Chapter 8
Cellular Automata in Ecological Modelling

Broder Breckling, Guy Pe'er, and Yiannis G. Matsinos

The chessboard is the world;
the pieces are the phenomena of the universe;
the rules of the games are what we call the laws of Nature.

T. H. Huxley (1870)

Abstract Cellular Automata (CA) are models that generate large-scale pattern from small-scale local processes. CA deal with spatially extended dynamics using a grid structure. Successive states of cells, which are arranged on a grid, are calculated according to a set of rules. State transitions depend on the state of the single cells and the state of the cells in the local neighbourhood. Cellular Automata are applied as a modelling approach in many scientific disciplines and are used in ecology as one of the most popular model types to study spatially extended dynamics. The chapter starts with a brief historical overview about CA. It describes how CA function, and for which types of problems they can be employed. We present simple theoretical examples, followed by a more detailed case study from plant competition and grassland community dynamics. As an outlook, we discuss major fields of application with a special focus on the ecological context. Finally, we provide a brief overview and recommendations on the use of some of the software specialized in the field of CA modelling.

8.1 Introduction and Historical Background

Cellular Automata were conceptually developed by the Austro-Hungarian mathematician John von Neumann (1903–1957) during the 1950s. He was interested in simulating self-reproducing patterns. Instead of continuous approximations, he

B. Breckling (✉)
General and Theoretical Ecology, University of Bremen, Leobener Str., 28359 Bremen, Germany
e-mail: broder@uni-bremen.de

F. Jopp et al. (eds.), *Modelling Complex Ecological Dynamics*,
DOI 10.1007/978-3-642-05029-9_8,

used discrete (stepwise) representations of space and time. Together with Stanislaw Ulam, the work was developed at the Los Alamos Laboratory, where both were also involved in the Manhattan Project.[1] Ulam used the idea to study crystallization processes on a two-dimensional grid (or lattice). The first CA model that made the approach widely known dated from the 1960s, when the Cambridge-based mathematician John Conway developed the "Game of Life" (Gardner 1970). This is a simple grid based process where cells can switch between two states following simple rules (Sect. 8.3). Because of its simplicity and surprisingly interesting and complex behaviour, the Game of Life created a lasting enthusiasm.

During the 1970s, a series of applications of cellular automata models were developed in physics to study gas and liquid diffusion, crystallization processes, magnetic and spin phenomena (Forrester et al. 2007). CA were further used as stepwise (discrete) approximation models for partial differential equations (see Chap. 7).

During the 1980s, following a marked increase of computer availability and computation power, the application of cellular automata has seen a significant increase, especially in mathematics and physics. Scientists started to realize that a discrete representation of systems could provide simpler and more efficient approximations of spatially complex processes compared to continuous approximations. It was then that cellular automata machines were constructed, in order to handle parallel processing more efficiently (Toffoli and Margolus 1987). An important contribution to CA was made by Wolfram (1994), who systematically explored the overall dynamics of large classes of one-dimensional cellular automata using the software "Mathematica", which he developed initially for this purpose. Wolfram showed that simple, deterministic rules can generate complex patterns in space or time that look as if they were completely random. In ecology, CA successively became one of the most frequently used approaches to model spatially extended processes. Often, they are used in combination with other techniques such as individual-based models (Chap. 12).

Due to their ease of implementation and capacity to simulate spatial patterns, CAs have been widely applied to ecological problems related to spatial processes, such as epidemic propagation (Sirakoulis et al. 2000), plant population dynamics (e.g. Iwasa et al. 1991; Pascual et al. 2002), post-disturbance resilience (Matsinos and Troumbis 2002), colonization processes (Silvertown et al. 1992; Hobbs and Hobbs 1987), land-use and land-cover change (White et al. 1997) and spatial competition of corals (Langmead and Sheppard 2004, see also Chap. 17). Rietkerk et al. (2004) used a simple cellular automaton model based on the model of Thiery et al. (1995) in order to understand how scale-dependent feedback can explain a diversity of spatial patterns in self organizing savannah ecosystems. Moustakas et al. (2006) developed a CA to analyse the interaction between fish schools and fleets of fishing vessels, in order to assess the efficacy of conservation measures.

