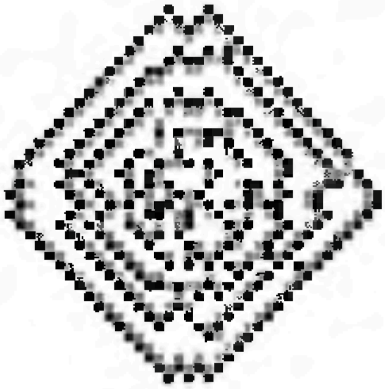

Chapter 6
Cellular Automata and Related Models

In Chapters 2 and 3, we saw models that describe the state of a system in time (e.g., the number of individuals as a function of time). However, in some applications, space is important as well. In Chapter 4, we saw how PDEs can be used to describe systems that depend on both time and space. In this chapter, we consider another way to include spatial information in a mathematical model, namely, with cellular automata.

6.1 Introduction to Cellular Automata

Cellular automata are models where all variables, both independent and dependent, take on discrete values. Their main characteristic is *locality*; that is, individuals or particles are only affected by their nearest neighbors. Cellular automata are therefore natural models for infectious diseases, forest fires, or excitable media where the contact between two individuals is typically of local nature. The dynamics of a cellular automaton is defined by *local* rules. In the case of infectious diseases, such a rule could be, "If at least one of my neighbors is infected, I will become infected myself." In a cellular automaton model, space is represented by discrete points in space, called "cells." The cells can be seen either as the locations where individuals live or, sometimes, as the individuals themselves. They can be interpreted as biological cells, as impoundments on a river with separated fish populations, as territories of birds, and so on.

Cellular automata and related models are interesting to biologists for several reasons. First, many structures in biology are discrete, and a natural model to describe such structures would be a discrete one (for example, DNA sequence data are discrete). Second, rules for birth, death, or migration can be specified in a straightforward manner. Third, the time courses and patterns of cellular automata can be interpreted directly in biological terms.

One of the first cellular automaton models was developed around 1952, by John von Neumann [32], in the context of self-replication. At that time, the question of how a complex system can create another system of the same complexity (e.g., a copy of itself) was discussed. The question seems trivial today, since we're familiar with the concept of DNA replication. But at that time, the idea was revolutionary. Von Neumann investigated his model by pencil and paper only. This is quite remarkable, since his model had 29 possible

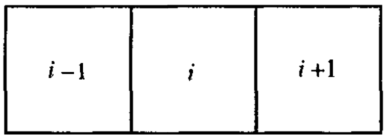

Figure 6.1. *One-dimensional neighborhood.*

states of the cells and, in a proof of self-replication, he outlined an initial configuration of about 200,000 cells.

The first tool to investigate cellular automata generally is to create a simulation. Until reasonably powerful computers were available, only a few results on cellular automata were published [166]. Broad exploration of cellular automata started with the availability of computers and the advent of user-friendly simulation tools.

The study of deterministic discrete systems is now a very active field. The theory is still developing and less established than the theories for continuous or stochastic systems. The behavior of discrete spatial systems is very rich and cannot be summarized with general results. From a theoretical point of view, cellular automata are very general structures. For example, there are cellular automata capable of universal computation, analogous to Turing machines [20, 62].

Before we give some specific examples of cellular automata, it helps to properly define a cellular automaton.

Definition 6.1. *A cellular automaton is a tuple $A = (G, E, U, f)$ of a grid G of cells, a set of elementary states E, a set defining the neighborhood U, and a local rule f.*

In classical cellular automata, we have $G = \mathbb{Z}^d$, the d-dimensional square grid. Instead of grid points, we draw cells which can be colored to indicate the state of the cells. We will extend this definition later to more general graphs. In an infectious disease model, "white" could indicate a susceptible and "black" an infectious individual. The possible states of each cell is an element of the set of *elementary states* $E = \{\text{white}, \text{black}\}$. Likewise, we could use $E = \{0, 1\}$. In one dimension ($d = 1$), a cell is often influenced by adjacent cells only. If we denote the cell at position i with x_i, then an example of a simple neighborhood would be $U(x_i) = \{x_{i-1}, x_i, x_{i+1}\}$ as shown in Figure 6.1. In two dimensions ($d = 2$), the most common neighborhoods are the *von Neumann* and the *Moore neighborhoods*. Let $x_{i,j}$ be the position of a cell. The von Neumann neighborhood (Figure 6.2, left) is described by $U(x_i) = \{x | \|x_i - x\|_1 \leq 1\}$. The Moore neighborhood (Figure 6.2, right) is described by $U(x_i) = \{x | \|x_i - x\|_\infty \leq 1\}$. Note that the use of a von Neumann or Moore neighborhood is not consistent throughout the literature. Sometimes the center cell $x_{i,j}$ is included in the neighborhood; sometimes only the surrounding cells are taken.

