Error Analysis of 3-qubit Standard and Modified Grover's Algorithms through Noisy Channels

Tyler King

Cornell University
ttk22@cornell.edu

July 12, 2022

Abstract

Grover's algorithm, a quantum alternative to classical search algorithms, provides a quadratic speedup to time complexity, running in $O\sqrt{N}$ compared to the classical alternative's $O(N)$. This quadratic speed up allows for computational advantages on the same hardware.

We start by introducing the math underpinning Grover's algorithm and the quantum circuit necessary to obtain a $|110\rangle$ state with highest probability. Our implementations of Grover's algorithm utilize a single MCT gate along with the "compute-copy-uncompute" method (chained Toffolis with an AND gate). Both scenarios are tested on quantum hardware (*ibmq_lima*) along with substantial simulated noisy results (each trial using one million shots on the *qasm_simulator*). A set of noisy results results are compared with a t-test to check which technique is more effective along with confirming whether the two techniques are statistically distinct.

1 Introduction

Quantum computing is a relatively new field that allows for the speed up of classical algorithms by utilizing quantum mechanics such as superposition and entanglement [9]. A well-known example is Grover's algorithm, a quantum algorithm for unstructured search that runs with $O(\sqrt{N})$ evaluations, which is quadratically faster than classical computing alternatives that need $O(N)$ evaluations [7]. This algorithm is known to be asymptotically optimal for Quantum search algorithms [3]. In this paper, we look to analyze the error of Grover's algorithm for multi-qubit states through noisy channels using IBM's quantum simulators when applying Grover rotations in two distinct ways [5, 6]. While these two different styles have identical theoretical performances, their differences become more apparent when running on hardware and viewing the transpiler code [5].

Since classical search algorithms must go through and individually check every possibility, on average they need $N/2$ cases and N cases in the worst scenario, giving them the linear time complexity showcased earlier [10]. However, Grover's Algorithm operates under very different principles. The system is initialized with a uniform superposition existing across all states [7]. Then, $r(N)$ Grover iterations are applied, where each iteration consists of an oracle and a Grover diffusion operator [5, 7, 8]. These two processes repeatedly "rotate" the superposition vector towards the target vector (which is the state that is being searched for) [12]. This algorithm requires $\approx \frac{\pi}{4}\sqrt{N}$ iterations ($O(\sqrt{N})$ runtime), which is quadratically faster than the classical alternative [3, 7, 12].

Both the standard approach with Grover's Algorithm along with a simple modification for multi-controlled z-gates (MCZ gates) are tested using three qubits with IBM's software API known as Qiskit (quantum information software kit) [1, 12]. This enables us to run Grover's algorithm on IBM's hardware while also giving us access to simulators (*ibmq_lima_*simulator) that note expected probability distributions of the qubit states.

We simulate error for single- and multi-qubit processes using Qiskit's noise feature, comparing the results of Grover's algorithm across these channels to errors in IBM's hardware [1]. These computations allow us to model the relationship between noise and accuracy, giving future targets for qubit errors to achieve accurate results on quantum circuits such as Grover's algorithm. A matched pairs t-test is then applied to test for statistically significant differences between Grover's algorithm and the proposed modification [13].

Structure of the paper. The rest of the paper is organized as follows. Section 2 introduces and discusses Grover's algorithm in a geometric and algebraic sense. Section 3 introduces the implementation of standard Grover's algorithm using an MCT gate. Section 4 introduces the implementation of modified Grover's algorithm using the "compute-copy-uncompute" method. Sections 5 and 6 introduce noise and the expected results for errors along single- and multi-qubit operations. Section 7 is the conclusion.

2 Grover's Algorithm

Grover's search algorithm is comprised of four key stages: initialization, oracle, amplification, and measurement [5]:
Initialization stage: creates a superposition of all states
Oracle Stage: The solution vector is marked by flipping the amplitude of the target state
Amplification stage: The target state's amplitude is magnified by reflecting the vector across the mean
Measurement stage: All qubits are measured and the final state is recorded