[1]The Manhattan project covered the initial initiatives in the USA to develop nuclear weapons of mass destruction. The leading physicists worked for this project during the Second World War.

These models refer to basic biological processes: dispersal and competition. It is the variation in the strength and scale of feedbacks between cells in the automaton that influence the outcomes in terms of structure and scale of patchiness. This illustrates the general nature of scale-dependent processes underlying self-organized patchiness in ecosystems.

8.2 Cellular Automata: The Components

Though cellular automata can handle very complex spatial situations and quite difficult rule systems, the conceptual basis is quite simple, easy to understand and applicable with almost any conventional or object-oriented programming language.

A cellular automaton consists of a large number of cells, which are connected to a grid and can change their state individually. For all cells, a neighbourhood is defined that constitutes the surrounding area that influences the state transitions of each particular cell. Finally, there is a set of rules defining how each of the potential states of a cell and the states of the neighbourhood will determine the transition between cell states.

The Cells

Cellular automata models use cells as the units of operation. Cells can be considered as a storage space, with a defined number of state variables that can either be discrete or continuous. The most simplistic CAs consist of cells that can switch between two different states (binary), to be represented e.g. by black and white, on and off, dead or alive, etc. But it is also possible to have a cell's state being characterized by a larger number of variables. For example, when modelling soil processes using a CA, the cell could represent a square meter of the ground and have storage space for variables such as water content, organic material, temperature, etc.

The Grid

In a CA, each cell is surrounded by other neighbouring cells. The grid can be visualized by drawing the cells as nodes and the connection to adjacent cells as edges. A grid can be finite or infinite, (for simulation purposes only finite) and can have different topologies. For instance, cells along a line with one neighbour to the right and one to the left would represent a one-dimensional grid, cells with four neighbours (North, South, East and West) would represent a two-dimensional grid, etc. In principle, any topological structure would be possible (Fig. 8.1).

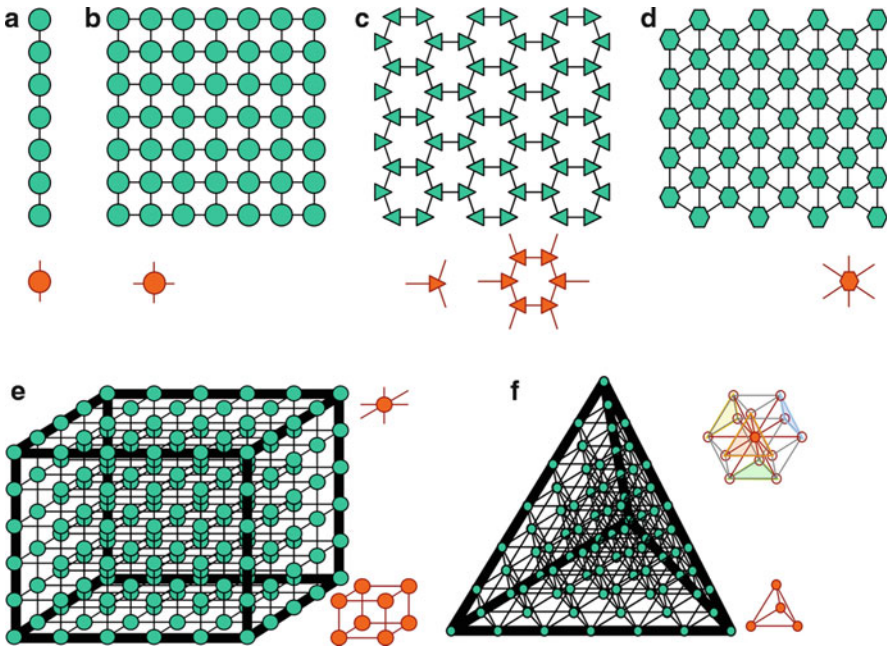

Fig. 8.1 Examples of one-, two- and three-dimensional grids of different topologies. (**a**) Linear grid: each cell has two neighbours; (**b**) 2D rectangular grid: each cell has four neighbours; (**c**) 2D triangular grid: each cell is connected to three neighbours; (**d**) 2D hexagonal grid: each cell has six neighbours; (**e**) 3D cubic grid: each cell inside the grid has six neighbours; (**f**) 3D tetrahedral grid: A cubic grid is not the only possibility to model in three dimensions. An alternative could consist of cells connected at the edges of stapled tetrahedrons. Please keep in mind that the given neighbourhood relations do not apply for margin cells