So far, we have a grid of cells with several possible elementary states and a neighborhood. The collection of all states of the cells of the grid is called the *state of the grid*. Mathematically, we define the *state of the grid* as a map $z : G \mapsto E$, which gives each grid cell an elementary state.

The *local rule*, f, describes what happens from one time step to the next. Let z be the state of the grid. Then $z(x)$ is the state of cell x and $z|_{U(x)}$ is the occupation of the

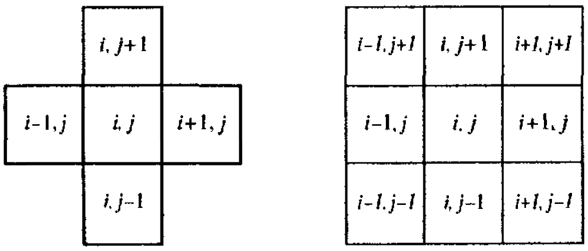

Figure 6.2. *Illustration of the von Neumann neighborhood (left) and the Moore neighborhood (right).*

neighborhood of cell x. The new state of cell x depends only on the states of its neighbor cells,

$$z(x)^{t+1} = f(z^t|_{U(x)}),$$

where the superscripts t and $t + 1$ denote the time steps. The local rule is applied to all cells in parallel, that is, *synchronously*, and is usually implemented in mapping two arrays alternating on each other. One of the arrays holds the current state at time t, while the other array holds the updated state at time $t + 1$.

6.1.1 Wolfram's Classification

Let us look at a class of cellular automata with $A = (\mathbb{Z}, \{0, 1\}, \{x_{i-1}, x_i, x_{i+1}\}, f)$, that is, the one-dimensional nearest-neighbor automata with states zero and one. There are $2^3 = 8$ possible occupations of the neighborhood. The local function can map to each possible neighborhood configuration to zero or one, which gives us $2^8 = 256$ different local functions, that is, 256 different automata. If we look at their dynamics, we will find quite different patterns. Wolfram [164] heuristically proposed four qualitative classes to characterize cellular automata:

Wolfram class I: From any initial configuration we get a fixed point; that is, the pattern becomes constant.

Wolfram class II: Simple stationary or periodic structures evolve. Small changes in the initial configuration affect only a finite number of cells.

Wolfram class III: "Chaotic" patterns emerge. Changes in the initial configuration affect a number of cells linearly growing in time.

Wolfram class IV: Complex localized patterns evolve with long-distance correlations. The effect of changes in the initial configuration cannot be predicted.

To identify each of the 256 local functions, Wolfram has constructed a simple system, known as Wolfram's enumeration: First, we order the eight possible neighborhood occupations $U_0 = 000$, $U_1 = 001$, $U_2 = 010, \ldots, U_7 = 111$, enumerated like three-digit binary numbers from 0 to 7. For a given local rule f, let $c_i = f(U_i)$ be the next state of cell x

Table 6.1. *Examples of Wolfram's enumeration.*

i	0	1	2	3	4	5	6	7	
U_i	000	001	010	011	100	101	110	111	
2^i	1	2	4	8	16	32	64	128	
Rule 254, c_i	0	1	1	1	1	1	1	1	$\sum c_i 2^i = 254$
Rule 50, c_i	0	1	0	0	1	1	0	0	$\sum c_i 2^i = 50$
Rule 90, c_i	0	1	0	1	1	0	1	0	$\sum c_i 2^i = 90$

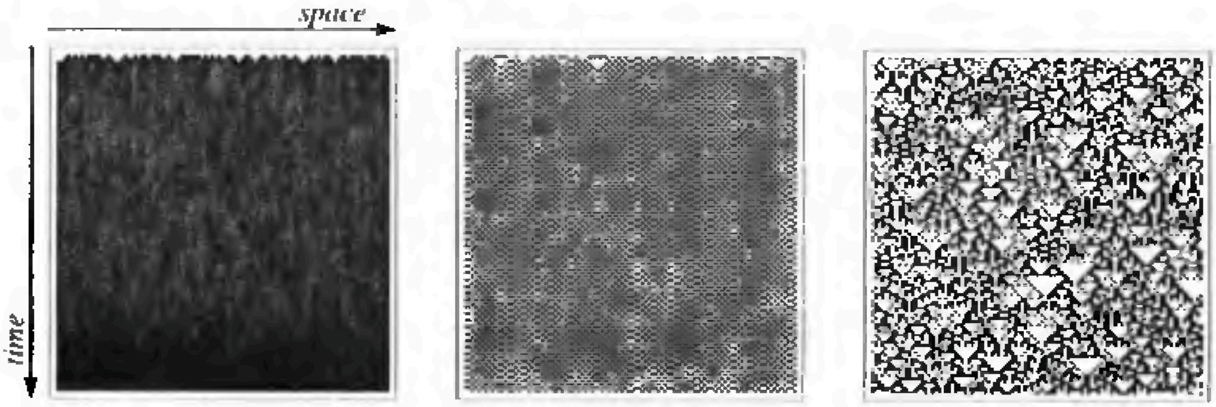

Figure 6.3. *Space-time patterns of the one-dimensional nearest-neighbor automata that correspond to the rules* 254, 50, *and* 90 *(from left to right). The first row in each square shows the initial configuration, which is the same for all three simulations. The following rows show successive iterates. White indicates* $z(x) = 1$ *and black indicates* $z(x) = 0$. *The boundary cells are kept constant* 0 *(white).*

