

RÉSEAUX D'AUTOMATES

Eric Goles
Universidad Adolfo Ibanez
Chile
Antonio.shaco@gmail.com

Topics:

- 1) Neural or Threshold Networks: dynamics; energy; complexity
- 2) Application to the and Schelling Segregation Model and bootstrap percolation complexity .
- 3) Regulation Networks: dynamics and Robustness.
- 4) Ants models and its complexity.
- 5) Cellular Automata Communication Problems
- 6) Sand Piles and avalanches (if we have time)

Neural or threshold Networks

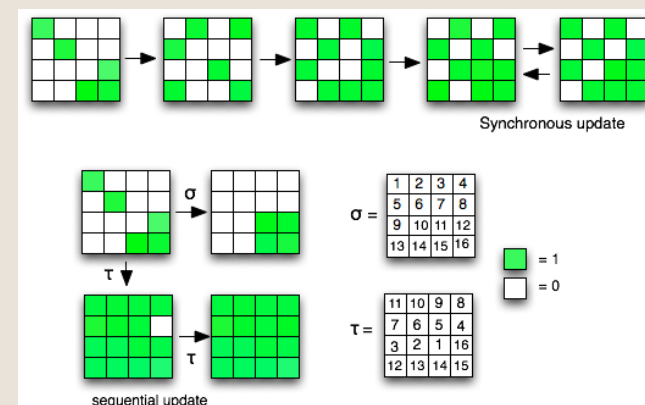
We consider a 4x4 lattice with periodic conditions, nearest interactions, states 0 or 1, and the local majority function:
If the number of ones is bigger or equal to the number of zeros then the site takes the value 1

$$x'_{ij} = 1 \quad \text{iff} \quad x_{i-1,j} + x_{i+1,j} + x_{i,j-1} + x_{i,j+1} \geq 2$$

Situation: thesard sans sujet dans un seminaire a l'IMAG Exposeurs: deux physiciens: Maynard et Rammal (1978)



Dynamics: two cycles and fixed points; different behavior for different updates



Neural networks

$$x'_i = s\left(\sum_{j=1}^n w_{ij}x_j - b_i\right) \text{ for } 1 \leq i \leq n$$

$W = (w_{ij})$ The weight matrix

$b = (b_i)$ The threshold vector

$$s(u) = \begin{cases} 1 & \text{iff } u \geq 0 \\ 0 & \text{otherwise} \end{cases}$$

Given the labels of the nodes $\{1, \dots, n\}$

Block sequential update: blocks are iterated one by one from left to right in a prescribed order:

$$(\sigma_1, \dots, \sigma_{n_1})(\sigma_{n_1+1}, \dots, \sigma_{n_2}) \dots (\sigma_{n_{q-1}+1}, \dots, \sigma_{n_q})$$

σ is a permutation

The synchronous (parallel) update: $(1, 2, \dots, n)$

A sequential update: $(1)(2) \dots (n)$

The first to remark the different iteration modes was Francois Robert, *Discrete Iterations* (Springer, 1986)

For arbitrary matrices W previous model may accept, iterated in parallel or block-sequentially, long period cycles and long transients ... But when W is symmetric the network admits short periods and an energy: (E.G and J.Olivos,

Discrete Mathematics, 1980, Discrete Applied Maths, 1981; E.G, SIAM J of Computing, 1982; E.G, F. Fogelman, Discrete Applied Maths(1985))

$$E(x(t)) = -\sum_{i=1}^n x_i(t) \sum_{j=1}^n w_{ij}x_j(t-1) + \sum_{i=1}^n b_i(x_i(t) + x_i(t-1))$$

Further, if $\text{diag}(W) \geq 0$, any sequential update admits the energy (E.G., F. Fogelman, G. Weibusch, Disc. Applied Maths. 1982)

$$E(x) = -\frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n w_{ij}x_i x_j + \sum_{i=1}^n b_i x_i$$

Which implies that:

1) for the synchronous iteration the attractors are only Fixed points or two cycles !!

2) For any sequential iteration with $\text{diag}(W) \geq 0$ there are only fixed points

3) For the parallel update $\Delta E = E(x(t)) - E(x(t-1)) < 0$ if and only if $x(t) \neq x(t-2)$

For the sequential update $\Delta E = E(x') - E(x) < 0$ iff $x' \neq x$ So, the attractors are only fixed points.

4) In both situations transients are bounded by $\alpha ||W|| ||x|| ||b||$

Some applications:

1) A neural equation with memory

$$x(t) = s\left(\sum_{k=1}^n w_k x(t-k) - b\right)$$

(la paramecie: le comportement de cette unique neurone a été Étudié par plusieurs chercheursM.Cosnard, M. Tchuente., T.de Saint Pierre. and E.G ... ce que constitu un abus too much !!!Trop de neurones pour etudier un organisme unineuronal !!!!!)