The Neighbourhood

The neighbourhood comprises the cells in the surrounding of a focal cell. The neighbourhood cells are defined as those that can influence the state of the particular focal cell. To determine which change occurs, the state of the focal cell and the states of the neighbourhood cells are evaluated. Usually, the neighbourhood consists of the directly adjacent cells, but the neighbourhood can have different extents and can vary in shape between rectangular, circular, etc. Other definitions are possible as well, e.g. that each cell selects a random number of other cells as neighbours – regardless where they are located on the grid. In case of a rectangular two-dimensional grid (Fig. 8.1b), the most commonly used neighbourhood comprises the four direct neighbours (Fig. 8.2b). In the CA terminology, this is also called von Neumann neighbourhood. If the eight directly adjacent cells are considered as neighbours, it is called Moore neighbourhood (see Fig. 8.2c), named after the US-American mathematician Edward F. Moore (1925–2003).

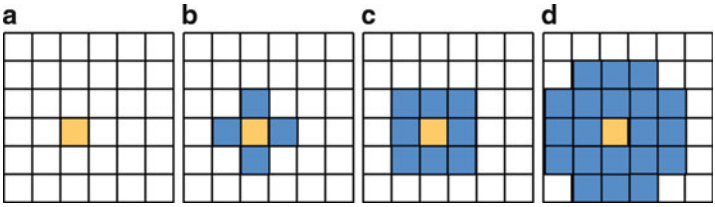

Fig. 8.2 Neighbourhoods in cellular automata: (**a**) focal cell; (**b**) von Neumann neighbourhood comprising the four adjacent cells – North, South, East, West; (**c**) Moore neighbourhood considering eight directly adjacent cells; and (**d**) a neighbourhood which consists of the nearest and second-nearest cells. Other definitions of neighbourhoods are also possible

The Rules

Rules of the CA are fundamental to specify how cells change their states. There can be an arbitrary number of rules. The rule-set for a CA applies to all cells. The current state of a particular cell and the states of the cells in the neighbourhood determine which of the rules are applied to change the cells state. The rules must consider all possible combinations of situations which can occur in the neighbourhood of a cell. A very simple example of a rule for a CA would be that the state of a cell can be any natural number, and that the subsequent state of a cell in the next step is the sum of the states of the neighbouring cells. This rule would constitute a deterministic CA. It would, however, also be possible to add stochasticity, e.g. by determining that for a given probability the state of the focal cell is zero.

Running a CA: The Iteration

A CA is processed step by step. One step (one iteration) comprises an application of the rules to all cells on the grid. To process a CA, the initial state of all cells must be set. This initial configuration is used for the first update. Cell by cell the rules are applied, taking into account the state of each cell and the state of cells in the neighbourhood. This yields the next state of all the cells. To avoid a bias of the update procedure, the new state of each cell is saved in a separate interim grid, so that the transition is applied only once after all the states (or transitions) of all cells have been calculated. Then the iteration can be repeated until a termination condition is met. The termination condition can be a maximum number of iterations, a pre-defined state of the grid, or an interruption by the user.

Boundary Conditions

Practical applications cannot work with infinite grids. The grid has to be spatially limited and a specification is required on how to process the cells at the boundary,

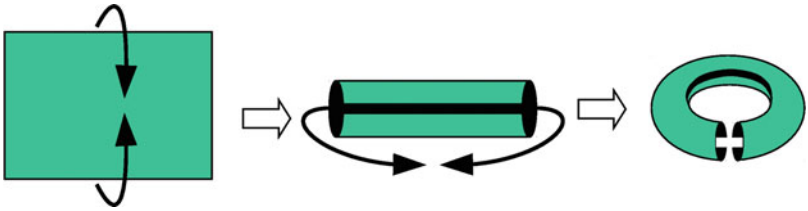

Fig. 8.3 Applying torus boundary condition in a rectangular grid by connecting opposite edges

where their neighbourhoods can be different from those situated in the inside of the grid. There are alternative ways in which boundary conditions can be specified. To this end the following solutions are frequently taken:

- Setting a different neighbourhood at the boundaries, taking into consideration that cells at the boundary have a different number of neighbours compared to the other cells and therefore require an according adaptation of the rule-set.
- The grid can be framed by a number of outer cells that maintain a particular state without being updated.
- In case of a rectangular grid, boundary cells can take the cells of the opposite boundary as their neighbours (i.e. the Eastern edge of a grid connects to the Western, the Northern connects to the Southern edge). Topologically this yields a torus (a doughnut like shape), as shown in Fig. 8.3.
- Grid extension: in the case of a homogeneous background state the grid could be dynamically extended. This solution, however, is possible only to the limits of processing capacity.

8.3 An Easy Example: Conway's Game of Life

Conway's Game of Life (Gardner 1970) is an excellent example to familiarize with the concept of CAs and with the process of updating the grid cells. Since the rules are rather simple, it is even possible to solve smaller grid iterations on paper. In more complex models, this process can of course be done only by a computer.

The Game of Life CA uses a two-dimensional rectangular grid. The cells can have two states, either "alive" (black) or "dead" (white). The state they take in a succeeding iteration (the rule set) depends on their own state and the states of their eight adjacent neighbours (Moore neighbourhood):

- A white cell becomes black (alive) if exactly three cells in its neighbourhood are black.
- A black cell remains black if two or three neighbours are black.

- A black cell turns to white, if less than two neighbourhood cells are black (it "dies of solitude") or if more than three neighbourhood cells are black (it "dies of overcrowdedness").
- The game can start with any initial configuration of black and white cells. Depending on the initial configuration, different patterns emerge. There are configurations which lead to global expansion. Others end in stationary or in repetitive pattern which re-emerge after a number of iterations. Other initial configurations can lead to "extinction", with only white cells remaining (Fig. 8.4).

