Enhanced Purification Protocol for Bell States

Matthew Lewis Worthington

A senior thesis submitted to the faculty of
Brigham Young University
in partial fulfillment of the requirements for the degree of

Bachelor of Science

Jean-Francois Van Huele, Advisor

Department of Physics and Astronomy

Brigham Young University

ABSTRACT

Enhanced Purification Protocol for Bell States

Matthew Lewis Worthington
Department of Physics and Astronomy, BYU
Bachelor of Science

Due to the unpredictable and delicate nature of quantum states, quantum systems need processes to remove error and purify corrupted quantum bits. Purification protocols can reduce error and prevent corrupted states from multiplying in such systems.The primary errors come from corruption of the state whether through phase transitions or bit flips. By accounting for the various types of errors, different methods can be used to detect errors and increase the trustworthiness of a system.

By using a bit-flip check to find X and Y errors and a phase-flip check to detect Z and Y errors, we can ascertain if an error has occurred in the entangled pair. By using these two methods in tandem, it is possible to create a purification protocol that can detect the majority of errors that result in corruption of a system while minimizing the use of quantum resources. We will discuss the fidelity gain in entanglement qubits by using the purification protocol.

Keywords: purification protocol, quantum systems, quantum computation, error correction, quantum states,bit-flip, phase-flip, fidelity

ACKNOWLEDGMENTS

We acknowledge support from the College of Physical and Mathematical Sciences at Brigham Young University. We also acknowledge support from the Quantum Information group for the different ideas and discussions that led to the following thesis. Lastly support is acknowledge from Dr. Jean-Francois Van Huele for the advice, knowledge and direction given.

Contents

List of Figures

Chapter 1

Introduction

In this chapter introductory material, the comprehension of quantum purification protocol allowing for the understanding of all further Chapters. Section 1.1 introduces quantum information as a whole and the impact it will have. Section 1.2 introduces quantum circuitry and how quantum algorithms run on a computer. Section 1.3 gives a summary of the errors present in maximally entangled pairs. It is followed by Section 1.4 which looks into the standard protocol needed for Chapter 2. The final Section 1.5 gives a brief summary of the rest of the thesis.

1.1 Introduction to Quantum Information

Quantum Information uses the quantum behaviors at an atomic level to perform feats of information processing storage or transfer never thought possible 100 years ago. It is one of the fastest growing fields of science [1] due to the ramifications it could have in almost every aspect of life. “the recent interest in aspects common to quantum information and condensed matter has prompted a flurry of activity at the border of these disciplines that were far distant until few years ago.” [2] From quantum cryptography to quantum computers, the ideas that have been proposed have led many to speculate that we are on the edge of a quantum revolution bringing the promise of massive growth to every

field of science. This quantum revolution stems from the power of the qubit [3]. Qubits or quantum bits are the fundamental units in quantum information. A qubit is the fundamental state that can take on the state value $|0\rangle$, $|1\rangle$ or both $|0\rangle$ and $|1\rangle$ state at the same time. The material realization can be the spin of an electron, the polarization of light, or a cat that is both living and dead at the same time. This superposition of both $|0\rangle$ and $|1\rangle$ provides the power behind the information processing of quantum computers. When qubits in this superposition are measured, the qubit is forced in to either the state of $|1\rangle$ or $|0\rangle$. This measurement then allows for information on the state to be extracted. While it is unmeasured, it exists in both states simultaneously. Now this would not be so revolutionary if that was all, as it would mean that from the classical bit of two we now have two states and a third state that could essentially be either which would allow more information to be stored or processed. As more qubits are added to the system the number of possible states it can take grows exponentially because until a measurement is performed on the system, the quantum computer can be in every possible state that is represented in the system of qubits as shown in Eq. (1.1). On the other hand, a classical computer has to be in only one state as seen in Eq. (1.2).

$$|\phi\rangle = \frac{1}{\sqrt{4}}(|00\rangle + |10\rangle + |01\rangle + |11\rangle). \tag{1.1}$$

$$|\phi\rangle = |00\rangle. \tag{1.2}$$

This property allows for parallel computation on levels that outpace classical computers at a factor of 2^N where N represents the number of qubits [4]. This compares to classical computers where it only grows by a factor of $2N$.

Another quality of qubits is their ability to link with other qubits, in a process called quantum entanglement. This process makes it so that changes in the state of one qubit are reflected in the state of the other qubit regardless of distance. The mathematics behind quantum entanglement is simple: instead of one qubit in the superposition of

$$|\phi\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle), \tag{1.3}$$

we now have two qubits combined in the state(Bell) called $|\phi^+\rangle$.

$$|\phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle). \tag{1.4}$$

In Eq. 14 we have a superposition. Measurement of one qubit resulting in both qubits taking on the value either $|0\rangle$ or $|1\rangle$, we have two qubits where they are either both 0 or both 1. If we perform a measurement on the other qubit at a later time, it will always be (not accounting for error) in the same state as what was measured in the first qubit. The factor $\frac{1}{\sqrt{2}}$ is the normalization factor that, when squared, accounts for the 50 percent chance that the two qubits will be in the |00> state and the 50 percent chance they will be in the |11> state. This maximally entangled state is called a Bell State. The three other Bell States will be introduced later on.

The property of quantum entanglement implies correlated results at large distances and is instantaneous. However due to the randomness of the measurement in either part, this entanglement cannot be used for the transfer of information from one party to the other. Besides sending information from place to place, quantum entanglement provides key methods for different algorithms like Shor's Algorithm for factoring, Quantum Fourier Transformations, Ising Model etc.. [5]

The question to be addressed is: why are these improvements in computation not used today? One of the key downsides of quantum information is the fragile nature of quantum states. Quantum states can easily be corrupted. Due to the nature of storing information inside of microscopic or atomic level systems, any sort of disturbance leads to the states changing, causing errors in the quantum computer or algorithm. These disturbances can lead to states changing or the collapse of the superposition while an algorithm is processing. Not only do outside disturbances cause issues, but quantum systems also evolve through time. The more time passes, the more likely that a quantum system will collapse or change its state from the user's intention, thus introducing noise into the system. This process is called decoherence and stems from the fact that systems have a tendency to slowly lose their "quantumness" over time and slowly fall back into a classical system [5].

$$|\Phi^+\rangle = \frac{|00\rangle + |11\rangle}{\sqrt{2}}$$
$$|\Phi^-\rangle = \frac{|00\rangle - |11\rangle}{\sqrt{2}}$$
$$|\Psi^+\rangle = \frac{|01\rangle + |10\rangle}{\sqrt{2}}$$
$$|\Psi^-\rangle = \frac{|01\rangle - |10\rangle}{\sqrt{2}}$$

Figure 1.1 The four maximally entangled states that a system can take also known as Bell States

Another problem is scaling qubits. To create a single logical qubit, hundreds of physical qubits must be used to store the information to reduce the error. Creating this many qubits is resource intensive and hence very hard to scale up. The solution of creating the connection by entanglement is problematic, as the creation of entanglement itself is error prone [6].

Hence, methods to mitigate errors are key for the success of quantum information. Whether through hardware improvements that prevent errors or methods to mitigate errors from spreading, allows for the promise of quantum computing and information to continue. Error correction continues to grow and improve leading scientists to believe that the advent of the quantum revolution will happen in our lifetimes [7].

The research presented in this work aims to detects the error present in maximally entangled pairs and prevents those errors from propagating in a quantum circuit.

1.2 Quantum Circuitry

A maximally entangled qubit is a qubit entangled in one of the four Bell states, defined in Fig. 1.