2) Majority functions and Bootstrap percolation models

(Pedro Montealegre, E.G (2011))

3) Schelling Segregation

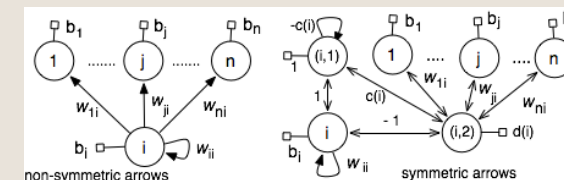
(Nicolás Goles-Domic, Sergio Rica, E.G(2010-11))

Symmetry is so restrictive?

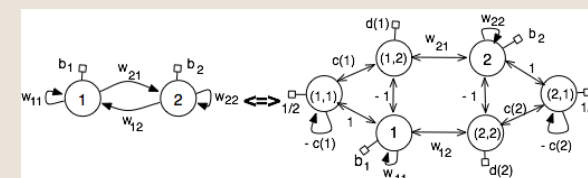
..... Non because one may simulate any non-symmetrical neural network in linear space by a symmetric one with an specific update mode ...

Theorem :Consider a neural network N with n sites, updated under the block-partition Y , then there exists a neural network N' , with $3n$ sites, updated under the block partition Y' which simulates N .

(E.G., Martin Matamala, IJCNN; Nagoya, Japan, 1993)



Remark: some diagonal weights are negatives



Example for n=2

Suppose $Y = \{I_1, \dots, I_p\}$ Then the new block-partition update is

$$Y' = (I_1), (\{(i,1)\}_{i \in I_1}), (\{(i,2)\}_{i \in I_1}), (I_2), \dots, (I_p), (\{(i,1)\}_{i \in I_p}), (\{(i,2)\}_{i \in I_p})$$

Neurons (1,1) and (1,2) copies the current state of neuron i when they are updated sequentially. So, they are a kind of memory and from (1,2) we may connect symmetrically to the others neurons

To give an other kind of answer I have to introduce a complexity measure usually used in theoretical computer science.

The class **P**: problems which we can solve in a serial computer in polynomial time.

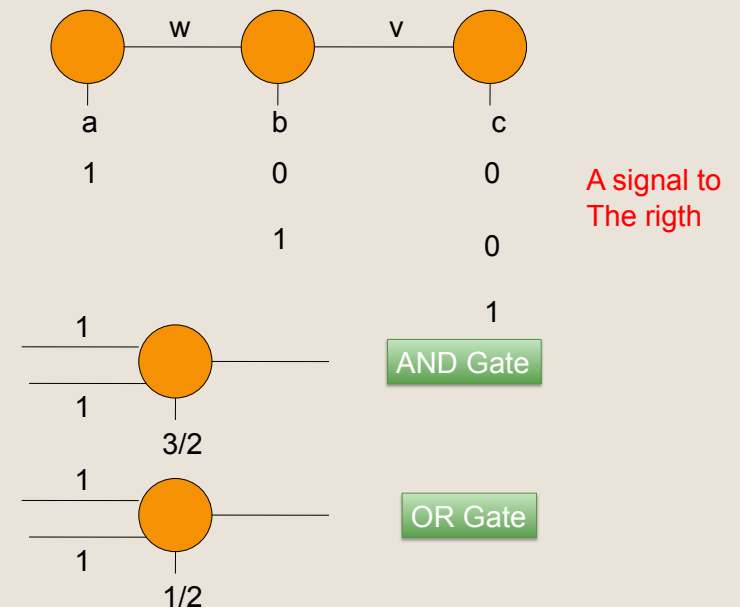
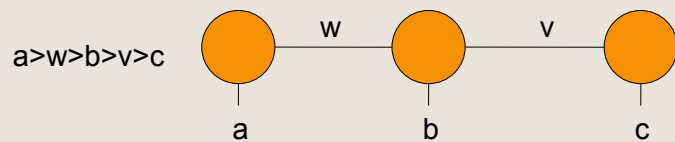
The class **NC**: problems which can be solved in a parallel machine (say a PRAM) in poly-logarithmic time by using a polynomial number of processors.

It's direct that NC is included in P (in a serial computer we may simulate a parallel one !!! But it is an open problem if $P \neq NC$).