with neighborhood U_i, for $i = 1, \dots, 7$. Then the number of this local rule is $\sum_{i=0}^{7}(c_i 2^i)$, as shown in Table 6.1. Let us discuss the rules with the numbers 254, 50, and 90.

Rule 254: Consider a simple infection automaton where a cell becomes infected if one of its neighbors is infected. The local function f for this situation is

$$f(z|_{U(x)}) = \begin{cases} 1 & \text{if } s \geq 1, \\ 0 & \text{otherwise,} \end{cases}$$

where $s = x_{i-1} + x_i + x_{i+1}$ is the number of infectious neighbors of cell $x = x_i$. As shown in Table 6.1, the number of this rule is $2+4+8+16+32+64+128 = 254$. The dynamical behavior of rule 254 is quite easily understood. If we start with a random configuration of ones and zeros, we reach a state where all cells are in state one. This means that all cells become infected, as shown in Figure 6.3 (left). If we start with all cells zero, then all cells will stay zero forever. In both cases, a stationary pattern is reached. Therefore, rule 254 is an example of an automaton from Wolfram class I.

Rule 50: Another simple infection model is rule 50. A cell becomes infected if one of its neighbors is infected, but infectious cells recover after one time step. Thus,

$$f(z|_{U(x)}) = \begin{cases} 1 & \text{if } z(x) = 0 \text{ and } s \geq 1, \\ 0 & \text{otherwise.} \end{cases}$$

Starting from a random configuration, we get a pattern where every cell periodically changes from zero to one and back (Figure 6.3, middle). Therefore, rule 50 is an example which belongs to Wolfram class II.

Rule 90: For an example of Wolfram class III, consider rule 90. It maps 000, 010, 101, and 111 to zero, and 001, 011, 100, and 110 to one. The patterns evolving have been described as "fractal-like" or chaotic (Figure 6.3, right).

There are no one-dimensional nearest-neighbor automata with class IV behavior, but we will come back to this later.

6.1.2 The Game of Life

In this section, we discuss one of the most popular cellular automata: the *Game of Life*, proposed by Conway (see [133]). It is a two-dimensional cellular automaton, in which each cell is either dead or alive. A living cell stays alive if it has two or three living neighbors; otherwise it dies. A dead cell becomes alive if exactly three neighbors are living. Biologically, these simple rules mimic a birth process where cells may die by isolation or overcrowding. More formally, we have $A = (\mathbb{Z}^2, \{0, 1\}$, Moore neighborhood without the cell itself, $f)$ with the local function

$$f(z|_{U(x)}) = \begin{cases} 1 & \text{for } s(x) = 3, \\ 1 & \text{for } s(x) = 2 \text{ and } z(x) = 1, \\ 0 & \text{otherwise,} \end{cases}$$

where s is the number of living neighbors

$$s(x) = \sum_{y \in U(x)} z(y).$$

From these simple rules, an astonishing behavior emerges. If we start with a random configuration of dead and living cells in a rectangular area of the grid, the pattern often becomes 2-periodic (but it may take some time). Groups of three living cells alternating in shape or stationary 4-blocks often appear (see Figure 6.4). However, configurations exist with a much more interesting behavior (these often have visual names like "ship" or "beehive"). Simple configurations can expand and resemble fireworks before they collapse again. The rich patterns can hardly be described here but should be viewed like a movie. Numerous simulation tools for the Game of Life are available, and readers are invited to experiment on their own. We conclude with a summary of observations that are interesting from a more theoretical point of view:

- The evolution of most initial configurations cannot be predicted. Small changes in the initial configuration may lead to totally different patterns.

- There are patterns with different, sometimes large, periods. For example, a configuration known as "queen bee" has period 30.

