Building 2-, 3-, and 5-qubit Quantum Random Walks on Near-Term Quantum Computers

Zachary W. Briscoe and Bethan Cordone
(Dated: May 9, 2022)

In this paper we explore the properties of the quantum random walk algorithm using 2, 3, and 5 position qubits. We implemented variations of the 2 position qubit walk including the circular, number line, and weighted coin circuits. We found the 2 position qubit walks agreed with classical expectations for allowed states and probabilities. At walks of 3 steps or more for the 3 and 5 position qubits circuit we found a "clockwise" bias in movement that appears to be non-classical. [a]

I. INTRODUCTION

The classical random walk (CRW) is a powerful algorithm used by scientists and engineers to effectively model stochastic processes in the world. Some examples of these processes include gaseous diffusion, the price of a rapidly fluctuating stock, or the motion of electrons in a conductor. At a fundamental level, these systems all possess three main components: An agent, a coin, and an incrementer. The agent moves around in space while the coin randomly determines the direction of the agent. The incrementer updates the position of the agent and the state of the coin after each cycle. The drawback of the classical RW is that the computation time for a walk scales linearly with the number of steps in the walk, causing some computational problems to become intractable due to time and hardware constraints.

The quantum random walk (QRW) is an analogous algorithm to the CRW which makes use of quantum properties to provide potential speed improvements. The first of these effects is the reversible unitary evolution of the quantum walk. Time evolution is a reversible, unitary operation in the quantum world, so walks that evolve in a quantum computer's state space can theoretically be "fast-forwarded", producing longer walks in a shorter time period. Another quantum effect that could lead to an improvement over the classical walk is quantum interference. In a quantum walk, repeatedly flipping the quantum "coin" operator without measurement causes "state interference", which allows the walk to spread out significantly faster in certain circumstances. Finally, the wavefunction collapse that arises from measurement of the walker's position increases the randomness of the walk when compared to the classical walk. The end result of utilizing these quantum effects is an algorithm that is truly random and can provide users with a quadratic-time speed increase over the CRW, as demonstrated in [1].

In this paper, we look to explore the implementation of QRWs on near-term quantum computing hardware. We use the QRW circuits from [2] and [3] as a starting point for exploring different variations of the QRW on both the quantum simulator in the Qiskit software package and on current IBM hardware. We explore the properties of these circuits including the probability distributions and error rates for different initial conditions and numbers of steps.

[a] The github repository for this project can be found at the following link: https://github.com/bcordone/quantum-random-walk-project.git

II. TWO POSITION QUBIT QUANTUM RANDOM WALK

A. Methodology

The circuits in this section have 2 position qubits which are entangled to create 4 possible position states and a single qubit that acts as a coin. We constructed the circuit using X, CNOT, and Toffoli, and H gates in Qiskit. We explored 4 variations of a 2 position QRW: the circular QRW which had 3 variations Single Direction (SD), Double Direction (DD) , and Weighted Coin (WC), and the Number Line (NL). We constructed the SD circuit from the Qiskit article Studying Quantum Walks on Near-Term Quantum Computers. We built the other variations for the circular QRW based on this circuit. We created the NL circuit from a modification of the methods in paper Implementation of Quantum Random Walk on a Real Quantum Computer.

The circular QRW space is shown in Figure 1 where movement between the "end states" $|00>$ and $|11>$ is allowed. In the first variation of this circuit (SD), the walker could not change directions. We developed a variation of this, the Double Direction (DD) circuit, which could change directions after each step. Alternative to the circular QRW, the NL circuit, also shown in Figure , does not allow the walker to travel between end states so movement between $|00>$ and $|11>$ is prohibited.

