Towards Agend-Based Modeling: Cellular Automata

Computational Models for Complex Systems

Paolo Milazzo

Dipartimento di Informatica, Università di Pisa
http://pages.di.unipi.it/milazzo
milazzo@di.unipi.it

Laurea Magistrale in Informatica
A.Y. 2019/2020

Introduction

Agent-Based Modeling is a modeling approach in which system components are represented as agents able to

- take decisions
- perform actions
- interact with other agents and the environment

Agents behaviors is often specified using a high-level (programming) language

Agent-Based Simulation is a form of Discrete Event Simulation that consists in "executing" agents concurrently

Agent-Based Modedling is a natural approach for complex systems

Spatial Aspects of Agent-Based Models

- Very often, agents move in an 2D/3D environement
- Agent position and spatial characteristics of the environment influence the system dynamics
  - interaction with neighbours (and notion of neighbour)
  - spatial constraints (e.g. roads) and obstacles
  - spatial distribution of resources (e.g. food) or areas with different characteristics (metropolitan areas, open fields, rivers, lakes, ...)

Agent-Based Models and Cellular Automata

- Cellular Automata (CA) allow describing 1D, 2D or 3D environments
- The environment consists of a matrix of cells
- Each cell has its own state that can evolve by means of rules
- CA are simpler than Agent-Based Models but
  - can be used to model some types of Complex Systems with a spatial structure
  - the way they model spatial aspects of the environment is usually adopted also by Agent-Based Modelling methods
- So... it makes sense to study Cellular Automata and then Agent-Based Modelling methods...

Resources Available Online

This lesson is mostly based on the companion slides for the book

- Bio-Inspired Artificial Intelligence: Theories, Methods, and Technologies by Dario Floreano and Claudio Mattiussi, MIT Press

The original slides are available here:
`baibook.epfl.ch/slides/cellularSystems-slides.pdf`

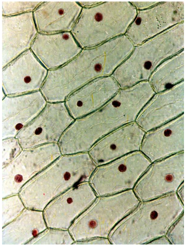
Moreover, on the paper

- Cellular Automata and Applications by Gavin Andrews available online.

Cellular Automata

Motivation

Evolution has rediscovered several times multicellularity as a way to build complex living systems



- Multicellular systems are composed by many copies of a unique fundamental unit - the cell
- The local interaction between cells influences the fate and the behavior of each cell
- The result is an heterogeneous system composed by differentiated cells that act as specialized units, even if they all contain the same genetic material and have essentially the same structure

Cellular Automata

Fields of Application

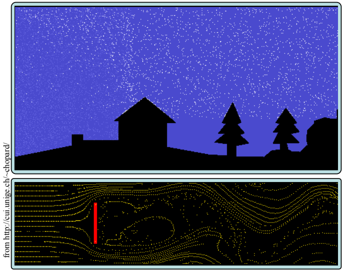
The concept of “many simple systems with (geometrically structured) local interaction” is relevant to:

- **Artificial Life** and **Evolutionary Experiments**, where it allows the definition of arbitrary “synthetic universes”.
- **Computer Science** and **Technology** for the implementation of parallel computing engines and the study of the rules of emergent computation.
- **Physics**, **Biology,** and other sciences, for the modeling and simulation of complex biological, natural, and physical systems and phenomena, and research on the rules of structure and pattern formation.
  - More generally, the study of **complex systems**, i.e., systems composed by many simple units that interact non-linearly
- **Mathematics**, for the definition and exploration of complex space-time dynamics and of the behavior of dynamical systems.

Cellular Automata

Modeling complex phenomena

Many complex phenomena are the result of the collective dynamics of a very large number of parts obeying simple rules.



Unexpected global behaviors and patterns can emerge from the interaction of many systems that “communicate” only locally.

