

Article

# A Quantum Cellular Automata Type Architecture with Quantum Teleportation for Quantum Computing

Dimitrios Ntalaperas <sup>1</sup>, Konstantinos Giannakis <sup>2,\*</sup> and Nikos Konofaos <sup>1,\*</sup>

<sup>1</sup> Department of Informatics, Aristotle University of Thessaloniki, Biology Building, Main University Campus, 54124 Thessaloniki, Greece; ntalaperas@csd.auth.gr

<sup>2</sup> Department of Informatics, Ionian University, Tsirigoti Square 7, 49100 Corfu, Greece

\* Correspondence: kgiann@ionio.gr (K.G.); nkonofao@csd.auth.gr (N.K.)

Received: 15 November 2019; Accepted: 10 December 2019; Published: 17 December 2019



**Abstract:** We propose an architecture based on Quantum Cellular Automata which allows the use of only one type of quantum gate per computational step, using nearest neighbor interactions. The model is built in partial steps, each one of them analyzed using nearest neighbor interactions, starting with single-qubit operations and continuing with two-qubit ones. A demonstration of the model is given, by analyzing how the techniques can be used to design a circuit implementing the Quantum Fourier Transform. Since the model uses only one type of quantum gate at each phase of the computation, physical implementation can be easier since at each step only one kind of input pulse needs to be applied to the apparatus.

**Keywords:** quantum cellular automata; quantum information; cellular automata; quantum gates; quantum circuits

---

## 1. Introduction

Feynman [1] was the first to propose that quantum mechanical principles could be harnessed in order to design computers that could, in principle, be more powerful than their classical counterparts, while Margolus [2] demonstrated the existence of quantum-based computational models that could perform reversible computation in the form of Reversible Cellular Automata (RCA). Various quantum computational models, being able to exhibit universal behavior, have also been defined and shown to be equivalent. In particular, Deutsch [3] defined the formulation of the Quantum Turing Machine (QTM), Yao [4] formulated the Quantum Circuit Model and demonstrated its equivalence with the QTM, while Watrous [5] defined one-dimensional Quantum Cellular Automata and showed that there is an efficient way to simulate a subclass of them by a QTM. On the other hand, quantum computation can be described in the prism of multiple angles (e.g., via a logical/algebraic one [6]).

Quantum algorithms, that is, algorithm that rely on quantum computational models, have already been developed and have been shown to outperform their classical analogue. Specifically, Shor [7] designed a quantum algorithm that can factorize composite numbers achieving a superpolynomial speedup, while Grover [8] demonstrated how searching in an unsorted database can be performed by a quantum algorithm with a quadratic speedup. Physical implementation of a quantum computer can pose a challenge, since quantum systems tend to decohere due to unwanted couplings with the environment. Various implementation architectures that try to overcome this problem have however been proposed [9,10].

There is a growing interest on actual implementations of quantum cellular automata, seeking the most appropriate underlying architecture(s) that will facilitate the effective and fault-tolerant functionality of these computational machines. For an in-depth survey on the progress in quantum cellular automata, the interested reader is referred to Reference [11] by Arrighi and [12] by Farrelly. In a recent work, Karafyllidis and Sirakoulis proposed an alternative computational model based on quantum cellular automata [13]. In their approach, the underlying Quantum Cellular Automata (QCA) form a one-dimensional lattice and computation takes place using the idea of quantum walks, that constitutes a proven universal quantum computation model [14]. A similar study was undertaken by Costa et al. in Reference [15], where they showed that two quantum walkers models (the coined and the staggered quantum walkers [14]) can be translated into quantum cellular automata.

One shared characteristics of the vast majority of implementation architectures is that they allow for only neighboring qubits to interact. Due to this limitation, a number of schemes have been studied, where a general quantum algorithm can be converted to an equivalent one that allows only nearest neighbor interactions [16,17]. Typically, these schemes convert a quantum circuit to one consisting of gates acting only on neighboring qubits at the cost of introducing a number of additional gates to the original circuit.

In this paper, an architecture that enforces only nearest neighbor interactions and the application of only one type of a quantum gate per computational step is introduced. This new architecture is based on Quantum Cellular Automata (QCA). The goals of the new architecture are—(a) To convert a generic quantum circuit to an equivalent one having only nearest neighbor interactions and (b) to allow only a specific quantum gate to act on all the qubits for the duration of one computational step. This latter limitation is introduced since in many implementations it is difficult to localize signals corresponding to two different quantum gates. In the architecture presented, a single signal is applied in each step and the same order of error is introduced to all qubit states (since only one type of gate is executed in each step, the error in each qubit state is the same). The only allowed non local interactions are those of quantum teleportation which have been shown to have a nearly zero error rate in various implementation schemes [18].

The present paper is structured as follows: First, an overview of a QCA is given along with an overview of a specific type of QCA proposed by Karafyllidis [19], which operate by applying a single two-qubit quantum gate over the whole quantum register in each step. Afterwards, an overview of nearest neighbor architectures is presented and it is demonstrated how quantum gates that involve qubits over an arbitrary distance, can be performed by using only local interactions and quantum teleportation in a 2D Grid model architecture which is based in ideas introduced by Rosenbaum [20]. Then, we introduce our proposal where the two techniques are combined to this new architecture that retains the advantages of both methods. Similar to the models of Karafyllidis and Rosenbaum, the architecture presented in this work is a high-tier architecture, the benefits of which stem from the way the computations are performed; it can, in theory, be applied to any scheme that allows local qubit interactions and quantum state teleportations. For this new architecture, the execution of an arbitrary quantum gate is demonstrated and the amount of extra operations required per gate is calculated. Finally, an example of developing a quantum circuit implementing the Quantum Fourier Transform (QFT) is given. This demonstrates how a generic quantum computation can be performed in our model.

## 2. Theory

In this Section, the brief description of a quantum cellular automaton (QCA) is provided along with its corresponding operations and circuit models, as proposed by Karafyllidis in Reference [19].

### 2.1. Quantum Cellular Automata

Arrighi et al. [21] gave the axiomatic definition of a QCA by defining QCA as a unitary operator  $G$  over a quantum labeled graph (QLG). Briefly, a QLG is defined as a tuple

$$\Gamma = (V, E, H),$$

with  $V$  being the nodes,  $E$  is a subset of  $V \times V$  that denotes the set of edges and  $H$  a set of Hilbert spaces being the labels.  $H^x$  denotes the Hilbert space of node  $x$  with  $\Sigma^x$  its possibly infinite countable canonical basis, also known as the alphabet of node  $x$  [21]. Using this definition, a QCA is defined as a unitary operator over QLG  $\Gamma$  with the following restrictions:

- $V$  forming a grid ( $V = \mathbb{Z}^n$ )
- $E$  connecting nodes  $x$  to  $x + z$ , with  $x \in \mathbb{Z}^n$  and  $z \in \{0, 1\}^n$
- $H$  being of finite dimension

Various QCA operational descriptions (e.g., Perez et al. [22]) can be shown to be equivalent to the axiomatic description proposed by Arrighi et al. Descriptions that model the unitary operations as quantum gates constitute can be convenient. Such methods result in quantum circuit structures for QCA.

One such a quantum circuit structure was proposed by Karafyllidis [19]. In this structure, there are two-qubits per cell, the  $s$ -qubit (state qubit) and the  $c$ -qubit (controlled qubit). At each computational step  $t$  the state of the  $j$ th cell can be written as  $|s_j^t c_j^t\rangle$ . Since there are four basis cell for each cell, the general state of each cell during evolution is:

$$|s_j^t c_j^t\rangle = c_{0,j}^t |00\rangle + c_{1,j}^t |01\rangle + c_{2,j}^t |10\rangle + c_{3,j}^t |11\rangle,$$

with  $c_{i,j}^t$  being complex numbers. The state of the QCA consists of the tensor product of all the cell states and can be written as

$$|S\rangle = \otimes_{i=1,n} |s_i^t c_i^t\rangle.$$

In each step, a controlled operation between a control and a target qubit of adjacent cells takes place, followed by a unitary transformation of the two qubits of each cell. State  $S^{t+1}$  is obtained from state  $S^t$  by applying operator  $R$  via:

$$|S^{t+1}\rangle = R |S^t\rangle,$$

with the operator  $R$  given by:

$$R = R_e R_I,$$

where  $R_e$  describes the evaluation phase and  $R_I$  the interaction phase, both given respectively by:

$$\begin{aligned} R_e &= \dots \otimes U \otimes U \otimes \dots \\ R_I &= \dots \otimes CN \otimes CN \otimes \dots, \end{aligned}$$

where  $U$  denotes an arbitrary unitary operation and  $CN$  a Controlled-NOT operation.

