

CELLULAR AUTOMATA MODELS IN BIOLOGY

D. G. GREEN

Ecosystem Dynamics Group, Research School of Biological Sciences, Australian National University,
GPO Box 475, Canberra, ACT 2601, Australia

"The whole is greater than the sum of its parts"—Unknown

INTRODUCTION

Besides being the basis for modern computers, Alan Turing's conceptual automaton (the "Turing machine") also provides a useful way of viewing organisms. For instance, an organism (cf. automaton) is affected by the outside world (cf. inputs) and it behaves (cf. outputs) according to its current mood or condition (cf. state) and nature (cf. program). The "cellular automaton" [1] extends this analogy to provide a way of viewing whole populations of interacting "cells". By building appropriate rules into a cellular automaton, one can simulate many kinds of complex behaviour, ranging from the motion of fluids governed by the Navier–Stokes equations [2] to outbreaks of starfish on a coral reef [3].

CELLULAR AUTOMATA

A *cellular automaton* is an array of identically programmed automata that can interact with one another. With the advent of parallel computing, the "cellular automaton" has become central to computing science [1]. Equally as important is the vast range of potential applications of the idea [2]. The crucial difference between cellular automata and sets of independent automata is that the cells interact. These interactions can completely change the overall behaviour of a set of objects, so we need to consider cellular automata as whole systems, rather than as many individuals.

Despite their conceptual simplicity, cellular automata are capable of an astonishing variety of behaviour (Fig. 1). An important property is that they tend to be "self-organizing". That is, starting from complex, random cell configurations, the rules governing the system cause patterns to emerge from initial chaos. Even simple automata (Fig. 1) behave in ways reminiscent of complex biological systems. For example, the fate of any initial configuration of a cellular automaton is either: (a) to die out; (b) become stable or cycle with fixed period; (c) to grow indefinitely at a fixed rate; (d) to grow and contract irregularly [1]. Although the behaviour of arbitrary cellular automata is essentially unpredictable [2], empirical studies [1] reveal several classes of automata. Each class is characterized by qualitatively similar patterns or limiting forms: (a) spatial homogeneity; (b) stable or periodic structures; (c) chaotic behaviour; or (d) complicated local structure, some propagating.

APPLICATIONS TO BIOLOGICAL SYSTEMS

Many natural phenomena can be modelled as cellular automata. For instance, linear automata provide good models of pattern formation on mollusc shells [4] and the growth of dendritic crystals, such as snowflakes, are modelled as hexagonal automata growing from a seed [1].

The game "Life", invented by the Cambridge mathematician Gordon Conway, is a simple 2-D analogue of basic processes in living systems. The game traces changes in the patterns of "living" cells through time. The state of each cell depends on the number of living neighbours it has: with < 2 living neighbours cells they die from isolation, with > 3 they die from overcrowding; with exactly 3 living neighbours a birth occurs in a dead cell. As with linear automata, these rules tend to produce order from arbitrary initial configurations. Ultimately, most configurations either disappear or break up into isolated pieces of pattern.

