Periodicity in Quantum Cellular Automata

Georgios I. Tsormpatzoglou and Ioannis G. Karafyllidis

Department of Electrical and Computer Engineering
Democritus University of Thrace 67100 Xanthi, Greece
{geortsor,ykar}@ee.duth.gr

Abstract. Studies of quantum computer implementations suggest cellular quantum computer architectures. These architectures are based on the evolution of quantum cellular automata, which can possibly simulate both quantum and classical physical systems and processes. It is however known that except for the trivial case, unitary evolution of one-dimensional homogeneous quantum cellular automata with one quantum bit (qubit) per cell is not possible because of the no-go lemma. In this paper, we define quantum cellular automata that comprise two qubits per cell and study their evolution using a quantum computer simulator. The evolution is unitary and its linearity manifests itself as a periodic structure in the probability distribution patterns.

Keywords: Quantum Computer, Quantum Cellular Automata, Simulation, Periodicity.

1 Introduction

Cellular Automata (CAs) were proposed by von Neumann as a model of self-replicating systems [1]. The CA model has been successfully used for the simulation of physical systems and processes and served as basis for parallel computer processor architectures [2,3,4].

One of the main issues in quantum information processing concerns the possible quantum computer architectures [5,6]. Feynman examined the possibility of using CAs both as models of quantum systems and as quantum computer architecture in 1982 [7]. Several studies of quantum computer implementations suggest that cellular architectures are the natural quantum computer architectures [8,9]. CAs that evolve according to quantum mechanics are referred to as quantum cellular automata (QCAs).

Studies revealed that, except for the trivial case, unitary evolution of one-dimensional QCAs is impossible [7,10,11]. This is known as the "no-go lemma" and presents a major obstacle for the construction of cellular quantum computer architectures. In order to overcome the obstacle, an alternative QCA structure using two qubits per cell was proposed in [12,13].

G.C. Sirakoulis and S. Bandini (Eds.): ACRI 2012, LNCS 7495, pp. 585–590, 2012.

2 Definition of QCAs with Two Qubits per Cell

In the QCAs that we are about to define, the same evolution rule applies to all cells at all times. The number of spatial dimensions of the QCA is one, i.e. it forms one-dimensional lattice.

Each QCA cell comprises two qubits. The first qubit is the controlled qubit (c-qubit) and the other is the state qubit (s-qubit). The state of the jth QCA cell at computation step t is written as:

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

where $c_{n,j}^t$, $n = 0, 1, 2, 3$ are the basis-state probability amplitudes and therefore are complex numbers. The global state of the QCA at computation step t, S^t, is the tensor product of the states of all of its cells.

Three different neighborhood sets are considered for the QCAs defined here. The first set of the jth QCA cell neighborhood comprises the same cell and the $j + 1$ cell, the second set the same cell and the $j - 1$ cell and the third set the same cell and both the $j - 1$ and $j + 1$ cells.

The quantum computation starts with a preparation phase in which the qubits may be set in basis-state superposition and entangled states may be produced.

The QCAs defined here, evolve according to a global unitary rule R. The rule R is applied in two phases. In the first phase, which is the interaction phase, the state of the c-qubit at each cell is changed according to the states of the s-qubits in the neighboring cells. In the second phase, which is the evaluation phase, a two-input quantum gate or a combination of quantum gates is applied to the c-qubit and the s-qubit at each cell, thus the information flow in the QCA lattice and simultaneous change of all s-qubit states. Therefore, R is given as a product of the unitary operators R_I and R_E which correspond to the interaction and evaluation phases:

$$|S^{t+1}\rangle = R|S^t\rangle = R_E R_I|S^t\rangle \tag{2}$$

Figure 1 shows three QCA evolution rules as quantum circuits.

In the interaction phase, a quantum Controlled-NOT (CNOT) gate is applied to the two cells of each neighborhood or in the case of Fig. 1(c) a Controlled-Controlled-NOT (CCNOT) gate is applied to the three cells of each neighborhood. The CNOT gates are applied to all neighborhoods and the operator R_I is given by the tensor product:

$$R_I = \cdots \otimes \text{CNOT} \otimes \text{CNOT} \otimes \text{CNOT} \otimes \cdots \tag{3}$$

During the interaction phase, the states of all s-qubits remain unaltered and the states of all c-qubits change.

