Cellular Automata in random starting states (Ch. 6) and

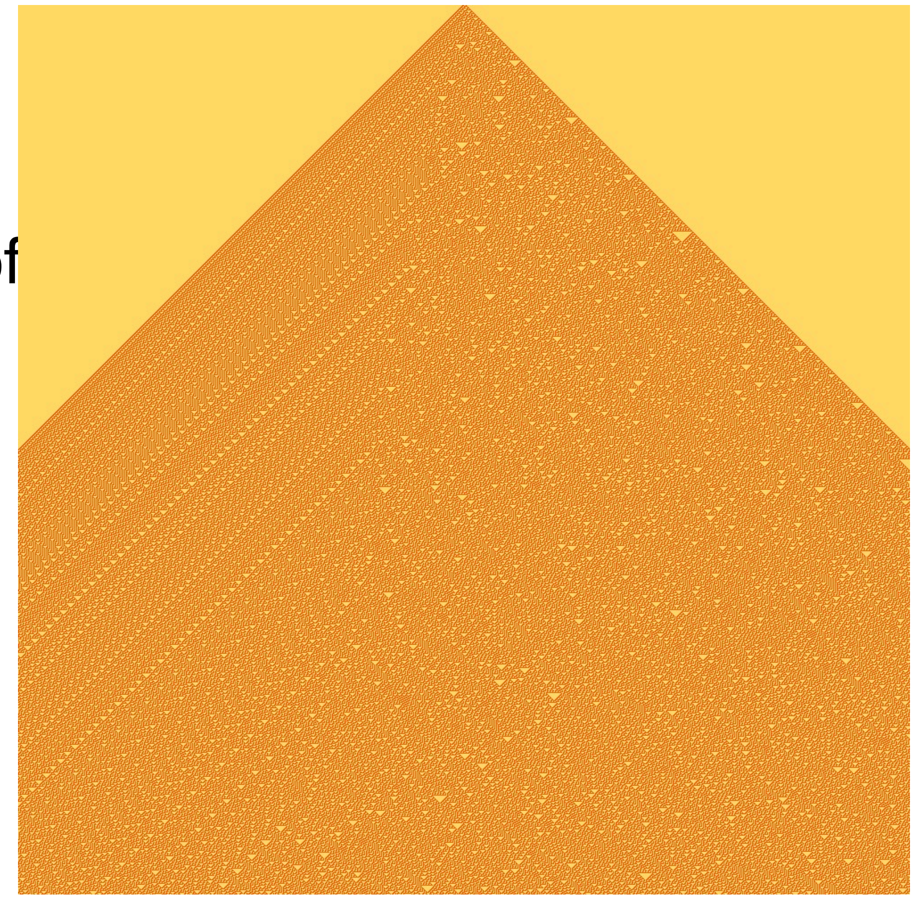
Modeling "Everyday systems" with Cellular Automata (Ch 8)

S Wolfram. (2002) *A New Kind of Science*

Presenter: Richard Tillett

When we last read about 1D CA

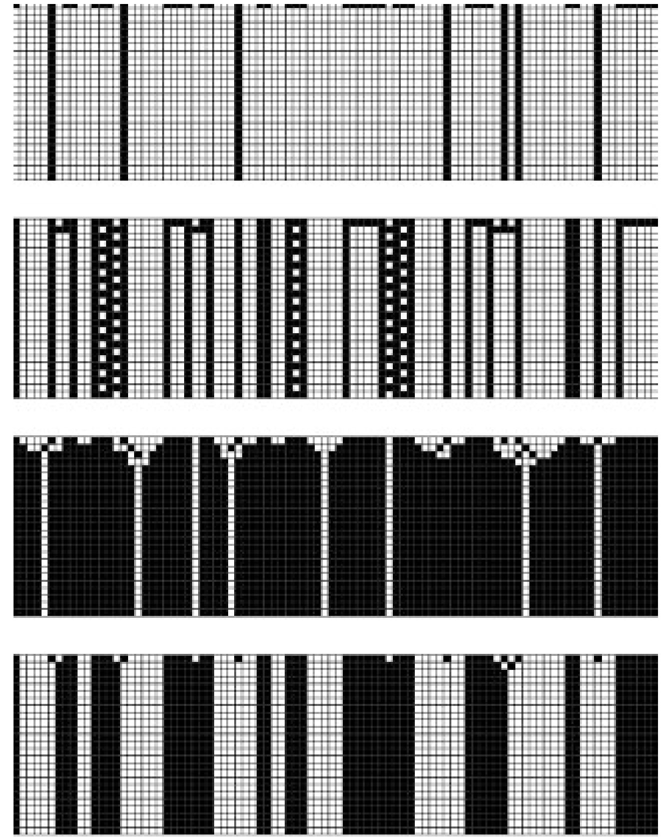
- Line 1 as a single black square for two-color CA
- Also, totalistic CA
  - Rules concern averages of neighbor cells
  - 2187 unique totalistic CA rules



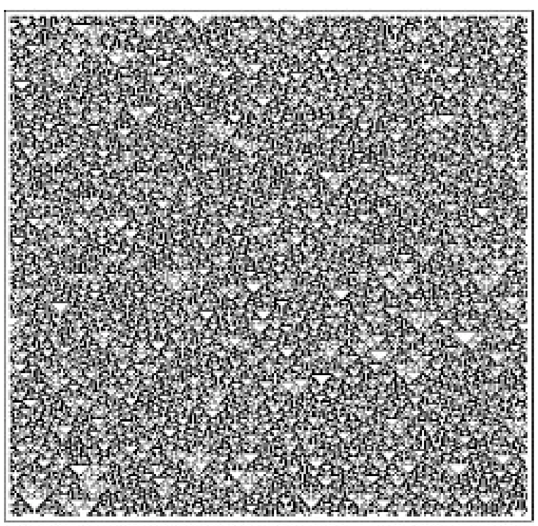
Rule 30

From random initial conditions

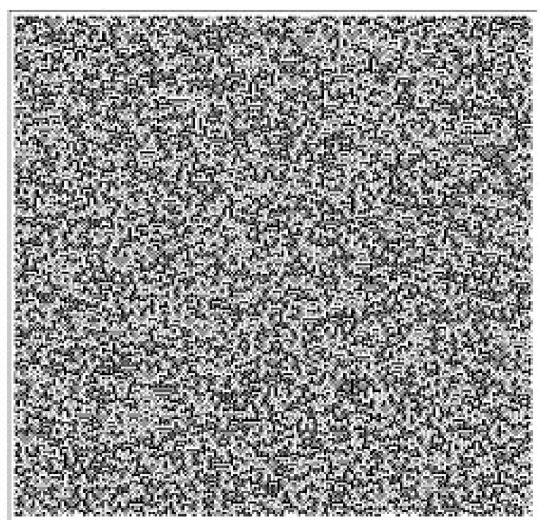
- Starting from randomness
  - Even with random start states, many CA will organize (right)