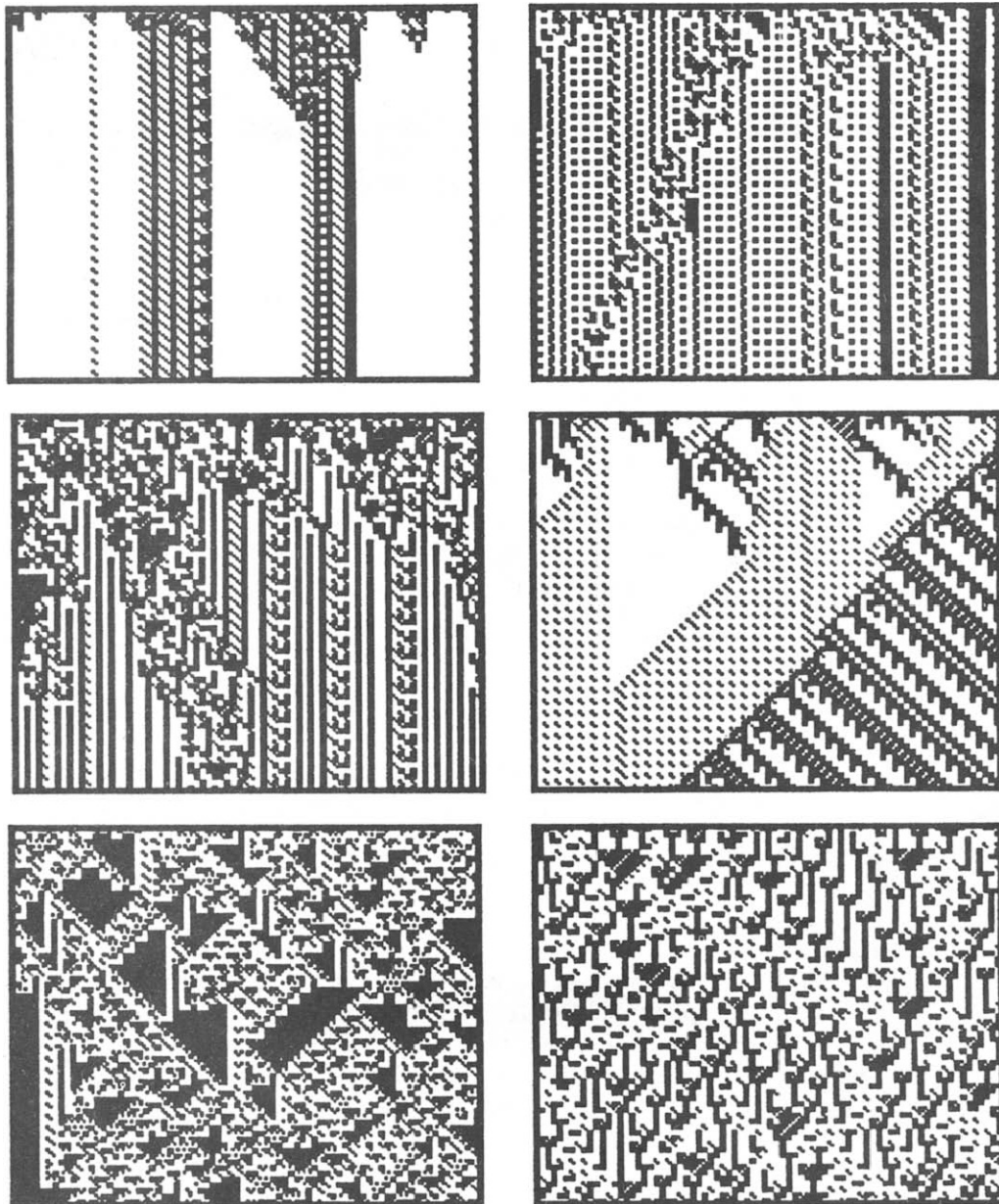


Fig. 1. Simple linear automata. Successive generations (layers) are produced by applying transition rules to the layer above. In the copying process, the state s' of each daughter cell is determined by the state s of the parent cell, plus the states a , b of the two cells adjacent to its parent. In the examples shown, each cell in the first row was assigned at random to one of three states. Notice that the rules governing replication of the string cause patterns to emerge.

Cellular automata models of landscapes consist of fixed arrays in which each cell represents an area of the land surface [5–8]. The states associated with each cell correspond to environmental features, such as coral cover or topography. This approach is compatible with both pixel-based satellite imagery and with quadrat-based field observations. It also enables processes that involve movement through space (e.g. fire, dispersal) to be modelled in a “natural” fashion (Fig. 2).

Cellular automata are usually considered to be simple grids (in n -dimensions). However, allowing other arrangements of the cell population makes it possible to represent many more biological systems as cellular automata. For example, models of growth [4] call for cellular automata in which the “cell” population grows and the spatial arrangement of cells reflects the architecture of the organism concerned (Fig. 3). Models of behaviour [9] call for cellular automata in which the cells correspond to individual animals, with the relationships between cells variable and reflecting either



Fig. 2. A cellular automaton model of a fire spreading through a landscape [5]. The fire front is marked in black and the area of devastation in grey. The model treats ignition as an epidemic process and cell states include fuel loadings and other relevant parameters. In this scenario a fire ignites in forest (shaded) and is initially driven by a north wind. A strong windshift from the northeast drives it across open fields (white), whereupon it threatens a town (black dots). Other features shown include roads and a lake.

the changing spatial locations of the animals, or else the social structure of a group. “Turtle geometry” [10] provides an excellent starting point for modelling animal movement through space.

