*Bull. Nov. Comp. Center*, Comp. Science, 17 (2002), 17–29


# Discrete-continuous models for spatial dynamics simulation*

O. Bandman

Fine-grained parallel models of spatial dynamics are analyzed from the point of view of the relationship between their continuous and discrete constituents. The models are ranged from the absolutely continuous partial differential equations to the absolutely discrete cellular automata. In the interval between them, there are cellular neural networks, cellular neural automata, and probabilistic cellular automata. A generalized representation of the above models is proposed, which is assumed to be a background for a unified technology for the fine-grained parallel programming. The models are analyzed and compared on the basis of the results, obtained at the Supercomputer Software Department.

## 1. Introduction

There exist a number of mathematical models of reaction-diffusion phenomena. Formally, they represent spatial dynamics of an abstraction of a certain physical value (density, pressure, velocity, temperature), further referred to as concentration. It is customary to describe spatial dynamics in terms of the PDE systems. This model is fully continuous, i.e., all its variables are in the domain of real numbers. Unfortunately, the PDEs expressed in the differential form, are of little practical use, because they are adequate only for obtaining analytical solution. The latter is possible in a limited number of very simple cases. Hence, in real life the computer is involved to obtain the solution and to this end numerical methods are being developed. In such methods the PDE is approximated by a finite difference representation, where time and space are discrete, and the concentration is left continuous. There are a lot of numerical methods for the PDE solution. Among them the explicit ones may be considered as fine-grained parallel, because computation at all the points of a discrete space may be executed in parallel, and hence, they admit any spatial partition to allocate the task on a multiprocessor computer. However, explicit methods have the following disadvantages: in order to provide acceptable computational stability the time step (and/or the spatial step) should be taken small enough, which results in considerable computational complexity.

*Supported by the Russian Foundation for Basic Research under Grant 01.00.00026.

In this connection, the search for the new mathematical models, to be more adequate to modern parallel computers, seems to be fairly natural [1]. Most of attempts are based on the Cellular Automata (CA) approach. A classical CA is an absolutely discrete model of spatial dynamics. It is represented by an array of identical cells operating in parallel and computing the same Boolean function of the states of the cell neighbors. It is well known, that in spite of each cell simplicity, the CAs are capable of simulating very complex phenomena. Thus, there exists a scope of the CA models (gas dynamics, diffusion, acoustic waves, phase transitions, etc.), which are expected to have considerable promise. The CA model have some advantages over the finite difference PDE: they are free from round-off errors and absolutely stable. At the same time there are yet many problems to be solved before the CAs become widely used, among them the most significant are the following: providing the adequacy of the CA operation rules to given physical parameters of a phenomenon to be simulated, and eliminating the "automaton noise".

In-between the above two extremes (the PDE and the CA) there are a number of intermediate models, in which both discrete and continuous functions are used, the arguments (time and space) being left discrete in all the cases. Among them the following seem to have considerable prospects: Lattice–Boltzmann [2], gas-lattice with weight connections [3], cellular-neural associative memory [4], cellular-neural networks [5], cellular-neural automaton [6, 7]. Each of the above models is destined to describe a certain class of reaction-diffusion processes. Some of them have the PDE equivalents, not yet being their approximation. These are the Lattice–Boltzmann and Gas–Lattice models whose equivalent is the Navier–Stokes PDE. Some others have no equivalent PDE, these are cellular-neural associative memory and cellular-neural networks, whose parameters are obtained through training them by prototypes. Finally, a cellular-neural automaton is considered to be a hybrid one [6], which is a combination of the CA-diffusion with a reaction approximation by Boolean arrays.

Here an attempt is made to construct a generalization of these models and to show that they form a class of fine-grained parallel representations of reaction-diffusion phenomena. The goal of such a generalization is twofold: to simplify the comparative analysis of the existing models and to make the development of the new ones more justified. For this purpose a number of models of the above series which are under investigation at the Supercomputer Software Department are analyzed, and their properties are compared. In Section 2, formal definitions and a generalized form of the cellular system representation are given. Section 3 is concerned with two basic fine-grained models whose properties are inherited by those, proposed especially for the reaction-diffusion simulation and studied at our department. The results of this study are presented in Section 4.