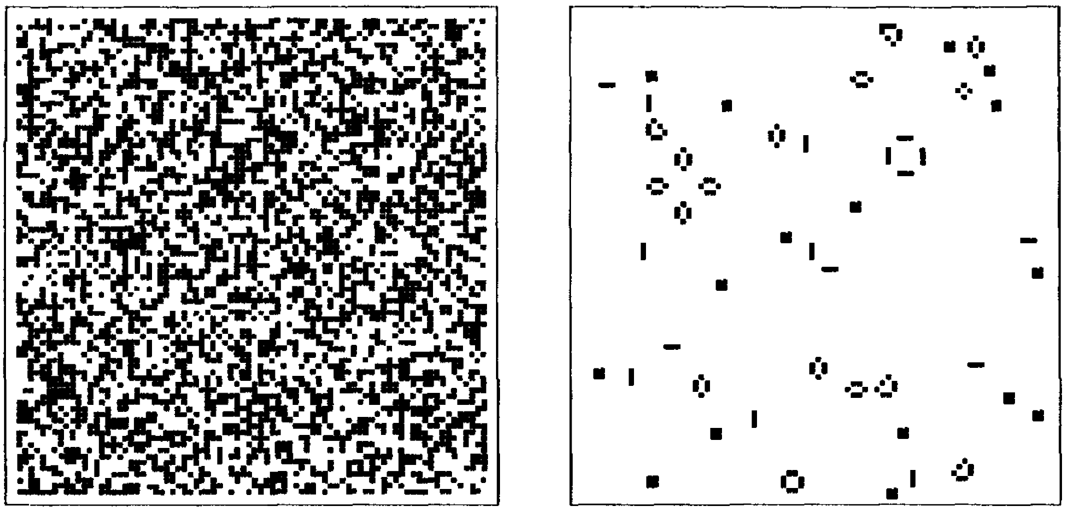

Figure 6.4. *Two-dimensional spatial patterns of Conway's Game of Life. Initial configuration (on the left) and after* 1300 *iterations (on the right). The boundary cells are kept constant* 0 *(white).*

- A configuration named "glider" travels diagonally across the grid. There is also the "glider gun" which generates a stream of gliders. It is an example of an unbounded pattern which expands forever, developing from an initial configuration with finite support.

- It has been shown that the Game of Life is computationally universal; that is, everything which is computable can be computed with the Game of Life [133], which indicates that this cellular automaton is very complex.

- Given the rich behavior of the Game of Life it is plausible that it is an example of Wolfram class IV.

6.1.3 Some Theoretical Results on Cellular Automata

After these two examples of cellular automata, the question of analytic results for cellular automata arises. Wolfram's classification is mainly phenomenological. An example of a theoretical result is the algorithm of how to construct, for every (finite) automaton A with arbitrary neighborhood, another automaton $\bar{A}$ simulating it with a von Neumann neighborhood. Usually, if we simplify the neighborhood, we have to "pay" with a larger number of elementary states; that is, we usually will have $|E| \ll |\bar{E}|$. Similarly, if we construct an automaton $\bar{A}$ with only the elementary states $\{0, 1\}$, we usually have to use a much larger neighborhood $\bar{U}$ with $|\bar{U}| \gg |U|$. This sounds useful in deriving the dynamics of a new automaton to find an analogous known automaton. However, an equivalent automaton with a basic neighborhood may have a very complicated, nonintuitive local function that is not well investigated.

As we mentioned, cellular automata are a developing field. Massive simulations are used to characterize automata models by tools from statistical physics, for example. Theoretical results for numerous classes of automata have also been obtained by using a wide variety of methods. In the following sections we discuss Greenberg–Hastings automata.

6.2 Greenberg–Hastings Automata

We follow [71] and investigate the Greenberg–Hastings automata, which have been formulated as models for excitable media such as the cardiac muscle, cultures of *Dictyostelium discoideum*, forest fires, or infectious diseases. For an infectious disease, each cell of the automaton represents an individual or the territory an individual lives in. Individuals are classified by their epidemiological state. They are either susceptible, infectious, or immune. An important assumption is that infection can only happen by direct contact between individuals, and only neighboring cells have contact. An infectious disease can then be modeled by the following rules:

- If a susceptible cell has at least one infectious neighbor, it becomes infectious itself; otherwise it stays susceptible.
- An infected cell stays infectious for $a > 0$ time steps; then it becomes immune.
- An immune cell stays immune for $g > 0$ time steps and then becomes susceptible again.

These rules can be used to model infection with different infectious or recovery periods. Let the set of elementary states be $E = \{0, 1, \ldots, a, a+1, \ldots, a+g\} = \{0, \ldots, e\}$. Then an infection is described by the transitions

$$\underbrace{0}_{susceptible} \longrightarrow \underbrace{1 \longrightarrow \cdots \longrightarrow a}_{infectious} \longrightarrow \underbrace{a+1 \longrightarrow \cdots \longrightarrow a+g}_{immune} \longrightarrow \underbrace{0}_{susceptible} .$$

More formally the local function of the Greenberg–Hastings automata reads

$$f(z|_{U(x)}) = \begin{cases} 1 & \text{if } z(x) = 0 \text{ and } s \geq 1, \\ z(x)+1 & \text{if } 0 < z(x) < e, \\ 0 & \text{otherwise,} \end{cases} \tag{6.1}$$

where s is the number of infectious cells in the neighborhood of x. Finally, we choose $G = \mathbb{Z}^2$ and the von Neumann neighborhood.

