SURFACE DYNAMIC PROCESS SIMULATION WITH THE USE OF CELLULAR AUTOMATA*

Maja Adamska-Szatko, Justyna Bała

Department of Geoinformatics and Applied Computer Science
Faculty of Geology, Geophysics and Environmental Protection
AGH University of Science and Technology
Al. Mickiewicza 30, 30-059 Kraków, Poland

(Received February 18, 2010)

Cellular automata are known for many applications, especially for physical and biological simulations. Universal cellular automata can be used for modelling complex natural phenomena. The paper presents simulation of surface dynamic process. Simulation uses 2-dimensional cellular automata algorithm. Modelling and visualisation were created by in-house developed software with standard OpenGL graphic library.

PACS numbers: 07.05.Tp, 89.75.–k, 92.40.Ha, 45.70.Cc

1. Introduction

Surface dynamic processes (landslides, avalanches, flows) are very dangerous to people. This is why it is so important to investigate the phenomena and represent their realistic modelling. Simulation may help to decrease the risk in the future determining probable routes, building special security barriers in chosen places, *etc.* However, discussed natural phenomena are of very complex character and it makes the simulation difficult to run. Cellular automata are known for many applications, especially for physical and biological simulations. Important feature of universal cellular automata is the influence of a single cell on the whole system. This mathematical model is adequate for analysing and simulating dynamic complex natural phenomena. Finding proper physical parameters are still main problems in natural phenomena modelling.

In the last decades, CA proved to be a valid alternative to differential equations in simulating complex natural phenomena [1, 2]. Many attempts of simulating flow-type landslides have recently been carried out by several authors, also through CA models with satisfactory results [3–6]. Among

* Presented at the Summer Solstice 2009 International Conference on Discrete Models of Complex Systems, Gdańsk, Poland, June 22–24, 2009.

these efforts, the CA model SCIDDICA (Simulation through Computational Innovative methods for the Detection of Debris flow path using Interactive Cellular Automata) was developed for simulating simple cases of flow-like landslides. Subsequent releases of SCIDDICA were successfully applied to the 1984 Mt. Ontake, Japan, debris avalanche [7] and to the 1992 Tessina, Italy earth flow [8]. Recent applications concerned debris flows which had occurred in May 1998 at Pizzo d'Alvano, Italy [9, 10].

The above mentioned papers present simulation results for examples of particular flow-type landslides. CA algorithms were strictly adjusted to investigated processes considering topological and geomorphological conditions. In this paper the authors' main purpose is to create CA algorithm for landslide simulation in southern part of Polish mountains. The paper presents initial simulation results of sample general dynamic surface process. Simulation uses simple 2-dimensional cellular automata algorithm. The authors focused on the analysis of different neighbourhood types (Moore or von Neumann) and model resolution influence on the results of simulation. The other purpose of research was to visualise general surface process in most effective and realistic way. Modelling and visualisation were created by software developed with standard OpenGL graphic library.

2. Cellular automata

This paper considers one of the simplest examples of cellular automata — two dimensional cellular automata. 2d cellular automata consists of regular network n of single cells c. System evolves discrete in time t. Cellular automata behaviour is defined by initial configuration and updating rule. Each cell c may be in one of given states. The number of state s is a finite number and it is identical for each automata cells. State of each cell c is updated at next time step t. Cells c have the same rule for updating. States of all cells define the state of the whole system. Single cell behaviour is defined by strict rules. To define the rule one needs to determine neighbourhood type and boundary conditions. Cell neighbourhood consists of cells which have common edges or corners with given cell (Moore neighbourhood) or only edges (von Neumann neighbourhood). Particle is randomly added to each single cell in every time step. State of single cell in depends on states time step $t+1$ of neighbouring cells and state of the given cell in time step t.

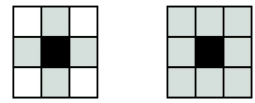

Fig. 1. Classical neighbourhood of 2-dimensional cellular automata.

