Bull. Nov. Comp. Center, Comp. Science, 25 (2006), 1–12

Synchronous versus asynchronous cellular automata for simulating nano-systems kinetics*

Olga L. Bandman

Abstract. A class of asynchronous cellular automata (ACA) whose evolution simulates physicochemical kinetics of nano-systems is defined. It is characterized by multicell probabilistic transition rules and stochastic character of their application. To simulate real processes in real time ACAs should have huge cellular arrays and a very long evolution. So, the problem of parallel implementation of ACA algorithms is actual. But as distinct from a synchronous case, it has not any good and simple solution. To overcome this difficulty, a method of ACA approximation by a block-synchronous cellular automata (BCA) is proposed, analyzed and experimentally studied.

1. Introduction

Cellular Automata (CA) have been widely recognized to be capable of modeling nonlinear spatial processes in the 80-s, when a series of fundamental works were published [1, 2]. Gas-Lattice [3], reaction-diffusion [4] and Lattice-Boltzmann [3] CA models are now well studied and used, although conventional methods based on numerical solution of PDEs also exist. A little later, CA models were proposed for simulation of the phenomena which cannot be described with PDEs because of their nonlinear and discrete character. The bright examples are "movable cellular automata" [5] modeling deformation and cracks in solid bodies and CA-models of percolation beds control [6]. The above mentioned CA deals with the so-called "stylized" or abstract particles endowed or not with a velocity vector as cell states, whose transition rules provide the CA evolution identical or similar to the phenomena under simulation. The next stage of CA simulation development is manifested by using CA for mimicking the behavior of real atoms and molecules in real time, i.e., for simulating kinetics on nano-level. It has been stimulated by an increase in computation power, which nowadays allows one to observe movements and interactions of molecules on the surface of square microns during some microseconds on the personal computer monitor. Of course, physicists or chemists who study nano-system kinetics want to increase these parameters. So, the problem of computation efficiency improvement is of urgent need.

*Supported by Russian Academy of Sciences, Basic Research Program of No. 13.15 (2006).

There are two ways in searching for the problem solution. The first is to optimize the CA transition rules and the mode of their application. The second is to increase the computer power by using parallel implementation on multi-processor systems. In both approaches, the success can be achieved if specific features of the CA-models used are exploited as much as possible. Most of simulations in nano-system kinetics use a restricted class of CA, whose main features are as follows:

- Most of transition rules prescribe changing a group (one or more) cell states at once mimicking an act of adsorption, diffusion, sublimation or chemical transformation.

- The mode of a CA operation is asynchronous, which means that for each time step only one updating is performed to one randomly chosen place of the cellular array.

- The updating of cell states obeys probabilistic transition rules applied in random order.

Such a stochastic character of CA functioning brought about a term Monte Carlo simulation, which is commonly used among nano-physicists and nano-chemists. The CA-models with the above properties are referred to as *kinetic asynchronous CA* (ACA). The reason for using the terminology from CA-theory is that the methods proposed are based on CA-theory and its extensions, particularly, the formalisms used follow a formal model called Parallel Substitution Algorithm, which is a generalization of the classical CA [11].

The class of kinetic ACA models being defined, the ways of reducing simulation time are analyzed with the inference that the most universal way of simulation time reduction is approximation of an ACA with a block-synchronous CA. The reason is that as distinct ACA, synchronous CA from allow formation of packages for interprocess data exchange when implemented in multiprocessor systems, which yields almost ideal parallel efficiency. Since ACA have a very dense exchange, this makes the parallelization of their direct functioning meaningless. It follows therefrom that the need to develop approximation algorithms and evaluate their efficiency is relevant, which is the aim of the paper.

In Section 2, the class of kinetic ACA is introduced formally and its important features are outlined. In Section 3, the algorithm of ACA approximation by a block-synchronous CA is presented and approximation error is assessed.

2. Properties of kinetic asynchronous cellular automata

Kinetic ACA simulate phenomena consisting of sets of elementary action, directly mimicking physical movements or chemical transformations of atoms on a crystalline lattice [7,9]. In the ACA, the lattice corresponds to a cellular space Ω, which is a set of cells, a cell being a pair (a, m), where $a \in A$ is a symbol of state alphabet A, $m \in M$ ia a cell name. The following alphabets are mainly used in kinetic ACA:

- Boolean alphabet $A = \{0, 1\}$, representing existence or absence of a certain particle in a cell.
- A set of Boolean vectors $A = \{(v_1, \dots, v_b) : v_k \in \{0, 1\}\}$, whose k-th component represents a particle moving in the direction of the k-th neighboring cell.
- A set of characters or symbols representing notations of atoms or molecules (Al, CO, H_2, etc.).
- A set of integers, when several molecules are allowed to be allocated in a single site.