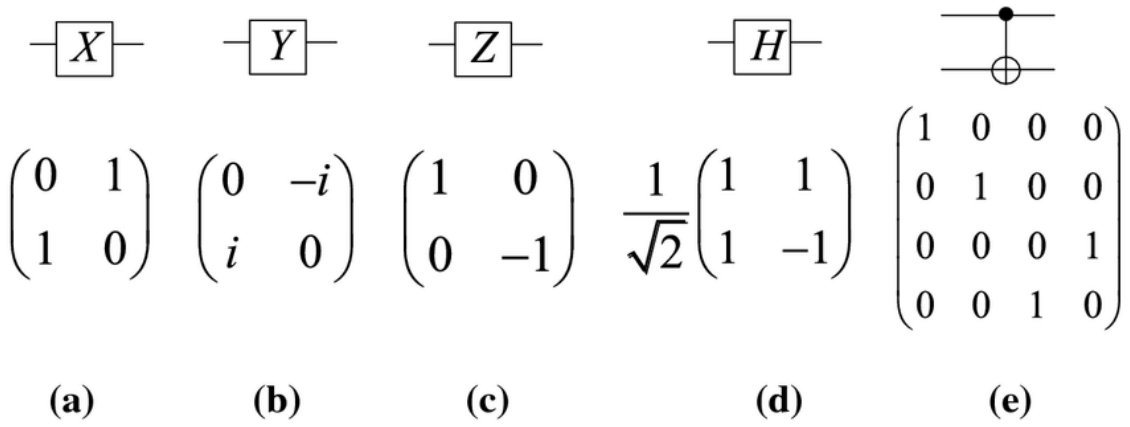

Figure 1.2 Basic gates of quantum circuits

The various basic gates of quantum circuits: X,Y,Z, Hadamard and Controlled-NOT

The $|\phi^+\rangle$ Bell state is the easiest to create "Given these initial conditions, the Bell state—the state of maximal entanglement between two qubits—that can be obtained with the fewest number of quantum gates is the $|\phi^+\rangle$ Bell state. The experimental application of quantum gates introduces noise and comes at a monetary cost. This makes the $|\phi^+\rangle$ Bell state the most efficient and affordable given the initial states." [8]. To learn how to create these states and how quantum algorithms operate, it is necessary to look into how quantum computers perform operations on qubits through the usage of quantum circuitry.

Quantum computers use the same principles as classical computers in that a circuit is created to manipulate the various bits or qubits within the system. The circuits work based on gates that are performed successively during the process. These gates can connect qubits and perform state changes based on the matrix associated with them.

The matrices in Fig. 1.2 show the different gate operations that can be performed. The X (Fig. 1.2(a)) gate causes the $|0\rangle$ to flip to a $|1\rangle$ and vice versa. The Hadamard (Fig. 1.2(d)) gate or H-gate performs a superposition operation to take the $|0\rangle$ to a $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ superposition as shown in Figure 1.3. To produce an entangled state the CNOT gate (Fig. 1.2(e)) between two qubits is used.

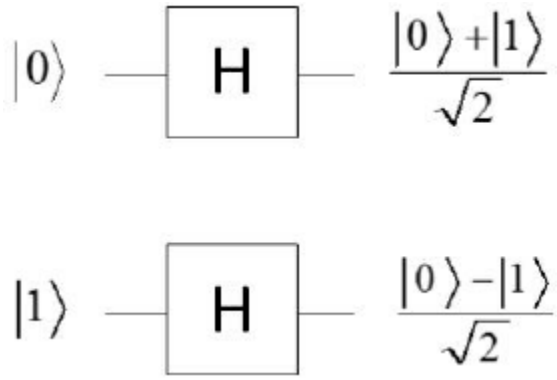

Figure 1.3 The Hadamard Gate

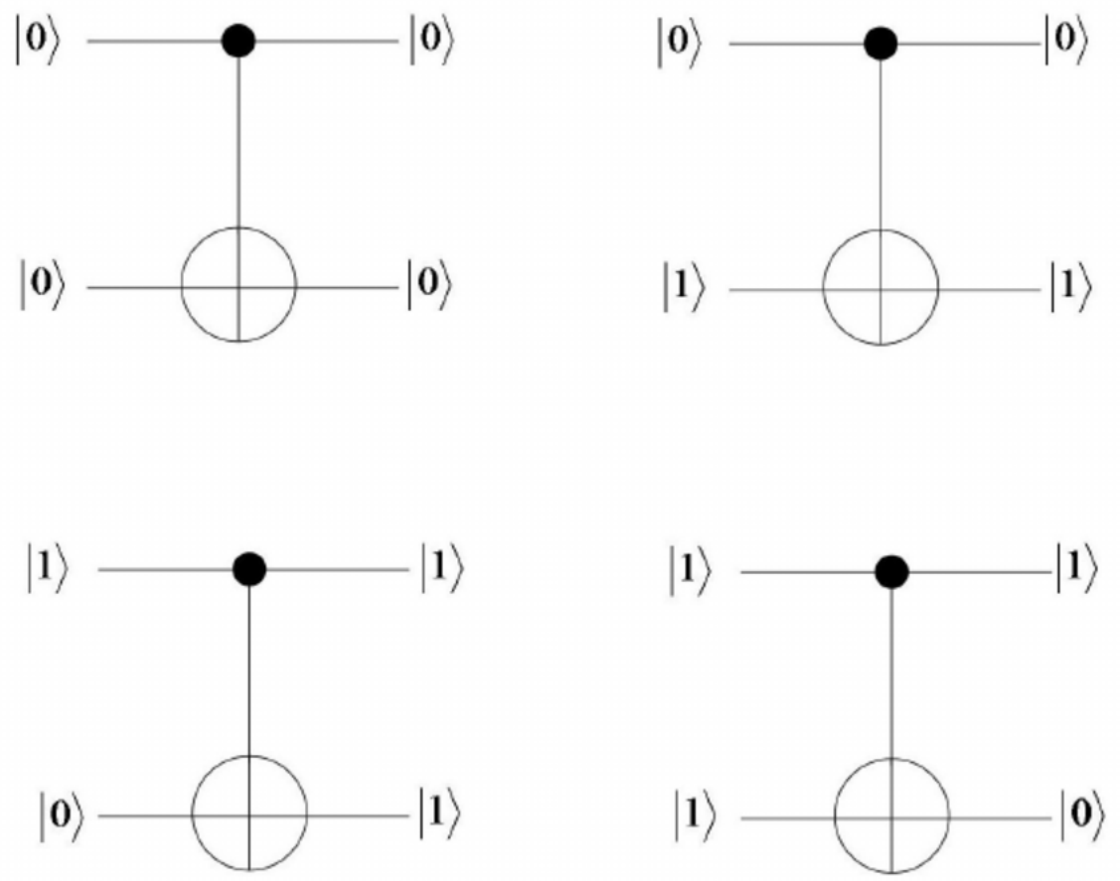

Figure 1.4 The four outcomes of the qubits after passing through a CNOT gate. As can be observed, the state will cause a change based on the control qubit(black dot) and the control (circle plus)

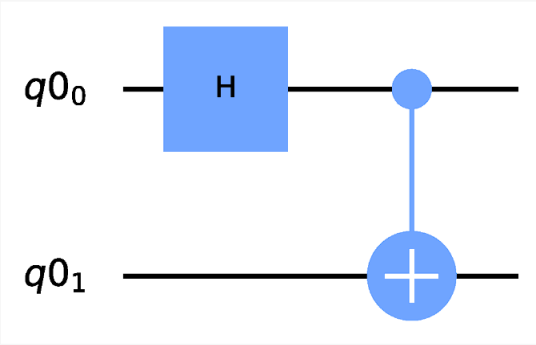

Figure 1.5 The most basic circuit for entanglement of two qubits

The CNOT gate, in Fig. 1.4, changes the state of the target qubit (circle plus) based on the state of the control qubit (black dot). If the control qubit is in the $|0\rangle$ no change occurs, but if the control qubit is in the $|1\rangle$ state, then it causes the target qubit to flip its state as if operated on by a single qubit X gate. If a superpositioned qubit is used as the control qubit, the target bit is forced into a superposition based on the control bits superposition, hence an entangled state, such as can be seen in Fig 1.5.

This entangled state is the $|\phi^+\rangle$ state and it is maximally entangled as long as the target bit was initialized as $|0\rangle$. To produce the other Bell states more gates are needed such as the X gate, Y gate, or Z gate, utilizing more quantum resources. Therefore, it is best to use the $|\phi^+\rangle$ state for operations requiring entanglement.

1.3 Entanglement Errors

As we use the $|\phi^+\rangle$ state for maximally entangled particles any spontaneous occurring other Bell states become errors in the system. Each Bell state represents a different error that could occur inside a quantum system. The most common errors that can occur are errors along the X,Y,Z basis as shown in Fig 1.6. Each provides a new challenge to detect.