Finally, we explored the effects of a "weighted" coin on the DD circuit. A "biased" or "weighted" coin favors a single outcome. The hadamard gates creates the superposition of states $\frac{1}{\sqrt{2}}(|0> + |1>)$ which should have equal probability of measuring $|1>$ and $|0>$ and acts as an unbiased coin. By initializing the coin in a $\frac{1}{\sqrt{5}}(|0> + 2|1>)$ state we create a bias in the coin toss towards a certain direction.

Figure 2 shows the circuit components for a single step of the circular QRW. For SD, only a single H gate was

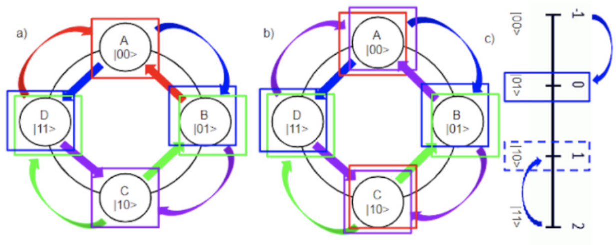

FIG. 1. The walking space with the entangled position qubits labeled for SD (a), DD (b), and NL (c) circuits with initial conditions $|00>$. The walker travels on arrows for previous steps so the change in color indicates the additional step taken from the last iteration: blue (1 step), purple (2 steps), green (3 steps), and red (4 steps). The boxes indicate the allowed end states for that step number. The dashed line indicates a different starting state of $|11>$. The straight arrows are counterclockwise direction while looped arrows are clockwise direction.

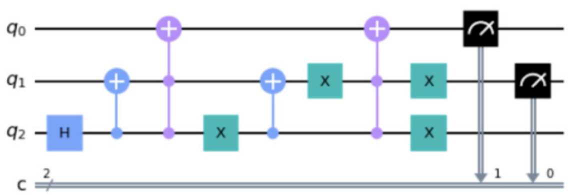

FIG. 2. A single step of the circular QRW. The coin operator creates a superposition of the $|0>$ and $|1>$ states. The other gates rotate the superposition and measurement at the end collapses the superposition to a single result. The walker travels in both directions simultaneously and a direction is only determined with measurement which is performed after all steps are taken.

used at the beginning of the walk and subsequent steps were repeated without the H gate. For DD, the H gate was repeated for the second step and then any subsequent steps after this did not contain the H gate. The SD is analogous to only performing a single coin toss and sticking to the direction of the result while for DD a coin toss is performed after every step allowing for a change in direction. Figure 3 shows the elements of the NL circuit. The WC circuit was the same as in Figure 2 except that we replaced the first H gate with a qubit initialized to the state $\frac{1}{\sqrt{2}}(|0>+|1>)$ and repeated only the second iteration with a H gate.

We ran each circuit on the simulator and on the IBM hardware. On the simulator we ran up to 4 steps of initial condition $|00>$ and $|10>$ for SD or $|01>$ for DD. We ran 1 step (SD and DD) and 2 steps (DD) for the circular QRW on the IBM hardware with initial condition $|00>$. For the NL, we ran up to 4 steps for both "end" points and ran 1 step for the $|00>$ initial condition on the IBM hardware. For the WC we ran up to 4 steps of initial conditions $|00>$ and $|10>$ on the simulator and 1 step on the IBM hardware with initial condition $|00>$.

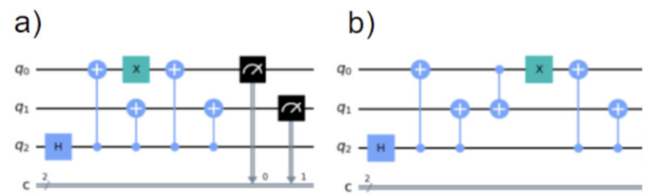

FIG. 3. The elements of the NL circuit. For a single step, we use circuit (a). For multiple steps, we use circuit (b) which is attached to circuit (a) and any additional steps use circuit (a).