Companion slides for the book *Bio-Inspired Artificial Intelligence: Theories, Methods, and Technologies* by Dario Floreano and Claudio Mattiussi, MIT Press 4

Cellular Automata

Modeling cellular systems

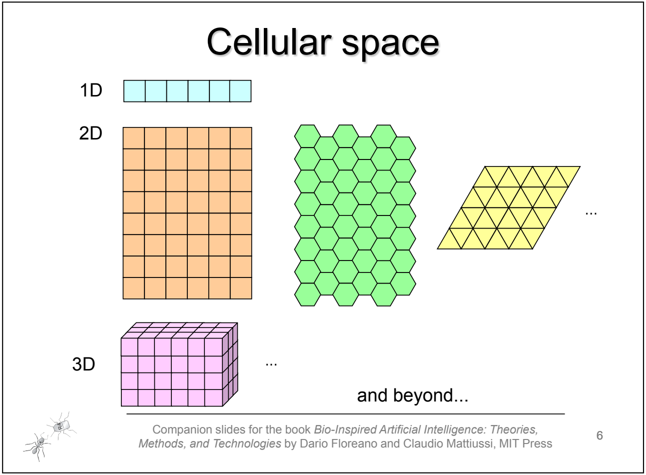
We want to define the simplest nontrivial model of a cellular system.
We base our model on the following concepts:

- *Cell* and *cellular space*
- *Neighborhood* (local interaction)
- Cell *state*
- *Transition rule*

We do not model all the details and characteristics of biological multicellular organisms but we obtain simple models where many interesting phenomena can still be observed

- There are many kinds of cellular system models based on these concepts
- The simplest model is called *Cellular Automaton* (CA)

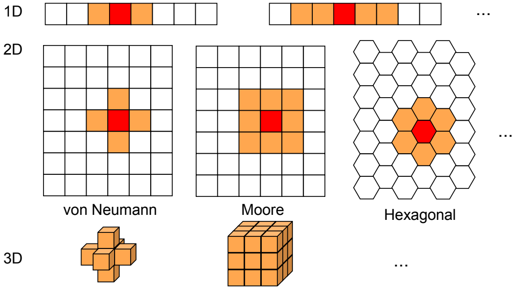
Cellular Automata



Cellular Automata

Neighborhood

- Informally, it is the set of cells that can influence *directly* a given cell
- In *homogeneous* cellular models it has the same shape for all cells



Companion slides for the book *Bio-Inspired Artificial Intelligence: Theories, Methods, and Technologies* by Dario Floreano and Claudio Mattiussi, MIT Press

7

Cellular Automata

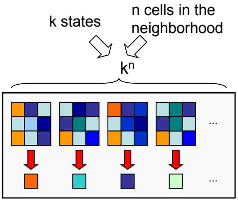
State Set and Transition Rule

The value of the **state** of each cell belong to a finite set, whose elements we can assume as being numbers. The value of the state is often represented by cell colors. There can be a special **quiescent state** s_0.

$$S = \{s_0, \ldots, s_{k-1}\} = \{0, \ldots, k-1\} = \{\bullet, \ldots, \bullet\}$$

The **transition rule** is the fundamental element of the CA. It must specify the new state corresponding to each possible configuration of states of the cells in the neighborhood.

The transition rule can be represented as a **transition table**, although this becomes rapidly impractical.

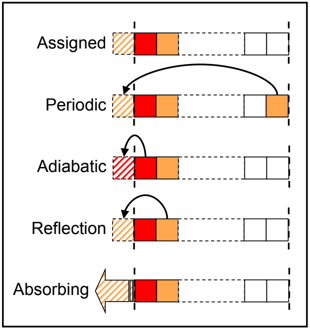


transition table

Cellular Automata

Boundary Conditions

- If the cellular space has a boundary, cells on the boundary may lack the cells required to form the prescribed neighborhood
- *Boundary conditions* specify how to build a "virtual" neighborhood for boundary cells