If the number of particles does not exceed the critical value then the system is stable. When exceeding the critical value unstable behaviour begins. Unstable behaviour means that particles are redistributed to neighbouring cells to relax given cell's state. Redistributions relax some cells but may as well be the reason of next perturbation in other cells. Each perturbation may be small and end quickly or start an avalanche perturbation covering almost the whole system. A sample 2d cellular automata is shown in Fig. 2. The critical value in this example is 4. Figure 2 presents 3×3 grid with some initial configuration of elements (black dots). Particles added randomly to the grid are white dots. White squares on the grid represent unaffected area, grey squares are active during an avalanche (they redistribute or receive elements). The numbers above the grid are numbered time steps. Time steps where the avalanche begins are presented as numbers with letters and show sample sequence of single perturbations — in fact all they (4a and 4b) occur at the same time step, but it is better to separate each single perturbation to present it on the diagram. The numbers below the grid are numbers of elements added to the grid (on the left) and removed from the grid (on the right) at each perturbation (when the avalanche begins).

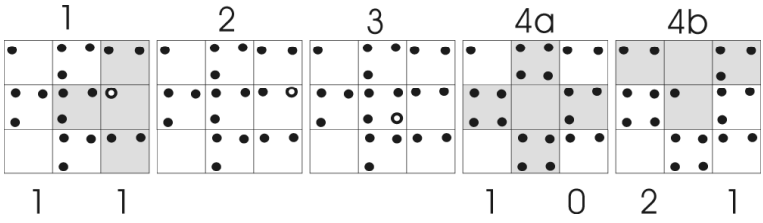

Fig. 2. 2-dimensional cellular automata iteration [11].

3. Simulation of dynamic surface processes

There has been a lot of experimental studies of real systems behaviour described by cellular automata, *e.g.* [2–4, 12]. This mathematical model is very useful for physics simulations (fires, earthquakes, dust spreading, explosions, liquid or granular flows, *etc.*) and biological or sociophysical ones (epidemies, crowd behaviour, *etc.*) [13]. Main cellular automata advantage is that very simple algorithms (basic level mathematical equations) give very complex and realistic results. This mathematical model is appropriate for simulating complex phenomena by splitting one large scale problem into many small elements — cells. This paper presents simulation of dynamic surface process. One of the examples of such a process is lavaflow (Fig. 4). Cellular automata have been used for natural flow simulations, both classical cellular automata and MCA (Macroscopic Cellular Automata), which introduce extension of classical CA.

3.1. Algorithm description

Simple cellular automata algorithm was created to simulate surface process movement. The sample terrain is represented by regular network of triangles with known height values. Terrain network is projected on cellular automata network. Each cell is described by two values — one representing the terrain height and the other representing chosen simulated process (slide, avalanche, flows) propagating. Network borders are barriers which prevent cells from propagating in wrong direction (border cells have large height values).

During simulation the state of each cell is described by the terrain height and the logical value representing the simulated process. Value may be true (there is given process) or false (no process). For each cell in each time step the transfer function gives the direction of given process propagation. Next state is calculated for each cell basing upon its current state, neighbouring cells state and the terrain height. The state of a cell in next time step (S_{t+1}) is described as the function:

$$S_{t+1} = f(S_t) = f(n, l, \Delta h)\,, \tag{1}$$

where: f — transition function, S_{t+1} — state of a cell in next time step, S_t — state of a cell in previous time step, n — number of neighbours, l — logical value of propagation, Δh — height difference between a cell and its neighbour.

Flow rule represents dynamical surface process with gravity driving force propagating from higher terrain to lower levels. Simulation was made for sample general dynamical surface process. For particular investigated process (slides, mud or snow avalanches, lava flows) one needs to consider additional parameters describing given model as for example density, viscosity, rock type, *etc.*

3.2. Implementation

Simulation has been made by the use of in-house developed software. The application has been written in C with OpenGL (Open Graphics Library). OpenGL graphic library is standard in computer graphics and scientific visualisation [4]. 'GL Utility Toolkit' library makes it possible to use easily the application for different platforms, for example Windows and Linux.

