2015 International Symposium on Nonlinear Theory and its Applications
NOLTA2015, Kowloon, Hong Kong, China, December 1-4, 2015

On Computations by a Brownian Cellular Automaton on a Triangular Space

Teijiro Isokawa†, Koji Ono†, Ferdinand Peper‡, and Nobuyuki Matsui†

†Graduate School of Engineering, University of Hyogo
2167 Shosha, Himeji, Hyogo, 671-2280, Japan
‡Center for Information and Neural Networks, National Institute of Information and Communications Technology, and Osaka University, 1-4 Yamadaoka, Suita, Osaka, 565-0871, Japan
Email: isokawa@eng.u-hyogo.ac.jp, peper@nict.go.jp, matsui@eng.u-hyogo.ac.jp

Abstract—This paper describes a Cellular Automaton (CA) in which cells are triangular, and computations are driven by fluctuations of cell configurations according to a stochastic search process. The proposed CA is computationally universal, which is proven by embedding computationally universal primitives, connections, and signals of a Brownian circuit onto this CA.

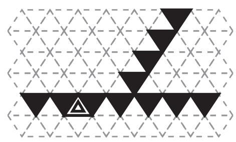

Figure 1: Cell space on triangular ACA

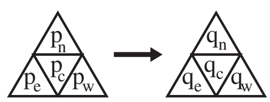

Figure 2: Transition on triangular ACA

1. Introduction

Cellular Automata (CA) have attracted increasing attention as architectures for computers with nanometer-scale devices (nano-computers), because their regular structures and local connectivity offer much potential for manufacturing based on molecular self-assembly [1, 2, 3, 4]. An obstacle to the realization of nanocomputers is the effect of noise and fluctuations in operating nanometer-scale devices, in which amplitudes of circuit signals could be comparable to those of noise and fluctuations. Assuring the normal operation of circuits will be difficult in the framework of traditional techniques, such as the suppression of noise or the correction of errors caused by noise.

For this reason, alternative approaches to circuit operations need consideration. One possible approach is to make use of noise as information carrier or—more indirectly—for the operations of circuits [5, 6], in a way that is sometimes found in biological systems [7]. When realized in terms of CA, noise-driven computation is described by the term *Brownian Cellular Automata (BCA)* [8]. BCA are a type of asynchronous CA, where certain local configurations propagate randomly in the cellular space, resembling Brownian motion. The BCA in [8] is proven to be computationally universal through the embedding of so-called Brownian circuits [9, 10] on the cell space. This BCA consists of rectangular identical cells, each of which has one of three states, whereby only three transition functions are required for establishing computational universality.

For realizing computational circuits by BCA on nanometer scales, the structure of cells could be different from rectangular, such as hexagonal or triangular, because molecules can be naturally configured on non-rectangular grids on nanometer scaled systems [11, 12]. There has also been research on asynchronous CAs with non-rectangular structures [13], but non-rectangular BCA have yet to attract efforts.

This paper presents a computational universal BCA, in which cells have a triangular structure. Computational universality in the proposed BCA is shown by embedding a set of primitive elements, connections, and signals of a Brownian circuit on the cell space, as in [9, 10]. The numbers of states and transition rules in this CA are 8 and 11, respectively.

2. Preliminaries

2.1. Asynchronous cellular automaton with triangular cells

The CA in this paper is a two-dimensional asynchronous CA of identical cells, each of which can assume one of a finite number of states at a time. A cell has a triangular shape, i.e., each cell is connected to three neighboring cells, and the cellular space is constructed by cells that are configured as triangles alternating in two different orientations, as shown in Fig. 1.

Cells undergo transitions in accordance with transition rules that operate on each cell and its direct three neighbors, as shown in Fig. 2. The output domain of the transition rules is also the input domain. In ACA, transitions of the cells occur at random times, independent of each other. Furthermore, it is assumed that neighboring cells of the cells being in transition never undergo transitions at the same time to prevent a situation in which such cells simultaneously write different states into the same location.

We assume that the transition rules are rotation symmetric, i.e., one transition rule has three rotated analogues.

*Koji Ono has been with Glory Limited, Japan, since 2011.

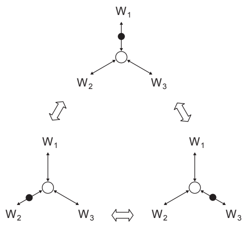

Figure 3: Hub and its possible transitions. A token is denoted by a black blob. Fluctuations cause a token to move between any of the Hub's three wires W_1, W_2, and W_3 in any order.