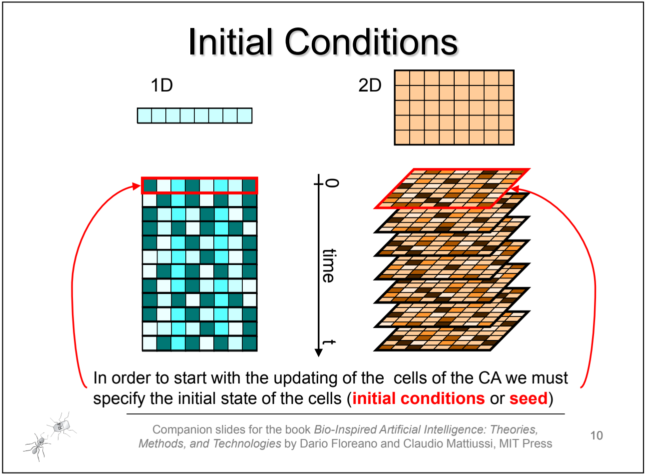


Some common kinds of boundary conditions

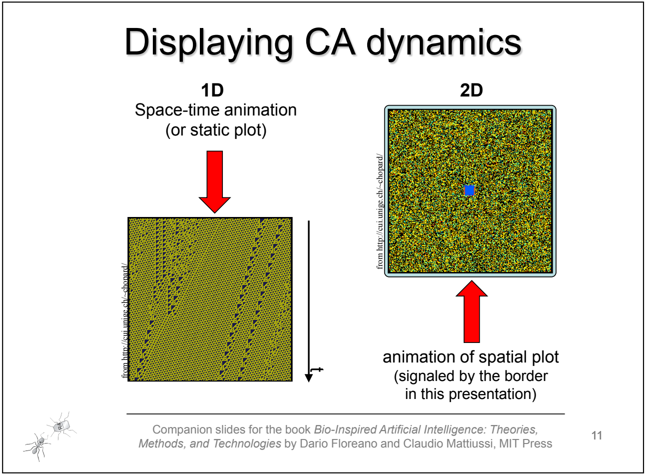
Companion slides for the book *Bio-Inspired Artificial Intelligence: Theories, Methods, and Technologies* by Dario Floreano and Claudio Mattiussi, MIT Press

9

Cellular Automata



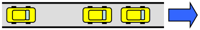
Cellular Automata



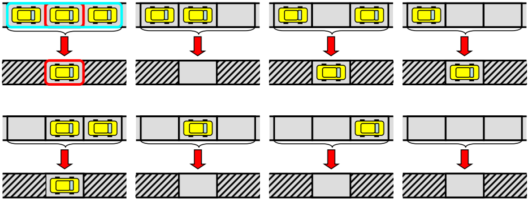
See `http://cui.unige.ch/~chopard/CA/Animations/CA/random.html` for an animation of the 2D example

Cellular Automata

Example: Modeling Traffic

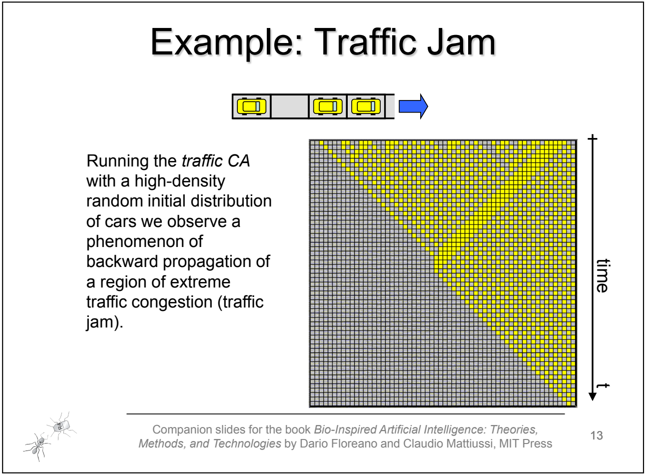


We construct an elementary model of car motion in a single lane, based only on the **local** traffic conditions. The cars advance at discrete time steps and at discrete space intervals. A car can advance (and must advance) only if the destination interval is free.



Companion slides for the book *Bio-Inspired Artificial Intelligence: Theories, Methods, and Technologies* by Dario Floreano and Claudio Mattiussi, MIT Press 12

Cellular Automata



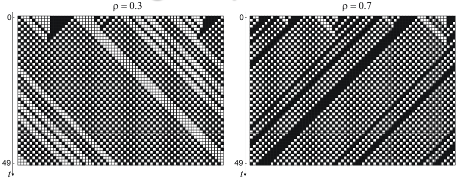
In a 1D-CA
each row shows a step

At each step, the state of all cells is updated according to the rules

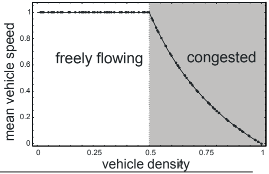
The dynamics of the systems (queue of cars) emerges from the rules describing local behaviors (individual cars)

Cellular Automata

Emergent phenomena



There is a qualitative change of behavior for ρ = 0.5. In the language of physics there is a *phase transition* between the two regimes at the *critical density* ρ = 0.5



Companion slides for the book *Bio-Inspired Artificial Intelligence: Theories, Methods, and Technologies* by Dario Floreano and Claudio Mattiussi, MIT Press 14

Cellular Automata

In practice...

