Dynamic Systems Modeling with Stochastic Cellular Automata (Evolutionary versus Stochastic Correlation Approach)

Andrej Dobnikar, Simon Vavpotič and Andrej Likar*

Faculty of Computer and Information Science, University of Ljubljana, Slovenia
*Faculty of Mathematics and Physics, University of Ljubljana, Slovenia

A new approach to dynamic systems modeling is given. Stochastic Cellular Automata (SCA) are used as the basic computational module. The dynamic systems are considered as time and space dependent, where time dependencies are supposed to be given with some differential equations (DE), while space influences are not known. The basic idea of our approach is to use heuristics for the design of SCA and some stochastic search algorithm to optimize free model parameters. Two non-gradient optimization algorithms are used and evaluated on the two case studies: diffusion and migration of Cs in soil and forest fire spread problem. They are Evolutionary Algorithm (EA) and Stochastic Correlation Algorithm (ALOPEX). We show that with some modifications, both algorithms are capable to solve the two case problems, though there are some important differences between them.

Keywords: modeling, cellular automata, heuristics, stochastic optimization.

1. Introduction

The modeling of unknown systems dynamics is still one of the biggest challenges of modern science. Finding the model, for example, of a natural phenomenon enables one to gain deeper insight into the observed appearance on the one hand and to enable one to forecast its activity on the other. Although well known phenomena already have some reasonable physical interpretations, this does not usually allow us to use them for prediction purposes. The reason for this is that such dynamical systems are time and space dependent, where the time dependencies are normally given with some differential equations, while the space influences are captured within unknown space-dependent parameters. We believe that the model should provide us with a tool to acquire these values.

This paper describes dynamic systems modeling with evolving cellular automata. The basic idea is to use stochastic cellular automata and heuristics for their design and some stochastic search algorithms to optimize free model parameters. We compare this approach with more traditional mathematical modeling. Within this paper we follow [2], where the contribution of a heuristic method should either results in: high-quality solutions, produce faster, higher-quality solutions, easier implementation or a broader range of applicability, in relation to other approaches. We believe that the heuristics we use fulfill at least some of the above criteria and we intend to prove it through our experimental work. Two non-gradient optimization (search) algorithms are used and evaluated on two real world problems: diffusion and migration of Cs in soil (1D problem) and forest fire spread (2D problem). The first is based on Evolutionary Algorithm (EA) and the second on Stochastic Correlation Algorithm (ALOPEX).

The paper is organized as follows: stochastic cellular automata are defined first, then we describe two versions of evolutionary algorithms that are used in our experimental work. Section 4 provides the basic concept of a stochastic correlation algorithm. A heuristic approach of modeling with SCA is outlined in Section 5,

after a brief description of a more traditional (mathematical) modeling, through a description of our experimental work. We conclude with some comments and plans for our future work.

2. Stochastic Cellular Automata (SCA)

'Cellular automata are discrete dynamical systems whose behavior is completely specified in terms of a local relation, much as is the case for the large class of continuous dynamical systems defined by partial differential equations. In this sense, cellular automata are the computer scientist's counterpart to the physicist's concept of "field",' [13].

We want to follow this basic idea. Therefore, our goal is to implement a physical model of a dynamical system with the help of *SCA*. We chose to use an evolution-based procedure in order to optimize the space-dependent parameters. This approach differs from the one described in [11] significantly. Instead of searching for the most appropriate non-homogeneous and deterministic structure of the Cellular Automata, we look for a stochastic and homogeneous version of CA, which we call SCA. Stochasticity is due to the frequent feature of natural phenomenon (like diffusion or fire spread) and homogeneous structure of CA because of its usual location-invariant dynamics.

SCA is n-D array ($n = 1, 2, \ldots$) of finite automata (cells). Each automaton (cell) is described with:

$$\mathrm{SCA}(\vec{r}) = \langle \mathbf{S}, \vec{p}, \mathbf{L}, f \rangle \qquad (1)$$

where:

$\vec{r}$ — n-D vector, indicating the cell
$\mathbf{S}$ — set of all possible states of the cell
$\vec{p}$ — probability vector, describing state transitions for each neighboring combination
$\mathbf{L}$ — set of local parameters
f — fitting value of a cell

$\vec{p}$ and $\mathbf{L}$ are parameters of unknown values. In the approach proposed in this paper a heuristic method is used to design the state transition diagram of SCA and an evolution-based search procedure to optimize the corresponding free parameters, in order to achieve the greatest possible fitting with the measured data of the investigated phenomenon.