The quantum circuit depicted in Figure 1 performs this operation. This architecture exhibits a periodic structure in the output probability patterns.

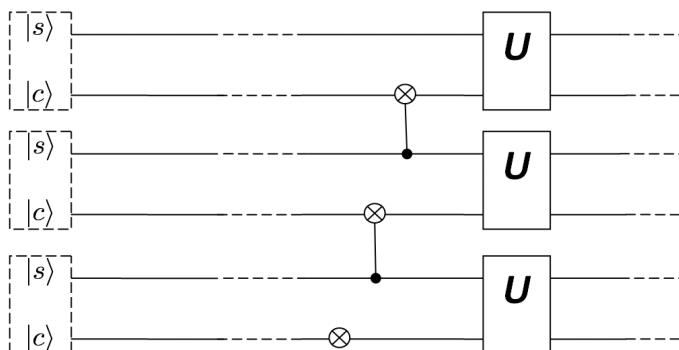


Figure 1. Quantum Cellular Automata (QCA) architecture, as proposed in Reference [7].

### 2.2. Single Gate Operations

In a typical scenario, qubits are manipulated by applying a set of external signals to a set of two-level quantum systems that constitute the quantum register. As a typical example, consider the archetypal Rabi oscillation qubit operations depicted in Figure 2. Two spin 1/2 systems  $q_1$  and  $q_2$  constitute a quantum register of size two and a magnetic pulse with strength equal to

$$B = B_1(\cos(\omega t)\hat{x} + \sin(\omega t)\hat{y}) + B_1\hat{z}$$

is applied to qubit  $q_2$ . Let  $\omega_0 = \gamma B_0$  and  $\omega_1 = \gamma B_1$ , with  $\gamma$  being the gyromagnetic ratio. Then, if we choose  $\omega = \omega_0$  and apply the pulse for a time period of  $t = \pi/\omega_1$  the equivalent quantum gate of a bit flip is applied to qubit  $q_2$ .

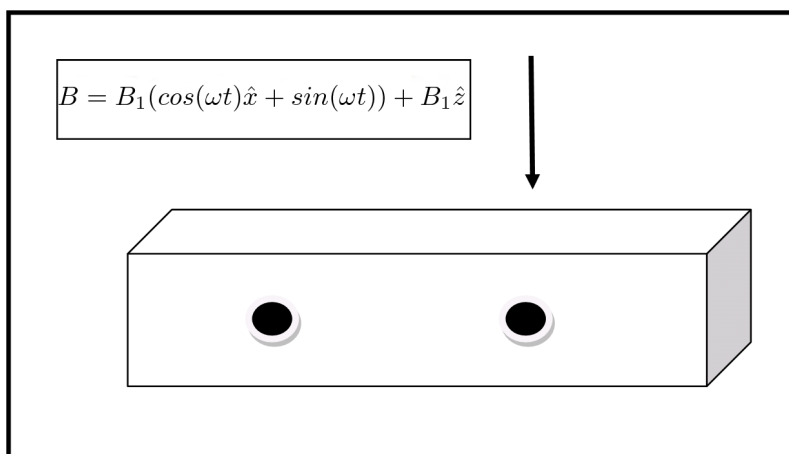


Figure 2. Archetypal spin qubit and qubit operation based on Rabi oscillation.

It is the case, however, that the above pulse can, also, alter the state of qubit  $q_1$ . This is possible, unless: (a) the pulse is extremely localized as to have negligible overlap with  $q_1$  or (b) the resonance frequency of the first qubit is significantly different from that of the second or (c) the separation distance of the two qubits is big enough, thus, the field will couple to the spin of the first qubit. Depending on the implementation architecture, the above conditions may be difficult to control. This is attributed to the fact that localized pulses in momentum space may have a spatial extent that overlaps with the wavefunction of an adjacent qubit or to the fact that having different resonance frequencies may require (depending on the system) complex manipulation of the pulse's parameters. Another contributing factor for this struggle is the use of physical systems that have a different eigenstate spectrum to represent qubits. Increasing the separation distance, on the other hand, may render the interaction between adjacent qubits impossible, thus prohibiting two-qubit operations which are required for universal quantum computing.

All the above conditions may be alleviated if the requirement of performing only one quantum gate to the whole register is imposed in a manner similar to that of Karafyllidis model [19]. The way this is achieved, that is, by converting an arbitrary quantum circuit to an equivalent satisfying this condition (up to some extra teleportation operations) is presented in Section 3.

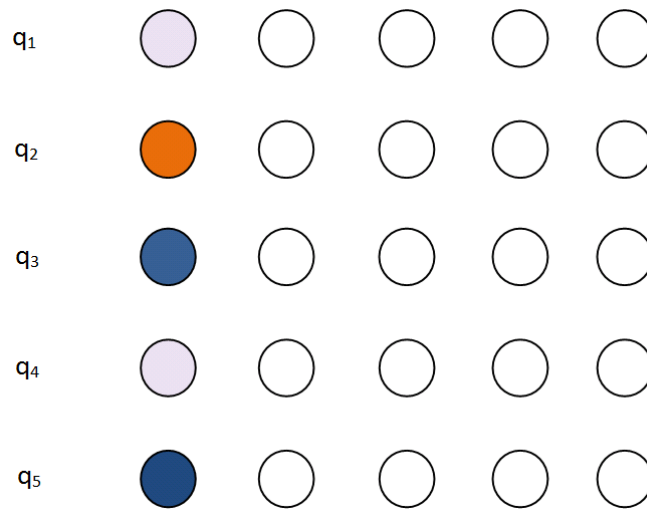
### 2.3. Nearest Neighbor Circuits

Most implementation techniques allow for only local interactions between qubits, therefore prompting the development of various techniques for converting a generic quantum circuit to one consisting only of gates that act upon neighboring qubits. These techniques typically make use of the SWAP gate and result in a significant increase of the number of quantum gates of the circuit. Saeedi et al. [17] have developed sophisticated algorithms which can reduce the cost of this conversion compared to the naive approaches to a factor up to 82%, with the average improvement being around 50%. On the other hand, Rosenbaum [20] has demonstrated a set of techniques for applying a series of reallocations of qubits in a 2-dimensional grid by using quantum teleportation, so that qubits that are to interact over a distance of more than one qubit apart are transported to adjacent positions.

Allowing for teleportation in two dimensions, may further reduce the cost of adding extra gates for achieving nearest neighbor architecture, since the reordering can be done in fewer operations. Quantum teleportation has been studied and shown to be feasible in various models. D'Ariano et al. [23] have demonstrated the existence of QCA that achieves approximate phase-covariant cloning of qubits while Pfaff et al. [18] have demonstrated the teleportation of arbitrary quantum states between diamond spin qubits over long separations. Of particular interest to QCA is the work of Brennen and Williams [24] who have defined optimal pulse sequences for distributing entanglement over a QCA in a 1-dimensional Ising spin chain.

In the following paragraphs, we give a general description of how a general quantum gate between qubits positioned at arbitrary positions can be performed. Since any quantum gate can be decomposed to a set of gates having at maximum two qubits for input, only interactions between two qubits are considered in this work.

Figure 3 depicts the initial configuration of a qubit register. The register resides at the leftmost part of the two dimensional grid. Same colors indicate that the two qubits will be the input to the same quantum gate.



**Figure 3.** Nearest Neighbor interactions in a two dimensional grid. Initial configuration. Qubits of the same color are to interact with each other while qubits having a unique color will be acted upon by a single-qubit gate.

