ABSTRACT

Quantum computing is an emerging technology at the forefront of science. We briefly overview the fundamental principles and history of quantum computing before discussing the unique strengths of quantum computers and demonstrating the quantum advantage in optimization problems. We show how quantum computers can effectively group individuals by their relationships. Additionally, we provide evidence of the scalability of quantum algorithms as compared to classical counterparts. We then describe how these computations can be negatively affected by noise and consider how to account for this in future use. Our analysis of noise includes the application of the Central Limit Theorem and the Empirical Rule to calculate broad bounds for future expected error, in the context of circuits consisting primarily of NOT and Hadamard gates, two fundamental operators discussed in detail. We also created an error distribution for an empty quantum circuit, centered around 1.38%, with a standard deviation of 0.395%. Finally, we examine the concepts of quantum entanglement and the No-Cloning Theorem and incorporate the two into quantum teleportation. We demonstrate how these ideas are necessary in quantum computing and in future quantum technologies.

I. INTRODUCTION

What is Quantum Computing?

Quantum computing is a developing technology that utilizes the properties of quantum mechanics, or how particles behave at the subatomic level to perform computations. While a classical computer uses bits, which hold a binary value of 0 or 1, a quantum computer uses quantum bits known as qubits (1). The quantum state of the qubit simply represents the probability that it will be a 0 or 1. Qubits can be $|0\rangle$, $|1\rangle$, or in a superposition of both states, where it is both $|0\rangle$ and $|1\rangle$. A quantum computer manipulates these qubits while they are in superposition to perform computations.

Integral to quantum computing, superposition is commonly demonstrated by the double-slit experiment, which involves shooting subatomic particles like electrons through two slits at a detecting screen (2). One might expect to see two areas of concentrated dots detected on the screen as a result of the particles going through either the first or second slit. However, an interference pattern, a series of bands with variable intensity that arises from
waves colliding with each other, is observed instead (Figure 1). The particle goes through both slits at the same time and interferes with itself. If the particle goes through both slits at the same time, it must be in a superposition. This is evidence for the wave nature of particles. This means that these particles can be described by a wave function, which is used to find the probability of finding a particle at a given point at a certain time. Interestingly, when the particles are observed as they go through the slits, the particle-like behavior returns and the interference pattern disappears. The act of observing a particle “collapses” the wave function of the particle, yielding a set state instead of a superposition of states. In quantum computing, this means the act of measuring the state of a qubit results in an irreversible loss of its state of superposition, precluding its use in future measurements.

The state of a qubit can be represented by a state vector. A state vector is represented by $|\psi\rangle$ to show the superposition of the two states. The overall state vector is described in the equation $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$. $\alpha$ and $\beta$ can be any complex number and represent the probabilities of the qubit being in state $|0\rangle$ or $|1\rangle$ respectively, such that $|\alpha|^2 + |\beta|^2 = 1$. By squaring the magnitudes $\alpha$ and $\beta$ and adding them, we should get 1, as the qubit must be observed in one of the two states.

The two-dimensional state vector of qubits can be three-dimensionally visualized with the Bloch Sphere (Figure 2). $|0\rangle$ and $|1\rangle$ are represented by state vectors pointing straight up and down, respectively.

Presently, a well-known implementation of quantum computers is superconducting qubits, used by companies such as Google and IBM (3). At extremely low temperatures (~20 milliKelvin), metals become superconductors and electrical resistance drops to zero. A Josephson Junction, a loop of two superconducting metals (often aluminum) with a very thin piece of insulating material obstructing it, will stop most electrical currents. However, due to a phenomenon called quantum tunneling, small particles have a slight chance of “tunneling” through the energy barrier and going to the other side. After quantum tunneling, particles will be in superposition, creating an artificial atom that can be used as a qubit. The qubit can be manipulated with microwaves to perform operations. When the qubit needs to be measured, the superposition of the artificial atom gets collapsed into the excited or ground state, which corresponds with a binary 1 or 0.

**History of Quantum Computing**

Quantum computing is still in its infancy. In 1981, Richard Feynman first introduced the idea of computers taking advantage of quantum mechanics to simulate nature in ways that classical computers could not (4). Peter Shor then demonstrated a more concrete application in 1995 when he proposed a quantum algorithm that in theory would easily factor extremely large numbers (5). Quantum computing became a reality in 1998 when Dr. Isaac L. Chuang, Dr. Gershenfeld and Dr. Mark G. Kubinec created the first quantum computer that had two qubits (6).
In 2017, Chinese scientists made a breakthrough in quantum teleportation, which is the transfer of exact quantum states from one particle to another without the particle physically traveling across space (7). They took advantage of quantum entanglement, a phenomenon where two particles in superposition are paired in such a way that observing one will affect the state of the other. In the experiment, one of the entangled particles was on Earth and the other was in low Earth orbit, breaking the record for the farthest quantum teleportation distance. In 2019, Google was the first to claim that they had achieved quantum supremacy over classical computers by solving in three minutes and twenty seconds a problem that would have potentially taken a classical computer ten thousand years (8).

**Advantages and Disadvantages**

Since quantum computers use qubits, they can be significantly faster than classical computers. Quantum computers could theoretically take only seconds to perform calculations that would take classical computers thousands of years (8). They can analyze large amounts of data and pick up on patterns that classical computers might miss. Because quantum computers use qubits that are in superposition, they can analyze multiple states at once, enabling them to perform calculations faster.

Despite the plethora of advantages conferred using quantum computers, several drawbacks have hindered progression into mainstream use. For example, current technology only enables qubits to have a lifetime of 0.5 milliseconds—the quantum state is not maintained for very long and the qubits cannot be utilized in the future, since they have been reduced to classical bits. Quantum computers undergo decoherence when they lose their quantum state. Decoherence can cause noise or error in the results of a quantum computation. To reduce noise, quantum computers are cooled to temperatures approaching absolute zero. Although cooling can reduce noise, noise remains a prominent issue. Additionally, quantum computers are difficult to build, as a full quantum computer would need to be composed of thousands of qubits, which are extremely difficult to build (3). The qubits need to be initialized in a known state, and universal gates used in classical computing should apply. These difficulties make quantum computers expensive to both build and maintain.