  - Others will not (below)



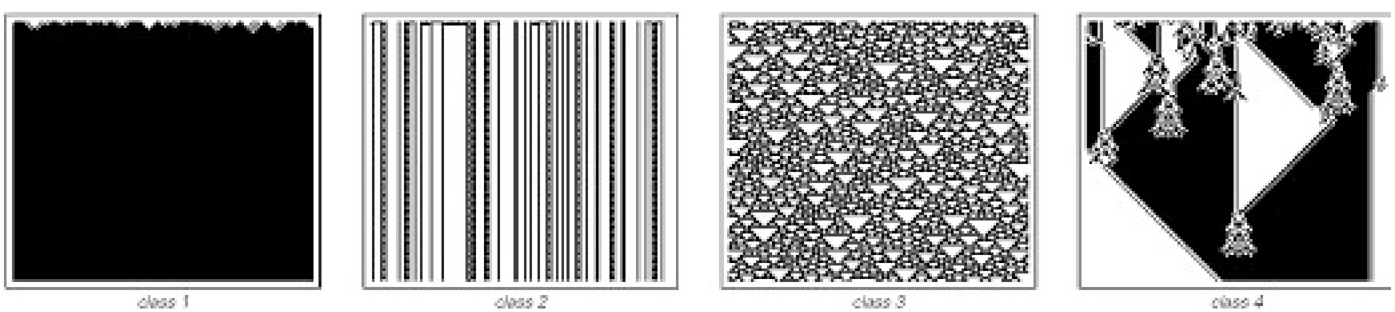
rule 90



rule 105

Four broad classes “discovered”

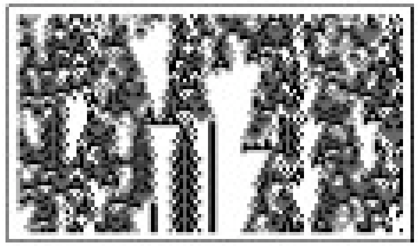
- Informatively, these classes are named by sequential integers
  - 1: Solid state from all or almost all cond.
  - 2: Converge to stable or short-period repeat forms (where the “vertical lines” distribute will vary by Line 1’s)
  - 3: “[In] many respects random, though triangles & other such small-scale structures are essentially always at some level seen” (Huh?)
  - 4: A mixture of order and randomness (as opposed to 3…)
    - Simple local structures that move and interact

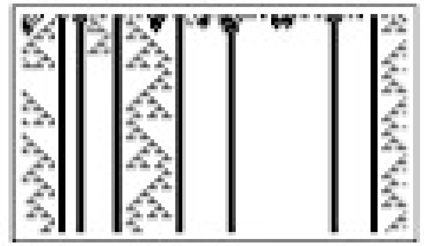


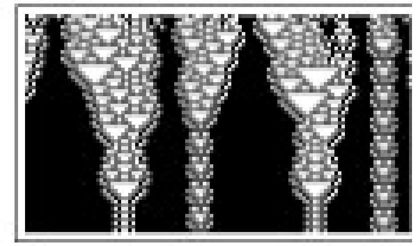
1 2 3 4

“Rare” borderline automata


code 219


code 438


code 1380


code 1632

- The above are totalistic nearest-neighbor 3-color CA
- “[W]ith almost any general classification scheme there are inevitably borderline cases which get assigned to one class by one definition and another class by another definition”
  - If definitions for each class aren’t mutually exclusive or have 2 or more criteria, is it really still a general classification scheme?
  - Why wouldn’t one replace this ARBITRARY scheme with phylogenic-tree clustering, Eisen clustering, local and/or global similarity scores, statistics estimating degree of order, or ANYTHING BUT OUT OF A HAT. n=2187 is all

Start condition sensitivity

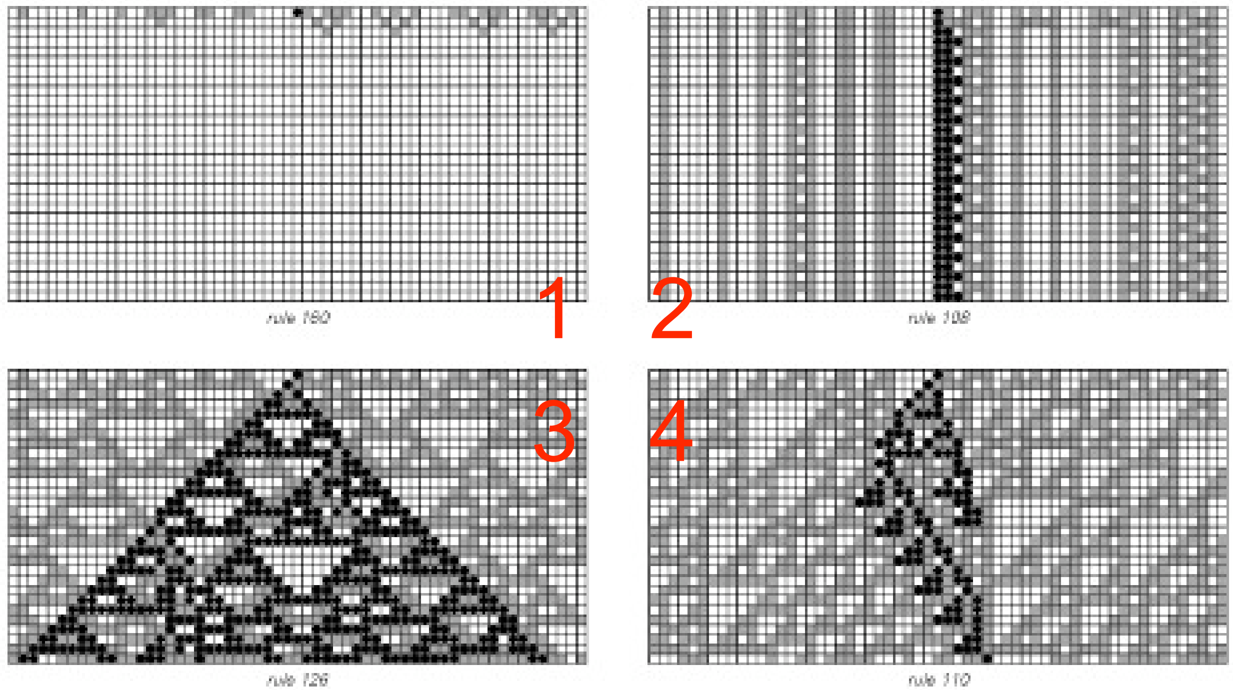
<u>Sensitivity reveals how each type handles information</u>

1=insensitive (information beyond their rules is irrelevant)

2= new mildly-different local change (local "interactions" only)

3=systemic propagation (long-range propagation of even the smallest differences)

4= sporadic ("intermediate," can go either *transient-local* or *wide*



The effect of changing the color of a single cell in the initial conditions for typical cellular automata from each of the four classes identified in the previous section. The black dots indicate all the cells that change. The way that such changes behave is characteristically different for each of the four classes of systems.

