Modeling of a dynamic discrete cellular system

Olga Opalikhina

Department of Mathematics and Mechanics, Saint-Petersburg University of Aerospace Instrumentation (SUAI), 67, Bolshaya Morskaya str., Saint-Petersburg, 197000 Russia

E-mail: sokosapsa@mail.ru

Abstract. Cellular automata have found application in the study of discrete dynamic processes of physical reality. The article deals with the evolution of a discrete dynamic cellular system modeled in the Wolfram Mathematica computer environment. A comparative analysis of various variants of the cellular automaton model implementation is given: according to rules 90, 105 and 110. An algorithm for modeling the deformed state of a solid body using rule 110 is proposed. To simulate local structures that simulate stress concentrators, the algorithm is supplemented by generating a sequence of pseudo-random numbers. The simulation results showed that the appearance of stress concentrators corresponds to the transition from deterministic to chaotic behavior of a discrete cellular system. The article also estimates the possibility of using a discrete cellular system to simulate the turbulent motion of an incompressible fluid based on a generalized local balance model. The appearance of turbulence zones can be estimated by changing the pattern of the cellular automaton.

1. Introduction

The concept of a cellular automaton was first given by John von Neumann and Konrad Zuse [1,2]. The scientists considered a cellular automaton as the equivalent of a Turing machine. The theory of cellular automata by John von Neumann was extended by Arthur Walter Burks [1,2]. Based on the work of John von Neumann, John Horton Conway created a cellular automaton called the game of Life [1,3]. Tommaso Toffoli raised the question of using computer-equivalent information-preserving models in the simulation of physical laws [7]. Issues of the theory and application of cellular automata were considered by Stephen Wolfram. Classifying cellular automata, Stephen Wolfram calculated $2^8=256$ variants of elementary cellular automata, calling them numbers from 0 to 255 [1,4-6,8].

Cellular automata have found application in the theory of dynamical systems in the study of collective phenomena: ordering and chaos, turbulence, fractality, and symmetry breaking [1,4-6,8]. When describing a discrete physical process by a system of differential equations, we proceed to a continuous model that describes a discrete structure with a certain degree of accuracy. Therefore, for the mathematical description of a discrete structure, it is better to use a discrete model – a cellular automaton. In this case, the evolution of a cellular automaton is equivalent to a change in the state of a physical discrete structure. When studying the turbulence of an incompressible fluid, the methods of numerical modeling of the Navier-Stokes equations are used. Analytical methods for solving the Navier-Stokes equations give only approximate values of the flow velocity. Discrete cellular models are adequate to the Navier-Stokes equations [8,9]. Cellular automata are also used in the study of components of the stressed and deformable state of structural elements, in the description of micromechanical structures [10, 11].
2. Evolution of a one-dimensional cellular model

The initial cellular automaton is a one-dimensional binary discrete model with two cell states: “0” and “1”. When modeling a discrete cellular system, we will consider a triad of cells.

For all the cells of the automaton, the ratio is valid [1,4,8]

\[ y_{i+1} = f(y_i, y_{i-1}, y_{i+1}), \quad (1) \]

where \( f \) is the cell transition function; \( y_{i+1} \) is the state of the \( i \)-th cell at the next time moment \( t+1 \); \( y_{i-1} \) is the state of the \( i-1 \)-st cell at the current time moment \( t \); \( y_i \) is the state of the \( i \)-th cell at the current time \( t \); \( y_{i+1} \) is the state of \( i+1 \) cell at the current time \( t \).

To form a model, let us set a rule: let each cell go to a new state by applying a logical operation containing the sum modulo 2.

Let the rule of the sum modulo 2 describe the behavior of an automaton with cells without memory [4,8]

\[ y_{i+1} = y_i \oplus y_{i+1}, \quad (2) \]

where \( \oplus \) (Xor) is the logical operation of the sum modulo 2.

Let's form 40 steps of cell model evolution:

\[ \text{In}[] := \text{ArrayPlot}[\text{CellularAutomaton}[90, \{\{1\}, 0\}, 40], \text{ColorRules} \rightarrow \{1 \rightarrow \text{Red}, 0 \rightarrow \text{Blue}\}] \]

The new states of the central cell are shown in figure 2.

\[ \text{Figure 2. New states of the central cell (rule 90).} \]

Rule 90 is matched by the expression \( 1 \cdot 2^6 + 0 \cdot 2^5 + 1 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 = 90 \).