**How Quantum Computers Are Programmed**

Quantum computers can be programmed by using a combination of classical and quantum gates. The classical gates most often include the NOT gate, the CNOT gate, and the Toffoli gate (Figure 3). The NOT gate flips the value from $|0\rangle$ to $|1\rangle$ or from $|1\rangle$ to $|0\rangle$. The CNOT gate has a control and a target. If the control is $|1\rangle$, the CNOT gate performs a NOT on the target, and if the control is $|0\rangle$, the target does not flip. The Toffoli gate works in a similar way to the CNOT gate, except it has two or more controls. A commonly used quantum gate is the Hadamard gate, which puts the qubit into a superposition between $|0\rangle$ and $|1\rangle$, making it...
impossible to know which state it is in until it is measured. All gates apply a rotation to the state vector by multiplying it with a matrix, which can be visually represented by the Bloch Sphere.

The NOT, CNOT, and Toffoli gates can be used to create a variety of other gates, such as the OR gate (Figure 3B) and the AND gate (Figure 3C). The OR gate uses the CNOT and Toffoli gates to change the output to $|1\rangle$ if either of the two inputs are $|1\rangle$. In Figure 3B, q[0] and q[1] (both representing qubits) are the inputs and q[2] is the output. The first CNOT gate checks if q[1] is true or false, the second CNOT gate checks if q[0] is true or false, and the Toffoli gate checks if both are true. If either one or both of the inputs are true, the output will be equal to 1. The AND gate switches the output to 1 only if both of the inputs are true. The state of q[2] starts as zero, and if both q[0] and q[1] are true, q[2] will also be true. The Toffoli gate checks the state of q[0] and q[1].

II. RESULTS: OPTIMIZATION

Taxi Problem

Consider a puzzle where three people, Alice, Bob, and Charlie, must fit in two taxis, where each taxi has room for only two passengers. The relationships between the three people will influence how desirable a certain grouping will be. Suppose Alice and Bob are friends; putting them together in one taxi is favorable. Similarly, Bob and Charlie are friends, so any taxi that holds both is favorable. However, Alice and Charlie are enemies (Figure 4), so putting them together is unfavorable (9). With the conditions set, we can now start devising a solution.

In order to solve this problem efficiently, we first need another way to store quantum data. As mentioned earlier, a qubit’s state can be represented by a sphere with the $|0\rangle$ and $|1\rangle$ states on opposite ends of the Z-axis. A Hadamard gate sends a default qubit in the $|0\rangle$ state into a superposition by moving it to the “right” $|\rightarrow\rangle$ state on the X-axis, exactly between the $|0\rangle$ and the $|1\rangle$ states, perpendicular to the Z axis (10). We can visualize this rotation on the Bloch Sphere (Figure 5).

The taxi problem requires manipulating a qubit by rotating it around the Y-axis using an RY gate. These rotations can affect whether qubit favors one state or another. We can visualize this change by imagining a view of the XZ plane with the Y-axis pointing away from us. From the “right” state, a positive rotation (clockwise around the Y-axis) will turn the qubit towards the “down,” or $|1\rangle$ state, while a negative

---

Figure 4: A visualization of the relationships between the three passengers: Alice (A), Bob (B), Charlie (C). A green line connects friends and a red line connects enemies.

Figure 5: The Hadamard gate’s effect on a qubit in its default state. The state of the Hadamard is represented by the white dot on the surface of the sphere.
rotation turns the qubit toward the “up,” or |0⟩ state. A rotation angle of π/2 will fully turn the qubit from one pole to the other and going beyond that will start returning the qubit to a position between |0⟩ and |1⟩ again. The closer the qubit is to the |0⟩ or |1⟩ state, the greater the probability of measuring a 0 or 1, respectively. This rotation technique is useful because it allows us to store incremental pieces of information as rotations on a qubit. We will use this technique to track the “favorability” of each combination, which is the total number of friend pairs minus the total number of enemy pairs in a combination.

Three qubits are used to represent each person, q[0] for Alice, q[1] for Bob, and q[2] for Charlie. The car that a person is in is represented by the measurement of their qubit: a 0 is the first taxi, and 1 is the second. We will need a fourth qubit q[3] to represent the favorability of the combination. A |1⟩ for q[3] represents a favorable combination, with more friends together and more enemies apart, and a |0⟩ represents an unfavorable combination.

In order to compute combinations, we first need to apply a Hadamard gate to put all three person qubits into an equal superposition of |0⟩ and |1⟩, creating all the possibilities for which taxi each person is in. Next, we check each two-person combination with a CNOT gate and a NOT gate, together forming an XNOR gate, which will return a 1 if the two people are in the same car and a 0 if they are in different cars. Since we need the original state of the qubits to check the existence of other pairs and then measure at the end to actually know the combinations processed, we insert another NOT gate and CNOT gate to reverse the changes we made (Figure 6). Next, we apply a Hadamard gate to the q[3] qubit, rotating it to the |+⟩ state. Using a controlled RY gate, we rotate q[3] if the XNOR gate returns an outcome of 1. If the two people that are together are friends, we rotate the qubit in the positive direction towards the |1⟩ state, and if they are enemies, we rotate the qubit in the negative direction towards the |0⟩ state. Finally, we have to account for the scenario in which all three qubits are the same, corresponding to all three people being in the same taxi. This would result in two positive rotations and one negative rotation for a net positive rotation, but since it is forbidden by the rules, we want it to result in an unfavorable outcome of 0. To do this, we insert a Toffoli gate to check if all three passengers are in taxi 1, and if they are, we flip q[3] from a |1⟩ to a |0⟩. We then apply a NOT gate to qubits 0, 1, and 2 and use a triple-checking Toffoli gate to check if all three people are in taxi 0. A final column of NOT gates returns q[0], q[1], and q[2] to their original post-Hadamard states. We can finally measure all four qubits, collapsing the superposition to
give us a graph of favorable and unfavorable solutions. The full quantum composer code can be seen in Figure 7.