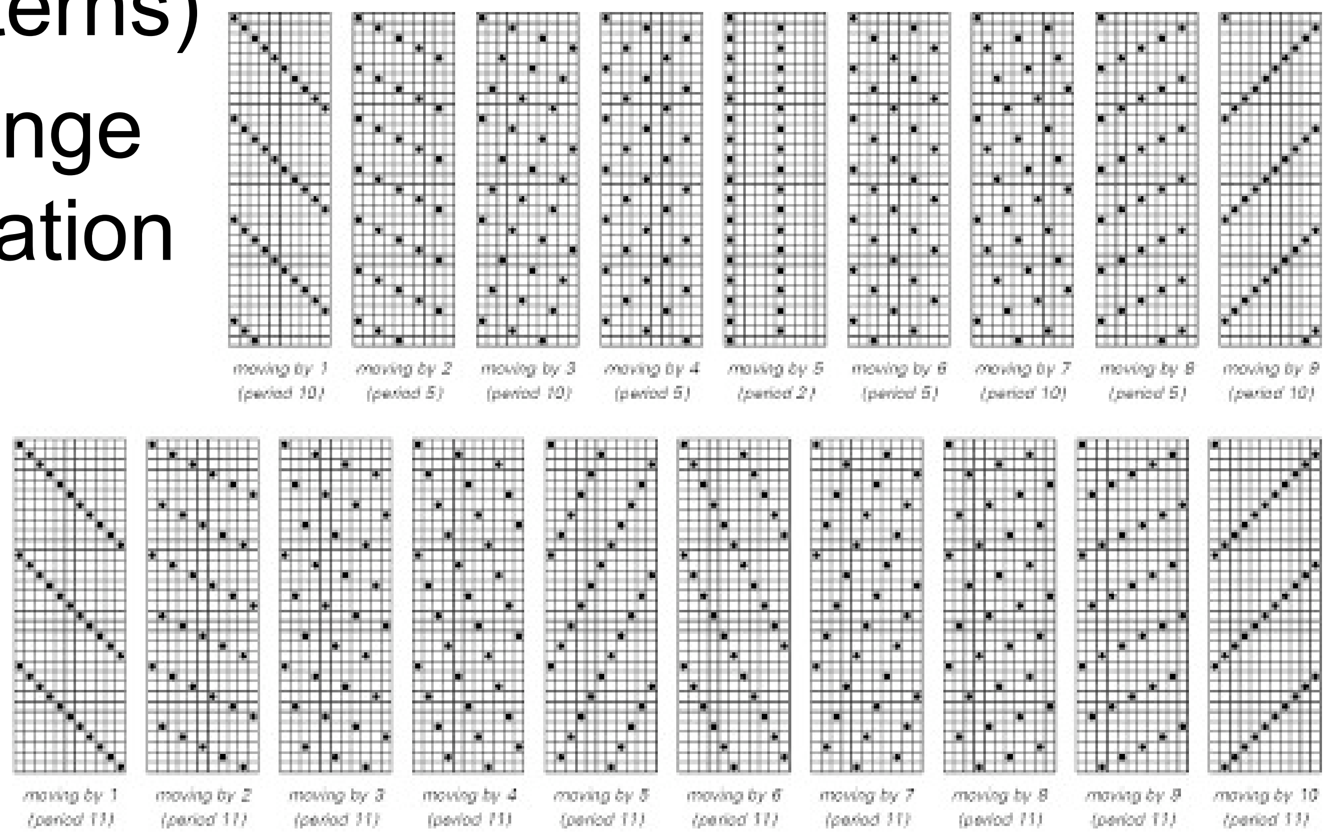
A single cell (Line 1, black dot) is changed in init. conds

Black dots= all changed cells

Class 2 systems as systems of limited size

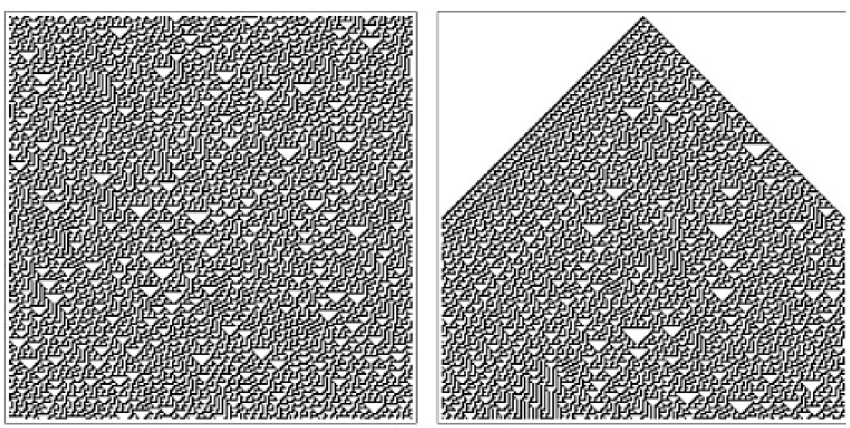
- Eventually repetitive (recall: stripes or short period patterns)
- No long-range communication
- Acting just like these

1 dot / line
Dot moves *n* spaces in each next line
Period is dependent on *n* and size
Max(period)=11 here



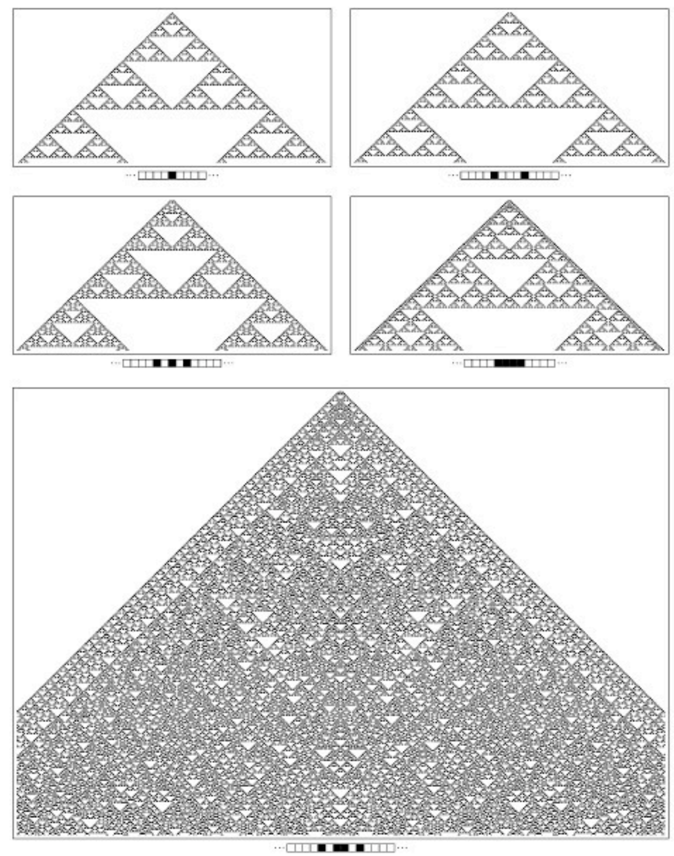
“Randomness” in class 3 CA

- Degrees of “randomness”
  - **Rule 30**: “random” for ordered starts too (but “random” in a similar way? How “random” is that?)



