On Molecular Implementations of Cellular Automata

Ferdinand Peper[1], Anirban Bandyopadhyay[2], Hiroshi Oono[3], Satyajit Sahu[2], Ranjit Pati[4],
Subrata Ghosh[2], Teijiro Isokawa[3], and Daisuke Fujita[2]

[1] Nano ICT Group, National Institute of Information and Communications Technology, 588-2 Iwaoka,
Nishi-ku, Kobe, 651-2492 Japan, peper@nict.go.jp
[2] Advanced Nano Characterization Center, National Institute for Materials Science, 1-2-1 Sengen,
Tsukuba, Ibaraki, 305-0047 Japan
[3] Division of Computer Engineering, University of Hyogo, 2167 Shosha, Himeji, 671-2201 Japan
[4] Department of Physics, Michigan Technological University, Houghton, Michigan, 49931 USA

Abstract: Molecular realizations of Cellular Automata that exhibit computational behavior have been the holy grail of nanocomputer architectures. This paper discusses a recently discovered molecular Cellular Automaton, and compares its features with those of conventional models. We find that the interaction rules in molecular Cellular Automata tend to be of a mixed variety, ranging from those with long-distance interactions to the more conventional direct-neighborhood type of rules. The probabilities according to which rules are applied in the molecular Cellular Automaton tend to be influenced by the patterns on the cellular space, resulting in much more volatile dynamics than in conventional models.

Key–Words: Molecular Electronics, Cellular Automata, Architectures, Nanotechnology

1 Introduction

Cellular Automata have their origin in the 1950's investigations of John von Neumann on the logical organization and principles of biological self-reproduction. Over many decades, they have been extremely useful for the simulation of natural phenomena, as formal models of computation, etc. In recent years they have attracted a different kind of attention: from nanotechnologists who aim to build computers shaped after Cellular Automata. Their regular structures are ideal from a fabrication point of view, because they can be formed through bottom-up processes like molecular self-assembly. As such they are a viable alternative to today's optical lithography processes used to fabricate computer chips, which—with chips' ever-decreasing feature sizes—are running into the diffraction limit of light.

Apart from the advantage that the regular structure of a Cellular Automaton gives, there is also a disadvantage: computing on such structures is significantly more complex than on conventional computers, since all that is available are simple cells with the complexity of finite automata, all connected to only their direct neighbors. Information can only flow by a limited distance in each computing step, and operations are bound by the limited complexity of the cells. Rather than the instruction-based mode of operation in conventional computers, Cellular Automata operate by integrating the operations of all cells together, such that a desired computation emerges.

The attempts to compute on Cellular Automata have produced some useful models [12], in which cell complexity—as measured by the number of states per cell and the number of transition rules to update the states—has gradually decreased over the years. A low complexity of cells is important from an implementation point of view, because it translates into better physical realizability. Still, there remains a gap between the idealized models of Cellular Automata and their physical realizations. While traditional Cellular Automata allow cells to influence only cells that are within a fixed neighborhood, in physical realizations the interactions between cells act at longer distances, and they may even be more flexible due to the propensity of interactions to change with the updates of states. A recent publication on the implementation of a Cellular Automaton on a molecular layer [3] provides a good illustration of this. In this paper, we discuss the differences and similarities of this molecular Cellular Automata with traditional Cellular Automata, and possible ways to bridge the gap between them.

Cellular Automata have been subject to attempts of physical realization for decades. The Cellular Automaton Machine (CAM) by Toffoli and colleagues [13] attracted much attention in the 1980's. Less know, but in the same decade, are the efforts by the chemist Forrest L. Carter to realize Cellular Automata in which chemical bonds between molecules are used to store states of cells [5]. The 1990's gave birth to Quantum Dot Cellular Automata, in which cells are implemented as four or five dots containing electrons in certain configurations [10]. A Cellular Automaton proposed with physical realizations in mind is in [6]; each cell can be programmed as logic gates, wires, etc., but since cells are quite complex in this model, they may be less suitable for molecular realizations. One of the first realizations of molecular Cellular Automata [8] uses CO molecules arranged on a copper surface to conduct simple logical operations by moving molecules around on the surface.

This paper is organized as follows. Section 2 describes the basic model for the molecular cellular automaton. This is followed by Section 3, in which the rules of interaction between cells are discussed and compared with the transition rules of conventional Cellular Automata. A new phenomenon in Cellular Automata, i.e., the formation of networks on the cellular space, is discussed in Section 4. This paper finishes with Conclusions.

