Computer Simulations

A practical approach to simulation

Semra Gündüç

gunduc@ankara.edu.tr

Ankara University Faculty of Engineering,

Department of Computer Engineering

Cellular Automata

Cellular Automata

Introduction to Cellular Automata

- Cellular automata provide a fascinating class of dynamical systems based on very simple rules of evolution yet capable of displaying highly complex behavior.
- S. Wolfram, “A New Kind of Science”, (Wolfram Research Europe 2002) https:www.wolframscience.com
- S. Wolfram, editor, Theory and Applications of Cellular Automata (World Scientific, Singapore, 1986).
- T. Toffoli and N. Margolus, Cellular Automata Machines (MIT Press, 1987).

Cellular Automata

Introduction to Cellular Automata

- Cellular automaton refers to the dynamics of a collection of cells.
- Each one of these cells can be in one of the finite set of states.
- The time evolution is discrete,
- At each step, the new value for a cell depends only on the current state of itself and its neighbors.

Cellular Automata

Introduction to Cellular Automata

- These systems are fascinating in that deceptively simple rules can give rise to extremely complex behavior.
- Even though the formulation of a cellular automaton may seem almost trivial, there are a huge number of possible rules.
- Furthermore, slight changes in the rules can dramatically change their behavior.

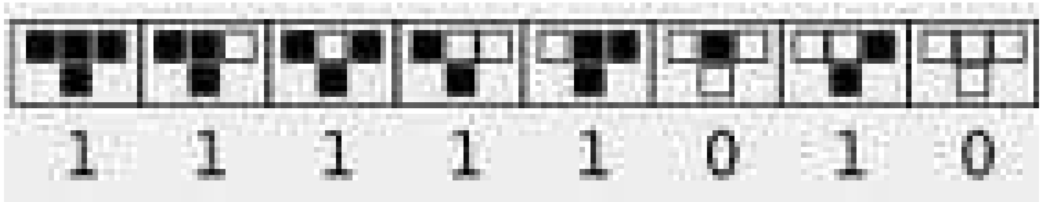
Cellular Automata

Introduction to Cellular Automata Example:

- Consider one dimensional models where each cell can take only one of two possible states.
- Each cell has two neighbours,
- The evolution of a given cell to the next time step depends only on the current values of the cell and each of its two neighbors
- In this case there are $2^3 = 8$ possible arrangements for the cell and its neighbors.
- Since the cell can take only two values 0 or 1 ("dead" or "alive") for each one of these 8 possible arrangements, $2^8 = 256$ rules exist even for this most simple model.

Cellular Automata

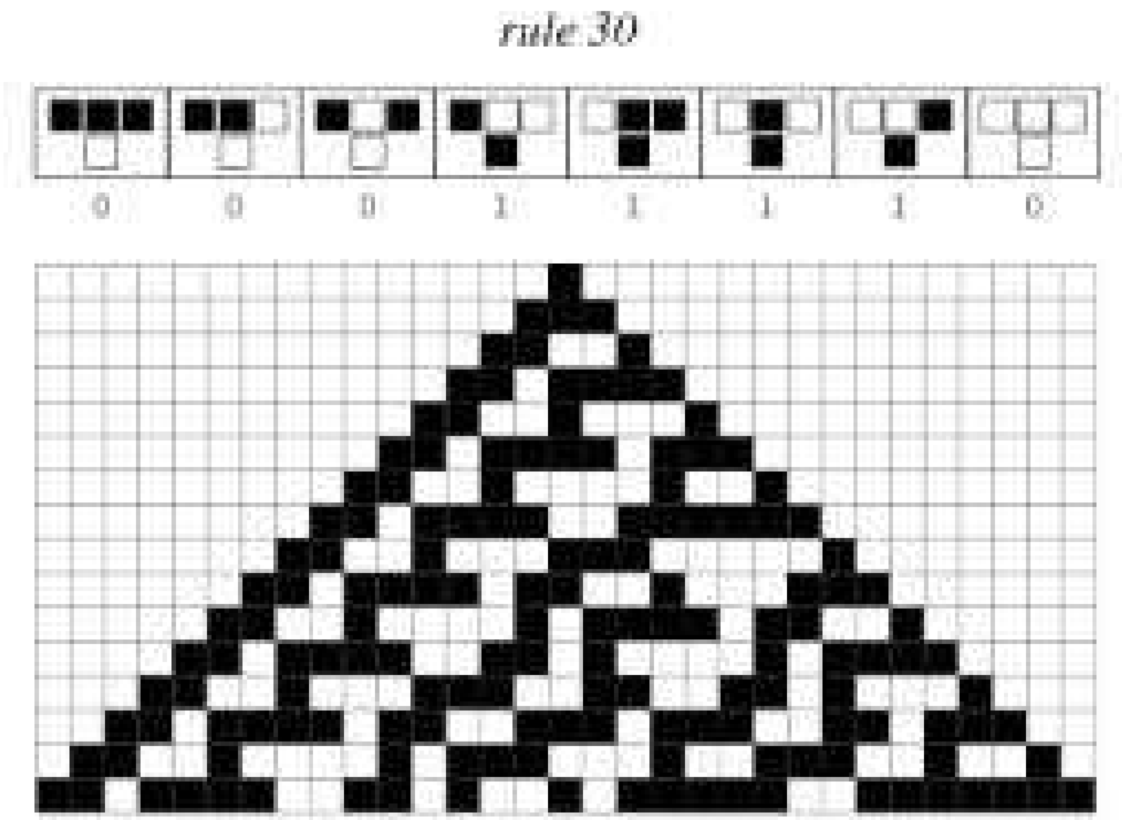
Introduction to Cellular Automata



Rule 150 = $1 \times 2^7 + 1 \times 2^6 + 1 \times 2^5 + 1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 0 \times 2^0$