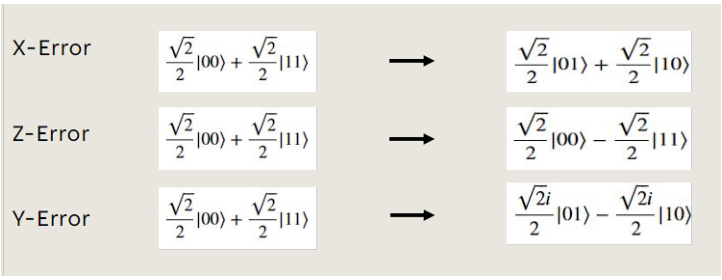

Figure 1.6 Representation of the different types of errors in one qubit

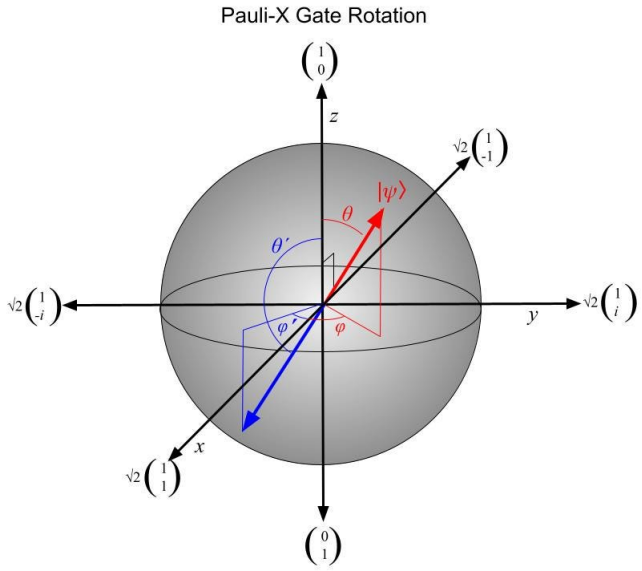

Figure 1.7 The representation of a quantum state through a Bloch Sphere

X errors causes bit-flips as previously discussed, Z errors are phase changes in the Bell state such as $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ to a $\frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$ changing the relative phase. Relative phase is part of the quantum state and the best way to represent a phase shift is on the Bloch Sphere, a useful representation of a quantum system and the angle corresponds to the phase, as seen in Fig. 1.7.

For maximally entangled qubits, the phase that can lead to an undetected error propagating is a phase shift in the Z-basis. Other types of phase shifts are less important and so will not be focused on for this protocol [9]. Y errors are both phase and bit flip changes, rotating into the imaginary plane. An example is $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ to $\frac{i}{\sqrt{2}}(|0\rangle - |1\rangle)$

Failing to detect these errors, various gate operations need to be implemented. Failing to detect these errors, allows for the propagation of incorrect states, leading to measurements that are meaningless or misleading.

1.4 Basic Purification

Purification is one method of error correction that can be used to prevent errors from propagating in a quantum circuit. Purification uses a newly entangled pair of qubits to improve the fidelity of another pair. Fidelity is the trustworthiness that a pair of qubits is in the correct state for quantum operations. As the fidelity of an entangled pair approaches the value of 1, it is more likely to be in the correct state. A fidelity of 1 would imply a 100 percent chance it is in the correct state. Due to the nature of quantum systems, it is impossible to have 100 percent certainty. It can only come from a measurement being performed, which ruins the quantum state. Thus purification attempts to approach a fidelity of one by using sacrificial and target pairs. By performing measurements on the sacrificial pair, we can obtain some information from the target pair without collapsing its superposition, allowing the user to trust their target pair and hence increasing the fidelity.

Four qubits q0 and q1, q2 and q3 are entangled two-by-two using a standard Hadamard and a CNOT gate. Both pairs are ideally in the target state $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$. Qubits q0 and q1 are the target pair while q2 and q3 are the sacrificial pair. By using two additional CNOT gates, errors that could possibly appear on the target pair can transfer onto the sacrificial pair and then through measurement on different bases be detected by the observer.

Measurements can be performed in different bases to detect different errors. As there are X, Y, and Z errors, measurements in the XYZ basis leads to different errors being seen. For example if one measures in the X basis one can see Z phase errors or Y errors. In the Z basis one can see X

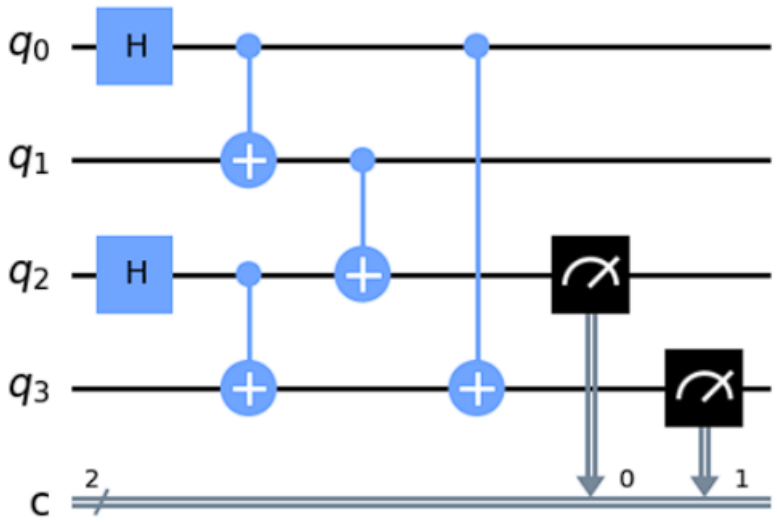

Figure 1.8 The basic purification protocol utilizing a target pair and sacrificial pair [10]. The H-gate and initial CNOT gate perform the entanglement as shown in figure 1.5. The second CNOT gate on q1 to q2 and q0 to q3 are the gates to pass the error over onto the sacrificial pair q2,q3. The final two symbols on q2 and q3 are the measurement onto the classical bit of c that stores the information.

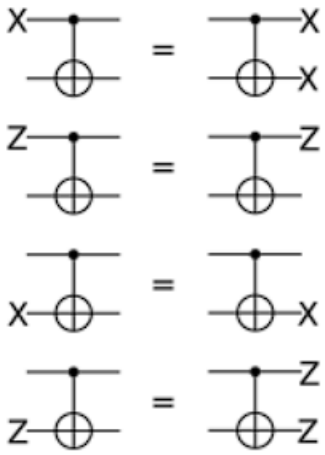

Figure 1.9 Method in how errors propagate through a CNOT gate. The letters indicate corresponding errors on the qubit

and Y errors and lastly in the Y basis X and Z errors can be seen. Errors in each of the different bases propagate in different manners.

By using how the errors pass on the CNOT gate, as shown in Fig. 1.9, it is possible to detect different errors present in the target pair by measuring the sacrificial pair. Z errors pass from the control qubit to the controlled qubit while X errors pass from the control qubit to the controlled qubit. By passing the errors from the the target pair to a sacrificial pair, we can measure for different errors. With these different methods of propagation and optimization of the protocol, a new protocol can be made that diminishes the usage of quantum resources while dramatically increasing the fidelity of the entangled pair.

1.5 Overview of Thesis and Chapters

The thesis will focus on the development of a new protocol created by the author. It will look into the actual protocol explaining the different parts of the circuit and how the fidelity changes. Then data will be presented as the protocol is run on a simulated quantum computer. Data collection is

performed to find the fidelity change for different initial fidelities for different configurations of the protocol to be presented. From there we will discuss how this protocol performs and future methods that will be implemented along with applications.

Chapter 2

Methods

The thesis focuses on the method of purification for maximally entangled systems using the least amount of resources with maximum fidelity gain. This is done by the usage of different methods discussed in the following sections. Section 2.1 focuses on the method of bit error detection and Section 2.2 looks into phase error detection. Section 2.3 then combines the two circuits to develop the full protocol and explains how it functions in detail. Finally, Section 2.4 discusses fidelity and how it is affected by the circuit as a whole.

2.1 Bit Error Detection

To optimize the purification protocol such that we increase the fidelity as much as possible, it is necessary to look at the errors and examine how best to detect these errors utilizing the least amount of quantum resources possible. As there are two main errors that can occur within maximally entangled particles, the protocol needs to focus on checking and preventing these two errors from spreading. By creating two different circuits, one based on checking for bit-flip and a second circuit checking for phase-flips, we can combine the two to create a comprehensive circuit that is able to detect the majority of errors on the target pair.