If we add the state of the cell itself \( y_i \) to formula (2), we get a fractal structure with memory

\[ y_{i+1} = y_{i-1} \oplus y_i \oplus y_{i+1}, \quad (3) \]

Let's create a cellular automaton with a fractal structure with memory:

\[ \text{In}[] := \text{ArrayPlot}[\text{Boole}[\text{CellularAutomaton}[[\text{Xor @ @ # &}, \{\}, \{\{\text{True}, \text{False}\}, 40]\}, \text{ColorRules} \rightarrow \{1 \rightarrow \text{Red}, 0 \rightarrow \text{Blue}\}]\]
The resulting cellular automaton pattern is shown in figure 3.

**Figure 3.** Cellular Automaton Pattern (rule 105).

40 steps of the cell model evolution were formed. This structure corresponds to rule 105. To generate a new state of the central cell of the triad according to rule 105, we use the rewrite system:

\[
\text{In[1]} := \text{RulePlot[SubstitutionSystem[{{{1, 1, 1} \rightarrow 0}, {1, 1, 0} \rightarrow 1}, {1, 0, 1} \rightarrow \{1, 0, 0} \rightarrow 0, \{0, 1, 1} \rightarrow 1}, \{0, 1, 0} \rightarrow \{0, 0, 1} \rightarrow 1}]}, \text{Appearance} \rightarrow \text{"Arrow"}, \text{ColorRules} \rightarrow \{1 \rightarrow \text{Red}, 0 \rightarrow \text{Blue}\}]
\]

The new states of the central cell are shown in figure 4.

**Figure 4.** New states of the central cell (rule 105).

Rule 105 corresponds to the expression \(1 \cdot 2^6 + 1 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 = 105_{10}\). Let's create a cellular automaton according to rule 110 by simultaneously replacing the value in any cell by the sum modulo 2 \(\oplus\) (Xor) of its two neighbors:

\[
\text{In[2]} := \text{ArrayPlot[IntegerDigits[NestList[BitXor[#, 2 \#] &, 1, 40], 2, ColorRules} \rightarrow \{1 \rightarrow \text{Red}, 0 \rightarrow \text{Blue}\}]
\]

The resulting cellular automaton pattern is shown in figure 5.

**Figure 5.** Cellular Automaton Pattern (rule 110).

40 steps of the cell model evolution were formed. To generate a new state of the central cell of the triad according to rule 110, we will overwrite the values:

\[
\text{In[3]} := \text{RulePlot[SubstitutionSystem[{{{1, 1, 1} \rightarrow 0}, {1, 1, 0} \rightarrow 1}, \{1, 0, 1} \rightarrow \{1, 0, 0} \rightarrow 0, \{0, 1, 1} \rightarrow 1}, \{0, 1, 0} \rightarrow \{0, 0, 1} \rightarrow 1}]}, \text{Appearance} \rightarrow \text{"Arrow"}, \text{ColorRules} \rightarrow \{1 \rightarrow \text{Red}, 0 \rightarrow \text{Blue}\}]
\]

The new states of the central cell are shown in figure 6.

**Figure 6.** New states of the central cell (rule 110).

Rule 110 corresponds to the expression \(1 \cdot 2^6 + 1 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 0 \cdot 2^0 = 110_{10}\). The cellular automaton, numbered 110, combines chaotic and deterministic results and is universal. With the help of this machine can be implemented in almost any algorithms \([4,8,12]\).
3. Modeling the deformed state of a solid body

To research the deformed state of a solid body, we choose a cellular system that is implemented similarly to rule 110. The elements of such a system will be three one-dimensional cellular arrays, whose coordinate planes are equivalent to a Cartesian frame of reference and coincide with the direction of the principal stresses. If a deformable solid model is subjected to uniform all-round tension, you can consider a one-dimensional array.