The naming set $M = \{(i, j, k) : \ i = 0, \dots, I; \ j = 0, \dots, J; \ k = 0, \dots, K\}$, is a set of integer coordinates indicating the place where a particle (or some particles) may be allocated. There is no restriction on the naming set structure. Among the most popular are Cartesian arrays, the arrays based on crystalline and Delonet lattices, as well as those providing a structural isotropy. For simplicity, when the naming set type is not specified, a single symbol m is used instead of a set of coordinates.

On the naming set M, a naming function $\varphi : M \to M$ may be defined. If $m' = \varphi(m)$, then m' is a neighbor of a cell named m. A set of naming functions determines a *template*

$$T(m) = \{m, \varphi_1(m), \dots, \varphi_q(m)\}, \tag{1}$$

which enumerates the neighbors of a given cell m. The set of cells with the names from $T(m)$

$$S(m) = \{(v_0, m), (v_1, \varphi_1(m)), \dots, (v_q, \varphi_q(m))\} \tag{2}$$

is called *local configuration*, $T(m)$ being its *underlying template*. The cell $(v_0, m) \in S(m)$ is further referred to as *reference cell* for $S(m)$.

Two local configurations $S(m)$ and $S'(m)$ with the same reference cell, whose underlying templates are written in the form of substitution

$$\vartheta(m) : S(m) \to S'(m) \tag{3}$$

represent an elementary act of cellular array updating. The underlying templates of the right-hand side and the left-hand side of (3) are in the following relation:

$$T'(m) \subseteq T(m). \tag{4}$$

The cell states in the right-hand side of (3)

$$S'(m) = \{(u_0, m), (u_1, \varphi_1(m)), \ldots, (u_p, \varphi_p(m))\}, \quad p \leq q,$$

are values of the transition functions, i.e.,

$$u_k = f_k(v_0, v_1, \ldots, v_q), \quad k = 0, 1, \ldots, p. \tag{5}$$

When the relation between underlying templates is given as a strong inclusion, some states of a subset of cells $S'' \subset S(m)$ remain unchanged, serving only as variables in (5) and being referred to as *context* in the substitution.

In kinetic ACA, the substitutions of the form of (3) are to simulate elementary acts of physical-chemical nano-processes. Some simple typical examples with $A = \{a, b, c, \ldots\}$, $M = \text{Gas} \cup \text{Solid}$, are as follows:

- *Adsorption.* A molecule a is adsorbed from a gas $\Omega' = \{(a, g) : g \in \text{Gas}\}$ to an empty site on a solid surface $\Omega = \{(a, m) : m \in \text{Solid}\}$ with probability p_a

$$\vartheta_a : \{(\emptyset, m)(a, g)\} \xrightarrow{p_a} \{(a, m)(\emptyset, g)\}.$$

- *Sublimation.* A molecule b is desorbed from a site on a solid surface to a gas with probability p_b

$$\vartheta_a : \{(a, m)(\emptyset, g)\} \xrightarrow{p_b} \{(\emptyset, m)(a, g)\}.$$

- *Reaction.* If the molecules a and b occur in the neighboring sites on the surface, they react forming a molecule ab, which outgoes to a gas with probability p_{ab}

$$\vartheta_{ab} : \{(a, m)(b, \varphi(m))(\emptyset, g) \xrightarrow{p_{ab}} \{(\emptyset, m)(\emptyset, \varphi(m))(ab, g)\}.$$

- *Diffusion.* If a molecule occurs in the neighborhood of an empty site, it moves there with probability p_d

$$\vartheta_d : \{(a, m)(\emptyset, \varphi(m)) \xrightarrow{p_d} \{(\emptyset, m)(a, \varphi(m))\}.$$

Besides the above elementary substitutions, many others more complicated are also used, for example, such ones that act in a 3D space, or have an extended neighborhood, or use certain conditions as context parts, etc. Moreover, most of real processes consist of many elementary acts. This fact is reflected in ACA transition rules by superposition of several elementary substitutions as follows:

$$\theta(m) = \vartheta_s(\vartheta_{s-1}(\ldots,(\vartheta_1(m)))), \tag{6}$$

which is further referred to as *local operator.* Its underlying template

$$T_\theta(m) = T'_1(m) \cup \cdots \cup T'_s(m). \tag{7}$$

is a union of those of substitutions included.

