CELLULAR AUTOMATA IN THERMODYNAMIC AND CLOUD DYNAMICS SIMULATIONS

Alisson Rodrigo da Silva[1] **and Maury Meirelles Gouvêa Jr.**[2]

Pontifical Catholic University of Minas Gerais, Brazil

alisson.sistemas@gmail.com(1)
maury@pucminas.br(2)

Abstract

In real system simulations, the application of cellular automata has been shown as an interesting option, because it can represent an emergent behavior and its implementation is simple. This paper presents a method for simulating thermodynamic systems, such as cloud dynamics, with cellular automata. In accordance with thermodynamic principles, this paper presents an isolated system model that describes temperature dynamics. The model uses the Von Neumann neighborhood of five cells, each with two possible states: the presence or absence of a cloud or a part of it. Our model uses three weather properties, as follows, condensed cloud water particles, temperature and outer winds. Two types of experiments were performed to validate the model proposed: one with a warm body in the center of the environment and another with a cloud.

Keywords: Cellular automata, Thermodynamic systems, Cloud dynamics, Simulation.

Presenting Author's Biography

Maury Meirelles Gouvêa Jr. teaches and undertakes research in the Institute of Exact Sciences and Informatics, Pontifical Catholic University of Minas Gerais. He obtained a Ph.D. degree in Artificial Intelligence from the Centre of Informatics, Federal University of Pernambuco, in 2009. His research interests include evolutionary algorithms, neural networks, and dynamic and adaptive systems.

1 Introduction

Cellular automata (CAs) [1] are discrete models on which many areas, such as computation, mathematic, physic, complexity science, and biology, are conducting research. CAs consist of a grid of cells, each with a finite number of states. For each cell, a set of cells, called neighborhood, is defined for the specified cell. At each iteration, a new state of each cell arises in accordance with the current state of the cell, the states of the cells in its neighborhood, and some fixed rules. Typically, the rule to update the state of a cell is the same for all cells and does not change over time, and is applied to the whole grid simultaneously.

In real system simulations, the application of cellular automata has been shown as an interesting option, because it can represent an emergent behavior and its implementation is simple. CAs consist of a n-dimensional grid of cells with the same behaviors described by a set of transition rules [2]. CAs use a defined number of neighbor cells that interact with each other, creating a local interaction and then a global behavior. These interactions reflect the system dynamics based on the transitional rules.

This paper presents a method for simulating thermodynamic systems, such as cloud dynamics, with cellular automata. A thermodynamic system is concerned with the flow and balance of energy. Three types of thermodynamic systems are distinguished depending on the types of interaction and energy exchange taking place between the system and its surrounding environment: (i) an isolated system is isolated in every way from its environment, and it does not exchange heat, work or matter with its environment; (ii) a closed system can exchange energy, heat and work, but not matter with its environment; and (iii) an open system exchanges energy and matter with its environment. A boundary that allows the exchange of matter is said to be permeable. In isolated systems, it is observed that as time passes, internal rearrangements decrease and stable conditions are reached. Properties, such as pressures and temperatures, tend to equalize, and matter arranges itself into one or a few homogeneous phases. A system in which all processes of change have ended is considered to be in a state of thermodynamic equilibrium.

In accordance with thermodynamic principles, this paper presents an isolated system model that describes temperature dynamics. This model is also used to simulate cloud dynamics. Two types of experiments were performed: one with a warm body in the center of the environment and another with a cloud. Clouds are formed from the condensation of water vapors present in the atmosphere. After it is formed, a cloud is moved by winds, and changes both its location and its properties, such as, temperature, pressure, density, and humidity. These properties strongly influence cloud dynamics. The model proposed uses the Von Neumann neighborhood of five cells, each with two possible states: the presence or absence of a cloud or a part of it. Our model uses three weather properties, as follows, condensed cloud water particles, temperature and outer winds. The transition rules are based on thermodynamic principles and weather concepts.

The rest of this paper is organized as follows. Section 2 introduces general concepts of cloud dynamics and cellular automata. Sections 3 and 4 describe the isolated thermodynamic system and the cloud dynamics model with cellular automata, respectively. Section 5 concludes the paper and suggests new implementations for future studies.