A candidate to be intrinsically serial is to compute the truth value of a circuit (**CVP**): we have to do that layer by layer Without a big surprise one may probe that it is P-Complete, i.e. any other other problem in P can be reduced to it. It is also not difficult to prove that the monotone (only **AND** and **OR** gates) circuit problem (**MCVP**) remains P-Complete

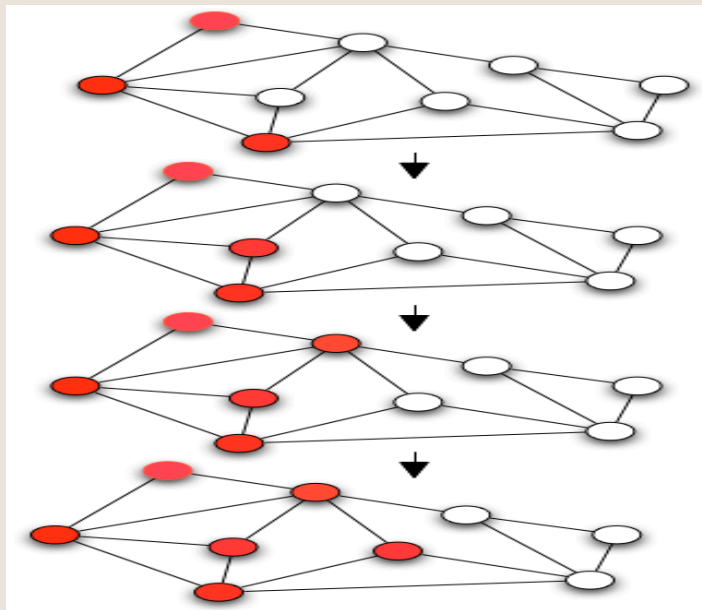
In this context we proved that before to reach the steady state symmetric neural networks could be very complex: i.e the decision problem if a given node will be 1 or 0 under a given dynamics is, in General P-Complete.

Convey of information in a symmetric network:



So any circuit can be coded in a neural net
By layers and to cross wires Ok Consider the
Graph in non-planar

Also one may define diodes, i.e
Configurations to give orientation to signals.



Bootstrap Percolation

Given a finite non oriented graph $G=(V,E)$

And an initial configuration of 0's and 1's

Consider the strict majority function operating at each node

What is the relationship between the graph and the
proportions of 1's such that iterated in parallel
every node will become 1?

Decision problem **PER**: given an initial
configuration and a specific node at value 0.
does there exist $T>0$ such that this
node becomes 1?

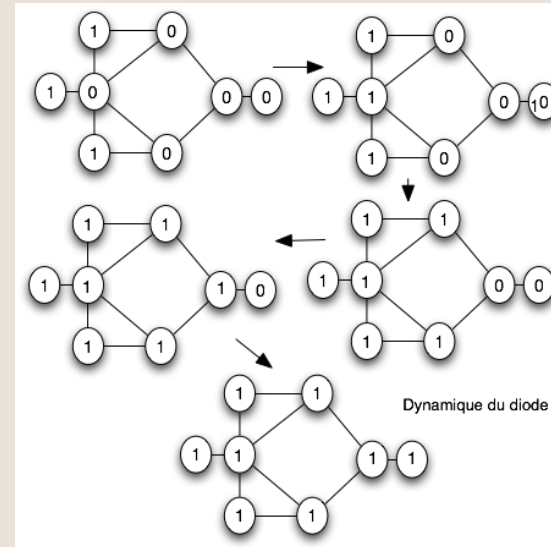
Theorem (Pedro Montealegre, E:G (1911))

If the graphs may have vertices with degree ≥ 5 ,
PER is P-complete.

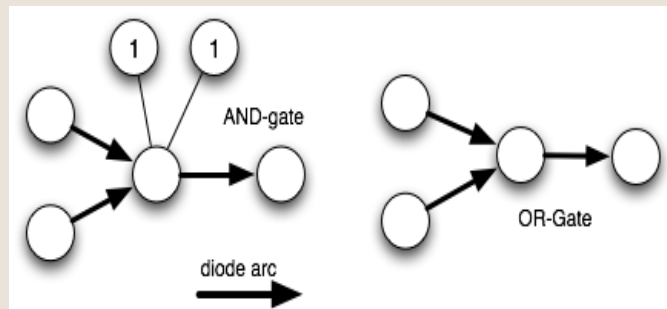
If the maximum degree ≤ 4 , **PER** belongs to NC

Clearly **PER** belongs to P, because in almost $O(n)$ steps the dynamics arrives to the steady state.

The proof of P-Completeness consist to simulate the monotone circuits behavior inside the strict majority dynamics.



Information
only flows to
the right



For the case maximum degree ≤ 4 one may reduce the problem to compute connected and biconnected components in the graph, which one may do in a PRAM in $O((\log n)^2)$

See Jajace ne pas une blague

The Schelling Segregation model

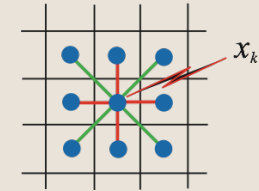


(Nicolás Goles-Domic, Sergio Rica, E.G.
PHYSICAL REVIEW E 83, 056111 (2011)
And work in progress.