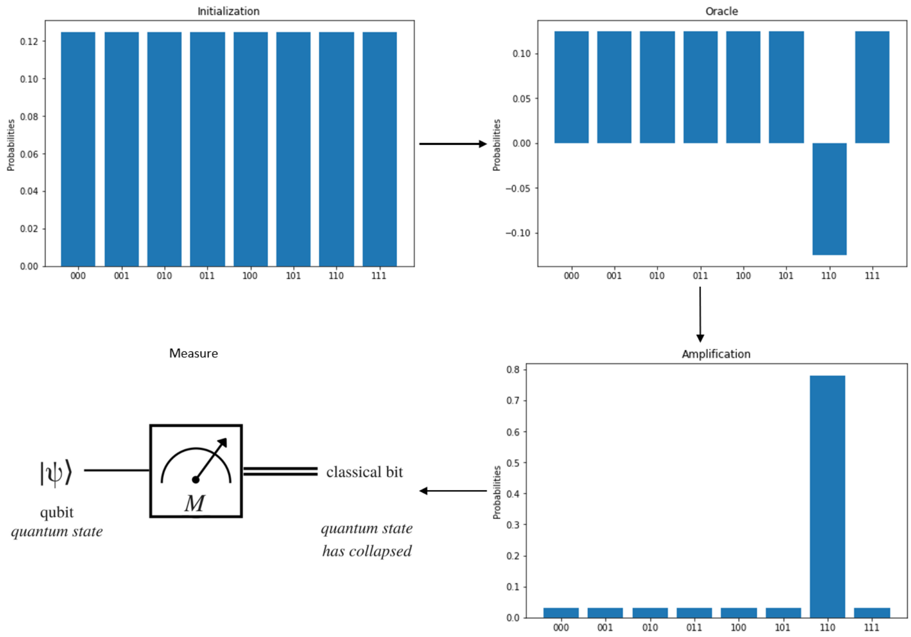


Figure 1: Probabilities of states during Grover's algorithm in all 4 stages on a 3-qubit quantum computer [4, 11]. Note that the Oracle and Amplification stages are repeated $\approx \frac{\pi}{4}\sqrt{N}$ times.

Remark 2.1. *The probability of measuring $|110\rangle$ in the oracle stage is not negative, but the amplitude is negated to make it easier to visualize. The probability of measuring $|110\rangle$is still $|-12.5\%| = 12.5\%$.*

The algorithm is initialized by creating a superposition state, denoted by the expression $H^{\otimes n}$ (which represents applying Hadamard gates to all n qubits. Then, the oracle stage is applied. This can formally be rewritten as a unitary matrix U_ω where

$$U_\omega |x\rangle = \begin{cases} |\phi\rangle & \text{if } \phi = \omega \\ -|\phi\rangle & \text{if } \phi \neq \omega \end{cases}$$

Note that this means for our 3-qubit case where our target state $\omega = 110$,

$$U_\omega = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

U_ω can be rewritten as

$$U_\omega = \begin{bmatrix} 1&0&0&0&0&0&0&0\\ 0&1&0&0&0&0&0&0\\ 0&0&1&0&0&0&0&0\\ 0&0&0&1&0&0&0&0\\ 0&0&0&0&1&0&0&0\\ 0&0&0&0&0&1&0&0\\ 0&0&0&0&0&0&1&0\\ 0&0&0&0&0&0&0&1 \end{bmatrix} - \begin{bmatrix} 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&2&0\\ 0&0&0&0&0&0&0&0 \end{bmatrix} = I - 2 \underbrace{|110\rangle\langle 110|}_{\text{projection operator}}$$

More generally, for some target state $|a\rangle$, $U_\omega = I - 2|a\rangle\langle a|$. However, measuring at this point would be useless because, while the coefficient of the $|110\rangle$ state is negative, the amplitude is still the same. This means that, when measuring, the probability of achieving any state is still $\frac{1}{2^n} = \frac{1}{2^3} = \frac{1}{8}$. This is why we apply amplitude amplification, significantly amplifying the amplitude of the target state while shrinking the amplitudes of other states. This process is achieved by reflecting all states over the mean [14]. To do this for some state $|s\rangle$, we preserve the components of the states along $|s\rangle$ and negate the components of the states orthogonal to $|s\rangle$. Formally, the unitary matrix U_s (also known as the Grover diffusion operator) represents this step, where $U_s = |s\rangle\langle s| - I$.

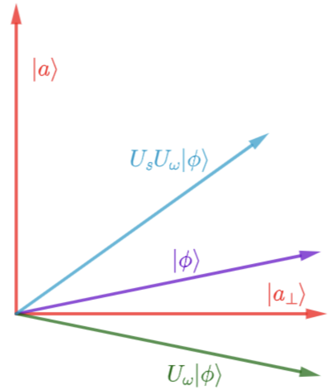


Figure 2: Geometric interpretation of Grover's algorithm. U_ω is applying an oracle while U_s is amplifying the marked state $U_\omega|\phi\rangle$.

The transformation U_sU_ω rotates the initial state $|\phi\rangle$ towards the target state $|a\rangle$. After t steps our new vector becomes $|\phi_t\rangle = (U_sU_\omega)^t|\phi\rangle$ for some initial superposition state $|\phi\rangle$.