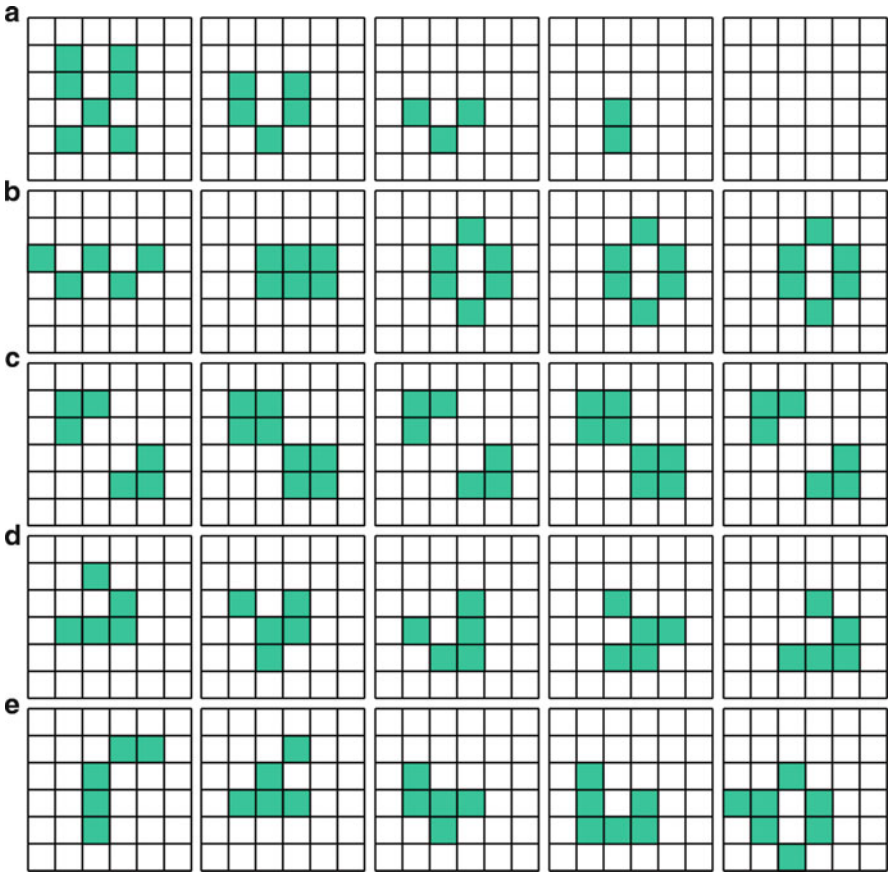

Fig. 8.4 Some pattern types which can occur in the game of life. (**a**) An initial configuration which "dies out" after four iterations; (**b**) an initial configuration which generates a stationary pattern; (**c**) an initial configuration which generates an oscillating pattern (re-emerging after a number of iterations); (**d**) a pattern that re-emerges but shifts location with time ("glider"); (**e**) an initial configuration generating a complex irregular pattern with parts that die out and others that oscillate or remain stationary. This is the so-called r-pentomino, which grows a large, irregular pattern taking more than 1,000 iterations before it becomes stationary and/or periodic

8.4 Examples of Pattern Generating Mechanisms

The Game of Life is a deterministic CA. Other applications may also include stochastic rules. There are certain types of interactions that can be found in different contexts that give rise to a specific category of pattern. We present three examples: A self-scaling random pattern, a spiral wave pattern, and a diffusion-limited aggregation.

Self-Scaling Random Pattern

A simple mechanism to generate a macroscopic pattern starting with a random configuration is to let each cell adapt to the state of the majority of its neighbours. Successive updates lead to growing homogeneous patches. The process can end up self-stabilizing. In case the boundaries of the resulting patches shift randomly, there is the possibility that such a system would finally end with a homogeneous grid with the same state for all cells. There are a few applications in a biological context, e.g. in tissue differentiation (Nijhout et al. 1986; Rasmussen et al. 1990) (Fig. 8.5).

Spiral Wave Pattern

This type of pattern is quite important for some self-organization processes in biology. It can be found also under the technical term of "excitable media". The term refers to the tendency of the CA, when applied to a two-dimensional grid, to form patterns of waves of excitation which move across the grid in an undamped manner. Though the pattern can be quite complex, the underlying mechanism is relatively simple. Each cell can have one of three states. The first one is called "excitable". Being excitable, a cell remains in this state as long as there is no stimulus from the neighbourhood. The second state is called "excited". When being in this state, each "excitable" cell in the neighbourhood of the "excited" cell will shift to the state "excited" as well. An excited cell remains for a certain number of iterations in this state, and then transits to a state called "refractory". Being in a refractory state, a cell cannot be excited again regardless of the neighbourhood states. After some iterations it returns to the state "excitable". The numbers of iterations which can be specified for the phases of "excited" and "refractory" influence the shape of the emerging macroscopic pattern.

A grid started with only "excitable" cells would remain as it is. A grid started with only "excitable" and "refractory" cells would end up in an overall state of being excitable as well. However, when there are a few excited cells, a spreading pattern can occur, which can organize the rest of the grid spatio-temporally. For random initial configurations of a sufficiently large grid, spiral wave patterns frequently occur