To implement and run a CA experiment

1. Assign the geometry of the CA space
2. Assign the geometry of the neighborhood
3. Define the set of states of the cells
4. Assign the transition rule
5. Assign the boundary conditions
6. Assign the initial conditions of the CA
7. Repeatedly update all the cells of the CA, until some stopping condition is met (for example, a pre-assigned number of steps is attained, or the CA is in a quiescent state, or cycles in a loop,...).

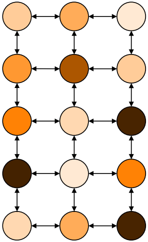
Companion slides for the book *Bio-Inspired Artificial Intelligence: Theories, Methods, and Technologies* by Dario Floreano and Claudio Mattiussi, MIT Press 15

Cellular Automata

Informal definition of CA

A Cellular Automaton is

- a **geometrically structured** and
- **discrete** collection of
- **identical** (simple) systems called **cells**
- that **interact** only **locally**
- with each cell having a local **state** (memory) that can take a **finite** number of values
- and a (simple) **rule** used to **update** the state of all cells
- at **discrete time** steps
- and **synchronously** for all the cells of the automaton (global "signal")



Companion slides for the book *Bio-Inspired Artificial Intelligence: Theories, Methods, and Technologies* by Dario Floreano and Claudio Mattiussi, MIT Press

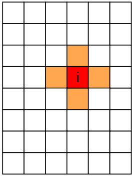
16

Cellular Automata

Formal definition of CA

A Cellular Automaton is

- an **n-dimensional lattice** of
- **identical** and **synchronous** finite state machines
- whose state s is updated (synchronously) following a **transition function** (or transition rule) ϕ
- that takes into account the state of the machines belonging to a **neighborhood** N of the machine, and whose geometry is the same for all machines



$$s_i(t+1) = \phi(\, s_j(t)\,;\, j \in N_i\,)$$

Companion slides for the book *Bio-Inspired Artificial Intelligence: Theories, Methods, and Technologies* by Dario Floreano and Claudio Mattiussi, MIT Press 17

Cellular Automata

Special Rules

The transition table of a generic CA can have an enormous number of entries. Special rules can have more compact definitions.

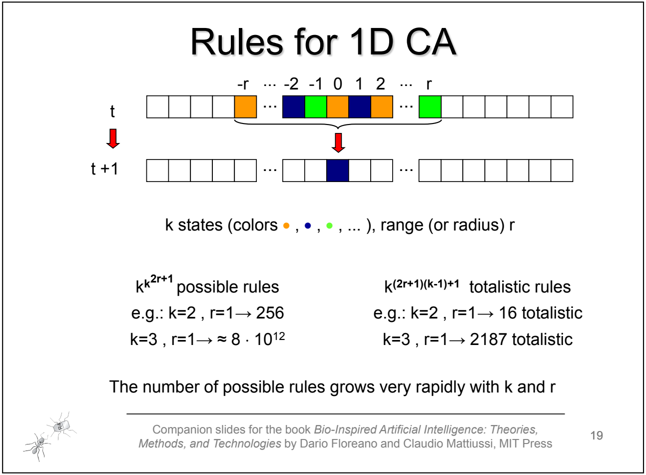
A rule is **totalistic** if the new value of the state depends only on the **sum** of the values of the states of the cells in the neighborhood