3. Evolutionary Algorithm (EA)

An evolutionary algorithm was developed by J. Holland and his students in 1960, [4]. Originally, it was designed as a formal system for adaptation rather than an optimization system. Its basic features are the strong emphasis on recombination (crossover), use of a probabilistic selection operator (proportional selection) and the interpretation of mutation as a background operator. Due to the nature of the operators, this algorithm is also called genetic algorithm. Later, several methods of simulating evolution were developed. Certain historical subsets of evolutionary algorithms include evolution strategies, evolutionary programming and genetic algorithms, [1]. We used only two methods within our experimental work. The first is a population-based procedure that relies on a random variation (with crossover and mutation) and selection and is indicated as genetic algorithm (GA). The second relies on evolution of individual, again with the help of genetic operators crossover and mutation. We mark it with CPA (cellular programming algorithm), [11], because it was developed for the purpose of the cellular automata synthesis. The modifications within CPA and GA, in relation to the original algorithms, are due to the stochastic feature of the SCA. Crossover calculates new probabilities for the state transitions from the two individuals (parents) or from the two neighboring cells, by considering probability rules. Mutation replaces the chosen probability values with new ones, again within the limits [0,1]. We apply a variant of GA in the case of 1 D problem (migration of Cs in soil) and a modified version of CPA in case of 2D problem (forest fire spread).

4. Stochastic-Correlation Optimization Algorithm (ALOPEX)

Instead of an error gradient, ALOPEX [14] uses correlations between changes in individual parameters and changes in global error measurement. In the case of 2D SCA, k-th parameter of

(i, j) cell P_{ij}^k is changed according to the expression:

$$P_{ij}^k(n) = P_{ij}^k(n-1) + \delta_{ij}^k(n) \qquad (2)$$

where:

$$\delta_{ij}^k(n) = \begin{cases} -\delta & , \text{ with } p_{ij}^k(n) \\ +\delta & , \text{ with } 1 - p_{ij}^k(n) \end{cases}$$

where:

$$p_{ij}^k(n) = \frac{1}{1 + e^{-C_{ij}^k(n)/T(n)}},$$

$$(Boltzman\ distribution)$$

$$C_{ij}^k = \Delta P_{ij}^k(n) \cdot \Delta E(n)$$

$$\Delta P_{ij}^k(n) = P_{ij}^k(n-1) - P_{ij}^k(n-2)$$

$$\Delta E(n) = E(n-1) - E(n-2)$$

δ is a fixed step of parameter change and T is a 'temperature' which is high at the beginning and low at the end of the 'learning' procedure — it gives a stochastic feature to the algorithm. If it is too high, the probability is close to 0.5 and if it is too low, the probability is close to 1:

$$T(n) = \frac{1}{NM} \sum_i \sum_j \sum_k \sum_{n'=n-N}^{n-1} |\langle C_{ij}^k(n') \rangle|, \qquad (3)$$

if n is a multiple of N, otherwise it remains unchanged. N denotes the number of iterations when T is updated. M is the total number of parameters in the system that we want to optimize. $|\langle C_{ij}^k(n) \rangle|$ is equal to the average absolute value of $C_{ij}^k(n)$ over the interval N. When correlations are small (near minima of error surface), T is reduced and vice versa. There are several similarities with simulated annealing, but also some important differences. In ALOPEX, correlations are used instead of changes in error, all parameters are changed at every iteration, and all parameters are updated at the same time.

