How cellular automaton plays Minesweeper

Article *in* Applied Mathematics and Computation · September 1997

DOI: 10.1016/S0096-3003(96)00117-8

Andrew Adamatzky
University of the West of England, Bristol

NORTH-HOLLAND

How Cellular Automaton Plays Minesweeper

Andrew Adamatzky

Biophysics Department
St. Petersburg State University
Kubinskaya Street 56-64
St. Petersburg, Russia

ABSTRACT

The Windows Minesweeper game seems to be known to almost all who use personal computers. Mines are distributed inside the bounded room, and empty sites are filled with ciphers, indicating how many neighboring mines exist. At the start of a game we have all mines hidden and wish to mark them all. We construct a two-dimensional deterministic cellular automaton which marks all mines populated on the $n \times n$ size lattice in $\Omega(n)$ time. Every cell of this automaton has 25 closest neighbors including itself and 27 states.

> "*Knowing when to mark a square as a mine is the key to winning the game.*"
>
> —Help. Section, "Marking a Square"

1. INTRODUCTION

Sophisticated progress in the VLSI technology has reattracted interest of the wide scientific community to the massively parallel computing implemented on the processors with cellular automata architecture. It provides a chance to build up a correct theory of cellular automata algorithms, whereas data as computers are embedded in the same space. Using identification theory of cellular automata (dealing with how to reconstruct local transitions rules from the global snapshots of automaton evolution), we designed several nontrivial algorithms for the problems of computational geometry [1]. They are the so-called wave and reaction-diffusion algorithms where waves originating from data sources spread around lattice, react with each

APPLIED MATHEMATICS AND COMPUTATION 85:127–137 (1997)

655 Avenue of the Americas, New York, NY 10010 0096-3003/97/$17.00
PII S0096-3003(96)00117-8

other and form stationary structures in the result of interaction. Distribution of data on the lattice is an input and final stationary structures are the outputs. We demonstrated that cellular automata algorithms can be used efficaciously in the constructing convex hull [2], discrete Voronoi diagram [3]. Moreover, these algorithms were mapped on the structure of the chemical processors intended to approximate planar Voronoi diagram as described in an upcoming paper in Advanced Materials for Optics and Electronics.

In this paper, we try to amalgamate the theory of cellular automata computations with cellular automata games. The most known is a Life game discussed for two and three dimensions in many books and papers (see e.g., [4–6]). A simple combinatorial game of cellular automata origin is presented in [7]. Sutner analyzed the game when source configuration is transformed into target one by the sequence of local transformation (changing of the nodes states) of rectangular lattice [8]. Our attention was focused on the Minesweeper* game because most personal computers users are familiar with it. As it was said in the beginning of the Help "...playing Minesweeper you are presented with a mine field, and your objective is to locate all the mines as quickly as possible. To do this, you uncover the squares on the game board that do not contain mines, and you mark the squares that do contain mines. The trick is determining which squares are which". This is a formulation of the problem which will be solved by appropriate cellular automaton in the paper.

RESULTS

Let us recall what cellular automata are. A cellular automaton (CA) is the tuple $A = \langle \mathbf{L}, \mathbf{Q}, u, f \rangle$, where $\mathbf{L}$ is a d-dimensional lattice or array of $n \times n$ cells ($d \geqslant 1$), $\mathbf{Q}$ is a finite nonempty cell state set, u is a neighborhood (u: $\mathbf{L} \to \mathbf{L}^k$), and f is a cell state transition function (f: $\mathbf{Q}^k \to \mathbf{Q}$). In our case, we will use two-dimensional CA, $d = 2$.

When we ask does the cell (site) contain a mine, we should analyze not only closest neighbors of this cell, but second-order neighbors as well. So, the neighborhood of cell x will consists of 25 cells (full neighborhood with radius 2 including central cell): $u(x_{ij}) = (x_{i-2j-2}, \ldots, x_{ij}, \ldots, x_{i+2j+2})$, $i, j = 1, \ldots, n$. The state of a cell should indicate whether the cell contains a mine and, if not, how many mines are around the cell. Therefore, cell state set will be a direct product $\mathbf{Q} = \{\cdot, \#, \circ\} \times \{0, 1, \ldots, 8\}$, where "$x^t = \#$" means "it

*Microsoft Windows Minesweeper game, © 1985–1992 Microsoft Corp., by R. Donner and C. Johnson.