A substitution $\vartheta(m) : S(m) \to S'(m)$ is said to be applicable to a cell named $m \in M$ if

$$S(m) \in \Omega. \tag{8}$$

If (8) is not true, nothing happens, and an application attempt is considered to fail. Otherwise, transition function (5) is computed and the states of cells in $S'(m)$ are replaced by the obtained values. When it is done for all $\vartheta(m)$ and all $m \in M$, $\Omega(t)$ is transferred to the next *global state* $\Phi(\Omega(t)) = \Omega(t{+}1)$, Φ being called a *global operator.* An application of a global operator is further referred to as an *iteration.*

The sequence

$$\Omega(0),\ \Omega(1),\ \ldots,\ \Omega(t),\ \ldots$$

is referred to as CA *evolution.* A set of all possible evolutions of ACA $= \langle A, M, \theta\rangle$, starting with the same $\Omega(0) \in A \times M$ is denoted as $\Gamma(\Omega(0))$. Its cardinality is enormous,

$$|\Gamma| = |M|! \times |C|, \tag{9}$$

where $|M|!$ is the number of permutations of $|M|$ items, $|C|$ is the amount of probabilistic choices in the transition rules.

There are different modes of space-time distribution of the local operator application to perform a transition from $\Omega(t)$ to $\Omega(t+1)$. The most important of them are as follows:

1. *Synchronous mode.* In all cells, when computing next states, transition functions (5) use the state values of the neighboring cells $(v_l, \varphi_l(m)) \in \Omega(t)$. The transition to the next state values occurs after all transition functions in all cells are computed.
2. *Asynchronous mode.* Each cell transits to its next state just after a local operator is applied to it, a transition function being computed using state values of the neighboring cells at the moment. Cells for the local operator application are chosen at random.

The main difference between synchronous and asynchronous modes of operation is in the fact that in the synchronous mode, transition function arguments in all cells at the t-th iteration are the states of cells from $\Omega(t)$, while in the asynchronous mode, half of the whole amount of arguments used in transition functions at the t-th iteration are states of the cells which have already transited to the next state and belong to $\Omega(t+1)$. This is the reason why two CA having equal A, M, Φ and starting with the same $\Omega(0)$ may have quite different evolutions. The following example shows it convincingly.

Example 1. Simulation of wave propagation using the HPP-model. The HPP-Gas is a Lattice-Gas model known as the first and unsuccessful model for hydrodynamics, is considered to be useful for simulating acoustic and electro-magnetic waves [12]. The CA is assigned by the naming set $M = \{(i,j) : i,j = 0,\ldots,N\}$, and the alphabet $A = \{\boldsymbol{v} = (s_1, s_2, s_3, s_4) : \ s_l \in \{0,1\}\}$ which is a set of Boolean vectors of length 4, where $s_l = 1$ means that in the corresponding cell there exists a particle moving in the direction of the l-th neighbor with unit speed. The neighbors are numerated according to $T(i,j) = (i,j), (i-1,j), (i,j+1), (i+1,j), (i,j-1)$. The operation is synchronous, each time being divided into two phases: propagation and collision. In the propagation phase, each particle moves one cell forward following the direction of its speed, in the collision phase, two particles turn to perpendicular directions if they occur to meet. The local operator is $\theta(m) = \vartheta_1(\vartheta_2(\vartheta_3(m)))$, where

$$\begin{array}{ll}
\vartheta_1(i,j): & \{(s_l,(i,j)),(s_l',\varphi_l(i,j))\} \to \{(s_l',(i,j))\}, \quad l = 1,2,3,4; \\
\vartheta_2(i,j): & \{(0,1,0,1),(i,j)\} \to \{(1,0,1,0),(i,j)\}; \\
\vartheta_3(i,j): & \{(1,0,1,0),(i,j)\} \to \{(0,1,0,1),(i,j)\}.
\end{array} \tag{10}$$

Figure 1 shows three global states of the CA evolution which mimics the propagation of a round wave initiated by a dense square spot in the center of the array.