In Figure 8, we can see that the combinations that are more favorable all start or end with the same digits, representing Alice and Bob together or Bob and Charlie together. The two combinations with Alice and Charlie together are heavily unfavorable, as well as the two combinations forbidden by the rules, 000 and 111. Note that since we rotated q[3] by just $\pm\pi/3$ radians rather than the full $\pm\pi/2$ to $|1\rangle$ or $|0\rangle$ respectively, it still has a small component in the $|1\rangle$ direction that appears in the graph as the nonzero favorability amounts for the unfavorable arrangements. We have now visually verified the data to be correct.

Due to the simplicity of this problem, solving it with classical computers or even manually is relatively easy. However, the quantum circuit is still theoretically faster. Since we have the three qubits in superposition, all possibilities are calculated at once, unlike classical computers which can only go one combination at a time.

**Extending the Taxi Problem: Bus Problem**

To complicate the taxi problem, a fourth person is added—David. The taxis are traded for buses that can hold any number of passengers. The relationships become more complex: Alice is enemies with both Bob and Charlie, Bob is friends with Charlie and David, and David is a friend of Alice and an enemy of Charlie (Figure 9).

Part of the advantage of quantum computers is scalability; the problem can be tackled in a similar manner to before. Each individual is associated with one qubit that is put into superposition with a Hadamard gate. Each pairing of individuals is checked to determine if they share a vehicle. This is done by applying an XNOR gate to every possible combination of passengers—in this case, six. If the combination is deemed favorable, the “favorability” qubit is rotated $\pi/4$ radians towards $|0\rangle$. Otherwise, the qubit is rotated $\pi/4$ radians towards $|1\rangle$ (Figure 10). This will make the qubit more likely to output $|0\rangle$ or $|1\rangle$ when measured.
The results are shown in Figure 11. As mentioned before, the value of each digit determines if Alice, Bob, Charlie, or David, respectively, are in either bus 0 or 1. For example, the combination “1010” notes that Alice and Charlie are in one taxi, and Bob and David are in the other. The frequency shows the number of times the “favorability” qubit was measured as 0 or 1. The most favorable combinations involve Alice and David sharing one bus, with Bob and Charlie sharing the other. The least favorable combinations involve Alice and Bob sharing a bus while Charlie and David share the other. The results also show partial favorability, allowing one to not just list the combinations of passengers, but effectively rank them.

The true importance of these results is not the specific results, rather the knowledge that the quantum computer running this code is processing every combination simultaneously. In theory, one could continue to add passengers to the problem with even more complex relationships and return similar results just by adding more XNOR gates and distinct rotations of the “favorability” qubit. As more advanced quantum computers are constructed, the only limiting factors are the number of qubits available and the amount of subsequent error. Bear in mind that the results provided for both the taxi problem and the bus problem are from a quantum computer simulator, as noise (discussed below) and other forms of error can skew the results.

Comparison to a Classical Approach

To give a clearer idea of how much more scalable the use of quantum computing for such optimization problems is, let us consider a classical computer’s approach to the Taxi and Bus problems (reference Appendix I for code).
In computing, the efficiency of an algorithm is often measured in terms of how the number of operations needed to be performed increases with the size of the input. In ascending order of rate of increase, this could be linear, where the number of operations scale proportionally with the input; quadratic, where they scale with the square of the input; or exponential, where the number of operations increases exponentially as the input size increases; and so on. The rapidly accelerating nature of exponential increase means that exponential-order algorithms are generally unfavorable, as the number of operations can easily become too large for computers to feasibly handle.

For the aforementioned taxi problem, the size of input is given by the number of people involved. For the computer to calculate favorabilities, it has to check each pair of people for each combination: these are our operations. Unlike a quantum computer, a classical computer can only operate on one possibility at a time: iterating through them, checking the pairs for each one by one. Each of the 3 people have 2 options: either in vehicle 1 or vehicle 0, making for $2^3 = 8$ possibilities covered over three nested loops. When a fourth person is added, we have to account for their two options for each of the existing eight arrangements with another nested loop, doubling the total number of possibilities to make $2^4 = 16$ possibilities that the computer must iterate through and check pairs of. Every time a new person is added to the input, the number of possibilities it operates on is multiplied by two; the number of operations increases exponentially with the input size.

In the quantum solutions, instead of iterating through each possibility, we can set the people qubits into a superposition of every possibility at once by placing a Hadamard gate in front of each of them. The operations we do to check pairs on the superposition happens to each of its component parts—each possibility—simultaneously. To go through the new possibilities when increasing from three to four people, we need only add one more qubit with one more Hadamard gate instead of having to double the computations by adding a further nested loop, though of course there are now 6 pairs of people to check instead of 3. The number of operations will only depend on the number of pairs of people that need to be checked for those many people, given by $C = \frac{n(n-1)}{2}$, where $n$ is the number of people, meaning it only scales quadratically with the input size. As discussed earlier, this is much more efficient than the exponential classical algorithm.

Therefore, while a classical computer may take exponentially longer to solve an optimization problem as the number of variables—and in turn the number of possibilities to iterate through—increases, a quantum computer can handle all possibilities simultaneously, and so quickly gain the advantage for more complex problems.

III. RESULTS: NOISE

Introduction to Noise

Unwanted variations when performing quantum computations, termed noise, is fundamental in quantum computing due to the instability of current qubit designs (11). The current qubit only lasts half a millisecond before it loses its quantum effect (12). Quantum effects
of qubits mainly vanish due to entanglement with the outside environment, which complicates their use in computations because interactions with the environment are necessary to run programs and obtain outputs. The entanglement with the surrounding environment is referred to as environmental decoherence. The gradual loss of information and entanglement due to the lack of isolation can occur in the presence of stochastic electromagnetic fields, defects in the material, vibrations, and fluctuations in temperature (13, 14). Despite the careful engineering of the quantum computer with careful control of the surroundings, such as the use of physical substrates and minimizing temperature to isolate the qubits, the interactions still exist (15). Although decoherence would not occur if the qubits were left completely isolated, making measurements and performing calculations using those qubits would be impossible.