## 2. Cellular form of spatial dynamics representation

A conventional representation of the reaction-diffusion process is a PDE system, each equation being represented as follows:

$$\frac{du^{(k)}}{dt'} = d^{(k)}\Delta u^{(k)} + F^{(k)}(u^{(1)}, \ldots, u^{(l)}), \tag{1}$$

where $u^{(k)}$, $k = 1, \ldots, l$, is concentration of the $k$-th substance involved in the reaction-diffusion process under simulation, $F^{(k)}(\cdot)$ is a reaction function, $d^{(k)}$ is a diffusion coefficient in the $k$-th equation, and $\Delta u$ is a Laplacian,

$$\Delta u = \frac{\partial^2 u}{\partial x_1^2} + \frac{\partial^2 u}{\partial x_2^2},$$

where $(x_1, x_2)$ are coordinates of a 2D continuous space.

After the time and the space discretization of (1) with $x_1 = ih$, $x_2 = jh$ and $t' = t\tau$, $i$, $j$, $t$ being integers, $h$, $\tau$ are space and time steps, (1) takes the following form:

$$\begin{aligned} u_{i,j}^{(k)}(t+1) \;=\; & u_{i,j}^{(k)}(t) + \frac{\tau d^{(k)}}{h^2}(u_{i-1,j}^{(k)}(t) + u_{i,j-1}^{(k)}(t) + u_{i+1,j}^{(k)}(t) + \\ & u_{i,j+1}^{(k)}(t) - 4u_{i,j}^{(k)}) + \tau F^{(k)}(u^{(1)}, \ldots, u^{(l)})). \end{aligned} \tag{2}$$

In order to construct a fine-grained generalized representation, suitable for the reaction-diffusion processes simulation, a concept of *algorithm architecture* is further used. This concept is appropriate to the fine-grained parallelism, it determines spatial relations of the algorithm operators. Time is given by natural numbers $t = 0, 1, \ldots$, space – by an infinite set of the cell names $M = \{1, 2, \ldots\}$. Such a natural cell state number indexing of the cell names is used due to its universality, since for any regular spatial cell structure there exists a one-to-one mapping $\xi : \{(i, j)\} \to M$.

The data to be processed are represented by a *cellular array* $\Omega$ given as a set of *cells*,

$$\Omega = \{(u, m) : u \in \Upsilon, m \in M\},$$

where $u$ is a cell state variable with values from $\Upsilon$ considered as *cell state alphabet*. The structure of cell interactions in a cellular array is defined by a *template* $T(m)$, which is represented by a set of naming functions

$$T(m) = \{\phi_0(m), \phi_1(m), \ldots, \phi_n(m)\}, \tag{3}$$

where $\phi_g(m)$, $g = 0, 1, \ldots, n, \phi_0(m) = m$, is a *naming function* $\phi : M \to M$, whose value is the name of the $g$-th neighbor in the *neighborhood* of the

cell named $m$. In the case of the 2D Cartesian cellular space, the naming functions are usually of the form of shifts: $\phi_g(\xi(i,j)) = \xi(i+a, j+b)$, $a$, $b$ being integers.

An extended concept of a template referred to as *weight template* $W$ is used. In it, a *weight function* $f_g(u)$ is associated with each naming function $\phi_g(m)$, so that each element of $W(u,m)$ is a pair of two functions,

$$W(u,m) = \{(f_g(u), \phi_g(m)) : g = 0, 1, \ldots, n\}. \tag{4}$$

In terms of the cellular algorithm architecture, $T(m)$ is considered to determine each cell connections, and $W(u,m)$ – the weights of the connections.