2.1 Accuracy

After t iterations on n qubits, Grover's algorithm has a

$$t \cdot \left(\left[\frac{2^n - 2t}{2^n} + \frac{2(2^n - t)}{2^n} \right] \frac{1}{\sqrt{2^n}} \right)^2$$

probability of measuring the correct state [4]. This means that one iteration on our 3-qubit system will measure the correct state roughly 78.125% of the time:

$$1 \cdot \left(\left[\frac{2^3 - 2}{2^3} + \frac{2(2^3-1)}{2^3} \right] \frac{1}{\sqrt{2^3}} \right)^2$$

$$\left(\left[\frac{6}{8} + \frac{14}{8} \right] \frac{1}{\sqrt{8}} \right)^2$$

$$\left(\frac{20}{8\sqrt{8}} \right)^2$$

$$\frac{400}{512} = .78125 = 78.125\%.$$

2.2 Time Complexity

Recall that the angle between two vectors is $\cos\theta = \frac{u \cdot v}{\|u\|\|v\|}$. This means that, for an initial state with 0 iterations of Grover's algorithm,

$$\cos\theta_0 = \frac{|\phi\rangle \cdot |a_\perp\rangle}{\| \, |\phi\rangle \, \| \| \, |a_\perp\rangle \, \|}.$$

Note that since $|a_\perp\rangle$ has 1 state as a 0, it will look something like:

$$|a_\perp\rangle = \frac{1}{\sqrt{N-1}} \begin{bmatrix} 1 \\ \vdots \\ 1 \\ 0 \\ 1 \end{bmatrix}$$

where $N = 2^n$. Since

$$|\phi\rangle = \frac{1}{\sqrt{N}} \begin{bmatrix} 1 \\ \vdots \\ 1 \\ 1 \end{bmatrix}$$

$|\phi\rangle \cdot |a_\perp\rangle = \frac{N-1}{\sqrt{N}\sqrt{N-1}}$. Furthermore, since both vectors are normalized, we know that $\| \, |\phi\rangle \, \| = \| \, |a_\perp\rangle \, \| = 1$. Thus,

$$\cos\theta_0 = \frac{\frac{N-1}{\sqrt{N}\sqrt{N-1}}}{1 \cdot 1}$$

$$\cos\theta_0 = \frac{\sqrt{N-1}}{\sqrt{N}}.$$

To create a simpler formula to compute the angle, we apply the trigonometric identity $\sin^2\theta + \cos^2\theta = 1$.

$$\sin^2\theta_0 + \cos^2\theta_0 = 1$$

$$\sin^2\theta_0 + \left(\frac{\sqrt{N-1}}{\sqrt{N}} \right)^2 = 1$$

$$\begin{aligned}\sin^2\theta_0 &= 1 - \frac{N-1}{N}\\ \sin^2\theta_0 &= \frac{1}{N}\\ \sin\theta_0 &= \frac{1}{\sqrt{N}}.\end{aligned}$$

After t iterations, the angle between $|\phi_t\rangle$ and $|a_\perp\rangle$ can be represented by $\theta_t = (2t+1)\theta_0$. This is because each U_ω operation negates θ_t and each U_s operation reflects $|\phi_t\rangle$ across $|\phi_0\rangle$ (where the angle between them is $\theta_t + \theta_0$), which results in a final angle of $2(\theta_t + \theta_0) - \theta_t = \theta_t + 2\theta_0$. Therefore, $\theta_{t+1} = \theta_t + 2\theta_0$, or $\theta_{t+1} - \theta_t = 2\theta_0$. Writing this out as a telescoping series (up to the t-th term) yields:

$$\begin{aligned}\theta_t - \cancel{\theta_{t-1}} &= 2\theta_0\\ \cancel{\theta_{t-1}} - \cancel{\theta_{t-2}} &= 2\theta_0\\ &\vdots\\ \cancel{\theta_2} - \cancel{\theta_1} &= 2\theta_0\\ \cancel{\theta_1} - \theta_0 &= 2\theta_0.\end{aligned}$$

Which, when summing up the left- and right-hand sides, yields $\theta_t - \theta_0 = (2t)\theta_0$ or $\theta_t = (2t+1)\theta_0$. Since we are attempting to measure $|\phi_t\rangle$ when it is $\approx |a\rangle$, we are looking for $\theta_t \approx \frac{\pi}{2}$. Because $\sin\theta_0 = \frac{1}{\sqrt{N}} \sim \theta_0$, the number of iterations necessary is

$$\begin{aligned}(2t+1)\theta_0 &\approx \frac{\pi}{2}\\ (2t+1)\frac{1}{\sqrt{N}} &\approx \frac{\pi}{2}\\ t &\approx \frac{\pi}{4}\sqrt{N}.\end{aligned}$$

Since the $\frac{\pi}{4}$ is irrelevant for sufficiently large values of N, the runtime can be simply written as $O(\sqrt{N})$.

3 Standard Implementation: Grover's Algorithm

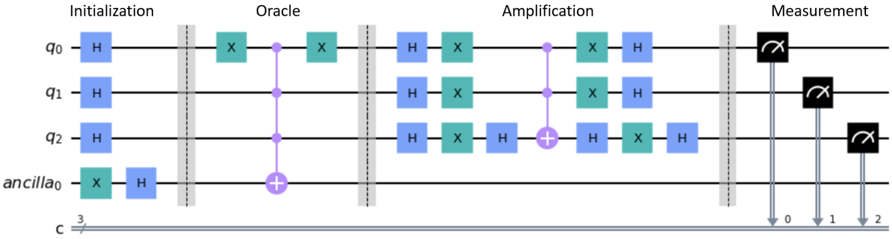


Figure 3: Stages of Grover's algorithm for the standard implementation on a 3-qubit quantum computer.

