# COMPUTATION WITH COMPETING PATTERNS IN LIFE-LIKE AUTOMATON

Genaro J. Martínez, Andrew Adamatzky, Kenichi Morita, and Maurice Margenstern

Unconventional Computing Center, University of the West of England, United Kingdom
Institute of Nuclear Science, National Autonomous University of Mexico, Mexico
Hiroshima University, Higashi-Hiroshima, Japan
Université de Metz, Metz Cedex, France

UC2010 9th International Conference on Unconventional Computation 2010
University of Tokyo, Tokyo, Japan, June 21-25, 2010

Special Session on Cellular Automata, Algorithms and Architectures (CAAA 2010)
Caen, France, July 1, 2010

# Objectives

(a) Search *simple systems* able to make computations.

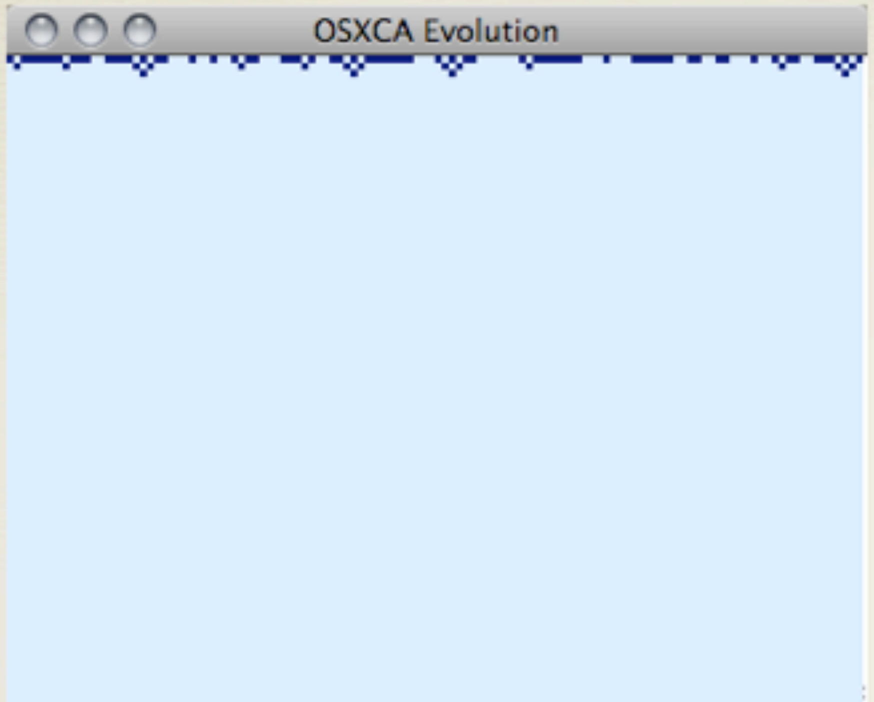
(b) Proof that a *chaotic system* (class III in Wolfram's classification) can be organised to perform computation as well, and not only complex systems.

(c) Exploit the *competing pattern* phenomenon to perform computations.

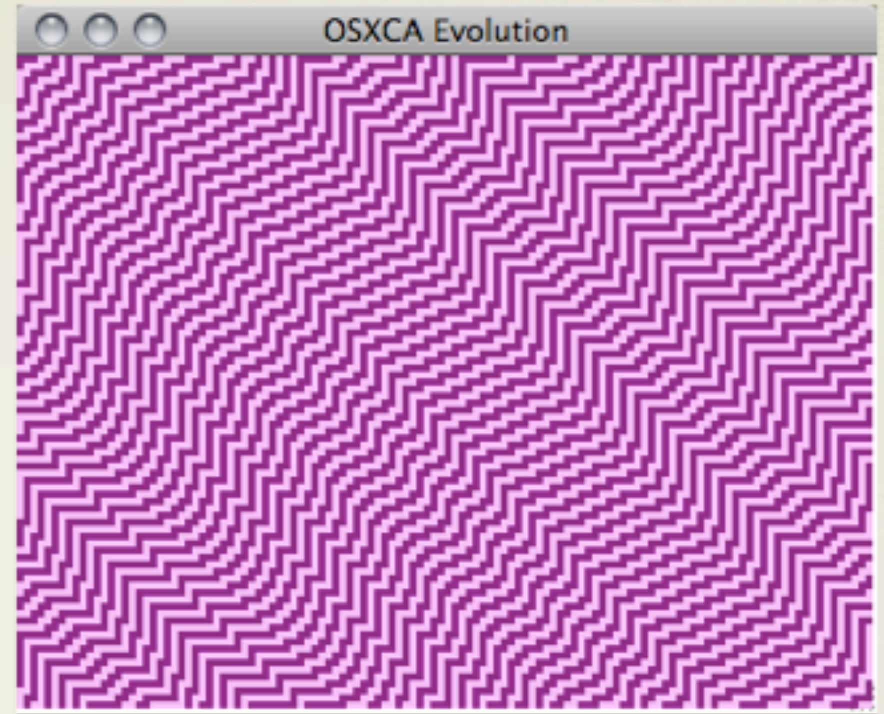
(d) Inspire from *propagation patterns* phenomena computing models in other fields.

# Cellular automata behaviour: Wolfram's classification

class I: uniform



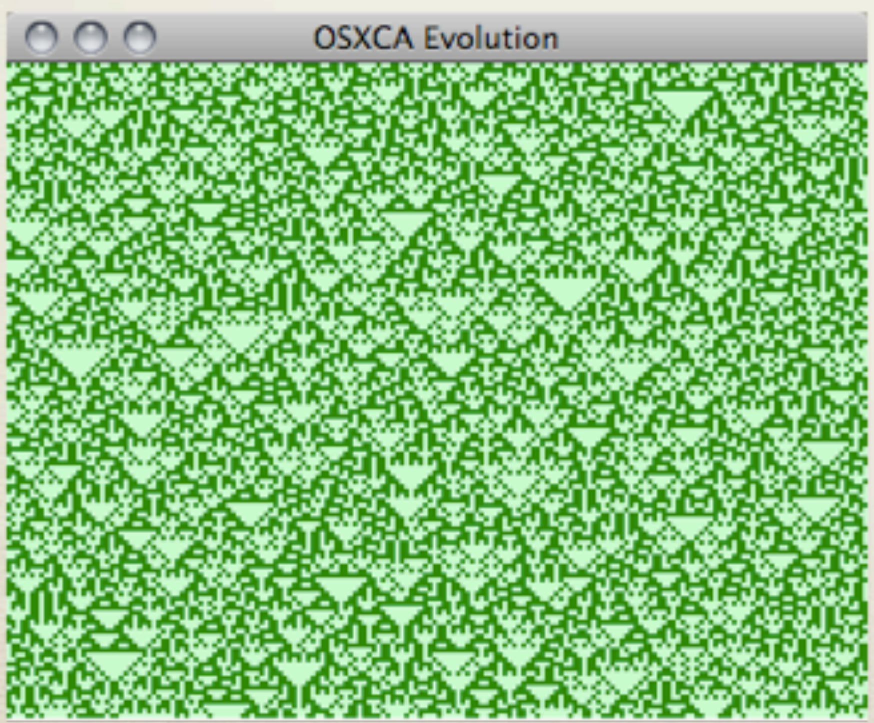

Rule 32



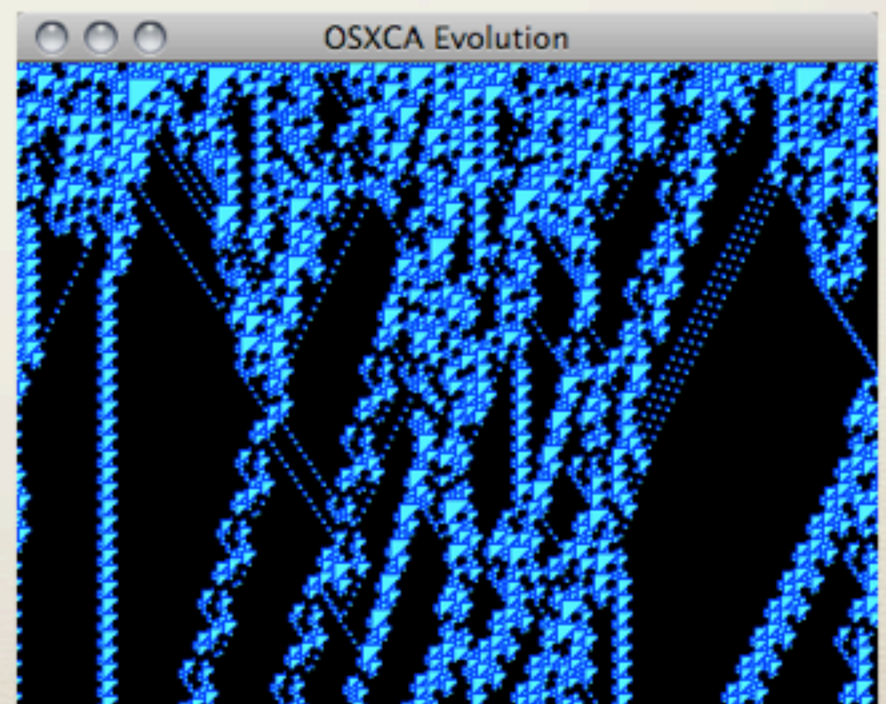

Rule 15

class II: periodic

class III: chaotic




Rule 90




Rule 110

class IV: complex

# two-dimensional CA dynamic

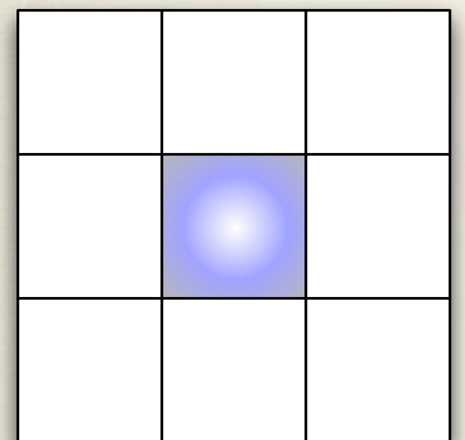
In this work, we will concentrate in two dimensional case.

Specifically on Life-like cellular automata.