In the particular example, qubit 1 is to interact with qubit 4, qubit 3 with qubit 5 and qubit 2 will be operated by a single-qubit operation. Only teleportation in horizontal and vertical directions of the grid should be allowed.

The listing in Algorithm 1 shows the general algorithm for relocating the qubits on the grid. A global position variable holds the current depth to which a qubit will be teleported. For each qubit, it is examined whether it partakes in a quantum gate operation. Two cases are distinguished:

- The qubit partakes in a single-qubit operation. In this case, the qubit is teleported horizontally to position. Position is decreased by one and the qubit is marked as teleported. The case where the qubit partakes in no operation at all is handled the same way.
- The qubit partakes in a two-qubit operation. In that case the qubit is not marked as teleported and it is teleported horizontally to position. The qubit it interacts with is then teleported to position-1. Position is decreased by two and the two qubits are marked as teleported.

Marking in the above procedure is done classically by marking the index  $i$  that the qubit has in the quantum register by using classical software.

Figure 4 shows a snapshot example of the algorithm when run in the configuration depicted in Figure 3. It is clear that if all the qubits are teleported vertically down to the bottom row, only local interactions are needed in order to execute the computational step required at the starting configuration.

**Algorithm 1:** Algorithm for grid rearrangement

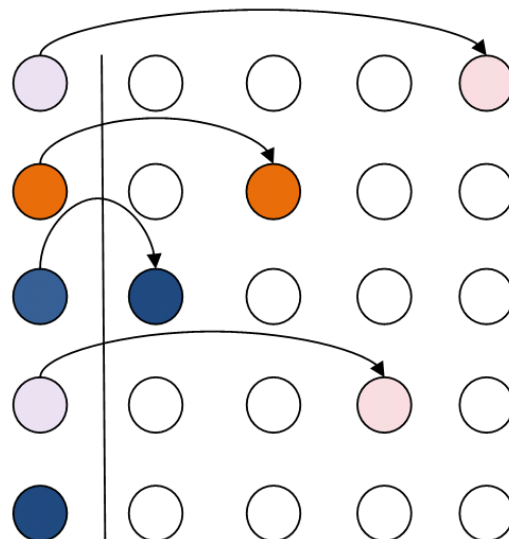
---

```

1 begin
2   Int Position = N
3   Bool Teleported[N]
4   i = 1
5   repeat
6     if qubit(i) interacts_with qubit(j) && !Teleported[j] then
7       Teleport (qubit(i), Position)
8       Teleport (qubit(j), Position-1)
9       Position -= 2
10      Teleported[i]=Teleported[j]=true
11    end
12    if qubit(i) in single operation then
13      Teleport (qubit(i), Position)
14      Position -=1
15    end
16    i +=1
17  until i > N or every qubit was checked;
18 end

```

---



**Figure 4.** Applying the algorithm for grid rearrangement in an example configuration. Qubits of the same color will partake in a single two-qubit operation. The leftmost column is the initial configuration, after the transposition the qubits are teleported to the right in the position indicated by the corresponding color.

### 3. Arbitrary Quantum Operations

The two models of Karafyllidis and Nearest Neighbor presented before are now combined in order to develop a new model for quantum gate interactions. The goal of the model is twofold:

- Each operation is to be performed only by neighboring qubits.

- In each step of the computation, a single two-qubit gate is to be applied to the whole register holding the current state of the system.

Quantum teleportation is not considered in the set of the above operations, since, as already mentioned, there are various architectures that allow for nearly error-free quantum state transportation. For example, Avalle and Serafini [25] have introduced a class of completely positive maps on a qubit array which generalizes the notion of a QCA allowing it to capture all possible stochastic maps on classical probability distributions. Their work allows for the modelling and computation of error due to dephasing or amplitude damping over a general quantum channel. For entangled states that may typically result from two-qubit quantum operations as the ones supported by our architecture, the situation may be more complex, especially when only partial teleportation (where only a subset of the states is to be teleported) is considered. In this case, Prakash and Verma [26] have demonstrated that the fidelity of the teleportation depends on the information of the input state. Bandyopadhyay and Sanders [27] have established lower bounds on the fidelity of the teleportation equal to the fully entangled fraction (FEF) of the Werner state where the FEF is defined as:

$$F(\rho) = \max_u \langle \psi^+ | U^\dagger \otimes I \rho U \otimes I | \psi^+ \rangle,$$

with  $U$  a unitary state and  $\psi^+$  the maximally entangled state and the Werner state of a  $d \times d$  dimensional system being defined as the state satisfying

$$\rho_{AB} = (U_A \otimes U_B) \rho (U_A^\dagger \otimes U_B^\dagger)$$

for all unitary operators acting on  $d$ -dimensional Hilbert space. On the other hand, Yu et al. [28] have demonstrated that when a non-maximally entangled state is used as a quantum channel, then a generic mixed state can be teleported with probability equal to  $1 - (1 - C^2)^{1/2}$ , where  $C$  is the concurrence of the quantum channel where the concurrence provides a measure of the channels entanglement [29].

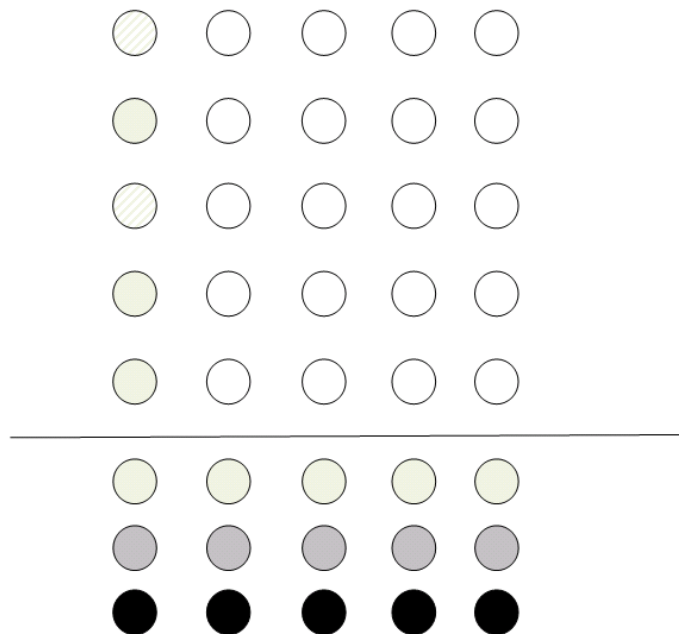
As previously stated, only one-qubit and controlled two-qubit operations are considered. Indeed, such sets of quantum gates (e.g., the Hadamard gate, the  $\pi/8$  gate and the Controlled-NOT gate) that are suitable for universal quantum operations already exist [30].

Since the mechanism for each case (single-qubit operations and controlled operations) is different, it is preferred to, also, treat each case separately.

### 3.1. Single-Qubit Operations

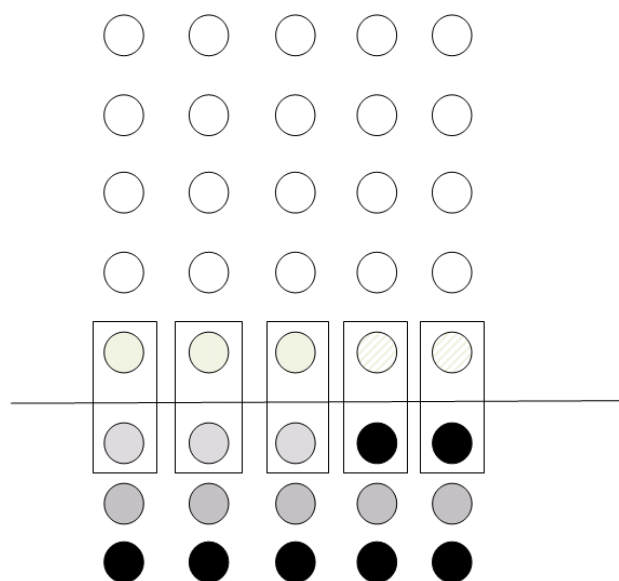
Suppose a single-qubit operation is to be performed on some of the qubits. The nearest neighborhood condition is then automatically fulfilled. However, if the operation is applied to the whole register, all the qubits will be altered (even those not partaking to the computation). In order to circumvent this, all single-qubit operations are converted to controlled operations. The goal is to configure the register so as the qubits needed for the computation to be controlled by ancilla qubits (which are initially in prepared states  $|1\rangle$  and  $|0\rangle$ ). Figure 5 shows an example of the configuration of a grid; qubits with dashed lines are those that need to be transformed according to some unitary operation  $U$ . Grey qubits are ancilla qubits prepared in state  $|0\rangle$ , while qubits in black color are prepared in state  $|1\rangle$ .