2 Molecular Cellular Automaton

The molecular Cellular Automaton reported on in [3] has a complex dynamics emerging from its structure. At the bottom is a gold Au(111) surface on which two layers of a small molecule reside. This molecule, 2,3-Dichloro-5,6-dicyano-1,4-benzoquinone (DDQ), is asymmetric, resulting in it being an electric dipole, which is responsible for its "rich" behavior. Each of these layers organize themselves according to a hexagonal grid, and the molecules of the top layer settle above the spaces between molecules in the bottom layer. Each molecule can be in four states, which are numbered 0, 1, 2, and 3. Depending on its state, a molecule has a different electrical charge or a different distribution of charge among its chemical bonds. State 0 is the neutral state, in which the charge of a molecule is neutral. It is comparable with the so-called *background state* in conventional Cellular Automata. State 1 is a state in which a molecule contains one excess electron. State 2 is again a neutral state, but electrons are distributed on the molecules bonds in a different way than in a molecule in state 0. The different charge distribution results in a state-2 molecule being more stretched in shape, whereas a state-0 molecule is more curved along its axis. Finally, we have state 3, which is characteristic for a molecule with two excess electrons.

When a molecule in the bottom layer is in state 1 or 3, it is electrically charged, and this charge will eventually leak away to the gold surface. Bottom layers like those of DDQ molecules tend to have a dynamic behavior in which molecules self-organize into convex macro-cells of similar states, but more refined dynamics in which individual molecules behave differently from surrounding molecules is unlikely to emerge, even though there is a significant level of fluctuation at the room temperature at which the experiments in [3] have been conducted.

A very different behavior is observed from the molecules at the top layer. Electrically isolated from the gold surface by the bottom layer, the top-layer molecules will retain their electrons, and only intermolecular interactions on the top layer will move the electrons around. This movement of charges is at the base of the rich dynamics that can be observed in the top layer.

3 Rules of Interaction

Like in conventional Cellular Automata there are rules of interaction between cells, but here is where the similarity stops. In the molecular Cellular Automaton, rules are mostly driven by electrical charges, and the forces involved act on larger distances than is usual for conventional models.

The first rule, described in [3] and called *Rule 1*, states that negatively-charged molecules that are distributed over the molecular layer within a distance of about 15 molecules (which corresponds to approximately 15 nm), will influence each other by forming a so-called *Positive Pseudo Charge (PPC)* in their center. In the case of four molecules in state 1, for example, arranged as a square, there will be a PPC at the cross points of the two diagonals. This PPC will attract molecules in state 1 or in state 3. If there are many molecules within the 15 nm distance range, there will be a complicated system or forces acting on all molecules. As a result, electrons from negatively charged molecules will tunnel to neighboring molecules in the direction of a PPC. We will say in this case that the molecules in state 1 or 3 move over the surface, since this looks like what happens to an observer. Simulations of the rules can be found in the video at [4].

The interactions between molecules involve different charges, i.e., molecules in state 1 and in state 3, and the size of the charges determines the speed by which they move to neighboring molecules. This phenomenon, observed as *Rule 2* in [3], has no counterpart in conventional Cellular Automata, since the latter tend to update their cells at equally timed steps (in case of synchronous models) or at random steps (in case of asynchronous models).

The third rule described in [3] and called *Rule 3*, brings us back to more familiar territory, since it closely resembles the so-called *Billiard Ball model (BBM)* [7], proposed in 1982. This model is based on elastic collisions of identical balls that move diagonally over a square grid. When combined with fixed rail cushion segments positioned appropriately, the model gives rise to structures to deflect a signal into a different direction, to delay it, to shift it a number of cells, to cross it with another signal, and to let it operate a switch in combination with another signal. Together these constructions are sufficient to achieve universal computation. In the molecular cellular array, two molecules in state 1 or 3 that move on a collision course will behave very similar to balls in the BBM, except that in the molecular case we have a hexagonal grid. A hexagonal grid is no impediment to achieve universality, as the results in [2] and [11] show. What is still missing from the molecular model currently, however, is that no fixed rail cushion segments have yet been observed, so we are just a few steps short of universality.