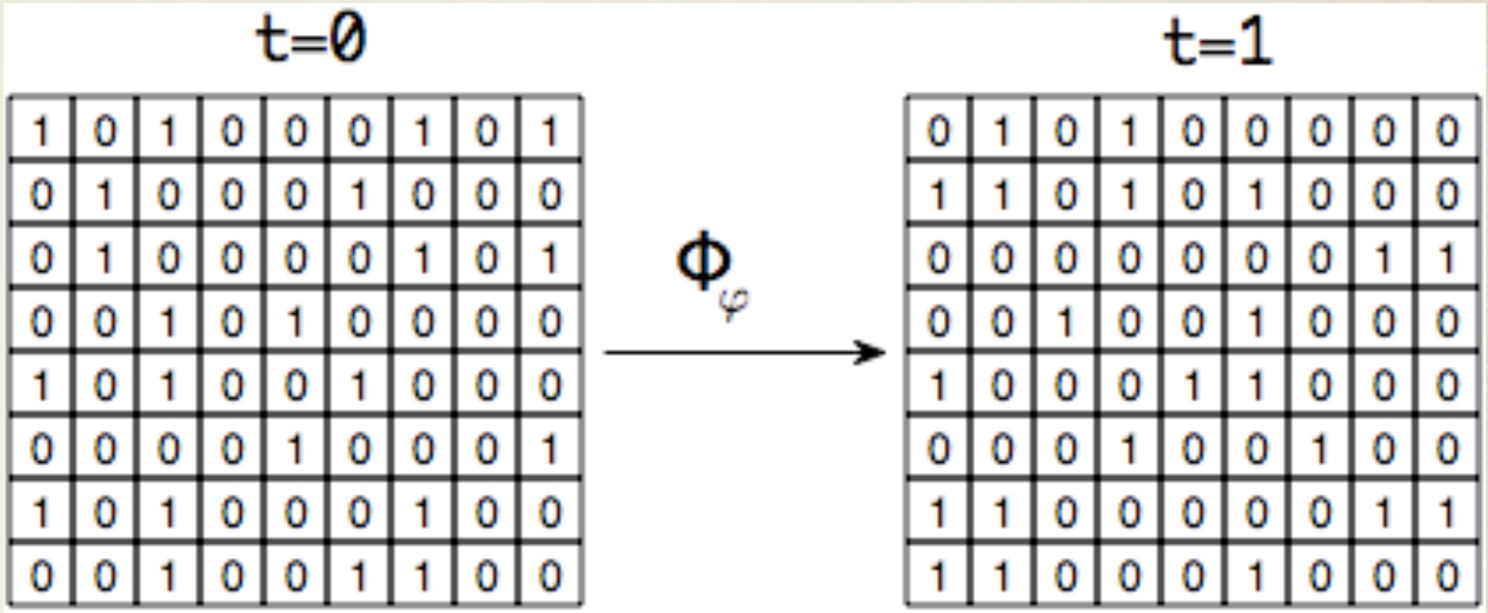
# two-dimensional CA dynamic

Moore neighbourhood

$\mathcal{V} = 8$ cells, $x \in \Sigma = \{0, 1\}$



evolution space in two dimensions




Evolution rule $\varphi$, two classic notations for semi-totalistic or additive rules:
$B/S$ or $R(S_{min}, S_{max}, N_{min}, N_{max})$

$$\varphi(\mathbf{x}_0, \mathbf{x}_1, \ldots, \mathbf{x}_\mathcal{V}) = \begin{cases} 1 \quad \text{if} & \begin{cases} \mathbf{x}_0 = 0 \quad \text{and} \quad N_{min} \leq \sum_{i=1}^{\mathcal{V}} \mathbf{x}_i \leq N_{max} \\ \mathbf{x}_0 = 1 \quad \text{and} \quad S_{min} \leq \sum_{i=1}^{\mathcal{V}} \mathbf{x}_i \leq S_{max} \end{cases} \\ 0 \quad \text{other case} & \end{cases}$$

# the Game of Life CA

The *Game of Life* made its first public appearance in the October 1970 issue of *Scientific American*, in Martin Gardner's "Mathematical Games" column.

There is a plenty of computing devices 'made of' Conway's Game of Life (GoL) cellular automaton. Examples include:

1. A register machine [Conway 1982],

2. A direct simulation of Turing machine [Chapman 2002, Rendell 2002],

3. A complete set of logical functions [Rennard 2003],

4. A design of a universal constructor [Goucher 2009].

These implementations use principles of collision-based computing where information is transferred by gliders propagating in an architecture medium. Theoretical result regarding GoL universality is only a tiny step in a long journey towards real-world implementation of the collision-based computers [Toffoli 1998].

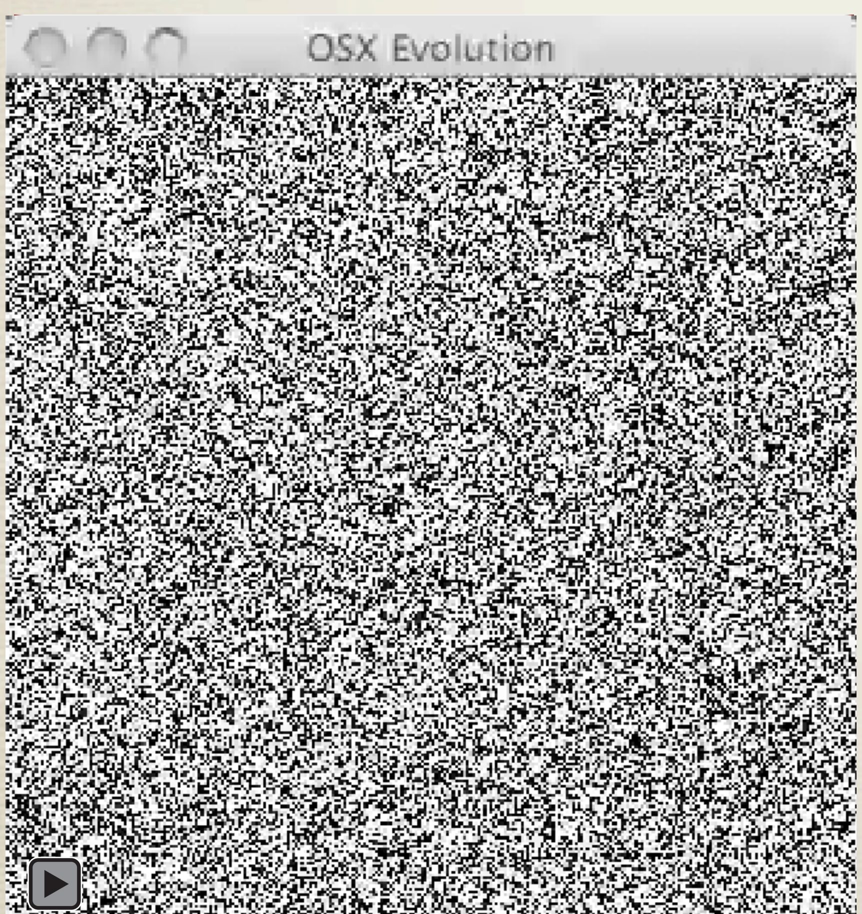
# the Game of Life

The *Game of Life* evolution rule is represented as: $B2/S23$ or $R(2,3,3,3)$

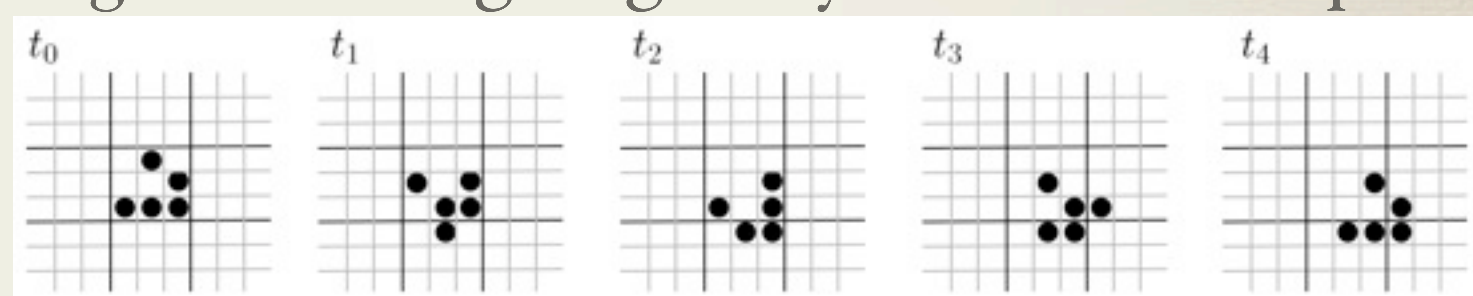
**Rules of GoL:**

1. **Survivals**. Every counter with two or three neighbouring counters survives for the next generation.
2. **Deaths**. Each counter with four or more neighbours dies (is removed) from *overpopulation*. Every counter with one neighbour or none dies from *isolation*.
3. **Births**. Each empty cell adjacent to exactly three neighbours is a birth cell.