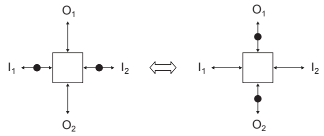

Figure 4: CJoin and its possible transitions. If there is a token on only one input wire (I_1 or I_2), this token remains pending until a signal arrives on the other wire. These two tokens will then result in one token on each of the two output wires O_1 and O_2.

Consequently, when we represent the transition in Fig. 2 as

$$(p_c, p_n, p_e, p_w) \to (q_c, q_n, q_e, q_w),$$

the following three rules also exist:

$$\begin{array}{lcl} (p_c, p_e, p_w, p_n) & \to & (q_c, q_e, q_w, q_n) \\ (p_c, p_w, p_n, p_e) & \to & (q_c, q_w, q_n, q_e) \\ (p_c, p_w, p_n, p_e) & \to & (q_c, q_w, q_n, q_e) \end{array}$$

2.2. Brownian circuits

A Brownian circuit is a circuit in which signals, represented by tokens, fluctuate forward and backward, and find their way from input to output through random fluctuations. The random search property allows signals to backtrack their way out of deadlocks, which may occur in token-based circuits due to waiting conditions that can not be satisfied. Computational universality in Brownian circuits can be achieved by a set of three primitives [9, 10].

The first primitive is the *Hub*, which contains three wires that are bidirectional (Fig. 3). There will be at most one signal at a time on any of the Hub's wires, and this signal can move to any of the wires due to its fluctuations.

The second primitive is the *Conservative Join (CJoin)*, which has two input and two output wires, all bi-directional (Fig. 4). The CJoin synchronizes two signals passing through it. Signals may fluctuate on the input wires, and when processed by the CJoin, they will be placed on the output wires where they may also fluctuate. The operation of the CJoin may also be reversed, and the forward / backward movement of the two signals through it may reoccur an unlimited number of times. Due to this bidirectionality, there is no distinction between the input and output wires of the CJoin, though we still speak of input and output, since the direction of the process is eventually forward.

The third primitive is the *Ratchet*, which restricts the movement of tokens through itself to one direction, thus effectively transforming a bidirectional wire into a unidirectional wire (Fig. 5). The ratchet is used to speed up searching in circuits. Since searching cannot backtrack over a ratchet, it will consume less time, but a Ratchet cannot be placed at positions that interfere with the search process in a circuit, so it should be carefully applied.

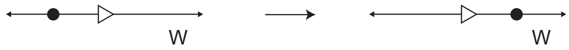

Figure 5: Ratchet and its transition. The token on wire W may fluctuate both before and after the ratchet, but once it moves over the ratchet it cannot return. The ratchet thus imposes a direction on a bi-directional wire.

Symbol								
State	0	1	2	3	4	5	6	7

Figure 6: Symbols representing a cell's state

We will prove that the proposed cellular automaton is computationally universal by showing that it can construct all the three primitives of a Brownian circuit as well as their connection paths and signals.

3. Universality in triangular BCA

The proposed BCA uses eight states for constructing all the elements in a Brownian circuit. The symbols associated with their states are shown in Fig. 6, and the 11 transition rules for updating cells are shown in Fig. 7. Subsequent sections describe how to embed each of primitives onto cellular space.

3.1. Signal and signal path

Signals and signal paths are the basic elements in a Brownian circuit. Figure 8 shows an implementation of signal and signal paths on triangular BCA. Paths are represented by alignments of cells with state 1, and a signal on the path is represented by a cell in state 2. The signal on the path flows by applying transition rule #1 (Fig. 7) to one of the cells neighboring to the signal cell. Due to the neighborhood defined in the CA, the signal appears on alternate sides of a path when it proceeds. Signals are undirected,