Remark 3.1. *Note that the extra NOT and Hadamard gates in the initialization stage on the ancilla$_0$ qubit exists only to change the final target state to 110. This is to make sure that incorrect quantum circuits would not be measured at 000.*

We begin by creating a superposition state in the initialization stage, and then proceed to "mark" (i.e. negate) our target state in the oracle stage. This is done with the NOT gates both before and after the MCT gate, flipping the state of the q_0 qubit. Since the final state reads the qubits backwards (i.e. $q_2q_1q_0$), the state that is marked by the oracle is 110.

From our earlier definition of the amplification stage, the unitary operator $U_s = |s\rangle \langle s| - I$. This can be redefined as

$$2(H^{\otimes n} |0\rangle)(\langle 0| H^{\otimes n}) - H^{\otimes n} H^{\otimes n},$$

where $H^{\otimes n}$ is the Hadamard gate being applied to all qubits. Factoring this expression yields:

$$H^{\otimes n} \left(2 |0\rangle \langle 0| - I\right) H^{\otimes n}.$$

Note that U_s and $-U_s$ are equally effective since both unitary matrices rotate the initial state towards a state orthogonal to $|a_\perp\rangle$.

$$-H^{\otimes n} \left(2 |0\rangle \langle 0| - I\right) H^{\otimes n} = H^{\otimes n} \left(I - 2 |0\rangle \langle 0|\right) H^{\otimes n}$$

which, for our $n = 3$ qubit case, can be represented by the matrix:

$$H^{\otimes n} \begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} H^{\otimes n}$$

Note that this matrix is representative of an MCZ gate with 3 qubits. To create an MCZ gate, we utilize Hadamard gates before and after an MCT gate, allowing for a functional equivalent of MCZ to be implemented on Qiskit's software interface.

3.1 Simulator

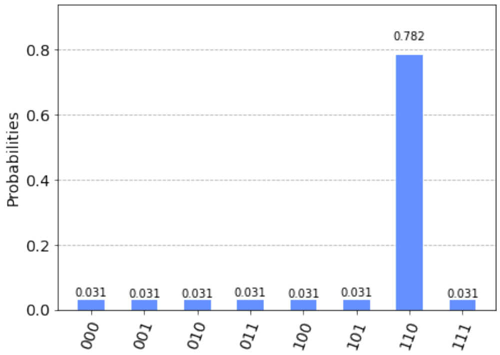


Figure 4: Simulated results using *qasm_simulator* of Grover's algorithm on a 3-qubit computer with one million trials.

As expected, the probability of measuring the correct state on the simulator is roughly 78.125%, which was the percentage proven earlier in Section 2.1. Furthermore, since all seven other states experienced the same amplitude magnifications in both the oracle and amplification stages, they must share the same amplitude. Thus, their expected amplitudes are $\frac{1-.78125}{7} = 0.03125$, which is roughly what was observed from the simulated results.

3.2 Hardware

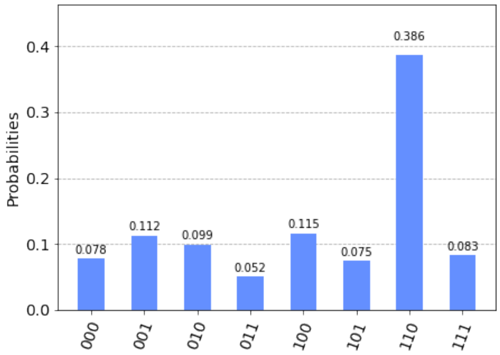


Figure 5: Results on the *ibmq_lima* for standard Grover's algorithm on a 3-qubit computer with one million trials.

After running on the *ibmq_lima*, it is clear that the error rates on qubit operations greatly inhibit Grover's algorithm from measuring the correct state roughly 78.125% of the time. When looking at the transpiler code (the code that is run on IBM's backend) it is clear why.

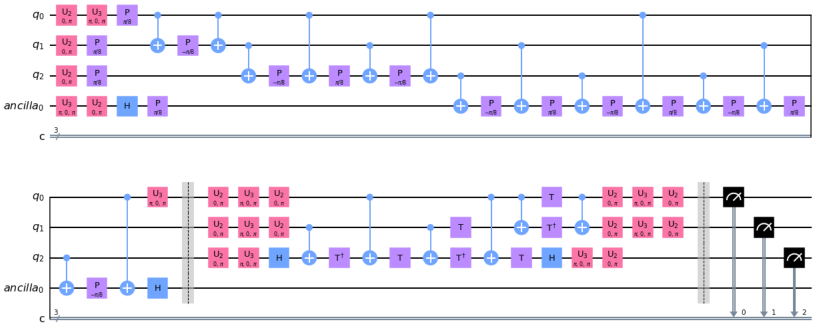
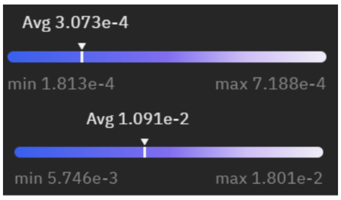


Figure 6: Transpiler code for Grover's algorithm on the *ibmq_lima*.