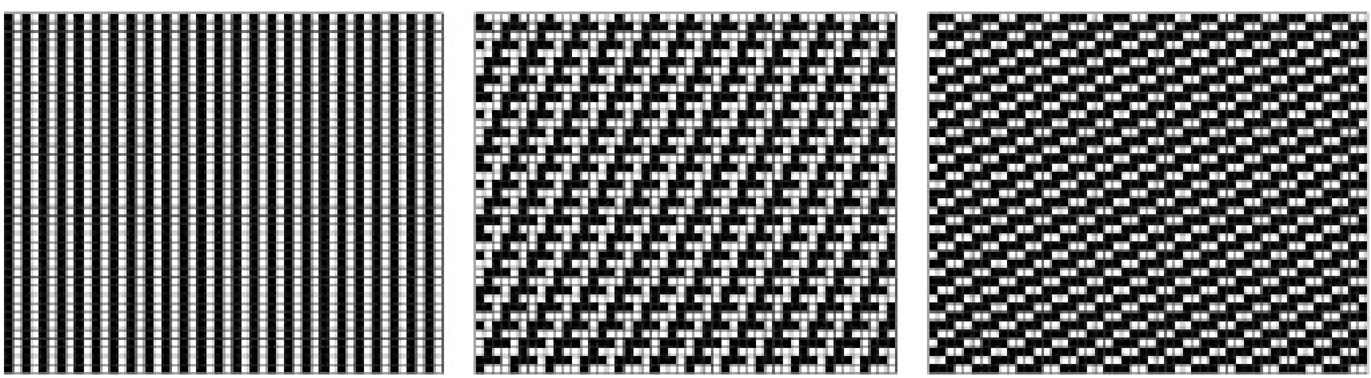
Comparison of the patterns produced by the rule 30 cellular automaton starting from random initial conditions and from simple initial conditions involving just a single black cell. Away from the edge of the second picture, the patterns look remarkably similar.

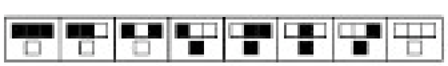
- **Rule 22**:
  - can go nested
  - Can also be re-perturbed into “random”



Special starting conditions

- Ex. **Rule 30** has handful of conditions inducing order
- Repeated blocks in line 1 function like systems of limited size
  - Periods are Rule- and block-dependant (obvious?)…also

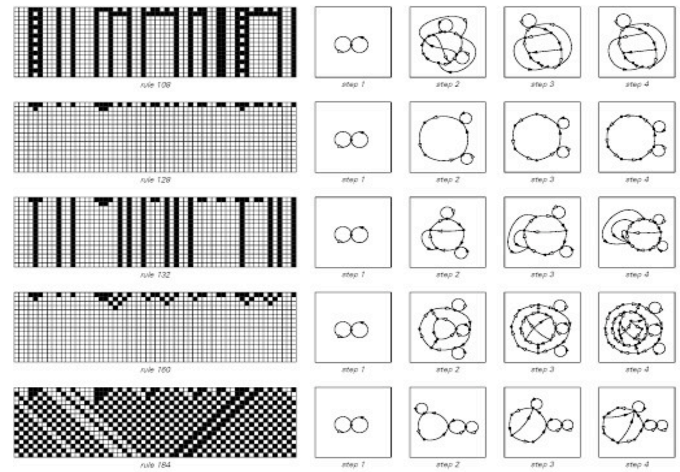




Examples of special initial conditions that make the rule 30 cellular automaton yield simple repetitive behavior. Small patches with the same structures as shown here can be seen embedded in typical random patterns produced by rule 30. At left is a representation of rule 30. Finding initial conditions that make cellular automata yield behavior with certain repetition periods is closely related to the problem of satisfying constraints discussed on page 210.

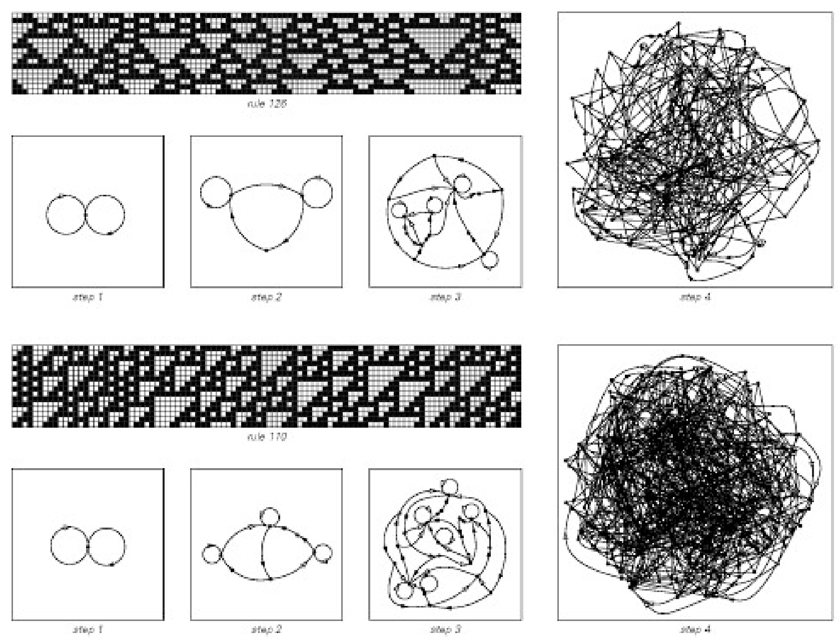
Randomly-initiated CA as Attractors

- Line 1 has 2^n possible configurations
- *most* CA
  - The number of possible Line states <u>shrinks</u> with successive iterations
  - **Rephrase**: consider the potential "degeneracy" among some subset of the 2^n line 1 states, "coding" for the same line 2, forming basins of attraction for their common product, many lines



Networks representing possible sequences of black and white cells that can occur at successive steps in the evolution of several class 1 and 2 cellular automata. These networks never have more than about t^2 nodes after t steps.