became known at time t that cell x contains a mine"; "$x^t = \circ$" means "at time t site x becomes open"; "$x^t = \cdot$" means "site x is closed at time t", and figures $0, 1, \dots, 8$ indicate number of mines being in the closest $(x_{i-1, j-1}, \dots, x_{i+1, j+1})$ neighbors. So, duple $\langle x^t, \nu(x) \rangle$, where $x^t \in \{\cdot, \circ, \#\}$ and $\nu(x) \in \{0, 1, \dots, 8\}$, determines the state of cell x at time step t. Moreover, neighbors of x will be able to know $\nu(x)$ only when $x' = \circ$.

When a human plays Minesweeper, he decides whether a site is mined or not, based on circumstantial facts about correlation between ciphers drawn in the neighboring open sites and numbers of mines around them. The deterministic player uncovers the site if he finds that there is at least one neighboring open site, cipher on which corresponds to the number of already marked mines, or there is a neighboring open site with zero cipher. If a player does not mean to uncover a site he may mark it by pointing on the mine. There is a straightforward way to detect a mine: the site is definitely mined if it is a single covered neighbor of such an uncovered site cipher on which there is more than a number of marked mines around.

We offer the following rule for cell state transitions

$$x^{t+1} = \begin{cases} \circ, & (x^t = \cdot) \wedge \Big(\exists y \in u_1(x) \colon y^t = \circ \\ & \qquad \wedge \Big(\nu(y) = 0 \vee \nu(y) = \sum_{z \in u_1(y)} \chi(z^t, \#)\Big)\Big) \\ \#, & (x^t = \cdot) \wedge \Big(\exists y \in u_1(x) \colon y^t = \circ \wedge \sum_{z \in u_1(y)} \chi(z^t, \cdot) = 1 \\ & \qquad \wedge \Big|\nu(y) - \sum_{z \in u_1(y)} \chi(z^t, \#)\Big| = 1\Big) \\ \cdot, & \text{otherwise} \end{cases}$$

where $u_1(x_{ij}) = (x_{i-1, j-1}, \dots, x_{i+1, j+1}) \setminus x_{ij}$, $u_1(x_{ij}) \subset u(x_{ij})$.

A condition for transition $\cdot \rightarrow \circ$ corresponds (with some additions) to the first hint "If an uncovered square already has the correct number of adjacent mines marked, clear around it" ["Using Strategies and Hints", Help for Minesweeper]. Whereas, the useful hint "If an uncovered square is labeled 1, and there is only one covered square touching it, that covered square must be a mine" can be expanded to the condition for transition $\cdot \rightarrow \#$.

The initial, covered in the whole play board, would not be acceptable by CA because it is unable to open any cell. So, at the first step of the game, we help CA and give him a mine field with some uncovered intact sites. Only zero-cipher sites from the given ones can be the sources for waves of uncovering. As shown in Figure 1, only two from three initially uncovered sites have no mines around (Figure 1, $t = 0$). These sites were a sources of the waves spreading in all directions. Starting as excitation or reaction-diffusion waves (Figure 1, $t = 1, 2$), waves of the uncovering change their shape with time (Figure 1, $t = 3, 4, \dots$) due to random distribution of the mines around the mine field. So the whole front of the wave becomes fractured as a coast line. A game is assumed to be finished when CA falls in the fixed point of its global evolution, i.e., configuration of it becomes stationary, $c^{t+1} = c^t$, as at time $t = 27$, Figure 1.

Beginning our discoursing, we assert that CA is deterministic: $\forall w \in \mathbf{Q}^k \exists! a \in \mathbf{Q}$: $f(w) = a$. Contrary to a human player, CA does not make random moves and cannot uncover sites when the abovementioned conditions are not satisfied. Both this fact and nonperiodic boundary conditions lead to the situations when there are regions which cannot be opened. Really, the regions have such combination of the closed, opened, and marked sites on the boundaries that make it impossible to move (Figure 2). From the numerous computational experiments, we found that a situation when the field open in the whole has an island of uncovered sites, it has a very low probability. Rather (under the great ratio of mines) the game will be finished when tiny islands (originating from initially uncovered single sites) of the open and marked sites are in the ocean of closed ones.