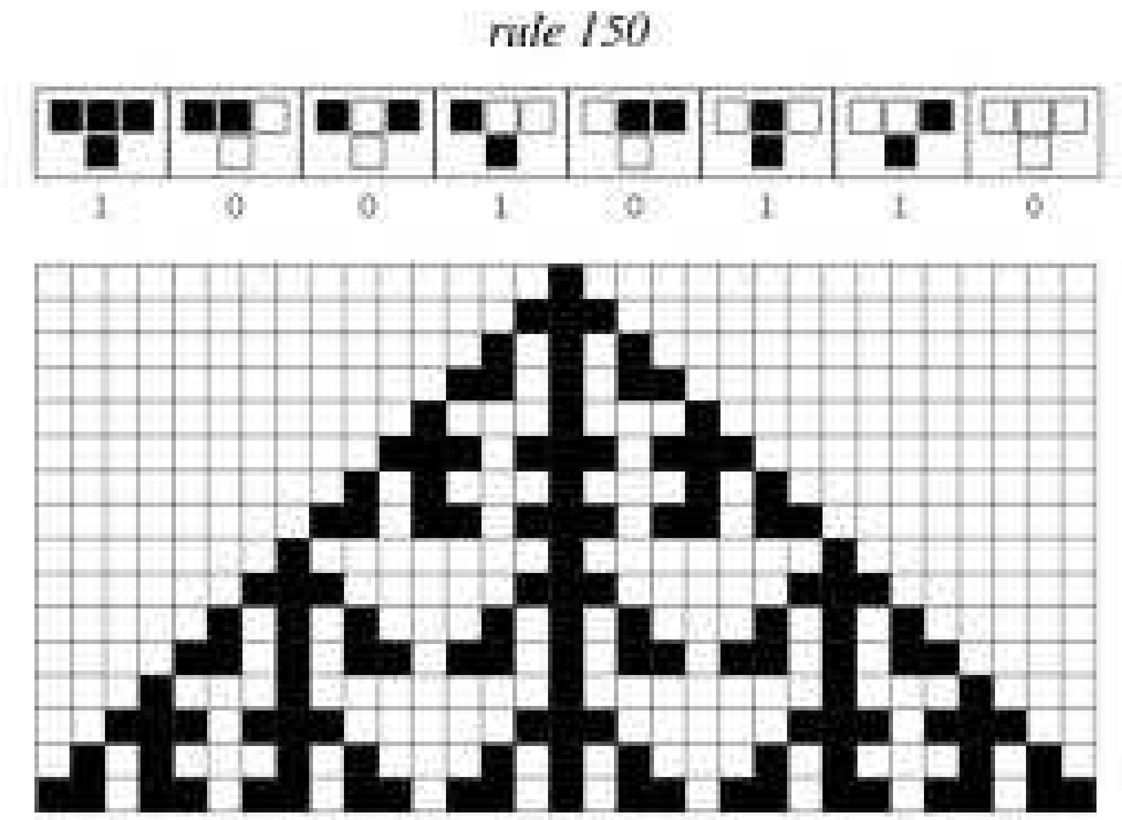
Cellular Automata

Introduction to Cellular Automata



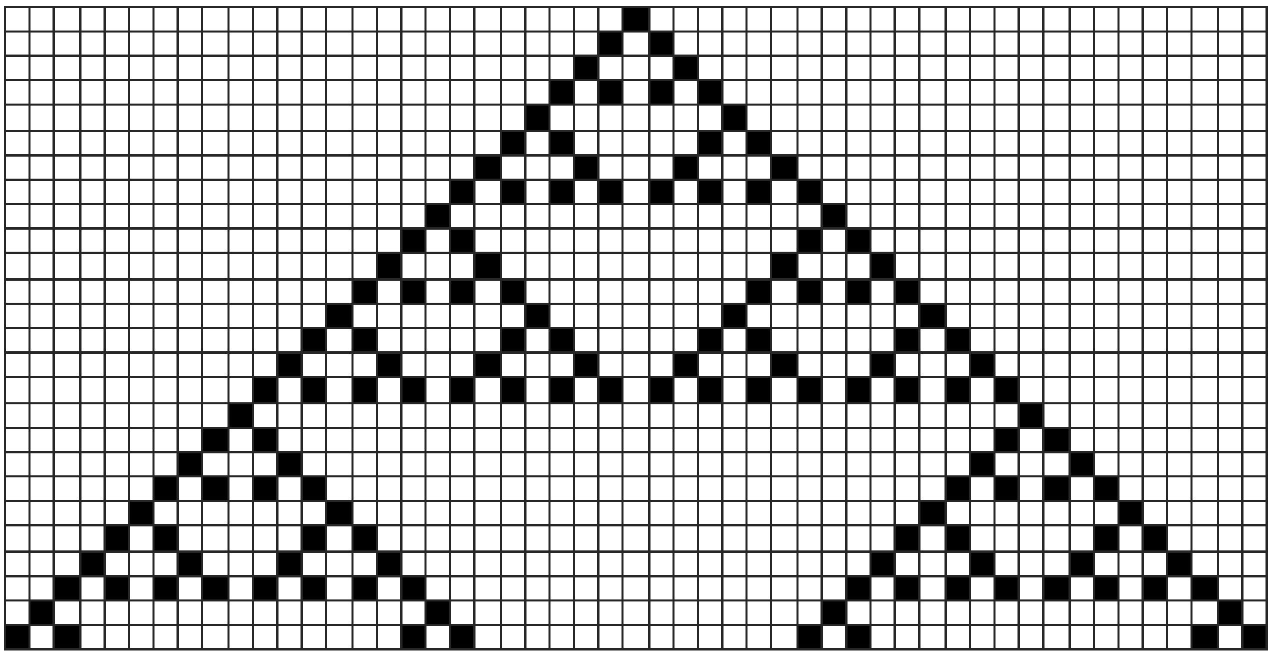
Cellular Automata

Introduction to Cellular Automata



Cellular Automata

Introduction to Cellular Automata