The above rule (6.1) implies that either a cell stays susceptible or it moves through the infection–recovery cycle to finally become susceptible again.

As in [71], let us concentrate on the case $a, g \geq 1$, with $a + g \geq 3$ and $1 \leq a \leq e/2$; that is, we have at least the same number of immune as infectious stages. Possible choices for $e \leq 6$ are shown in Table 6.2. Starting with a configuration with a finite number of

Table 6.2. *Greenberg–Hastings automata for $e \leq 6$.*

e	a = infectious stages	g = immune stages
3	1	2
4	1 or 2	3 or 2
5	1 or 2	4 or 3
6	1, 2, or 3	5, 4, or 3

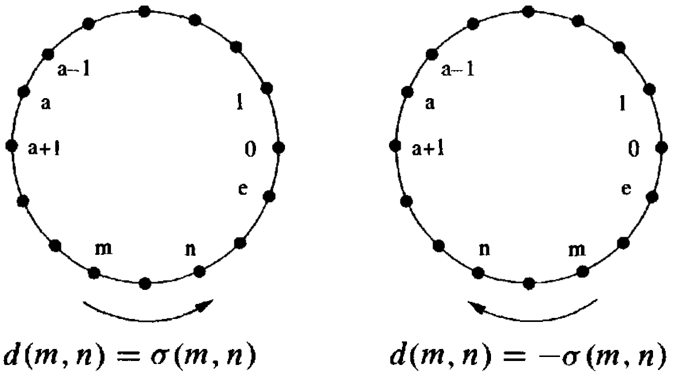

Figure 6.5. *Distance and signed distance between states.*

infected cells (finite support), there are two possibilities. Either the epidemic "dies out" (i.e., in any finite region of the grid all cells will be in state zero in finite time), or the pattern persists (i.e., there is at least one cell which becomes infected again and again). Greenberg, Greene, and Hastings [71] found a condition which allowed them to predict the fate of initial configurations. To investigate this prediction in some detail, we define the distance between states in E.

Definition 6.2.

(a) *The* distance *between two states* $m, n \in E$ *of cells is defined as*

$$d(m,n) = \min\{|m-n|, e+1-|m-n|\}.$$

(b) *We identify every state* $k \in E$ *with the point* $\exp(2\pi k/(e+1))$ *on the unit circle in the complex plane. The* signed distance *between two states* $m, n \in E$ *is*

$$\sigma(m,n) = \begin{cases} d(m,n) & \textit{if the arc } \overline{mn} \textit{ is oriented counter-clockwise,} \\ -d(m,n) & \textit{otherwise.} \end{cases}$$

In Figure 6.5, we illustrate how states can be located on the unit circle (mathematically speaking, how we calculate distances between states modulo $e+1$).

We see that the distance is the shorter arc between two states, therefore $d(m,n) = d(n,m)$. The above rule (6.1) implies that a cell can only advance by 1 state per unit of time, which implies $0 \le d(z(x)^t, z(x)^{t+1}) \le 1$. The signed distance includes information on the orientation of the arc, so we have $\sigma(m,n) = -\sigma(n,m)$.

The following definition focuses on the topological structure of cells.

Definition 6.3. *A* cycle *is an ordered n-tuple* $(x_1, \dots, x_n, x_{n+1})$ *such that* $x_1, \dots, x_n$ *are distinguished,* $x_{n+1} = x_1$, *and* x_{i+1} *is a neighbor of* x_i *for* $i = 1, \dots, n$.

Thus, a cycle is an ordered set of cells such that two successive cells are neighbors and the last and the first cells of the cycle are the same ones. To combine information on arrangement and states of the cells, we introduce the following definition.