PROPOSITION 1. *CA $\mathcal{A}$ finishes Minesweeper on a lattice of $n \times n$ cells with all uncovered or marked sites in $\Omega(n)$ time.*

A wave of computations spreads with the unit velocity, therefore, when we have at time $t = 0$ one open site (with no neighboring mines) being at one of the corners of the lattice, the front of the uncovering reaches the opposite corner in $\Omega(n)$ time. Really, CA may come to the stationary configuration in much less time because of the appearing of some impassable barriers.

PROPOSITION 2. *CA loses at one of two Minesweeper games when any site of lattice is mined with probability more than* 0.15.

Explanatory diagrams on Figures 3 and 4 present results of the experiments with CA playing Minesweeper under various probabilities of mine distribution. They display that critical probability of mine distribution is $0.10 \div 0.16$. In about half of the games, about half of the sites remain

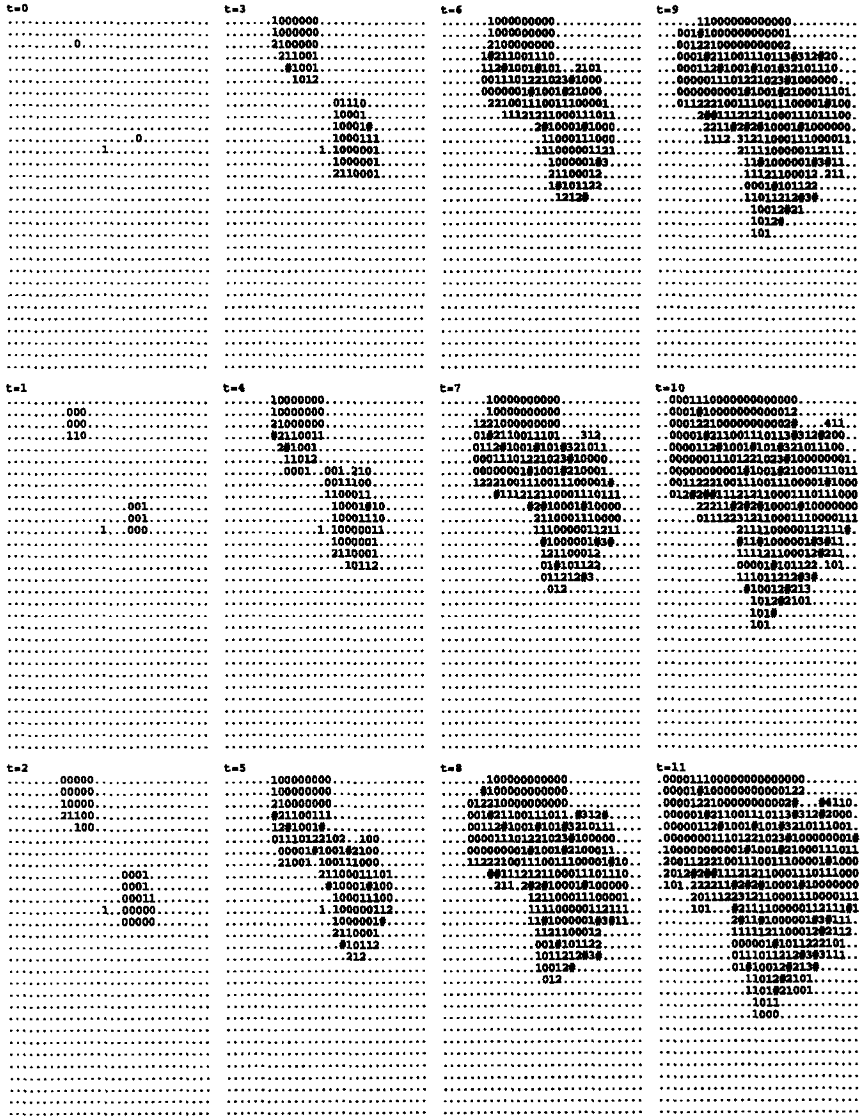

FIG. 1. Evolution of two-dimensional CA A playing Minesweeper. Closed sites are drawn by dots, marked sites (which seem to be mined) by #, and open site are filled with ciphers from 0 to 8 indicating how many mines are in the closest neighborhood of site.