Created software may load any real terrain model from text file with XYZ coordinates or randomly generate sample terrain (using fractal algorithm). There is a possibility of controlling network size and changing process parameters. Chosen model may be scaled or rotated in any way during simulation what simplifies the observation of the process.

3.3. Simulation results

Three models have been used for the simulation: real models (Tatra mountains and African volcano Meru terrains) and sample fractal generated terrain. Network 500×500 was used in both examples. Simulation starting point of given process is chosen randomly. Sample fractal terrain results are presented in Fig. 3. Figure 4 shows randomly generated result for volcanic area.

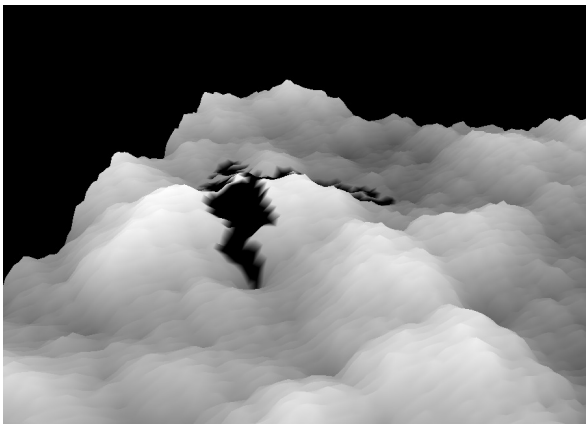

Fig. 3. Sample fractal generated terrain simulation.

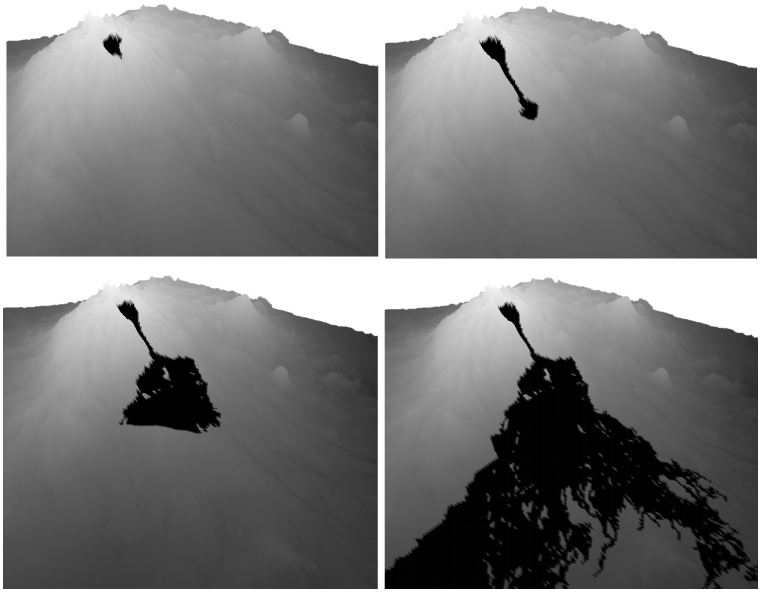

Fig. 4. Simulation steps — model of Meru volcano, Africa.