2 Cloud Dynamics and Cellular Automata

Clouds are formed from the condensation of water existing in the humid air in the atmosphere. The elevation of air is the key process in the production of clouds, because when it rises and comes into contact with low temperatures, cold air makes it possible for clouds to form. This elevation can be produced by convection, convergence of air streams, topographical elevation or frontal lifting [3].

Clouds may be in a liquid or solid state, or may be a mixed composition of water and ice. The composition of a cloud depends on its altitude. After having formed, clouds are moved by winds in all directions. When a cloud is moved in a vertical direction, its altitude changes as do its properties, such as temperature, pressure, kinetic energy, density and humidity. On rising, there is a cooling of condensed cloud water particles that may become partially or completely frozen. On the other hand, when a cloud goes to a lower altitude, it goes to a higher temperature environment; therefore, precipitation may arise and spread the cloud.

The dynamics, growth, motion and dissipation of clouds are complex. Thus, it is important to understand these dynamics in order to allow an efficient implementation of the real system [4]. The basic elements necessary to simulate clouds are velocity, air pressure, temperature, water vapor, and condensed cloud water. These water content variables are of mixed ratios, i.e., the mass of vapor or liquid water per unit mass of air. We consider a system of equations that models cloud dynamics in terms of velocity and condensed cloud water variables.

A cellular automaton is formally defined as a discrete mathematical model, implemented in computers, automated by deterministic rules, and its conduct of an element within a homogeneous set will be based both on the state of its own attributes and those of the attributes of the neighboring elements [5].

A CA is characterized by its cell space and its transition rule. The cell space is a lattice of N identical cells arranged in a d-dimensional grid, each with an identical pattern of local connections to other cells. When we consider the lattice is of finite length, boundary conditions are applied resulting in a circular lattice. A transition rule provides the next state for each cell, as a function of the configuration of its current neighborhood. At each step of time, every cell of the lattice updates its states according to this rule [6].

As to the CA-dimensional rule contained in each cell, it is essentially a finite state machine, usually specified in the form of a table of rules. These are called elementary cellular automata. The neighbors of a cell are adjacent cells, or cells on the right and left. Thus a cell is connected to air local neighbors (cells) where r is related to the radius, so that each cell has a neighborhood of $2r + 1$. A neighborhood is made up of three cells, so there are 23 = 8 possible patterns for a neighborhood. There are therefore 28 = 256 possible rules. Wolfram [2] proposed a numbering scheme for the elementary CAs, in which the output bits are ordered alphabetically, as in the transition rule, and are read from right to left to form a base number in decimal notation between 0 and 255.

For CAs, dimensional cells are arranged in a two-dimensional space (represented in the form of a grid), the neighborhoods most widely used are the Von Neumann neighborhood, consisting of 5 cells (central cell and 4 neighbors, up, down, left , right.) and the Moore neighborhood, consisting of 9 cells (the Von Neumann neighborhood of more cells in the diagonal.).

Cellular automata are used in simulation and emulation of real systems [1], such as:

- Simulation of bacterial or viral behavior, crystal growth, coral, rocks and other natural elements, behavior of gases, spread of fires, population development, economic, behavior of land, rivers and topographies, and forecast of plant growth;
- Video: generating random pictures, image filters and distortions;
- Music: melody-generating digital noise and sound;
- Mathematics: alternative to replacement differential equations;
- Computer: random number generation, cryptography, and conceptual design of parallel computations mass; and
- 3D animation: particle simulation and generating textures.

3 Thermodynamic Model with Cellular Automata

This section presents the model to describe an isolated system with cellular automata. In isolated systems, as the time passes, internal rearrangements decrease and stable conditions are reached. Properties, such as pressures and temperatures, tend to equalize, such that the processes of change come to an end and the system reaches the state of thermodynamic equilibrium.