B. Results

Figure 1 indicates the "allowed" end states for the walker starting in initial state $|00>$ for all circuit variations. Both the SD and DD circuits have the same paths for the first step as the coin determines whether the walker moves "clockwise" or "counterclockwise" to the states $|01>$ or $|11>$. Both circuits have the same end states for step 3 as they did for step 1 although the paths the circuits are "allowed" to take are different (the default only has one way while the DD has 2 ways to get to one of the end states). The circuits vary for the even states, the SD circuit can only travel 2 steps clockwise or counterclockwise to arrive at state $|10>$. For 4 steps, the walker can only travel in a single direction all the way around the walking space and back to the initial state $|00>$. For the even DD states, the walker is allowed to change directions so that it can travel back to the $|00>$ or forward to the $|10>$. Alternatively to both circuits, the NL circuit prohibits movement between end states meaning that initializing the circuit in these states results in a single allowed state for 1 step.

Figure 4 shows the probability distributions for the circuits on the IBM hardware. The simulator results have the same distributions except there were no errors in prohibited states so the probability was split between the allowed states. We found that the simulator results were as expected for the SD and DD circuits. The probabilities distributions were approximately 50% for cases with two allowed states and 100% for cases with a single allowed state. The probabilities and allowed states follow a cyclical pattern depending on the number of steps. This pattern repeats every 4 steps for SD and every 2 steps for DD. Changing the initial conditions changes the ordering of the allowed state and steps pattern but the cyclic nature is preserved. These results are classicially expected as demonstrated in Figure 1.

The simulator results for the NL are similar to the IBM hardware result in Figure 4(c) except the allowed states occur with 100% probability. When starting at the $|00>$ state the only allowed state is $|01>$ and when starting $|11>$ the only allowed state is $|10>$. This was as expected. When we tried states that were not initial conditions, it does not give the desired outputs and prob-

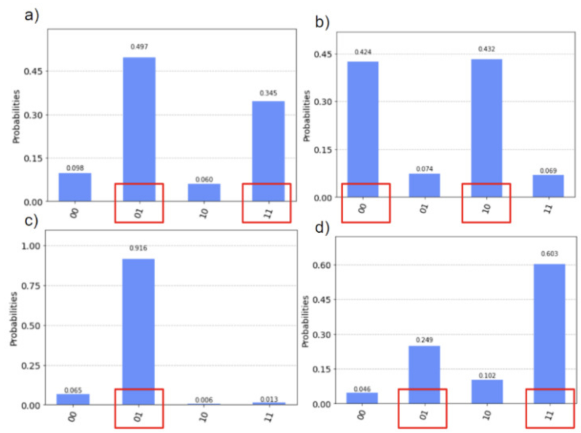

FIG. 4. The probability distribution of QRW outcomes for 1 step SD and DD (a), 2 steps DD (b), 1 step NL (c), and 1 step WC (d) on the IBM hardware. The red boxes are the allowed states.

abilities. This circuit is not generalizable to any initial starting condition.

The probability distribution for the WC results on the simulator for steps 1 and steps 3 are similar to Figure 4(d) except the probability split is closer to 20/80. The results for steps 2 and steps 4 have the same results as the DD circuit. The probability distribution for 1 step is about 20% that the particle moves in the clockwise direction and about 80% that it moves in the counterclockwise direction. The probability distribution of the allowed states are reversed for 3 steps. This happens because the coin is weighted to move with more probability in the counterclockwise direction. This means that when the particle can only travel 1 step it ends up having the highest probability of ending up in the $|01>$ state while when its allowed to travel 3 steps in the highest probability direction it ends up in the $|01>$ state.

