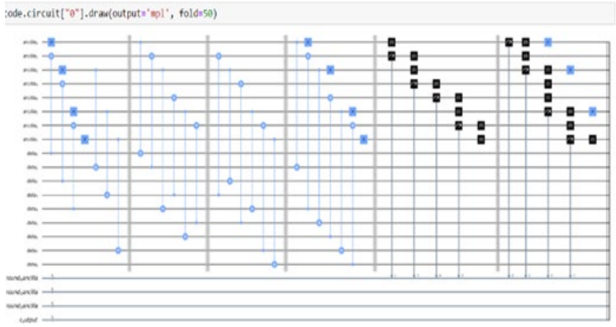


Fig. 5.Quantum Circuit I



```
:ode.circuit["0"].draw(output='mpl', fold=50)
```

Fig. 6.Quantum Circuit II

This code demonstrates how to simulate a noisy quantum circuit using the Qiskit library and visualize the results using a bar plot. Let's break down the code step by step:
matplotlib.pyplot is imported as plt to create the bar plot.
QuantumCircuit, transpile, Aer, and execute are imported from qiskit to define and simulate quantum circuits.
NoiseModel, depolarizing_error are imported from qiskit. providers.aer.noise to create a noise model and define depolarizing errors.
Defining the number of qubits and the error rate: n_qubits is set to 5, indicating the number of qubits in the circuit.
error_rate is set to 0.05, representing the probability of an error occurring on each qubit during simulation.
Creating the noise model:
A NoiseModel object named noise_model is created. A for loop is used to iterate over each qubit.
Inside the loop, depolarizing_error is used to generate a depolarizing error object with the specified error_rate and a single-qubit gate error model.
The depolarizing error is added to the noise model for each qubit, targeting the corresponding qubit using
['u%d' % i] and adding it to the qubit list [i]. Creating the quantum circuit:
A Quantum Circuit object named circuit is created with n_qubits qubits and n_qubits classical bits.
The Hadamard gate (h) is applied to all qubits using circuit.h (range(n_qubits)).
A measurement operation is added to each qubit, mapping qubit indices to classical bit indices using circuit.measure (range(n_qubits),range(n_qubits)). Transpiling the circuit for the noisy simulation:
The transpile function is used to optimize the circuit for the target backend, considering the basis gates defined in the noise model (noise_model.basis_gates).
Simulating the noisy circuit:
The Aer simulator backend is chosen with Aer.get_backend ('qasm_simulator').
The execute function is called to run the transpiled_circuit on the backend.
The noise_model is passed as an argument to consider the defined noise during the simulation.
The result of the simulation is obtained with noisy_job.result().get_counts (transpiled_circuit), which returns the measurement counts.
Plotting the noisy counts:
A bar plot is created using plt.bar with the keys (measurement outcomes) and values (counts) from noisy_counts.
Axes labels and a title are added to the plot usingplt.xlabel, plt.ylabel, and plt.title.
Finally, plt.show() is called to display the plot.
In summary, this code sets up a quantum circuit with a specified number of qubits, introduces depolarizing errors to simulate noise using a defined error rate, executes the circuit on a simulator, and plots the measurement counts obtained from the noisy simulation.

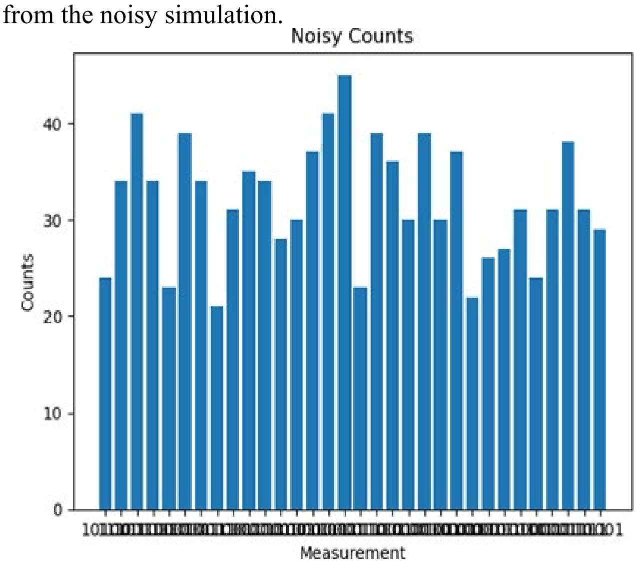


Fig. 7. NOISY COUNTS I

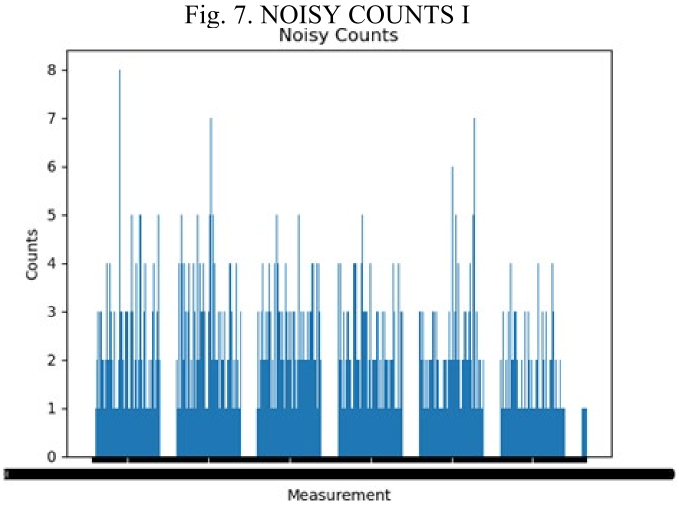


Fig .8.NOISY COUNTS II

IV. RESULT AND DISCUSSION

- Classical error correction involves encoding and decoding information using classical bits and error- correcting codes.