It has been tested that when using cellular automata for such simulation the results are different for different types of cells neighbourhood (4 or 8 adjacent cells). Figure 5 shows the simulation results for the same terrain

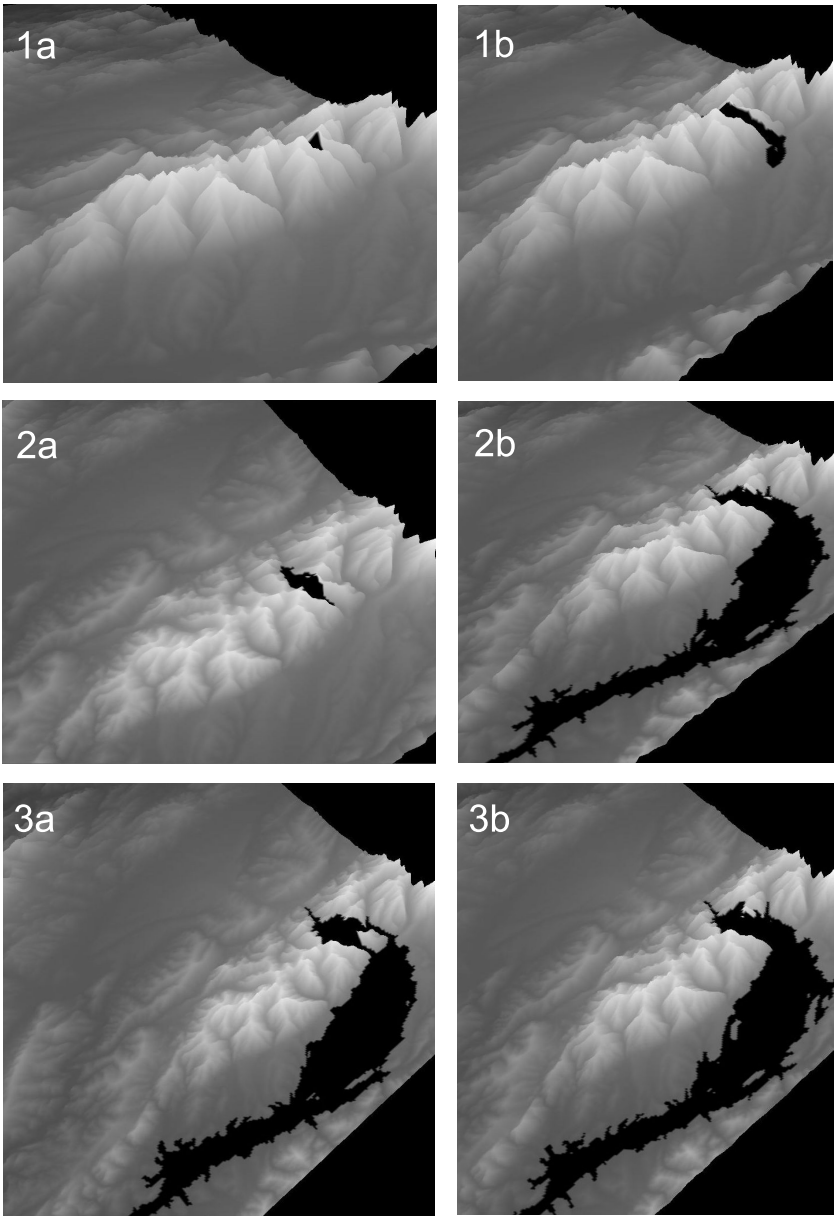

Fig. 5. Simulation steps for different neighbourhood type — Tatra mountains, Poland.

— Tatra mountains, the same number of iterations and different neighbourhood. Simulation steps are numbered 1a, 2a and 3a for von Neumann neighbourhood and 1b, 2b and 3b for Moore neighbourhood. Simulation time steps are 10 seconds (1a and 1b), 100 seconds (2a and 2b) and 1000 seconds (3a and 3b). Starting point is the same for both simulations, but results differ. Screenshots of corresponding simulations were taken at the same time step (for example 2a and 2b) to show differences. Significant difference of process spreading time has been observed for analysed neighbourhood types. Rangeland and area limits do not cover. The conclusion may be drawn after many simulations that with Moore's neighbourhood the simulation results give more realistic effects for topographically differentiated model.