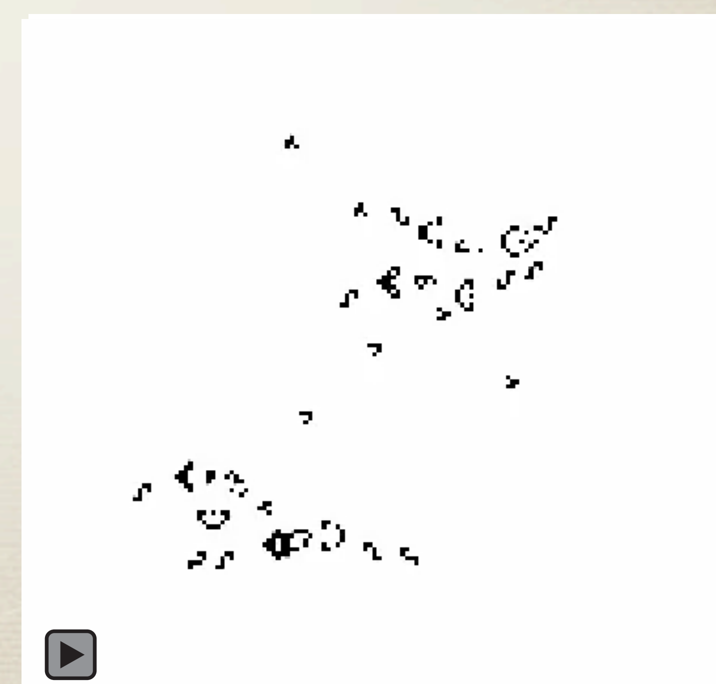
random evolution of *GoL*




*glider* moving diagonally late of four steps

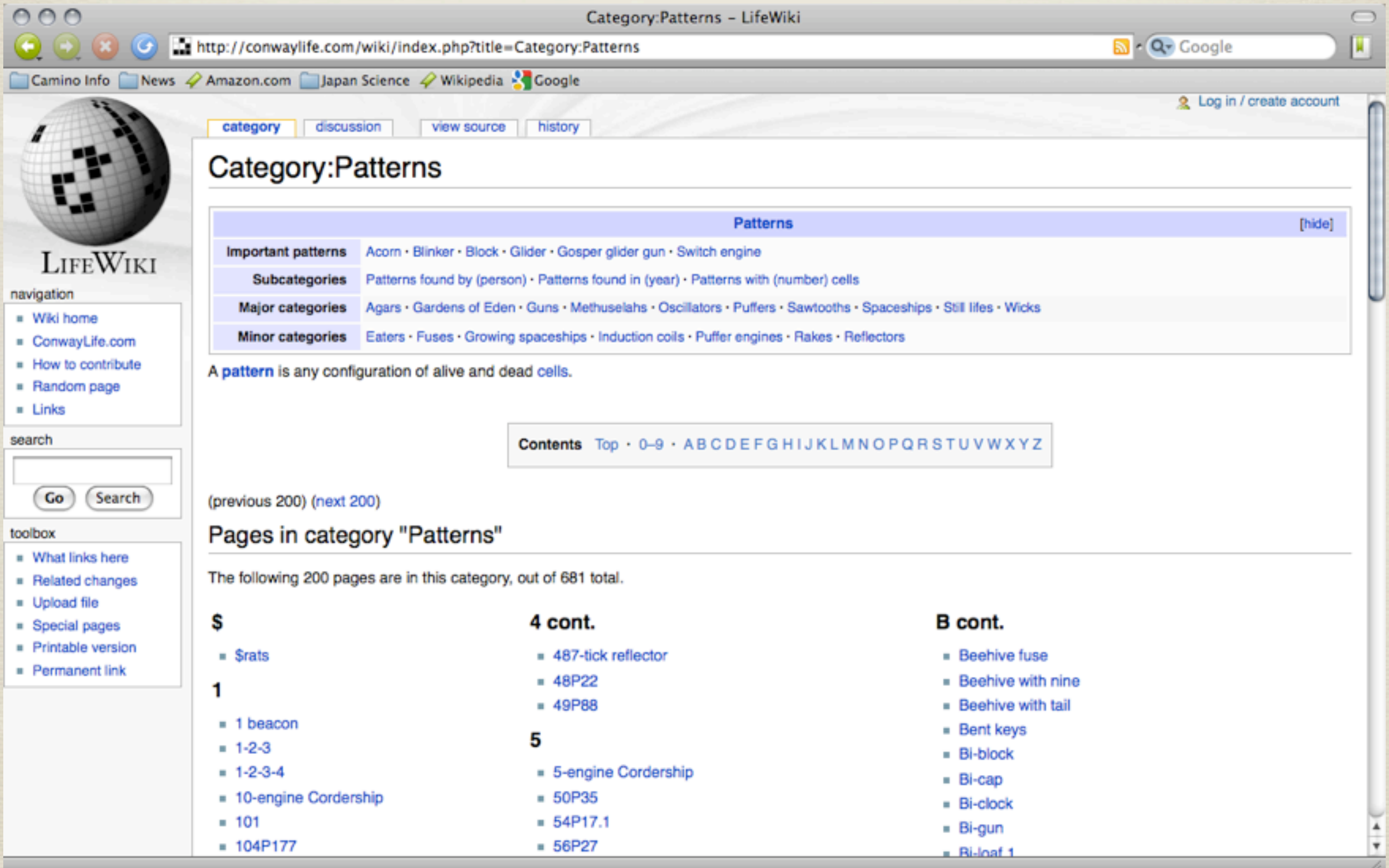



doing engineering with GoL



# the Game of Life

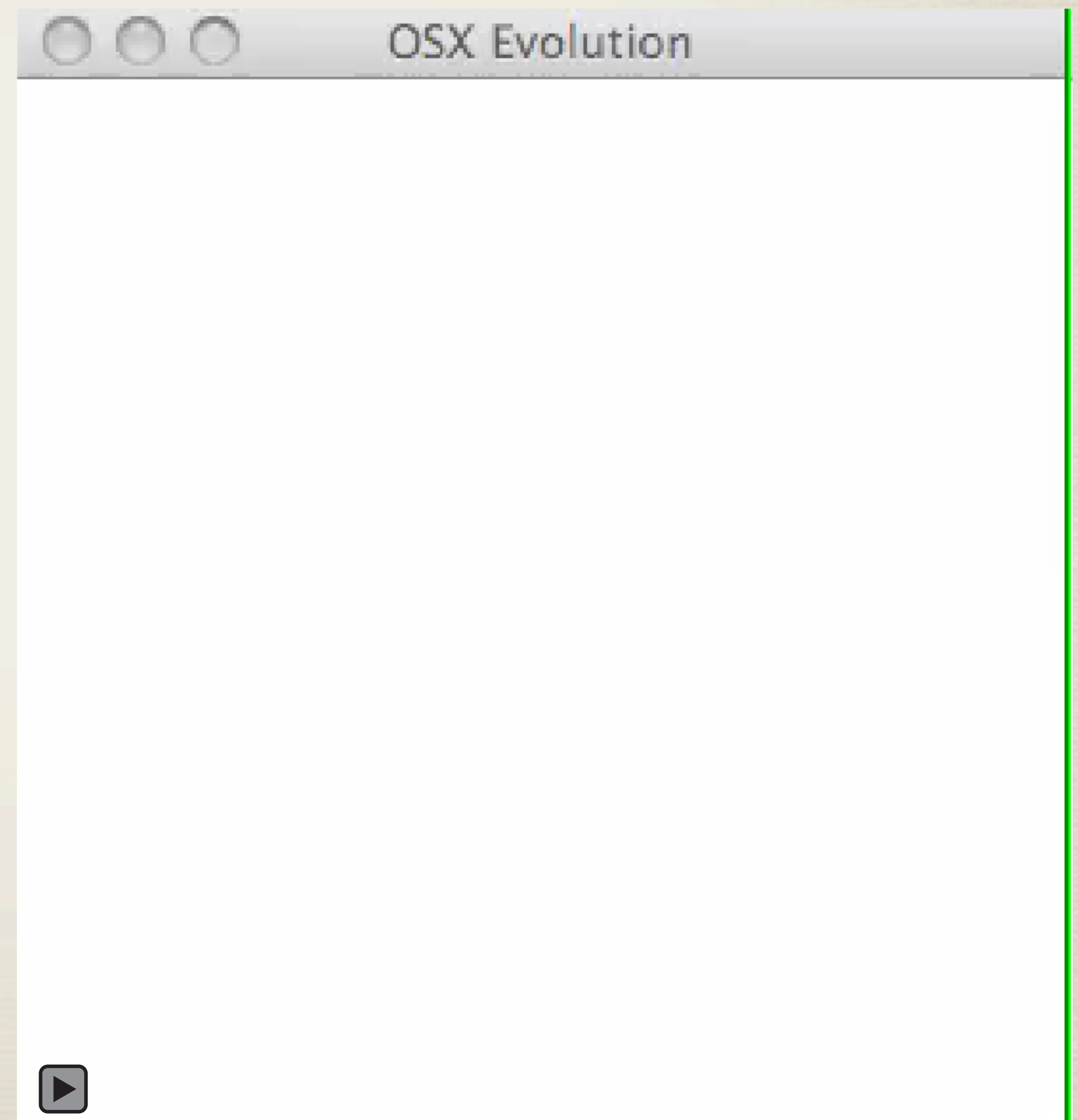
Late of 40 years yet a number of complex patterns are discovered in *Life*, relating hundred of hundred of patterns and complex constructions.



# most simple rules as Life-like rule B2/S2345

B2/S2345 is a chaotic CA. Dynamics of Life rule B2/S2345 is described for the next conditions. Each cell takes two states '0' (dead) and '1' (alive), and updates its state depending on its eight closest neighbours (Moore neighbourhood):

* **Birth**: a central cell in state 0 at time step t takes state 1 at time step t+1 if it has exactly two neighbours in state 1.
* **Survival**: a central cell in state 1 at time t remains in the state 1 at time t+1 if it has two, three, four or five live neighbours.
* **Death**: all other local situations.



# Life-like rule B2/S2345

Mean field polynomial

rule B2/S2345:

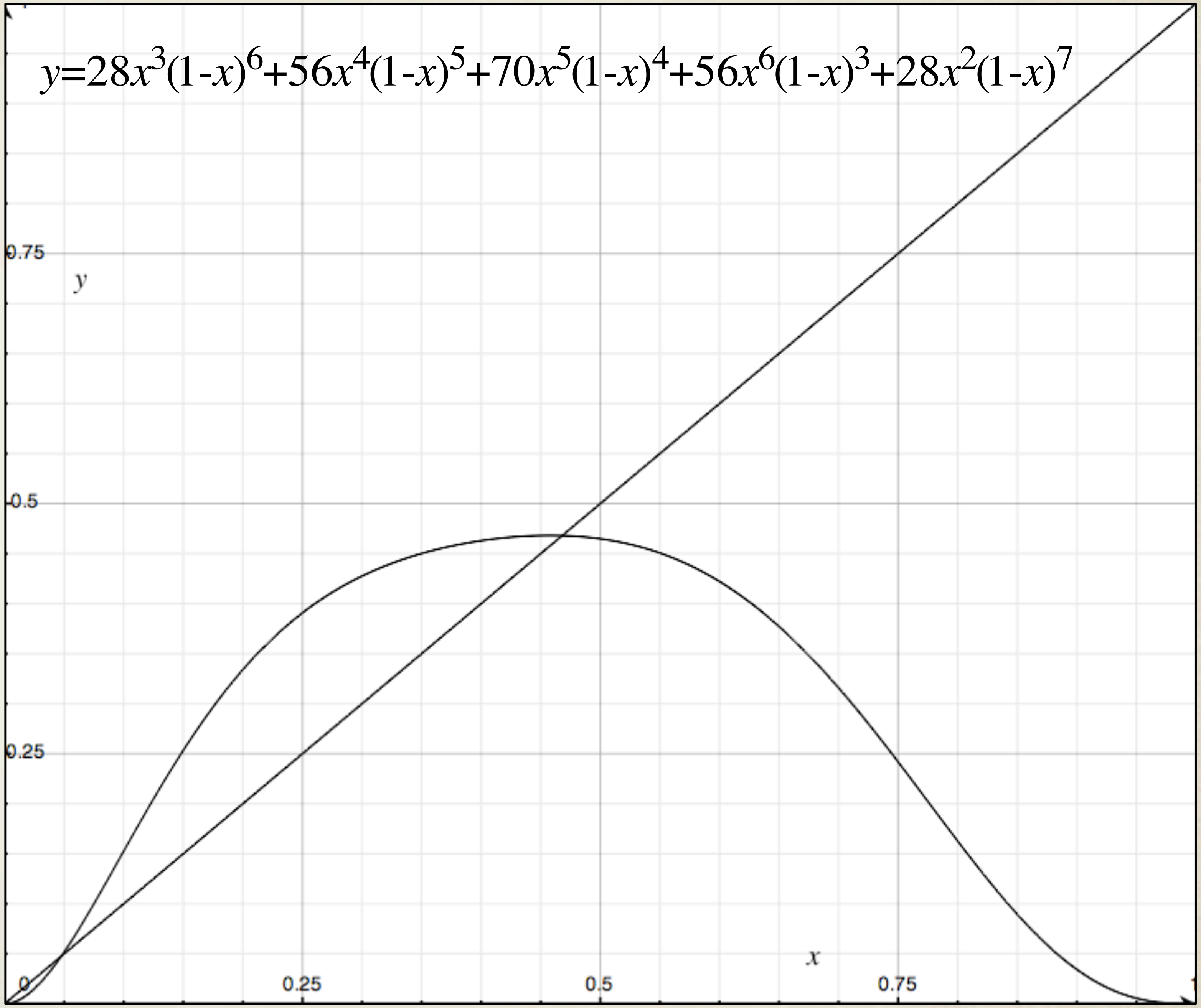
stable point: 0.468

unstable point: 0.0443

while *Life* has:

stable point: 0.37

unstable point: 0.193

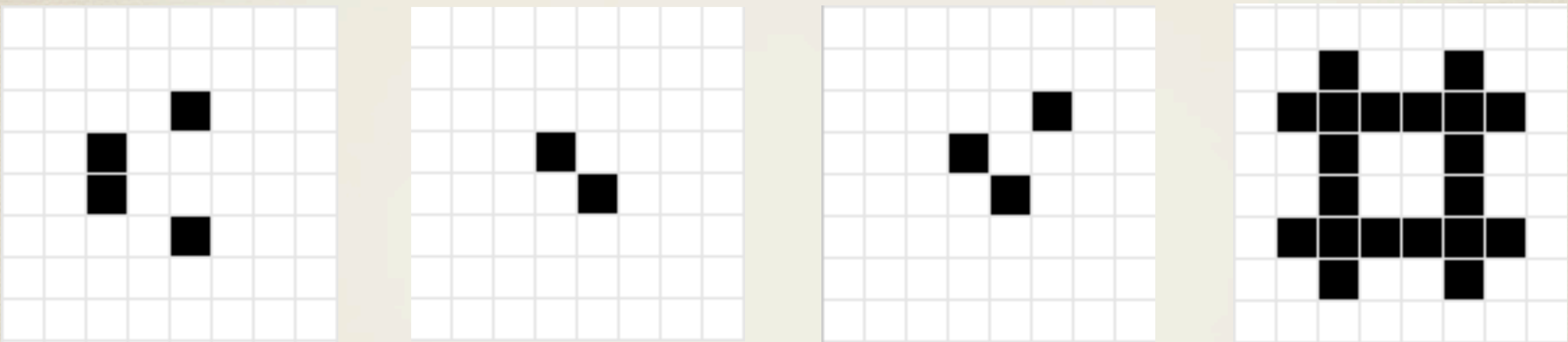



Howard A. Gutowitz and Jonathan D. Victor

Local structure theory in more that one dimension, Complex Systems 1(1), 57-68, 1987.

# Complex patterns emerging in Life rule B2/S2345

While *GoL* evolves with hundred of complex patterns, B2/S2345 evolve with few patterns.