The high number of single- and multi-qubit gates stacks up, resulting in a greater likelihood of an error occurring that propagates throughout the circuit. This is furthered by the need for SWAP gates (which greatly increase the count of the higher-error multi-qubit gates) to allow certain operations between qubits to occur on the *ibmq_lima*.



Avg 3.073e-4

min 1.813e-4 max 7.188e-4

Avg 1.091e-2

min 5.746e-3 max 1.801e-2

Figure 7: Error rates for single- and multi-qubit gates respectively on the *ibmq_lima*.

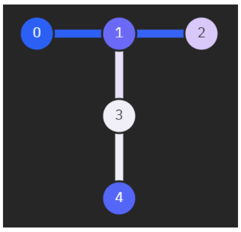


Figure 8: Orientation of qubits on the *ibmq_lima*.

4 Modified Implementation: Grover's Algorithm

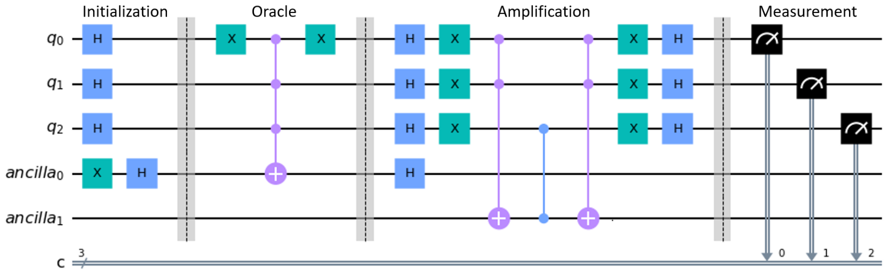


Figure 9: Stages of modified Grover's algorithm for the standard implementation on a 3-qubit quantum computer.

Note that the oracle stays the same in the modified version along with the Hadamard gates and NOT gates on both sides of the amplification stage. Thus, we only attempt to modify the implementation of the MCZ gate. To do so, we apply the "compute-copy-uncompute" method proposed by Charlie Bennett [2].

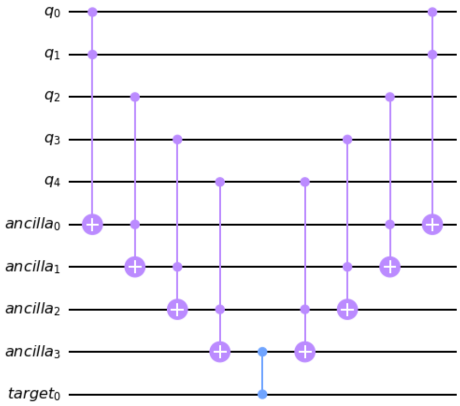


Figure 10: Example of "compute-copy-uncompute" method with 5 control qubits and 1 target qubit [12].

Remark 4.1. *The "compute-copy-uncompute" method can be applied to a multi-controlled version of any single qubit operation (Pauli gates, Hadamard gates, etc). In our case, we chose to use an MCZ gate because it is relevant to the amplification stage of Grover's algorithm.*

Recall that the goal of multi-controlled gates is taking the AND of all control qubits and applying the gate to the target qubit if the AND is 1. By using a cascading chain of Toffolis, we are able to take the AND of all control qubits. The result of the first Toffoli is stored in the first ancilla, and the AND of that result and the next control qubit is stored in another ancilla. This process continues until the target qubit is reached. After the single qubit operation is applied, the control and ancilla qubits must be returned to their initial state, resulting in the uncomputation of the earlier cascading Toffoli gates.

For our case with only two control qubits and one target qubit, the MCZ gate can be written as:

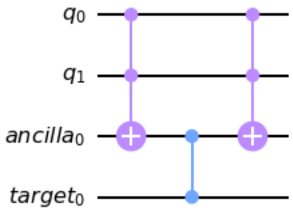


Figure 11: "compute-copy-uncompute" method with 2 control qubits and 1 target qubit. Note that for the circuit defined at the beginning of the section, $target_0 = q_2$.

This substitution allows for a new circuit to be developed that still applies Grover's algorithm.

4.1 Simulator

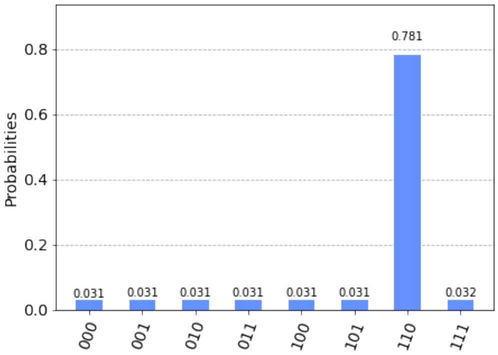


Figure 12: Simulated results using *qasm_simulator* of modified Grover's algorithm on a 3-qubit computer with one million trials.