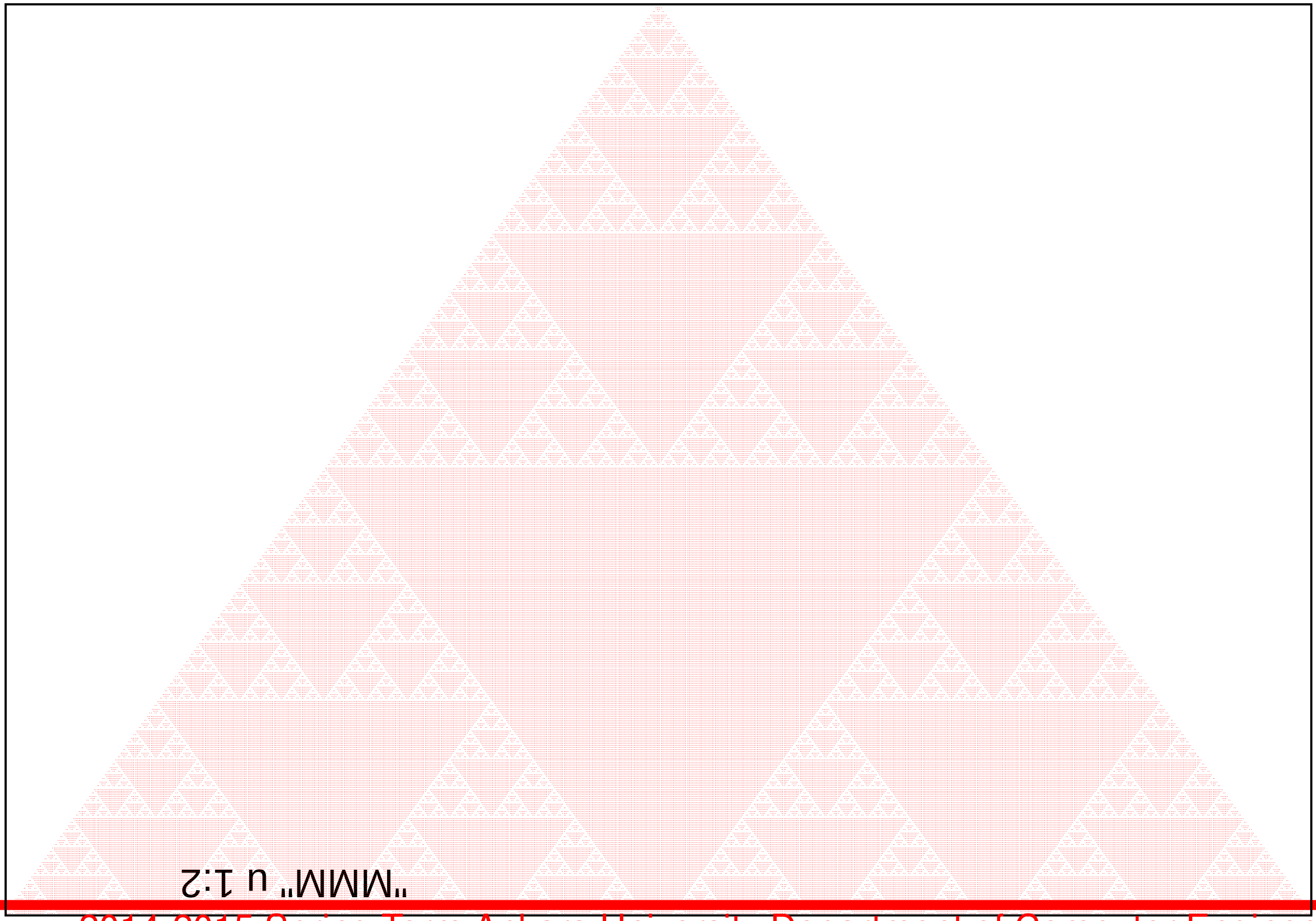


Rule 90 :

$0 \to 0; 1 \to 1; 2 \to 0; 3 \to 1; 4 \to 1; 5 \to 0; 6 \to 0; 7 \to 0$

Cellular Automata

Introduction to Cellular Automata



Cellular Automata

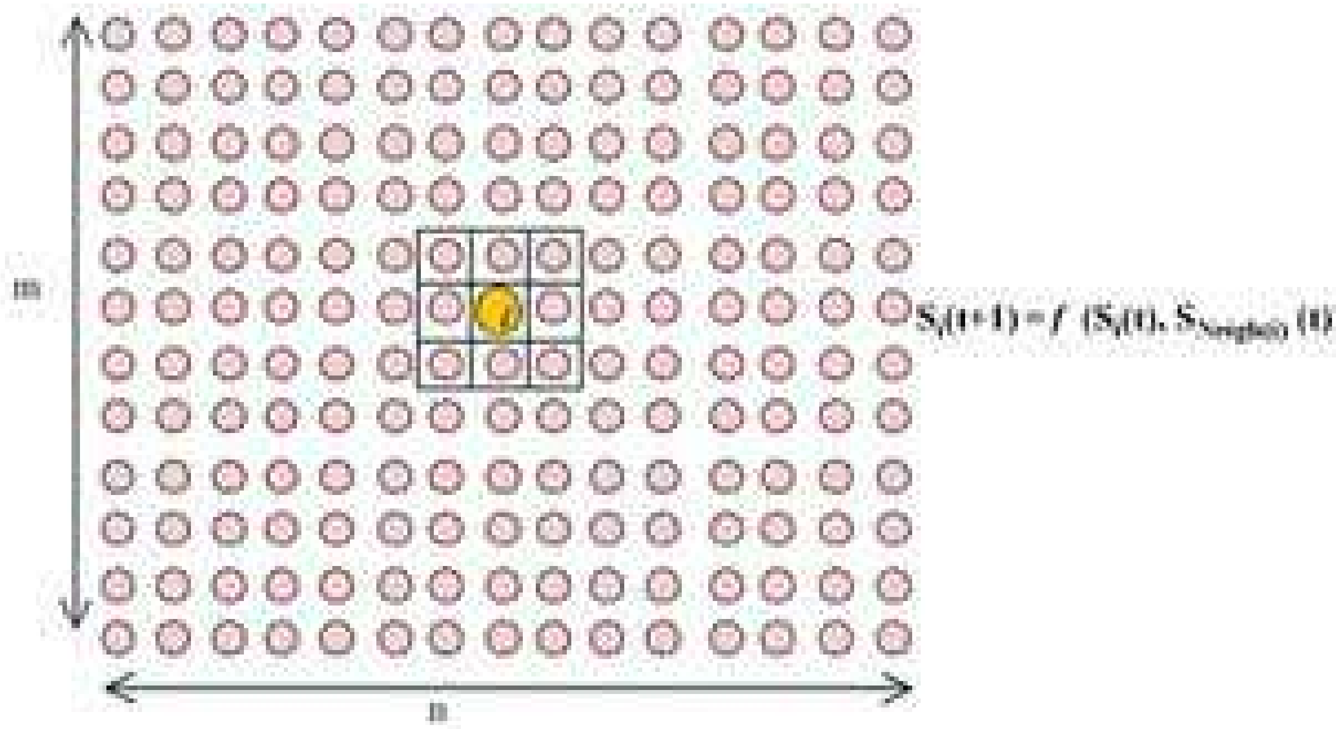
Introduction to Cellular Automata

Example:

- If a two dimensional model is assumed, then the number of cells are 9,
- considering cells with only two states possibe arrengements are: $2^9 = 512$
- A a general rule needs to specify the next state of the cell for each of these arrangements. This gives $2^{512} = 1.3 \times 10^{154}$ possible rules.
- compare this with the approximate age of the universe ($\sim 4 \times 10^{17}$ seconds)

Cellular Automata

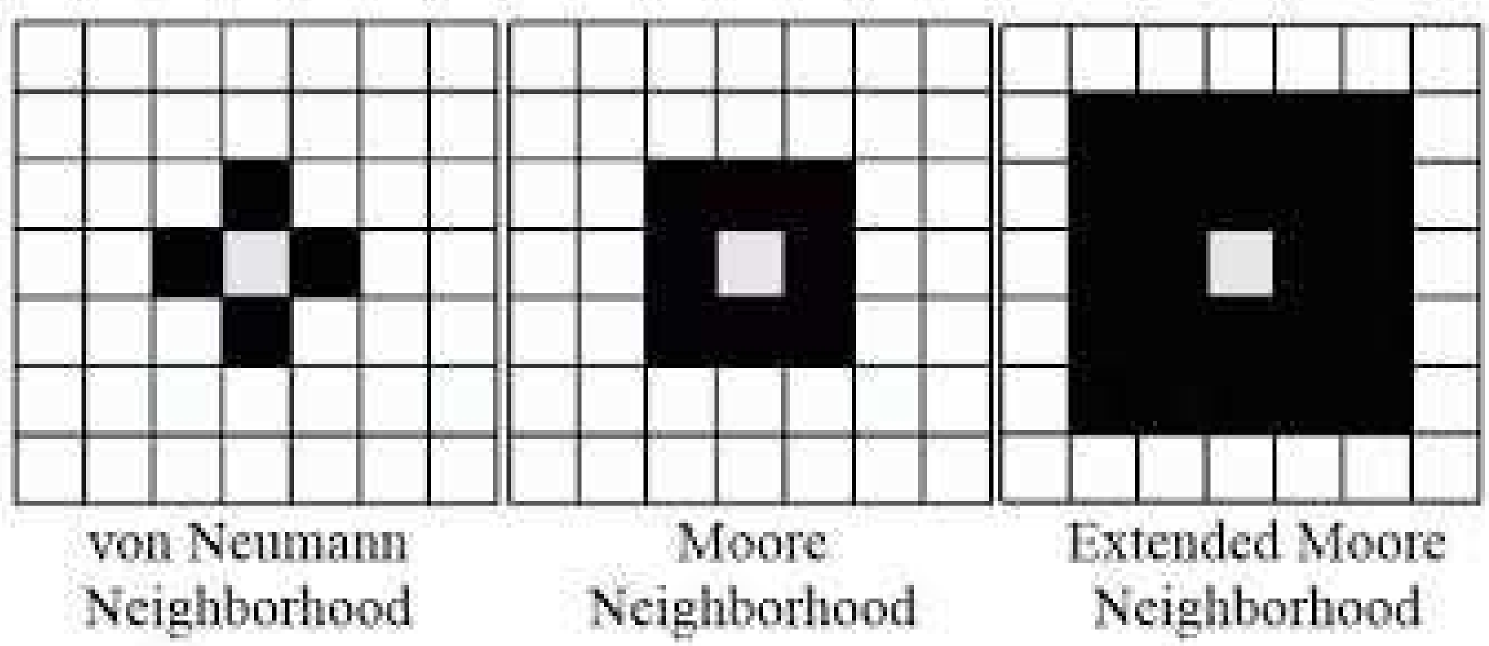
Introduction to Cellular Automata



Eight neighbours of the cell in two dimension.

Cellular Automata

Introduction to Cellular Automata



Cellular Automata

Introduction to Cellular Automata

- If it is restricted, next state of the cell depends on the total of only the four nearest neighbors, then there are $2^{10} = 1024$ possible rules to consider.
- With a discrete set of states, cellular automata have the appealing feature of being easily implementable entirely by logical operations, the natural functions of computer circuitry
- Also, the state of several cells can be stored and manipulated within a single computer word

Cellular Automata

Introduction to Cellular Automata

- Using such tricks, these models can often be implemented to run extremely fast.
- Such models may supply simulation methods as good as or better than the conventional use of floating point fields on a discrete grid.
- With this motivation, considerable attention has been paid to cellular automata.
- Another advantage of this approach is the ability to work with arbitrary boundary conditions.

Cellular Automata

Conway's Life

- Perhaps the most famous cellular automaton model is Conway's "Game of Life"
- E. Berlekamp, J. Conway, and R. Guy, Winning Ways for your Mathematical Plays, volume 2, (Academic Press, ISBN 0-12-091152-3, 1982) chapter 25.

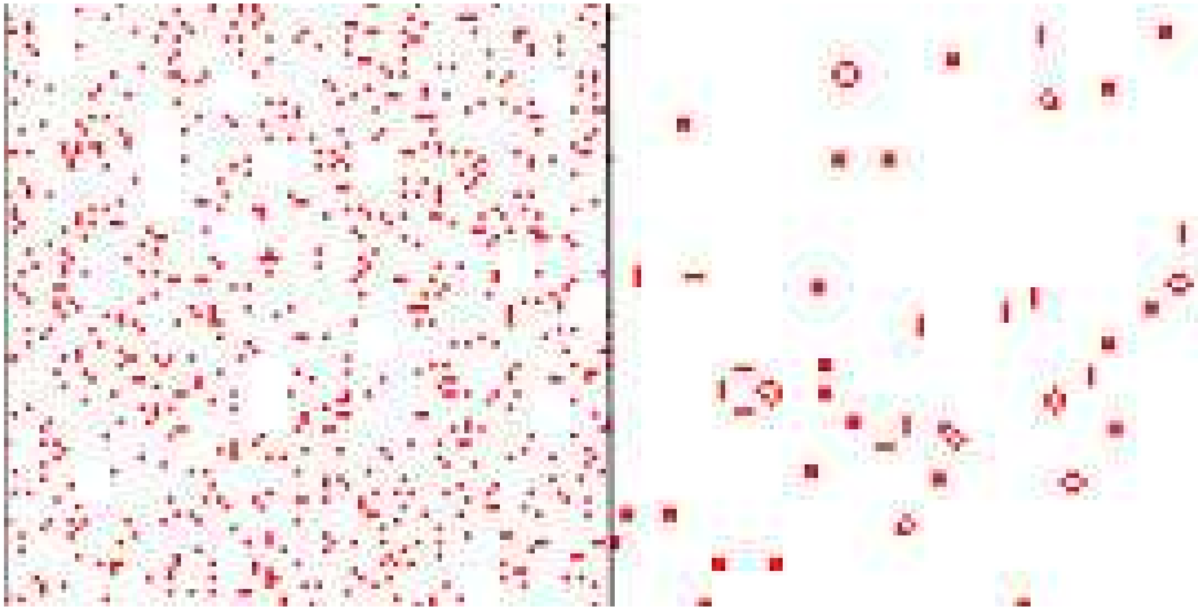
Cellular Automata

Conway's Life

- The rule involves the eight cell neighborhood,
- if a cell is initially "dead" it becomes alive
  - if and only if it has exactly three live neighbors,
  - or "parents."
- A living cell dies
  - of loneliness if it has less than two live neighbors,
  - and of overcrowding if it has more than three live neighbors.
- Only in the case of exactly two or three live neighbors does it survive.