**Figure 5.** Initial configuration of the computational grid. The horizontal line separates the qubits partaking to the main computation from the ancilla qubits.

In accordance with the procedure described in Section 2.1, the qubits will be teleported to their corresponding positions at the bottom line of the grid. In the next step, the ancilla qubits are teleported according to whether the operation is to be performed or not. After all the previous teleportation procedures, the final state of the grid is depicted Figure 6. After the state is prepared, the controlled- $U$  operation can be applied simultaneously to the whole register consisting of the data and the teleported ancilla data. Thus, by applying a signal operation across the whole register, the equivalent operation of localizing the operation has been achieved at the cost of  $3N$  extra qubits and  $2N$  teleportation operations.



**Figure 6.** Single-qubit operation-Grid configuration.

### 3.2. Two-Qubit Quantum Operations

For two-qubit operations, the algorithm presented in Section 3 is slightly modified. The second qubit (i.e., the one used as a control) is teleported to the same column as the first qubit. Particularly, it is teleported at the position where the corresponding ancilla qubit of the column resides (Figure 7). The controlled operations are performed in a way similar to the one described in Section 3.1, where the qubits not partaking in the operations are used as input to a controlled operation with a control qubit set to state  $|0\rangle$  via teleportation. The control qubit can then be teleported to the “vacant” spot (circle in red outline in Figure 7) by using two teleportation operations, so that the data register may once again reside in a single line of the grid. The number of operations is the same as those in Section 3.1 plus two extra steps for the final teleportation.

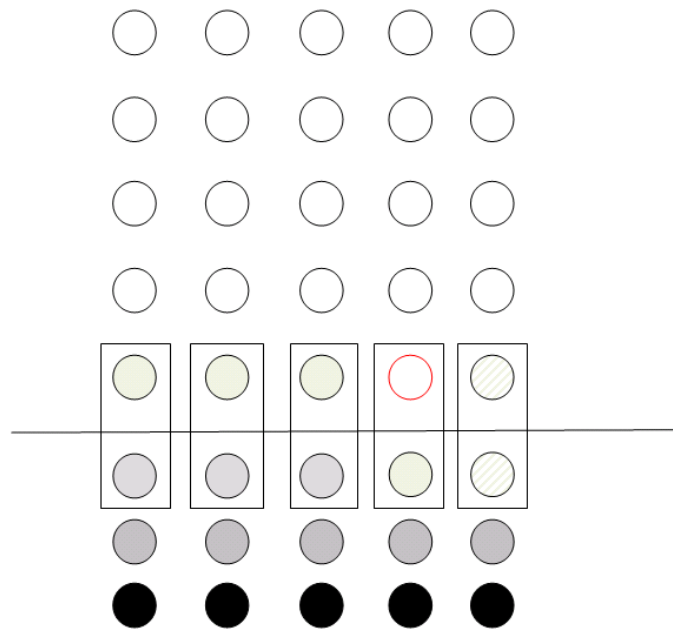


Figure 7. Two-qubit operation–Grid configuration.

### 4. General Algorithm Flowchart

In order to better visualize the execution, the flowchart of the general algorithm is presented in this section, summarizing all the functionalities of the algorithm. The flowchart is depicted in Figure 8. The algorithm loops for  $N$  iterations, where  $N$  is the size of the extended original quantum circuit. Extended circuit in this context means that quantum gates are performed one at a time, except for the case that the same gate is applied to different places in the quantum register. In the latter case, these gates are grouped in the same step in the extended circuit.

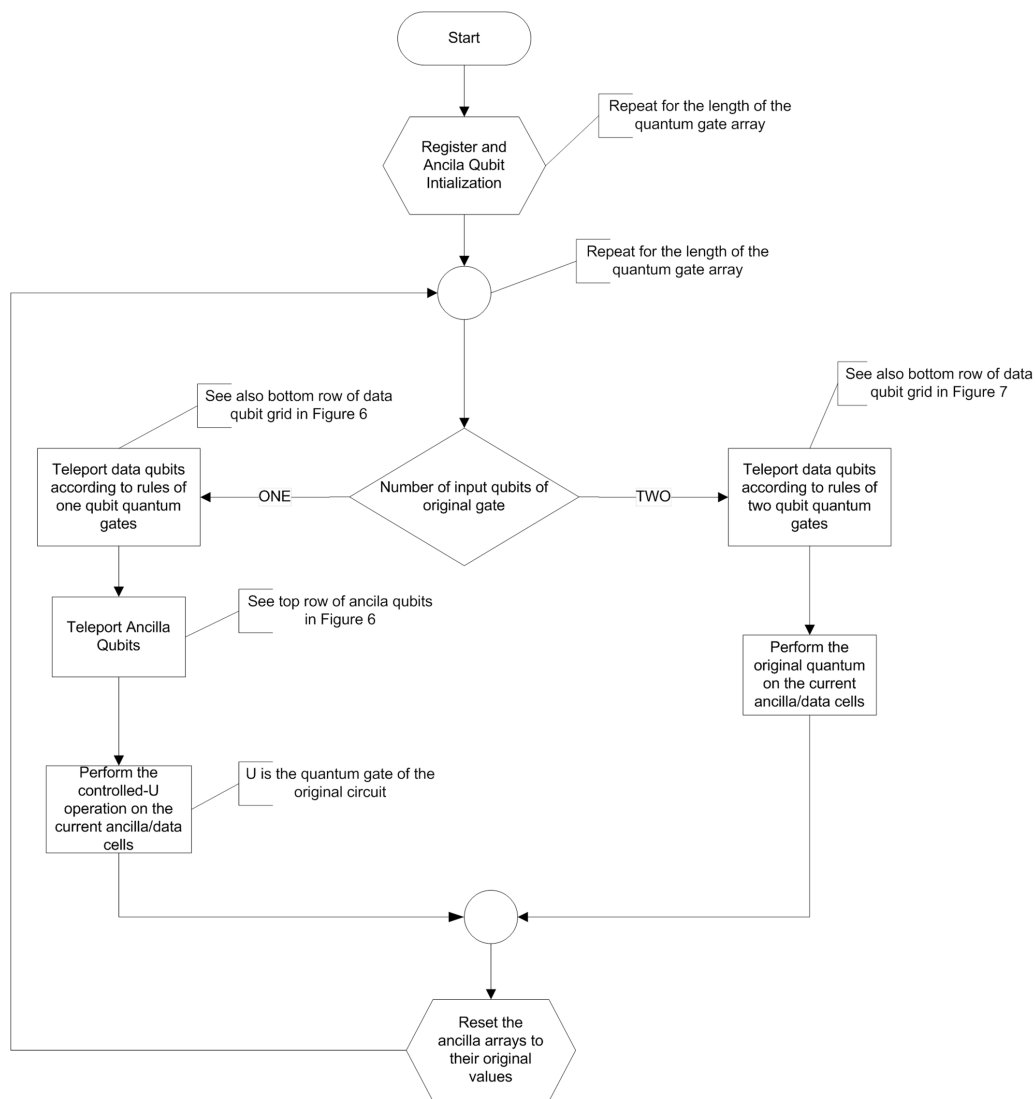


Figure 8. Flowchart of the generic algorithm. Main loop exit and algorithm termination are omitted.

The algorithm converts the computational step of each gate to a set of equivalent gates consistent with the rules presented so far.

- If the quantum gate is a single-qubit gate, the transformation presented in Section 3.1 (Figure 5) is applied
- If the quantum gate is a two-qubit gate, the transformation presented in Section 3.2 (Figure 6) is applied

This process continues for  $N$  steps and the final measurement is performed to the working register. It is essential to restore any ancilla registers to their initial status, since they will be used again each time the working register is rotated back to the same position.