Similar to the results obtained from standard Grover's algorithm, the percentages obtained in the simulated results of the modified Grover's algorithm are roughly .78125 for the correct state and .03125 for each of the incorrect states.

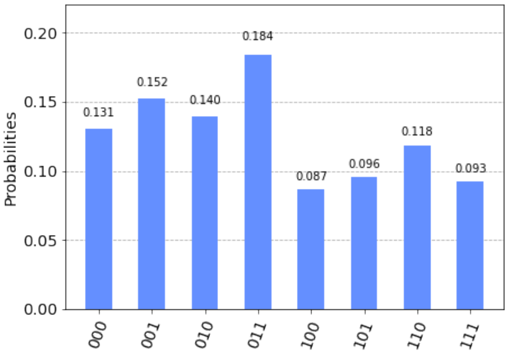


Figure 13: Results on the *ibmq_lima* for modified Grover's algorithm on a 3-qubit computer with one million trials.

4.2 Hardware

Due to a much higher count of gates on the transpiler code for this algorithm (in particular, there are 13 more CNOT gates, which tend to have higher error rates and thus impede accuracy more than any single-qubit operations), the probability of measuring the correct state is much lower since the chance of an error occurring that propagates throughout the circuit is higher.

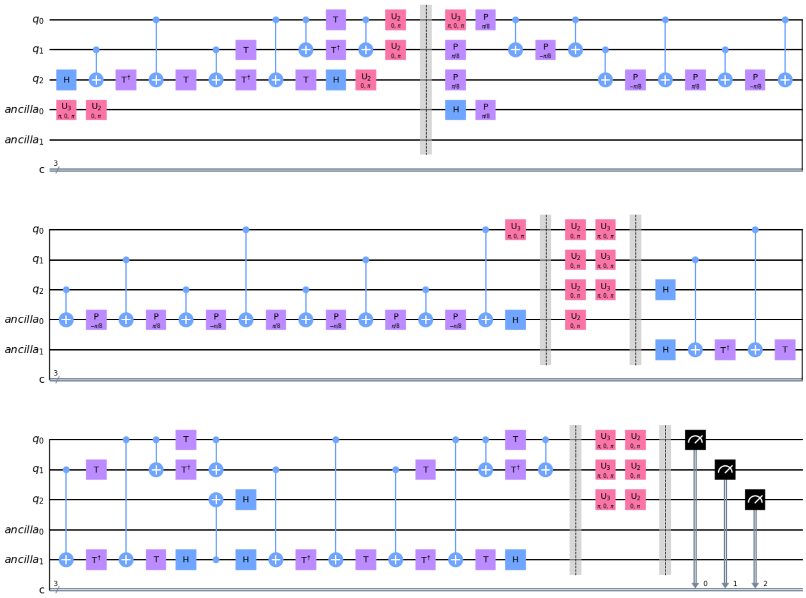


Figure 14: Transpiler code for Grover's algorithm on the *ibmq_lima*.

It is also worth noting that, due to the high number of single- and multi-qubit gates, the quantum state may decompose before it reaches the measurement stage, resulting in further inaccuracies.

5 Noise

Utilizing Qiskit's Noise Models, it becomes possible to simulate the expected accuracy of Grover's algorithm given certain error rates for single- and multi-qubit operations.

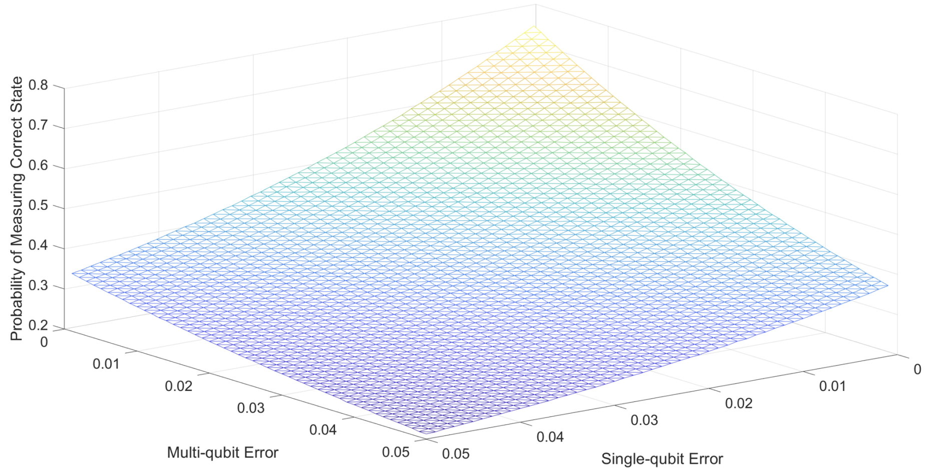


Figure 15: Probability of measuring the correct state of standard Grover's algorithm given error rates for single- and multi-qubit operations ranging from .001 to .05 by .001 increments.