$$s_i(t+1) = \phi(\Sigma_j s_j(t) \; ; j \in N_i)$$

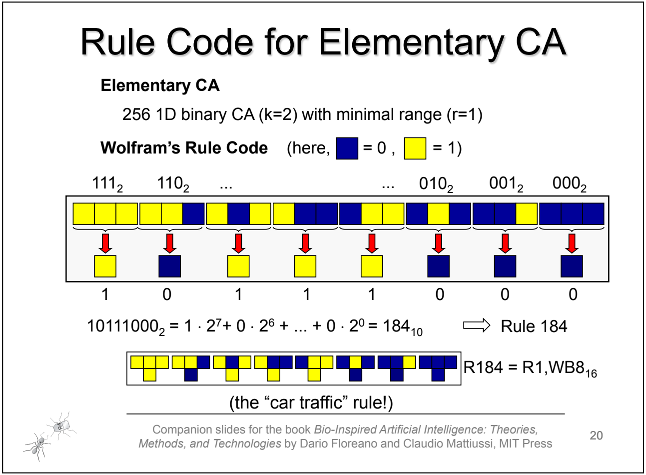
A rule is **outer totalistic** if the new value of the state depends on the value of the state of the updated cell and on the sum of the values of the states of the other cells in the neighborhood

$$s_i(t+1) = \phi(s_i(t) \, , \Sigma_j s_j(t) \; ; j \in N_i \, , j \neq i)$$

Cellular Automata



Cellular Automata

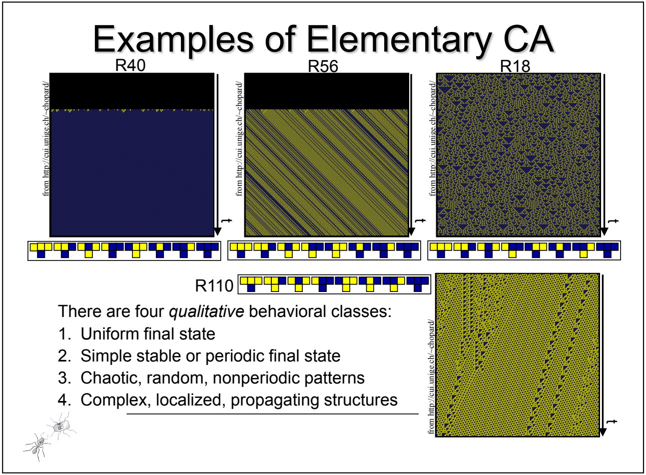


©Dario Floreano and Claudio Mattiussi

Elementary CA = 1D-CA with binary states and minimal neighborhood

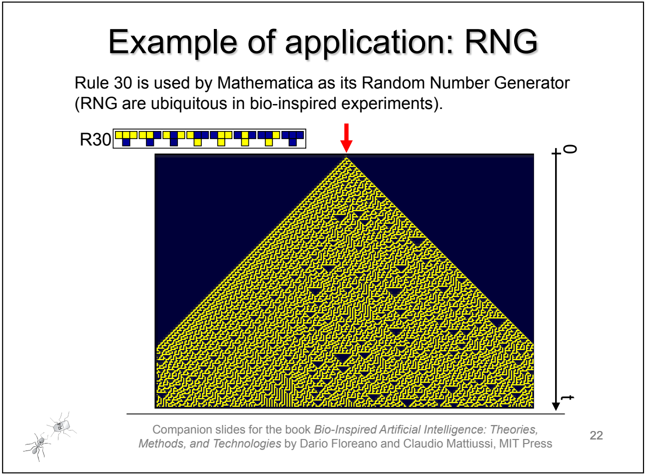
Only 256 elementary CAs can be defined...

Cellular Automata



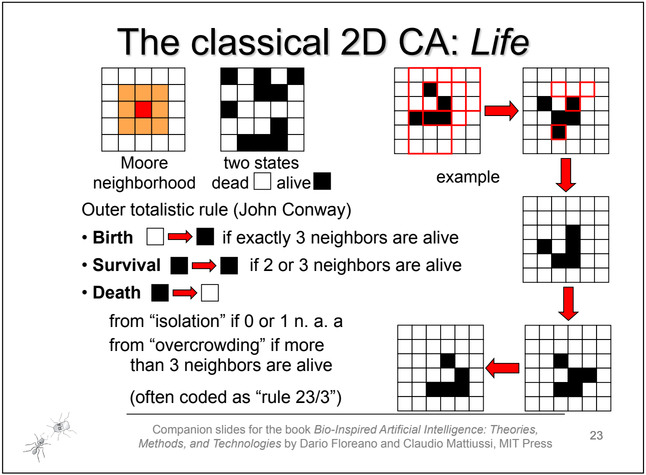
©Dario Floreano and Claudio Mattiussi

Cellular Automata



Steven Wolfram's recommendation for random number generation from rule 30 consists in extracting successive bits in a fixed position in the array of cells, as the automaton changes state