The Model of Segregation by Shelling

Thomas C. Schelling (1969 - 1972)

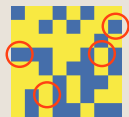
- Lattice one or two dimensional with periodic conditions
- State $x_k = \pm 1$
- Neighborhood Moore (green and red arrows) and von Neumann (red arrows)
- Tolerance threshold $\theta \in \{1, \dots, |V|\}$



Happiness threshold

An individual is unhappy if there are more than θ individuals on the other state in its neighborhood

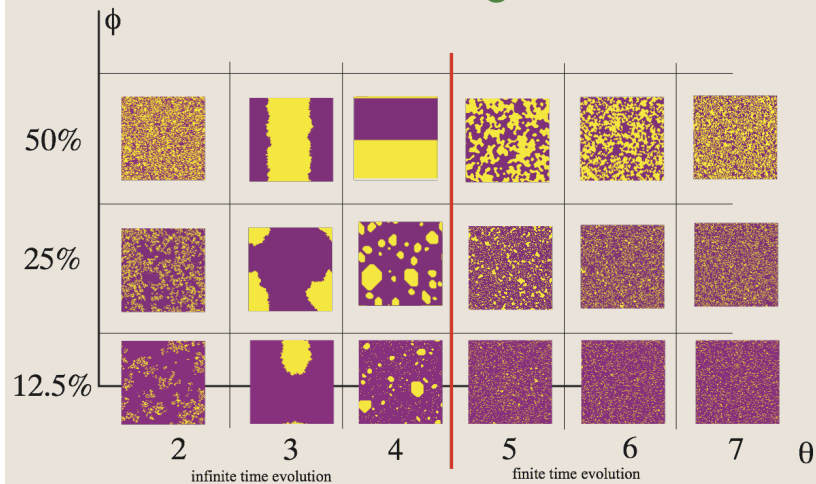
eg. For the Moore's neighborhood and $\theta = 5$ then :



The update rule

At each step, one lists the unhappy individuals of both species, and then randomly (for instance) one exchanges two individuals of opposite value.

Phase diagram for Moore's neighborhood



Comments

- A tendency of segregation.
- A tendency of a diminution of the interfaces
- But! there is a strong frustration.