The model proposed uses the Von Neuman neighborhood with five cells, Figure 1. The cell temperature behavior is derived from the thermodynamic principles in

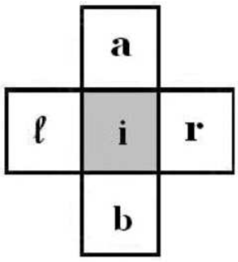


Fig. 1 Von Neumann Neighborhood

which heat transfer between neighborhood cells is provided. In each iteration, the whole grid is updated with new cell temperatures, as follows

$$T_i(k+1) = T_i(k) + \alpha\, \Delta T(k), \qquad (1)$$

where

$$\Delta T(k) = \frac{\sum_i T_i(k) - (N_o - 1)T_i(k)}{N_o}, \qquad (2)$$

for $i = l, r, a, b$, N_o is the number of cells per neighborhood, α is a constant that defines the step size of the temperature update, $T_i(k)$ is the temperature into i-th cell at iteration k, and the subscripts l, r, a, b mean the neighbors on the left, on the right, above, and below neighbors of an i-th cell.

Now, two simulations of isolated thermodynamic systems are presented in 30×30 and 50×50 grids. The model was implemented with a Web platform, using C-Sharp language at ASP.net framework. For these simulations, the temperature interval was [0, 50] degrees, the initial temperature was 0^oC, and $\alpha = 0.123456$. Figure 2 shows the temperature intervals and their respective colors.

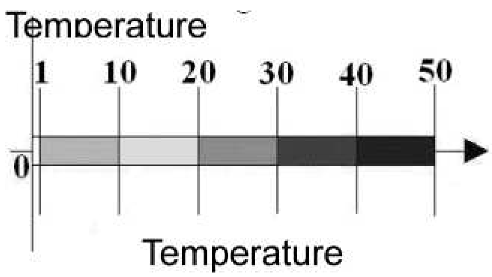


Fig. 2 Legend of colors used in isolated thermodynamic system model.

Figure 3 shows six iterations of each simulation. In the simulation of the 30×30 grid, Figure 3(a), a warm body of 50^oC and 50%-grid area was initialized into the center of the grid. The heat of the warm body spread quickly – see iteration 4. In iteration 27, the extreme area of the grid had been warmed, by heat from the warm body, to a temperature of between 1 and 10 ^{o}C.

Iteration 48 shows that the center of the warm body started to cool, because its heat spread throughout the grid. Iterations 112 and 369 show that the heat continued to spread, thus providing the temperature equalization. Iteration 472 shows the moment at which the grid temperature was totally equalized between 1 and 10 ^{o}C.

In the simulation of the 50×50 grid, Figure 3(b), a warm body of 50^oC and 80%-grid area was initialized into the center of the grid. The system behavior was similar to that of the first simulation, 30×30 grid. The heat spread quickly throughout the whole grid. After iteration 46, the heat of the center of the grid started to decrease. Iterations 112 and 134 show the beginning of the temperature equalization throughout the whole grid. Iteration 485 shows the grid with its temperature totally equalized between 20 and 30 ^{o}C, higher than that of the first simulation because of the larger area of the warm body (80%-grid area).

These simulations showed that a cellular automaton model can simulate a thermodynamic system. In both of them, the heat in the center of grid spread throughout the whole grid until the thermodynamic equilibrium.

4 Cloud Dynamics with Cellular Automata

The model proposed simulates only one cloud into the grid. The cloud has an initial size that may be modified by weather events. All initial simulation parameters are set at random. The model has so called inner variables and outer events. The inner variables are the cell properties, while the outer events involve the whole grid. The inner variables are the number of condensed cloud water particles and temperature. The outer events are to do with the insertion of winds and condensed cloud water insertion into the cells, both with random properties. The following sections explain these parameters and events.

4.1 Inner Variable

The inner variables are the cell properties, as follows, the number of condensed cloud water particles, n, and the temperature, T. For each iteration, a thermodynamic law is used in each cell in order to describe its dynamics. The cell temperature behavior is set as described in Section 3.