A subset of cells

$$N_W(m) = \{(u_0, m), \ldots, (u_n, \phi_n(m))\} \tag{5}$$

in a cellular array $\Omega$, $N_W(m) \subseteq \Omega$ forms the *neighborhood* of a cell $(u_0, m)$ with $W(u,m)$ and $N(m)$ having one and the same underlying template $T(m)$. This enables us to define the following operation:

$$W(u,m) \odot N_W(m) = \{f_0(u_0), f_1(u_1), \ldots, f_n(u_n)\} \tag{6}$$

which serves as argument to compute a diffusion part of the cellular algorithm

$$u_D(m) = \Phi(W \odot N_W(m)) = \Phi(f_0(u_0), f_1(u_1), \ldots, f_n(u_n)), \tag{7}$$

where $\Phi(\cdot)$ is a *diffusion operator* reflecting the diffusion process peculiarities.

Introducing the projections of $W(u,m)$ and $N(m)$ into the sets $\{f_g(u)\}$, and $\{u_g\}$, $g = 0, \ldots, n$, the vectors $\boldsymbol{W}(u,m) = (f_1(u), \ldots, f_n(u))$ and $\boldsymbol{U}(m) = (u_0, \ldots, u_n)$ may be used. It is especially helpful when weight functions are of the form $f_g(u) = w_g u$, and $\Phi(\cdot)$ is a summation. Then the diffusion operator reduces to the scalar product

$$u_D(m) = \langle \boldsymbol{W}(u,m), \boldsymbol{U}(m) \rangle. \tag{8}$$

When a diffusion part is in the form of spatially distributed Boolean values, then transformations from the Boolean spatial form to a continuous one and vice versa are needed. To perform the first transformation an *averaging template* $Av(m)$ and the corresponding averaging neighborhood $N_{Av}(m)$ are introduced:

$$Av(m) = \{(a_0, m), (a_1, \psi_1(m)), \ldots, (a_q, \psi_q(m))\}, \tag{9}$$

$$N_{Av}(m) = \{(u_0, m), (u_1, \psi_1(m)), \ldots, (u_q, \psi_q(m))\}, \tag{10}$$

$a_j \in \{0, 1\}$. The averaging procedure transforms a Boolean spatial function into a continuous one as follows:

$$u'(m) = \Psi(Av(m) \odot N_{Av}(m)) \tag{11}$$

With $\Psi(\cdot)$ being the mean value (11), yields

$$u'(m) = \frac{1}{q} \sum_{j=1}^{q} a_j u_j. \tag{12}$$

When diffusion is represented in a continuous form, $Av(m)$ degenerates into $Av(m) = m$, $u' = u_0$.

With the above definitions the following state equation in the cellular form is as follows

$$u(m, t+1) = \Phi(W(u, m) \odot N_W(m)) \oplus F(\Psi(Av(m) \odot N_{Av}(m))), \tag{13}$$

where $\oplus$ is a *diffusion-reaction operator* which denotes some kind of a combination of diffusion and reaction values, most frequently used in the arithmetic summation.

When a multicomponent process is under simulation, the cellular array is considered to be a layered structure, each layer corresponding to one of the components. Let $k = 1, \ldots, l$ be a layer index. Then

$$M = \bigcup_k M^{(k)}, \quad \Omega = \bigcup_k \Omega^{(k)}, \quad \Omega^{(k)} = \{(u, m^{(k)}\}, \quad u \in \Upsilon, \quad m^{(k)} \in M^{(k)},$$

$|M^{(k)}| = |M^{(j)}|$ for all $k, j = 1, \ldots, l$. Numeration of names in all the layers is identical, so that $m_i^{(k)} = m_i^{(j)}$ for all $m_i \in M$, $k, j \in \{1, \ldots, l\}$.