Examples: Class 1&2

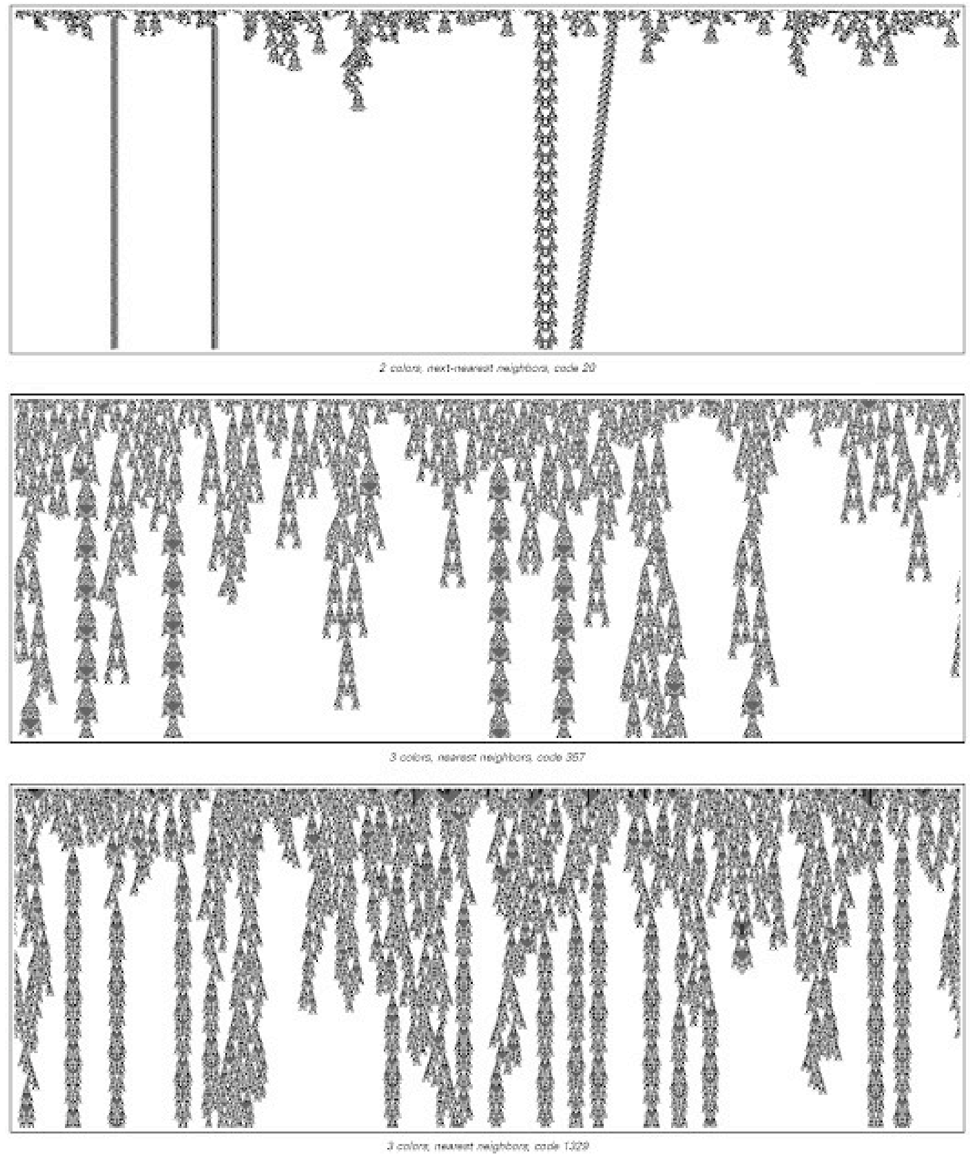


Networks representing possible sequences of black and white cells that can occur at successive steps in the evolution of typical class 3 and 4 cellular automata. The number of nodes in these networks seems to increase at a rate that is at least exponential.

Examples: Class 3&4

Structures in Class 4

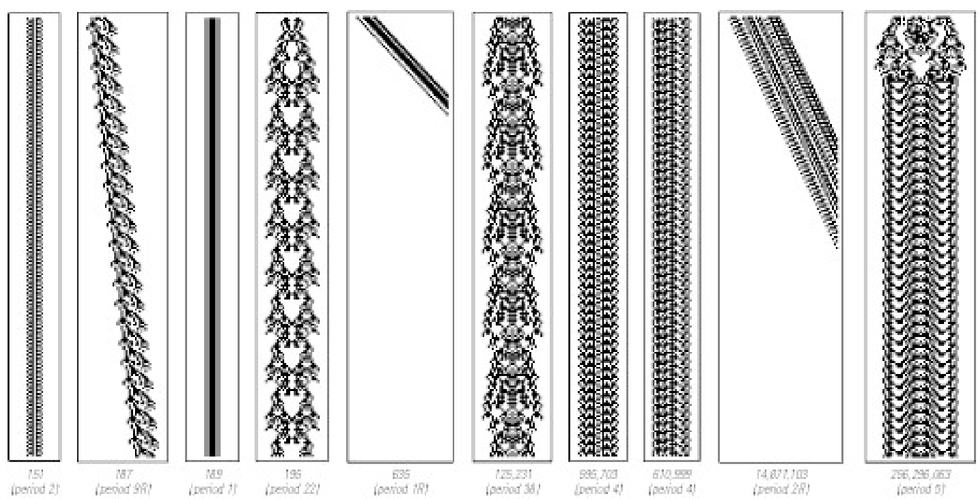
- Some persistent structures



Three typical examples of class 4 cellular automata. In each case various kinds of persistent structures are seen.

Structures in Class 4

- Persistent structure can be hard to come by in class 4
  - Complete search for Code 20 from 2.5 x 10^{10} possible setups finds only 10.



Persistent structures found by testing the first twenty-five billion possible initial conditions for the code 20 cellular automaton shown on the previous page. Note that reflected versions of the structures shown are also possible. The base 2 digit sequences of the numbers given correspond to the initial conditions in each case, as on the previous page.

Chapter 6 Conclusions

- 4 broad classes of rules are described for randomly-initiated CA, by structure features, with few borderline cases

- Most CA progressively shrink possible line states: can be described as attractor networks

Ch 8: Modeling “things” with CA

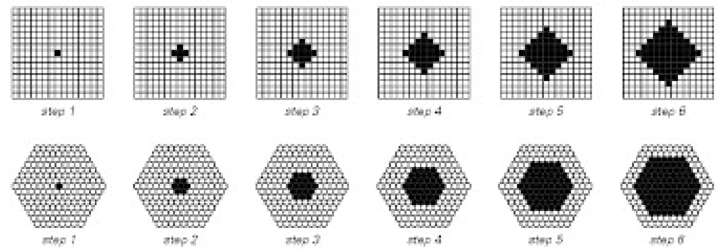
- Outline
  - On modeling
  - Crystal and snowflake models
  - Fractures and breaking of materials
  - Fluid flow
  - Claims regarding evolution and CA
  - Plant growth
  - Animal growth
  - Finance

On modeling

- Are models as made in the current mode flawed?
- Must a model share characteristics with the observed phenomena?