### 5. Application—Quantum Fourier Transform

Quantum Fourier Transform (QFT) resides at the core of Shor’s algorithm. The circuit implementing the Quantum Fourier Transform is demonstrated recursively in Figure 9. In each step, qubit  $n$  is used as a

control for controlled phase operations on each of the other qubits. A single Hadamard transformation is applied to qubit  $n$  at the end.

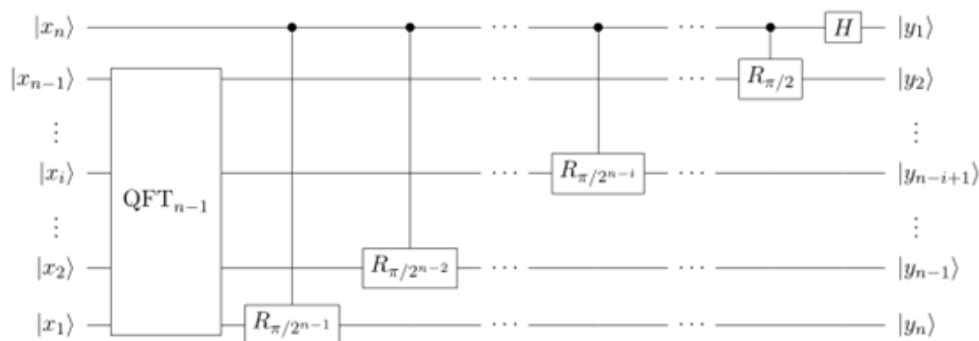


Figure 9. The Quantum Fourier Transform (QFT) circuit.

To demonstrate how the proposed architecture works in a generic scenario, we will show the details of how a sequence of controlled-phase operations and a Hadamard gate are performed. Actually, this can be extended to the complete QFT circuit, since  $QFT^n$  consists of  $n - 1$  controlled-phase operations plus a Hadamard operation.

To begin with, we consider the state of the register where each qubit is in an unknown superposition which depends on the previous computational steps. For the case of a five-qubit register, this is depicted in the left side of Figure 10. Each qubit is represented with a different shade of green color. This shade is used to mark the position of the qubit throughout the computation and is not representative of the qubit’s state.

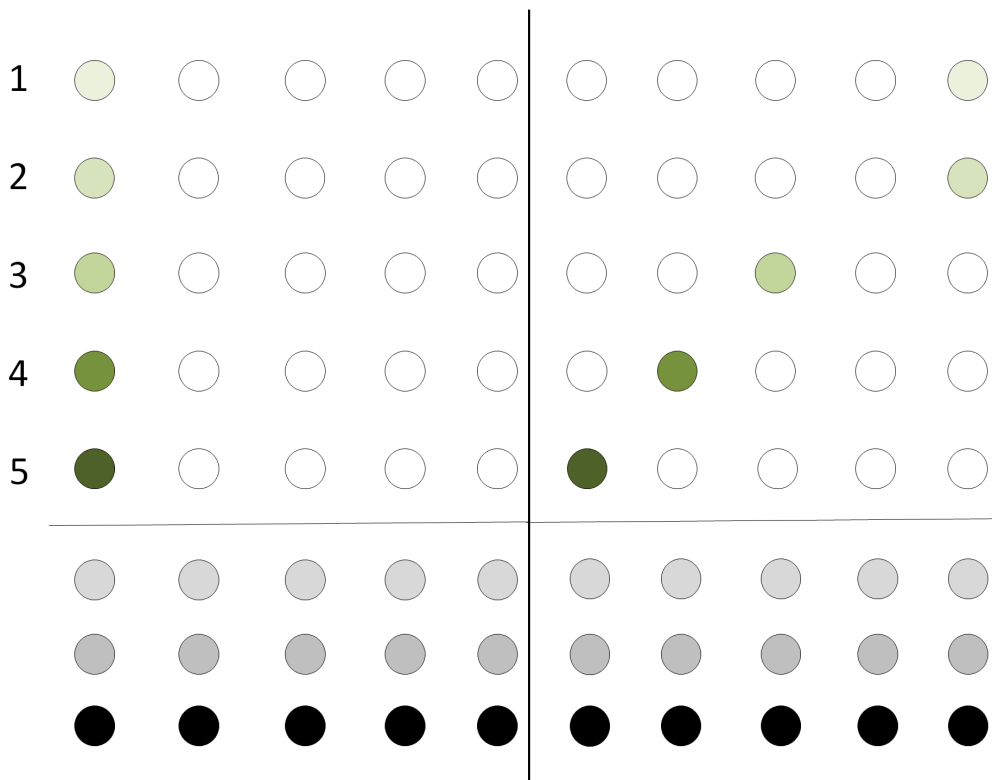


Figure 10. Controlled  $R_{\pi/2}$ . From initial configuration (left side) to the first horizontal teleportation. Qubits 1 and 2 will interact so they are teleported to the same column.

At first, the controlled  $R_{\pi/2}$  gate is to be performed between qubits at row positions 1 and 2. In accordance with the algorithm for performing a two-qubit gate, the qubits are first teleported horizontally as displayed in the right side of Figure 10.

The process continues by vertical teleportation and the grid reaches the configuration depicted on the left side of Figure 11. In the same instance, we demonstrate the teleportation of qubits in state  $|0\rangle$  to the relevant positions in the ancilla qubits row. The ancilla qubit in column 4 is not used and, thus, the corresponding teleportation is not performed.

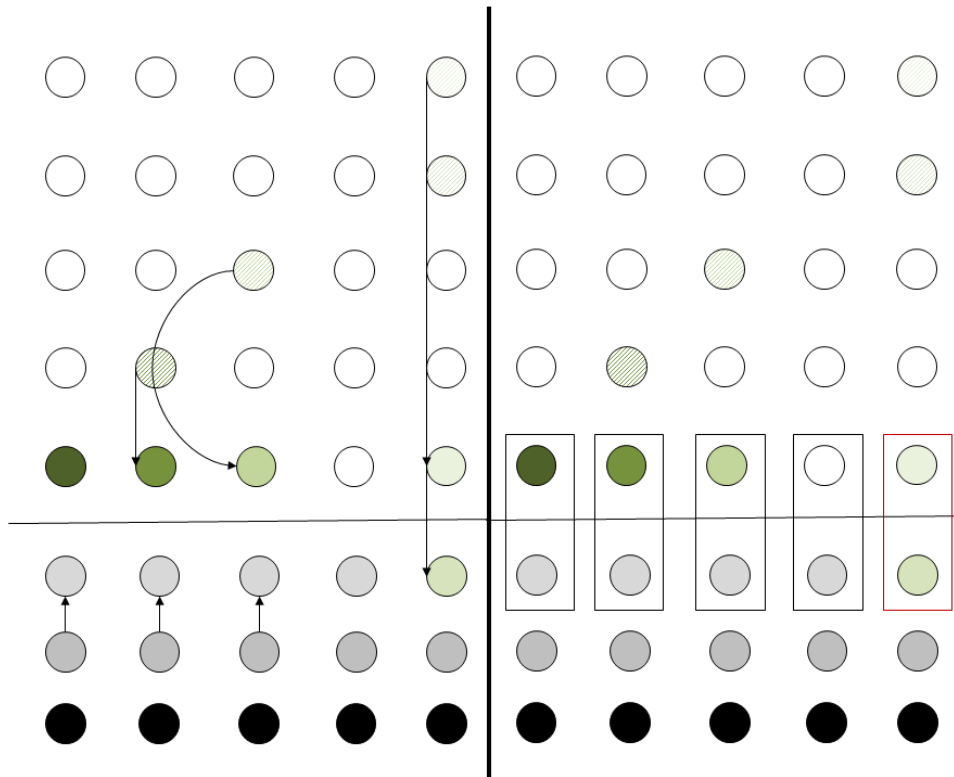


Figure 11. Controlled  $R_{\pi/2}$ . Vertical teleportation and the application of the two-qubit quantum operation.