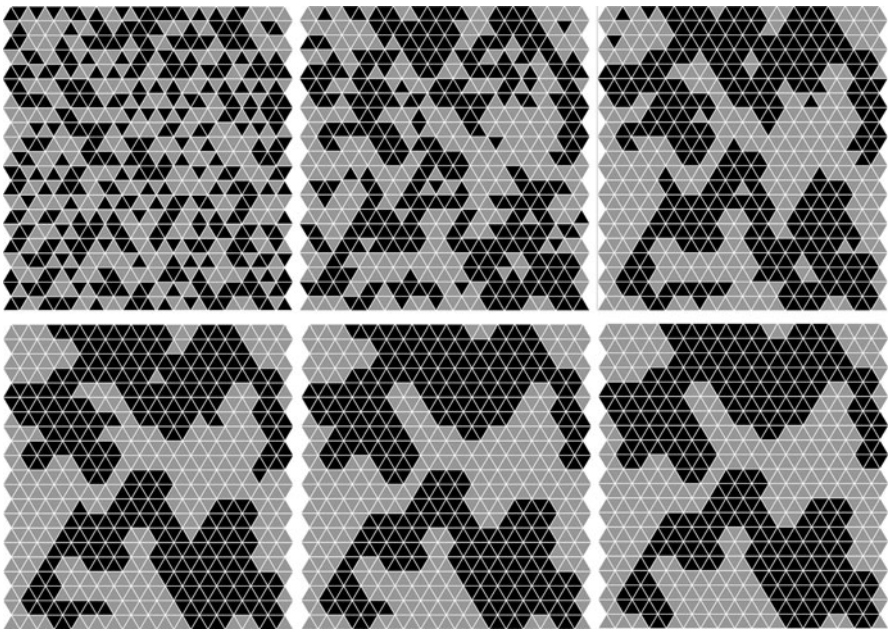

Fig. 8.5 The pattern on a triangular grid (three neighbours to each cell) was obtained by applying the following rules: The initial configuration is a random distribution. If the state of the majority of the neighbours differs from the focal cell's state, it has a 50% probability to change to the state of its neighbours. After a number of iterations, the pattern reaches a stable state, where each cell's state conforms to the majority of its neighbours. Shown are the iterations 0, 2, 5, 8, 11, and 14

(Fig. 8.6). Comins et al. (1992) used a cellular automaton employing deterministic rules to explore the spatial dynamics of a host–parasitoid interaction resulting in spiralling spatial patterns.

Diffusion-Limited Aggregation

Diffusion-limited aggregation is a process that can be observed in the successive growth of river systems, in certain forms of organic growth, involving branching, and in some inorganic immobilization processes. Again, the basic underlying rules are relatively simple. Cells can be in three types of states, which can be called "empty", "mobile" and "fixed". A "mobile" cell shifts the state of any (randomly chosen) "empty" neighbouring cell into "mobile", while turning back to an "empty" state. This simulates random movement of a particle across the grid. If a cell is in the state "mobile" and has a cell with the state "fixed" in its neighbourhood, the "mobile" cell changes its state to "fixed" and remains in this state for the rest of the simulation, regardless of the states of neighbouring cells. To obtain non-trivial results, it is required that a sufficiently large number of "mobile" cells

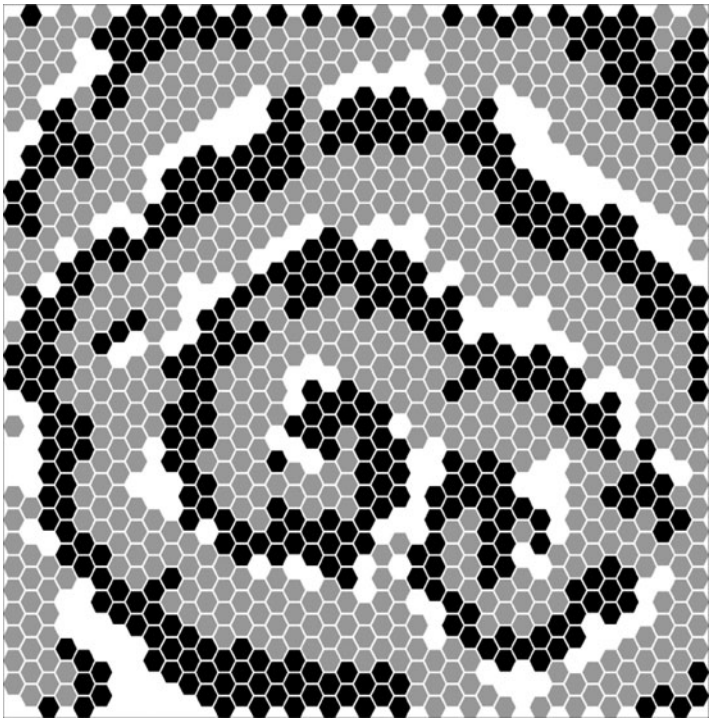

Fig. 8.6 A spiral wave pattern on a hexagonal grid; excitable cells (*white*), excited cells (*black*), refractory cells (*grey*); see text for details of state transitions

are started, and that at least one "fixed" cell exists on the grid. The emerging pattern is a random structure with a self-similar pattern (Fig. 8.7). Examples for the application of diffusion-limited aggregation models are CAs for predicting urban development (White and Engelen 1993), investigating the spatial distribution of plants and animals (which often seem to form fractal patterns; Kendal 1995), and studying pattern formation, e.g. in plant morphogenesis (Fleury 1999) (Fig. 8.7).