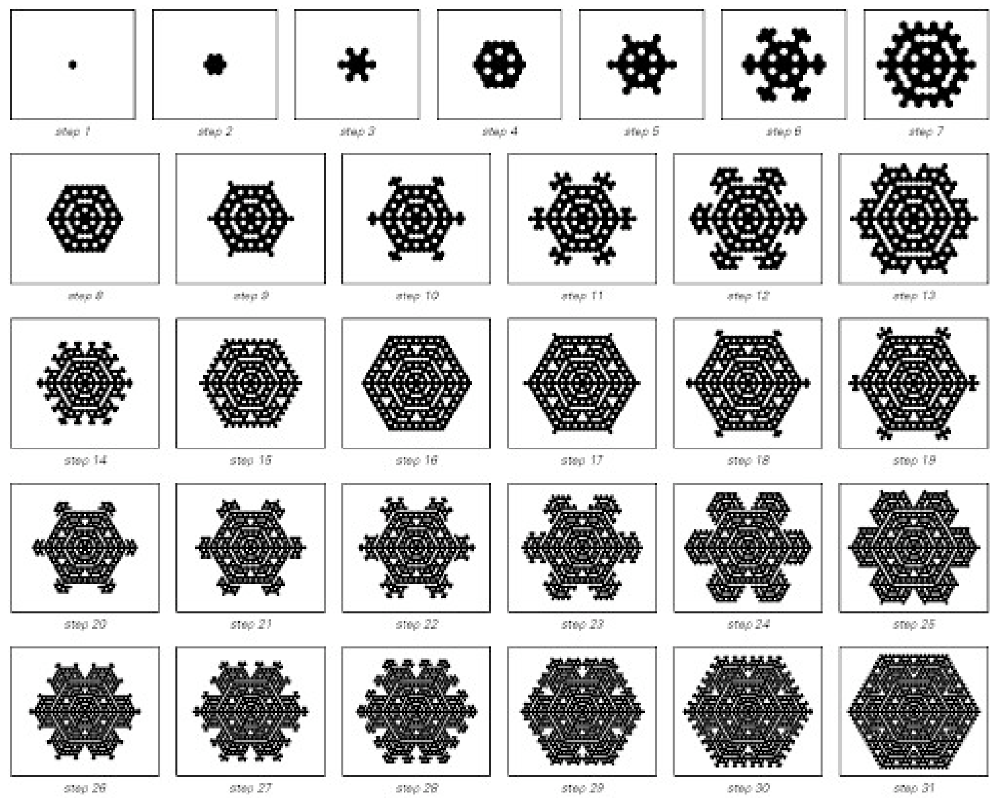
Crystals and snowflakes

- Crystal CA
  – Assume hexagonal cell arrangements
  – Program: cell -> black if any neighbor = black
  – Insert black cell seed…



Cellular automata with rules that specify that a cell should become black if any of its neighbors are already black. The patterns produced have a simple faceted form that reflects directly the structure of the underlying lattice of cells.

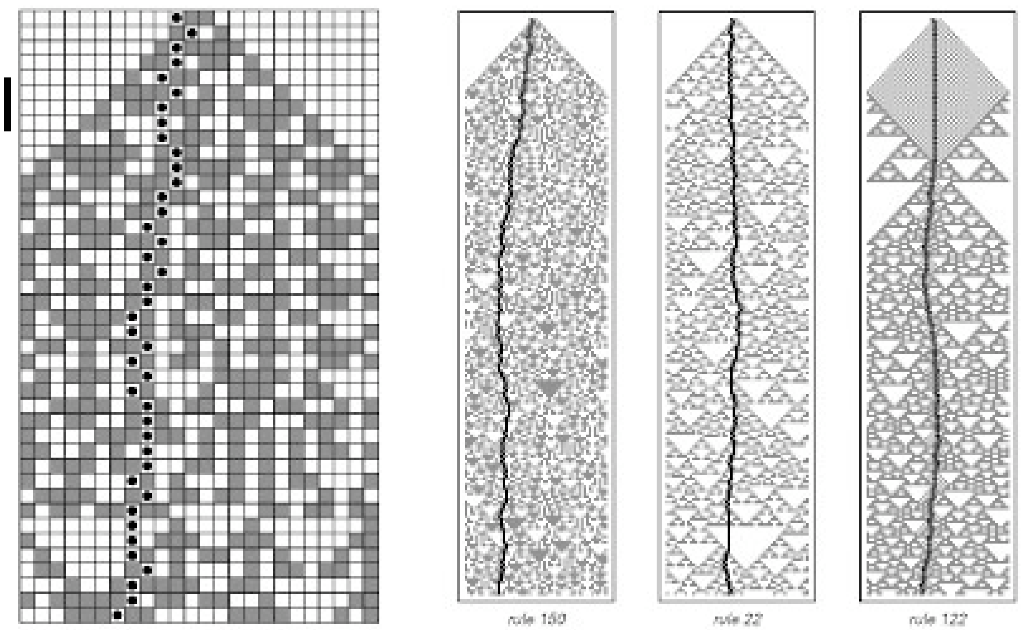
- Snowflake CA
  – Enthalpy of fusion modeled by
  – cell -> black if exactly 1 neighbor was black in the previous time-increment



Breaks and Fractures

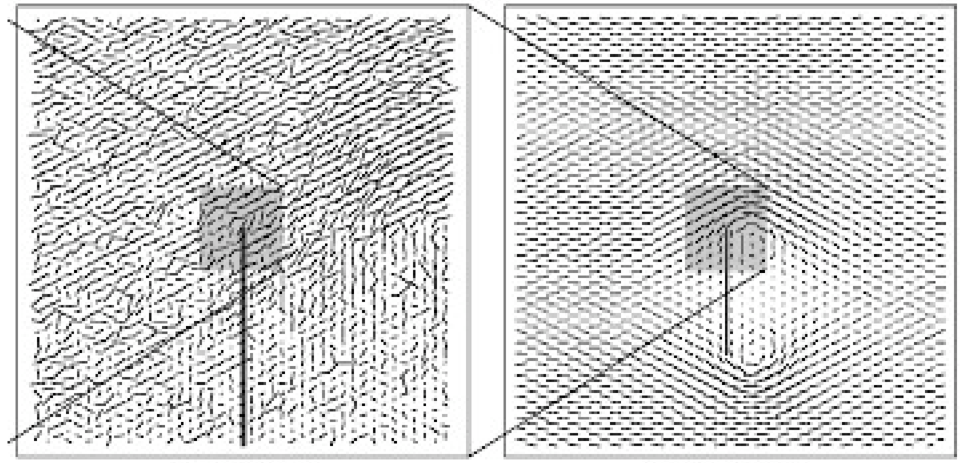
- Fracture propagation is conserved for
  – Scaleless?: geological events and small objects share gross pattern
  – Conserved among wide range of solids and composition?