Quantitative behavior

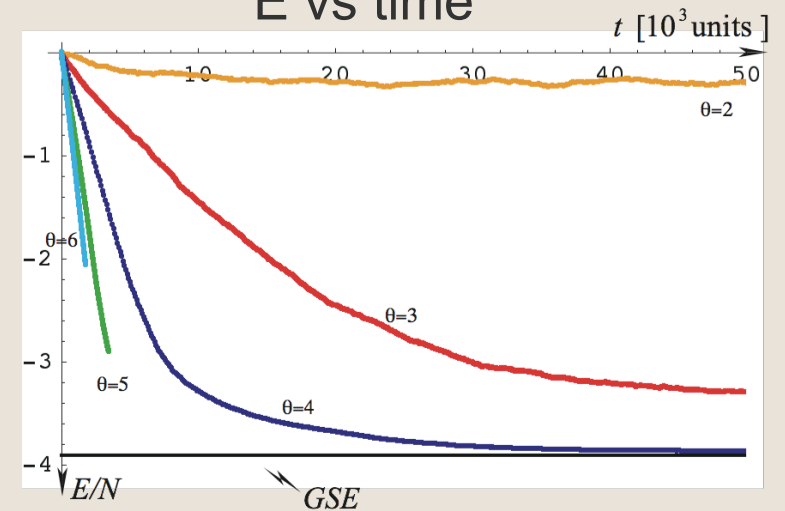
$\theta \geq 5$: the energy decreases

$$E[\{x\}] = -\frac{1}{2} \sum_{k=1}^N x_k \sum_{i \in V_k} x_i$$

In general, if V is the neighborhood, the energy decreases
If and only if

$$\theta > \frac{|V|}{2}$$

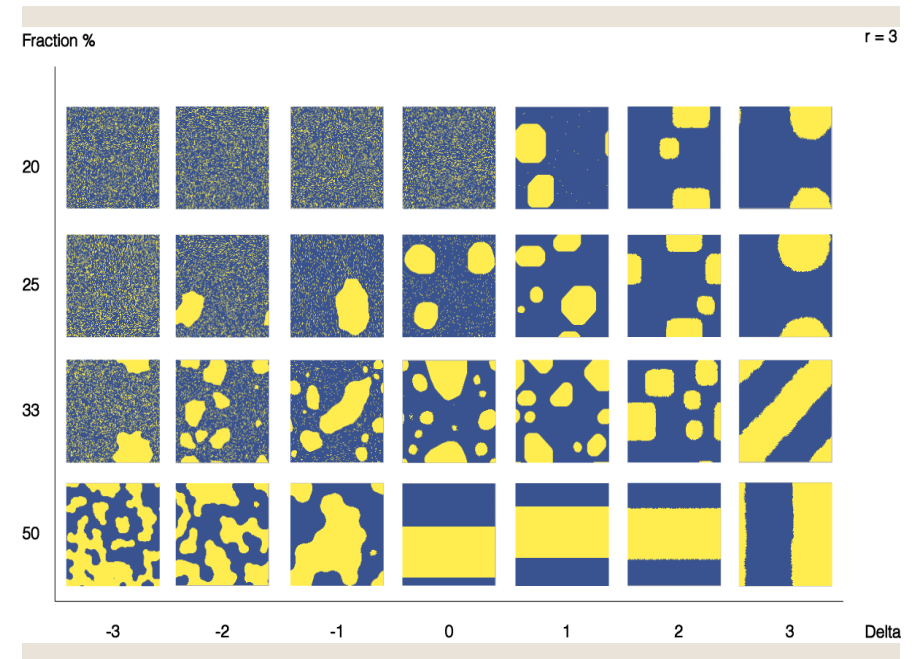
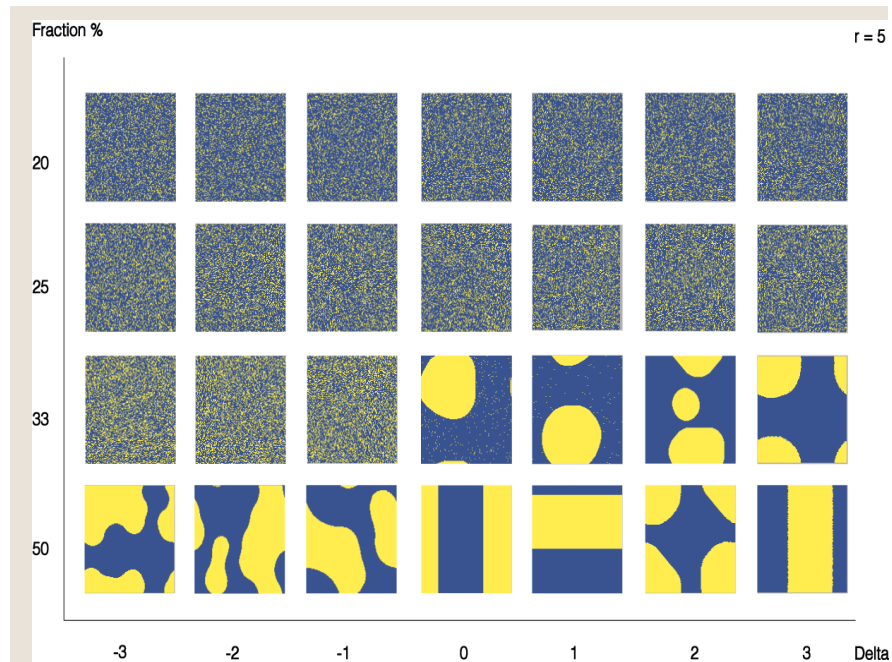
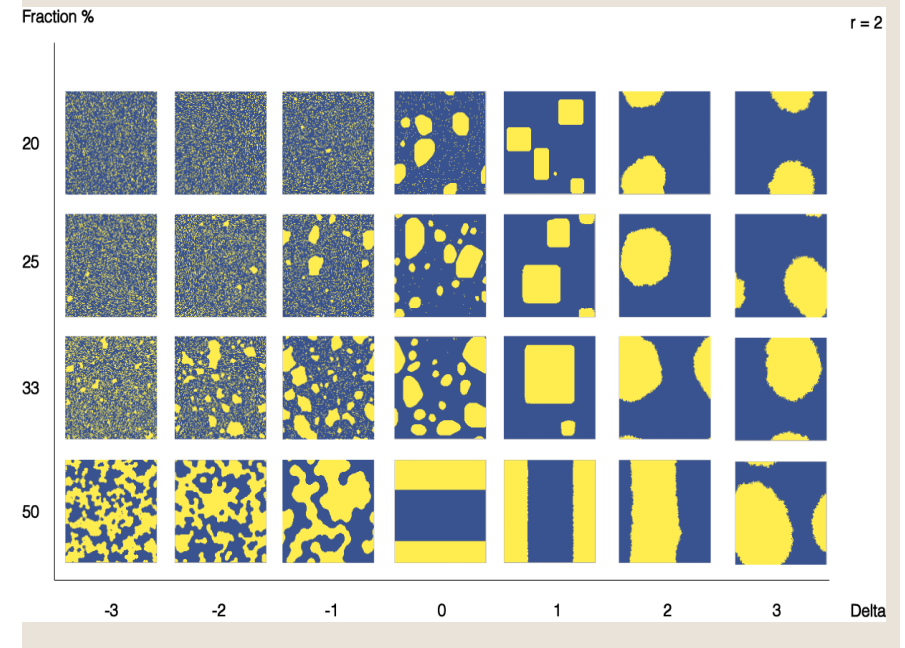
E vs time



Geometrical interpretation

It is easy to see by a transformation of the energy that
Minimize it, is equivalent to minimize the perimeter of
The clusters so the dynamics try to do that !!