First, we need to examine bit-flips. The basic protocol is able to detect bit-flip errors most of the time, only failing when both qubits are in the incorrect state. The basic protocol uses two pairs to transfer bit-flip errors onto the sacrificial pair through two CNOT gates. Another manner of detection without using the resources to build an entangled pair is to temporarily entangle a qubit with the target pair and then untangle it, allowing for errors in the pair to pass onto the arbitrary qubit which can then be measured through some observation device depending on the system.

$$|0\rangle\left|\phi^{+}\right\rangle = \frac{1}{\sqrt{2}}(|000\rangle + |011\rangle) \xrightarrow[CNOT]{} \frac{1}{\sqrt{2}}(|000\rangle + |111\rangle) \xrightarrow[CNOT]{} \frac{1}{\sqrt{2}}(|000\rangle + |011\rangle) = |0\rangle\left|\phi^{+}\right\rangle \tag{2.1}$$

$$|0\rangle\left|\psi^{+}\right\rangle = \frac{1}{\sqrt{2}}(|001\rangle + |010\rangle) \xrightarrow[CNOT]{} \frac{1}{\sqrt{2}}(|001\rangle + |110\rangle) \xrightarrow[CNOT]{} \frac{1}{\sqrt{2}}(|101\rangle + |110\rangle) = |1\rangle\left|\psi^{+}\right\rangle \tag{2.2}$$

By following the arrows, one can see putting the arbitrary qubit into an entangled state with the target pair and then untangling. To understand we have three qubits in a superposition. In this case we have three qubits $|abc\rangle$ with a being the arbitrary qubit, and bc being the entangled bits. This corresponds to $|000\rangle$ and $|011\rangle$. We see that if the state is in the $|\phi^{+}\rangle$ the arbitrary qubit remains in the $|0\rangle$ state. If the target pair is in the incorrect state of $|\psi^{+}\rangle$ or $|\psi^{-}\rangle$, meaning a bit-flip of some degree, the arbitrary qubit flips to a $|1\rangle$ state. By performing a basic measurement then on the arbitrary qubit, we can extract the information of whether a bit-flip occurred within the target pair. This arbitrary qubit can then be used multiple times within the same circuit as measuring it does nothing to its state and does not cause any changes. The circuit can be constructed in this manner for detecting a bit-flip change as seen in Figure 2.1.

2.2 Phase Error Detection

As we have now been able to create a circuit that allows for the detection of bit-flips within an entangled pair for minimal resources it is necessary now to examine how to detect a phase-flip

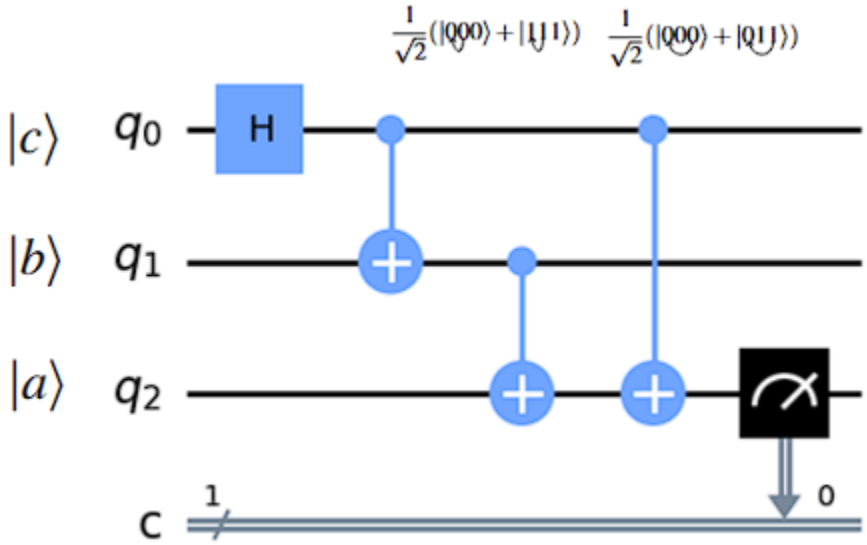

Figure 2.1 A protocol for checking for a bit-flip error in an entangled system

within an entangled pair. Maximal phase change or Z-errors exists when there are qubits put into a superposition. The easiest way to detect a phase change is to change the basis of the pair and applying the bit-flip check using the CNOT gates. By using a series of gates as shown in Figure 2.2 we can encode the phase into the bit information and the bit information in the phase. For example if the phase is $+$ then after the series of gates the new bit information becomes $|00\rangle + |11\rangle$ while a phase of $-$ becomes $|01\rangle + |10\rangle$. Using a simple bit-flip check, we can detect a phase error in the entangled pair. With the passing of the bit-flip check, we perform the series of gates again to return the entangled pair into its original state.

By using the bit-flip protocol and phase-flip protocol we can attempt to catch both errors. The complete circuit can then be made and put into form as shown in Figure 2.3.

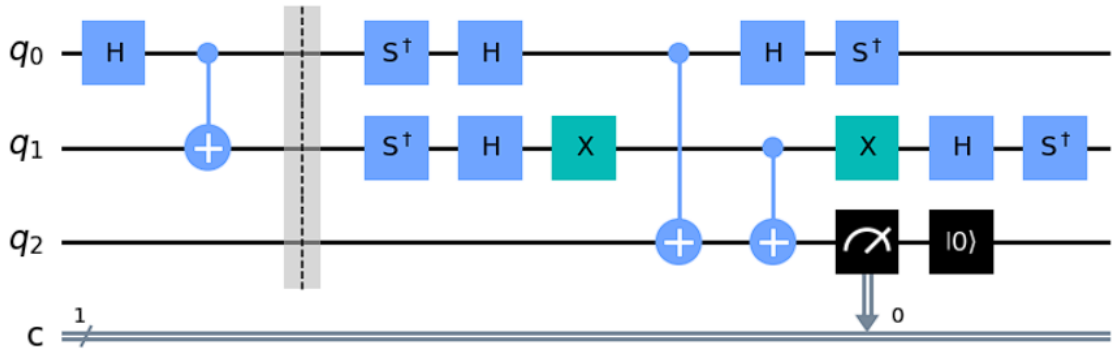

Figure 2.2 A protocol for checking for a phase error in an entangled system. q0 and q1 are the entangled pair put into an entangled state by the H-gate and CNOT gate. The arbitrary qubit is q2 which allows for the bit-flip check to be performed. It is then measured at the end of the protocol. The S^{+} induces a phase change to get the phase to being encoded into the bit information.

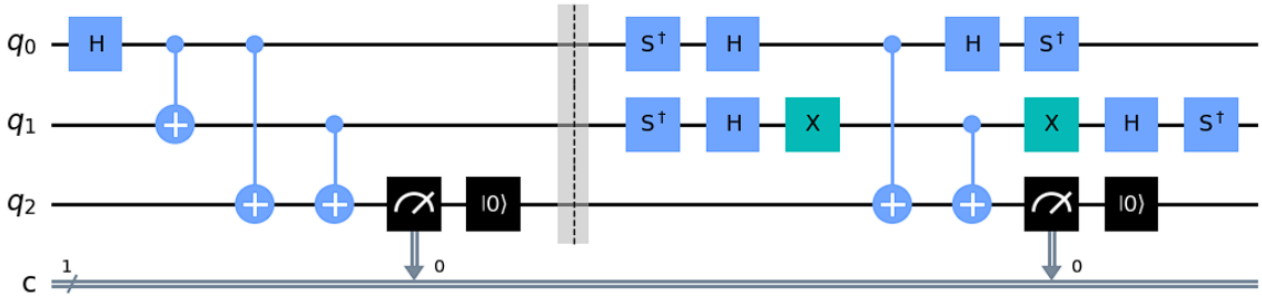

Figure 2.3 Completed circuit for determining error in a maximally entangled system

2.3 Fidelity

As mentioned before, fidelity is how trustworthy an entangled system is. Its value ranges from 0 to 1, where a fidelity of 0 means 0 percent chance of being in the right state while 1 means a 100 percent chance of being in the correct state. 0.5 would give us the least amount of information as there is a 50 percent chance of being in the correct state or 50 percent chance of being in the wrong state. Due to the uncertainty of quantum systems and evolution of quantum systems through time, it is impossible to have a fidelity of 1 as that implies a collapse of the superposition and the particle's state being known which brings us back to classic computing. Therefore, it is the goal to increase the fidelity to as close to one as possible, so that it is highly unlikely that the state, is in the incorrect state, thereby letting the pair be used in other operations without fear of error of decoherence.