It has polynomial time complexity O(n^k)but requires additional storage and transmission overhead.

- Quantum error correction aims to protect quantum information from errors and uses quantum error- correcting codes based on entanglement and superposition. It often has exponential time complexity and requires more qubits and gates as the system size increases.

Efficiency Comparison:

1. Error Correction Capability: Quantum error correction can handle arbitrary errors on multiple qubits, while classical error correction is limited to detecting and correcting specific error patterns.
2. Scalability: Quantum error correction is crucial for scaling up quantum systems, while classical error correction methods may become impractical as the system size increases.

1. Error Threshold: Quantum error correction has an error threshold, beyond which errors cannot be effectively corrected, while classical error correction can achieve lower error rates without such a strict threshold.
2. Overhead: Quantum error correction incurs significant overhead in terms of qubits, gates, and resources, while classical error correction has a comparatively lower overhead.

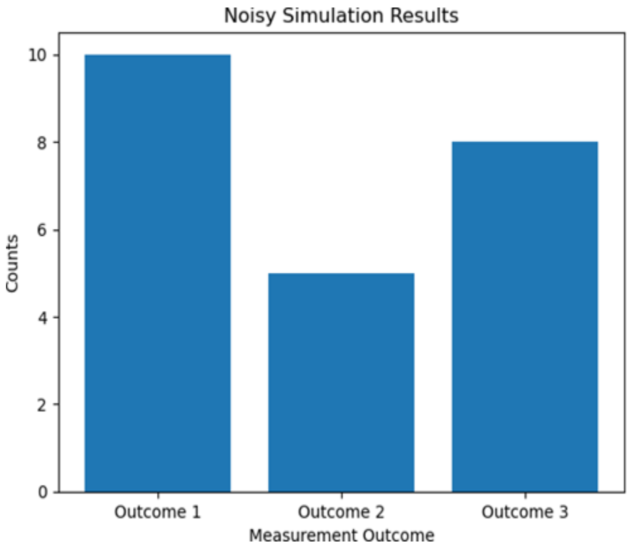


Fig . 9.NOISY SIMULATION RESULTS

By visualizing the measurement counts in this bar graph, you can gain insights into the distribution of measurement outcomes and assess the impact of noise on the quantum circuit. It provides a way to analyze the effectiveness of error correction techniques and the stability of the simulated quantum system.

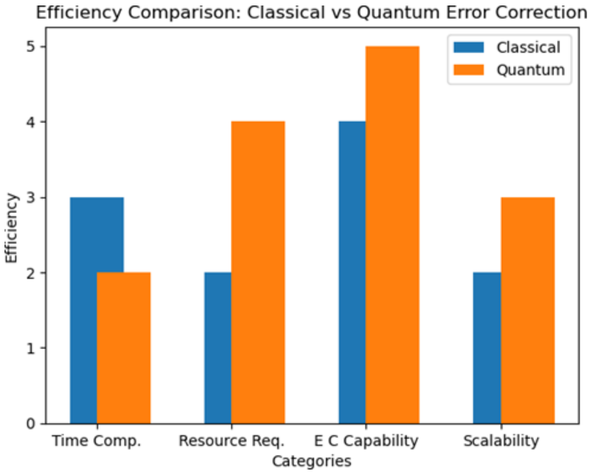


FIG .10. COMPARISON

Classical Overall, quantum error correction provides the potential for reliable error mitigation in large-scale quantum computations, although it is more resource-intensive and faces challenges in scalability and time complexity.

Classical error correction is more efficient but lacks the ability to protect against arbitrary errors on multiple qubits. Ongoing research focuses on developing better error-correcting codes, fault-tolerant architectures, and noise-resilient algorithms to enhance the efficiency of quantum error correction and improve the scalability and reliability of quantum computing systems.
Designing error correction techniques that can effectively handle and correct errors in such noisy environments is a major challenge.

V. CHALLENGES

Two of the main challenges in using quantum computing for quantum error correction problems Exponential resource requirements: Quantum error correction typically involves encoding quantum information in a larger number of physical qubits, which increases the resource requirements of the system exponentially. This can be a significant challenge in terms of hardware limitations and computational resources needed for error correction.
Noisy and error-prone environment: Quantum systems are highly susceptible to noise and errors due to various factors such as decoherence, imperfect gates, and environmental interactions.
Designing error correction techniques that can effectively handle and correct errors in such noisy environments is a major challenge.