In a small neighborhood of a point $M$ of a rigid body, we select a small surface element - an element of unit volume $\Delta v$. In the Cartesian coordinate system, let us set the radius vector of the point $M$ of this volume $\Delta v$: $r = xi + yj + zk$. Due to the deformation, the point $M(r)$ will move. Its new position in the Cartesian system is defined as $r_1 = r + V(r)$, where $V(r)$ is a shift vector.

We represent the vector field $V(r)$ in a small neighborhood $r$ in the form

$$V(r) = V(r_0) + ((r-r_0)\nabla)V(r_0) + \ldots = V(r_0) + (r-r_0)\Phi_c + (r-r_0)\Phi_a + \ldots,$$

where $\Phi_c$ and $\Phi_a$ are the symmetric and antisymmetric dyads calculated at the point $M_0$, respectively; $\Phi$ is a vector gradient, $\Phi = \Phi_c + \Phi_a = \nabla \cdot V$, $\nabla \times V = \nabla \times V$.

Let's write the expressions for the dyad

$$\Phi = \sum_{i,j=1}^{3} a_{ij}/e_j \cdot e_i,$$

$$\text{Div} \Phi = \nabla \cdot \Phi,$$

$$\text{Rot} \Phi = \nabla \times \Phi,$$

where $a_{ij}$ are coordinates of the dyad $\Phi$ with respect to the basis mutual vectors $e_1, e_2, e_3$; $a_{11}=a_{xx}, a_{22}=a_{yy}, a_{33}=a_{zz}, e_1=e^1=i, e_2=e^2=j, e_3=e^3=k, \nabla \times \Phi = \text{Hamilton operator}.$

The geometric model of the deformed state is determined by expression (4). It follows from expression (4) that the deformation is determined by the transfer $V(r_0)$, the tension $(r-r_0)\Phi_c$, and also by the infinitesimal rotation $(r-r_0)\Phi_a = \frac{1}{2} \text{rot}(V(r_0)) \times (r-r_0)$. Here $\Phi_c$ is the symmetric dyad (deformation tensor). The deformation tensor $\Phi_c$ satisfies the expressions

$$a_{xx}x^2 + a_{yy}y^2 + a_{zz}z^2 + 2a_{xy}xy + 2a_{xz}xz + 2a_{yz}yz = 1,$$

$$\Phi_c = \frac{1}{2} (\Phi + \Phi^*),$$

where $a_{xx}, a_{yy}, a_{zz}, a_{xy}, a_{xz}, a_{yz}$ are the Cartesian coordinates of the tensor $\Phi$, $\Phi^*$ is the dyad conjugate to $\Phi$.

Let us fix a point $M_0$ on the space curve as the origin. The arc length differential $ds$ can be expressed as

$$ds^2 = (dr)^2 = \left(\frac{\partial r}{\partial \nu}\right)^2 d\nu^2 + 2 \frac{\partial r}{\partial \nu} \frac{\partial r}{\partial \mu} d\nu d\nu + \left(\frac{\partial r}{\partial \mu}\right)^2 d\mu^2, r = r(\nu, \mu).$$

The relative deformation $\varepsilon$ of the selected element can be calculated by the formula [13]
where \( ds_1 \) – arc length differential after deformation (moving a point \( M(r) \)).

The following algorithm is proposed for modeling the deformed state of a solid. 1. To determine the position of the element of the cellular template, we calculate the fractal dimension \( D(\mu) \)[14]. We represent a one-dimensional cell array in 3-dimensional graphics. At the first step of the evolution of the cellular automaton, we divide the unit cube selected in the vicinity of the point of the solid body \( M \) into 27 small volumes (figure 7). The number of small volumes \( n \) and the fractal dimension \( D(\mu) \) are related by the expression [14]

\[
n \propto \mu^{-D(\mu)}.
\]

Choosing the scale factor \( \mu = 0.3333333333333333 \), we get \( 27 \propto (0.3333333333333333)^{-3} \). The fractal dimension of this element is \( D(\mu) = 3 \). Without changing the scale factor, we calculate the fractal dimension of the unit cube at the next step in the evolution of the cellular automaton. Divide the edge of the unit cube \( a \) into five segments of equal length. The cubic element now consists of 125 small volumes (figure 7): \( 125 \propto (0.3333333333333333)^{-4} = 4.394920562153781 \). The dimension of the cellular template does not coincide with the dimension of the modeled structure, which indicates a change in its pattern. At the next step, we divide the edge of the unit cube \( a \) into seven segments, we get 343 small volumes, etc. The results of calculating the fractal dimensions are shown in table 1.