Many biological phenomena are best represented, not as a single cellular automaton, but rather as sets of distinct cellular automata that interact with one another. For example, it is clearly desirable to distinguish organisms (e.g. insects) from their environment [11, 12] and to represent sessile organisms (e.g. plants, corals) separately from motile organisms (e.g. herbivores, starfish). It is also desirable to distinguish processes that happen on vastly different time or spatial scales. For instance, in a model of forest dynamics a fire is best represented as a growing collection of burning fuel cells, each of which corresponds to (and affects) a cell in the forest grid [5].

A final variation of cellular automata suggested by biological applications is an automaton in which the programming, although initially uniform, can mutate or vary. This type of model is inspired by the feedback that occurs between organisms and their environment, by questions of “nature vs nurture” and the need to understand the interplay between genetic variation and survival strategies in heterogeneous environments [6].

General properties of the above biologic variations of cellular automata have yet to be characterized. However, landscape models perform in much the same way as simple 2-D automata [5].

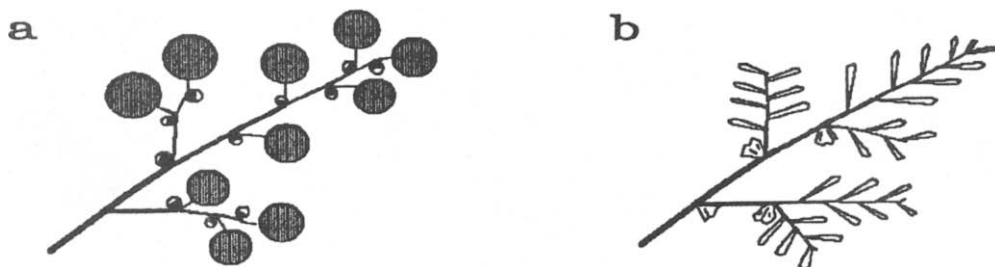


Fig. 3. Growth patterns for two tree species. Simplified models are given below the diagrams. Names in angle brackets denote growth structures and arrows denote growth processes. The states of particular structures are indicated by colons. The models differ syntactically in the timing of fruit production and semantically in the shapes of leaves and fruit.

(a) *Eucalyptus alpina*

<bud:leaf,angle>	→	<meristem>	<bud:stem,angle>
<bud:stem,angle>	→	<stem>	<bud:leaf,angle + increment>
<meristem>	→	<leaf><fruit>	<bud:juvenile, 0>
<bud:juvenile, angle>	→		<bud:stem,angle>/delay = 1 year

(b) *Hakea salicifolia*

<bud:leaf,angle>	→	<meristem>	<bud:stem,angle>
<bud:stem,angle>	→	<stem>	<bud:leaf,angle + increment>
<meristem>	→	<leaf>	<bud:juvenile, 0>
<bud:juvenile,angle>	→	<fruit>	<bud:stem,angle>/delay = 1 year

That is, regular processes tend to produce patterns (Fig. 4). More importantly, the resulting patterns and dynamics can be linked to issues of biological significance. For example, seed dispersal causes clumps of vegetation to form, thus counteracting competition and promoting the persistence of established plant communities. Dispersal is just one example of a *convolution process*, i.e. each cell affects a whole neighbourhood (or vice versa). A great many landscape processes (e.g. fire spread, animal movement) can be represented as convolutions. Because of the context-sensitive nature of most biological phenomena, convolution predominates in many other sorts of automata as well.

SYNTACTIC MODELLING

An important practical issue in modelling with cellular automata is how to define their programming. As the actions of automata are governed by rules, it is natural to express these rules within the syntax of a formal language (e.g. Fig. 2). Syntactic methods *per se* have become important means of modelling in biology [7, 8].