After the vertical teleportation, the grid has reached a configuration that allow the actual computation to take place. This is shown on the right side of Figure 11. Each box represents a controlled  $R_{\pi/2}$  operation. The first four operations leave the computational qubits unaffected, since the ancilla qubit (which is used as a control) is in state  $|0\rangle$ . The last operation (denoted by the box in red outline in the right side of Figure 11) is the actual  $R_{\pi/2}$  operation between qubits 1 and 2 that is required by QFT. All these operations are performed in parallel, so that in each step, a single two-qubit operation is performed both to the whole register and the ancilla qubit array. As required by the model to perform the computation, only nearest neighbor operations are applied, whilst a single kind of signal is required, too.

Then, in order to complete the computation, the Hadamard gate must be performed on the first qubit. At first, the cleanup actions of the previous operation must be performed. This, as described in Section 3.2, involves the teleportation of qubit 2 to the “vacant” position of the register. After the teleportation, the register that now contains the original qubits can be used to initiate the procedure needed for the Hadamard gate. This initial configuration is depicted in the left side of Figure 12. The grid is depicted “rotated”. This rotation is not physical but rather a visual cue to help visualizing the process; the qubits should be considered to occupy the same places in the grid as before.

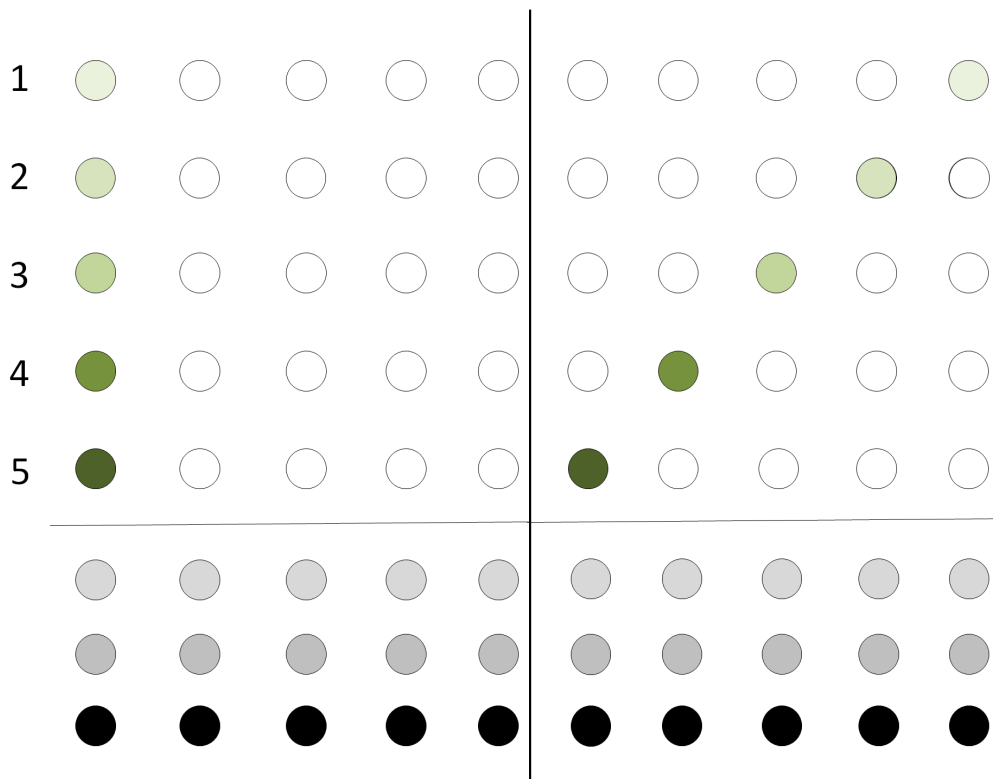


Figure 12. Hadamard Gate. From initial configuration to the first horizontal teleportation.

In the right side of Figure 12, the configuration of the grid after the first horizontal teleportation may be seen. This step is similar as before, the exception being that now all qubits are teleported in their respective columns since no qubit will act as control.

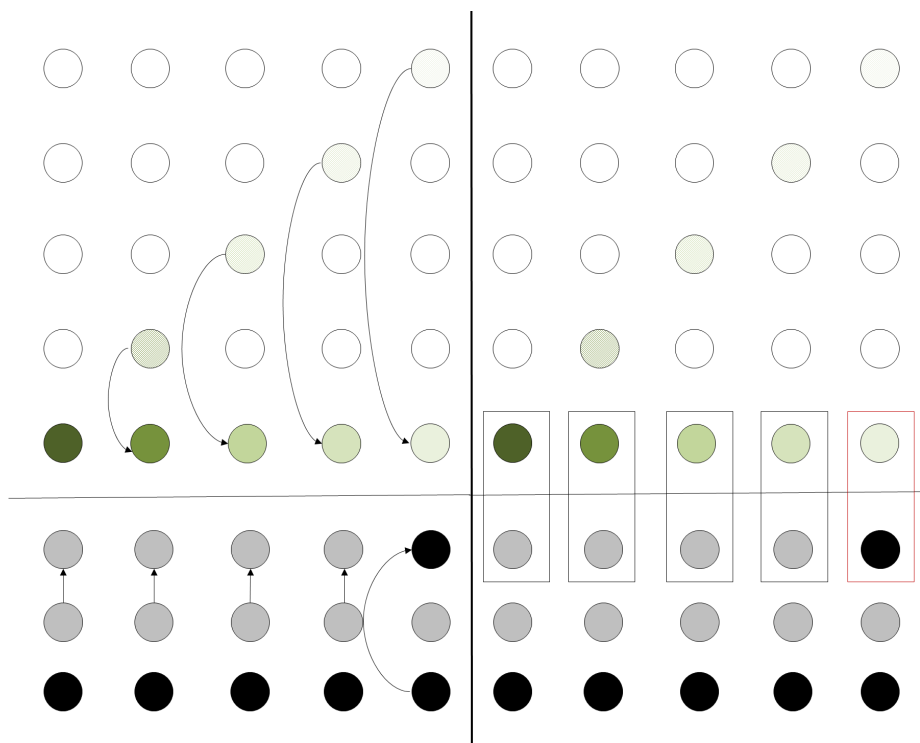
The algorithm proceeds with the vertical teleportation, as can be seen in the left side of Figure 13. In parallel, the needed ancilla qubits are teleported. For this operation, the ancilla qubits of the first four columns are set to state  $|0\rangle$  by teleporting qubits already set to state  $|0\rangle$ . This ensures that the state of the corresponding qubits in the working register will remain unaltered. For the 5th column, a qubit in state  $|1\rangle$  is teleported in the corresponding ancilla position. Similarly to the previous gate, a controlled Hadamard is performed in a pairwise fashion to all qubits of the working register and the ancilla register. Since only the qubit of the fifth column is affected, this operation is equivalent to a Hadamard gate performed on said qubit.

Therefore, it has been demonstrated that the proposed model is at least capable of performing any computation that can be described by a Quantum Circuit and is, in this sense, Turing complete (since the quantum circuit model has already been shown to be Turing-complete [31,32]).

For computing the total additional space and number of steps that the model introduces compared to a quantum circuit of size  $N$ , we note the following:

- The grid needs  $N^2$  working qubits plus  $4N$  qubits for the row of the ancilla qubits. An additional  $2N$  qubits holding prepared states of  $|0\rangle$  and  $|1\rangle$  are needed if we wish to perform all the ancilla teleportation operations in parallel.
- Each computational step needs  $N$  horizontal teleportation operations plus  $N$  vertical (the extra two teleportation in the case of two-qubit gates can be ignored). There are, also, at max  $N$  teleportation operations for the ancilla qubits plus at max  $N$  initialization operations, since we need the qubits

holding the prepared states to be returned to their original states after the teleportation operations. Finally, there are  $N$  controlled operations in each phase of the computations.