5. Modeling of Dynamic Systems

We are surrounded by a number of different natural phenomena that can be classified as dynamic systems. It is of vital interest to know their dynamics in order to be able to analyze them from different points of view, in particular to predict their actions in future. There is a traditional mathematical modeling, that uses differential equations in order to describe the time dependencies of such dynamic systems. However, space influences and stochastic nature of the phenomena make such modeling very complicated so that the solution, if there is any, is barely feasible. In such cases we need an alternative and here we suggest a heuristic modeling in combination with SCA, together with two non-gradient optimization algorithms, evolutionary-based (GA, CPA) and stochastic correlation algorithm (ALOPEX), to search all free model parameters. In the rest of the paper we shall demonstrate this alternative modeling approach on two case studies from the real world, migration of Cs in soil and forest fire spread.

5.1. 1D Case Study: Diffusion of Cs in Soil

Careful measurement of specific activity of Cs in soil samples taken layer by layer in autumn 1999 in Slovenia (near the nuclear power plant Krško) is confronted with the results of two types of modeling: traditional mathematical modeling and heuristic modeling with evolving SCA. The deposit that influences the content of Cs in undisturbed soil is due to the nuclear weapon tests (from 1952 to 1972) as well as to the Chernobyl accident in 1986 (short term deposit). The main purpose of modeling this phenomenon is to predict the Cs migration in future.

A.) Mathematical modeling

Several mathematical models of diffusion and migration processes have been published so far [5, 6, 12]. They all use the diffusion-convection differential equation:

$$D\frac{\partial^2 u}{\partial z^2} = \frac{\partial u}{\partial t} + v\frac{\partial u}{\partial z}, \qquad (4)$$

where D is the diffusion constant and v is the migration speed.

Different forms of the Green function which obey the diffusion-convection equation are used. Recently, a solution was published, [7], which also considers the boundary condition (no leak current of the activity across the soil surface):

$$(D\frac{\partial u}{\partial z} - vu)|_{z=0} = 0 \qquad (5)$$

This boundary condition alters the previously-used Green function considerably. It leads to the form:

$$G(z,t) = \frac{e^{\frac{v}{2D}(z-\frac{vt}{2})}}{\sqrt{\pi Dt}} \left[e^{-\frac{z^2}{4Dt}} - \frac{v}{2}\sqrt{\frac{\pi t}{D}} e^{\frac{v}{2D}(z-\frac{vt}{2})} \right.$$
$$\left. \left(1 - \Phi\left(\frac{z+vt}{\sqrt{4Dt}}\right)\right) \right] \quad (6)$$

where $\Phi(x)$ is the error function defined as:

$$\Phi(x) = \frac{2}{\sqrt{2\pi}} \int_0^x e^{-t^2} dt$$

With the help of $G(z,t)$, the solution for the depth profile was obtained from:

$$u(z,t) = \int_0^\infty G(z, t-t') f(t') e^{\frac{t-t'}{\tau}} dt' \quad (7)$$

The fallout function $f(t')$ was taken [7] as:

$$f(t') = \alpha\delta(t' - t_C) + \beta F(t') \quad (8)$$

where:

α, β - are scaling factors for nuclear test weapons fallout deposit $F(t')$ (7, 7 kBq/m^2) and the Chernobyl accident deposit δ (19, 1 kBq/m^2).

Free parameters (α, β, D, v) are optimized with Marquardt method [8] by controlling $u(z,t)$ with experimental data (near NPP Krško, 99). The result is very good, except for the greater depths (Fig. 1).

Cs depth profile [Bq/kg]
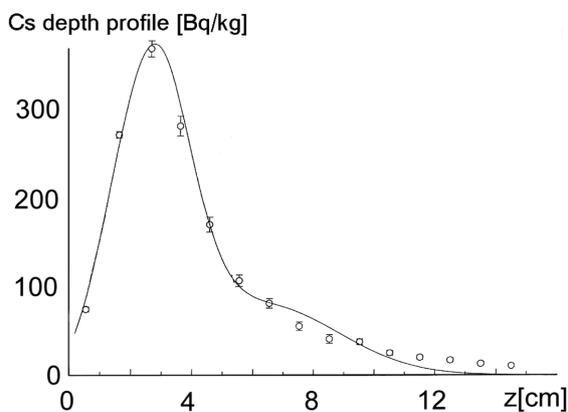
Fig. 1. The measured profile of Cs (circles) and the theoretical fit using the diffusion — convection model (solid line).