Cellular Automata



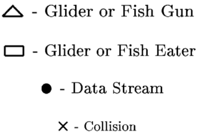
Although very simple, the rules of Conway's Game of Life allow creating patterns with interesting behaviors:

- blinkers and other periodic oscillators
- gliders/spaceships able to move
- glider guns able to periodically create new gliders

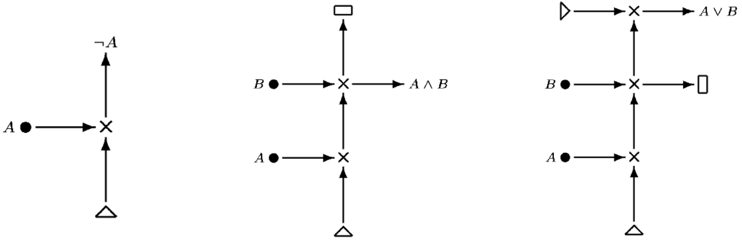
See animations at `https://en.wikipedia.org/wiki/Conway's_Game_of_Life`

Computation in the Game of Life

Game of Life elements (gliders, guns, etc...) can be used as components of a computing device



Logical operators from Game of Life

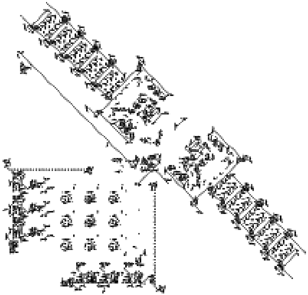


NOT AND OR

Computation in the Game of Life

Game of Life elements (gliders, guns, etc...) can be used as components of a computing device

Game of Life encoding of a Turing machine:



High-resolution image:

`https://www.conwaylife.com/w/images/4/49/Turingmachine_large.png`

Video:

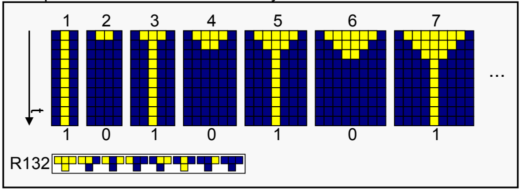
`https://www.youtube.com/watch?v=My8AsV7bA94`

Cellular Automata

Computation with CA

CA used as input-output devices. The initial state is the input. The CA should go to a quiescent state (fixed point), which is the output.

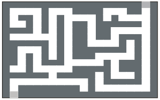
Example: Remainder after division by 2



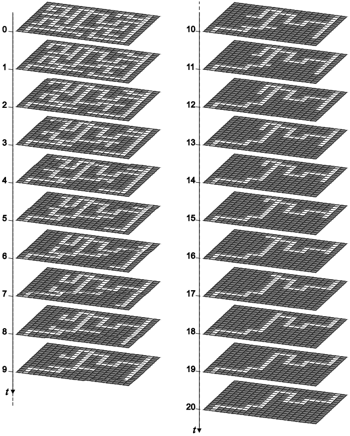
The difficulty stems from the fact that we use a local rule to evaluate a property that depends on information distributed globally.

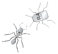
Cellular Automata

Example: CA maze solver



- Given a maze the problem consists in finding a path from the entrance to the exit.
- The conventional approach marks blind alleys sequentially
- The CA solver removes blind alleys in parallel

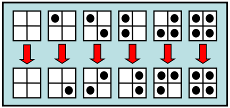




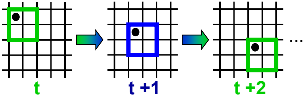
Cellular Automata

Particle CA

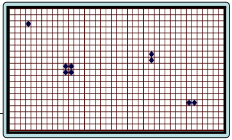
CA can be used to model phenomena that involve particles. The transition rule can be specified in terms of the motion of particles within **blocks** of two by two cells (block rules).