CA model: at each step, CA-rules used to update cells, where the black dot, as leading crack point follows displacements as they "shake."



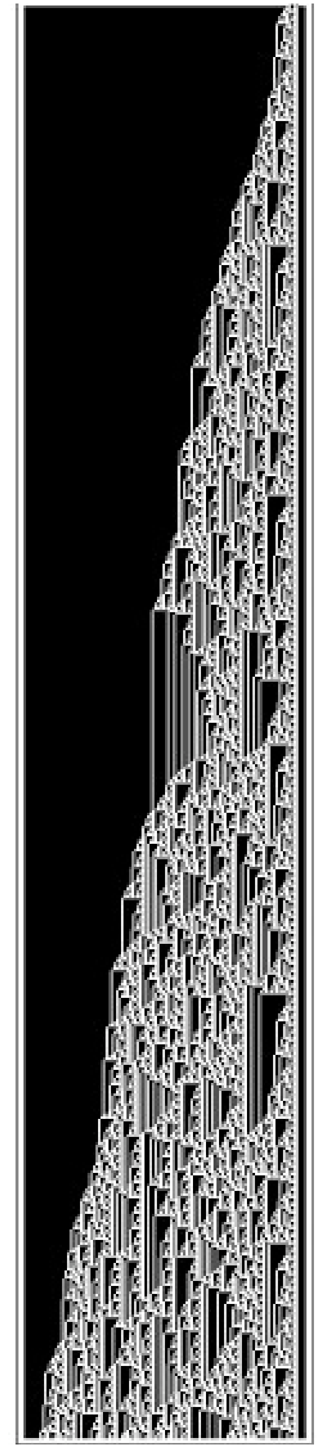
A very simple cellular automaton model for fracture. At each step, the color of each cell, which roughly represents the displacement of an element of the solid, is updated according to a cellular automaton rule. The black dot, representing the location of a crack, moves from one cell to another based on the displacements of neighboring cells, at each step setting the cell it reaches to be white. Even though no randomness is inserted from outside, the paths of the cracks that emerge from this model nevertheless appear to a large extent random. There is some evidence from physical experiments that dislocations around cracks can form patterns that look similar to the gray and white backgrounds above.

Flow & Liquids



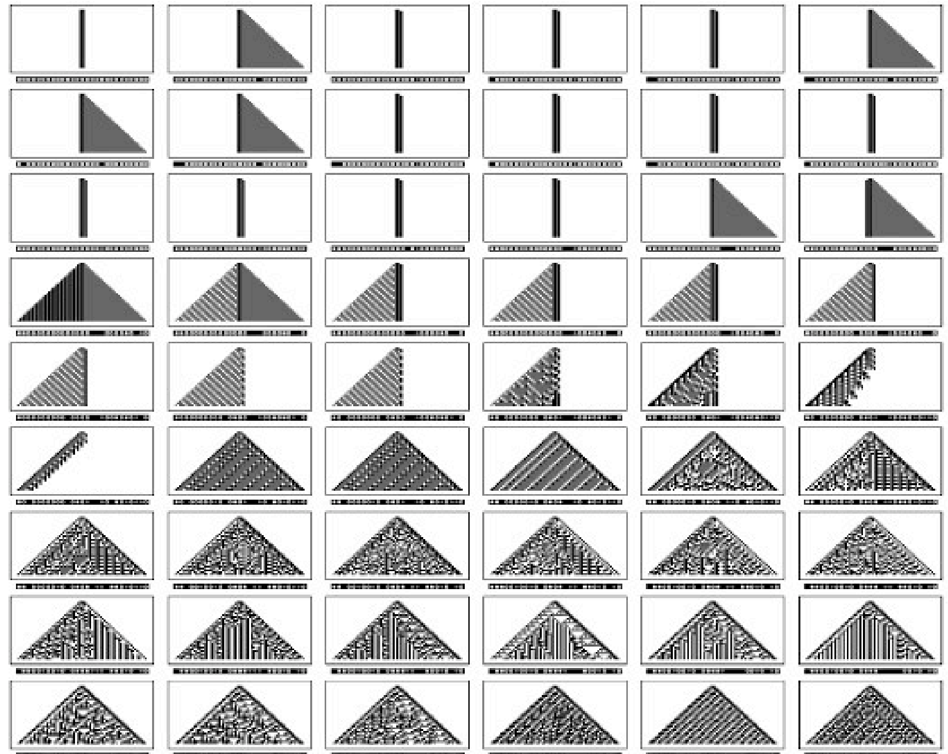
(b) 5 x 5 averages (c) 25 x 25 averages

- Wolfram describes fluid that flows past an object, where acceleration-> flow > eddies pair spirals > periodic break of eddies into wake > further turbulence
- Then, starting from non-random states, CA programs model bulk/continuous properties like flow(above), Rule 225 looks like turbulence (right)



Wolfram vs.natural selection

- Basic claim: complexity in biology spontaneously occurs all the time as a property of all things (its unclear)
  - Also, natural selection **somehow** suppresses complexity
  - Pigmentation pattern aberrations are rarely or never deleterious
  - Apparently, selection is an optimization algorithm