The mathematical modeling is obviously very demanding. An alternative approach is therefore beneficial. In the following, the SCA approach is described.

B.) SCA modeling

1D SCA is suggested, where each cell (automaton) is responsible for the description of the depth profile (value of Cs concentration) within a range of 1cm (Fig. 2).
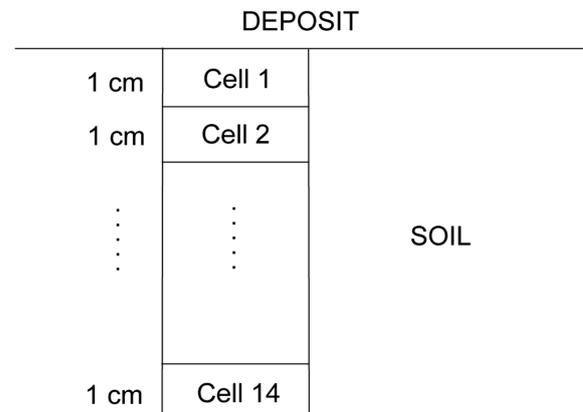
Fig. 2. Cellular automata structure.

The following heuristics is used: concentration of cell i ($i = 1, 2, \ldots, N$) is increased if the neighboring concentrations are higher and is decreased otherwise. The changes also depend on diffusion constant and migration speed. The concentration is described with the digital value of the corresponding cell's state. The state of cell i is changed according to the equation:

$$u_i(t+1) = u_i(t) + P_i \cdot \Delta^+ - (1 - P_i) \cdot \Delta^-, \quad (9)$$

where:

Δ^+, Δ^- — are step factors proportional to p_i and q_i, respectively, and:

$$P_i = \frac{p_i}{p_i + q_i} \quad Q_i = (1 - P_i) = \frac{q_i}{p_i + q_i} \quad (10)$$

$$p_i = D^* \left[(1 + \text{sgn}(u_{i-1} - u_i)) \frac{(u_{i-1} - u_i)}{2} \right] v^* +$$
$$+ D^* \left[(1 + \text{sgn}(u_{i+1} - u_i)) \frac{(u_{i+1} - u_i)}{2} \right]$$

$$q_i = D^* \left[(1+\text{sgn}(u_i-u_{i-1})) \frac{(u_i-u_{i-1})}{2} \right] v^* +$$

$$+ D^* \left[(1+\text{sgn}(u_i-u_{i+1})) \frac{(u_i-u_{i+1})}{2} \right]$$

where D^* and v^* are again estimates of diffusion constant and migration speed, P_i and Q_i are probabilities of i'th cell concentration increment and decrement, respectively, and $sgn(x)$ is the sign function of x, being $+1$ if $x > 0$, 0 if $x = 0$ and -1 if $x < 0$. Note the influence of both D^* and v^* on p_i and q_i in the direction of the migration, and the influence of only D^* on p_i and q_i in the opposite direction. The equations above, therefore, clearly indicate nonlinear dependence, which would not be the case with the discrete version of the diffusion-convection differential equation.

The deposit (fallout of Cs) at the surface is the same as before. This is shown in Figure 3 with the values, real up to the scaling factors α^* and β^*.
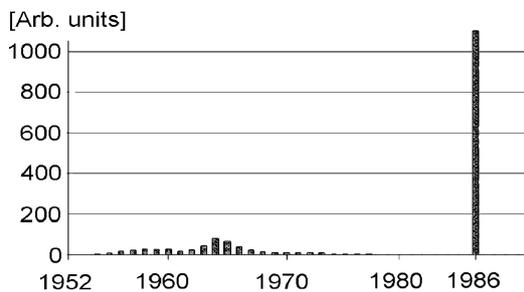
Fig. 3. Nuclear experiment deposits of radioactive Cs between 1952 and 1977 and the Chernobyl deposit of Cs in 1986 in arbitrary units.