8.5 Case Study: Competition and Dispersal in Grassland Communities

As an example of a more complicated, recent model applied for the study of ecological questions, we elaborate on a CA model of a grassland community dynamics (Matsinos and Troumbis 2002). The model illustrates nicely how local interactions (dispersal and competition) determine overall community structure. The model focuses particularly on the effect of resilience in communities on gap-creating

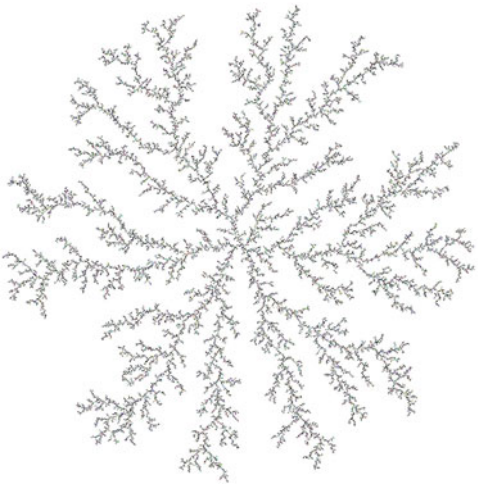

Fig. 8.7 Diffusion limited aggregation, produced with the fractal generator FRACTINT. A large number of mobile cells are started in the periphery, which attach by chance to a centrally located "fixed" cell, bringing up the typical diffusion-limited aggregation-pattern

disturbances (i.e. fire), imposed at different spatial extents. Model simulations were based on data from an experimental community with five grassland species.

A lattice of 200 $\times$ 200 square cells was used, with at most one individual of each species occupying a cell at a time. Global rules applying at the local scale determine the state of the system at each time step. The degree of influence of a cell to neighbouring cells follows a negative exponential decrease. The biological processes simulated are seed dispersal and competition. The following main procedures were used:

- Every individual of each species i produces seeds at a given rate p_i. We assume that the distribution of times between individual seed production is exponential, with a mean of $1/p_i$.
- The probability that a seed disperses from one cell to another depends on the species dispersal type (local, medium or long) and the distance between cells.
- Displacement of species is modelled in occupied cells depending on the competitive advantage of the invading species. The process of seed dispersal to neighbouring empty cells is modelled using a probabilistic algorithm.

The model starts with an assignment of seeds to donor cells in a random manner but with a frequency that is inversely related to donor cell distance. If a cell receives multiple seeds from different plant species, a random variable linked to the competition coefficient of the species determines the winner at that cell. All seeds are then eligible to sprout and will germinate at the next growing season; dormancy is not considered; the model does not consider environmental variability between years.

Parameters from an experimental biodiversity study of grasslands in Lesbos, Greece were used for the model specification. The experimental study was part of the European-wide research project BIODEPTH (Hector et al. 1999), aiming to

Table 8.1 Matrix of displacement probabilities that were used in the model

Displacing species *i*	Resident species *j*				
	Phalaris	*Bituminaria*	*Hordeum*	*Hirschfeldia*	*Lagoecia*
Phalaris	–	0.2	0.6	0.15	0.3
Bituminaria	0.1	–	0.01	0.03	0.04
Hordeum	0	0	–	0.01	0.01
Hirschfeldia	0.03	0.01	0.01	–	0.02
Lagoecia	0.1	0.01	0	0.01	–

investigate diversity–productivity relationships in natural grasslands. Experimental plots consisting of 2, 4, 8, 18, and 32 native species were established and maintained since 1997 in seven European countries. For the parametrization of our model we chose the two-species configurations, extracting relative biomass changes and estimating competition strength to simulate the interactions between five plant species: *Phalaris coerulescens*, *Hordeum geniculatum*, *Hirschfeldia incana*, *Lagoecia cuminoides*, and *Bituminaria bituminosa*. Among the five species, pair experiments have shown their competitive hierarchy and biomass changes in the plots yielded information on the competitive strength. This was assumed to translate into displacement probabilities in the model (Table 8.1).