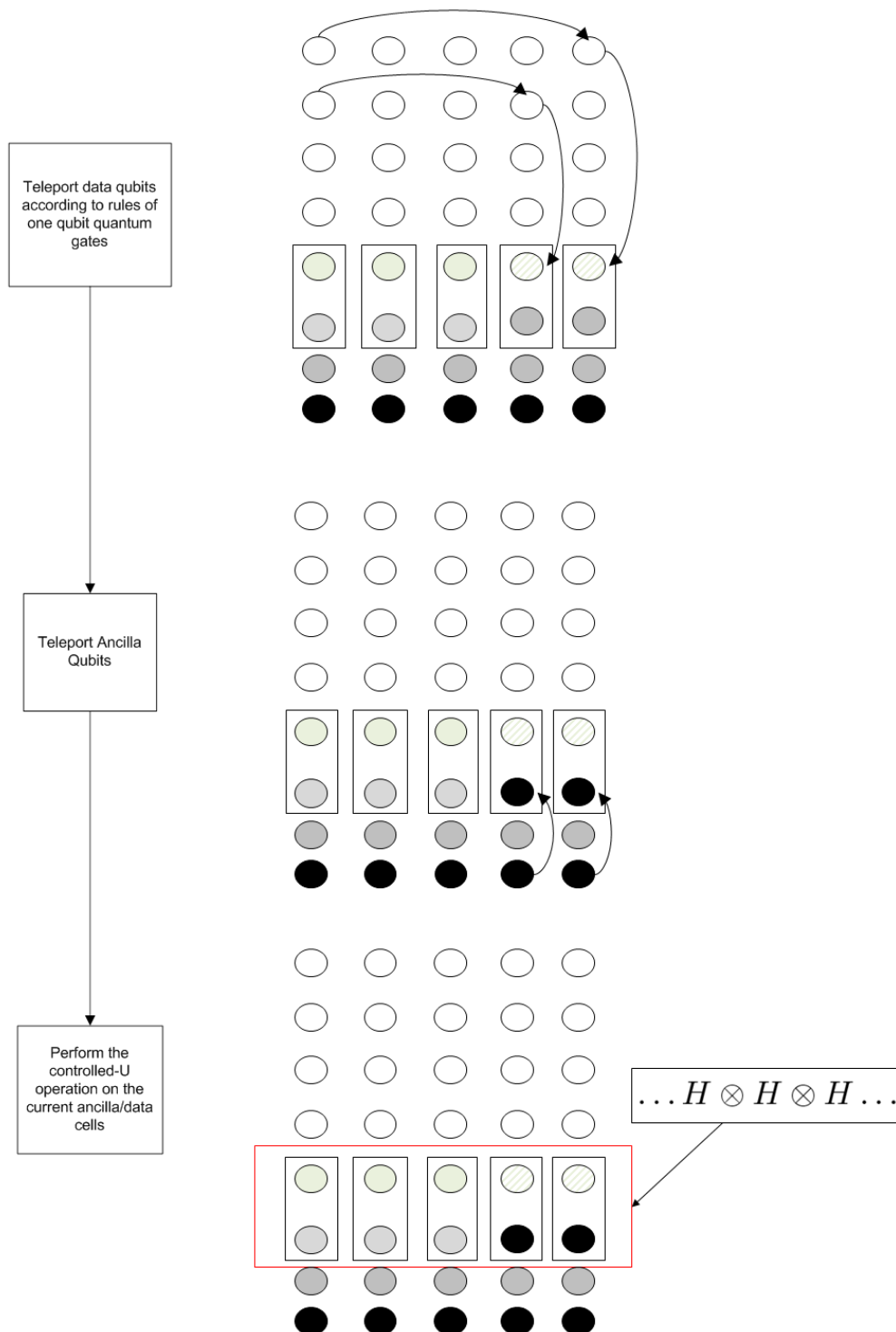
**Figure 13.** Hadamard gate. Vertical teleportation and applying the gate using control qubits in pre-prepared states.

The above calculations yield a cost of additional  $\mathcal{O}(n^2)$  space and  $\mathcal{O}(n)$  additional steps. However, as has been demonstrated, the additional steps can be performed in parallel with each operation being executed in three phases. Phase one consists of the initial horizontal teleportation, phase two contains the vertical teleportation and phase three consists of the quantum controlled operations (ancilla teleportation operations and re-initialization may be performed during phases one and/or two).

The benefit from all these, however, is that the model permits only nearest neighbor quantum operations which can be performed using localized signals. Furthermore, a single kind of signal is performed during each computational step. It should be noted that, ignoring quantum teleportations (which can be regarded as qubit transfer operations), the depth of the original circuit does not change despite the introduction of extra quantum gates. This is due to the parallel execution of these gates in each step and it is in contrast with various nearest neighbor architectures that require an increase in the quantum circuit’s depth in the general case. Including teleportations, the increase of circuit depth is up to a constant per each step.

*Flowchart of Quantum Fourier Transform Operations*

The Quantum Fourier Transform consists of a set of Hadamard and controlled-Phase operations. The various stages of the execution of each gate and their correspondence with the new algorithm’s flowchart are depicted in Figure 14 (single-qubit operations) and Figure 15 (two-qubit operations).



**Figure 14.** Stages for the execution of the Hadamard gate. It is implied that the operation performed in the last stage is the controlled Hadamard operation.



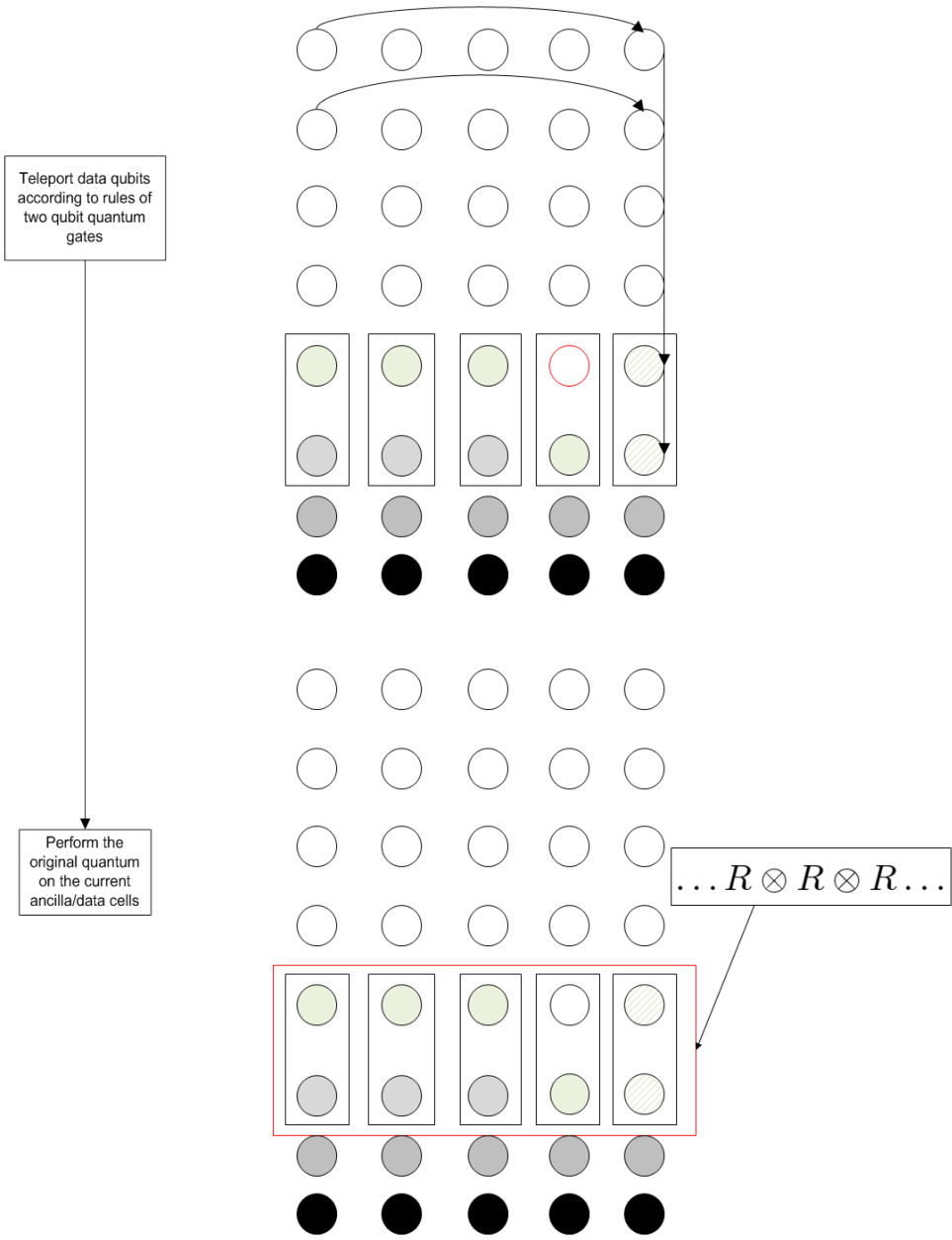


Figure 15. Stages for the execution of the controlled-Phase operation.