With this input data we made two simulation runs, one with evolution-based algorithm (GA) and another with stochastic correlation algorithm (ALOPEX) in order to optimize the free parameters of the SCA model ($\alpha^*, \beta^*, D^*, v^*$), so that it would fit best with the experimental data. We first considered the above parameters as global parameters. For each search method (GA, ALOPEX) we tested several variants and for each variant we made ten tests. For the evaluation we took the average tests of the best variants. Two variants of GA were tried, one with discrete (GA_d) and another with real (GA_r) parameter values. We also tested two versions

of ALOPEX, one with fixed amounts (AL_f) of parameter changes and the other with changeable steps (AL_c), following the general principle of Rechenberg's 1/5 success rule, [1]. Table 1 gives the average number of simulation runs needed to reach the error obtained by mathematical modeling (50 Bq/kg, [7]), for the four mentioned methods, where the error is calculated as the square root of the sum of squared differences between the measured and calculated concentrations. Timing of the best result (AL_f) is shown in Fig. 4.
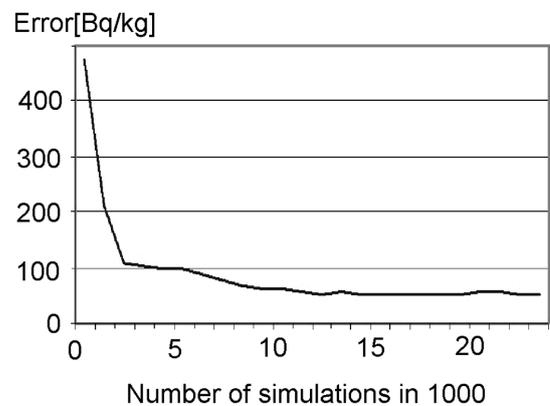
Fig. 4. Timing of the fixed step Alopex method.

Method	Av. no. of steps ($\times$ 1000)
GA_d	35.9
GA_r	27.5
AL_f	21.9
AL_c	32.7

Table 1. Average number of simulation runs
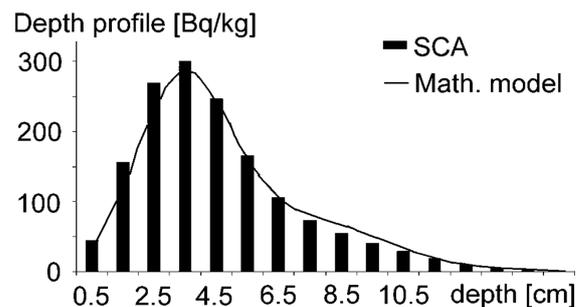
Fig. 5. Prediction of Cs migration in soil for the year 2003.

An interesting result was obtained by combining GA_r and AL_f (GAAX), which means that we used the evolution-based procedure together with the learning algorithm (the approach that gained the name Lamarckism, [1], opposite to Darwinism). The average number of simulation runs to reach an error of 50Bq/kg was 20.5, which indicates that the combining methods are compatible. We also tried to improve the above results (decrease the error) by considering D^* and v^* as local parameters. In this case, we reached an error of 2.7 Bq/kg in 11,000 generations (11 million simulation runs). This exceptional result was obtained with the GA_r method. Finally, it is interesting to look at the forecasting of our results. The global parameters, obtained with mathematical and SCA modeling, were used and some further simulations were performed in order to obtain predictions of the Cs migration for the year 2003 with the two methods (Fig. 5). Time-space diagram for AL_c is further given in Fig. 6. It was not possible to get a similar diagram with mathematical modeling, which is due to the numerical algorithms that were used within the mathematical modeling.
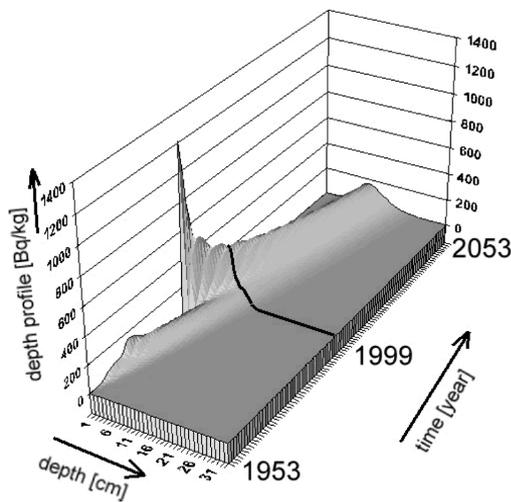
Fig. 6. Time-space diagram of Cs migration in soil obtained by SCA (AL_c) modeling .