Another cause of noise is coherent quantum errors. The coherent errors are a result of imprecise controls of the qubits, namely finite resolution of measurements (16). Transpiling the code—the transformation from the coding language to a language that the quantum computer can understand—changes a certain gate to commands involving a finite number of significant figures. For example, the transpiled code for a Hadamard gate involves two RZ gates with a rotation of 1.5707963267948966 radians (which is roughly, but not exactly, 90 degrees). Furthermore, the control signal quality and qubit sensitivity can make the actual rotation to be slightly off. The small error in the rotations, especially when compounded by multiple gates, can lead to large errors or an incorrect output.

There are other sources of error as well, such as qubit initialization, loss, and leakage. Qubit initialization is the process of setting up and creating the qubit; when the qubits are initialized, there may be variation in the state of the qubits. Qubit loss is the physical loss of a qubit from a system, where the qubit is no longer inside the quantum register. Loss can occur when there is an absorption of particles like an atom or an ion into the walls of a quantum computer. This makes measurement of the qubit impossible. Quantum leakage occurs as a result of the qubit system containing more than two electronic levels, although the qubit itself should only have two states. Imprecise control of the system can result in some qubits ending up in higher energy levels (16). Finally, there are measurement errors, where the recorded output is different from the actual output. Even if a quantum computer performs all the computation correctly, the measurement can still result in errors; an output of 1 may be incorrectly read by the quantum computer when the collapsed state is actually a 0. For example, the average measurement error for quantum machines can be 6% to 8% (17).

Methods of Analysis of Noise Trends and Distributions

Error distributions and trends of quantum noise were investigated using IBM’s Quantum Lab, with Qiskit. Scripts were written in Quantum Lab and run directly, using Qiskit API credentials to connect to a remote quantum computer, IBMQ Manila. All figure error bars represent 2 times standard error and normal distributions were fit directly to histogram bar data.

Distribution of Error for Empty Quantum Circuits

A preliminary analysis of noise for an empty quantum circuit resulted in a distribution centered around 1.38% error for each run (14.15 errors for each 1024 trials, on average). In order
to obtain a distribution of error, 40 different runs were conducted, in order to take advantage of the Central Limit Theorem, which asserts that a sampling distribution must be approximately normal for a sufficiently large number of samples, n. The standard deviation of the distribution was calculated to be approximately 0.395%. Thus, according to the empirical rule, for variability in normal distributions, roughly 99.7% of empty circuit quantum runs will result in error within 3 standard deviations of the mean, or between 0.198% and 2.566% (Figure 12). Therefore, it is likely that, even without the presence of any actual gates or operations, over 1% of quantum computing results are in fact inaccurate, providing a baseline for future operations.

**Error Distributions for Small Quantum Circuits of Hadamard Pairs and NOT Gates**

Running NOT gates on a quantum computer consists of rotating the qubit in a specific direction, generating the opposite value. Like all quantum computing operations, however, NOT gates are susceptible to noise. Once again, 40 jobs were run for circuits of 1, 2, 3, and 4 sequential NOT gates, in order to attempt to characterize error and provide reasonable bounds. Overall, there seems to be higher error for those combinations of NOT gates that are supposed to yield a 1 result (i.e. odd numbers of gates) as compared to those designed to yield a 0 result (i.e. even numbers of gates), with no overlapping between the two sets of distributions. Similarly, error actually appears to decrease as additional pairs of NOT gates are added, most prominently shown in the distributions of odd numbers of NOT gates (Figure 13).

The error distribution for 1 NOT Gate is centered around 5.12%, with 3 standard deviation bounds of 3.08% to 7.15%. Similarly, the 3 NOT Gate circuit error is centered around 4.26%, with bounds of 2.07% to 6.45%. These
error measurements stand in stark contrast to the same metrics for 2 and 4 sequential NOT Gate circuits, with centers at 1.62% and 1.47% respectively, and bounds of 0.25% to 2.99% and 0.330% to 2.56% (Figure 14). Therefore, it is likely that quantum error for circuits constructed from odd numbers of sequential NOT gates averages around 5%, while averaging only around 1-2% for circuits of even numbers of sequential NOT gates.

Quantum Hadamard gates can act as their own inverses. With that in mind, a Hadamard gate pair has no theoretical net effect on a qubit value, although it may contribute to additional error. In order to analyze how increasing numbers of Hadamard gate pairs contribute to quantum error, 40 jobs were run for combinations of 1, 2, and 3 Hadamard gate pair circuits. No difference was ultimately found between the centers of the error distributions for the increasing combinations of gates, although variability may actually decrease, if only slightly, as the number of gates is increased. All distributions were centered around 1.3%-1.4%, with plausible error values ranging below 5 to almost 30 counts, for every 1024 iterations (Figure 15).

The three distributions were centered at 1.36%, 1.37%, and 1.41% for 1, 2, and 3 pairs of Hadamard gates respectively (Figure 15). Furthermore, according to the empirical rule for values within n standard deviations of the mean in normal distribution, almost all calculations involving combinations of 1, 2, or 3 Hadamard gate pairs will likely yield error between 0 and ~3% maximum.

Error Trends for Large Quantum Circuits of NOT and Hadamard Gates

In order to better understand how increasing numbers of NOT gates in sequence affect quantum error, more runs were conducted, testing circuits of increasing numbers of NOT gates, on intervals of 5. Results once again varied in relation to the expected value of the output: $|1\rangle$ for sequences of odd numbers of NOT gates and $|0\rangle$ for sequences of even numbers of NOT gates (Figure 16). Separate trends exist for both models, with error starting out higher and increasing slower for circuits with expected value $|1\rangle$ than for circuits with expected value $|0\rangle$. Between 50 and 60 NOT gates in sequence, however, the error from the even sequences of NOT gates (expected value $|0\rangle$) actually overtakes or at least matches error from the odd sequences of NOT gates (expect value of $|1\rangle$), suggesting some kind of potential convergence at

![Figure 15: Error trends for NOT gate sequences of increasing lengths. One error measurement was taken for each circuit of n number of sequential NOT gates, represented on the x-axis.](image)

![Figure 16: Error trends for Hadamard gate pair sequences of increasing lengths. One error measurement was taken for each circuit of n number of sequential Hadamard gate pairs, represented on the x-axis.](image)
Later point, or divergence, as a horizontal asymptote forms for odd sequences and a vertical asymptote forms for even sequences (Figure 16). More exploration would be necessary to determine the specific shapes of these trends with certainty.