### Table 1. Scaling the simulated item

| \( \mu \) | \( D(\mu) \) | \( n \) | \( 27 \) | \( 125 \) | \( 343 \) |
|----------|----------|------|---------|---------|---------|
| \( 0.3333333333333333 \) | 3 | \( 4.394920562153781 \) | 5.313731247484266 |
| \( 0.6180339887498948 \) | \( 6.849035485767837 \) | 10.03655629107328 | 12.131311291951047 |

**Figure 7.** Modeling an item by rule 110 (Graphics3D).

2. The parameters of the deformed state of the modeled element are associated with the change in the pattern

\[
p_{ij}^{t+1} = p_{ij}^t - \varepsilon_{ij} \times \frac{\sum s_k}{S},
\]

where \( p_{ij}^t \) – the number of states of the modeled element (figure 7); \( t \) is the current moment in time; \( \varepsilon_{ij} \) is the value of the deformation tensor component, \( 0 \leq \varepsilon_{ij} < 1 \); \( s_k \) is the number of states of the cellular template; \( S \) is the number of cells participating in the computational process. For \( \varepsilon_{ij} = 0 \) \( p_{ij}^{t+1} = p_{ij}^t \). There
will be no further evolution of the cellular automaton, the fractal dimension of the elementary volume will remain equal to $D(\mu)=3.3$. Let us investigate the evolution of the cellular automaton. At small deformations $0 \leq \varepsilon_{ij} \ll 1$, the number of small volumes $n$ of the modeled element will increase, the pattern will change (figure 7).

To simulate stress concentrators that can change the evolution of the cellular system, we use the generation of pseudo-random numbers:

$$\text{In}[\]:=\text{With}\{\{u=\text{BlockRandom}[\text{SeedRandom}[1, \text{Method}\to \text{"Rule30CA"}]; \text{RandomInteger}[1,100]\}\}, \text{ArrayPlot}[\text{Sum}[\text{CellularAutomaton}\{110, \text{ReplacePart}[u,100\to i,100],\{i,0,1\}\}]*\text{Sum}[\text{CellularAutomaton}\{30, \text{ReplacePart}[u,100\to i,100],\{i,0,1\}\},\{i,0,1\}], \text{ColorRules}\to \{1\to \text{Red}, 0\to \text{Blue}\}]$$

The simulation results of stress concentrators are shown in figure 8.

**Figure 8.** Simulation of voltage hubs.

To generate a two-dimensional cellular automaton, we write:

$$\text{In}[\]:=\text{RandomInteger[]} \times \text{ArrayPlot}[\text{CellularAutomaton}\{\{\text{"Dimension"}\to 2, \"GrowthCases"\to \{3,4\}\}, \{\text{Table}[1,4]\}, \{0\}, \{\{100\}\}\}, \text{ColorRules}\to \{1\to \text{Red}, 0\to \text{Blue}\}]$$

The simulation results are shown in figure 10.

**Figure 10.** Simulation results of turbulent movement.

The modified structures of the cellular system are shown in figure 9. The results of modeling in the Wolfram Mathematica computer environment showed that after the introduction of additional local structures that simulate stress concentrators, the pattern of the cellular automaton, implemented according to rule 110, changes. Fractal structures obtained by rules 90 and 105 retain their pattern.

4. Turbulence research

The generalized local balance model for describing the turbulent motion of an incompressible fluid was developed earlier [15]. To study turbulence, we choose a two-dimensional cellular automaton. We apply the rules of the sum modulo 2 [4,8]

\[
y^s_{i,j} = y^r_{i,j} \oplus y^r_{i-1,j} \oplus y^r_{i+1,j} \oplus y^r_{i,j+1} \oplus y^r_{i,j-1}
\]

\[
y^s_{i,j} = y^r_{i-1,j} \oplus y^r_{i+1,j} \oplus y^r_{i,j+1} \oplus y^r_{i,j-1}
\]