Each cell has a cloud, or part of it, if the number of condensed cloud water particles is equal or larger than a threshold; otherwise, there is no cloud in the cell. The number of condensed cloud water particles into an i-th cell at iteration k is defined as a function of its current temperature, $T_i(k)$, as follows

$$A = \begin{cases} 0, & \text{if } T_i(k) > T_C \\ n_{\min}, & \text{if } T_i(k) = T_C \\ n_{\min}\,(T_C - T_i(k)), & \text{if } T_i(k) < T_C \end{cases}, \quad (3)$$

where T_C is the temperature of water condensation at current atmospheric conditions and $n_{\min}$ is a threshold, that is, the number of condensed cloud water particles that defines the presence of a cloud in a cell.

4.2 Outer Events

The outer events occur on the whole grid. These random events are the outer winds and the insertion of condensed cloud water, that occur as a probability p_w and p_i, respectively.

The outer winds have different directions, widths, and magnitudes. There are eight directions:

- north-south, and vice-versa;
- east-west, and vice-versa;
- southeast- northwest, and vice-versa; or
- southwest-northeast, and vice-versa.

The minimum and maximum wind widths are $w_{\min}$ and $w_{\max}$. The wind magnitude means the strength over a cloud. With a wind occurrence over a cell, its condensed cloud water are displaced in accordance to the wind magnitude and direction. The condensed cloud water displacement is computed as follows

$$n_i(k+1) = n_i(k)\,(1 - I), \quad (4)$$

where $I \in [0, 1]$ is the wind magnitude. The difference between $n_i(k+1)$ and $n_i(k)$ is the number of condensed cloud water particles that was displaced from i-th cell to a neighborhood cell. These displaced condensed cloud waters go to a neighborhood in accordance with the wind direction. For instance, if a wind with direction west-east cover the l-cell, Figure 1, the condensed cloud waters displaced will go to the i-cell.

In order to represent changes in the humidity of the weather, we implemented a condensed cloud waters insertion as a random event into the cells. The insertion of condensed cloud waters into an i-th cell is defined as follows

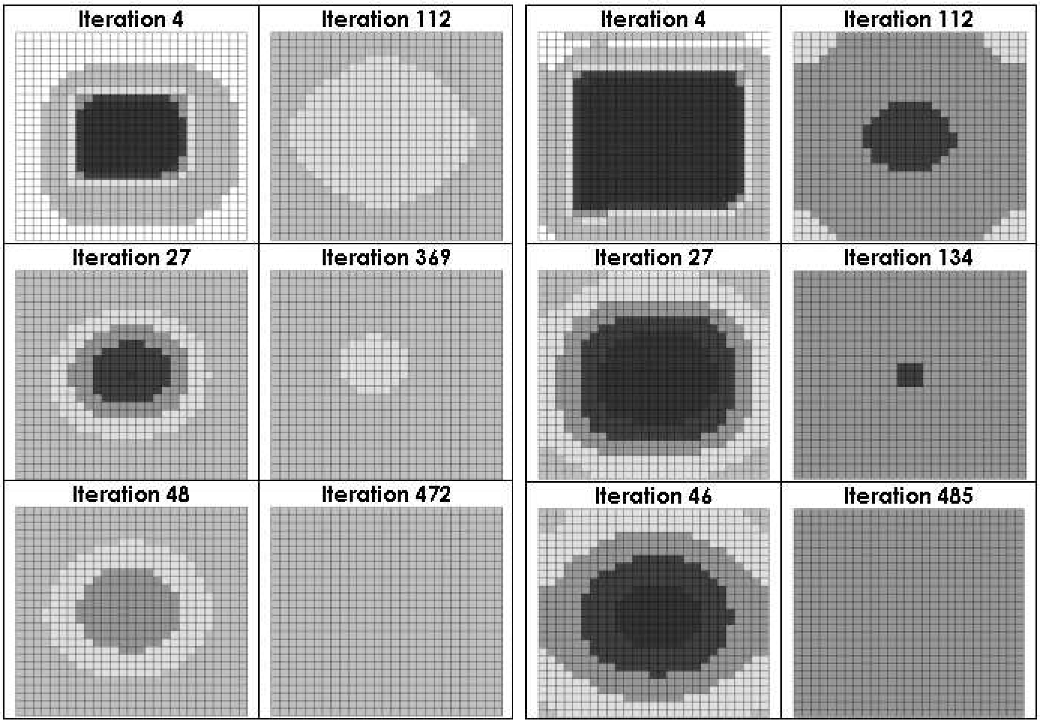
$$n_i(k+1) = n_i(k) + \beta\,\Delta n \quad (5)$$

where Δn is the maximum number of condensed cloud waters particles that can be inserted into a cell and $\beta \in [-1, 1]$ is a random variable that defines the number of condensed cloud waters particles to be inserted into a cell.