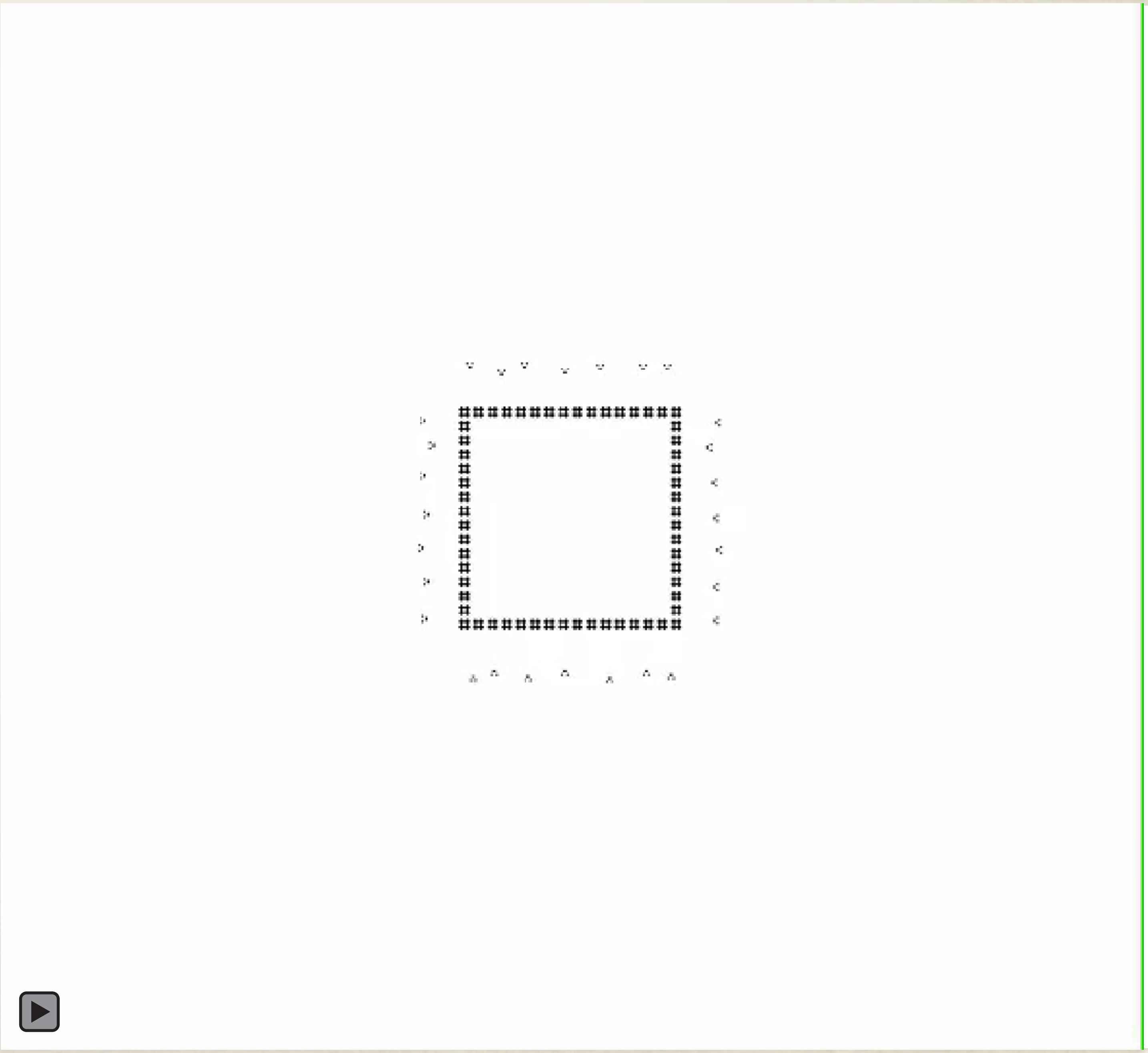
(a) (b) (c) (d)

Basic periodic structures in B2/S2345: (a) glider period one, (b) oscillator period two (flip-flop), (c) oscillator period two (blinker), and (d) still life configuration.

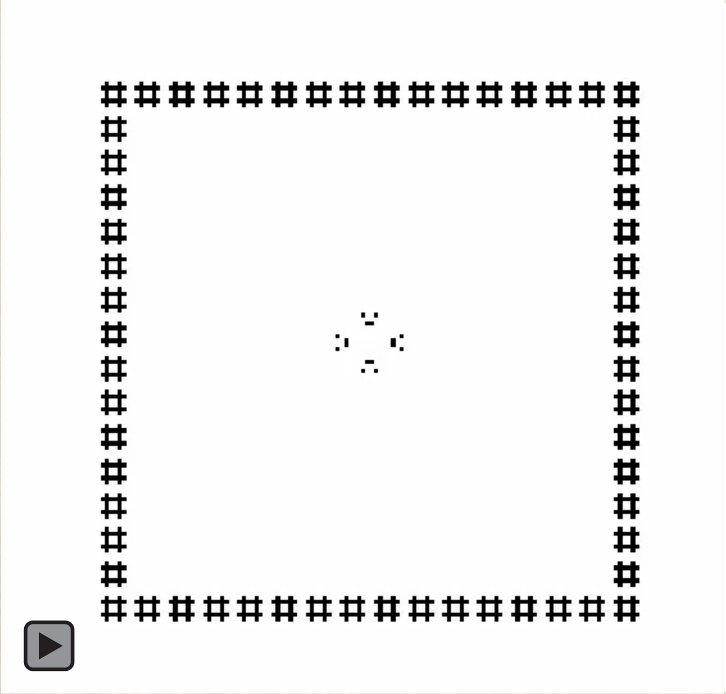
# Indestructible still life pattern

Some patterns amongst still life patterns in the rule B2/S2345 belong to a class of “indestructible patterns” (sometimes referred to as “glider-proof” patterns in GoL) which cannot be destroyed by any perturbation, including collisions with gliders. A minimal indestructible pattern, still life occupying a square of 6 x 6 cells.

impenetrable

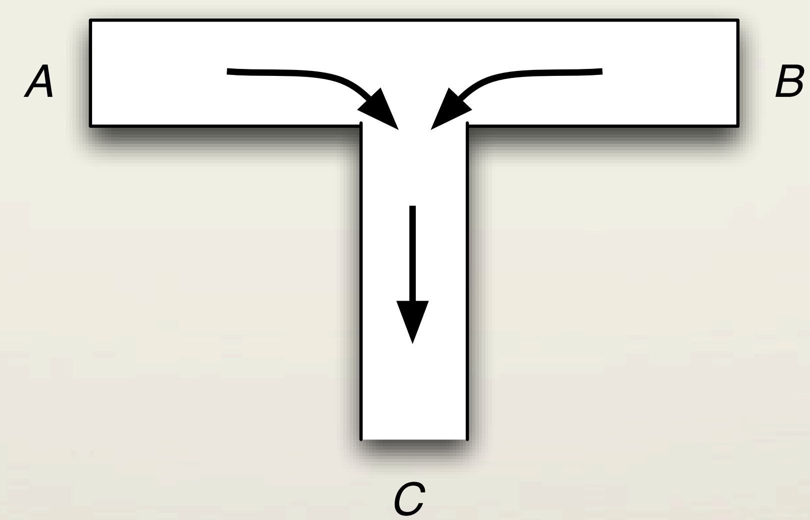


stopping super nova



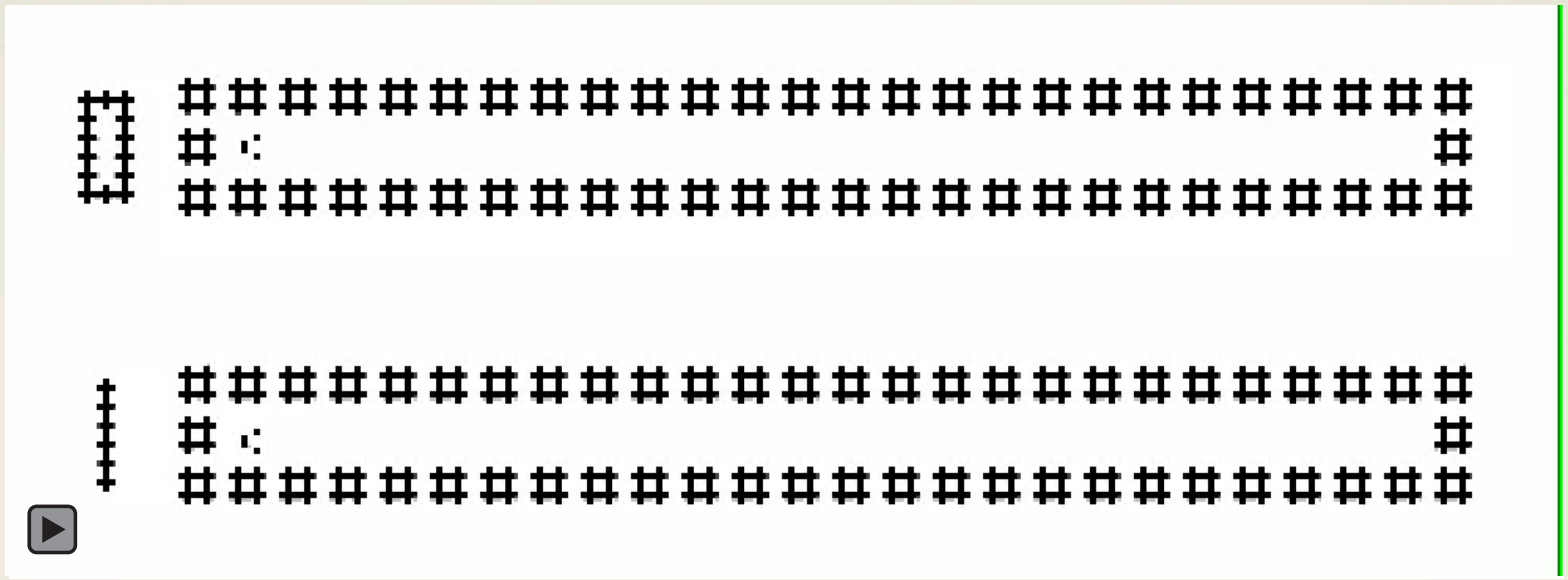
# Computing by competing patterns

The easiest way to control patterns propagating in a non-linear medium circuits is to constrain them geometrically. Constraining the media geometrically is a common technique used when designing computational schemes in spatially extended non-linear media. For example "strips" or "channels" are constructed within the medium (e.g. excitable medium) and connected together, typically using arrangements such as T-junctions. Each T-junction consists of two horizontal channels A and B (shoulders), acting as inputs, and a vertical channel, C, assigned as an output.