An error trend for increasing sequences of Hadamard gate pairs was also examined, yielding a relatively strong positive relationship. As more sequences of Hadamard gate pairs were added to the circuit, error generally seemed to increase, perhaps as a result of accumulation and compounding over greater and greater number of operations (Figure 14). Several outliers may be present in the data, although they do not destroy the overall trend. We suspect that the trend is most likely linear, but may be skewed due to several outlier values.

Noise Reduction Strategies

With all this in mind, it is imperative that some amount of error correction is employed to extract any useful computations from noisy quantum computers. Classically, error correction is done by using parity checks, which involves cloning a single classical bit over multiple bits to check how much error is introduced after a computation (Figure 17). While this approach can be employed for some quantum computing applications, in order to have a full quantum error correction (QEC) code, any correction must account for both a bit flip and phase flip instead of only a bit flip (16).

Error Corrected: 
Cloned Bit: 0
Cloned Bit: 0
Cloned Bit: 0

However, it is difficult to directly adapt this classical parity method to quantum computing because of the inability to clone quantum bits due to the No-Cloning Theorem, discussed in a later section (12). Instead of directly copying, the qubits must be entangled. Entangling them makes sure that error can be measured and corrected for without actually measuring the state of the data bit. This approach is what is taken with the three-bit Shor code (12). It uses three qubits to correct for one bit flipped qubit (16).

Another problem presented is the fact that quantum error correction must compensate for both bit flips, from a 0 to 1, and phase flips, an incorrect rotation around the Bloch Sphere, rather than just bit flips. The previous Shor code mentioned is not a full QEC because using 3 qubits only corrects for one at a time. To realize full quantum error correction a basic nine qubit code like the 9-Bit Shor code needs to be employed (16). A code like this uses nine qubits to quantum correct a single qubit from either one qubit flip, phase flip, or one of each. However, this corrects for only a single qubit. As more and more qubits are needed for error correction, less qubits are usable for computation.

Popular error correction methods employ topological codes that rely on the arrangement of qubits to correct quantum errors (18). One topological code is the surface code where qubits are arranged in a 2D grid with each error corrected bit surrounded by four “ancilla” bits (19).
(Figure 18). By measuring the ancilla bits you can extract how much error the data bit has accumulated.

While the above methods aim to correct errors directly, there are other methods that instead deal with the decoherence time to correct errors. One way of doing this is by decreasing the time a quantum algorithm has to run. This can be done by utilizing a hybrid of classical and quantum computers for necessary sections of an algorithm (13). This would help the problem of noise, but introduce hardware that would slow down overall computations.

These are only a few error correcting methods under development. As QEC advances, the full potential of quantum computers can be realized.

IV. QUANTUM ENTANGLEMENT AND TELEPORTATION

Quantum Entanglement

An entangled pair of particles can be represented mathematically as a single wave function and, when absent from observation, the particles exist in a superposition of states (20). Measuring one particle collapses the collective wave function, and observing a property of one entity in the pair instantaneously affects the property of the other after the superposition collapses—faster than the speed of light. A common example of this entails spin: a quantum property of a particle that describes an inherent angular momentum (21). The spin of a particle could have various orientations, and collapsing the superposition of an entangled pair by measuring the spin of one entity would produce a predictable spin for the other particle.

EPR Paradox

When quantum mechanics was conceived in the early 20th century, certain classical physicists like Albert Einstein expressed their disapproval with certain principles of quantum theory that seemed to violate the bedrock postulates of classical physics (21). In the famous Einstein-Podolsky-Rosen (EPR) paradox, some of the perceived limitations of quantum physics were elucidated, and the authors proposed the then-theoretical system that became known as quantum entanglement. To Einstein, this instantaneous “spooky action at a distance” infringed on a concept that was fundamental to his theories of relativity: locality, which states that objects can only affect their immediate surroundings because signals cannot travel faster than the speed of light. Entanglement flouts this principle by allowing for immediate communication between entities with no intermediate travel of information between the two. Uncomfortable with this description of subatomic reality, EPR’s authors proposed that entangled particles were encoded
with so-called “hidden variables” that already contained information about what state the particle would exhibit upon a wave function collapse triggered by the measurement of one particle, eliminating the need for faster-than-light communication (22).

Bell’s Theorem

In 1964, John Stewart Bell published a paper that proposed a mechanism to experimentally determine whether Bohr’s view of quantum mechanics or Einstein’s hidden variables view would prove successful (22). In one variation of the Bell test, two entangled photons are sent to two different detectors that can only be reached if the photons have a particular spin. Bell computed a set of mathematical inequalities that, if violated, would prove that quantum entanglement could not be explained by hidden variables embedded within the particles (23). Numerous experiments have demonstrated that Bell’s inequalities are consistently violated in these types of tests and that entangled pairs do in fact influence each other faster than the speed of light, suggesting that principles sacrosanct to relativity, like locality, may not apply in quantum entanglement (24).

Quantum Eraser

In another notable experiment, dubbed the delayed-choice quantum eraser, the double-slit setup is modified to assess the extent of the interconnectedness of entangled pairs (24). In this model, entangled pairs of photons are sent through a double-slit, producing the familiar interference pattern that quantum mechanics predicts based on wave-particle duality. However, if one particle is sent to a detector that can determine which slit it went through, and its entangled twin is sent towards a wall, the interference pattern disappears because of the mere observation of one entity in the pair. Even if the detector is placed farther away from the double slit than the wall where an interference pattern is expected, the pattern still disappears upon measurement of just one of the particles, suggesting a kind of retroactive influence. In yet another modification, if one of the particles is sent in the direction of the detector, but with its path scrambled in such a way that it is impossible to know through which slot it went, the interference pattern reemerges with the entangled twin. Even here, if the detector is placed farther away than the wall, the interference pattern manifests itself in what appears to be retroactive communication from the particle heading towards the detector. The instrumentation that makes it impossible to measure the path of one photon allows the other photon to remain in a superposition and produce the interference pattern that quantum mechanics predicts.