The fourth rule, called *Rule 4*, is a rule concerned with state 2 molecules. When a molecule in state 1 or 3 loses its electron(s) to a neighboring molecule, it becomes electrically neutral again. Rule 4 states that of the choice between the two neutral states 0 and 2, the molecule will always select to be in state 2 when becoming neutral. In other words, when a molecule moves over the surface, it leaves a trail of state 2 molecules. Rule 4 contains yet another action: when state 2 molecules cluster together, they will be thinned, reverting to state 0, until only isolated molecules in state 2 remain. The effect of this action is that trails will eventually disappear mostly. Behavior incurred by Rule 4 is quite common in conventional Cellular Automata, especially when the timing model is asynchronous [1]. A trail behind a signal is usually cleared up, because it would otherwise block the transmission of subsequent signals on the same path.

The fifth rule, called *Rule 5*, states that groups of molecules in states 1 or 3 behave like a single molecule in that state, when the groups have certain convex shapes. This appears proof of some degree of entanglement of neighboring molecules. In combination with Rules 1 and 3, it results in groups of molecules colliding like in the BBM model. There is one exception though, the BBM collision behavior only occurs when the contact area of the colliding groups comprise of not more than 1 molecule. A larger contact area will result in the groups breaking up. The group-like behavior of molecules is quite uncommon in conventional Cellular Automata, in which dynamics tend to be based on individual cells, rather than collectivism. It remains to be seen to what extent this type of behavior can be exploited in computation.

The sixth rule, called *Rule 6*, concerns groups of molecules that have concave shapes and that are comprised of states 1 and 3. There is a tendency of such groups to fill in the concavities in order to achieve convex shapes that are as symmetric as possible. This process is facilitated by the excess electrons in state 3 molecules. Distributing one of the two electrons of a molecule in state 3 to a state 0 molecule results in two molecules in state 1, thus increasing the number of non-neutral molecules. The reverse is also possible if symmetry and convexity is achieved by it: in that case one electron from a state 1 molecule may jump to another state 1 molecule, thus resulting in a state 0 molecule and a state 3 molecule. There appears to be some randomness involved when group shapes can be evolved in different ways. While this may not be unusual in conventional Cellular Automata, it may be difficult to exploit the phenomena emerging from Rule 6.

Rules 1 to 6 are the basic rules in the molecular cellular automaton, as far as they have been observed in [3]. There are, however, even more rules, that make the molecular Cellular Automaton very different from conventional models.

4 Formation of Molecular Networks

When a molecule is in state 1 or 3, the excess electrons residing on its bonds will, in combination with the asymmetric structure of the molecule, induce molecules to tilt sideways. This tilt tends to depend on the density of charge in the local area of the surface, and as a result some molecules on the top-layer will be nearer to each other than others. The probability that an electron tunnels from one molecule to a neighboring molecule is strongly dependent on the distance between the molecules: the probability exponentially decreases with increasing distance. In fact, this strong dependence is the principle according to which Scanning Tunneling Microscopes (STM) work. When the probability of tunneling from a molecule to another molecule is much larger than average, there is effectively a connection between the molecules, while molecules with below-average tunneling probability lack such a connection. Taken together, the connections form a network along which molecules exchange electrons. The topology of this network, being dependent on tilting of molecules, is influenced by the charge density in a local area. Eight different topologies have been observed [3], in which molecules have connections with several numbers of neighbors, varying from two to six. When a molecule has only two connections, there are only two neighbors an electron can go to, and this influences the type of rule that is dominant. Rule 5 is dominant in this case, which is not surprising, because the low mobility of electrons favors the group-like behavior of this rule. Connectivity with six neighbors, on the other hand, will allow electrons to flow in many different directions, and, not surprisingly, the collision-based behavior in Rule 3 is dominant in this case.

The dependence of rule dominance on the density of states 1 and 3 in a local area (i.e., on the local charge density) is very uncommon in conventional Cellular Automata, which tend to not discriminate between rules. The flexibility of the molecular Cellular Automaton in this aspect facilitates a rough degree of controllability over rule dominance. The reconfigurability of the molecular Cellular Automaton may be helpful in laying out circuits on the molecular layer, and thus implementing certain functionalities, like arithmetic, on the layer. It allows an effective use of the (rather restricted) molecular hardware, whereas in a conventional Cellular Automaton comparably useful dynamics may only be in reach through cells with much higher complexity.