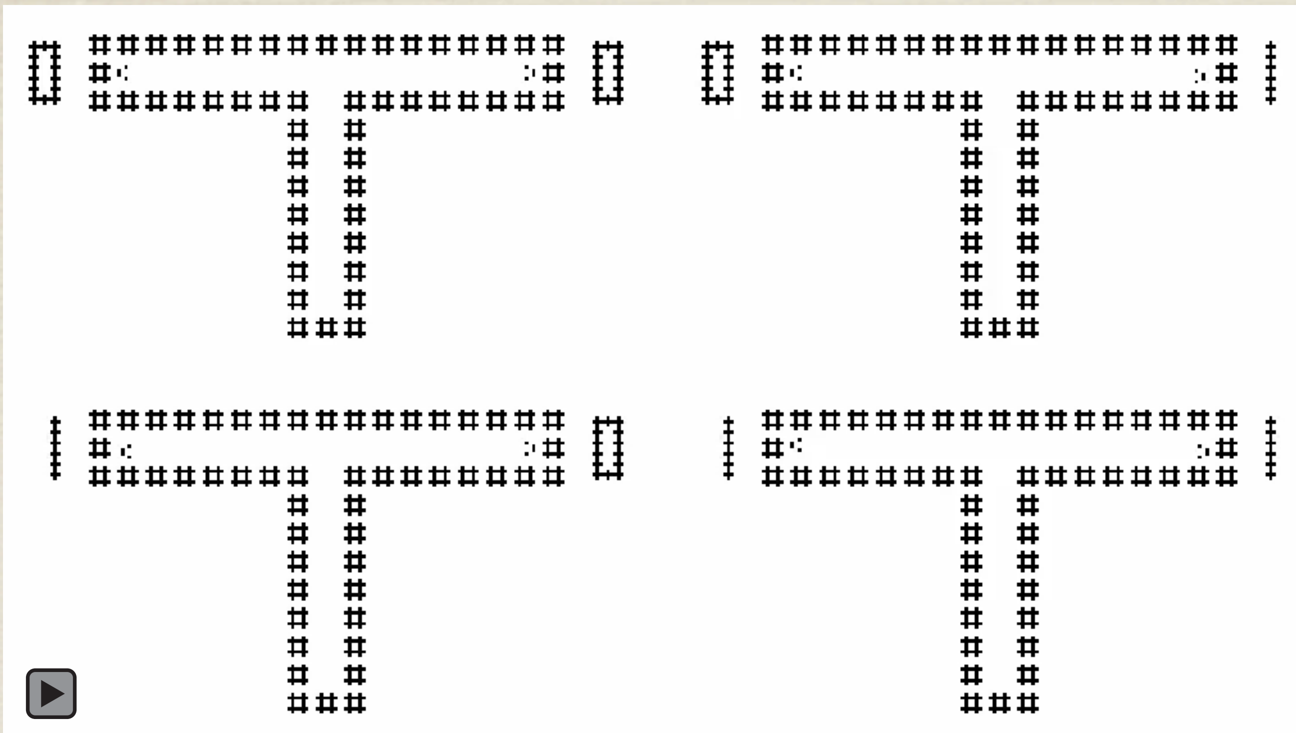

T-junction based control signals were suggested also in von Neumann [1966] works, and used by Banks [1971] and Codd [1968] as well in cellular automata.

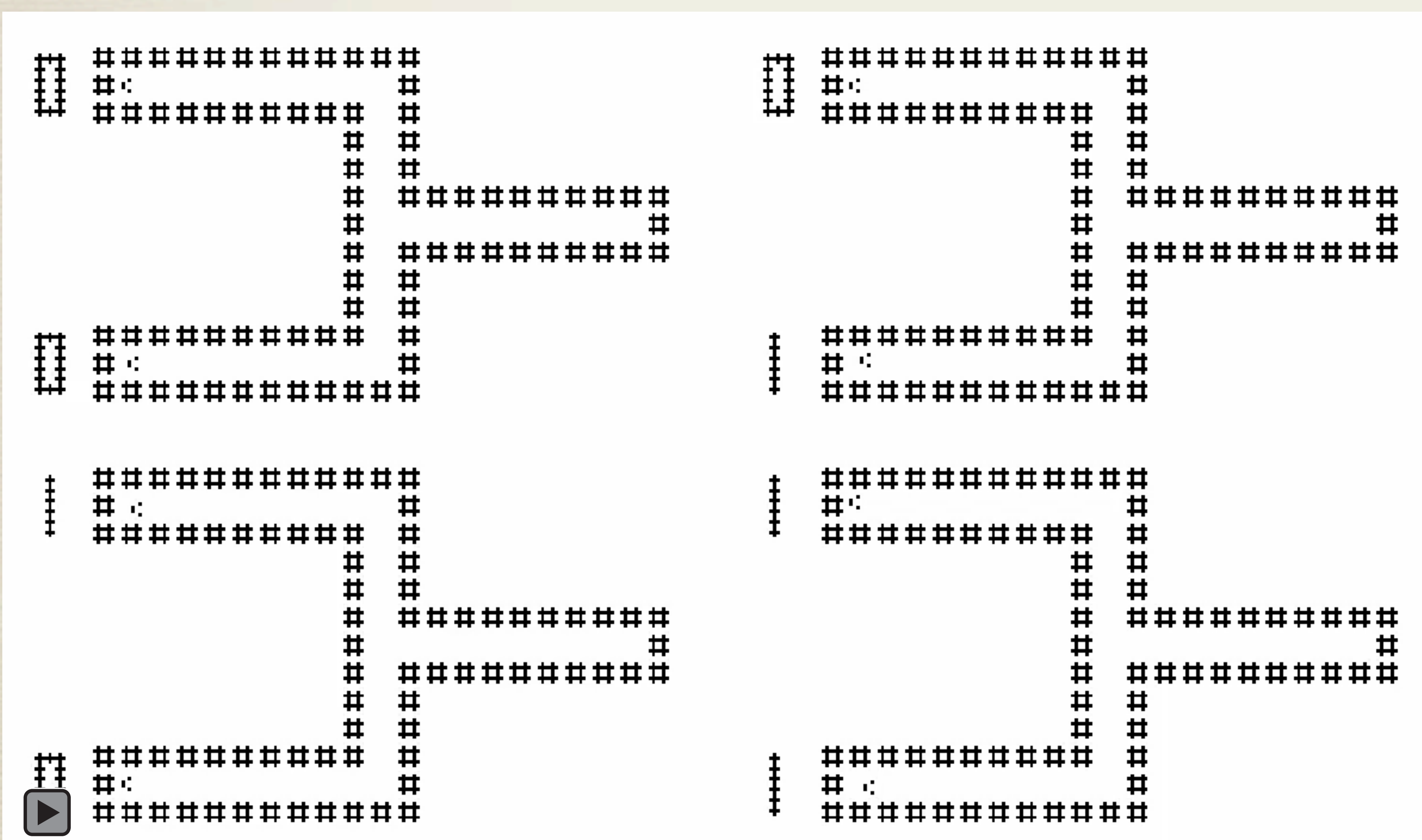
# Computing by competing patterns



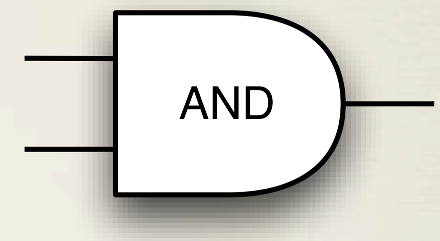
Feedback channels constructed with still life patterns (up) show the initial state with the empty channel and one glider respectively. The symmetric pattern represent value 0 (left), and non-symmetric pattern represent value 1 (right) late of glider reaction.