4.3 The Model Dynamics

The model dynamics is based on a discrete and iterative system. The temperature is started with 0^oC and a cloud with temperature and condensed cloud waters are chosen at random. All grid cell transitions are based on Equations 1-4 that change the cell states, providing the global effect in the grid. The pseudo-code of our model is presented as follows:

1. Initialize $[T_{\min}, T_{\max}]$, T_C, $[n_{\min}, n_{\max}]$, where $n_{\max} = n_{\min}(T_C - T_{\min})$, and the temperature,



(a) Simulation of the 30x30 grid   (b) Simulation of the 50x50 grid

Fig. 3 Isolated thermodynamic system simulations.

T_A, where $T_A > T_C$ is the environmental temperature;

2. Initialize the grid cells with $T = T_A$ and $n = 0$;
3. Create a cloud into the grid at random, with $T_i(k) < T_C$ and n_i defined by Equation 1;
4. Update cell temperatures by Equation 1;
5. Update number of condensed cloud water particles for each cell by Equation 3;
6. If $x < p_w$, where $x \in [0, 1]$ is a random number, then
   - Initialize the wind direction, I, $[w_{\min}, w_{\max}]$, at random;
   - Apply the wind on the grid, and use Equation 4 when it cover a cell with $n_i > 0$;
7. If $x < p_i$, then insert condensed cloud water on the all grid cells by Equation 5;
8. Go back to step 4 while a stop criterion is not satisfied.

4.4 Simulations and Result Analyses

This subsection presents two simulations with 30×30 and 50×50 grids. The temperature interval was [-1, -50] degrees. Figure 4 shows the temperature intervals and their respective colors. In these simulations, $T_A =$ 0^oC, $T_C = -10^oC$, $n_{\min} = 100$, $n_{\max} = 400$, and $\alpha = 0.123456$. Three grid sizes were used.

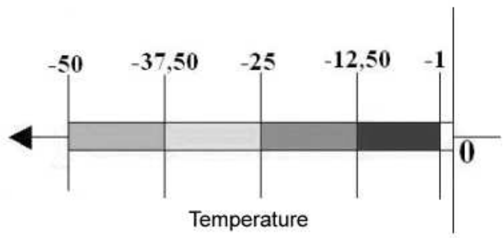


Fig. 4 Legend of colors used in the cloud dynamics simulations.

In the first simulation, grid 30×30, the minimum, $w_{\min}$, and maximum, $w_{\max}$, wind widths were 4 and 7, respectively. This simulation, Figure 5, showed fast change of the cloud behavior with respect to its area and temperature. Figure 5(a) shows the initial cloud state, at iteration 3. Figure 5(b) shows the iteration 88, where the thermal equilibrium just began. In Figure 5(c), iteration 118, the cloud dissipation is starting and it is finished after iteration 208, Figure 5(e). The cloud dissipated completely at iteration 220, and Figure 5(f) shows the last iteration 226.

In the second simulation, grid 50×50, $w_{\min} = 8$ and $w_{\max} = 16$. This simulation showed a similar behavior to that of the first simulation, with thermal equilibrium