Figure 4(a) and (b) show the probability distributions for 1 step of the SD/DD (these are identical for 1 step and 2 steps of DD circuits on the IBM hardware. When 2 steps are taken on the DD circuit. The error in the circuit is just short of double from 1 step to 2 steps. This is likely due to the fact that taking 2 steps requires twice the gates so these errors are likely coming from the gates. Figure 4(c) is the result of the NL and Figure 4(d) is the WC on the IBM software. The number line has the least error of any of the IBM results (about 7%) which makes sense since there is only 1 allowed state and only 1 step of the circuit is taken. The circuit also has fewer components than the circular QRW which could also explain the lower errors. The weighted coin results have about 15% errors which seemed to only decrease the most probable outcome. The least probable outcome was actually more probable when run on the hardware than it was when run on the simulation. The most probable outcome was 20% less probable on the hardware than it was on the simulation.

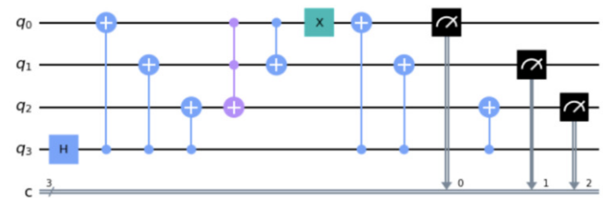

FIG. 5. A single step of the 3 position qubit circular QRW. The H gate is repeated for each iteration and all measurement is performed at the end. The incrementer and decrementer gates are constructed using NOT, CNOT, and Troffoli gates.

III. THREE POSITION QUBIT QUANTUM RANDOM WALK

A. Methodology

The circuit in this section is a 3 position circular QRW. Three position state qubits are entangled to create 8 possible position states. We constructed the circuit in Figure 5 based on the 5 position state qubit circuit that was diagrammed in the same paper the NL circuit was based on. We ran up to 5 steps of the circuit with initial conditions $|000>$ and up to 3 steps with initial conditions $|010>$. We ran a single step of the circuit on the IBM hardware starting in initial condition state $|000>$.

The walking space is outlined in Figure 6. Due to the increased position states from the 2 qubit case, every potential state is connected to only 2 other states meaning that all states are no longer either next to each other or directly across from each other. The cyclical step pattern observed in the 2 position qubits QRW is no longer expected. From a classical viewpoint we would expect to find a probability distribution that followed Pascal's triangle rule.

B. Results

Figure 7(a), (b), and (c) is the probability distributions for 3, 4, and 5 steps of the circuit with initial state $|000>$. Steps 1 and 2 have the expected classical probability distributions of about 50% for each state $|001>$ and $|111>$. Step 2 has 50% probability of state $|000>$ and 25% probability each for $|010>$ and $|110>$. At step 3, the expected classical probability begins to diverge from the observed probability outputs of the QRW. The most probable states are represented in Figure 6. Past 3 steps, these states should be equally probable in the clockwise and counterclockwise directions from a classical standpoint. A pattern emerges where the most probable outcome has a path which moves in the clockwise direction for all but one step. These observations indicate that there is a bias in the QRW circuit towards clockwise movement. This is a quantum observation that is not observed or explained classically.

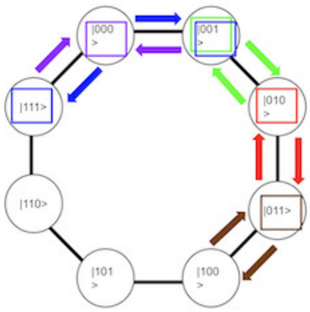

FIG. 6. The walking space for 3 entangled position qubits. The walker travels on arrows for previous steps so the change in color indicates the additional step taken from the last iteration. The paths for the most probable states only are shown. The boxes indicate the states that occurred with the highest probability with labels Blue (1 step), purple (2 steps), green (3 steps), red (4 steps), and brown (5 steps). The walker starts at $|000>$ and the inside arrows indicate the counterclockwise direction while the outside arrows indicate the clockwise direction.