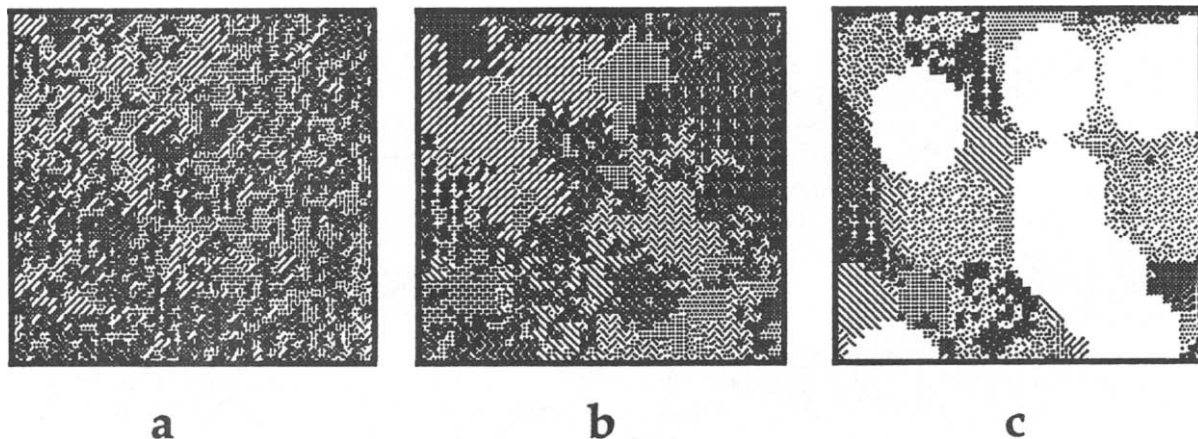


Fig. 4. Patterns of plant species distributions that emerge in three scenarios: (a) seeds of all species compete for any vacant growing site; (b) vacant sites are filled from local seed sources; (c) as in (b), but fires occur at regular intervals. Note the clumping that occurs in (b) and the large patches (and loss of species) generated in (c) when fire is added to the system.

Model structure:

<model> → **MODEL** *<world>* *<problem>* **RUN** [; *<model>* ; ...] **END**.
<world> → *<space>* *<time>* *<environment>* *<taxa>* *<processes>* **END**;

Space and time:

<space> → **SPACE** *<scale>* [, *<scale>* ...] *<initial state>* ;
<initial state> → (*file* | *patchy* | *random* | *uniform*) / *<option>* ;
<time> → **TIME** *<scale>* [; *<scale>* ; ...] ;
<scale> → *<variable>* = *<size>* ;
<size> → (*<minimum>* , *<maximum>*) [/ *<option>*] ;
<option> → *continuous* | *discrete* (/ *interval* = *<number>*) ;

Environmental definitions:

<environment> → **ENVIRO** *<variable>* ; *<e-type>* *<initial state>* ;
<e-type> → *space* | *time* | *neighbourhood* | *all* [*constant* | *random* | *gradient* = *<function of space or time>*] ;
<taxa> → **TAXON** *<name>* : (*fixed* | *mobile*) [*<initial state>* *<growth>* *<reproduction>* *<behaviour>*] [, *<taxa>* , ...] ;
<processes> → *<process>* [; *<process>* ; ...] ;
<process> → *<behaviour>* | *<disturbance>* | *<environmental change>* | *<growth>* | *<reproduction>* | *<succession>* ;
<succession> → **IF** *<condition>* **THEN** *<transition>* [, *<transition>* , ...] ;
<transition> → *<state>* -> *<state>* (/ *delay* = *<number>* , / *pr* = *<number>*) ;
<condition> → *<variable>* *<relation>* *<algebraic expression>* [*<connective>* *<condition>*] ;
<state> → *<name>* = *<value>* ;

Fig. 5. Tentative outline for an “ecosystem language” designed to simplify development of cellular automata models of ecosystem processes [13]. BNF notation is used (see the text). Terms in *italics* are undefined, but have intuitively obvious meanings.