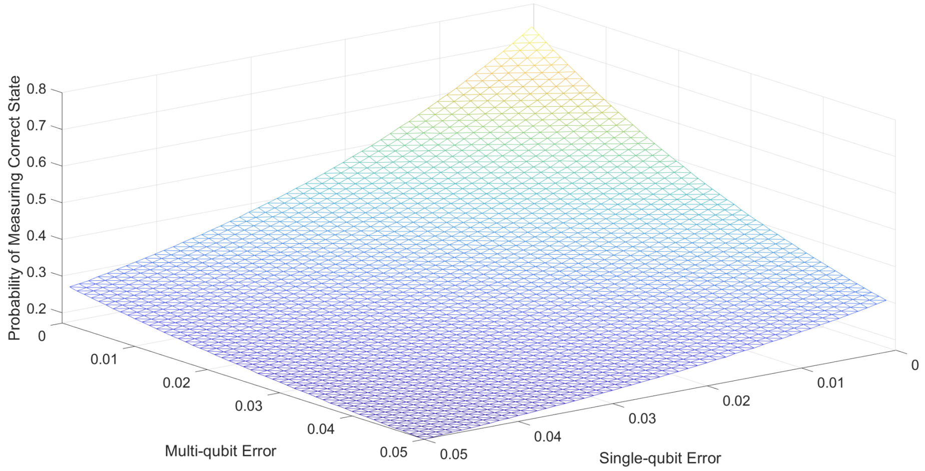


Figure 16: Probability of measuring the correct state of modified Grover's algorithm given error rates for single- and multi-qubit operations ranging from .001 to .05 by .001 increments.

Note that both graphs are slightly higher on the right side, indicating that multi-qubit errors are more likely to lower the measurement of the correct state. To observe which algorithm was more effective, we create a residual plot.

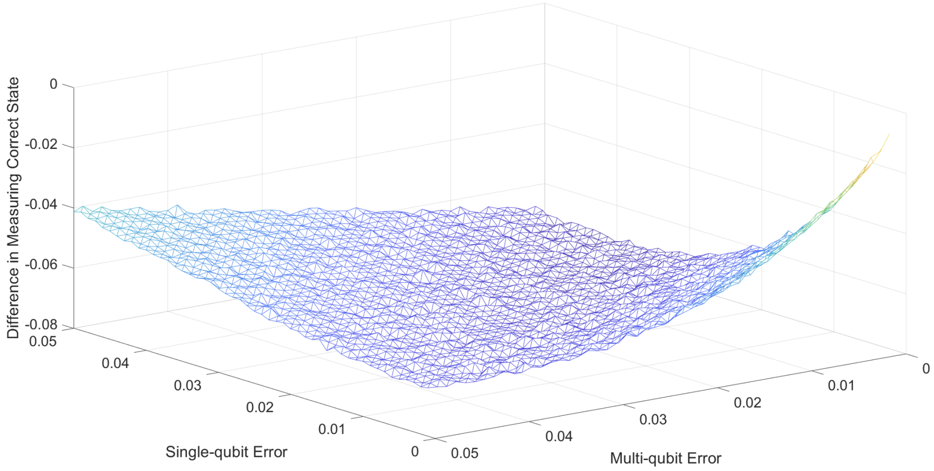


Figure 17: Residual Plot obtained from Figures 15 and 16. Probabilities obtained from standard Grover's algorithm were subtracted from probabilities obtained from modified Grover's algorithm. Note that the figure is rotated 90°.

Since the standard Grover's algorithm outperforms the modified Grover's algorithm in all 2500 cases, it is fair to state that it tends to be a more accurate algorithm (at obtaining the correct target state). To quantify this, we conduct a statistical analysis using a paired t-test.

Remark 5.1. *The difference between the two algorithms is minimal with small errors on both single- and multi-qubit operations because the accuracy is already extremely high. With mid-tier single- and multi-qubit errors, the difference is the largest because the transpiler code for the modified Grover's algorithm has significantly more gates. Finally, with higher single- and multi-qubit errors, the difference between the two algorithms again decreases because the accuracy of measuring the correct state is greatly reduced in both algorithms.*

6 Statistical Analysis

p_s: Population of probability of successful state measurement on the standard Grover's algorithm on a 3-qubit system with noise ranging from .001 to .05 (.001 increments) for both single- and multi- qubit errors
p_m: Population of probability of successful state measurement on the modified Grover's algorithm on a 3-qubit system with noise ranging from .001 to .05 (.001 increments) for both single- and multi- qubit errors
p_r: The residual plot of these two data sets where $p_r = p_m - p_s$
Null Hypothesis $H_0 : p_r = 0$
Alternative Hypothesis $H_a : p_r \neq 0$
Since p_r has a standard deviation of 0.008326 and a mean of -0.06006, our t-test yields

$$t = \frac{-.06006 - 0}{\frac{.008326}{\sqrt{2500}}}$$

$$t = -364.28.$$

Note that this t-test result yields an absurdly low P-value ($< .0000001$), indicating that there are extreme statistical differences between the two different graphs.