Cellular Automata

Introduction to Cellular Automata



Living structures starting from a random initial state.

Cellular Automata

Conway's Life

Project:

- A particularly amusing local configuration has five live cells (1's at the coordinates given below);
- Starting with coordinates $\{(0, 0), (0, 1), (0, 2), (1, 2), (2, 1)\}$.
- After four time steps this configuration returns to its original shape, but displaced by $(-1, 1)$.
- On an otherwise empty board, this "glider" continues to propagate as a single entity.

Cellular Automata

Conway's Life

- A large collection of fascinating life configurations can be found in the Wikipedia
- See Wikipedia, ”Conway's Game of Life,” http://en.wikipedia.org/wiki/Conway's life

Cellular Automata

Conway's Life

- Gliders allow information to be propagated over long distances,
- It has been proven that with a complicated enough initial configuration, one can construct a computer out of live cells on a life board.
- Special subconfigurations form the analog of electronic gates, which can control beams of gliders representing bits.
- Indeed, since life is capable of universal computation, one might imagine a life board programmed to simulate the game of life

Cellular Automata

Fredkin's modulo-two rule

- An extremely simple but highly amusing rule takes at each time step the "exclusive or" (XOR) operation between a site and its neighbors.
- This rule has the remarkable property of self replication.
- M. Gardner, Wheels, Life, and Other Mathematical Amusements (W. H. Freeman and Company, New York, 1983).

Cellular Automata

Fredkin's modulo-two rule

- Starting with any given initial pattern, after 2^n time steps copies of the original state occupy positions separated by 2^n spatial sites from the original in every direction as specified in the chosen neighborhood.

Cellular Automata

Introduction to Cellular Automata

Forest fire simulation- A cellular automata application

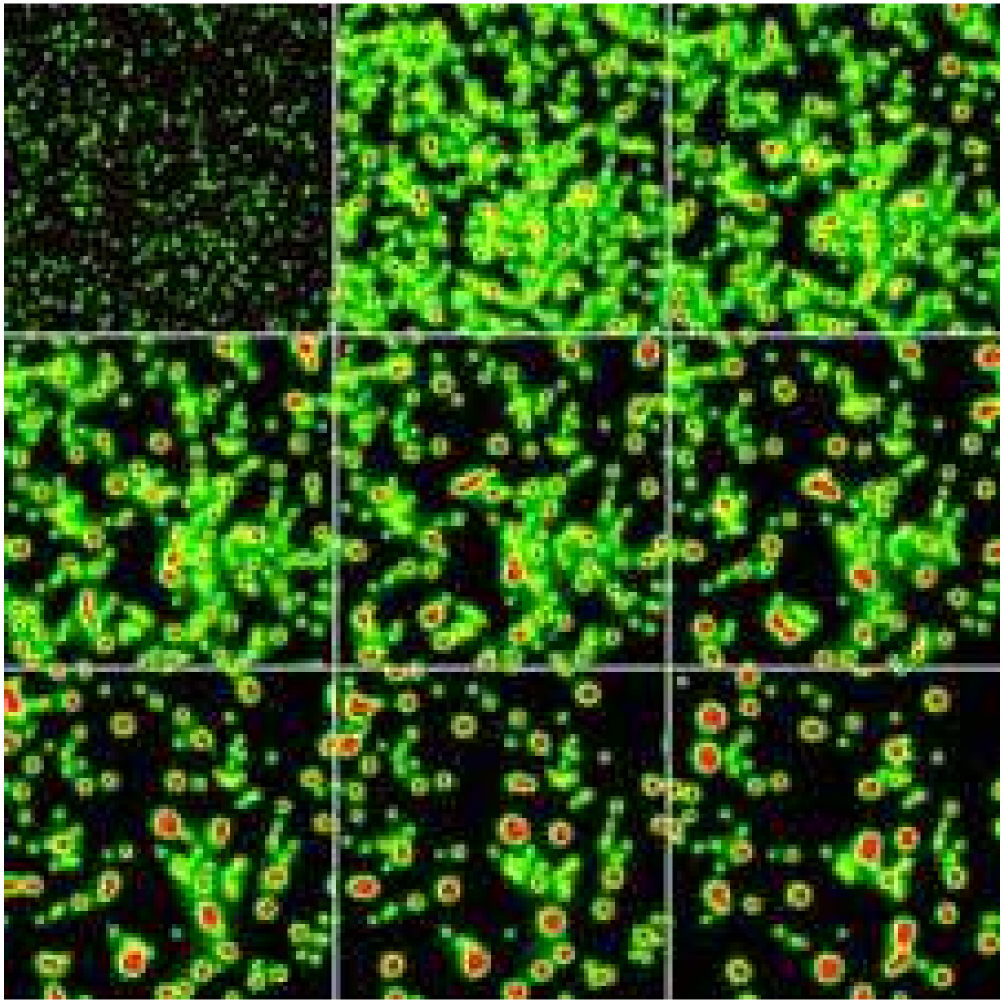
Cellular Automata

Forest Fires

- An amusing model of forest fires:
- Each cell has three possible states, (empty(0), a tree (1), or a fire (2)).
- On a finite system, one should,
  - randomly fill the lattice with empth spaces and trees.
  - light a fire somewhere to get the system started.
- For the updating step,
  - Any empty site can have a tree born with a small probability.
  - Any existing fire spreads to neighboring trees leaving its own cell empty.