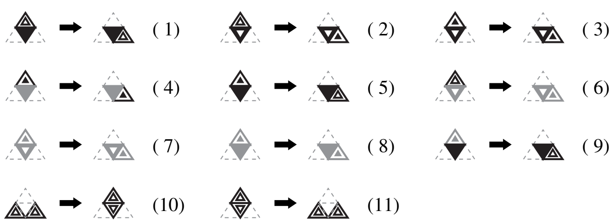

Figure 7: 11 transition rules used in this paper

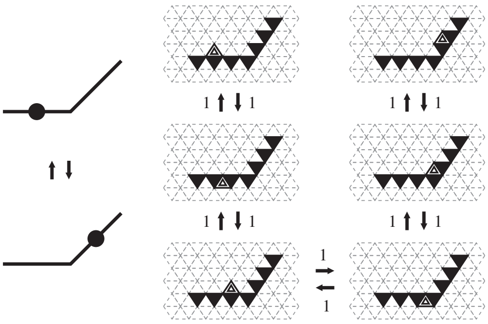

Figure 8: Signal and signal paths on triangular BCA where the number beside an arrow denotes the number of the transition rule that is applied

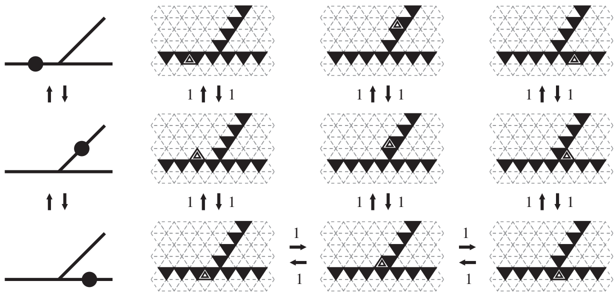

Figure 9: Hub configuration and its transition with a signal

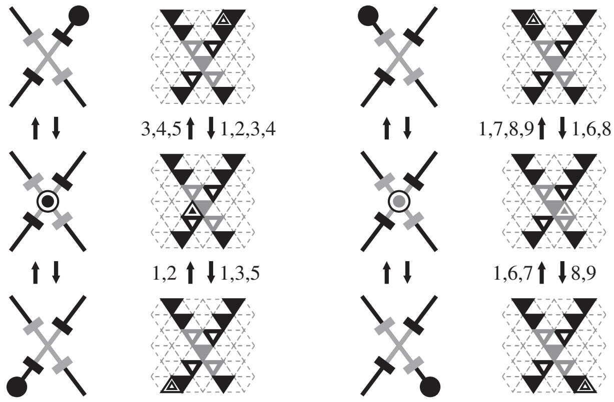

Figure 10: Crossing of signals through a crossing element

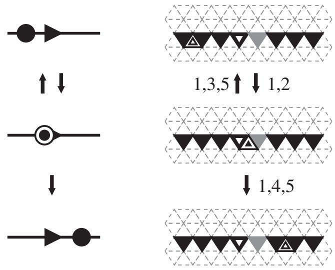

Figure 11: Ratchet configuration

thus the application of rule #1 results in a signal going forward or backward, depending on which cells are updated first (see the right side hand of Fig. 8). Paths may make a turn of 60°, as shown in Fig. 8, by the configuration of path cells, and the signal will proceed in this case according to rule #1.

3.2. Hub, crossing elements, and ratchet

Figure 9 shows a configuration of a Hub and transitions of signals on a hub. The hub can be simply constructed by connecting signal paths. A signal may go into the crossing point of a hub and may go out from either paths connected with the crossing point.

The crossing of paths is often necessary to construct useful circuits, and its implementation is also necessary. Other BCAs can simply implement this functionality due to sufficient information being available to encode the direction of a signal in the updating of a cell's state for the crossing element. However, cells in the triangular BCA have fewer neighborhood cells than rectangular BCA, thus another way to encode the direction of a signal in a crossing is necessary. In the proposed BCA, the direction of a signal is encoded in its state. The configurations of a crossing element are shown in Fig. 10, where signals coming from two directions can travel through the crossing element in an appropriate way.

The wires attached to a crossing element are equipped with *gates* for changing the state of an arriving signal. The gate cells are cells with state 4 or 6. The state of a signal at a gate cell with state 4 (resp. 6) changes to 5 (resp. 7). A signal with the state 5 (resp. 7) can only exit from the gate via a gate in state 4 (resp. 6). This way of encoding enables us to control appropriate directions of signals.

This technique can also be applied to implement a ratchet element that passes a signal in one way. Figure 11 shows a configuration of a ratchet with a signal. The signal coming from the east goes out to the west of the cell with state 3, and never goes back to the east due to no transition rules being available for going back.

3.3. Conservative Join

The Conservative Join (CJoin) accepts two incoming signals at its input terminals and redirects them to its output ports. Figure 12 shows a configuration of the CJoin element. When two signals get together at the center of the CJoin element, they are processed by this element. These signals proceed to their output ports after four stages of processing. It is necessary to align the phase of signals on a path after the output from a CJoin element.