Others phase diagrams with circle-neighborhoods
with different radios (Nicolas Goles-Domic Simulations):



Prediction, short-cuts and Computational Complexity

The real state problem: It is easy to know if
Some one would change house?

- The Real State Prediction problem (RSP)

- Will a site i , such that $x_i = -1$, have a non zero probability to change its state at some step $T \geq 1$?

We will first analyse the real state problem for one and the von Neumann neighborhood in two dimensions.



Nearest neighbors

For both parameters, $\theta \in \{1, 2\}$, RSP is easy to solve

For $\theta = 1$ belongs to NC

$$\theta = 1$$

-1 1 Boths are unhappy: swaps for $T = 1$

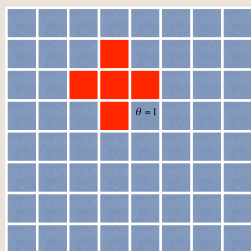
In general consider the nearest +1



So $P=0$ for $T < 4$ else $P > 0$

Two dimensions

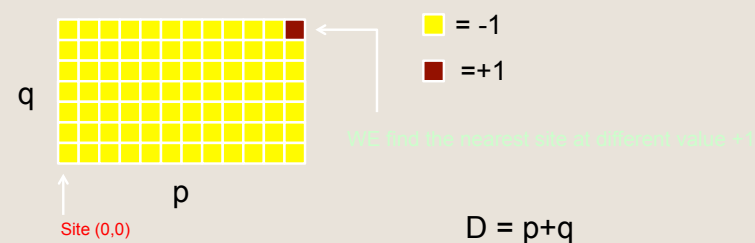
The von Neumann Neighborhood



$$\theta \in \{1, 2, 3, 4\}$$

Case $\theta = 1$ i.e. a site is unhappy iff there exists at least one neighbor in a different state

Further, in this case two neighbors in different state are both unhappy !!!



Clearly we may do it by a PRAM as we did in the one dimensional case

Case $\theta = 4$

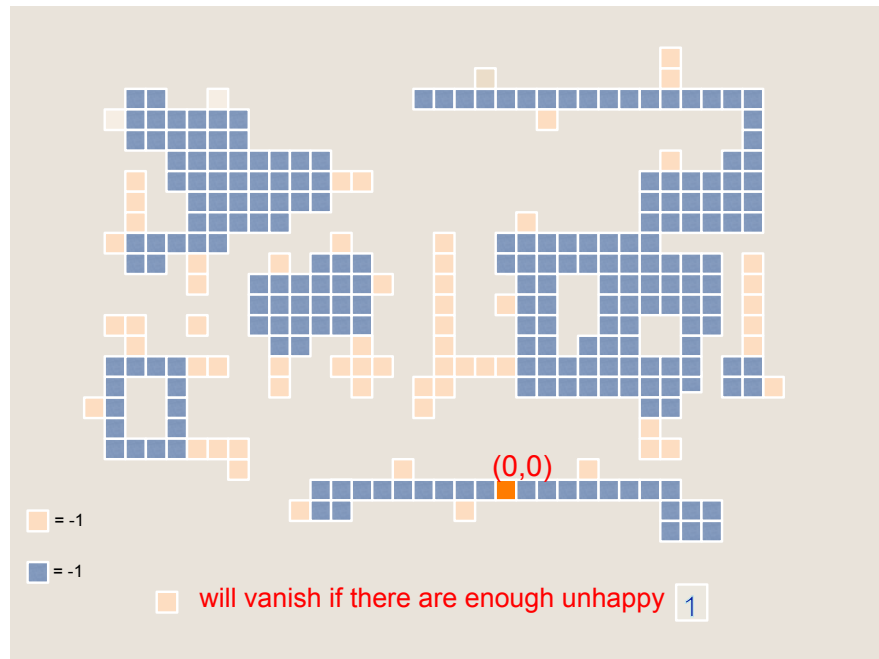
An unhappy site has to be in a very bad situation: every neighbor being in the other state

So we may now if there exists two unhappy people
In different state in $O(1)$

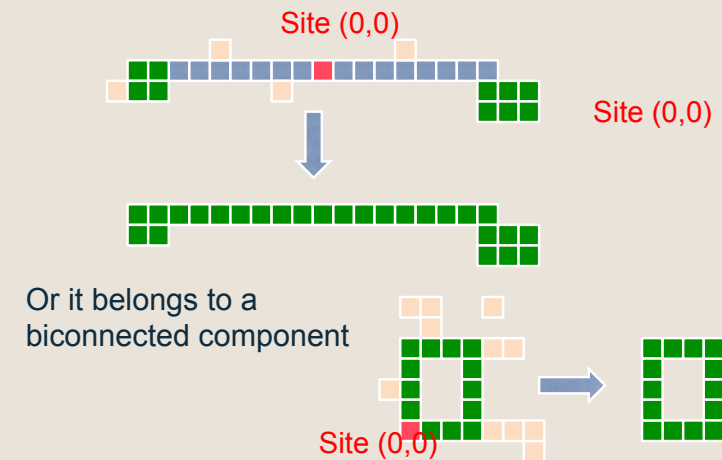
Case $\theta = 3$

Recall that for $\theta \in \{3, 4\}$ the operator E is an energy, so the dynamic converges to fixed points which are local minima of E .