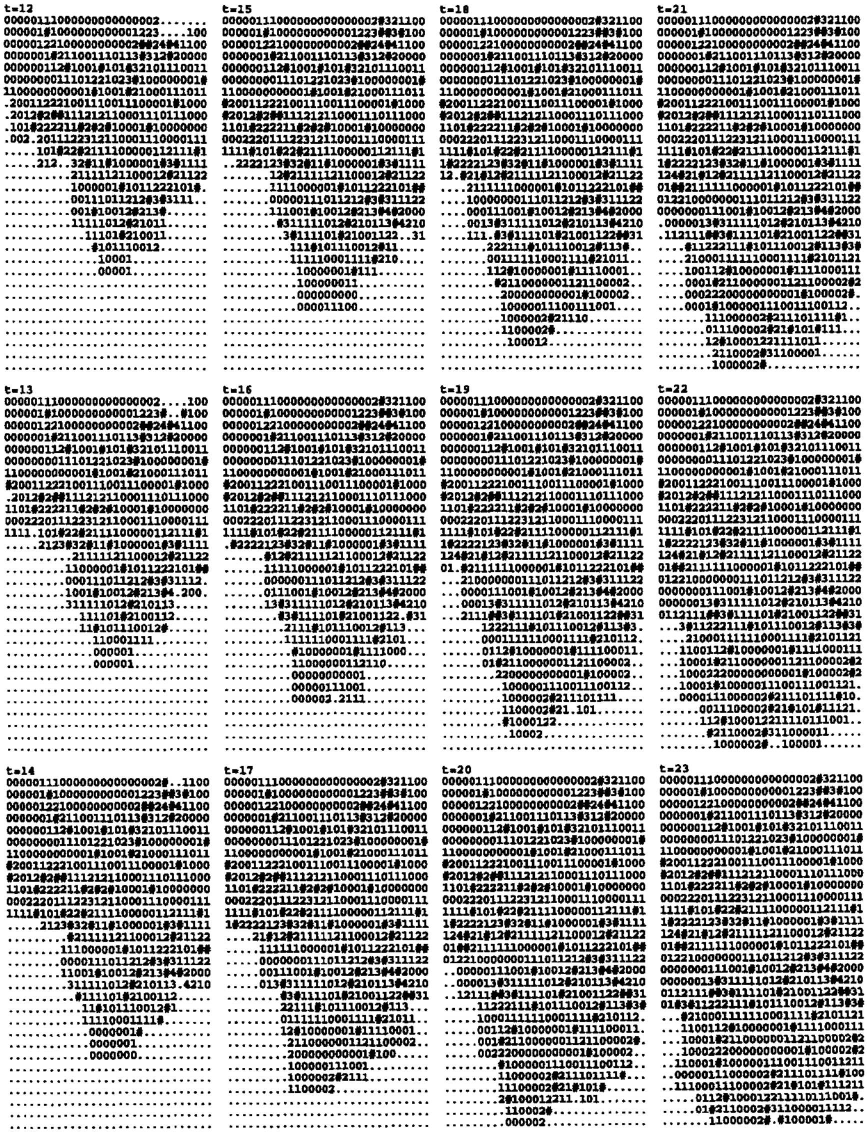

FIG. 1. *Continued.*

FIG. 1. *Continued*

uncovered when the playing field is a torus, and any site is mined with probability 0.10 ÷ 0.14. The similar results will be under fixed boundaries but with mine probability 0.14 ÷ 0.16.

There are several reasons for such low abilities of CA. From one hand, CA starts the game with only three sites (without mines) uncovered randomly. Waves of the uncovering spread only from the sites indexed with 0, i.e., without mines around. So, if, say, all three open sites have one or

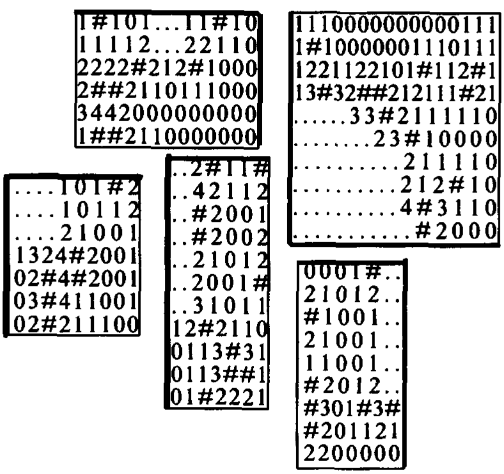

FIG. 2. Islands with impassable bounds. Frame means that these subconfigurations were cut out from larger ones. Thick lines indicate edge of the field.