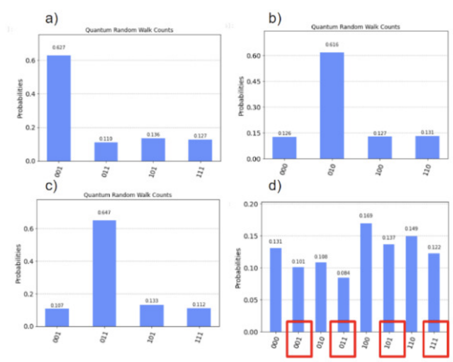

FIG. 7. The probability distribution of 3 steps (a), 4 steps (b), and 5 steps (c) on the simulator as well as the result of 1 step on the IBM hardware with initial condition $|000>$.

We accessed the output for initial condition $|110>$ for up to 3 steps. The same probability pattern is observed with steps 1 and 2 being classically explained and step 3 beginning to diverge. When starting with the new initial condition $|110>$, the most probably state $|111>$ has all the steps except one in the clockwise direction. The new initial condition supports the clockwise bias observation.

Figure 7(d) also has the output of 1 step of the circuit on the IBM hardware. The allowed states $|000>$ and $|111>$ were indistinguishable from the prohibited states due to the large errors. This is likely due to the increased number of entangled qubits.

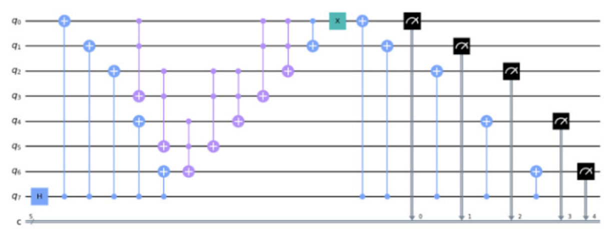

FIG. 8. A single step of the 5 position qubit circular QRW. We also needed a qubit for the coin and 2 ancillae qubits.

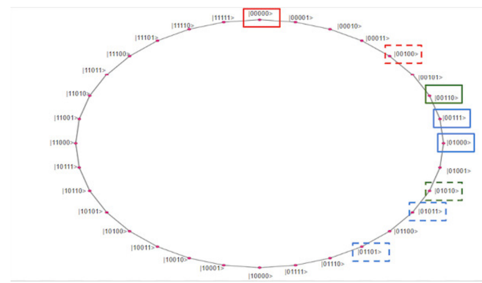

FIG. 9. The walking space for the 5 entangled qubits with 32 total position states. The boxes indicate the most probable states color coded for the number of steps: red (initial condition), green (10 steps), and blue (13 steps). The solid line indicate initial condition $|00000>$ while the dashed is $|00100>$.

IV. FIVE POSITION QUBIT QUANTUM RANDOM WALK

A. Methodology

The circuit in this section is the 5 position circular QRW. Five position state qubits are entangled to create 32 possible position states. We constructed the circuit in Figure 14 from the 5 position state qubit circuit that was diagrammed in the same paper we based the NL circuit on. We ran up to 4 steps of the circuit with an additional run of 10 and 13 steps using the initial conditions $|00000>$. We ran 1 step, 10 steps, and 13 steps with initial condition $|00100>$. We used 1 step of the circuit and changed the initial conditions to map the walking space which is shown in Figure 8. We ran all of these experiments on the simulator since the IBM hardware we had access to could not handle that magnitude of qubits. We have similar classical expectations for this space as we did for the 3 position circuit.

B. Results

Figure 9 is the walking space for the circuit, the allowed states can be traced and verified using this space. Figure 10 shows the probability distribution for the circuit with initial condition $|00000>$ and $|00100>$. The probabilities follow the same pattern as 3 positions for steps 1, 2,

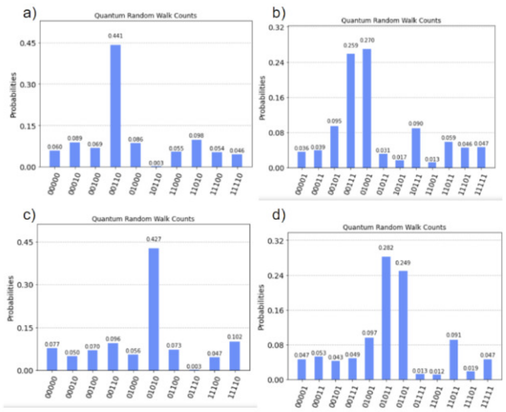

FIG. 10. The probability distribution of 10 steps (a) and 13 steps (b) on the simulator with initial condition $|00000>$ as well as 10 steps (c) and 13 steps (d) with initial condition $|00100>$.