Each of the two possible sets of operations of the main loop of the algorithm is mapped into the state of the computational grid with arrows denoting teleportation operations. As can be seen, apart from these teleportations, only one type of quantum gate is applied in the last stage of the procedure to the whole working register.

6. Conclusions

In this paper, a novel architecture has been proposed with the aim of combining the techniques of Quantum Cellular Automata and Nearest Neighbor interactions. The proposed architecture benefits from the advantages of both techniques; QCA model offers simplicity in design and the capability of applying a

gate array consisted of similar gates, whereas the Nearest Neighbor model offers an easier implementation scheme and a low error rate overhead.

While the proposed architecture requires that some extra computational steps and qubits have to be introduced (of linear and quadratic order respectively), this is compensated by the fact that the underlying operations are performed in a local and controlled manner, which in turn yields the decrease of the error rate of each computational step.

**Author Contributions:** All authors contributed equally.

**Funding:** This research received no external funding.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## Abbreviations

The following abbreviations are used in this manuscript:

QCA	Quantum Cellular Automata
QTM	Quantum Circuit Model
QFT	Quantum Fourier Transform
QLG	Quantum Labeled Graph

## References

1. Feynman, R.P. Simulating physics with computers. *Int. J. Theor. Phys.* **1982**, *21*, 467–488. [[CrossRef](#)]
2. Margolus, N. Physics-like models of computation. *Phys. D Nonlinear Phenom.* **1984**, *10*, 81–95. [[CrossRef](#)]
3. Deutsch, D. Quantum theory, the Church–Turing principle and the universal quantum computer. *Proc. R. Soc. Lond. A. Math. Phys. Sci.* **1985**, *400*, 97–117. [[CrossRef](#)]
4. Yao, A.C.C. Quantum circuit complexity. In Proceedings of the 1993 IEEE 34th Annual Foundations of Computer Science, Palo Alto, CA, USA, 3–5 November 1993; pp. 352–361.
5. Watrous, J. On one-dimensional quantum cellular automata. In Proceedings of the IEEE 36th Annual Foundations of Computer Science, Milwaukee, WI, USA, 23–25 October 1995; pp. 528–537.
6. Holik, F.; Sergioli, G.; Freytes, H.; Plastino, A. Logical structures underlying quantum computing. *Entropy* **2019**, *21*, 77. [[CrossRef](#)]
7. Shor, P.W. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Rev.* **1999**, *41*, 303–332. [[CrossRef](#)]
8. Grover, L. A fast quantum mechanical algorithm for database search. *arXiv* **1996**, arXiv:quant-ph/9605043.
9. Kane, B.E. A silicon-based nuclear spin quantum computer. *Nature* **1998**, *393*, 133. [[CrossRef](#)]
10. Vandersypen, L.M.; Steffen, M.; Breyta, G.; Yannoni, C.S.; Sherwood, M.H.; Chuang, I.L. Experimental realization of Shor’s quantum factoring algorithm using nuclear magnetic resonance. *Nature* **2001**, *414*, 883. [[CrossRef](#)]
11. Arrighi, P. An overview of quantum cellular automata. *arXiv* **2019**, arXiv:1904.12956.
12. Farrelly, T. A review of Quantum Cellular Automata. *arXiv* **2019**, arXiv:1904.13318.
13. Karafyllidis, I.G.; Sirakoulis, G.C. Quantum Walks on Quantum Cellular Automata Lattices: Towards a New Model for Quantum Computation. *Int. Conf. Cell. Autom.* **2018**, *11115*, 319–327.
14. Venegas-Andraca, S.E. Quantum walks: A comprehensive review. *Quantum Inf. Process.* **2012**, *11*, 1015–1106. [[CrossRef](#)]
15. Costa, P.C.; Portugal, R.; de Melo, F. Quantum walks via quantum cellular automata. *Quantum Inf. Process.* **2018**, *17*, 226. [[CrossRef](#)]
16. Kumar, P. Efficient quantum computing between remote qubits in linear nearest neighbor architectures. *Quantum Inf. Process.* **2013**, *12*, 1737–1757. [[CrossRef](#)]
17. Saeedi, M.; Wille, R.; Drechsler, R. Synthesis of quantum circuits for linear nearest neighbor architectures. *Quantum Inf. Process.* **2011**, *10*, 355–377. [[CrossRef](#)]

18. Pfaff, W.; Hensen, B.J.; Bernien, H.; van Dam, S.B.; Blok, M.S.; Tamini, T.H.; Tiggelman, M.J.; Schouten, R.N.; Markham, M.; Twitchen, D.J.; et al. Unconditional quantum teleportation between distant solid-state quantum bits. *Science* **2014**, *345*, 532–535. [[CrossRef](#)]
19. Karafyllidis, I.G. Definition and evolution of quantum cellular automata with two qubits per cell. *Phys. Rev. A* **2004**, *70*, 044301. [[CrossRef](#)]
20. Rosenbaum, D. Optimal quantum circuits for nearest-neighbor architectures. *arXiv* **2012**, arXiv:1205.0036.
21. Arrighi, P.; Nesme, V.; Werner, R. Unitarity plus causality implies localizability. *J. Comput. Syst. Sci. Int.* **2011**, *77*, 372–378. [[CrossRef](#)]
22. Pérez-Delgado, C.A.; Cheung, D. Local unitary quantum cellular automata. *Phys. Rev. A* **2007**, *76*, 032320. [[CrossRef](#)]
23. D’Ariano, G.; Macchiavello, C.; Rossi, M. Quantum cloning by cellular automata. *Phys. Rev. A* **2013**, *87*, 032337. [[CrossRef](#)]
24. Brennen, G.K.; Williams, J.E. Entanglement dynamics in one-dimensional quantum cellular automata. *Phys. Rev. A* **2003**, *68*, 042311. [[CrossRef](#)]
25. Avalle, M.; Serafini, A. Noisy quantum cellular automata for quantum versus classical excitation transfer. *Phys. Rev. Lett.* **2014**, *112*, 170403. [[CrossRef](#)] [[PubMed](#)]
26. Verma, V.; Prakash, H. Quantum Teleportation of Single Qubit Mixed Information using Werner-Like State as Resource. In Proceedings of the 12th International Conference on Fiber Optics and Photonics, Kharagpur, India, 13–16 December 2014.
27. Bandyopadhyay, S.; Sanders, B.C. Quantum teleportation of composite systems via mixed entangled states. *Phys. Rev. A* **2006**, *74*, 032310. [[CrossRef](#)]
28. Chang-Shui, Y.; Ya-Hong, W.; He-Shan, S. Teleportations of mixed states and multipartite quantum states. *Commun. Theor. Phys.* **2007**, *47*, 1041. [[CrossRef](#)]
29. Yin-Sheng, L. Channel’s concurrence and quantum teleportation. *Commun. Theor. Phys.* **2005**, *43*, 55. [[CrossRef](#)]
30. Nielsen, M.A.; Chuang, I.L. *Quantum Computation and Quantum Information*; Cambridge university press: Cambridge, UK, 2010.
31. Bernstein, E.; Vazirani, U. Quantum complexity theory. *SIAM J. Comput.* **1997**, *26*, 1411–1473. [[CrossRef](#)]
32. Molina, A.; Watrous, J. Revisiting the simulation of quantum Turing machines by quantum circuits. *Proc. R. Soc. A* **2019**, *475*, 20180767. [[CrossRef](#)]



© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).