Cellular Automata

Introduction to Cellular Automata



Forest fire simulation- Different phases of fire

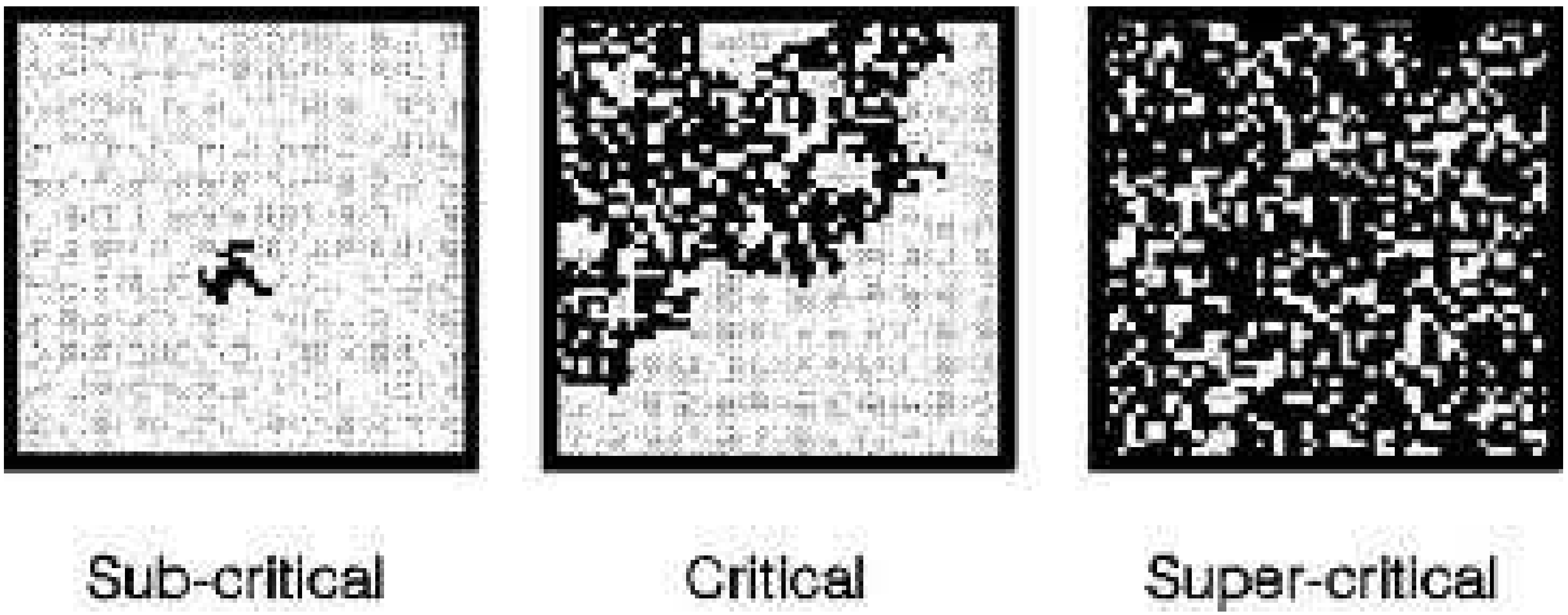
Cellular Automata

Forest Fires

- The rule here differs from those discussed previously in having a stochastic nature.
- As the system is made larger, the growth rate for the trees should decrease to just enough to keep the fires going.
- If too many trees grow, one obtains a large fire reducing their density, while if there are too few trees, fires die out.

Cellular Automata

Forest Fires



Forest fire simulation- Criticality

Cellular Automata

Forest Fires

- As the system becomes larger, the growth rate for the trees can be reduced without the fire expiring.
- In a steady state the system has fire fronts continually passing through the system,
- Perhaps there is a moral here that one should be careful about extinguishing all fires in the real world, for this may enhance the possibility for a catastrophic uncontrollable fire.

Cellular Automata

Sand Pile Model

Cellular Automata

Sand Pile Model

- For a review of this and several related models, see
- S. Clar, B. Drossel, and F. Schwabl, J. Phys.:Cond. Mat. 8, 6803 (1996).