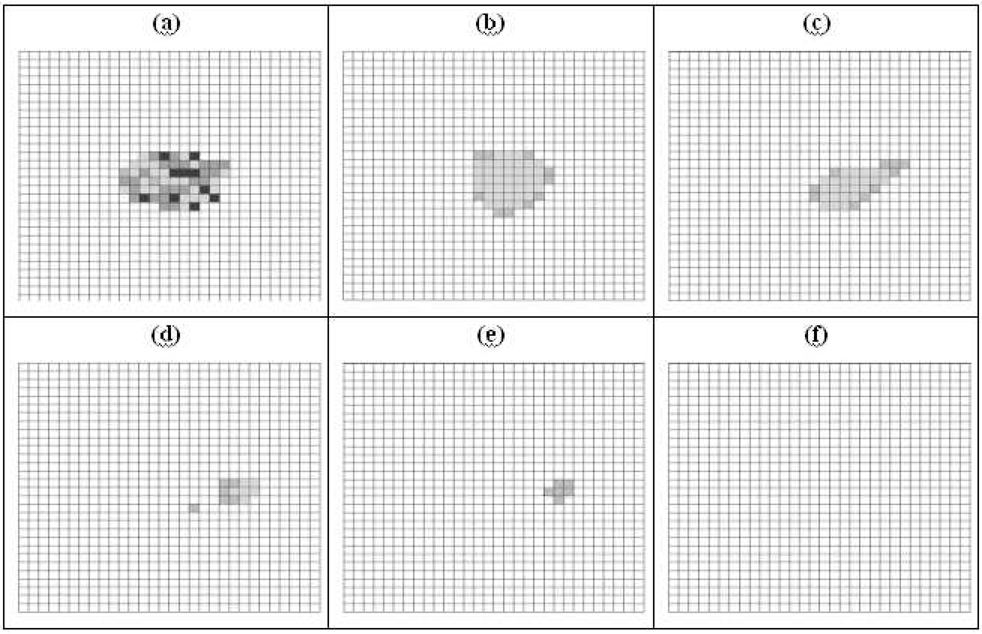


Fig. 5 Six states of the second simulation, grid 30×30.

throughout the grid and dissipation of the cloud by the wind actions. Due to the larger grid size with respect to the previous simulation, the cloud reaches the thermal equilibrium in a larger number of simulations.

Figure 6(a) shows the grid at iteration 3 in which it is possible to see the different temperature in the cloud, represented by the colors, due to the random initialization. Figure 6(b) shows the grid at iteration 118, where a thermal equilibrium has begun. In iteration 218, Figure 6(c), the cloud showed a area smaller than its initial state, Figure 6(a), due to the thermal equilibrium with the whole grid and wind actions. In iteration 318, Figure 6(e), the whole cloud converges to the thermal equilibrium and it is almost dissipated. The full dissipation occurred at iteration 352. Figure 6(f) shows the last iteration (356), where the whole grid converged to the thermal equilibrium and the cloud was dissipated.

The cloud dynamics model showed an expected behavior regarding some thermodynamics concepts, because in all experiments the whole grid reached thermal equilibrium resulting in total cloud dissipation to which the actions of wind also contributed. In a grid where a given temperature prevails over almost of its whole area, the environmental temperature set to 0^oC, is expected at infinite time into an undisturbed environment such that it reaches thermal equilibrium to the environmental temperature. These effects resulting from physical phenomena were more prevalent than those of the simulations with smaller grids. These results may be expected. Because of these smaller areas, thermodynamic equilibrium tends to be reached faster than in those of the larger grids, in addition to which the probability of a wind reaching the cloud is greater.

5 Conclusion

This article set out to construct a model to simulate thermodynamic systems using cellular automata. Two types of models were presented, an isolated thermodynamic system and a cloud dynamics model. The former showed that a cellular automaton can simulate a thermodynamic system. This first model was the basis for the second one, the cloud dynamics model.

The second model used a limited representation considering the variables that represent the dynamics of a real cloud. The number of condensed cloud waters particles per cell and the temperature and external winds in the two-dimensional grid were included. A two-dimensional CA with a Von Neumann neighborhood of 5 cells was used. The transition rules were defined based on the thermodynamic principle that defines the thermal equilibrium.