An asynchronous CA with the same A, M, Φ and $\Omega(0)$ evolves in quite a different way which is illustrated in Figure 2.

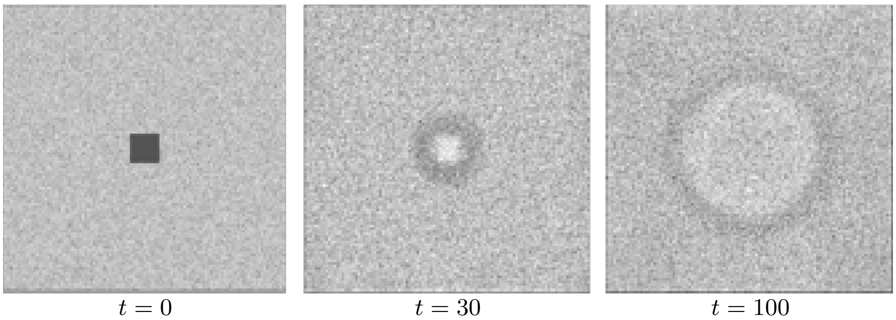

Figure 1. Three snapshots of synchronous CA evolution simulating a round wave, $|M| = 200 \times 200$

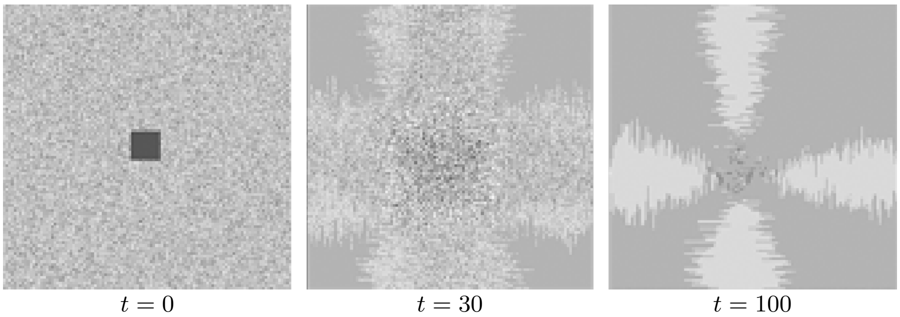

Figure 2. Three snapshots of ACA evolution of simulating a round wave, $|M| = 200 \times 200$

3. Block-synchronous CA

As is mentioned in the introduction, the problem of approximation of a kinetic ACA with a synchronous CA is very important, because its solution leads to efficient parallelization of the model. The concept of *approximation* has further the same meaning than that of approximation continuous space by a discrete mesh, i.e. a very large finite set Γ of possible evolutions is replaced by a smaller set γ, such that each evolution $\sigma_k \in \gamma$ is a representative of a subset $\Gamma_k \subseteq \Gamma$, all Γ_k, $k = 1, 2 \ldots, |\Gamma|/|\gamma|$, forming a partition on Γ. Such an approximation may be constructed by introducing a certain updating order, making the operation mode partially synchronized. To be formal, a CA approximating a given ACA=$\langle A, M, \theta \rangle$, should satisfy the following condition:

$$\gamma(\Omega) \subseteq \Gamma(\Omega) \qquad \forall\, \Omega \in A \times M. \tag{11}$$

There is another condition which is engendered by the danger of multicell synchronous updating incorrectness [11]. The fact is that if a local operator in a synchronous CA prescribes multicell updating, i.e., $|S(m)| > 1$, then some data may be lost. This occurs when a substitution, being applied simultaneously to different cells, makes an attempt to update the same cell. Sufficient condition to prevent it is as follows:

$$T'(m) \cap T'(m') = \emptyset \qquad \forall\, m, m' \in M \times M, \tag{12}$$

where $T'(m)$ is an underlying template for all $\vartheta(m) \in \theta(m)$. It is clear that classical synchronous automata having $|T'(m)| = 1$ are correct. So are ACA, because although they may have $|T'(m)| \geq 1$, their local operators can be applied only to one cell at the same time.