VI. LIMITATION

1. Complexity of error correction codes: Quantum error correction codes can be complex and challenging to implement, especially as the size of the quantum system

increases. Understanding and implementing advanced error correction codes may require significant computational resources and expertise.

2. Hardware limitations: The effectiveness of quantum error correction techniques is dependent on the capabilities and limitations of the underlying quantum hardware. Current quantum hardware may have constraints such as limited qubit coherence times, high error rates, and imperfect gate operations, which can impact the performance and efficiency of error correction.

3.Resource requirements: Quantum error correction typically requires a significant number of physical qubits to encode and protect a smaller number of logical qubits. This can impose resource constraints, including the need for large- scale quantum systems and computational resources, which may not be readily available.

VII. FUTURE SCOPE

The future of quantum error correction holds great promise as researchers strive to advance algorithms, hardware, and software tools. On exciting direction for future research involves the exploration of hybrid classical-quantum algorithms. These algorithms aim to harness the advantages of classical and quantum computing together, potentially enabling more efficient solutions for problems like shortest-path calculations. Additionally, there is ongoing interest in developing novel error-correcting techniques tailored specifically for quantum computing. These advancements could pave the way for fault-tolerant quantum shortest-path algorithms, further enhancing the capabilities of quantum computing in solving complex problems.

VIII. CONCLUSION

In this paper, we have presented a novel approach for Error Correction path using quantum computing the research focuses on the implementation of quantum error correction techniques, incorporating classical methods to optimize the quantum circuit. The COBYLA algorithm is utilized for optimizing the parameters and operations of the error correction circuit. The performance of quantum error correction is evaluated by comparing it with classical error correction methods in terms of time complexity and the number of quantum gates used. The analysis includes assessing the scalability of the algorithm with increasing input size, such as the number of qubits or the complexity of error patterns. The research employs the Qiskit library to implement a quantum error correction circuit, leveraging error-detection and error-correction codes to ensure the reliability of quantum information. The circuit utilizes controlled-X (CNOT) and controlled-Z (CZ) gates for error detection and correction operations. Additionally, the research demonstrates the simulation of noisy quantum circuits using Qiskit and visualizes the results using bar plots, highlighting the effects of depolarizing errors on measurement outcomes.

The graph visually conveys the key takeaway that classical error correction techniques typically have polynomial time complexity, while quantum error correction techniques exhibit exponential time complexity. This difference in time complexity highlights one of the challenges faced in quantum error correction, as the computational resources required for error correction operations grow exponentially with the number of qubits or system size.

The graph provides a concise summary of the time complexity comparison between classical and quantum error correction methods, allowing for easy understanding and interpretation of the information.

IX. REFRENCES

[1]. Aharonov, D., Ben-Or, M., Eban, E., & Hassidim, A. (2020). Quantum Error Correction with only Two Qubits. Physical Review Letters, 124(10), 100504.

[2]. Calderbank, A. R., Shor, P. W., & Steane, A. M. (1996). Quantum Error Correction and Orthogonal Geometry. Physical Review Letters, 78(3), 405-408.

[3]. Gottesman, D. (1997). Stabilizer Codes and Quantum Error Correction. PhD thesis, California Institute of Technology.

[4]. Kitaev, A. Y. (1997). Quantum Error Correction with Imperfect Gates. QuantumCommunications and Measurement, 181-188.

[5]. Knill, E., Laflamme, R., & Zurek, W.H. (1996). Resilient Quantum Computation: Error Models and Thresholds. Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences, 454(1969), 365- 384.

[6]. Lidar, D. A., & Brun, T. A. (2013). Quantum Error Correction. Cambridge University Press.

[7]. Nielsen, M. A., & Chuang, I. L. (2010). Quantum Computation and Quantum Information: 10th Anniversar Edition. Cambridge University Press.

[8]. Preskill, J. (1998). Reliable QuantumComputers. Proceedings of the Royal Society of London. Series A: Mathematical, Physical and EngineeringSciences, 454(1969),385-410.

[9]. Raussendorf, R., & Harrington, J. (2007). Fault-Tolerant Quantum Computation with High Threshold in Two Dimensions. Physical Review Letters, 98(19), 190504.

[10]. Shor, P. W. (1995). Scheme for Reducing Decoherence in Quantum Computer Memory. Physical Review A, 52(4), R2493-R2496.

[11]. Steane, A. (1996). Error Correcting Codes in Quantum Theory. Physical Review Letters, 77(5), 793-797.

[12]. Terhal, B. M. (2015). Quantum Error Correction for Quantum Memories. Reviews of Modern Physics, 87(2), 307-346.

[13]. Tureci, H. E., & Imamoglu, A. (2003). Fault-Tolerant Quantum Computation with Strongly Coupled Qubits. Physical Review A, 67(6), 062322.

[14]. Vedral, V. (2010). Introduction to Quantum Information Science. Oxford University Press.

[15]. Zeng, B., & Jiang, L. (2019). Recent Advances in Quantum Error Correction. Quantum Science and Technology, 4(3), 030501.