To determine the change of fidelity through the usage of the purification protocol developed, we need to map every possible state that the entangled system could possibly take before the protocol. After we check to see what the circuit does to the system, after evolving the state, we look to see if some undetected error was able to pass through the circuit. We then tally the undetected errors, and can then determine the final fidelity of the target pair. We will also need to include the sources of error from gate infidelity and depolarization from the CNOT gates and single gates.

The different states that the entangled systems could take have are tabulated in Fig. 2.4. The table shows each of the possible states the system could possibly take and examines how each of the states are evolved through the circuit. Let's take for example the I0 state. As I0 is in the correct state: as it evolves through the circuit, it does not change(if gate infidelity is not taken into account) when it emerges from the circuit nothing has changed and thus we detect that the target pair is in the correct state. If however the state is X0, we see that the circuit evolves such that both of the pairs are in the incorrect state afterwards. Therefore we discard the target pair and attempt to try the protocol again.

I $|\Phi^+\rangle$ $\frac{\sqrt{2}}{2}|00\rangle + \frac{\sqrt{2}}{2}|11\rangle$

Z $|\Phi^-\rangle$ $\frac{\sqrt{2}}{2}|00\rangle - \frac{\sqrt{2}}{2}|11\rangle$

X $|\Psi^+\rangle$ $\frac{\sqrt{2}}{2}|01\rangle + \frac{\sqrt{2}}{2}|10\rangle$

Y $|\Psi^-\rangle$ $\frac{\sqrt{2}}{2}|01\rangle - \frac{\sqrt{2}}{2}|10\rangle$

$|bc\rangle\,|a\rangle$

Before Bit-flip Check	After Bit-flip Check	Result	After Phase-Flip check	Result
I 0	I 0	pass	I 0	pass/good
I 1	I 1	-	-	-
X 0	X 1	-	-	-
X 1	X 0	pass	X 0	pass/bad
Z 0	Z 0	pass	Z 1	-
Z 1	Z 1	-	-	-
Y 0	Y 1	-	-	-
Y 1	Y 0	pass	Y 1	-

Figure 2.4 Table of every possible state combination of before the circuit and then evolved after. For each category we have the state of the entangled state represented by the letter of the first digit while the second digit represents the state of the arbitrary qubit. Explanation is given in the text.

The only possible error to escape detection comes in the form of the X1 error. This error puts the arbitrary qubit into the correct state as we also assume that the arbitrary qubit is in the wrong state. Coupling this fact with the lack of phase error and we see that the protocol does not detect any sort of error on the circuit. Due to this fact, it is assumed the entangled pair is in the correct state and is then used allowing corruption to spread in the algorithm. As mentioned before, in order to calculate fidelity we need to tally the total number of undetected errors namely one against the total number of states possible namely eight. We then multiply the total number by the infidelity of the gates ε and once more by 8 due to the number of errors the gates connecting to the entangled pair can cause. The infidelity of the gates in this protocol is considered to be 0.01 or a 0.01 chance of depolarization(error) due to the CNOT gates that are connected to the target pair. We then add the error from the single gates which amounts to adding the error rate of 0.0001 multiplied by the amount of the 10 gates in the circuit [9]

By using the equation as a basis and the above explanation we can account for depolarization in our fidelity calculator. Our final equation is thus

$$F = 1 - \frac{\varepsilon}{8} \cdot 4. \tag{2.3}$$

As a result, the final fidelity should come out to be 0.989 for qubits of fidelity 0.9 inputted. We assume qubits of this fidelity are used, as qubits of lower fidelity have a much higher chance of failing the protocol. We also assume the decoherence time is longer than the circuit. If it were not, any part of the detection circuit itself would be prone to state corruption leading to the protocol being functionally useless.

2.4 Optimization

Due to the modularity of the bit-flip and phase-flip checks it is possible to run an algorithm that can find the optimal number of each check for the maximal fidelity gain for different initial fidelities to reach some target fidelity. To perform this algorithm there are two important steps. The first is the data collection of how the fidelity changes depending on the number of the bit-flip and phase-flip checks. Second, performing quantum operations has a cost. For example, performing a measurement collapses the quantum state requiring more quantum operations to renew the state. Each quantum operation has some cost associated with it. Hence, a resource optimization algorithm that reduces resource usage and maxmizes fidelity gains will give us the optimal number of each check for a target fidelity.

To find the optimal number data collection is required. To find the final fidelity from some initial fidelity a simulation of the circuit is run. When the circuit runs there are three possible outcomes: A successful run (S) meaning the final quantum state is in the $|\phi^+\rangle$ at the final check. A detected error (D_e), where an error was detected by one of the checks, which terminates the circuit. Finally, an undetected error (U_e) could pass through the circuit. This undetected error is the source of infidelity

in the circuit. So the final fidelity equation can be generalized as the ratio of successes and detected errors over all runs

$$F_{final} = \frac{D_e + S}{D_e + S + U_e}. \tag{2.4}$$

This is the probability that our state is correct and can be used for a final fidelity calculation.

Given the final fidelity calculation, we then only need to perform a sufficient number runs on the circuit tallying up the final fidelity for each run for that particular initial fidelity.

For the resource optimization the we use the following equation based on the number of gates and diverse weights

$$R_{Lost} = (2W_C + W_Q) \cdot \theta_1 + (2W_C + 5W_S + W_Q) \cdot \theta_2 + C_s \cdot W_Q. \tag{2.5}$$

When examining the different checks, we find that a bit-flip check requires one qubit and two two-qubit gates, while a phase-flip check uses one qubit, five single-qubit gates, and two two-qubit gates. Since creating qubits and operating two-qubit gates uses more resources than applying single-qubit gates, we assign different weights to each. In our work, two-qubit gates (W_C) and qubits (W_Q) have a weight of 5 each, while single-qubit gates (W_S) have a weight of 1. These weights are assigned arbitrarily based on difficulty of implementation.

Additionally, each run of the protocol comes with a cost (C_s) that depends on the initial fidelity of the Bell pair. If the initial fidelity is low, there is a higher chance that errors will be detected, causing the pair to be discarded and the protocol to be run again, which wastes resources.

The goal is to find the optimal number of bit-flip checks (θ_1) and phase-flip checks (θ_2) that yield the highest fidelity gain for the lowest overall resource cost. We do this by combining the weighted cost of qubits and gates with the improvement in fidelity to set up an optimization problem. A minimization algorithm is then applied over the possible values of θ_1 and θ_2 to determine the best configuration that meets the fidelity target while minimizing the resource use.

Chapter 3

Results

This chapter focuses on the method of collection of data with the protocol created in Chapter 2. Data is then presented through various graphs. The analysis is given in Chapter 4. Section 3.1 gives the construction of the simulation and the motivation for using particular methods. Section 3.2 presents the data and gives a preliminary analysis including the possible errors in data collection.

3.1 Simulation for Data Collection

I have taken data including for simulated noise so as to predict the success rate of the purification protocol for a given fidelity. The data was taken through a program built on Qiskit, a quantum programming language [11]. To simulate the noise on a quantum computer using an initial fidelity of 0.9, it was necessary to build a code that includes the probability of depolarization each time there was an initialization for an entangled pair. The code takes into account these possible depolarizations by calling a function that randomizes the chance of an error occurring. For instance, when a pair is initialized into the $|\phi^+\rangle$ there is a 10 percent chance (for a fidelity of 0.9) of it being put into one of the other Bell States, each having a 3.33 percent chance of occurring. To include CNOT gate and single gate error a similar method was used. Whenever a single gate or CNOT gate is called on

each qubit of the gate, there is a chance of gate failure and error. For CNOT gates that means a 0.01 percent chance of failure while single gates have only a 0.001 percent chance of failure. Therefore we can simulate the noise that may occur during initialization of a Bell pair.