Entanglement in Quantum Computing

The process of quantum entanglement is integral to quantum computing (19). Figure 19 depicts a simple circuit that produces an entangled pair of qubits with a Hadamard gate to put q[0] in a superposition of states and a CNOT gate to equate the value of q[1] to q[0]. The entangled qubits in superposition are known as Bell states. Feeding the vector |00⟩ into the circuit produces the superposition |00⟩ + |10⟩ after the Hadamard gate, which transforms

![Figure 19: Bell pair producer circuit. Two qubits are entangled using a Hadamard and a CNOT gate. The value of q[1] is dependent on the value of q[0].](image)
into |00⟩ + |11⟩ after the CNOT gate. The following are the four possible Bell pairs that can be generated from this gate:

\[
\begin{align*}
H|00⟩ &= |00⟩ + |10⟩ \quad → \quad \text{CNOT}(|00⟩ + |10⟩) = |00⟩ + |11⟩ \\
H|01⟩ &= |01⟩ + |11⟩ \quad → \quad \text{CNOT}(|01⟩ + |11⟩) = |01⟩ + |10⟩ \\
H|10⟩ &= |00⟩ - |10⟩ \quad → \quad \text{CNOT}(|00⟩ - |10⟩) = |00⟩ - |11⟩ \\
H|11⟩ &= |01⟩ - |11⟩ \quad → \quad \text{CNOT}(|01⟩ - |11⟩) = |01⟩ - |10⟩
\end{align*}
\]

The process of entanglement is central to communication applications of quantum computing, including quantum teleportation.

**No-Cloning Theorem**

In addition to quantum entanglement, an integral aspect of quantum teleportation is the concept of the No-Cloning Theorem. However, it is essential to establish the definition of cloning under the lens of physics and mechanics (27). In this regard, cloning is the exact replication of an unknown substance’s characteristics, including its particle interactions, positions, and energy. In the case of quantum computing, however, cloning would serve as a vessel to measure one qubit copy while still preserving the original qubit for other purposes. For example, it would be convenient to have multiple copies of a qubit in which a program could run through and evaluate its state at different parts and times (28). This is significant because since qubits are in a superposition, the measurement of the qubit collapses the state to either a 0 or 1. As a result of this characteristic or property, it is difficult to determine and measure the very probability of that qubit becoming a 0 or 1. Likewise, cloning would be perceived as a viable alternative because of the opportunity to have one exact copy of a qubit that can be measured, while simultaneously having another identical qubit to conduct other functions. Furthermore, the No-Cloning Theorem prevents the existence of two identical copies because the original is destroyed. Moreover, the No-Cloning Theorem has been proven to be impossible as a result of a mathematical proof that reveals a contradiction in the attempt to clone the state of a particle or in qubit (full proof found in Appendix II.I).

Putting it simply, the contradiction shown on the math proof reveals that perfect cloning does not exist because cloning the entirety of an qubit is not equal to the cloning of the sum of its parts (information on Properties of Particles/Qubits Applied in No-Cloning Theorem are found in Appendix II.1). Suppose there was a qubit |ψ⟩ in a superposition of α|0⟩ + β|1⟩ and that there was a unitary operator that would clone |ψ⟩|0⟩ in its entirety and in its individual parts into |ψ⟩|ψ⟩. Likewise, cloning |ψ⟩ or (α|0⟩ + β|1⟩) should be equal to cloning α|0⟩ + β|1⟩ separately because of the transformation property, which states that transformations to a particle in superposition are distributed to all particle states independently. However, (α|0⟩ + β|1⟩)², which equates to the clone of the object’s entirety is not equal to (α|0⟩)² + (β|1⟩)², which is the cloning of the individual parts or superposition. As a result, the No-Cloning Theorem rejects the idea of being able to create a perfect clone of an unknown object or substance, however, it does not conflict with the following concept of quantum teleportation.
Quantum Teleportation

In quantum teleportation, Alice the sender wishes to convey information of an unknown quantum state of $q[0]$, represented by $(\alpha|0\rangle + \beta|1\rangle)$ (29). However, due to the No-Cloning Theorem, Alice cannot simply duplicate the qubit. Instead, Alice can send information to Bob about how he could change the state of his own qubit. This is a process known as quantum teleportation, which can be modeled by the circuit to the right (Figure 20 for circuit and Appendix II.3 for Python code).

This is done by entangling two qubits together and giving one to Alice ($q[1]$) and Bob ($q[2]$) (29). The two qubits are entangled together as a Bell pair by using the Hadamard and CNOT gates. Alice applies a CNOT gate to $q[1]$ and a Hadamard gate to $q[0]$ to put it in a superposition of the $|0\rangle$ and $|1\rangle$ states. She then measures $q[1]$ and $q[0]$ and sends these measurements to Bob, who will use them to understand how to change his quantum state. Depending on which bits Alice sends, the conditional NOT gates and Z gates will fire to change Bob’s qubit. The final measuring tool measures $q[2]$ for the user to verify that Bob’s qubit has the same state as $q[0]$. Because $q[0]$ was measured in the process of quantum teleportation, it is destroyed, which follows the No-Cloning Theorem (See Appendix II.4 for the mathematical derivation of the teleportation algorithm).

After building this circuit, we have gotten the following results confirming that quantum teleportation was successful:

<table>
<thead>
<tr>
<th>Input (State of $q[0]$)</th>
<th>Output (State of $q[2]$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$</td>
<td>0\rangle$</td>
</tr>
<tr>
<td>$</td>
<td>1\rangle$</td>
</tr>
</tbody>
</table>

Table 1: Results of quantum teleportation circuit.

Quantum Teleportation Experiments

One of the earliest experiments on quantum teleportation entitled was conducted in 1997 at the University of Innsbruck, Austria, by Bouwmeester et. al (30). The experiment involved polarizing an initial photon at 45° and entangling it with another photon. That other photon was also entangled with another photon set a distance away. The experiment found that the photon was “teleported”—by taking in certain measurements, the photon was able to alter its state to match the polarization of the initial photon (Figure 21).
Another experiment was conducted in the Canary Islands of Tenerife and La Palma by Herbst et al. in 2015 to test whether quantum teleportation can be used over a distance of 143 km (31). The experiment involved the entanglement of photons and followed similar concepts of quantum teleportation. Two photons were entangled and one of them was sent from La Palma to Tenerife with a transmitter telescope. At La Palma, the remaining photon was entangled with another one. By passing certain measurements, the photon changed into the same version of the original one at La Palma.