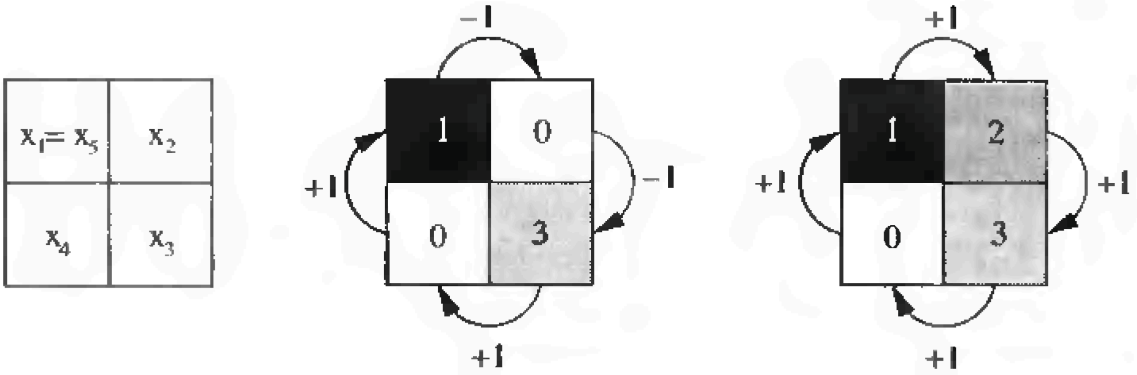

Figure 6.6. *Cycle with four different cells ($n = 4$). Let $a = 1, g = 2$, so that $e = 3$. Then both configurations show continuous cycles. The cycle in the middle has winding number zero; the one on the right has winding number one.*

Definition 6.4.

(a) *A cycle C is called* continuous *at time t if $d(z(x_i)^t, z(x_{i+1})^t) \leq a$ for $1 \leq i \leq n$.*

(b) *The* winding number $W_t(C)$ *of a continuous cycle at time t is defined as*

$$W_t(C) = \frac{1}{e+1} \sum_{i=1}^{n} \sigma(z(x_i)^t, z(x_{i+1})^t).$$

Figure 6.6 gives an example of a small cycle with four different cells ($n = 4, a = 1, g = 2, e = 3$). In both configurations, the four cells constitute a continuous cycle, one with winding number zero and the other with winding number one. We see that from these two initial configurations, different patterns develop (Figure 6.7). To achieve persistence, we basically have to ensure that susceptible cells of a cycle will be infected again. Only a cycle with nonzero winding number ensures persistence.

Now we can formulate the theorem for persistence.

Theorem 6.5 (Greenberg, Greene, and Hastings [71]). *Given a Greenberg–Hastings automaton with $a + g \geq 3$ and $1 \leq a \leq e/2$ and $a, g \geq 1$, a configuration with finite support is persistent if and only if there is a time $t' \geq 0$ where we find a continuous cycle C such that $W_{t'}(C) \neq 0$.*

In the theorem, the "if and only if" phrase suggests that we can determine the fate of any configuration, whether it will persist or it will die out. Unfortunately, this is not true. If we find a continuous cycle with winding number unequal to zero, then the configuration will persist. But even if there is no such cycle, as in the third example in Figure 6.7, it can happen that such a cycle evolves later. Therefore, we have only a sufficient, but not a necessary, condition for persistence if only the initial configuration is checked ($t' = 0$). A proof of Theorem 6.5 can be found in [71].

To obtain persistence for finitely supported initial conditions (only a finite number of cells is nonzero), it is sufficient to check a finite number of iterations for continuous cycles