Each layer computes its own diffusion part $C_D^{(k)}$ which depends on the corresponding concentration variables, $W^{(k)}(u, m)$, $Av^{(k)}(m)$ being different for different $k \in \{1, \ldots, l\}$. The reaction part $C_R(m)$ is a function of continuous or averaged values in the corresponding cells of all layers. To represent it in a cellular form a *reaction template* is introduced

$$R_L = \{(a_j u'_D, j) : \ a_j = 0, 1, \ \ j = 1, \ldots, l\}, \tag{14}$$

which is identical for all the layers. The corresponding reaction neighborhood is $N_R(m) = \{(u'_1, m^{(1)}), \ldots, (u'_l, m^{(l)})\}$ $u'$ being the averaged results of the diffusion operator execution.

Accordingly, the reaction part is

$$y(m) = F^{(k)}(R_L \odot N_R(m)). \tag{15}$$

Using the above notations a generalized form of multicomponent reaction-diffusion may be written down as a set of the following equations

$$u^{(k)}(m, t+1) = \Phi(W^{(k)}(m) \odot N_W^{(k)}(m)) \oplus F^{(k)}(R_L \odot N_R(m)), \qquad (16)$$

for $k = \{1, \ldots, l\}$.

The above equations together with the parameters $\Upsilon$ and $M$ form a *cellular system*. Being set in the initial state $\Omega(0)$, a cellular system starts functioning according to (16). A cellular system together with the parallel by cells, synchronous and iterative mode of computation constitutes *a cellular algorithm*.

A sequence of the cellular arrays $\{\Omega(0), \ldots, \Omega(t_s)\}$, obtained in the course of cellular algorithm execution, is referred to as *cellular array evolution*.

The evolution terminates when a *stable state* $\Omega_s$, such that

$$\Omega_s(t+1) = \Omega_s(t), \qquad (17)$$

is reached.

If any initial cellular array generates a single terminating evolution, then the algorithm is considered to be *deterministic*.

It is clear that the finite difference representation (2) is a particular case of (13). With $l = 1$ its cellular form is as follows

$$C_D(\xi(i,j)) = \tau d/h^2(u_{i-1,j} + u_{i,j-1} + u_{i,j+1} + u_{i+1,j} + (-4 + h^2/\tau d)u_{i,j},$$
$$C_R(\xi(i,j) = F(u_{i,j}), \quad u(\xi(i,j), t+1) = C_D(\xi(i,j) + C_R(\xi(i,j).$$

## 3. Basic fine-grained models

Two models of fine-grained parallel computations are considered to be fundamental. They are *cellular automaton* and *artificial neural network*. All others are either their modifications, or hybrids, possessing the features of both.

**3.1. The Cellular automata** (CA), introduced by von Neumann half a century ago, nowadays are intensively studied as models belonging to the so called *complex systems* [8], which are huge sets of simple interacting processing elements, whose parallel execution is capable of computing complex functions. A CA is represented by a discrete cellular space, each cell being associated with a finite automaton with inputs from its neighborhood according to a given template.

In terms of the generalized notations, a CA may be regarded as a cellular system with the following parameters:

- CA is a one-layer cellular system, $l = 1$.
- A cellular space has a spatial metrics, i.e., the naming set is a regular structure. Usually, it is a 1D or a 2D infinite lattice, either Cartesian or hexagonal.

- The neighborhood is formed of the "spatial neighbors", there existing a finite radius $r$ in a given metrics such that $N(m)$ is enclosed in the sphere of radius $r$ sphere with the center in $m$.
- The alphabet of states is Boolean, $\Upsilon = \{0, 1\}$.
- The weight functions $f_g(u) = a_g u$ with $a_g \in \{0, 1\}$, $g = 1, \ldots, n$. Hence $W(u, m) \odot N(m) = \{a_g u_g : \ g = 1, \ldots, n\}$.
- The diffusion operator $\Phi(\cdot)$ is a Boolean function, representing the transition rules of a finite elementary automaton.

The above "classical" model of a CA has many modifications. Two of them are worth to be considered.