The validation of the second model was made in simulations with different grid sizes and parameters. In this preliminary study, we did not compare our model with other ones for cloud dynamics. Thus, we only conducted a visual validation, considering basic thermodynamic principles. The simulations showed that our proposed model presented a satisfactory behavior, considering some thermodynamic principles. The wind actions also were considered coherent, because they moved the clouds until their full dissipation. In all experiments, the clouds, with heterogeneous temperature randomly initialized, tended to converge to a uniform temperature, reaching the thermal equilibrium. Another observed behavior was the thermal equilibrium between the cloud and grid, which always resulted in cloud dissipation. The wind actions also contributed to convergence of cloud temperature to environmental temperature, because they spread the clouds, accelerating the

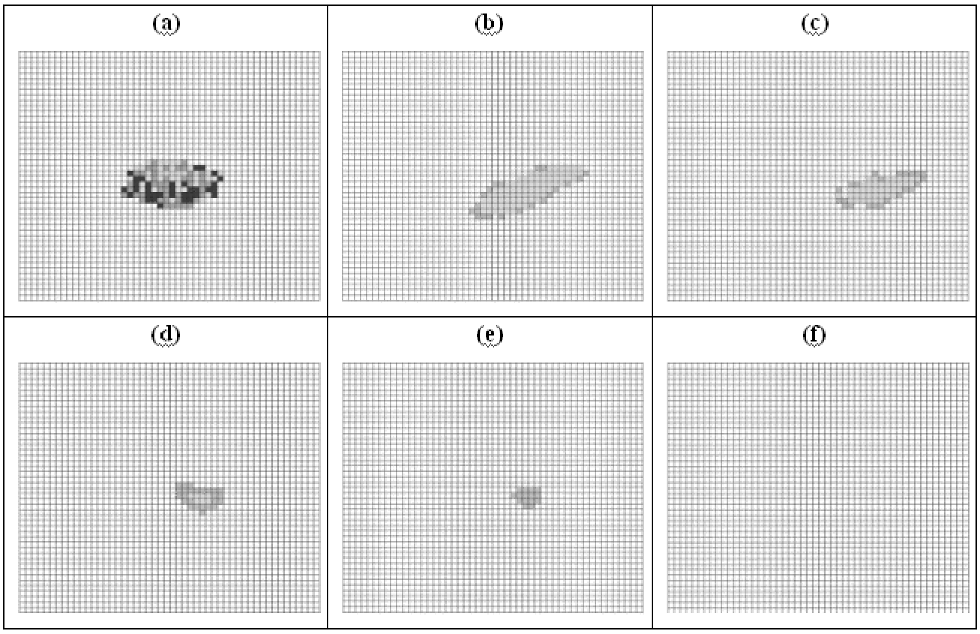


Fig. 6 Six states of the third simulation, grid 50×50.

thermal equilibrium. In all simulations, the clouds obtained a similar behaviors; in grid 50×50, the third simulation, the cloud remained in the grid longer than those in the other simulations.

As a follow-up to this study, other variables will be added, such as, pressure, kinetic energy, density and humidity, thus making the model more reliable. Another proposal is to simulate a three-dimensional space, approximating the model of a real system. The proposed model also may be implemented using the parallel computing paradigm, improving its performance and, consequently, its ability to perform in real time. The latter proposal is justified by the increase of variables involved, which feature a real atmospheric system. Thus, parallel computing may increase the model performance in a more complex scenario.

6 Acknowledgment

The authors gratefully acknowledge the assistance provided by the Center of Climatology of the Pontifical Catholic University of Minas Gerais (PUC Minas), especially its Coordinator, Dra. Adma Raia Silva, who supported us in building the model presented in this paper.

7 References

[1] Jean-Philippe Rennard. *Artificial Life: Where Biology Meets Computer Science*. FrontMatter & Associates Literary Agency, 2002.

[2] S. Wolfram. *A New Kind of Science*. Wolfram Media, Champaign, 2002.

[3] R. L. Vianello. *Meteorologia básica e aplicações*. Editora UFV, Viçosa, 2000.

[4] D. G. Andrews. *An Introduction to Atmospheric Physics*. Cambridge University Press, 2000.

[5] A. W. Burks. *Essays on Cellular Automata*. University of Illinois Press, Urbana, 1970.

[6] G. M. B. Oliveira. Autômatos celulares: aspectos dinâmicos e computacionais. In *III Jornada de Mini-cursos em Inteligência Artificial (MCIA). Sociedade Brasileira de Computação*, volume 8, 2003.