# Logic gates by competing patterns



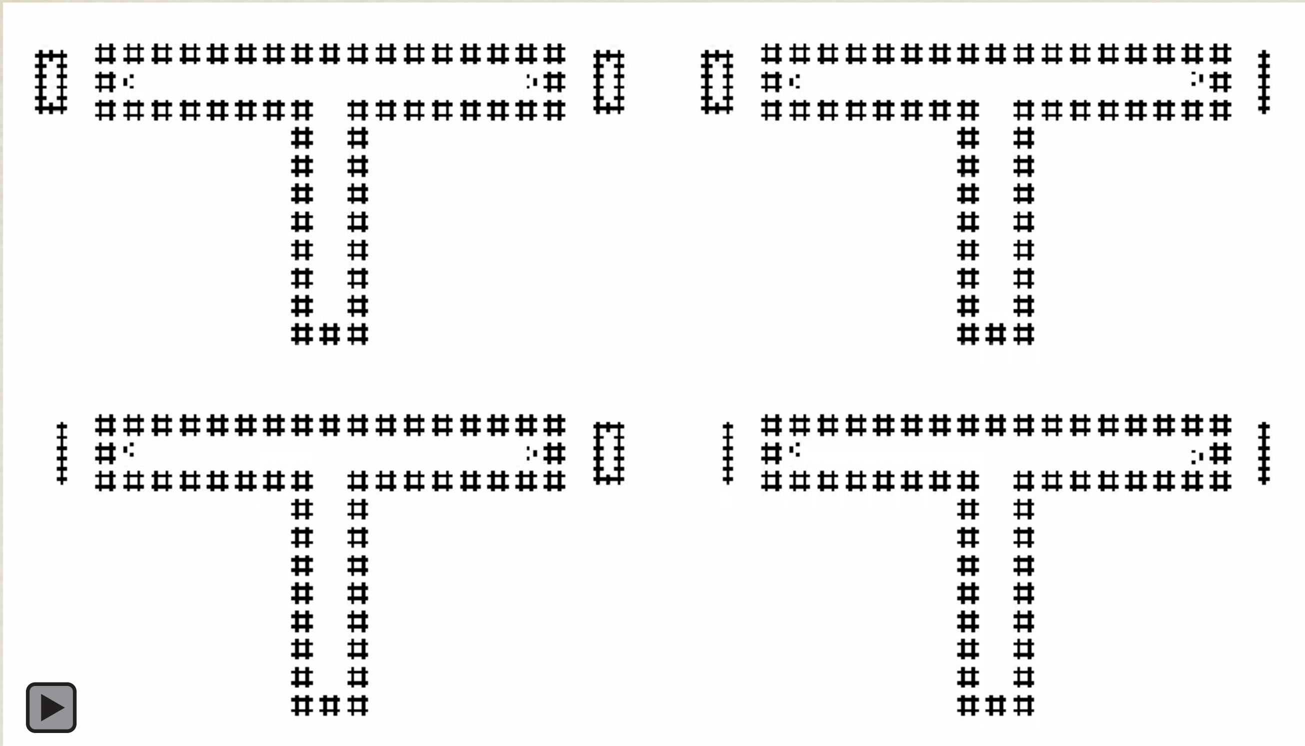


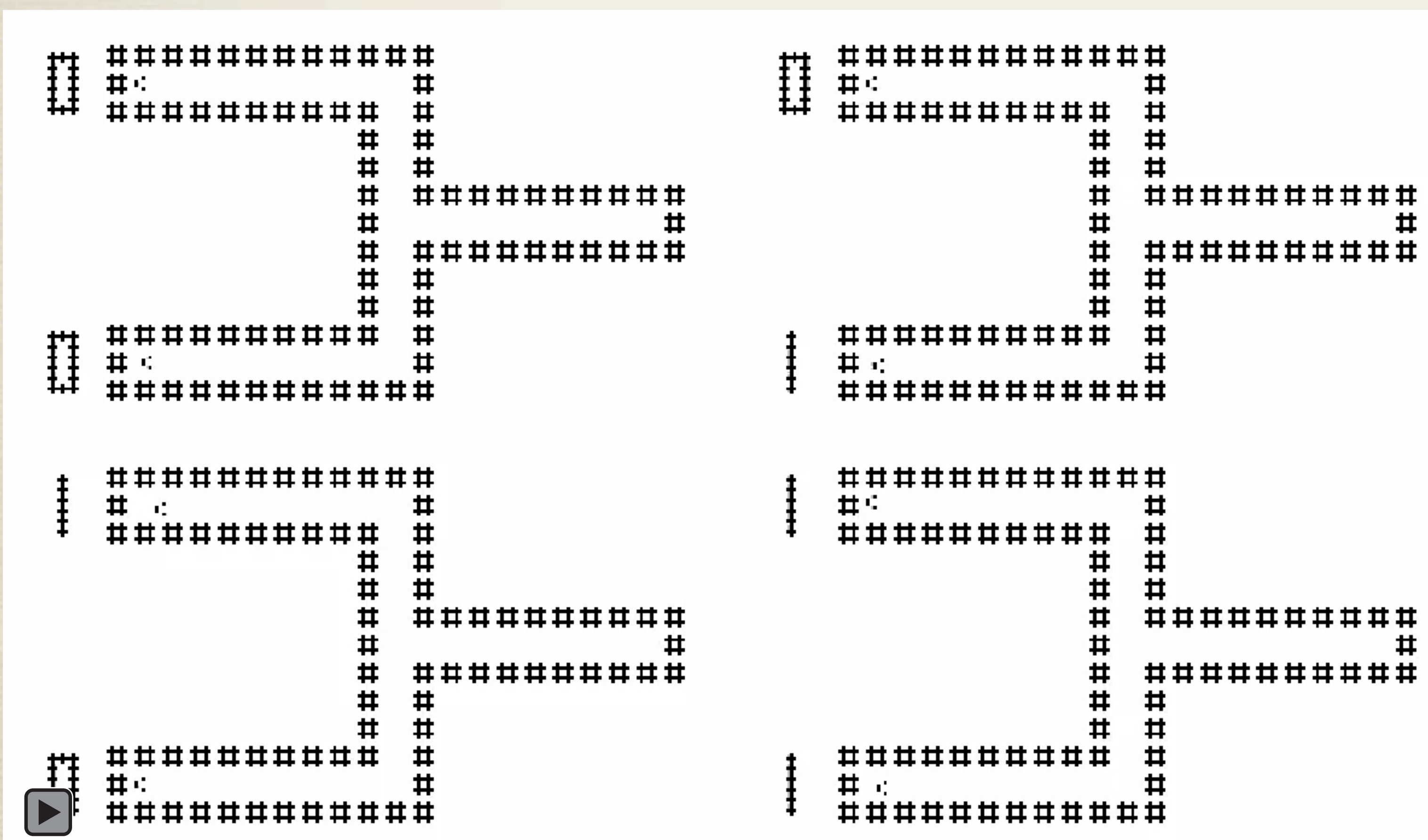
AND gate



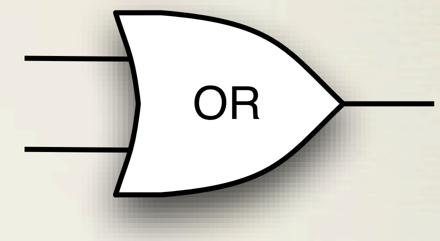

| a | b | AND |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

# Logic gates by competing patterns





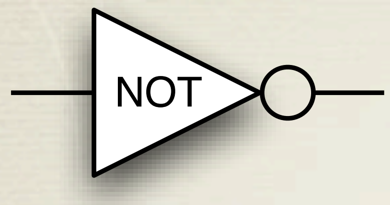
OR gate




| a | b | OR |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

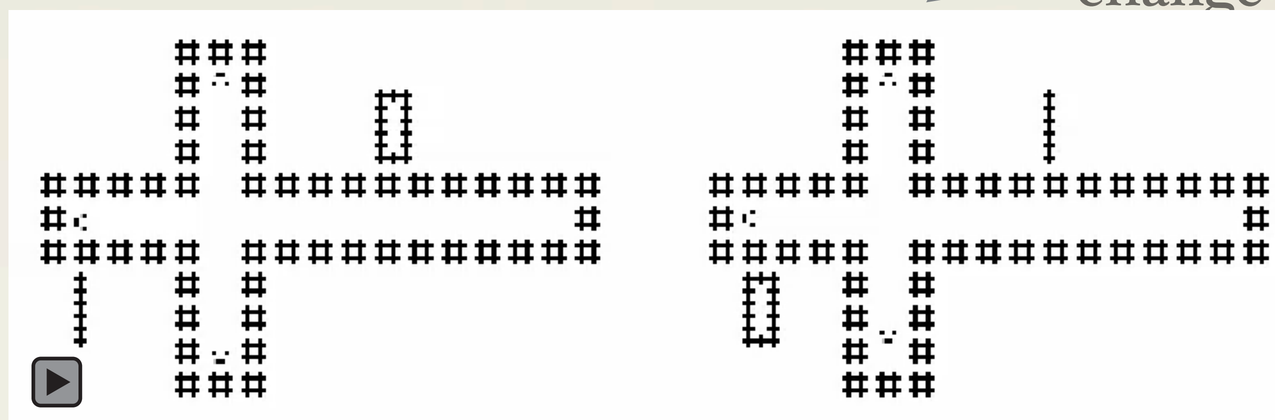
# Logic gates by competing patterns

## NOT gate

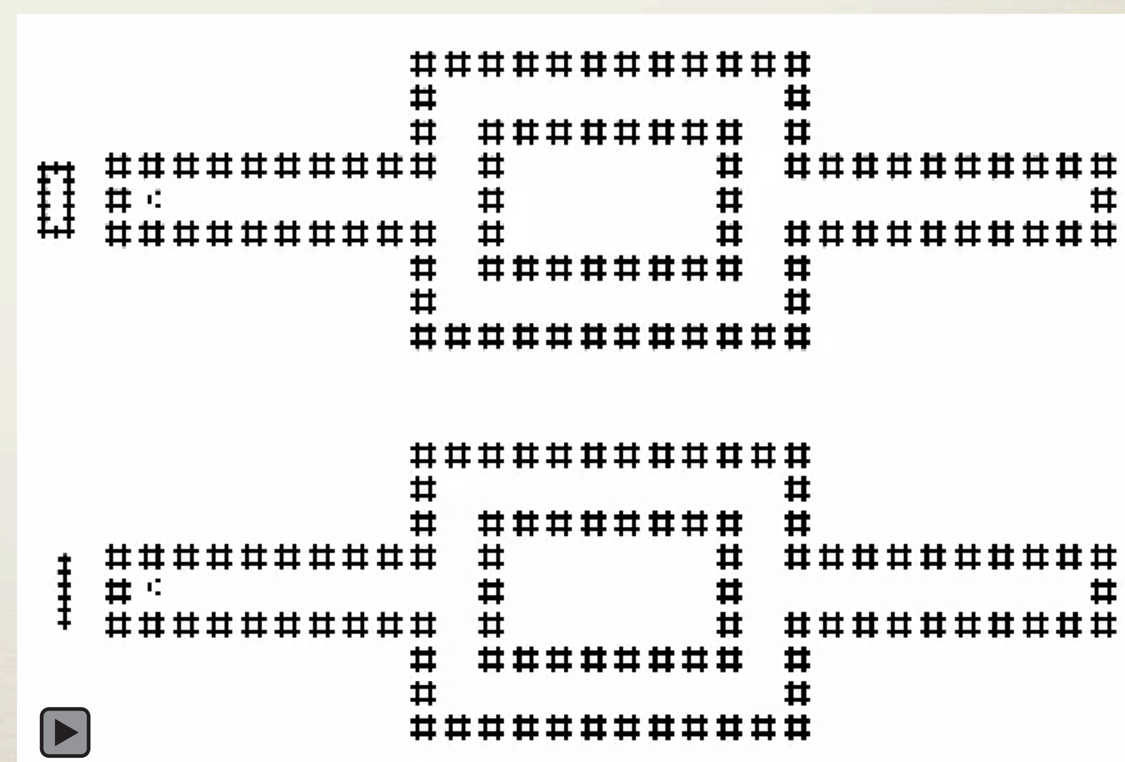



| a | NOT |
|---|---|
| 0 | 1 |
| 1 | 0 |

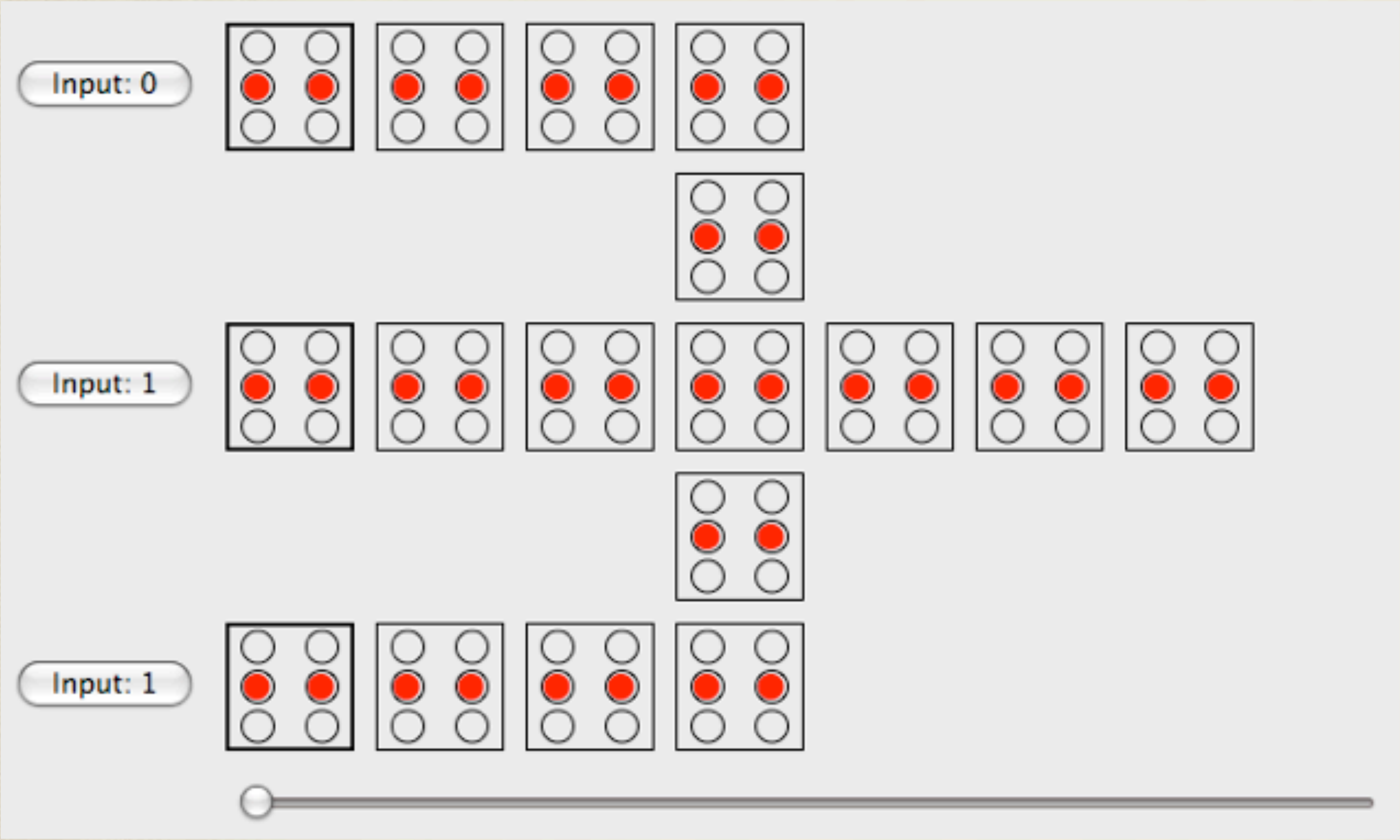
this vertical extra channel is static always and helps to change the pattern form



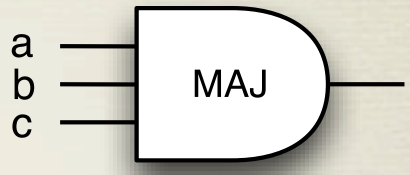
A *delay* is a very useful device that helps to synchronize times/signals and to get desired collisions