From the above it follows that the problem of ACA approximation is in constructing such a CA, that its mode of operation satisfy two conditions. On the one hand it should contain synchronous parts, on the other hand — correctness conditions (12) are to be met. Block-synchronous mode of operation seems to be appropriate.

Block-synchronous mode is an intermediate one between synchronous and asynchronous modes of operation: the transition $\Omega(t) \to \Omega(t+1)$ consists of q asynchronous steps, each one operating as a synchronous CA on a subset of cells. In more detail, the block-synchronous CA (BCA) mode of operation is as follows:

1. On the naming set M, a set of partitions $\Pi = \{\Pi_1, \dots, \Pi_k, \dots, \Pi_r\}$, each consisting of $G = |M|/r$ blocks, is defined as follows:
$$\Pi_k = \{B_k^1, \dots, B_k^G\}, \quad \bigcup_g B_k^g = \Omega, \quad B_k^g \cap B_k^h = \emptyset, \quad g \neq h. \tag{13}$$
2. Each block comprises r cell names defined by the block template
$$T_B(m_k) = \{m_k, \psi_1(m_k), \dots, \psi_r(m_k)\}. \tag{14}$$
the reference cell names $\{m_k^1, \dots, m_k^G\} = \Pi_k'$.
3. Each iteration time is divided into r steps. At the k-th step, $\theta(m_k)$ is applied synchronously to the cells of Π_k. The partitions Π_k, $k = 1, \dots, r$, being chosen in a random order.

Since the algorithm of a BCA functioning and its evolution depend on the size and the shape of the block-template T_B, the latter should be indicated in the BCA definition, which comprises four concepts:

$$\text{BCA} = \langle A, M, \theta, T_B \rangle.$$

Let an ACA=$\langle A, M, \theta \rangle$ be given with $T_\theta(m) = \{m, \varphi_1(m), \dots, \varphi_p(m)\}$ as an underlying template of θ. A $T_B(m)$ is called an *overlaying template* for $T_\theta(m)$ if
$$T_\theta(m) \subseteq T_B(m), \tag{15}$$
and if there exists a partition Π of M such that
$$T_B(m_i') \cap T_B(m_j') = \emptyset, \quad m_i' \neq m_j', \quad m_i', m_j' \in \Pi; \tag{16}$$
$$\bigcup_{m' \in \Pi} T_B(m') = M. \tag{17}$$

The case $T_B(m') = T_\theta(m')$ means that $T_\theta(m)$ has a shape, which may form a pavement on M. The simplest example is when $M = \{(i, j, k)\}$ and $T_\theta(m')$ has the shape of a parallelepiped in M.

The background for constructing a BCA approximating a given ACA is further formulated in the form of the following proposition.

Proposition. *A BCA* $= \langle A, M, \theta, T_B\rangle$ *is an approximation of an ACA* $= \langle A, M, \theta\rangle$ *if* $T_B(m)$ *is an overlaying template for* $\theta(m)$.

Proof. To prove that a BCA is an approximation of an ACA, it is sufficient to show that two conditions (11) and (12) hold, in the latter $T'(m)$ standing for $T_B(m)$. According to point 3 of the block-synchronous mode of application, each transition $\theta(\Omega(t)) \Rightarrow \Omega(t+1)$ of BCA may be expanded to a sequence

$$\sigma = \Omega(t),\ \Omega_1(t+\tau),\ \ldots,\ \Omega_r(t+r\tau), \tag{18}$$

where each $\Omega_k(t+k\tau)$ is a result of synchronous application of θ to $\Omega_{k-1}(t+(k-1)\tau)$, and $\Omega_r(t+r\tau) = \Omega(t+1)$, $\tau = t/r$.

According to (16), blocks do not intersect. So, condition (12) is satisfied for each synchronous step, i.e., the updating in blocks is completely independent. Hence, the result depends neither on order of blocks updating, nor of the mode of operation inside a step. Since it is true for all $\Omega_k(t)$, $k = 1, \ldots, r$, and $\bigcup_k \Omega_k = \Omega$, the transition $\Omega_1(t) \rightarrow \Omega_1(t+1)$ is a result of independent updating of all cells of $\Omega_1(t)$, which coincides with one out of $|M|!$ possible global operator applications to $\Omega_1(t)$. The above holds for any iteration and any initial $\Omega \in A \times M$. So, both conditions are proved to satisfy approximation conditions. □

The proposition ensures that the block-synchronous mode is a restriction of the asynchronous one. How serious is the discrepancy from the true process under simulation – it is the question which may be answered with a comprehensive experimental study. Our yet rather a poor experience exhibited no estimable difference between the two modes of CA operation, which is shown by the following example.