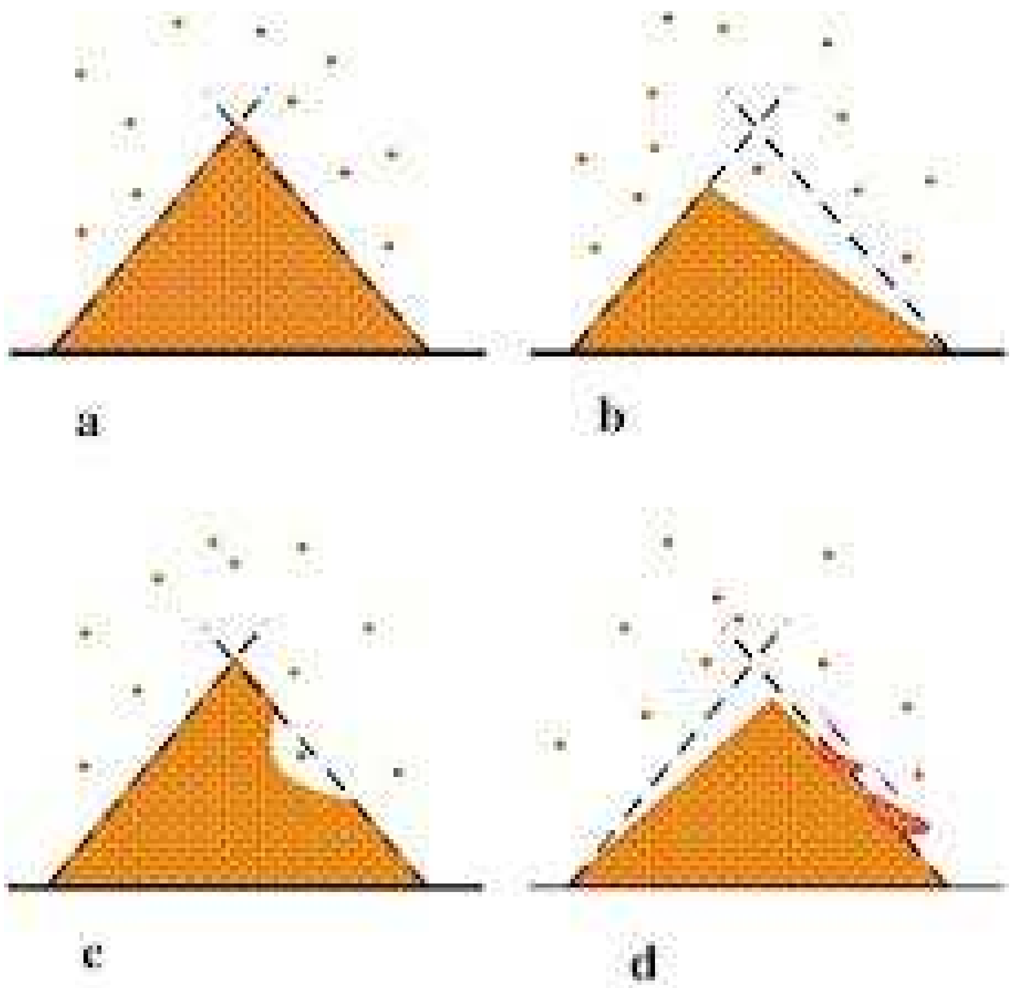
Cellular Automata

Sand Pile Model The phenomenon of sandpile

- On slowly adding grains of sand to an empty table, a pile will grow until its slope becomes critical and avalanches start spilling over the sides.
- If the slope becomes too large, a large catastrophic avalanche is likely, and the slope will reduce.
- If the slope is too small, then the sand will accumulate to make the pile steeper.
- Ultimately one should obtain avalanches of all sizes, with the prediction of the size for the next avalanche being impossible to determine without actually running the experiment.

Cellular Automata

Sand Pile Model



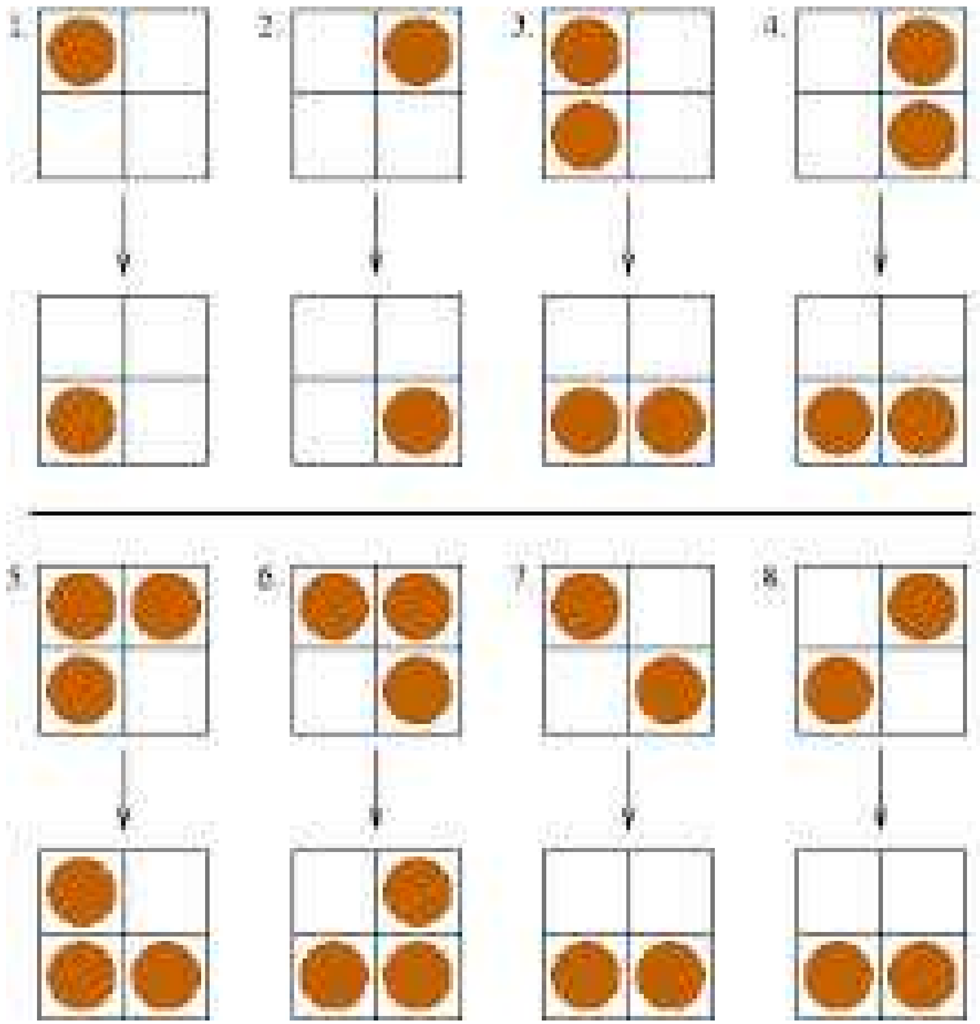
Cellular Automata

Sand Pile Model

- The original Bak, Tang, Wiesenfeld paper [1] presented a particularly simple model to mimic the sandpile idea.
- For this, each site of a two dimensional lattice has a state represented by a positive integer z_i.
- This integer can be thought of as representing the amount of sand at that location, or inanother sense it represents the slope of the sandpile at that point.
- P. Bak, C. Tang, and K. Wiesenfeld, Phys. Rev. Lett. 59, 381 (1987); Phys. Rev. A38, 3645 (1988).