5.2. 2D case study: Forest Fire Spread

Forest fire spread is a well known and a very serious problem in Mediterranean countries and elsewhere. Modeling the dynamics of fire spread would enable prevention of huge damage and many human casualties.

A.) Mathematical modeling:

The traditional mathematical modeling for fire spread is based on the fundamental equation of energy law:

$$\frac{\partial h}{\partial t} = -\mathrm{div}\vec{j} + \sigma \qquad (11)$$

h — specific enthalpy of medium
$\vec{j}$ — energy flux density
σ — specific power of heat generation

where:

$$\vec{j} = -\alpha(\vec{r})\mathrm{grad}(\mathrm{T}) + \rho c_p T \vec{\omega},$$

α — conductivity field
$\vec{r}$ — n-D vector, indicating the cell
T — temperature
ρ — air density
c_p — specific heat of air at const. pressure
$\vec{\omega}$ — wind velocity

The temperature spread is then:

$$\frac{\partial T}{\partial t} = \frac{1}{\rho c_p}\frac{\partial h}{\partial t} =$$
$$= \frac{1}{\rho c_p}\ \mathrm{div}(\alpha(\vec{r})\ \mathrm{grad}(T) - \mathrm{div}(\vec{\omega}T) + \frac{\sigma}{\rho c_p}$$
$$= \beta\ \mathrm{div}(\alpha(\vec{r})\ \mathrm{grad}(T) - \mathrm{div}(\vec{\omega}T) + \Omega \qquad (12)$$

where:

α, β, Ω — are unknown local parameters that depend on geographical and meteorological data.

The above equation represents the diffusion model of the forest fire spread. We shall omit its direct implementation, it is detailed elsewhere [3, 9]. The model is used for reference (measured) sequence generation within our experiments. The reason for this is the fact that we had no real data at our disposal.

B.) SCA modeling

Each cell in the 2D array of SCA covers one part of the area under investigation (Fig. 7). It means that we describe each part of the area through the states of relating stochastic automaton (cell). The structure of its state diagram

therefore follows the heuristics applied to this problem.

Fig. 8. State diagram of cellular automaton cell for fire spread research.

Fig. 7. Stochastic cellular automaton model for fire spread research.

SCA model of a forest fire spread can be interpreted also with the array ('forest') of cells ('trees'), where each cell has only three possible states (due to the fact that we are interested in the fire spread only): 0: unaffected tree (low temp., fuel available), 1: burning tree (rising temp., fuel being burned). 2: burned tree (falling temp., no fuel left). The fire from one cell within the diffusion model ignites a cell in its vicinity with some probability $v(\vec{r})$. In SCA approach the other probability $P(\vec{r})$ is in order, describing the fire spread to cell $\vec{r}$ because of its neighboring cells, considering also the wind and the structure (fuel) of the cell itself. As we have some knowledge about it (structure of terrain, meteorological data, for example), we distinguish between initial and unknown data. We designate the former with $p_0(\vec{r})$ and the latter with $p(\vec{r})$.

Total ignition probability is now:

$$P(\vec{r}) = p_0(\vec{r}) + p(\vec{r}) \qquad (13)$$

$p(\vec{r})$ is not known in advance. It depends on the state of all relevant neighboring cells. We want to optimize it for all cells in order to fit the SCA dynamics to the measured data.

The state transition diagram for all the cells in SCA is very simple due to the straightforward heuristics (Fig. 8).