The automaton space is **partitioned** in non-overlapping blocks



To allow the propagation of information the position of the blocks alternates between an odd and an even partition of the space (Margolus neighborhood).



32

Cellular Automata

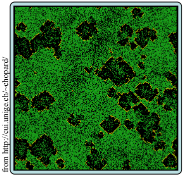
Probabilistic CA

So far we have considered only **deterministic** CA.

To model many phenomena it is useful to transition rules that depending on some externally assigned probability

Example: The **forest fire model**

- Each cell contains a green tree, a burning tree, or is empty
  - A burning tree becomes an empty cell
  - A green tree with at least a burning neighbor becomes a burning tree
  - A green tree without burning neighbors becomes a burning tree with probability *f* (probability of lightning)
  - An empty cell grows a green tree with probability *g* (probability of growth)



from http://cui.unige.ch/~chopard/

The parameters can be varied in a continuous range and introduce some “continuity” in the discrete world of CA models

Companion slides for the book *Bio-Inspired Artificial Intelligence: Theories, Methods, and Technologies* by Dario Floreano and Claudio Mattiussi, MIT Press

34

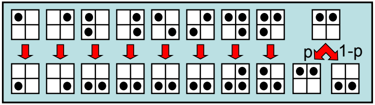
See http://cui.unige.ch/~chopard/CA/Animations/img-root.html for an animation of this model (and of many other models!)

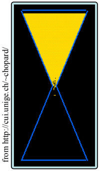
Cellular Automata

Complex Systems

Cellular systems allow the modeling and simulation of phenomena that are difficult to describe with conventional mathematical techniques

Example: The sand rule with friction





This kind of model permits the exploration of the behavior of *granular media*, which is difficult with conventional tools (e.g., PDEs)

Cellular Automata

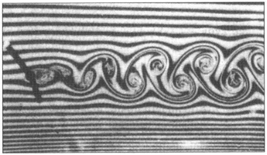


FIGURE 11. This is a picture of a vortex street disturbed by an obstacle. Wolfram, page 377.

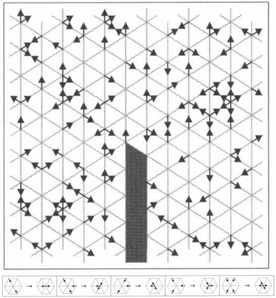


FIGURE 12. This is a picture of the liquid dynamic model at the most basic level. The fluid flow is from left to right. Also shown are the updating rules. Wolfram, page 378.

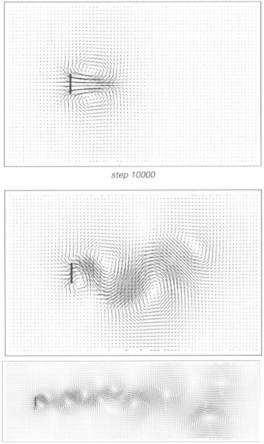


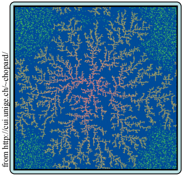
FIGURE 13. The three images above are all the same c.a., but at iterations 1,000, 4,000, and 7,000, from top to bottom. Each line vector is an average velocity vector of a 20X20 cell black. The vectors enter from the left in a regular way with a frequency that represents 0.4 of maximum speed. Wolfram, page 380.

Cellular Automata

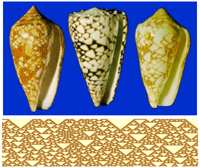
Structures and Patterns

One of the most fascinating aspects of biological and natural systems is the emergence of complex spatial and temporal structures and patterns from simple physical laws and interactions.

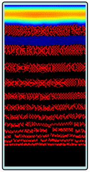
Cellular systems are an ideal tool for the analysis of the hypotheses about the local mechanisms of structure and pattern formation.

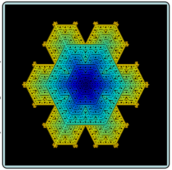


from http://cui.unige.ch/~chopard/



from http://www.btinternet.com





from http://cui.unige.ch/~chopard/

Companion slides for the book *Bio-Inspired Artificial Intelligence: Theories, Methods, and Technologies* by Dario Floreano and Claudio Mattiussi, MIT Press 36