Example 2. A simple model of epitaxial growth on Silicon (Si) surface is a composition of the two following actions:

1. Absorption of Si-atoms from gas with probability p_a. Each act of the absorption adds an atom to the surface increasing the thickness of the absorbed layer.
2. Diffusion of absorbed atoms over the surface. An atom diffuses to a neighboring cell if the state of the latter is less that of its own. If there is n such neighbors ($n = 1, 2, 3, 4$), then probability of the diffusion act is $p_d = 0.05^{4-n}$, and the choice among n possible directions to move to, is equiprobable, $p_d = 1/n$.

The process may be described by an ACA $= \langle A, M, \theta\rangle$, where $A = \boldsymbol{N}$, $M = \{(i,j) :\ i = 0, \ldots, I,\ j = 0, \ldots, J\}$. A cell $(a, (i,j))$ corresponds to a site on Si surface, where the thickness of the adsorbed layer is equal to a

atoms. The transition rule $\theta(i,j)$ is a superposition of two elementary local operators: ϑ_1 being responsible for absorbtion, ϑ_2—for diffusion.

$$\theta(i,j) = \vartheta_2(\vartheta_1(i,j))),$$

where

$$\vartheta_1 = \{(v_0,(i,j))\} \xrightarrow{p_a} \{(v_0+1,(i,j))\}, \tag{19}$$
$$\vartheta_2 = \{(v_0,(i,j)),(v_1,\varphi_1(i,j)),(v_2,\varphi_2(i,j)),(v_3,\varphi_3(i,j)),(v_4,\varphi_4(i,j))\} \xrightarrow{p_d}$$
$$\{(u_0,(i,j)),(u_1,\varphi_1(i,j)),(u_2,\varphi_2(i,j)),(u_3,\varphi_3(i,j)),(u_4,\varphi_4(i,j))\},$$

where

$$u_0 = \begin{cases} v_0 & \text{if } (\forall l : v_l \geq v_0) \vee (\xi > p_n), \\ v_k - 1 & \text{if } (v_k < v_0) \wedge (\xi \leq p_n), \end{cases}$$
$$u_k = \begin{cases} v_k & \text{if } (\forall l : v_l \geq v_0) \vee (\xi > p_n), \\ v_k + 1 & \text{if } (v_k < v_0) \wedge (\xi \leq p_n), \end{cases}$$

ξ being a random number in the interval $[0,1]$. In the asynchronous operation mode, the global transition $\Omega(t) \to \Omega(t+1)$ is a result of application of (19) to all cells in $\Omega(t)$ each being randomly chosen.

In the block-synchronous mode, each next global state is a result of nine synchronous steps. At the l-th step, local operator (19) is applied to the l-th reference cells of all blocks belonging to the l-th partition, blocks being 3×3 squares. The simulation process shows the formation of islands on the