# MAJORITY gate in quantum-dot CA



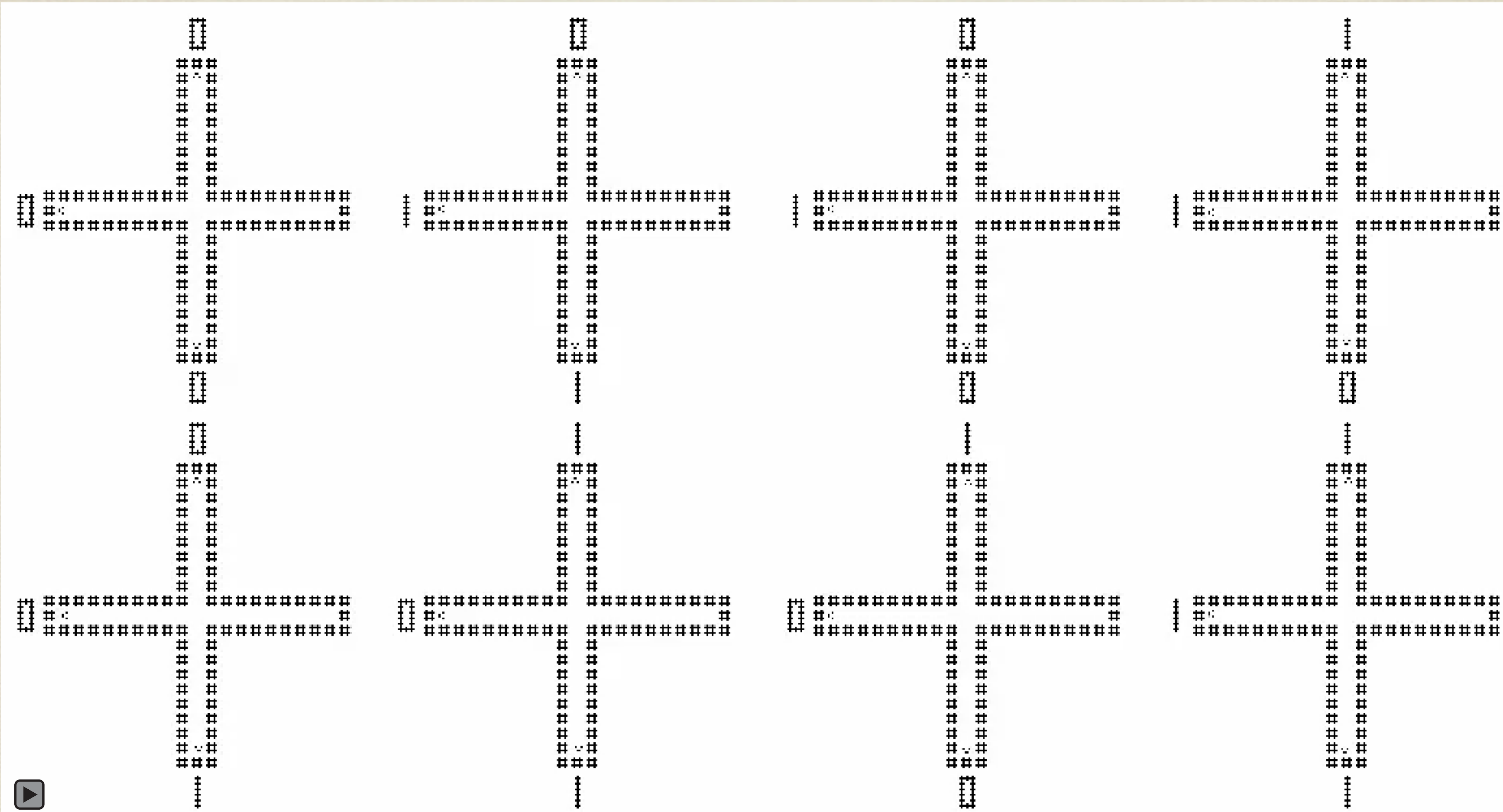

## MAJORITY gate




| a | b | c | MAJ |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 0 |
| 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 |
| 1 | 0 | 0 | 0 |
| 1 | 0 | 1 | 1 |
| 1 | 1 | 0 | 1 |
| 1 | 1 | 1 | 1 |

W. Porod, C. S. Lent, G. H. Bernstein, A. O. Orlov, I. Amlani, G. L. Snider, J. L. Mers
Quantum-dot cellular automata: computing with couple quantum dots
International Journal Electronics 86(5), 549-590, 1999.

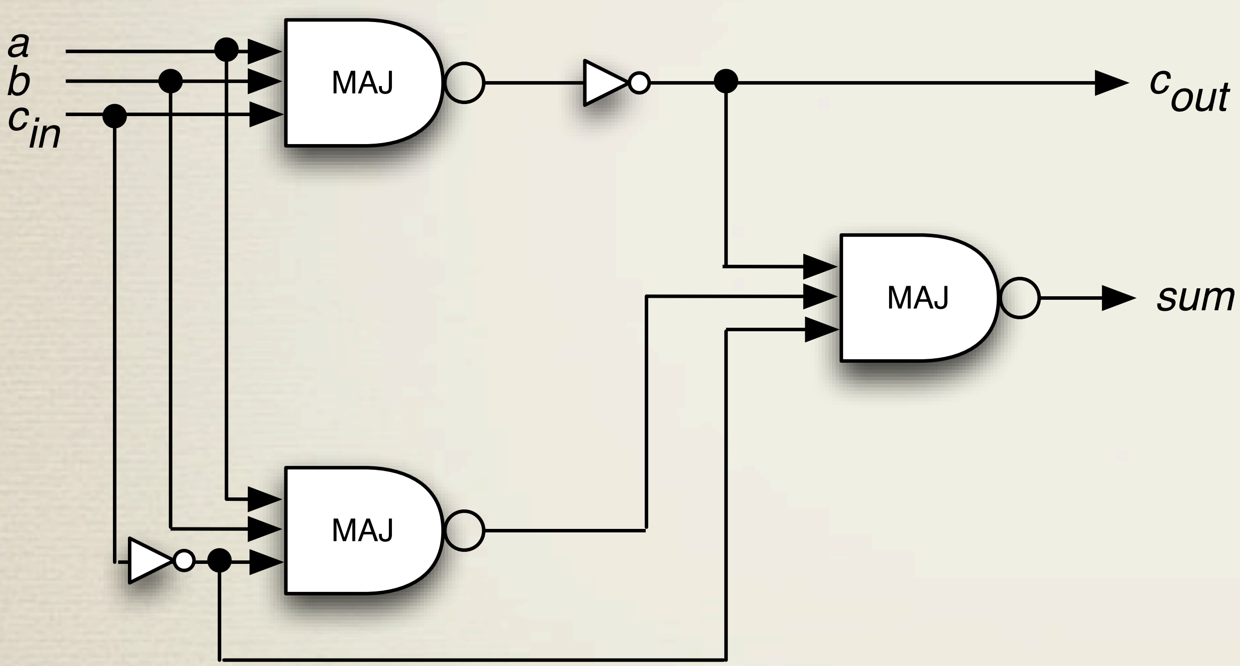
# MAJORITY gate in Life rule B2/S2345

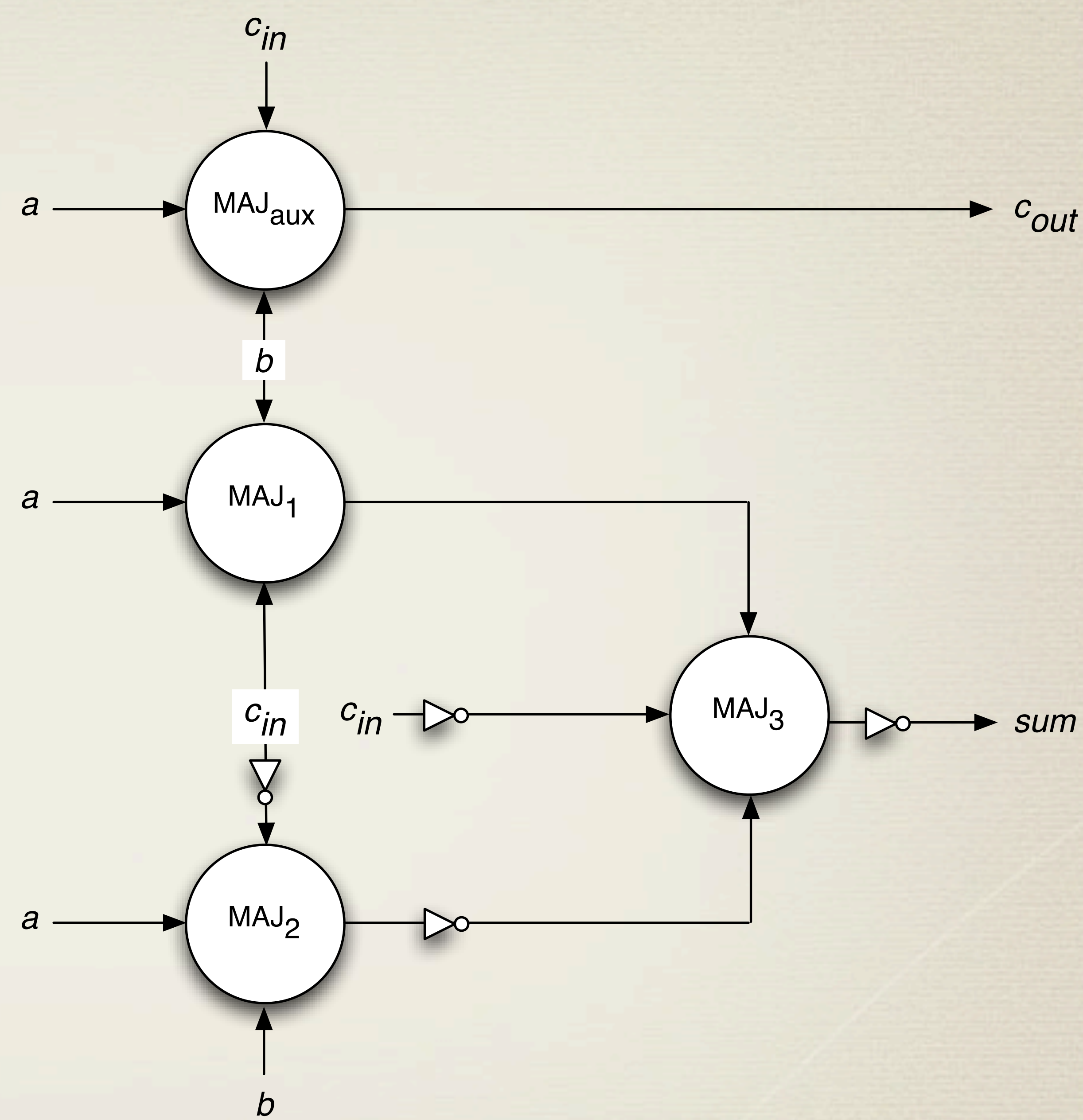


Genaro J. Martínez, Andrew Adamatzky, Ben D. L. Costello
On logical gates in precipitating medium: cellular automaton model
Physics Letters A 1(48), 1-5, 2008.