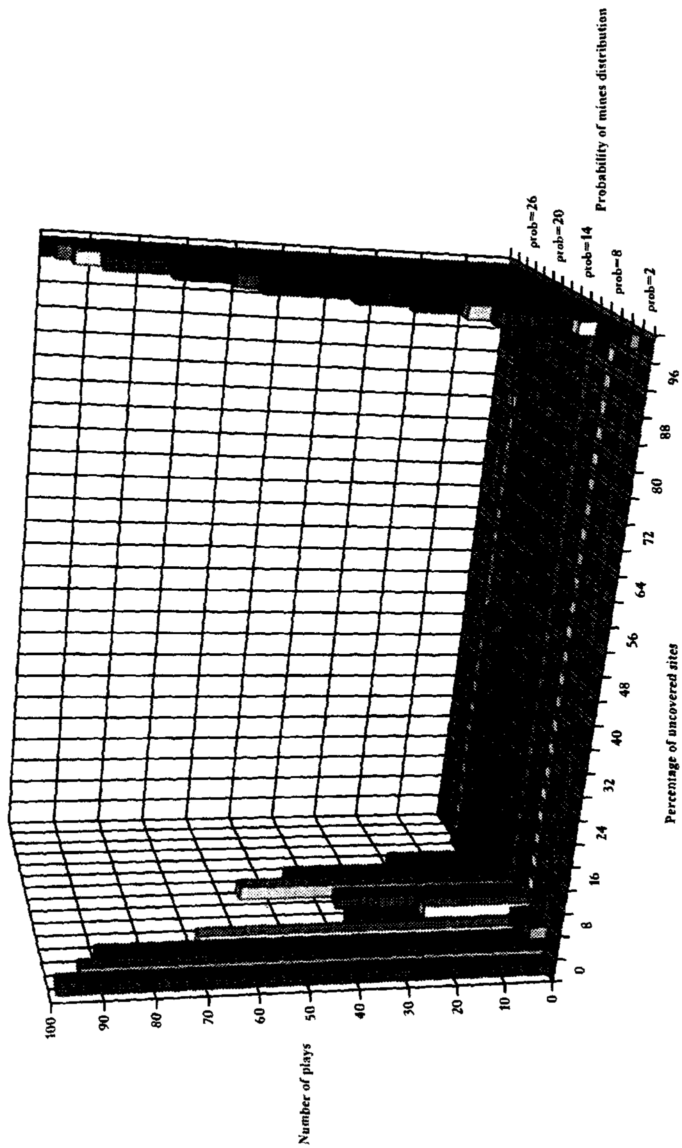

FIG. 3. Characterization of the "playing power" of CA when mine field has fixed boundaries.