In 2020, a team of researchers from the University of Science and Technology of China and the University of Geneva were able to perform three-dimensional quantum teleportation of the qutrit state of a photon (31). After taking a three-dimensional Bell state measurement, the team focused an ultraviolet laser on β barium borate crystals to produce three photon pairs.

**Quantum Cryptography**

An enticing application of quantum entanglement is the potential to generate unhackable communication techniques (25). A sender Alice and a receiver Bob each have their own sets of qubits. Since quantum teleportation never requires Alice to measure her qubits, she will only send specific instructions to Bob without collapsing the superposition of her qubits. Anyone who seeks to hack transmission would be unable to read the information of Alice’s qubits because they are superimposed. Neither Alice nor Bob themselves know the states of their qubits, making this system theoretically unhackable.

**CONCLUSION**

By explaining and solving an optimization problem, and later expanding its scope, we were able to demonstrate the superior scalability of quantum algorithms as compared to classical algorithms. Despite its potential, quantum computers are still limited by the prevalence of noise in calculations, although new error correction algorithms are in development. Quantum teleportation illustrates a fascinating application of quantum entanglement as well, with potential to shape the field going forward. As companies and governments continue to invest in quantum computing to capitalize on its immense potential, it may be possible for this emerging technology to eclipse classical computers and achieve broad quantum supremacy.

**REFERENCES**

   https://www.ncbi.nlm.nih.gov/books/NBK538701/
2. Do atoms going through a double slit ‘know’ if they are being observed? Physics World. 2015 May 26 [accessed 2022 Jul 27].
   https://physicsworld.com/a/do-atoms-going-through-a-double-slit-know-if-they-are-being-observed/
   https://www.npr.org/2017/07/14/537174817/scientists-teleport-a-photon-into-space
   https://betterprogramming.pub/what-is-a-quantum-computer-explained-with-a-simple-example-b8f602035365
    https://bits-and-electrons.github.io/bloch-sphere-simulator/
    https://www.quintessencelabs.com/blog/current-quantum-computers-noisy/
    https://spectrum.ieee.org/heres-a-blueprint-for-a-practical-quantum-computer
APPENDICES

Appendix I - Code for the solutions to the Taxi and Bus problems.

Classical Python code for the three-person and four-person taxi problem:

#3 Person Taxi Problem: Classical Approach
from itertools import combinations

relationships = [[0, 1, -1], [1, 0, 1], [-1, 1, 0]]

#Iterate through every arrangement of the three people in the two taxis
for i in range(2):
    for j in range(2):
        for k in range(2):
            people = [i, j, k]
            favorability = 0
            #Check the pairings and calculate the effect on favorability
            for x, y in combinations(range(3), 2):
                if people[x] == people[y]:
                    favorability += relationships[x][y]
                    #If everyone is in the same taxi, set favorability to 0
                    if (people[0] == people[1] and people[1] == people[2]):
                        favorability = 0
            print(people, favorability)

#4 Person Bus Problem: Classical Approach
from itertools import combinations

relationships = [[0, -1, -1, 1], [-1, 0, 1, 1], [-1, 1, 0, -1], [1, 1, -1, 0]]

#Iterate through every arrangement of the four people in the two buses
for i in range(2):
    for j in range(2):
        for k in range(2):
            for l in range(2):
                people = [i, j, k, l]
favorability = 0
#Check the pairings and calculate the effect on favorability
for x, y in combinations(range(4), 2):
    if (people[x] == people[y]):
        favorability += relationships[x][y]
    print(people, favorability)

---

**Code for the Four-Person Taxi Problem with the Qiskit Python Library:**

#4 Person Taxi Problem: Quantum Approach

relationships = [[0, -1, -1, 1], [-1, 0, 1, 1], [-1, 1, -1, 0], [1, 1, -1, 0]]
qc = QuantumCircuit(5, 5)

#Put Alice, Bob, Charlie, David in superposition.
#This single command produces every arrangement of the four people in the two taxis
qc.h(range(0, 4))

#Move the Favorability qubit to an initial superposition exactly between 0 and 1: neutral or zero satisfaction
qc.h(4)
for x, y in combinations(range(4), 2):
    qc.cx(x, y)
    qc.x(y)
    #If there is a combination of friends in the same taxi, rotate the Favorability qubit down towards the 1 state. If they are enemies, rotate it down towards the 0 state
    qc.cry((np.pi/4)*relationships[x][y], y, 4)
    qc.x(y)
    qc.cx(x, y)

qc.measure_all(add_bits=False)
qc.draw()

backend = Aer.get_backend('qasm_simulator')
job = execute(qc, backend, shots = 10000)
result = job.result()
counts = result.get_counts()
plot_histogram(counts)

Appendix II.1 - Properties of Particles Applied in No-Cloning Theorem

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Individual particles can be in superpositions (0 or 1)</td>
<td>$</td>
</tr>
<tr>
<td>2. Composite Systems: groups or combinations or particles are products of their components or sums of products of their components</td>
<td>$</td>
</tr>
<tr>
<td>3. Transformations to a particle in superposition are distributed to all particle states independently</td>
<td>$\alpha</td>
</tr>
</tbody>
</table>

Table A.II.1: Properties of Particles Applied in No-Cloning Theorem.