A formal language $L(G)$ consists of all possible expressions that can be derived from the underlying grammar G [14]. This grammar G consists of four elements: a set of basic symbols (the *alphabet*), a set of names for varying entities (the *variables*), a set of rules governing the constructing of permissible expressions (*words*) in the language (the *syntax*) and a set of expressions from which all other words in the language are derived (the *initial* or *starting words*). Syntax rules usually take the form “ $A \rightarrow B$ ”, denoting that some combination of symbols A gives rise to a new combination of symbols B . Languages are called *context-free* if in every rule in the syntax, A in the above expression is a variable; otherwise the language is called *context-sensitive*. In other words, a language is context-free if every rule depends solely on the variable being transformed and not upon the context in which it occurs. A language is called *regular* if every rule in the syntax has one of the three forms: “ $x \rightarrow a$ ”, “ $x \rightarrow ay$ ”, or “ $x \rightarrow ya$ ”, where a is a symbol from the alphabet and x and y are any expressions.

A convenient way of defining syntax (e.g. Figs 3 and 5) is a notation called Backus Normal Form (“BNF notation”). Its main symbols are as follows:

- terms enclosed in angle brackets (e.g. “ $\langle \text{noun} \rangle$ ”) are the names of variables in the language;
- arrows (“ \rightarrow ” or “ $:-$ ”) denote rules (i.e. definitions or transitions);
- terms in curly brackets (“{ }”) indicate possible repetitions;
- terms grouped in square brackets (“[]”) indicate alternative, which are separated by bars (“|”).

SUMMARY

Cellular automata provide global representations for context-sensitive processes that involve discrete state changes in interacting populations of discrete entities. The usual arrangement of cells is a rectangular grid, but different or flexible topologies are appropriate for some processes, such as growth. Many biological processes are best modelled as interactions between separate cellular automata. Convolution processes are common.

REFERENCES

1. S. Wolfram, Cellular automata as models of complexity. *Nature* **311**, 419–424 (1984).
2. G. Wilson, The life and times of cellular automata. *New Scient.* **120**, 44–49 (1988).
3. R. E. Reichelt, S. Bainbridge and D. G. Green, Crown-of-thorns dispersal in the Great Barrier Reef—a simulation study. *Maths Comput. Simuln* **30**, 145–150 (1988).
4. G. T. Herman and G. Rozenberg, *Development Systems and Formal Languages*. North-Holland, Amsterdam (1975).
5. D. G. Green, Modelling forest mosaics. In *System Modelling and Optimization* (Edited by S. Ikawa and K. Yajima), pp. 584–593. Springer, Heidelberg (1988).
6. D. G. Green, A. P. N. House and S. M. House, Simulating spatial patterns in forest ecosystems. *Maths Comput. Simuln* **27**, 191–198 (1985).
7. D. G. Green, R. H. Bradbury and R. E. Reichelt, Formal languages and biological pattern. *J. infer. deduct. Biol.* **5**, 47–66 (1986).
8. D. G. Green, R. H. Bradbury and S. Bainbridge, Embodiment of formal languages. *Maths Comput. Simuln* **30**, 39–44 (1988).
9. R. S. Westman, Environmental languages and the functional basis of behaviour. In *Quantitative Methods in the Study of Animal Behaviour* (Edited by B. A. Hazlett), pp. 145–201. Academic Press, New York (1977).
10. S. Papert, Uses of technology to enhance education. LOGO Memo No. 8, MIT Artificial Intelligence Lab., Boston, Mass. (1973).
11. P. Hogeweg and B. Hesper, Two predators and one prey in a patchy environment: an application of MICMAC modelling. *J. theor. Biol.* **93**, 411–432 (1981).
12. P. Hogeweg and B. Hesper, The ontogeny of the interaction structure in bumble bee colonies: a MIRROR model. *Behav. Ecol. Sociobiol.* **12**, 271–283 (1983).
13. D. G. Green and R. H. Bradbury, *Ecol. Modelling* (in press).
14. J. E. Hopcroft and J. D. Ullman, *Formal Languages and Their Relation to Automata*. Addison-Wesley, Reading, Mass. (1969).