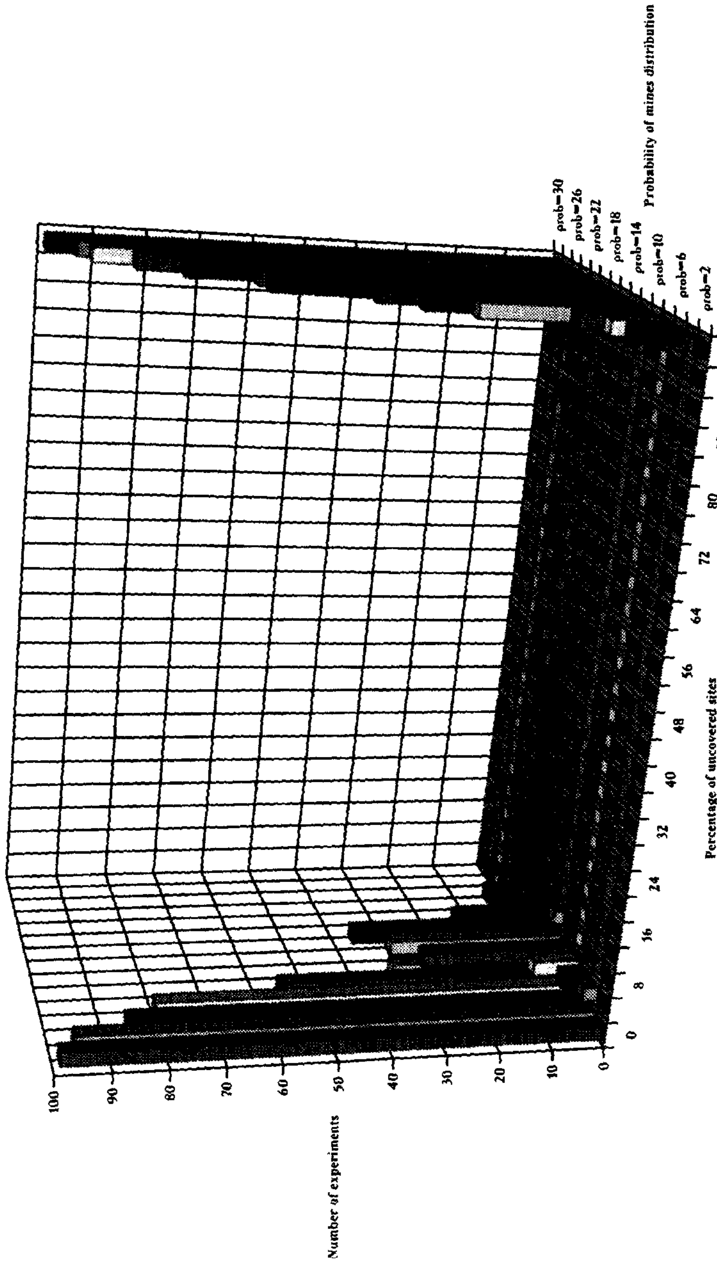

FIG. 4. Characterization of the "playing power" of CA when mine field is a torus.

more cipher, then there is no wave of uncovering in all, and such a configuration of CA remains stationary. On the other hand, CA cannot uncover or mark site in the situation of local uncertainty; therefore, increasing the ratio of mines, we increase probability that the initially uncovered site with zero cipher be encircled with impassable barriers.

Our algorithm without doubts can be modified to such one that sites uncoverable deterministically will be openable stochastically. A human begins to guess probabilistically "does, given site contain mine or not" when from analysis of the whole mine field one decides that there are no covered sites that can be opened or marked definitely. By analogy, any cell of CA A should switch to the regime of divining (making probabilistic moves) only if the current configuration of A is stationary, and some sites remain uncovered. To realize such switching procedures, one should increase cell memory, $x^{t+1} = f(x^t, x^{t-1})$, and supply CA with some broadcasting system. Cell x would switch itself to a divining regime if it knew that $(x^t = x^{t-1}) \wedge (\forall y \in \mathbf{L}\ y^t = y^{t-1})$.

CONCLUSION

A technique offered in this paper can be elaborated in the following ways: (i) detailed investigation of the global evolution of CA playing Minesweeper in different distributions of mines and various probabilities of mining, (ii) adaptation of the algorithm on the learning neural networks which are able to make random choices and have any broadcasting system, (iii) mapping of the problem onto the network of agents communicating with each other locally.

The first two paths seem to be quite technical, but the third one may find a reasonable applications in distributed intelligence, especially as there is interpretation of epistemic and actional distributed systems in the terms of cellular automata [9].

REFERENCES

1 A. Adamatzky, *Identification of Cellular Automata*, London: Taylor and Francis, 1994.

2 A. I. Adamatzky, Computation of discrete convex hull in homogeneous automata networks, *Neural Network World* 3, 241–254 (1995).

3 A. I. Adamatzky, Constructing a discrete generalized Voronoi diagram in reaction-diffusion media, *Neural Network World* 6, 635–643 (1994).

4 C. Bays, Candidates for the game of Life in three dimensions, *Complex Systems* 1, 373–400 (1987).

5 A. K. Dewdney, The game of Life acquires some successors in three dimensions, *Scientific American* 224, 112–118 (1987).
6 M. Gardner, *Wheels, Life and Other Mathematical Amusements*, W. H. Freeman and Co., New York, 1983.
7 R. Anderson, L. Lovasz, P. Shor, J. Spencer, E. Tardos, and S. Winograd, Disks, balls, and walls: analysis of a combinatorial game. *Amer. Math. Monthly* 481–493 (1989).
8 K. Sutner, The σ-game and cellular automata, *Amer. Math. Monthly* 97:1, 24–36 (1990).
9 A. I. Adamatzky, Identification of distributed intelligence, *Izvestija Akademii Nauk Rossii. Seria Tekhnicheskaja Kibernetika.* (Proc. Russian Academy of Sciences, Technical Cybernetics) 3 (1993) (in Russian).