The two non-gradient procedures are again used to optimize local parameters $p(\vec{r})$ for all the

cells in SCA. We applied Al_f (ALOPEX with fixed steps) and modified CPA. In both cases we evaluated the procedures by comparing the errors at different evolution steps, where the error (sum of state differences between the equivalent cells from measured (reference) and the generated configurations within the two sequences) is taken over the area of observation (cells in the "fire" state) and during the teaching sequence. Figure 9 shows the fitting increase of $(1 - \text{Error})$ during the evolution of SCA with the modified
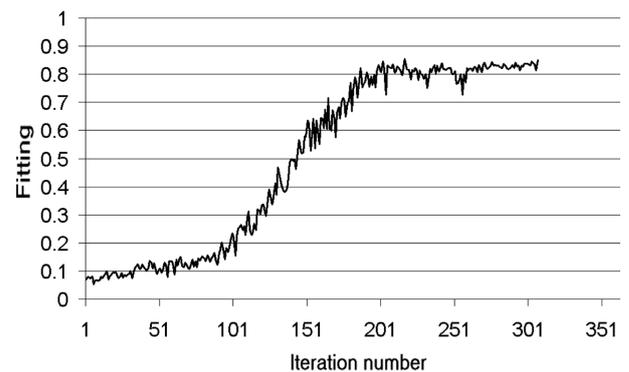
Fig. 9. Fitting $(1 - \text{Error})$ of SCA to the reference sequence of the cells in "fire" state with modified CPA algorithm.
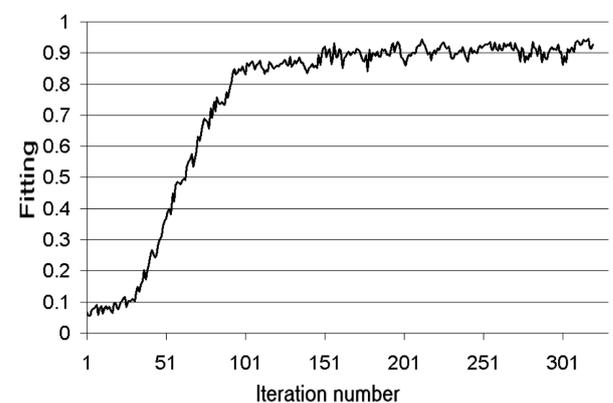
Fig. 10. Fitting $(1 - \text{Error})$ of SCA to the reference sequence of the cells in "fire" state with modified Al_f algorithm.

Al_f and Figure 10 with the modified CPA. Al_f is used here for each cell separately but with a local exchange of the parameters when this increases the fitting of a cell. Without this modification, Al_f procedure is too slow to be comparable with modified CPA algorithm. Though they both show the effect of incremental fitting, the modified Al_f outperforms CPA in both speed and accuracy. It is difficult to say what the reasons are, however we speculate that one of them is the fact that we partition the problem into subproblems with simpler dynamics within Al_f, which makes fitting easier.

6. Conclusion

SCA modeling was tested against more traditional mathematical modeling. For non-trivial dynamical systems it is difficult to find the proper analytical model. With the help of modern heuristics and a simple model of processing (array of stochastic automata), we are able to build models for relatively complex dynamical systems. Non-gradient optimization methods are essential for successful modeling with SCA. Both of the methods used, evolutionary algorithm EA (with two variants GA and CPA) and stochastic correlation algorithm (ALOPEX), proved to be successful. Each method has positive and negative features. Our general experience is that ALOPEX is easier to implement then EA, but it is also very sensitive to the values of the parameters, such as step size, N (number of steps for updating T), etc. On the other hand, EA is more robust, but usually slow, particularly if the search space is large. As they are, due to their features, compatible, we found combining them to be attractive. If, after each generation, the individuals are updated for a certain number of steps according to ALOPEX, then the convergence speed is considerably increased.

In our future work we want to extend the approach to other models like cellular neural networks and to prepare transformations of heuristics for different dynamics to cellular structures.

References

[1] T. BACK, D. B. FOGEL, T. MICHALEWICZ, *Evolutionary Computation 1, 2*, Inst. Of Physics Publ., 2000.

[2] R. S. BARR, B. L. GOLDEN, J. K. KELLY, M. G. C. RESENDE, W. S. STEWART (1995): Designing and Reporting on Computational Experiments with Heuristic methods, in *Proceedings of the International Conference on Metaheuristics for Optimization*, Kluwer Publishing, Norwell, MA, pp. 1–17

[3] A. DOBNIKAR, S. VAVPOTIČ, A. LIKAR, *Dynamic Systems Modeling with Evolving Cellular Automata*, in A. Dobnikar etal., *Artificial Neural Nets and Genetic Algorithms*, Springer, 1999.