In the evaluation phase, a two-input quantum gate or a combination of quantum gates is applied to the qubits of each cell, therefore the operator R_E is given by the tensor product:

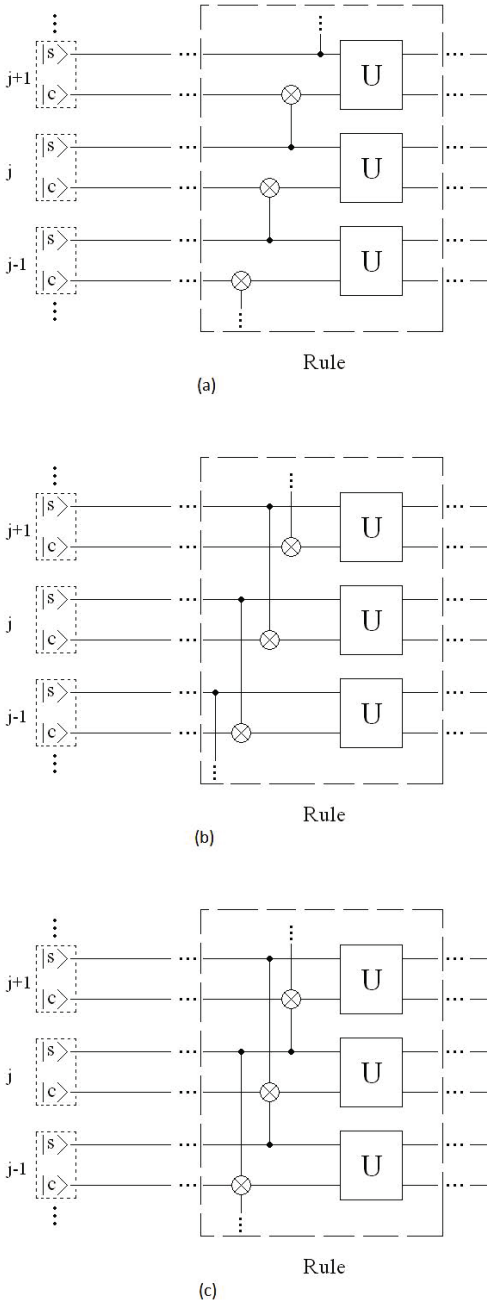

Fig. 1. (a) The evolution rule for the case where the neighborhood comprises the jth and the $j+1$ cells. (b) The evolution rule for the case where the neighborhood comprises the jth and the $j-1$ cells. (c) The evolution rule for the case where the neighborhood comprises the jth and both the $j-1$ and $j+1$ cells. Figure from [13].

$$R_E = \cdots \otimes \mathrm{U} \otimes \mathrm{U} \otimes \mathrm{U} \otimes \cdots \tag{4}$$

During the evaluation phase, both qubit states may change.

Constant and cyclic boundary conditions will be defined for the QCAs. In the case of constant boundary conditions a c-qubit at the end of the lattice is assumed to be controlled by a qubit which is constantly in basis-states $|0\rangle$ or $|1\rangle$ and a s-qubit is assumed to control a qubit which does not participate in the QCA global state. In cyclic boundary conditions the $j-1$ neighbor of the first cell is the last cell and the $j+1$ neighbor of the last cell is the first cell.

3 Evolution of QCAs with Two Qubits per Cell

The evolution of QCAs with two qubits per cell was studied using a quantum computer simulator presented in [14,15]. The graphical output of the simulator shows the probability distribution at each computational step.

Figure 2 shows the simulation of the evolution of three-cell QCAs. Comparison of Fig. 2(a), 2(b) and 2(c) shows that even in the simplest case, entirely different final states can be reached by changing the initial state of only one qubit.