To generate a two-dimensional cellular automaton, we write:

$$\text{In}[\]:=\text{RandomInteger[]} \times \text{ArrayPlot}[\text{CellularAutomaton}\{\{\text{"Dimension"}\to 2, \"GrowthCases"\to \{3,4\}\}\}, \{\text{Table}[1,4]\}, \{0\}, \{\{100\}\}\}, \text{ColorRules}\to \{1\to \text{Red}, 0\to \text{Blue}\}]$$

The results of modeling in the Wolfram Mathematica computer environment showed that after the introduction of additional local structures that simulate stress concentrators, the pattern of the cellular automaton, implemented according to rule 110, changes. Fractal structures obtained by rules 90 and 105 retain their pattern.
We will color the state of the cell “1” in red (a living cell, there are no vortices in the flow around the cylinder), “0” in blue (a dead cell, vortices have formed in the flow around the cylinder). To simulate the flow field, the previously calculated critical Reynolds numbers [6,17] were used. The simulation results showed that at Reynolds numbers greater than 2300, pronounced vortices are formed (figure 11). In the presence of roughness on the cylinder surface, a transition from laminar to turbulent flow is possible [18].

![Figure 11. Imitation of the current field.](image)

5. Conclusion
The analysis of previously carried out mathematical calculations [9, 15-17] and the results of modeling in the computer environment Wolfram Mathematica allow us to conclude that when studying the phenomenon of turbulence, preference should be given to two-dimensional cell arrays. Since automata without memory have a smaller set of logical operations, and the state of the cell itself can be added to the transition function, if necessary, it is possible to select cellular structures without memory. When modeling the deformed state of a solid body, the behavior of an automaton combining chaotic and deterministic results (rule 110) is of interest. The transition to chaotic behavior corresponds to the appearance of stress concentrators that can change the state of a solid body. The proposed modeling algorithm makes it possible to detect local stress concentrators in a selected element of a solid by changing the pattern of the cellular template. In the future, it is planned to carry out mathematical modeling of the destruction of the material of a structural element, to link the evolution of the cellular automaton with the development of cracks and local defects [11].

References
[1] Toffoli T and Margolus N 1991 *Cellular Automata Machines* (Moscow: Mir) p 280 (in Russian)
[2] Von Neumann J 1966 *Theory of Self-Reproducing Automata* ed A W Burks (Urbana: University of Illinois Press) p 408
[3] Gardner M 1970 *Scientific American* 223(4) 120–3
[4] Wolfram S 2002 *A New Kind of Science* (Champaign: Wolfram Media Inc) p 1197
[5] Wolfram S 1986 *Journal of Statistical Physics* 45(3-4) 471–526
[6] Wolfram S 1986 *Adv. Applied Math.* 7 123–169
[7] Toffoli T 1984 *Physica D:Nonlinear Phenomena* 10(1-2) 195-204
[8] Wolfram S 1994 *Cellular Automata and Complexity* (Urbana: Wolfram Research Inc) p 596
[9] Balonishnikov A and Opalikhina O 2020 *JOP Conference Series: Metrological support of innovative technologies* vol 1515 (Bristol: Institute of Physics and IOP Publishing Limited) 22021
[10] Shimizu Y, Hart R and Cundall P 2004 *Proc. Int. Conf. on Numerical modeling in Micromechanics via Particle Methods* - 2004 (Florida: CRC Press) p 448
[11] Opalikhina O V 2020 *Proc. of the X All-Russian Scientific and Methodical Conf. on Actual Problems of Teaching Mathematics in a Technical University* vol 8 (Omsk:OmSTU) pp 204–9 (in Russian)
[12] Wolfram S 2017 *An elementary introduction to the Wolfram Language. Second Edition*
[13] Rabotnov Yu N 1988 *Mechanics of deformable solid body* (Moscow: Nauka) p 712 (in Russian)
[14] Falconer K 2014 *Fractal Geometry* (New York: Wiley) p 398
[15] Balonishnikov A M 2000 *Physical Review E* 61(2) 1390–4
[16] Wu X and Moin P 2008 *Journal of Fluid Mechanics* 608 81–112
[17] Fraas A and Otsisik M 1971 *The calculation and construction of heat exchangers* (Moscow: Atomizdat) p 358 (in Russian)