Given results present dynamical surface processes quite good as realistic ones. The way of process propagating in relation to terrain deformations were shown in both examples. Software needs to be extended for simulation of specific process as for example slides, avalanches or flows. Considered additional parameters have huge influence on the simulation of given process. It is not enough to create simulation basing only on terrain features and gravity force. Extended application may be a helpful tool for monitoring and planning actions in natural disasters risk.

4. Summary

Computer simulations help us to foresee long-term causes of investigated processes. It is also a perfect tool for monitoring dynamical surface processes behaviour and planning adequate actions during natural disasters alerts (avalanches, slides, flows). Presented dynamical surface process model suits for simulations of local surface processes. Very high resolution for representing the terrain is needed to receive realistic results. Generating proper terrain model and extended network cellular automata processing are main problems of the simulation. Simulation gives different results for different neighbourhood type (Moore or von Neumann). With von Neumann neighbourhood simulation of the dynamical surface process is considerably slower than for von Neumann neighbourhood. From the optimising point of view it is better to choose the simplest one, but the type of neighbourhood needs to be chosen carefully as for the simulation validity. Simulation presented in this paper concerns general complex surface processes with the use of cellular automata algorithms. Effectiveness of the software has been shown and realistic results obtained. Present research allows to estimate the results approximately and more accurate analysis needs to be performed with comparison to real data. Currently the author works on developing algorithm with additional parameters for specific process.

This work was supported by project no.10.10.140.675 at the AGH University of Science and Technology, Faculty of Geology, Geophysics and Environmental Protection, Department of Geoinformatics and Applied Computer Science, Kraków, Poland.

REFERENCES

[1] T. Toffoli, *Physica D* **10**, 117 (1984).

[2] S. Di Gregorio, R. Serra, *Future Generation Computer Systems* **16**, 259 (1999).

[3] D. Barca, S. Di Gregorio, F.P. Nicoletta, M. Sorriso-Valvo, A cellular space model for flow type landslides, Computers and their application for development, Proc. Int. Symp. IASTED, Taormina, 30–32, 1986.

[4] E. Segre, C. Deangeli, *Nonl. Proc. Geophys.* **2**, 1 (1995).

[5] B.D. Malamud, D.L. Turcotte, *IEEE Computing in Science and Engineering* **2**, 42 (2000).

[6] A. Clerici, S. Perego, *Geomorphology* **33**, 1 (2000).

[7] S. Di Gregorio, R. Rongo, C. Siciliano, M. Sorriso-Valvo, W. Spataro, *Physics and Chemistry of the Earth* **24**, 97 (1999).

[8] M.V. Avolio, S. Di Gregorio, F. Mantovani, A. Pasuto, R. Rongo, S. Silvano, W. Spataro, *JAG* **2**, 41 (2000).

[9] D. D'Ambrosio, S. Di Gregorio, G. Iovine, V. Lupiano, L. Merenda, R. Rongo, W. Spataro, *Physics and Chemistry of the Earth* **27**, 1577 (2002).

[10] D. D'Ambrosio, S. Di Gregorio, G. Iovine, V. Lupiano, R. Rongo, W. Spataro, *Geomorphology* **54**, 91 (2003).

[11] D.L. Turcotte, *Fractals and Chaos in Geology and Geophysics*, Cambridge University Press, UK, 1997.

[12] D. D'Ambrosio, R. Rongo, W. Spataro, G.M. Crisci, S. Di Gregorio, G. Iovine, G. Niceforo, Lava flows simulation through cellular automata and genetic algorithms: calibration and validation for the Etnean area of study, Atti del the 6-th WSEAS International Conference on Systems Theory and Scientific Computation (ISTASC'06), Crete Island, Greece, August 18–20, 2006.

[13] T. Forsyth, *Cellular Automata for Physical Modelling, Game Programming Gems 3*, Charles River Media, USA, 2002.

[14] R.S. Wright Jr., B. Lipchak, *OpenGL Superbible, Third Edition*, Sams Publishing, USA, 2005.