Appendix II.2 - Proof of No-Cloning Theorem

**Proof:** Suppose there exists a unitary operator $U_{cl}$ that can clone an unknown quantum state $|\psi\rangle = \alpha |0\rangle + \beta |1\rangle$. Then:

$$|\psi\rangle |0\rangle \xrightarrow{U_{cl}} |\psi\rangle |\psi\rangle = (\alpha |0\rangle + \beta |1\rangle)(\alpha |0\rangle + \beta |1\rangle) = (\alpha^2 |00\rangle + \beta \alpha |10\rangle)(\alpha \beta |01\rangle + \beta^2 |11\rangle)$$

Using $U_{cl}$ to clone the expansion of $|\psi\rangle$ shows that

$$T(\alpha |0\rangle + \beta |1\rangle) |0\rangle = \alpha |00\rangle + \beta |11\rangle$$

$$\alpha |00\rangle + \beta |11\rangle \neq \alpha^2 |00\rangle + \beta \alpha |10\rangle + \beta^2 |11\rangle$$

Appendix II.3 - Python Code for Quantum Teleportation circuit on Qiskit

```python
from qiskit import QuantumRegister, ClassicalRegister, QuantumCircuit
from numpy import pi

def qreg_q = QuantumRegister(3, 'q')
creg_c = ClassicalRegister(1, 'c')
```
creg_d = ClassicalRegister(1, 'd')
circuit = QuantumCircuit(qreg_q, creg_c, creg_d)

circuit.h(qreg_q[1])
circuit.cx(qreg_q[1], qreg_q[2])
circuit.cx(qreg_q[0], qreg_q[1])
circuit.h(qreg_q[0])
circuit.measure(qreg_q[1], creg_d[0])
circuit.measure(qreg_q[0], creg_c[0])
circuit.x(qreg_q[2]).c_if(creg_d, 1)
circuit.z(qreg_q[2]).c_if(creg_c, 1)
circuit.measure(qreg_q[2], creg_d[0])

Appendix II.4 - Mathematical Derivation for Quantum Teleportation Algorithm

The state of qubit q[0] is represented by the following in Dirac notation (26):

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$  \hspace{1cm} (1)

The entanglement of Bob’s qubit and the Alice’s qubit can be represented by the following in Dirac notation:

$$|e\rangle = \frac{1}{\sqrt{2}} (|00\rangle + |11\rangle)$$  \hspace{1cm} (2)

Alice has two qubits and Bob has one, so there are three qubits total, and the Kronecker product is applied to $|\Psi\rangle$ and $|e\rangle$.

$$|\psi\rangle \otimes |e\rangle = \frac{1}{\sqrt{2}} (\alpha|000\rangle + \alpha|011\rangle + \beta|100\rangle + \beta|111\rangle)$$  \hspace{1cm} (3)

The CNOT gate is then applied to q[1] by checking whether q[0] will be 1. The Kronecker Product of the CNOT gate and Identity gate are first taken to alter the dimensions of the CNOT matrix so that it can be applied to $|\Psi\rangle \otimes |e\rangle$:

$$(CNOT \otimes I)(|\psi\rangle \otimes |e\rangle) = \frac{1}{\sqrt{2}} (\alpha|000\rangle + \alpha|011\rangle + \beta|110\rangle + \beta|101\rangle)$$  \hspace{1cm} (4)

The Hadamard gate is applied to q[0] to put it in a superposition. The Kronecker product of two Identity gates and the Hadamard gate is first applied so that the H matrix can be utilized. The equation is also rearranged in a way to only look at the classical bits measured from q[0] and q[1]:

$$(H \otimes I \otimes I) \frac{1}{\sqrt{2}} (\alpha|000\rangle + \alpha|011\rangle + \beta|110\rangle + \beta|101\rangle)$$  \hspace{1cm} (5)
\[
\frac{1}{2} (\alpha|000\rangle + \alpha|100\rangle + \alpha|011\rangle + \alpha|111\rangle + \beta|010\rangle - \beta|110\rangle + \beta|001\rangle - \beta|101\rangle)
\]

\[
\frac{1}{2} (|00\rangle(\alpha|0\rangle + \beta|1\rangle) + |01\rangle(\alpha|1\rangle + \beta|0\rangle) + |10\rangle(\alpha|0\rangle - \beta|1\rangle) + |11\rangle(\alpha|1\rangle - \beta|0\rangle)
\]

Once receiving these measurements, Bob can determine which bits he should use to alter qubit q[2] to match state \(|\psi\rangle\) of q[0]. The NOT gate will change 0 to 1 and 1 to 0, and the Z gate will change 1 to -1 and -1 to 1:

<table>
<thead>
<tr>
<th>Output</th>
<th>Instructions</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>00\rangle(\alpha</td>
</tr>
<tr>
<td></td>
<td>01\rangle(\alpha</td>
</tr>
<tr>
<td></td>
<td>10\rangle(\alpha</td>
</tr>
<tr>
<td></td>
<td>11\rangle(\alpha</td>
</tr>
</tbody>
</table>

Table A.II.4: Output and Instructions from Alice’s Measurements.

The final measuring tool allows the user to verify that Alice’s qubit has “teleported.” Bob is able match his qubit to the state of q[0] without measuring \(\alpha\) and \(\beta\).

Appendix II.5 - Other Experiments on Quantum Teleportation Arranged in Chronological Order:

<table>
<thead>
<tr>
<th>Date</th>
<th>Title/Description</th>
<th>Authors</th>
</tr>
</thead>
<tbody>
<tr>
<td>1997</td>
<td>“Experimental Quantum Teleportation”, one of the earliest experiments that transferred information about the polarization of a photon to another photon (11)</td>
<td>Dik Bouwmeester, et al.</td>
</tr>
<tr>
<td>2013</td>
<td>“Deterministic quantum teleportation of photonic quantum bits by a hybrid technique”, utilized a “hybrid technique involving continuous-variable teleportation of a discrete-variable, photonic qubit” (7)</td>
<td>Shuntaro Takeda, et al.</td>
</tr>
<tr>
<td>2015</td>
<td>“Teleportation of entanglement over 143 km”, teleported photons across 143 km from Tenerife</td>
<td>Thomas Herbsta, et al.</td>
</tr>
<tr>
<td>Year</td>
<td>Description</td>
<td>Authors</td>
</tr>
<tr>
<td>------</td>
<td>-----------------------------------------------------------------------------</td>
<td>----------------------------------------------</td>
</tr>
<tr>
<td>2020</td>
<td>“Teleportation Systems Toward a Quantum Internet”, reached $\geq 90%$ quantum teleportation fidelity of time-bin qubits at “the telecommunication wavelength of 1536.5 nm”</td>
<td>Raju Valivarthi, et al.</td>
</tr>
</tbody>
</table>

Table A.II.5: Quantum Teleportation Experiments.