1. The cell states $\Upsilon = \{\boldsymbol{u}_i = (u_1, \ldots, u_n) : \ u_g = 0, 1\}$ are Boolean vectors, and, therefore $\Phi(\cdot)$ is a system of Boolean functions. Such CAs are used in Gas-Lattice models, the most known ones being called the FHP [9]. Each component $u_g$ in the state-vector represents either the existence of $(u_g = 1)$ or the absence of $(u_g = 0)$ of a particle moving at unit speed towards the $g$-th neighbor. A modification of the FHP-model extending its domain of applicability, and a 3D variant are under investigation [10, 11].

2. The right-hand side of (15) represents the next states in a block of cells $B(m)$, which at the same time is a neighborhood $N(m) = B(m)$. The next state is computed as a mapping of $\{B(m)\} \to \{B(m)\}$. Such a modification is a subject of study in Parallel Substitution Algorithm theory [12]. A well-known particular case of the CA of such a type is called "CA with Margolus neighborhood" [13]. Apart from other numerous applications the diffusion simulation is one of the most important. In a 2D case, the Margolus neighborhood is a block of four adjacent cells

$$B(i, j) = \{(u_0, \xi(i, j)), (u_1, \xi(i + 1, j)), (u_2, \xi(i + 1, j + 1)), (u_3, \xi(i, j + 1)\}.$$

With $f_g(u) = u$, $g = 1, \ldots, l$, and $U = (u_0, u_1, u_2, u_3)$ a diffusion operator

$$\Phi(W(u, m) \odot N(m)) = \begin{cases} (u_1, u_2, u_3, u_0) & \text{if } p < 1/2, \\ (u_3, u_0, u_1, u_2) & \text{if } p \geq 1/2 \end{cases} \tag{18}$$

executes the rotation of states in the block clockwise or counterclockwise with probability $p = 1/2$.

Each iteration of the algorithm contains two steps: at the first step, formula (17) is applied to the cells whose coordinates sum $(i+j)$ is even, at the second step the same is done for the cells with odd $(i+j)$. The experimental study of the above algorithm has shown its good correspondence to the PDE solutions [14].

**3.2. The Artificial Neural Network** (ANN) is a concept of a class of models, usually referred to as *paradigms*. All of them manifest a fine-grained

parallelism, but do not possess the interaction locality property. Moreover, two ANN properties – weight connections and learning capability – have been inherited by a powerful reaction-diffusion model called *Cellular Neural Network* (CNN). The CNN is derived from the ANN paradigm called Hopfield Neural Associated Memory, which can be represented in terms of a cellular system as follows:

- Each node of the network is a processing element called a *neuron*.
- Array structure is given by a strongly connected graph. Hence, the cell neighborhood is formed of all neurons in the network. Weight templates are different for different cells, hence, weight functions depend not only on $u$ but also on $m$, i.e.,

$$W(u,m) = \{(w_0(m)u_0, m), \ldots, (w_n(m)u_n, \phi_n(m))\}$$

- The cell state alphabet is a "semi-continuous" one. This means that its domain is an interval $(0,1)$ or $(-1,1)$, which is the value of a "sigmoid" of a diffusion operator. Continuous, piecewise linear and fully discrete forms of a sigmoid function may be met in numerous studies of the Hopfield paradigm, the most widespread is a threshold one, which turns the alphabet of a state to the Boolean one.
- Weight templates entries are $f_g(u,m) = w_g(m)u_g$, $g = 0, \ldots, n$. A diffusion part is thus computed as follows

$$u_D = \Phi(W \odot N(m)) = \Theta\Big(\sum_{g=0}^{n} w_g(m)u_g\Big), \tag{19}$$

  where $\Theta(\cdot)$ is a sigmoid function.
- A reaction function $F(u) = 0$, although sometimes constants are added to provide the correctness of learning results [15].