Cellular Automata

Sand Pile Model



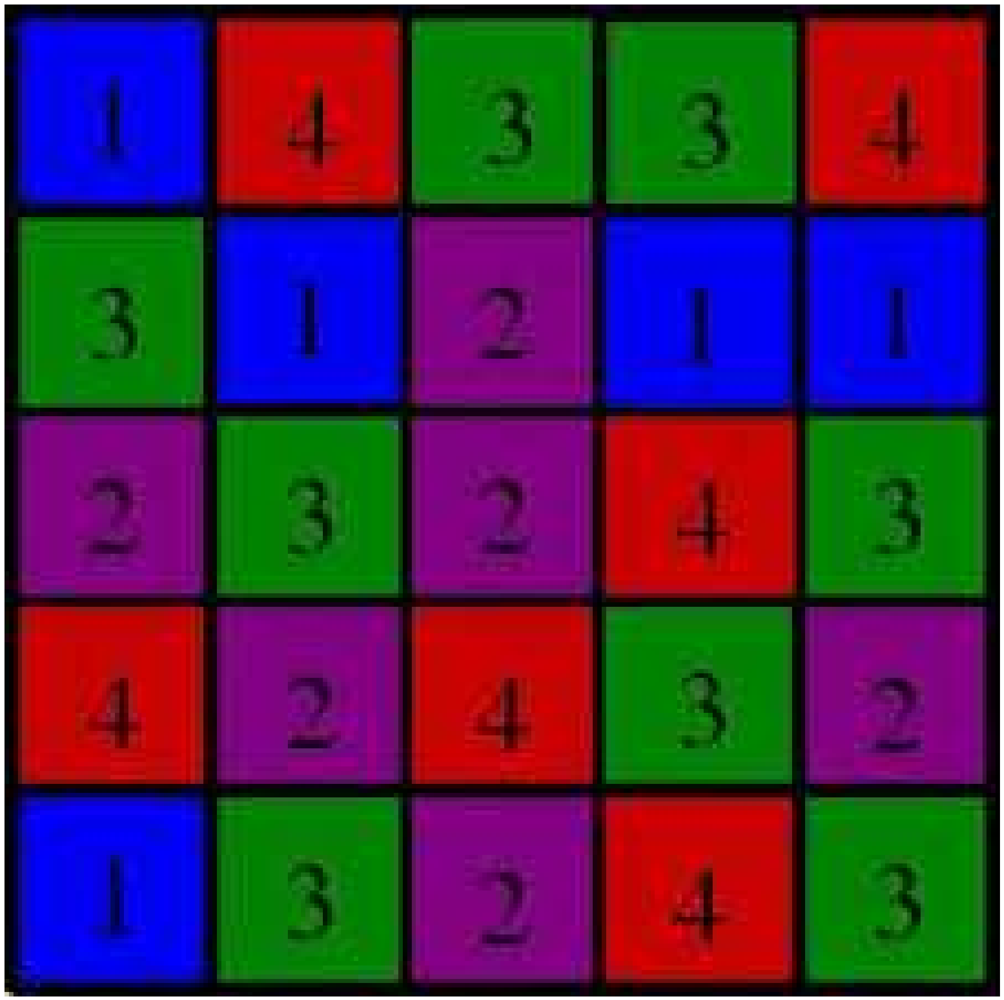
Cellular Automata

Sand Pile Model

- The dynamics follows by setting a threshold z_T above which any given zi is unstable.
- Time now proceeds in discrete steps. In one such step each unstable site with $zi \geq z_T$ “tumbles” or “topples”, dropping by z_T and adding one grain to each of its four nearest neighbors.
- This may produce other unstable sites, and thus an avalanche can ensue.
- This proceeds for further time steps until all sites are stable.

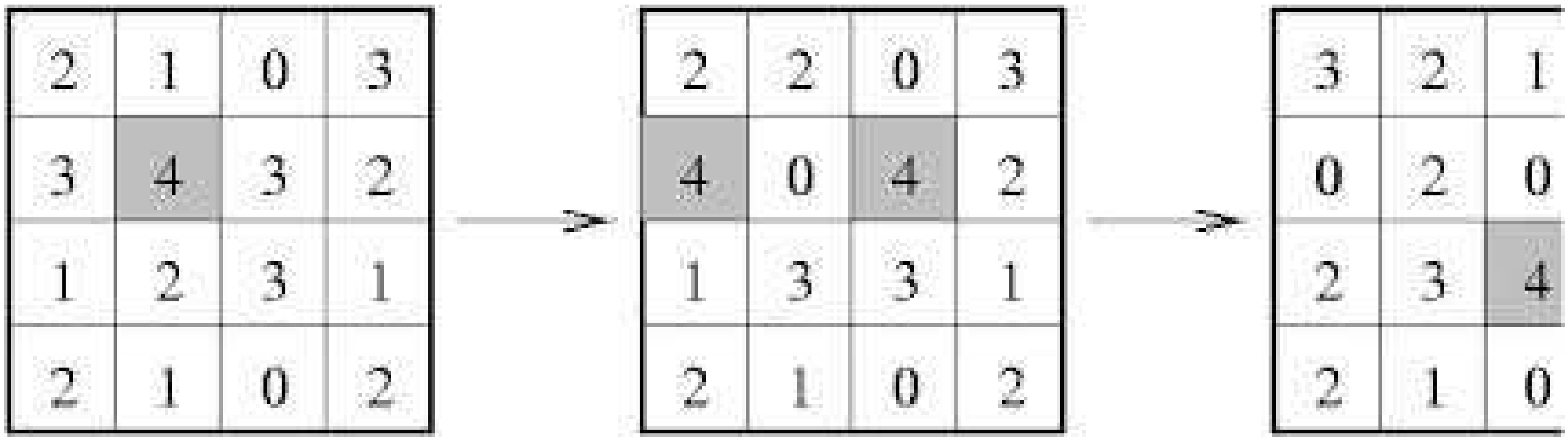
Cellular Automata

Sand Pile Model



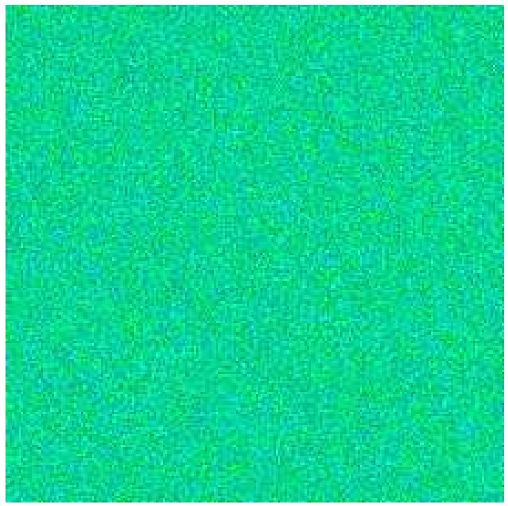
Cellular Automata

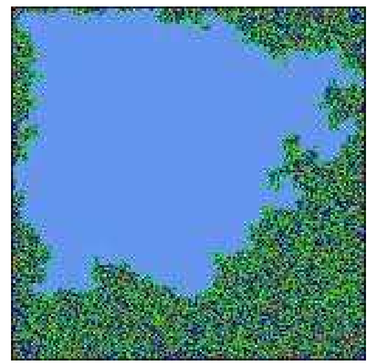
Sand Pile Model



Cellular Automata

Sand Pile Model





Cellular Automata

Sand Pile Model

A natural experiment consists of,

- adding a grain of sand to a random site
- measuring the number of topplings and the number of time steps for the resulting avalanche.
- Repeating this many times to gain statistics,
- the distribution of avalanche sizes and lengths displays a power law behavior, with all sizes appearing