# NOT-MAJORITY adder

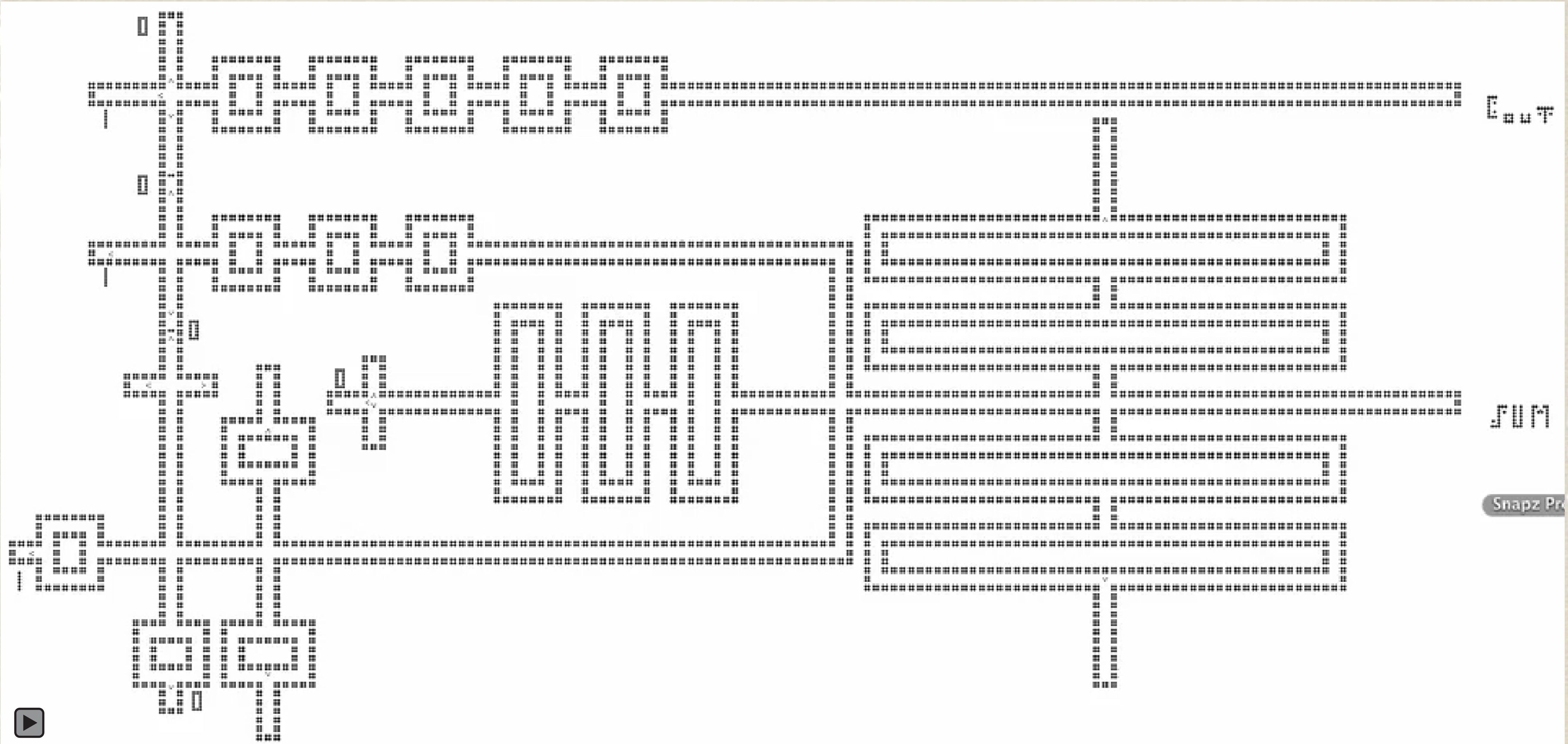
*with all previous elements hence we can design a NOT-MAJORITY adder circuit by competing patterns*




Circuit (left) and schematic diagram (right) by competing patterns of a full binary adder comprised of NOT-MAJORITY gates. Delay elements are not shown.

# NOT-MAJORITY adder



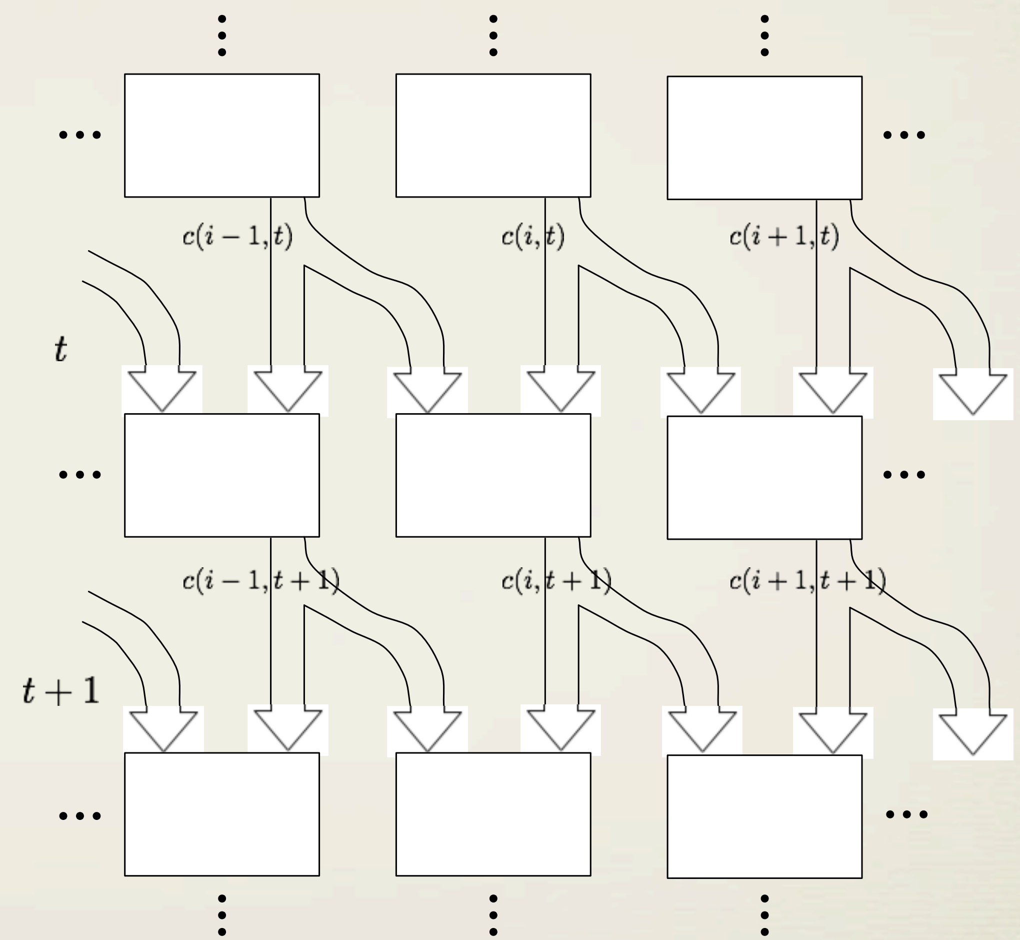

This simulation shows a case: *Inputs*: a=1, b=0, c=0. *Outputs*: carry out=0, sum=1.
Initial condition has 1,402 x 662 lattice with 58,759 cells in state 1.
Final configuration has a population of 129,923 cells to 1,439 generations.

# Limitations

However we have recognized a number of limitations on this model.

- Disadvantage of the approach presented is that computing space is geometrically constrained and the computation is *one-time-use*.

*looking a solution:*




- Also actually we do not have a way to develop a crossing signal and FANOUT gate that are essential to complete a feedback full circuit operation.

# Final remarks

* In future studies we are planning to implement the computing architecture designed in the paper to manufacture experimental prototypes of precipitating chemical computers; they will be based on crystallization of 'hot ice' [Adamatzky, 2009].

* We have demonstrated how a chaotic CA (class III in Wolfram's classification) has also capacities to performance computations.

* We show how develop computations by competing patterns.

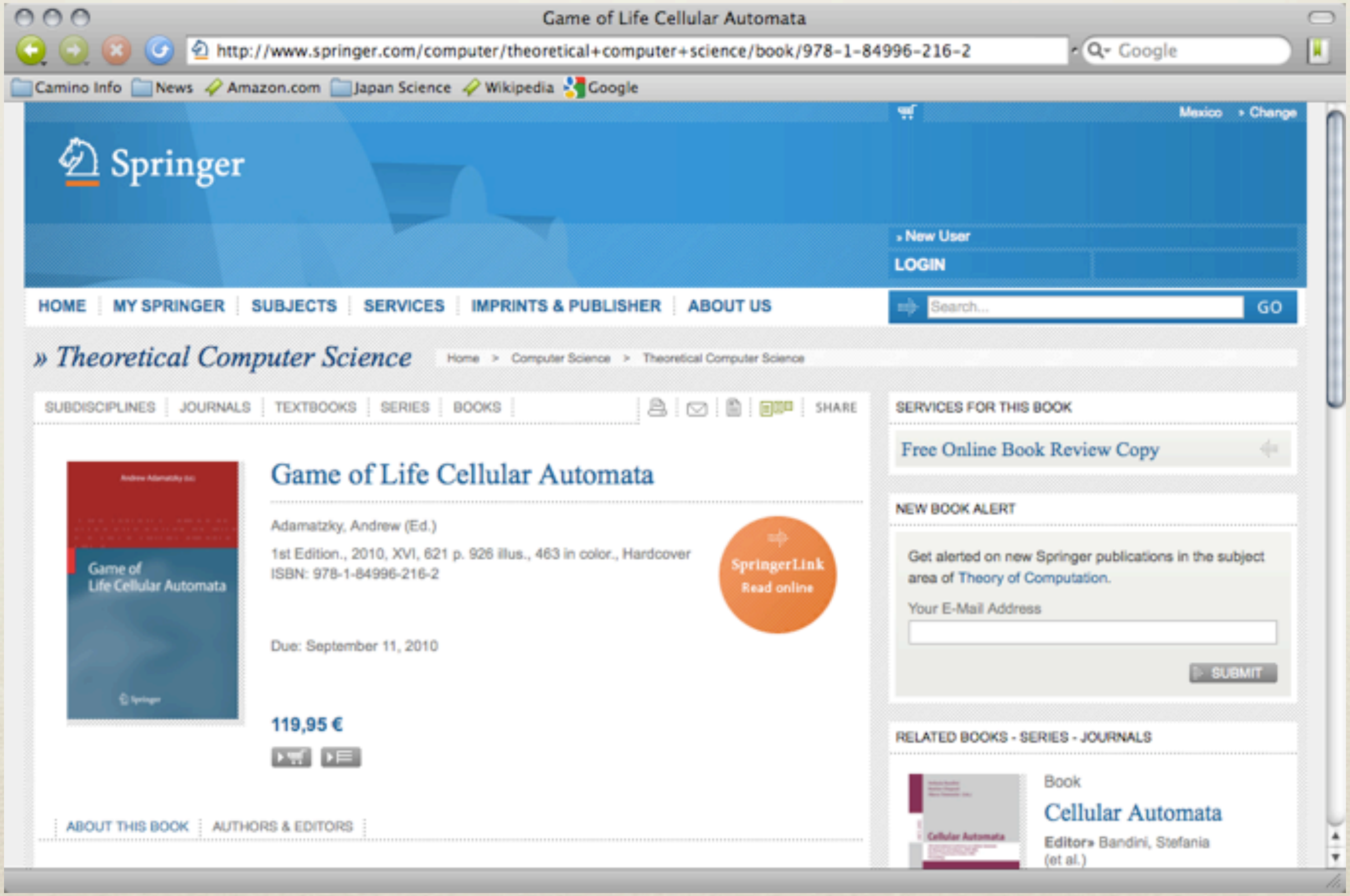
Andrew Adamatzky, Hot ice computer, Physics Letters A 374(2), 264-271, 2009.

# Life-like rules results comparison

| **CA** | **complex patterns** | **classification** | **B/S rule notation** | **computation capacities** | **universal** |
|---|---|---|---|---|---|
| Life | 600 | complex | B3/S23 | yes | yes |
| *B2/S2345* | *4* | *chaotic* | *B2/S2345* | *yes* | *no* |
| High Life | 130 | complex | B36/S23 | yes | no |
| Dead with Life | 30 | complex | B3/S012345678 | yes | no |
| Diffusion rule | 80 | chaotic | B2/S7 | no | no |
| B35/S236 | 55 | chaotic | B35/S236 | no | no |

# last news about of GoL

1. The Game of Life celebrate its 40 anniversary this year.
2. Martin Gardner died on May Saturday 22, aged 95.
3. A special book dedicated to GoL this year is edited and hoped on August 2010.



# the End

# Thank you for your attention!

RLE files, simulations, videos, sources and full paper to reproduce such results are available from:

http://uncomp.uwe.ac.uk/genaro/Life_dc22.html