A difficulty to realize the full-connection Hopfield associated memory has brought up the question, whether its storage capability degrades if the neighborhood in it is reduced to a local one [16]. The new system was called the *Cellular Neural Associative Memory* (CNAM). The CNAM is classified in [17] as "nonhomogeneous" cellular neuron network, because like in a Hopfield associative memory, the weight functions are different from cell to cell. Moreover, the CNAM is not intended to simulate reaction-diffusion processes. Nevertheless, its investigation has been very useful for this purpose, because it has clarified the stability conditions, and determined the learning principles for all types of Cellular Neural Networks (CNN) [17].

# 4. Hybrid fine-grained models

Hybrid models use both discrete and continuous computations [6], and possess properties of both basic models. Two hybrid models are studied at our department and presented below: Cellular Neural Automaton (CNA) and Cellular Neural Networks (CNN). The CNA is an integration of the CA diffusion with continuous reaction, the latter being approximated by a Boolean array with the use of a probabilistic neuron. So, a large share of computation in the CNA is in Boolean domain. The CNNs possess all the properties of the ANN, except the array structure inherited from the CA. Thus, its alphabet is semidiscrete, and most of the computation is in the domain of real numbers.

**4.1. The Cellular Neural Automaton** is an original model. It has been proposed in [7]. The CNA combines in a single iterative procedure the Boolean operations of the CA diffusion with Boolean approximation of the reaction part. Motivation to such an approach relies upon the existence of simple stable and effective CA models of diffusion. The CNA is intended to be used as an alternative to a reaction-diffusion PDE solution. It may be represented as cellular system of the form (15) with the following parameters:

- Array structure is a multilayer regular Cartesian lattice provided with a metrics, the number of layers being equal to the CNA order.
- Intercell communications are determined by weight templates, averaging templates, and a reaction template of the form (4), (9), and (14), respectively.
- Cell state alphabet is Boolean, though intermediate computations of a reaction operator are done in real numbers.
- Any diffusion CA algorithm may be chosen, particularly, the one given in Subsection 2.1. So, the diffusion operator is computed according to (17), resulting in Boolean cell states $u_D(m)$.
- Reaction operator is performed according to (12) for averaging the diffusion results and (15) – for the computing reaction part.

The next-state is computed as a probabilistic neuron value, which updates the diffusion resulting array by appending to it the reaction part as follows:

$$u(m,t+1) = \begin{cases} 1 & \text{if } u(m,t)=0,\ y(m,t)>0,\ T^+(y(m,t)) > \text{rand}(1), \\ 0 & \text{if } u(m,t)=1,\ y^(m,t))<0,\ T^-(y(m,t)) > \text{rand}(1), \\ u(m,t) & \text{otherwise}, \end{cases} \tag{20}$$

where rand(1) is a random number from the interval (0,1), $T^+(y)$ and $T^-(y)$ are probabilistic neuron functions, equal to probabilities of the event according to which the cell should change its state $u(m) = 0$ for $u(m) = 1$, and $u(m) = 1$ for $u(m) = 0$, respectively,

$$T^+(y) = \frac{y}{1-u'}, \quad T^-(y) = \frac{|y|}{u'}. \tag{21}$$

Computer simulation of a number of typical reaction-diffusion processes has shown the possibility of using the proposed model for studying phenomena in active media [18].

**4.2. The Cellular Neural Network.** The CNNs theory and possible applications have been proposed by L. Chua and his team [5]. The subject of their study is the so-called *standard CNN*. In a discrete form a standard CNN is represented by a cell functioning as follows:

$$v(t+1) = -v(t) + \langle \boldsymbol{W}, \boldsymbol{U}(t) \rangle + z, \tag{22}$$

$$u(t+1) = \frac{|v(t+1)+1| + |v(t+1)-1|}{2}, \tag{23}$$

where $v \in \boldsymbol{R}$ is an internal state of a neuron, $\boldsymbol{W}$ is a vector of connection weights equal for all neurons, $\boldsymbol{U}$ is a state neighborhood.