3, and 4. The first two steps have a classical probability distribution while steps 3 and 4 have the most probable state being all steps except one clockwise. For steps 10 and 13 this pattern seems to break down as there are two states that are most probable for step 13 and for step 10 the most probable states do not occur according to this pattern. The most probable states still occur at positions with paths that are biased; it is more probable for the walker to take steps in the clockwise direction than the counterclockwise direction. When we changed the initial condition to —00100¿ we got the same bias for clockwise movement.

V. CONCLUSION

We successfully reproduced the two-position quantum walks described in the introduction. Generally, our simulator data, real-world error rate data and circuit depth measurements matched those of the group in [1]. We also extended our two-qubit walk to operate in three-qubit and five-qubit position spaces. In these walks we generally observed a similar pattern of well-behaved simulator walks and error-filled real-world walks. Also, for the 3-qubit and 5-qubit QRW with iterations greater than 3, we observed a non-classical clockwise directional bias in the probability results.

We also observed that error rates increased with increasing walk length and position states. Based on these observations, we surmised that this rapid onset of errors could be due to the interference of wavefunctions caused by repeated use of the “coin” operator. This is the most logical explanation given the increasing nature of the errors and the longer coherence times necessary for these walks. In the future, some form of quantum error correction could be leveraged to reduce these coin-related errors.

In addition to observing an increasing multiplicity of state errors with increasing circuit complexity, we also observed that the circuit tended to move the agent clockwise on our cyclic walks. This effect has been observed before and is shown in [4]. A complete explanation of the effect was found in [5], which mentions that the asymmetry of the walk displacements arises from the “coin” qubit interfering with the position qubit when operating. This leads to a distribution which leans in this direction because of the fundamental treatment of basis states by the hadamard coin.

In the future, the QRW circuits developed in this paper could be used in a search algorithm to find a marked vertex on a graph. We were successful in producing a cyclic random walk, so future work on this algorithm would primarily center on integrating this walk into a larger framework that can mark vertices within the search domain and register when the marked vertex has been found. Additionally, new implementations of Graph Neural Networks (GNN) that utilize CRWs could see speed improvements in future by switching to using QRWs.

ACKNOWLEDGMENTS

We wish to acknowledge the support of our project mentor, Tyrus Cukavac, for his help and guidance during this project’s creation. We would also like to thank Professor Sebastian Will, and TAs Aden Lam and Joseph Lee for their help in advising our direction for this project.

[1] A. Ambainis, A. Gilyén, S. Jeffery, and M. Kokainis, Quadratic speedup for finding marked vertices by quantum walks, in *Proceedings of the 52nd Annual ACM SIGACT Symposium on Theory of Computing* (Association for Computing Machinery, New York, NY, USA, 2020) p. 412–424.

[2] W. Puengtambol, P. Prechaprapranwong, and U. Taetragool, Implementation of quantum random walk on a real quantum computer, Journal of Physics: Conference Series **1719**, 012103 (2021).

[3] S. Andersson and E. Wanzambi, Studying quantum walks on near-term quantum computers, (2021).

[4] Y. Omar, N. Paunković , L. Sheridan, and S. Bose, Quantum walk on a line with two entangled particles, Physical Review A **74**, 10.1103/physreva.74.042304 (2006).

[5] wslu42, Use quantum random walk to study phase interference (2022).