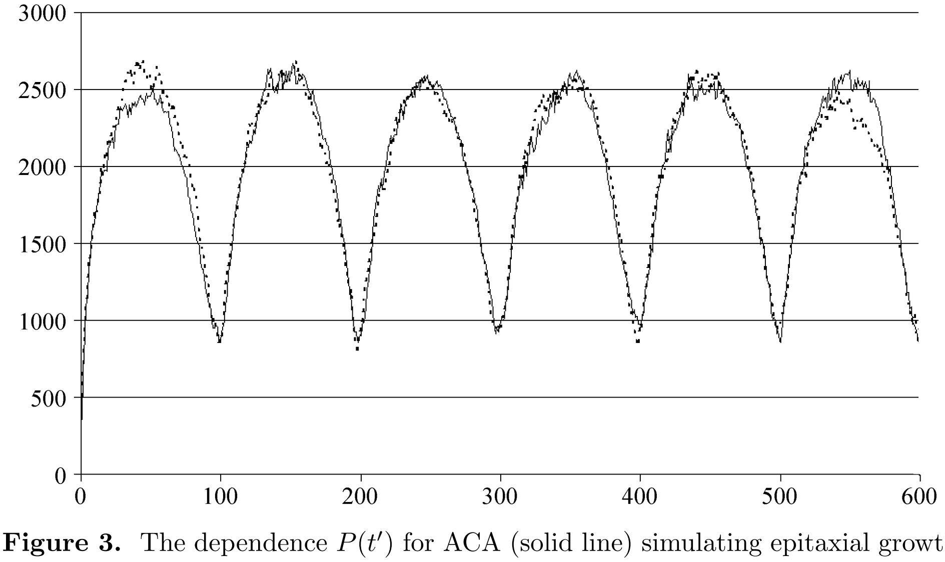

Figure 3. The dependence $P(t')$ for ACA (solid line) simulating epitaxial growth and its approximation by a BCA (dotted line), $|M| = 100\times 100$, $p_a = 0.2$, $t' = t/500$

surface. One of the features under investigation is the dependence $P(t)$ of the total perimeter of the islands on time. The time is measured in iteration numbers, P is measured in the number of cell pairs having different states. During the process, $P(t)$ exhibits oscillations. In Figure 3, the first waves $P_{\rm as}(t)$ and $P_{\rm bs}(t)$ of such oscillations are shown for asynchronous and block-synchronous modes of operation with nine array partitions into 3×3 blocks. It is seen that the perimeter curves almost coincide. A mean relative error of approximation is $E = 0.0412$.

4. Conclusion

The method of CA with the block-synchronous mode of operation, whose evolution approximates that of a kinetic ACA, is presented. The approximation error may be considered to be sufficiently good. Preliminary experimental parallel implementation of the BCA shows that the efficiency of 80 % can be achieved. But the investigation of its dependence on the features of the ACA transition rules, as well as of the relation between the block size and the array size are to be investigated in the future.

References

[1] Toffilli T., Margolus N. Cellular Automata Machines.—USA: MIT Press, 1987.

[2] Wolfram S. A New Kind of Science.— Champain (Ill., USA): Wolfram Media Inc., 2002.

[3] Rothman D.H., Zaleski S. Lattice-Gas Automata. Simple Models of Complex Hydrodynamics.— Cambridge: Cambridge University Press, 1997.

[4] Weimar J.R. Cellular Automata for reaction-diffusion systems // Parallel Computing.— 1997.— Vol. 23, No. 11.— P. 1699–1715.

[5] Popov V.L., Psakhie S.G., Theoretical basis for for modeling elastoplastic media by movable cellular automata. 1. Homogeneous media // Physicheskaya mesomechanika.— Vol. 4, No. 1.— P. 17–28 (In Russian).

[6] Bandini S., Mauri G., Pavezi G., Simone C. A parallel model based on Cellular Automata for the simulation of pesticide percolation in the soil // Lect. Notes Comp. Sci. / Ed. V. Malyshkin.— Berlin: Springer, 1999.— Vol. 16662.— P. 382–394.

[7] Neizvestny I.G., Shwartz N.L., Yanovitskaya Z.Sh., Zverev A.V. 3D-model of epitaxial growth on porous {111} and {100} Si surfaces // Computer Physics Communications.— 2002.— Vol. 147.— P. 272–275.

[8] Ziff R.M., Gulari E., Bershad Y. Kinetic phase transitions in irreversible surface-reaction model // Phys. Rev. Lett.— 1996.— Vol. 56.— P. 2553–2558.

[9] Elokhin V.I., Latkin E.I., Matveev A.V., Gorodetskii V.V. Application of statistical lattice models to the analysis of oscillatory and autowave processes on the reaction of carbon monoxide oxidation over platinum and palladium surfaces // Kinetics and Catalysis. — 2003. — Vol. 44, No. 5. — P. 672–700.

[10] Makeev A.G. Coarse bifurcation analysis of kinetic Monte Carlo simulations: a lattice-gas model with lateral interactions // J. Chemical Physics. — 2002. — Vol. 117, No. 18. — P. 8229–8240.

[11] Achasova S.M., Bandman O.L., Markova V.P., Piskunov S.V. Parallel Substitution Algorithm. Theory and Application. — Singapore: World Scientific, 1994.

[12] Simons N.R.S., Bridges G., Cuhachi M.A. Lattice-gas automaton capable of modeling three-dimensional electromagnetic fields // J. Comput. Physics. — 1999. — Vol. 151. — P. 816–835.