The neuron state $u \in (-1, 1)$ is a piecewise sigmoid function like (23) in its internal state. Its range has three parts: $u = -1$, $u = 1$ are saturated parts, and $u = v$ is a linear part. If the process under simulation is given by a single pair of the variables $(v, u)$, then a CNN is considered to be of first order, otherwise it is of second order (the third order of CNNs also exists, but has not been studied yet at our department).

Investigation results of a standard CNN show that the model is capable of simulating a wide range of reaction-diffusion phenomena, whose PDEs are sometimes unknown. Instead, several global states of the system, are available, serving as *prototypes* for the learning procedure [19].

The CNN is a multicomponent cellular system with the following parameters:

- Array structure is a multilayer regular Cartesian lattice provided with a metrics, the number of layers being equal to the CNN order.
- Intercell communications are represented by weight templates and a reaction template of the form (4) and (14), respectively. Weight functions are polynomials, most frequently of first order.
- Cell state alphabet is a "semi-continuous" one. This means that its domain is within the interval $(-1, 1)$.

- Diffusion operator

$$\Phi(U) = \Theta\Big(\sum_{g=0}^{n} f_g(u)\Big),$$

  where $\Theta(\cdot)$ is a piecewise sigmoid of the form (23).
- Reaction functions $F^{(k)}(\cdot)$ are linear (though polynomials of higher degrees are also possible).

The first order CNNs are used to simulate reaction-diffusion processes, whose evolution results in a stable state, representing a spatial pattern. If the template is symmetrical, i.e.,

$$\phi_g - \phi_h = 0 \rightarrow w_g = w_h,$$

then the system is stable. Whatever the initial global state is given, a CNN evolves to one of the stable states. Determining template parameters of a CNN by a number of given stable states is done by learning. A learning method has been developed [18], which provides a tool for studying the pattern formation processes, and enables us to investigate a relationship between CNN parameters and properties of patterns, which can be generated. The results of this study may be used to clarify the phenomenon of pattern formation in crystallization, biological evolution, ecological processes, etc.

The second order CNNs are represented by two-layer arrays with $W^{(k)}(u, m)$ in the form of (3) [19],

$$\begin{aligned} &R_L = \{(1, m^{(1)}), (1, m^{(2)})\}, \quad R_L \odot N_R = \{u^{(1)}(m), u^{(2)}(m)\}, \\ &F^{(1)} = \{(a_1 u^{(1)} + b_1, u^{(2)})\}, \quad R^{(2)} = \{(a_2, u^{(1)}) + (b_2, u^{(2)})\} \end{aligned} \tag{24}$$

with $b_2 = -b_1$.

Their evolution simulates emergency and propagation of different types of *autowaves* (propagating front, propagating pulse, oscillations). Functioning of a neuron pair $(m^{(1)}, m^{(2)})$ is studied using a method of qualitative theory of nonlinear systems, resulting in the relations between the CNN parameters (weight and reaction templates), which provide generation of a particular type of autowaves [21, 22]. Computer program package has been developed, which is used for experimental study of autowaves behavior under different external conditions.

## 5. Conclusion

It is shown that a number of fine-grained reaction-diffusion models, which combine discrete and continuous modes of computation, form a class of cellular systems. A generalized representation of such systems is proposed,

which is useful for the creation of a unified technology for parallel programming. Apart from the inherent parallelism, the proposed models have some advantages derived from discreteness of the diffusion part, which provides the absence of the round-off errors and the improvement of the computation stability.

## References

[1] Bandman O. Fine-grained parallelism in computational mathematics // Programmirovanie. – 2001. – № 4. – P. 5–20 (in Russian),

[2] Rothman D.H., Zaleski S. Lattice-Gas Cellular Automata. Simple models of complex hydrodynamics. – Cambridge: University Press, 1997.

[3] Chen S., Wang Z., Shan X., Doolen G.D. Lattice-Boltzmann computational fluid dynamics in three dimensions // J. Statistical Physics. – 1992. – Vol. 68, № 3/4. – P. 379–407.