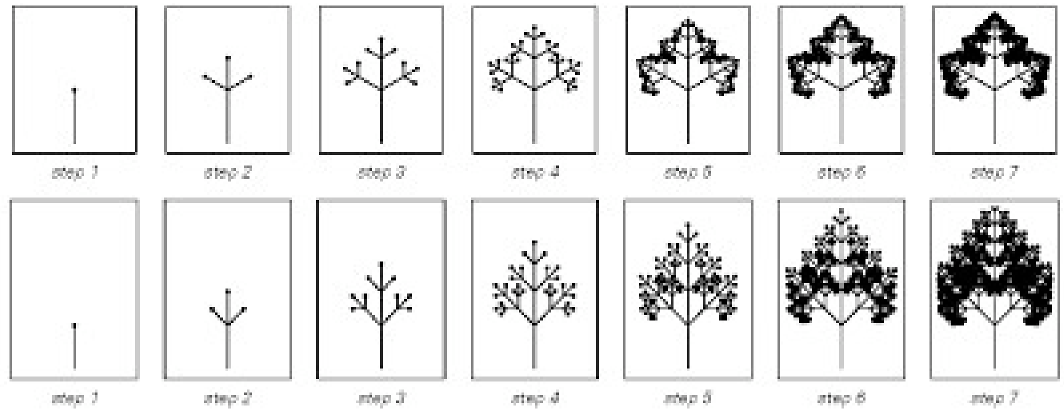


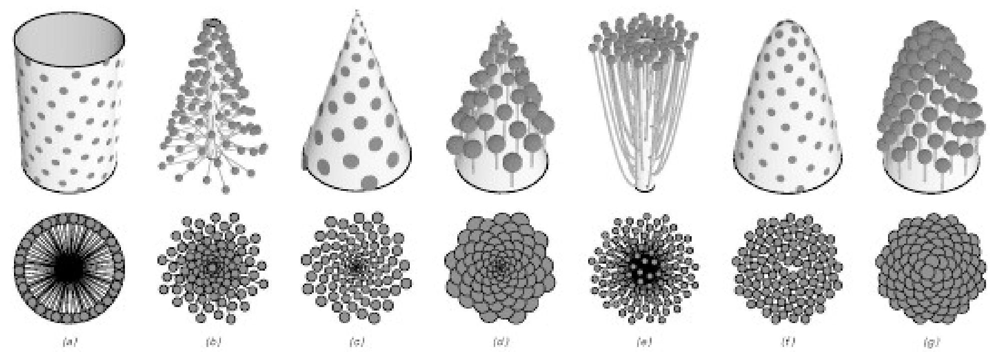
Apparently this set of rule-mutating automata disproves natural selection.

"But if complexity is this easy to get, why is it not even more widespread in biology? For while there are certainly many examples of elaborate forms and patterns in biological systems, the overall shapes and many of the most obvious features of typical organisms are usually quite simple. " THIS IS COMPLETELY WRONG

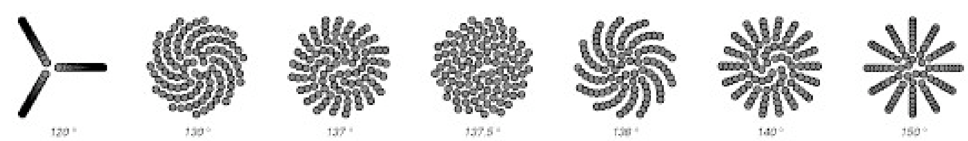
Plant growth patterning

- Iterative branching algorithm
- NetLogo demo
- Iterative processes may explain “golden ratio” spirals seen in plants
  - I do not follow his inaccurate plant biology





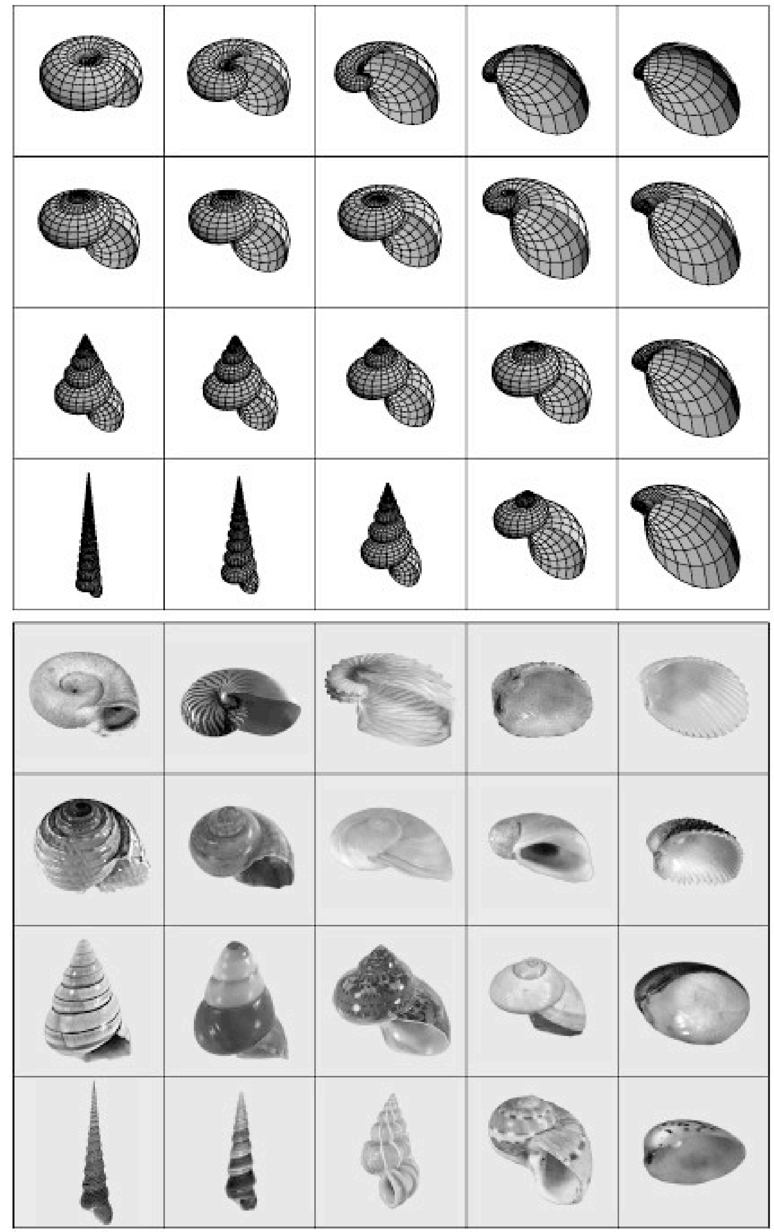
Examples of structures formed in various geometries by successively adding elements at a golden ratio angle 137.5°. Each of these structures is seen in one type of plant growth or another, as illustrated on page 409.



Overall patterns formed by successively adding elements at a variety of different angles. In each case the n^{th} element appears at coordinates $\sqrt{n}$ *{Cos[n θ], Sin[n θ]}*. Stripes are seen if θ/π (with θ in radians) is easy to approximate by a rational number. (The size of the region before stripes appear depends on *Length[ContinuedFraction[θ/π]]*.)

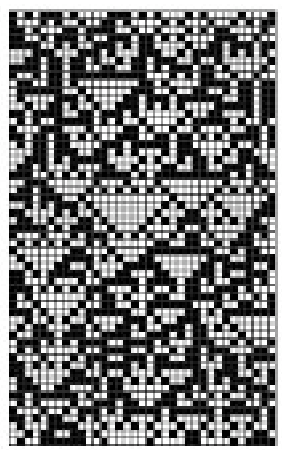
Animal growth

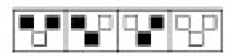
- Mollusk procedural shell growth is modeled well
- As I read it, he claims that “folding” and (hand-waving) size-mediate organogenesis?



Financial systems

- Can CA explain volatility in markets?





An example of a very simple idealized model of a market. Each cell corresponds to an entity that either buys or sells on each step. The behavior of a given cell is determined by looking at the behavior of its two neighbors on the step before according to the rule shown. The plot below gives as a rough analog of a market price the running difference of the total numbers of black and white cells at successive steps. And although there are patches of predictability that can be seen in the complete behavior of the system the plot on the right looks in many respects random.