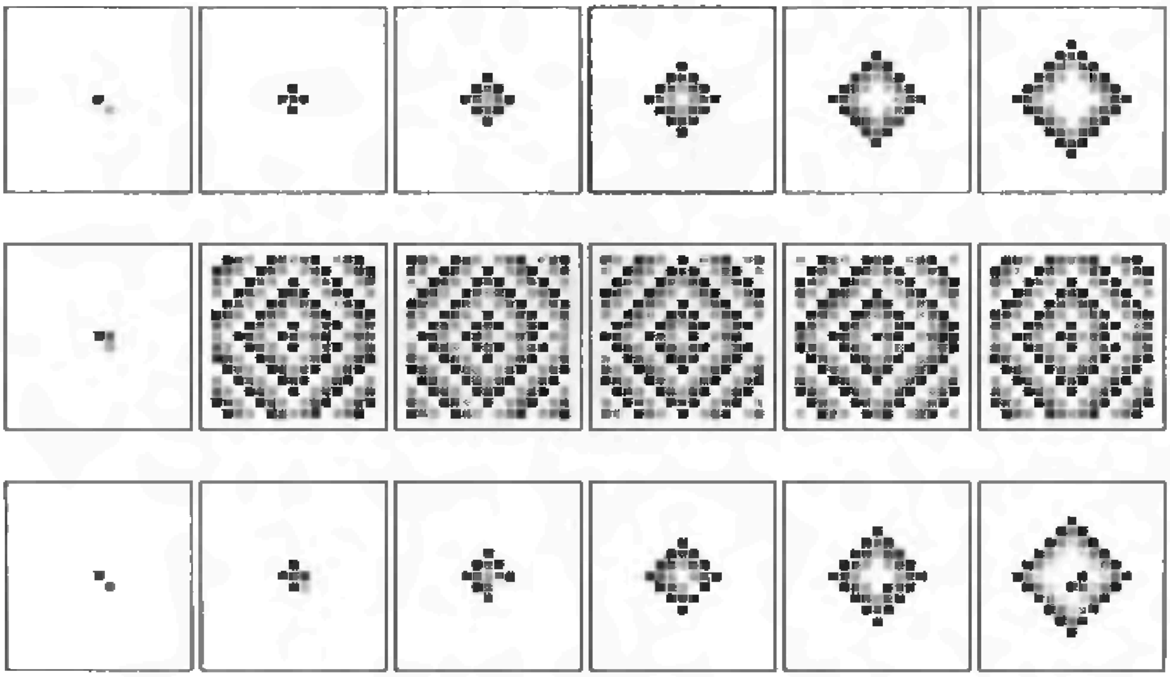

Figure 6.7. *Two-dimensional spatial patterns evolving from simple initial configurations. Each row represents one simulation and shows the patterns at* 0, 13, 14, 15, 16, *and* 17 *iterations. The cycle with winding number zero from Figure* 6.6 *(center) is used as the initial condition for the first row. An epidemic wave travels once across the grid and then dies out. The cycle with winding number nonzero from Figure* 6.6 *(right) is used as initial condition of the center row. It leads to persistence of the epidemic and a periodic pattern evolves. The last row shows an example where a continuous cycle with nonzero winding number evolves and leads to persistence. For these simulations, the boundary cells were fixed at* 0.

with nonzero winding number. An upper bound for the number of iterates that need to be checked can be computed explicitly [71].

6.2.1 Relation to an SIR Model

It is interesting to consider the connection between Greenberg–Hastings automata and the ODE model of Kermack–McKendrick [100], namely,

$$\begin{aligned}\dot{S} &= -\phi(S, I, R) + \gamma R,\\ \dot{I} &= \phi(S, I, R) - \alpha I,\\ \dot{R} &= \alpha I - \gamma R,\end{aligned}$$

where we use a general *incidence function* $\phi(S, I, R)$. In the classical Kermack–McKendrick model, a mass action term $\phi(S, I, R) = \beta S I$ is chosen. Further, α is the rate of immunization, so $1/\alpha$ is the mean time an individual stays in the infectious compartment. Therefore, $1/\alpha$ corresponds to the number of infectious states a of the Greenberg–Hastings automaton. Likewise, $1/\gamma$ corresponds to the number of recovery states g.

How is the incidence function related to the infection rule in the automaton? Here is the fundamental difference between the Kermack–McKenrick model and the Greenberg–Hastings automaton. The classical incidence function used in the Kermack–McKendrick model describes mass action. The individuals are well mixed, and every susceptible has the same chance to be infected by an infectious individual. Obviously this is not the case in the Greenberg–Hastings automaton. In this automaton, infectious cells can only infect neighboring susceptibles. This locality is the generator of the spatial patterns. The difference in the infection process leads to different time courses (or trajectories) of the two models. We can, however, approximate each model with the other. In particular, we can introduce a spatial mixing rule in the Greenberg–Hastings automaton, where the states of cells are exchanged at random. With increasing mixing rate, the automaton approximates the differential equation model. On the other hand, various incidence functions have been proposed for the ODE model to include the effect of local infection [84].

6.3 Generalized Cellular Automata

Classical cellular automata, as discussed so far, provide basic and very interesting models. However, sometimes they have to be adapted to fit a given biological process. The Greenberg–Hastings automata, for example, have been modified in the context of excitable media in many ways to get more realistic patterns. All four components of cellular automata—the grid, the elementary states, the neighborhood, and the local function—can be modified or generalized. Also, the synchronous updating method can be changed. There is an overlap between modified cellular automata and models that have been introduced in biology, physics, and chemistry under various names, such as *individual-based models* [72], *lattice gas models* [61], or *interacting particle systems* [109]. It is beyond the scope of this text to discuss which of these models may or may not be called cellular automata, but instead we introduce the most common modifications with examples.

6.3.1 Automata with Stochastic Rules

Returning to our rule in the Greenberg–Hastings model, namely, "a susceptible cell is infected if at least one of its neighbors is infectious," we note that infection is rare for a real infectious disease. Most diseases have a weaker level of infectivity, and the chance of becoming infected grows with the number of infected neighbors. Thus, a more realistic local rule is "the more infectious neighbors I have, the higher the probability I become infected." Since probabilities are involved, we obtain local functions with stochastic (random) components. Consider, for example,

$$f(z|_{U(x)}) = \begin{cases} 1 & \text{if } z(x) = 0 \quad \text{with probability } w(s), \\ 0 & \text{otherwise,} \end{cases}$$

where s is the number of infectious neighbors. Let p be the probability that an individual is infected by one of its neighbors. Then $(1 - p)^s$ is the probability that the individual is *not* infected in spite of s infectious neighbors. A good choice for the probability of becoming infected by s infected neighbors is $w(s) = 1 - (1 - p)^s$.