We will assume that the decoherence length of the Bell pairs or qubits used is longer than the protocol. The assumption is that the hardware used has reached the point where purification protocols or software based implementations of error correction are necessary for continued improvements.

I did not use a real quantum computer because of the inability for termination of the circuit and for the purpose of fine-tuning the parameters of the model. It is currently not possible to do an interruption of the circuit in the middle of the runtime. As the goal of the purification protocol is to attempt to maximally increase the fidelity and simultaneously minimizing quantum resources,the protocol has to terminate at the moment of detection of error. Without the ability to terminate at error detection, which is handled by classical logic, it is simpler to build a simulation to simulate noise for all parts of the circuit thus allowing for the termination of the protocol at any point at the detection of error. This allows me to fine tune the different parameters. The fine-tuning needs to be done classically because quantum computers are currently too fragile to handle it.

3.2 Data Taken

After the simulation was built, the circuit was ran on the Python program Qiskit. Data was then collected on the number of qubits used, on percentage of success, and on how much the fidelity increased with a final measurement. This was done in order to check for a false positive error, i.e., the error could have passed through the circuit without detection. This would be the phase and bit errors that could have possibly bypassed the checks without being detected. Data was then taken through two different methods. First, I checked how many qubits it took to extract a purified pair

that could be used. Second, I checked how much the fidelity improved depending on the initial fidelities.

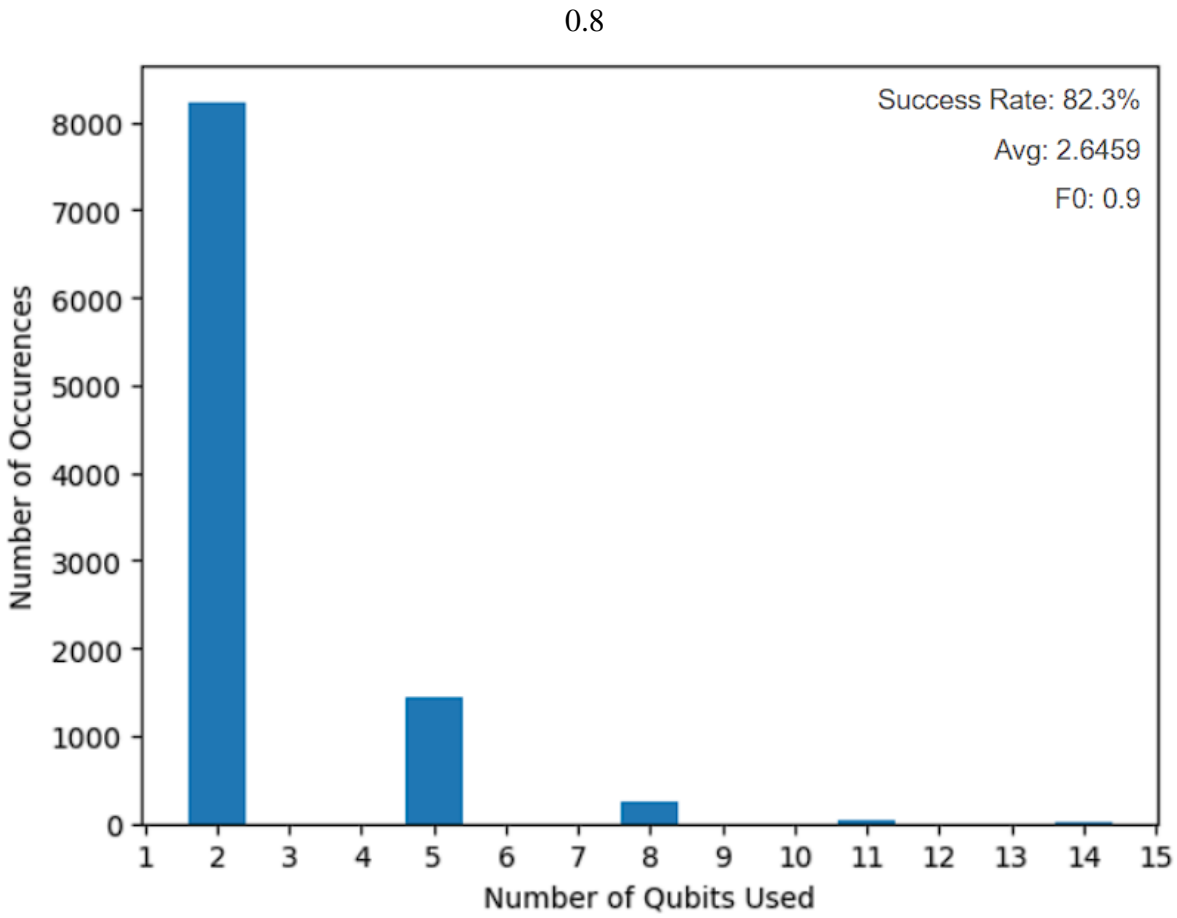

Figure 3.1

Figure 3.2 Graph of the data taken from the simulation. Success rate indicates the chance of a trial to be successful. The Avg is the average number of qubits it took to complete a successful run of the protocol. F0 is the initial fidelity. Explanation of how the protocol was tested is given in the text.

The data presented in Fig. 3.1 was taken on the simulation with 10,000 trials. Trials were done by running the protocol as many times as needed to have a successful run of the protocol. This allowed us to collect data on the number of qubits used on average and on the success rate. The

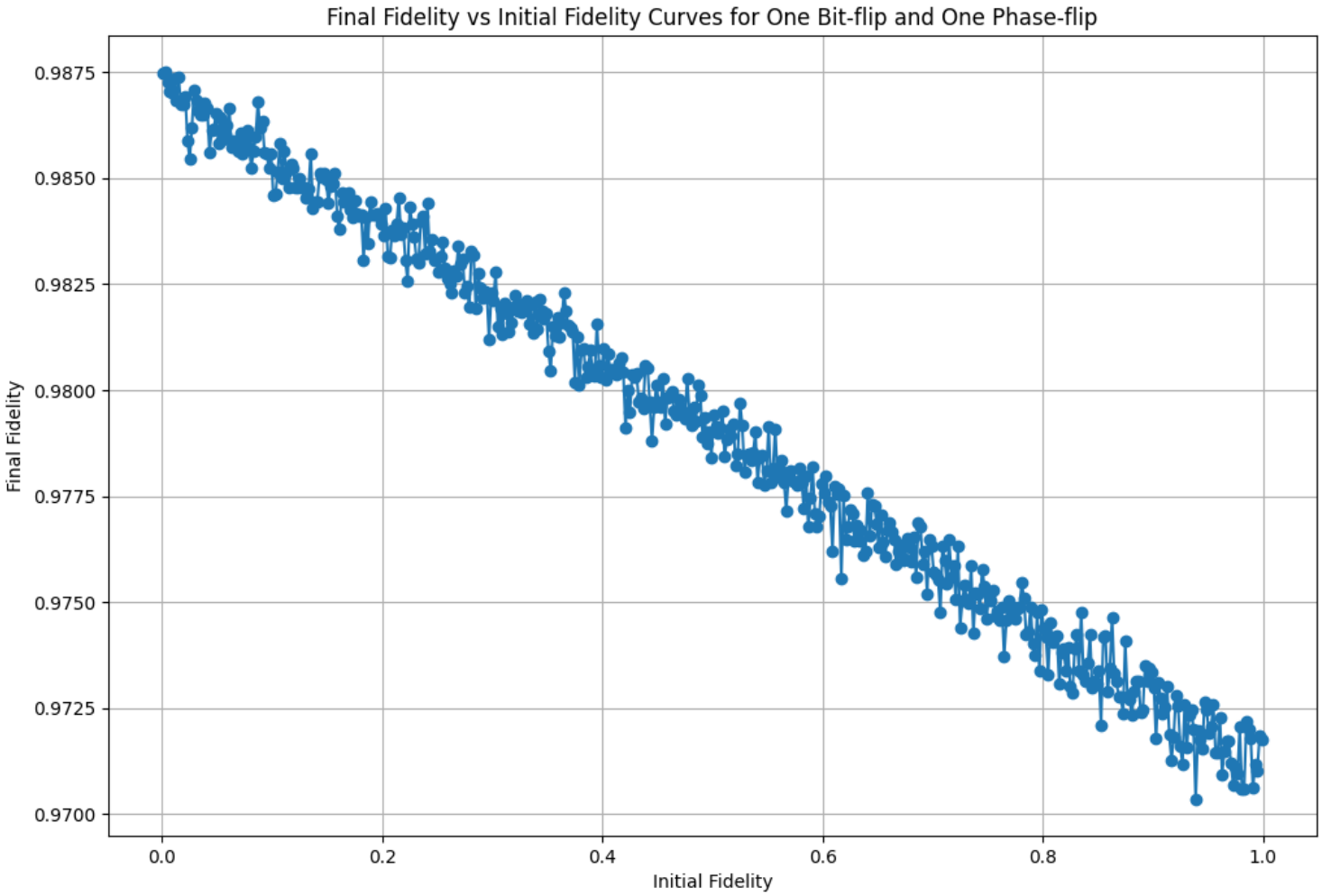

Figure 3.3 The total range of of initial fidelities to final fidelities for θ values of one for each check.