[4] J. H. HOLLAND, *Outline for logical theory of adaptive systems*, J. ACM 3 (1962).

[5] O. V. KONSHIN, *Mathematical model of ^{137}Cs migration in soil: Analysis of observation following the Chernobyl accident*, Health Physics 63(3) (1992), 301–306.

[6] M. KORUN, A. LIKAR, M. LIPOGLEVŠEK, R. MARTINČIČ, B. PUCELJ, *In-situ measurements of ^{137}Cs distribution in the soil*, Nulear. Instrumental Meth. B 93 (1994), 485–491.

[7] A. LIKAR, G. OMAHEN, M. LIPOGLAVŠEK, T. VIDMAR, *Diffusion and migration of ^{137}Cs in soil*, to appear in *J. Environ. Radioactivity*, 2000.

[8] W. H. PRESS, S. A. TEULKOSKY, W. T. VETTERLING, B. P. FLANNERY, *Numerical Recipes in Fortran*, Cambridge Univ. Press, (1994)

[9] D. D. RICHARDS, *Int. J. Wildland Fire*, 5(2): (1995), 63–72.

[10] E. SANCHEZ, M. TOMASSINI, *Towards Evolvable Hardware*, Springer, 1996.

[11] M. SIPPER, *Evolution of Parallel Cellular Machines*, Springer, 1997.

[12] P. SZERBIN, E. BOKORI, L. KOBLINGER, L. VERGVARI, A. UGRON, *Cesium-137 migration in Hungarian soils*, Sci. Tot. Eviron. (1999), 215–227.

[13] T. TOFFOLI, N. MARGOLUS, *Cellular Automata Machines*, MIT Press, 1988.

[14] K. P. UNNIKRISHNAN, K. P. VENUGOPAL, *Alopex: A Correlation-Based Learning Algorithm for Feedforward and Recurrent Neural Networks*, Neural Computations 6, (1994), 469–490 (1994).

[15] R. O. WEBER, *Int. J. Wildland Fire*, 1(4): (1991), 245–248.

Contact address:
Andrej Dobnikar
Faculty of Computer and Information Science
University of Ljubljana
Tržaška 25
1000 Ljubljana
Slovenia
e-mail: Andrej.Dobnikar@fri.uni-lj.si

Simon Vavpotič
Faculty of Computer and Information Science
University of Ljubljana
Tržaška 25
1000 Ljubljana
Slovenia
e-mail: `Simon.Vavpotic@fri.uni-lj.si`

Andrej Likar
Faculty of Mathematics and Physics
University of Ljubljana
Jadranska 19
1000 Ljubljana
Slovenia
e-mail: `Andrej.Likar@fmf.uni-lj.si`

ANDREJ DOBNIKAR was born on 26 October 1946 in Ljubljana, Slovenia. He received his B.Sc. (1971) and M.Sc. (1973) degrees in electrical engineering and Ph.D. degree (1984) in computer science, all from the University of Ljubljana. He became assistant professor of computer science in 1986, associate professor in 1991 and full professor in 1996. His research interests are neural networks, soft computing, parallel processing and adaptive systems. Andrej Dobnikar is a member of IEEE and ISNN.

SIMON VAVPOTIČ was born in 1972 in Ljubljana, Slovenia. He received a B.Sc. degree (1996) and M.Sc. degree (2001) from the University of Ljubljana and is working towards a Ph.D. in computer science. His research interests are neural networks, parallel processing and adaptive systems.

ANDREJ LIKAR was born on 1st of June 1948 in Ljubljana, Slovenia. He received his B.Sc. (1971) and M.Sc. (1974) degrees in physics and Ph.D. degree (1978) in physics, all from the University of Ljubljana. He became assistant professor of physics in 1981, associate professor in 1986 and full professor in 1991. His research interests are experimental nuclear physics, measurements in physics and optimal filtering.