[4] Bandman O.L. Cellular-neural models of time-spatial dynamics // Programmirovanie. – 1999. – № 1. – P. 1–14 (in Russian).

[5] Chua L. CNN: A Paradigm for Complexity. – Singapore: World Scientific, 1999.

[6] Bandman O.L. A hybrid approach to reaction-diffusion processes simulation // Lecture Notes in Computer Science. – Berlin: Springer, 2001. – Issue 2127. – P. 1–16.

[7] Bandman O.L. Cellular neural-automaton model of reaction-diffusion processes // Neuroinformatics and its Application. Proceedings of IX National Seminar, Krasnoyarsk, Oct. 5–7, 2001. – Krasnoyarsk: KGTU, 2001. – P. 10–11 (in Russian).

[8] Wolfram S. Cellular Automata Fluids 1: Basic Theory // J. Statistical Physics. – 1986. – Vol. 45, № 3/4. – P. 471–526.

[9] Frish U., d'Humiere D., Hasslacher R., Lallemand P., Pomeau Y., Rivet J. Lattice-gas hydrodynamics in two and three dimensions // Complex Systems. – 1987. – Vol. 1. – P. 649–707.

[10] Medvedev Yu. Modeling flows of fluids and gases with cellular automata // Modeling of nonequilibrium systems–2001. Proceedings of IV National Seminar (Krasnoyarsk, Oct. 12–14, 2001). – Krasnoyarsk: KGTU, 2001. – P. 98 (in Russian).

[11] Medvedev Yu. A modified cellular-automata model of fluid flow // New Information Technologies in Discrete Structures Study. Proceedings of III National Conference (Tomsk, Sept. 12–14, 2000). – P. 84–89 (in Russian).

[12] Achasova S., Bandman O., Markova V., Piskunov S. Parallel Substitution Algorithm. Theory and Application. – Singapore: World Scientific, 1994.

[13] Malinetsky G.G., Stepantsov M.E. Modelling diffusive processes by cellular automata with Margolus neighborhood // Zhurnal Vychislitelnoy Matematiki i Matematicheskoy Phisiki. – 1998. – Vol. 36, № 6. – P. 1017–1021 (in Russian).

[14] Bandman O. Comparative study of cellular automata diffusion models // Lecture Notes in Computer Science. – Berlin: Springer, 1999. – Issue 1662. – P. 395–409.

[15] Pudov S. Comparative analysis of learning methods of cellular-neural associative memory // Lecture Notes in Computer Science. – Berlin: Springer, 1999. – Issue 1662. – P. 108–1197.

[16] Pudov S. Limiting capability of cellular-neural associative memory // NCC Bulletin. Series Computer Science. – Novosibirsk: NCC Publisher, 1998. – Issue 9. – P. 93–98.

[17] Pudov S. Learning of cellular neural networks // Future Generation Computer Systems. – 2001. – Vol. 17. – P. 689–697.

[18] Bandman O. Cellular-Neural Automaton. A Discrete Model of Active Media Dynamics. – http://www.sbras.ru/ws/Lyap2001/23-34.

[19] Pudov S. First-order 2D cellular neural networks. Investigation and learning // Lecture Notes in Computer Science. – Berlin: Springer, 1999. – Issue 2127. – P. 94–97.

[20] Selikhov A.V. Some results of autowave modelling by cellular neural network // NCC Bulletin. Series Computer Science. – Novosibirsk: NCC Publisher, 1999. – Issue 10. – P. 83–92.

[21] Selikhov A. Emergence and propagation of round autowave on cellular neural network // Lecture Notes in Computer Science. – Berlin: Springer, 1999. – Issue 1662. – P. 20–33.

[22] Selikhov A. Autowave process formation conditions in a cellular neural network // Siberian J. of Computational Mathematics. – 2000. – Vol. 3, № 4. – P. 377–394 (in Russian).