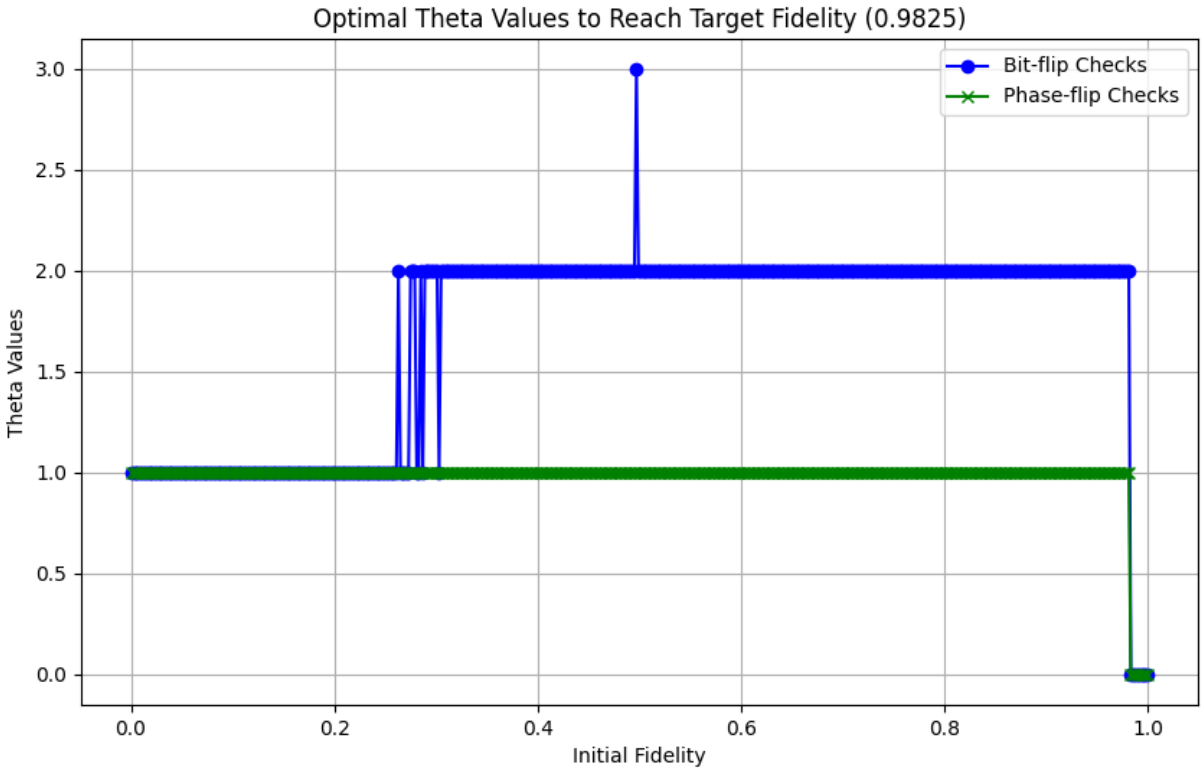

Figure 3.4 The optimal θ values for each check to reach the target fidelity f 0.9825. Bit-flip checks are marked in blue while the phase-flip checks are marked in green. The drop to zero results from the initial fidelity being higher than the target fidelity.

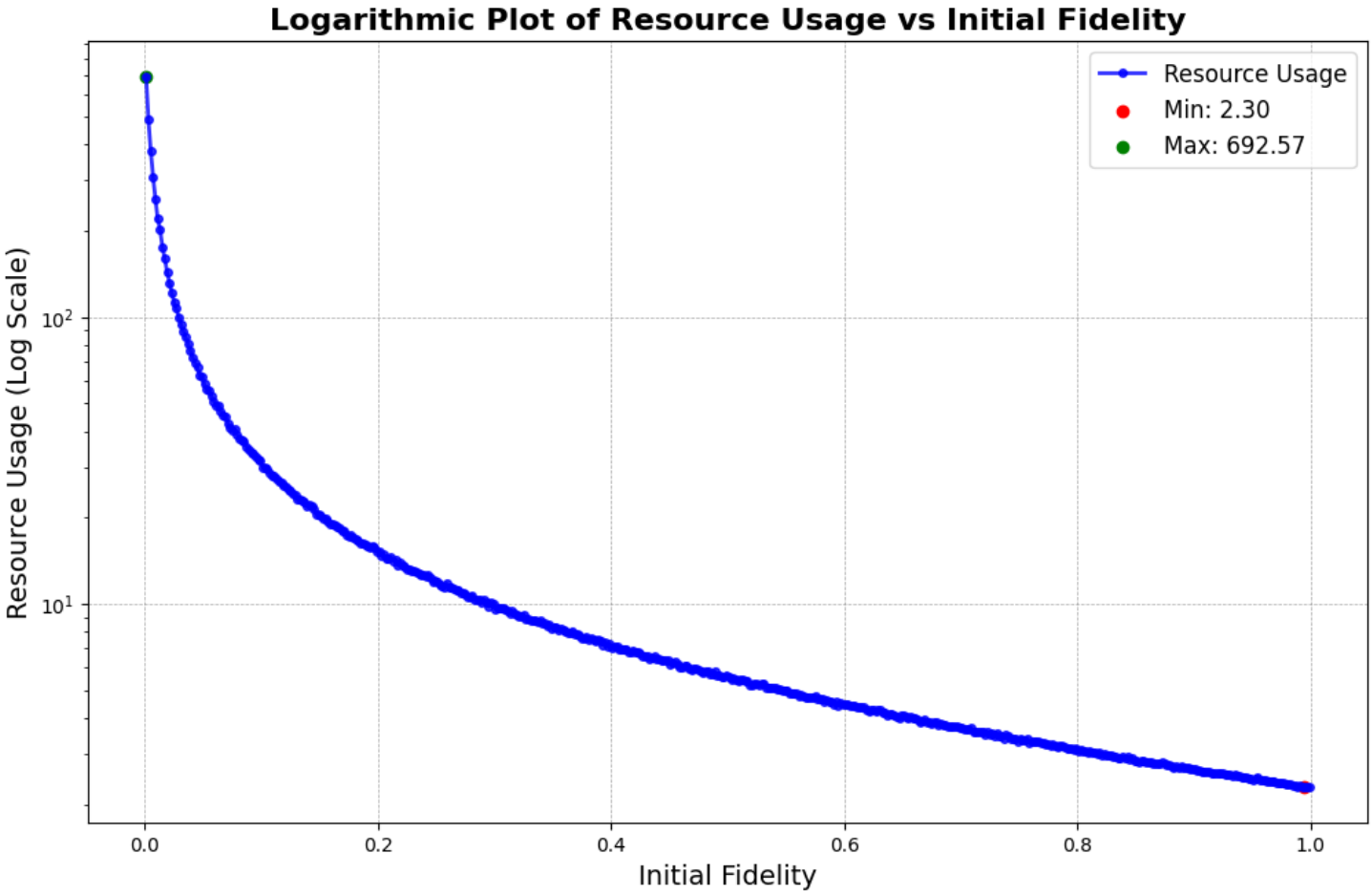

Figure 3.5 The number of resources used based on Eq. 2.5 for the initial fidelity as it ran through the optimization algorithm. Min represents the minimum amount of qubits used while Max represents the maxmimum amount of qubits used on average for a successful run.

instantiation error or the fidelity of the pairs is 0.9 in Fig. 3.1. A more comprehensive analysis is given in Section 4.2.