Results showed that longer distance dispersing plants have a competitive advantage in colonization success as compared to better competitors, especially in the cases of disturbance-mediated creation of gaps in coverage. An increase in species number led to more resilient communities and a higher percent cover of the landscape. A further model adaptation therefore incorporated

- A scale-related neighbourhood structure
- Asymmetrical hierarchy in competition
- Invasion processes

The neighbourhood structure in the model was based on the dispersal attributes of the different species, and showed significant change in final assemblage patterns where short-distance dispersers were found to decrease in abundance. Asymmetrical hierarchy (in terms of competition) was modelled as a stochastic process, and showed to alter the composition of steady-state communities significantly, favouring assemblages with low overall diversity. Invasion was shown to interfere and alter the overall pattern of abundance. The effect of disturbances was studied as well, examining whether the community is resilient to disturbances or tends to change subsequent to disturbances (e.g. Fig. 8.8). The approach highlighted the emergence of complex community patterns from simple local interactions. A great amount of information is necessary for the parametrization of such a model, yet the outputs of the model provide a broader understanding of patterns that are far too complex to grasp with any other tool. Therefore, despite the relative complexity of the model, it provides the means to gain understanding of complex patterns in nature, the underlying mechanisms of which are otherwise poorly understood.

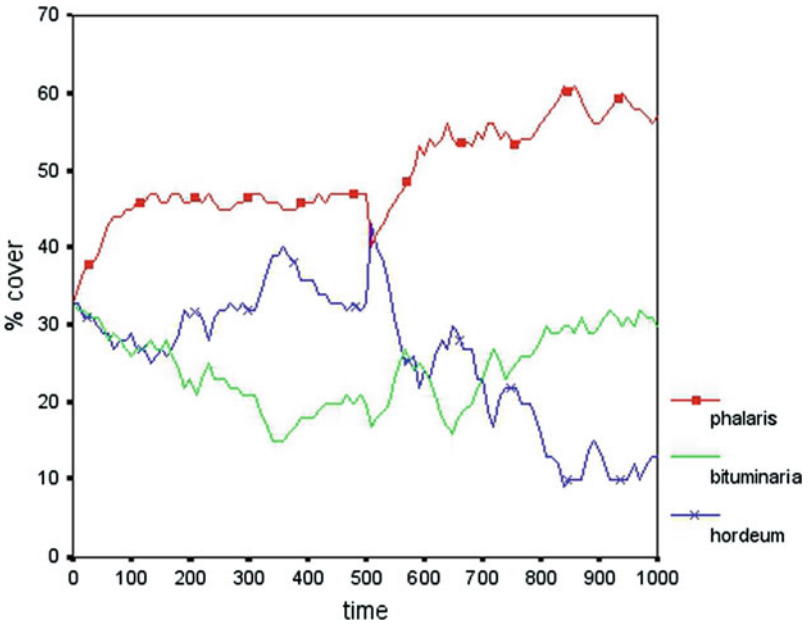

Fig. 8.8 Plot of relative abundance for 1,000 iterations for the three species: *Phalaris*, *Hordeum*, *Bituminaria* starting from same abundance (33%). Disturbance occurring on iteration 500 affects 20% of the landscape [from Matsinos and Troumbis (2002)]

8.6 Outlook and Applicability

In a unique way, Cellular Automata combine conceptual simplicity, the potential to expand simple interactions to complex structures, and an enormous range of application fields for quite demanding problems – with the potential to capture surprising self-organizing effects. This makes it worthwhile and desirable for any ecological modeller to familiarize with this approach.

It is possible to run CAs without much effort in pre-defined modelling environments, each of which specializes in a particular field of rule types. Yet, it is equally easy to escape the restrictions that customized software frequently have, and develop a unique CA according to one's specific applications, with the additional power to modify it to specific situations or explorations, e.g. by time-dependent or situation specific variations of the neighbourhood or through self-modifying rule systems. Other ecological modelling applications, and especially those that require a spatially structured input in order to provide an environment with particular statistical features, can be easily generated with a CA and used as a grid input. Clearly, Cellular Automata can contribute not only to strengthen ecological theory, but also for the development of predictive tools for ecology and conservation. In the process, one may reveal that modelling itself can be fun as well.