4. Conclusion

This paper presents an asynchronous cellular automaton of which cells are triangular. This CA is computational

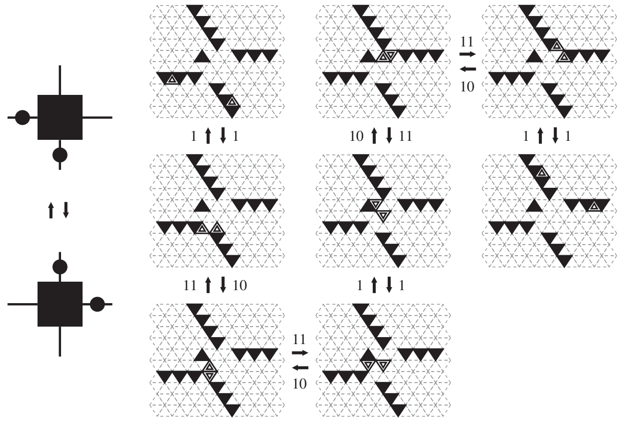

Figure 12: Configuration and transitions of signals in a Conservative Join

universal, because a type of asynchronous circuits called Brownian circuit can be embedded onto it. The number of states for a cell is 8 and the number of transition rules is 11. These are larger numbers (thus more complex) as compared to rectangular BCAs, so the proposed triangular BCA is more complex than the rectangular BCAs published thus far. Developing simpler constructions for triangular CA remains for our future work.

Acknowledgments

This work was supported by a Grant-in-Aid for Scientific Research on Innovative Areas "Molecular Robotics" (No. 25104523) of The Ministry of Education, Culture, Sports, Science, and Technology, Japan.

References

[1] M. Biafore, "Cellular automata for nanometer-scale computation," *Physica D*, vol.70, pp.415–433, 1994.

[2] L. J. K. Durbeck and N. J. Macias, "The cell matrix: an architecture for nanocomputing," *Nanotechnology*, vol.12, pp.217–230, 2001.

[3] F. Peper, J. Lee, S. Adachi, and S. Mashiko, "Laying out circuits on asynchronous cellular arrays: a step towards feasible nanocomputers?," *Nanotechnology*, vol.14, no.4, pp.469–485, 2003.

[4] F. Peper, J. Lee, and T. Isokawa, "Cellular nanocomputers: a focused review," *International Journal of Nanotechnology and Molecular Computation*, vol.1, pp.33–49, 2009.

[5] L. B. Kish, "Thermal noise driven computing," *Applied Physics Letters*, vol.89, no.14, pp.144104–1–3, 2006.

[6] S. Dasmahapatra, J. Werner, and K. P. Zauner, "Noise as a computational resource," *International Journal of Unconventional Computing*, vol.2, no.4, pp.305–319, 2006.

[7] T. Yanagida, M. Ueda, T. Murata, S. Esaki, and Y. Ishii, "Brownian motion, fluctuation and life," *Biosystems*, vol.88, no.3, pp.228–242, 2007.

[8] J. Lee and F. Peper, "Efficient computation in brownian cellular automata," *Proc. of 4th International Workshop on Natural Computing*, pp.47–56, 2009.

[9] J. Lee and F. Peper, "On Brownian cellular automata," *Proceedings of Automata 2008*, pp.278–291, 2008.

[10] F. Peper, J. Lee, J. Carmona, J. Cortadella, and K. Morita. "Brownian circuits: Fundamentals," *ACM Journal on Emerging Technologies in Computing Systems*, vol. 9, no. 1, article #3, pp.1–24, 2013.

[11] A. J. Heinrich, C. P. Lutz, J. A. Gupta, and D. M. Eigler, "Molecule Cascades," *Science*, vol.298, no.5597, pp.1381–1387, 2002.

[12] A. Bandyopadhyay, R. Pati, S. Sahu, F. Peper, and D. Fujita, "Massively parallel computing on an organic molecular layer," *Nature Physics*, vol.6, no.5, pp.369–375, 2010.

[13] S. Adachi, F. Peper, and J. Lee, "Universality of Hexagonal Asynchronous Totalistic Cellular Automata," *Proceedings of 8th International Conference on Cellular Automata for Research and Industry (ACRI2008)*, LNCS 5191, pp.91–100, 2008.