The data presented in Fig. 3.2 was taken by running 100 trials of 1000 different instances of the circuit. The 1000 instances of the circuit allows us to calculate the fidelity of that trial which was then averaged in the 100 trials to collect that specific data point. This was then run for the 500 different data points in the graph ranging from initial fidelity of 0.001 to 0.999. More detail is given in Section 4.2.

Following the procedure outlined in Section 2.4 we obtain the optimization results for the number of bit-flip and phase-flip checks to be used to reach the target fidelity of 0.9825. This number was chosen for a high final fidelity that nearly every initial fidelity could reach based on the data collected. The results of which checks to use are plotted in Fig. 3.3. The resource graph in Fig. 3.4 shows the cost of resources used to reach that target fidelity given some initial fidelity.

Chapter 4

Discussion

In this Chapter, a short discussion on the analysis of the data simulation is given. The analysis focuses mostly on how well the protocol performed, the change in fidelity, and lastly the data collected. Section 4.1 gives a brief summary of the thesis as a whole. Analysis is given on the data with conclusions drawn in Section 4.2. The final section, Section 4.3 then gives a summary of the conclusions and further research that can be performed.

4.1 Summary

In summary, different quantum processes to run a particular algorithm depends on the use of maximally entangled Bell pairs to run their algorithms but due to error from different sources it is hard for these algorithms to prevent problems from spreading in a quantum circuit. The protocol developed in this work detects errors to prevent the corruption spreading from qubit to qubit which can then generate high-fidelity pairs for the usage in these protocols. To generate these high-fidelity pairs, the protocol checks for the most-likely errors that occur in maximally entangled Bell pairs. The protocol is then optimized for a minimal use of quantum resources and for the maximal increase in fidelity.

4.2 Analysis

From the data, one sees that the protocol runs in such a way to allow for the maximization of fidelity with minimization of quantum resources. First, the fidelity increase is optimal for almost all fidelities larger than 0.5. If an entangled pair is able to pass through the protocol with a fidelity of 0.9 the increase is to 0.9735, which is more than a 7 percent increase in fidelity. This is impactful when running through thousands of entangled pairs. This operation only takes upwards of 3 qubits to achieve with a success rate of 82.3% for freshly instantiated pairs. For every successful run of the protocol the number of operations is 15. The only possible error that can pass the protocol undetected is the X 1 error which requires two incorrect instantiations of the arbitrary qubit. This means that the chances of this error occurring is 0.09 percent. Thus, it can be inferred that the protocol can detect almost every possible error and extract the information while leaving some amount of uncertainty as is required to not violate the no-cloning theorem or not have a fidelity of 1.

As seen in Fig. 3.2, an initial fidelity of 0.1 increases to nearly 0.9850. Despite this, it can be seen in Fig. 3.5, the number of qubits used to develop a qubit of this fidelity is nearly 700 on the resource scale, resulting in a major loss of quantum resources to develop a qubit of this fidelity. While the protocol boosts the fidelity of lower fidelities to higher fidelities, this is not its most important feature. The protocol increases the fidelity with low use of gates while using very few arbitrary qubits for this increase.

The protocol performs reasonably well compared to the Optimized Entanglement Protocol [9]. While the optimized protocol reaches fidelities that are higher, it require nearly 22 entangled pairs and 34 operations on both sides of the circuit. At the same time the fidelity reached after multiple operations is 0.995, in the ideal case. In practice, with a six qubit circuit for their operations, the fidelity only approached about 0.99.

The protocol works well when it comes to passing an error on to the target pair. From the CNOT gate only certain errors can pass through. From the bit-flip check the only possible error

that could pass through is a phase-flip error, which is not possible from a single qubit unless it is in a superposition. While the arbitrary qubit is put into a superposition with the target pair, it does not pass on any phase error as it gains the phase of the target pair. The phase error check could possibly pass on a phase error if the arbitrary qubit is in the incorrect state of 1. As mentioned, if the decoherence time is assumed to be longer than the circuit, then this should prevent bit errors from passing onto the target pair and instead be detected beforehand. This manner of preventing bit errors from spreading to the target pair is due to a bias towards bit error. Most protocols focus on measuring the bits of the pair they are using. Phase, while important, is less so when compared to information stored in the bits. Phase error is still able to cause problems with certain gates, for example, protocols that use the Hadamard gate will encounter error transfer if the phase is changed.

Examining the optimization in Fig. 3.3, we can see that the optimal values for the lower initial fidelity values are low. This occurs because detected failures contribute to increasing the fidelity gain. While very low fidelities can result in large fidelity changes, the resources used outweigh this fidelity change, as can be seen in Fig. 3.4. The resources lost for one successful entangled pair from a low fidelity pair is prohibitively expensive. Further examination of the optimization shows that higher fidelity values require more checks to reach the target fidelity. This follows because higher fidelity values are increasingly more difficult to improve due to the error rates of the circuits themselves. High-fidelity pairs of 0.999 will be reduced in fidelity as they pass through components with lower fidelity rates than their own fidelities. As mentioned before, the θ values are zero for the highest fidelities since there is no reason to run the checks on fidelities larger than the target fidelity. The best resource efficiency for fidelity gain occurs for initial fidelities in the range of 0.831 to 0.973.

Speculating on why the model emphasized bit-flip checks over phase-flip checks, we note that bit-flip checks require fewer resources and have an equal chance of increasing the fidelity due to the equal probability treatment of the phase and bit errors. A higher target fidelity was not

chosen because the quantum gate fidelities limit the extent of improvement; at high fidelities, the error rates are increasingly dominated by the circuit rather than by the instantiation. In conclusion, the analysis shows that the protocol is effective in detecting errors on an entangled pair and in generating high-fidelity pairs. With a complete mapping of the protocol, it becomes clear that it can both prevent errors and increase the fidelity of an entangled pair, while also generating high-fidelity pairs.

4.3 Conclusions and Further Research

Applying the protocol itself is simple for a variety of different algorithms that require high-fidelity Bell pairs and is simple to implement with a single usage of the protocol being sufficient for most cases. It is a useful protocol due to the minimization of the quantity of quantum resources while also being able to prevent the majority of errors from passing through with only the rare instance of one error which requires the rare chance of incorrect instantiation to occur.

The simulation may also be updated to the current version of Qiskit I might also be changed to run on a real current quantum computer. It is possible that classical logic can be applied and allow for the termination of the circuit at certain points. While it would mean that certain factors could not be controlled as closely as possible with the data given by IBM quantum computers it is still possible to extract meaningful data. Hopefully this means a more comprehensive data set can be taken to perform further analysis.

Bibliography

[1] "Quantum Physics Accelerates to Top of Soaring Scientific Fields,"

[2] O. V. Amico, Fazio, "Entanglement in Many-Body Systems," Quantum **80,** 517–576 (2008).

[3] J. P. Dowling and G. J. Milburn, "Quantum Technology: The Second Quantum Revolution,", 2002.

[4] M. Hajdušek and R. V. Meter, "Quantum Communications,", 2023.

[5] A. J. *et al.*, "Quantum Algorithm Implementations for Beginners," ACM Transactions on Quantum Computing **3,** 1–92 (2022).

[6] N. H. Nickerson, Y. Li, and S. C. Benjamin, "Topological quantum computing with a very noisy network and local error rates approaching one percent," Nature Communications 4 (2013).

[7] S. Pichai, "Our progress toward quantum error correction," https://blog.google/inside-google/message-ceo/our-progress-toward-quantum-error-correction/ (Accessed April 6, 2024).

[8] A. M. Gillam, "Designing a Map for Quantum Teleportation Protocols," Bachelor dissertation (University Brigham Young, Provo, Ut, 2022).

[9] S. Krastanov, V. V. Albert, and L. Jiang, "fimized Entanglement Purification," Quantum **3,** 123 (2019).

[10] R. Van Meter, in *Quantum Networking* (John Wiley Sons, Ltd, 2014), Chap. 9, pp. 175–194.

[11] Qiskit contributors, “Qiskit: An Open-source Framework for Quantum Computing,”, 2023.

Index