The local area over which a charge density is evaluated, as mentioned above, may have a variety of shapes, but all shapes appear to have in common that they are convex. These shapes can be organized as a Voronoi diagram, so in effect, the molecular Cellular Automaton has the ability to classify areas with respect to their charge densities into a Voronoi tesselation. This is a natural dynamics inherent in the molecular top layer.

5 Conclusions

The molecular Cellular Automaton in [3] is based on a very simple molecule, yet it generates surprisingly rich behavior. Whereas conventional Cellular Automata influence states of their cells through the application of transition rules without any rule being favored over others, the molecular Cellular Automaton has a subtle mechanism in place to change the dominance of rules based on the formation of molecular networks, which on their turn depend on densities of cells states 1 and 3. This makes the cell space less homogeneous and more malleable than in conventional Cellular Automata, with improved programmability as a possible benefit.

The structure of Cellular Automata is usually defined strictly, and in implementations this requires special measures when for some reason defects occur in cells. Strategies to deal with defects cover a range of solutions. One such solution is to use hardware redundancy, such that when a cell breaks down, some backup cell can be used instead, but this results in an increased complexity of cells, which remains unused most of the time. Another solution is to isolate defective cells and route information around them, but this tends to require an increased number of cell states and transition rules [9]. The molecular Cellular Automaton has its own solution: when a molecule disappears for some reason from the molecular layer, a new molecule floats in place of the resulting hole in the course of approximately one minute, as has been observed [3]. *Self-healing* is thus a natural behavior for the molecular Cellular Automaton.

The unique characteristics of the molecular Cellular Automaton may be difficult to model in the conventional automaton-based framework (witness for example Rule 1), but it also creates new opportunities, like a dynamics that is much richer than one would expect from the simple underlying molecular hardware. The promise of efficient physical realizations of Cellular Automata is now greater than ever, but it comes with new challenges to effectively use this new type of hardware.

Acknowledgements: Authors acknowledge JSPS Grants in Aid for Young Scientists (A) for 2009-2011, Grant number 21681015 (Govt of Japan). F.P. acknowledges *Incentive* funding of the National Institute of Information and Communications Technology for 2010-2011. R.P. acknowledges National Science Foundation (NSF) Award number ECCS-0643420.

References:

[1] S. Adachi, J. Lee, and F. Peper, “On Signals in Asynchronous Cellular Spaces”, *IEICE Trans. on Information and Systems*, Vol. E87-D, No. 3, pp. 657–668, 2004.

[2] A. Adamatzky, A. Wuensche, and B. De Lacy Costello, “Glider-based computing in reaction-diffusion hexagonal cellular automata”, *Chaos, Solitons & Fractals*, Vol. 27, No. 2, pp. 287–295, 2006.

[3] 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.

[4] A. Bandyopadhyay, R. Pati, S. Sahu, F. Peper, and D. Fujita, “Massively parallel computing on an organic molecular layer”, *Nature Physics*, Supplementary Movie 4, www.nature.com/nphys/journal/v6/n5/extref/nphys1636-s5.wmv, 2010.

[5] F.L. Carter, “The molecular device computer: point of departure for large scale cellular automata”, *Physica D*, Vol. 10, No. 1-2, pp. 175–194, 1984.

[6] L.J.K. Durbeck and N.J. Macias, "The Cell Matrix: an Architecture for Nanocomputing", *Nanotechnology*, Vol. 12, No. 3, pp. 217–230, 2001.

[7] E. Fredkin and T. Toffoli, "Conservative Logic", *International Journal of Theoretical Physics*, Vol. 21, Nos. 3–4, pp. 219–253, 1982.

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

[9] T. Isokawa, S. Kowada, Y. Takada, F. Peper, N. Kamiura and N. Matsui, "Defect-Tolerance in Cellular Nanocomputers", *New Generation Computing*, Vol. 25, No. 2, pp. 171–199, 2007.

[10] C.S. Lent, P.S. Tougaw, W. Porod, and G.H. Bernstein, "Quantum Cellular Automata", *Nanotechnology*, Vol. 4, No. 1, pp. 49–57, 1993.

[11] K. Morita, M. Margenstern, and K. Imai, "Universality of Reversible Hexagonal Cellular Automata", *MFCS'98 Satelite Workshop on Frontiers between Decidability and Undecidability*, Brno, Czech Republic, 1998.

[12] 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.

[13] T. Toffoli, "CAM: A High-Performance Cellular Automaton Machine", *Physica D*, Vol. 10, No. 1-2, pp. 195–205, 1984.