Remark 6.1. *While standard Grover's algorithm outperforms the modified Grover's algorithm for lower-qubit cases, we postulate that an increase in the number of qubits may result in the two algorithms performing similarly. This is because the MCT gate in standard Grover's algorithm will scale roughly equivalently to the modified MCZ gate.*

Remark 6.2. *The initial conditions to apply a t-test were not met, but we still conducted a paired t-test to note the statistical differences between the two sets of points from the two distinct Grover's algorithms.*

7 Conclusion

In this work, we introduce the concept of Grover's algorithm with both a geometric and linear algebra interpretation. We then implement Grover's algorithm 2 different ways on IBM's quantum hardware along with simulating expected results through noisy channels that modify single- and multi-qubit error. We show that the "compute-copy-uncompute" method is less effective than applying an MCT for the $n = 3$ qubit case by a significant margin for error rates between .001 and .05, even though their simulated results are roughly the same (when compiled on the *ibmq_lima_*simulator). Utilizing a paired t-test, we obtain a t-value of -364.28, indicating statistically significant differences between the two Grover's algorithms.

Since we only worked with 3-qubit systems, it is possible that the results we observed in this paper differ on higher-qubit systems, especially with the scalability of the "compute-copy-uncompute" method compared to utilizing higher-order MCTs. A further extension would be to utilize quantum random walks as an alternative to replace the amplification stage (Grover's diffusion operator) [8].

References

[1] Héctor Abraham et al. Qiskit: An open-source framework for quantum computing, 2021. doi:10.5281/zenodo.2573505.

[2] C. H. Bennett. Logical reversibility of computation. *IBM Journal of Research and Development*, 17(6):525–532, 1973. doi:10.1147/rd.176.0525.

[3] Charles H. Bennett, Ethan Bernstein, Gilles Brassard, and Umesh Vazirani. Strengths and weaknesses of quantum computing. *SIAM Journal on Computing*, 26(5):1510–1523, 1997. arXiv:https://doi.org/10.1137/S0097539796300933, doi:10.1137/S0097539796300933.

[4] Michel Boyer, Gilles Brassard, Peter Høyer, and Alain Tapp. Tight bounds on quantum searching. *Fortschritte der Physik*, 46(4-5):493–505, 1998. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/%28SICI%291521-3978%28199806%2946%3A4/5%3C493%3A%3AAID-PROP493%3E3.0.CO%3B2-P, arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/\%28SICI\%291521-3978\%28199806\%2946\%3A4/5\%3C493\%3A\%3AAID-PROP493\%3E3.0.CO\%3B2-P, doi:https://doi.org/10.1002/(SICI)1521-3978(199806)46:4/5<493::AID-PROP493>3.0.CO;2-P.

[5] C. Figgatt, D. Maslov, K. A. Landsman, N. M. Linke, S. Debnath, and C. Monroe. Complete 3-qubit grover search on a programmable quantum computer. *Nature Communications*, 8(1), Dec 2017. URL: http://dx.doi.org/10.1038/s41467-017-01904-7, doi:10.1038/s41467-017-01904-7.

[6] Konstantinos Georgopoulos, Clive Emary, and Paolo Zuliani. Modelling and simulating the noisy behaviour of near-term quantum computers, 2021. arXiv:2101.02109.

[7] Lov K. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing*, STOC '96, page 212–219, New York, NY, USA, 1996. Association for Computing Machinery. doi:10.1145/237814.237866.

[8] Anna Houk. Quantum random walk search and grover's algorithm - an introduction and neutral-atom approach. 06 2020.

[9] Abhijith J. et al. Quantum algorithm implementations for beginners, 2020. arXiv:1804.03719.

[10] Donald E. Knuth. *The Art of Computer Programming, Vol. 1: Fundamental Algorithms.* Addison-Wesley, Reading, Mass., third edition, 1997.

[11] Nimish Mishra. *Understanding the basics of measurements in Quantum Computation.* April 2019. URL: https://towardsdatascience.com/understanding-basics-of-measurements-in-quantum-computation-4c885879eba0.

[12] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition.* Cambridge University Press, USA, 10th edition, 2011.

[13] Randall Schumacker and Sara Tomek. *Understanding Statistics Using R*. Springer New York, 2013. doi:10.1007/978-1-4614-6227-9.

[14] Frank Zickert. *A Practical Guide To Quantum Amplitude Amplification.* April 2021. URL: https://towardsdatascience.com/a-practical-guide-to-quantum-amplitude-amplification-dbcbe467044a.