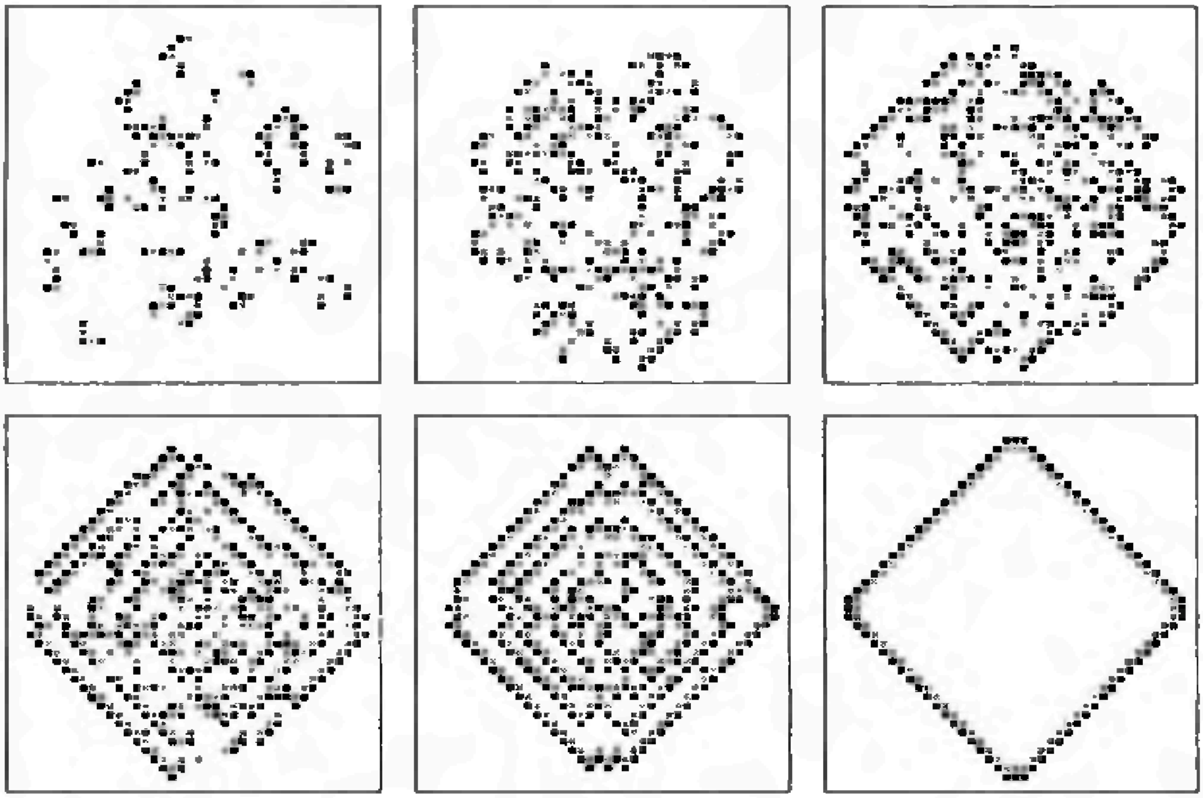

Figure 6.8. *Two-dimensional spatial patterns of a stochastic automaton evolving from an initial configuration of a square with nine infectious cells in the center (initial condition not shown). The infectivity parameter varies from* $p = 0.5$ *to* $p = 1$ *(in steps of* 0.1 *from top left to right bottom). Shown are states after* 30, 23, 22, 20, 18, *or* 18 *iteration steps, respectively.*

Let us consider a cellular automaton ($\mathbb{Z}^2$, $\{0, 1, 2\}$, von Neumann neighborhood, f) with $0 =$ susceptible, $1 =$ infectious, $2 =$ immune stage, and

$$f(z|_{U(x)}) = \begin{cases} 1 & \text{if } z(x) = 0 \quad \text{with probability } 1 - (1 - p)^s, \\ 0 & \text{otherwise.} \end{cases} \tag{6.2}$$

Figure 6.8 shows patterns evolving for different values of the parameter p. With $p = 1$, that is, immediate infection, one of the deterministic Greenberg–Hastings automata is recovered. With this particular initial configuration, the epidemic travels only once across the grid. The behavior for $p < 1$ is different. The epidemic is persistent, even for $p = 0.9$. It is plausible that the automata with $p \neq 1$ are more stable models; that is, the patterns produced are at least qualitatively the same when adding some stochasticity like a stochastic component in the local function, for example. This implies that models with stochastic components are the more realistic models. We see in Figure 6.8 also that the epidemic spreads slower with decreasing p and the epidemic will die out fast for very small p. To investigate the probability that the epidemic persists up to time t for a given value of p, we would need tools from stochastic theory, which exceed the scope of this book.