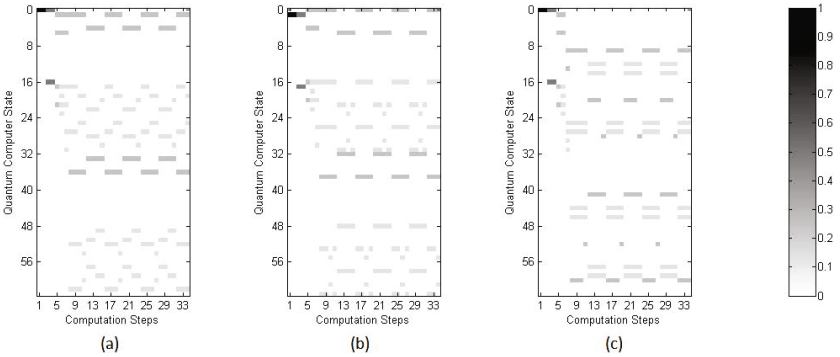

Fig. 2. Simulation of the evolution of a three-cell QCA. During the preparation phase, the first cell qubits are entangled using a Hadamard-CNOT gate and second and third cell qubits are set in basis-state superposition using a Hadamard gate. (a) The initial state is $|000000\rangle$. (b) The initial state is $|000001\rangle$. (c) Differs from (a) in the application of a CNOT gate at the third cell at the first evaluation phase. Figure from [12].

Figure 3 shows the simulation of the evolution of four-cell QCAs. The periodic structure of the probability distribution patterns is apparent.

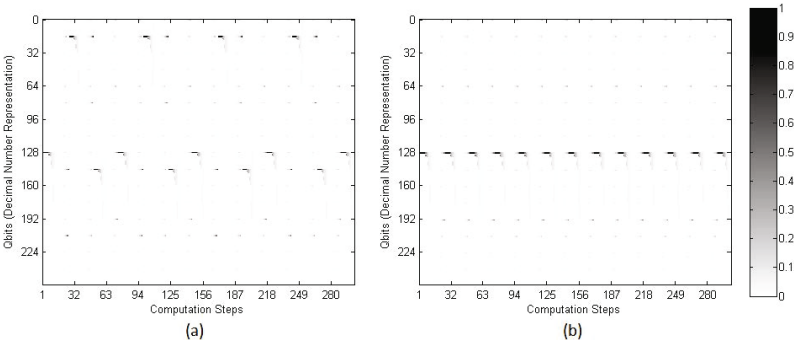

Fig. 3. Simulation of the evolution of four-cell QCAs. The initial state is $|10000000\rangle$. At evaluation phases a Hadamard gate acts on both qubits at each cell. (a) The rule is the one shown in Fig. 1(b). (b) The rule is the one shown in Fig. 1(c). Figure from [13].

4 Conclusion

Simulations of QCAs with two qubits per cell provided a strong indication for periodic evolution. Furthermore, the period of the probability distribution patterns varies with the evolution rule and the number of cells. This is an indication for possible ability to construct QCAs with desirable periods of evolution. If this is possible then many applications can be found for QCAs with two qubits per cell in quantum information processing.

The simulations, also revealed a rich behavior that stems from a high sensitivity on initial state and gate configuration at evaluation steps. This is a desirable property in computation. QCAs could be a promising architecture for the implementation of neural networks and CA using quantum computers. Another possibility of this architecture is the preparation of the state of a QR which may be used as the initial state for the application of a quantum algorithm.

References

1. von Neumann, J.: Theory of Self-Reproducing Automata. University of Illinois Press, Urbana IL (1966)
2. Karafyllidis, I., Thanailakis, A.: Ecological Modeling 99, 87 (1997)
3. Karafyllidis, I., Hagouel, P.I., Thanailakis, A., Neureuther, A.R.: IEEE Transactions on Semiconductor Manufacturing 13, 61 (2000)
4. Karafyllidis, I.: Biosystems 45, 1 (1998)
5. Bettelli, S., Serafini, L., Calarco, T.: arXiv:cs/0103009v3
6. De Raedt, H., Hams, A., Michielsen, K., Miyashita, S., Saito, K.: arXiv:quant-ph/0008015v2
7. Feynman, R.P.: International Journal of Physics 21, 467 (1982)
8. Ferguson, A.J., Cain, P.A., Williams, D.A., Briggs, A.D.: Physical Review A 65, 034303-1 (2002)
9. Twamley, J.: Physical Review A 67, 052318-1 (2003)
10. Grössing, G., Zeilinger, A.: Complex Systems 2, 197 (1988)

11. Meyer, D.A.: Physical Review E 55, 5261 (1997)
12. Karafyllidis, I.: Physics Letters A 320, 35–38 (2003)
13. Karafyllidis, I.: Physical Review A 70, 044301 (2004)
14. Karafyllidis, I.: IEEE Transactions on Circuits and Systems I 52, 1590-1596 (2005)
15. Karafyllidis, I.: Quantum Information Processing 2, 271 (2003)