A fixed component of, say -1 , is such that each element has at least two neighbors at the same state



So the site (0,0) at value -1 will never change if it belongs to a connected component such that there exist two different paths to stable clusters



The search of the connected component of -1's where site (0,0) belongs can be done in $O((\log(N))^2)$ with polynomial number of processor in a PRAM

Also one may compute biconnected components in $O((\log(N))^2)$

See JaJa's book et ce ne pas une blague !!!!

Finally we may compute the number of unhappy +1's in $O(1)$ with $O(N)$ processors

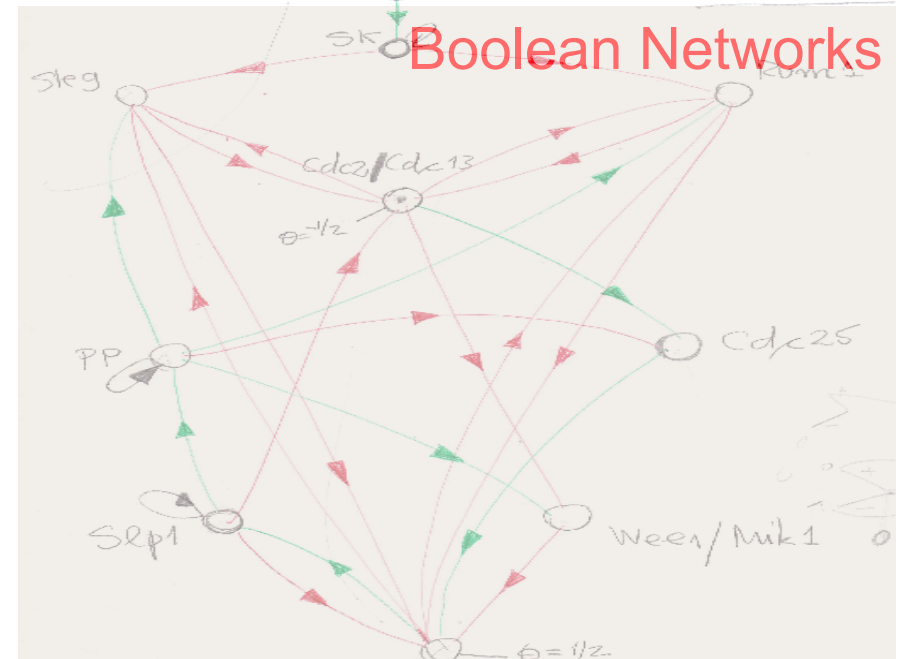
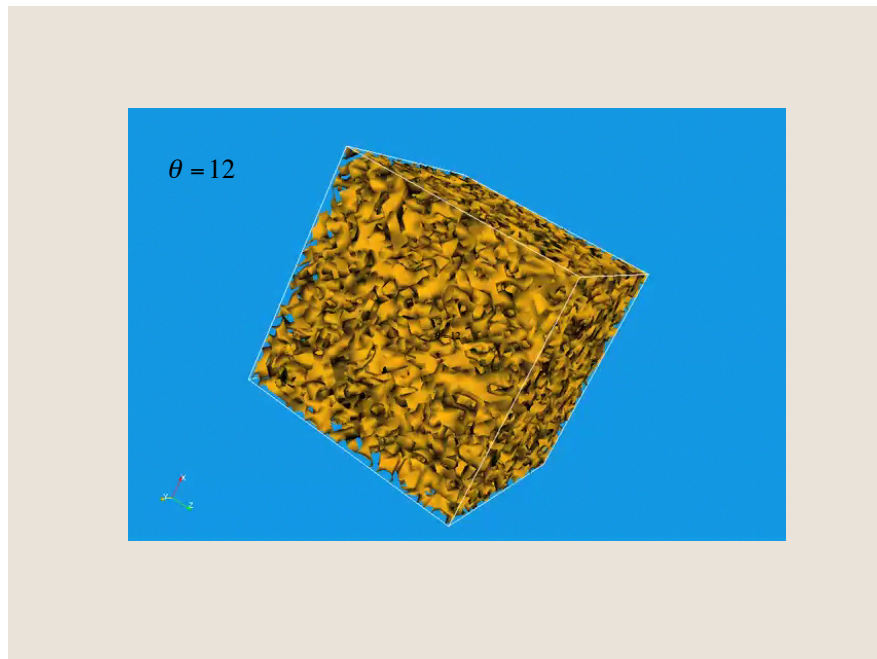
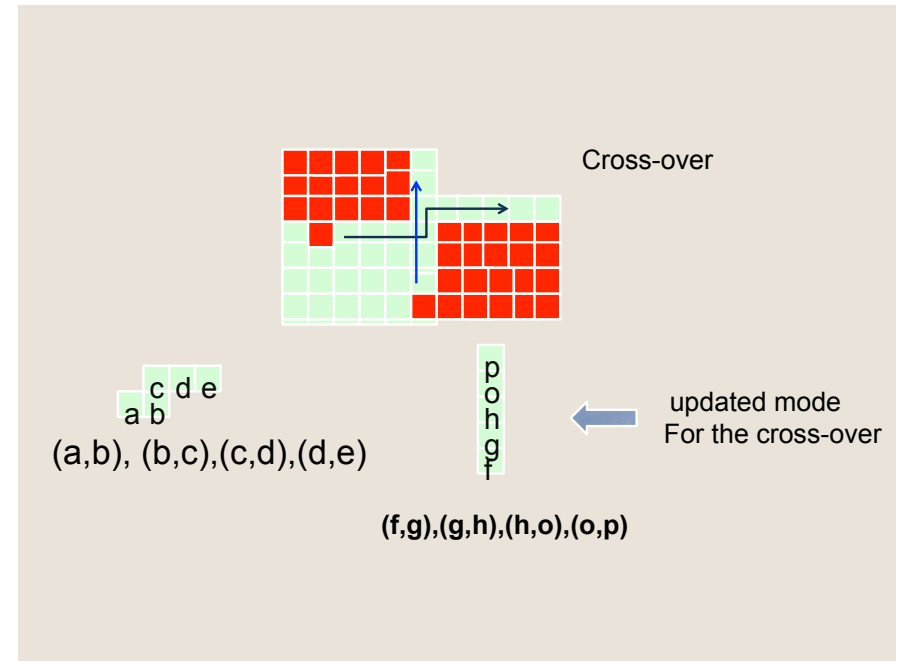
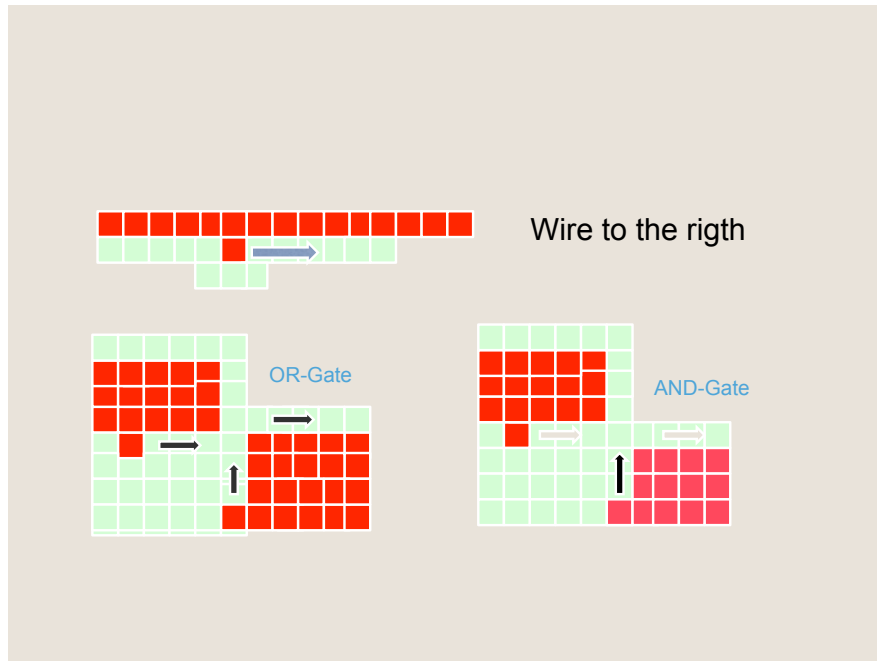
Remark: $N = n \times n$ the number of sites in the network

So the Schelling problem belongs to NC for $\theta \in \{1, 3\}$ and it is constant for $\theta = 4$

Now we have to see the complexity for $\theta = 2$

For $\theta = 2$ the segregation problem is P-Complete

It is in P because we will only accept nearest swaps (a,b) such that $d(a,b)=1$, so it is enough to compute the light-cone associated to the site (0,0)



History

- Stuart. Kauffman, Metabolic stability and epigenesis in randomly connected nets, J. Of Theor. Biol, 22, 437-67, 1969.
- François Robert, Discrete Iterations, Springer Verlag, 1986).

The results presented here were done in collaboration with Some colleagues and ph.d students:

Julio Aracena (Universidad de Concepción, Chile)

Andrés Moreira (Universidad Federico Santa Maria, Valparaíso, Chile)

Lilian Salinas (Universidad de Concepción, Chile)

Publications:

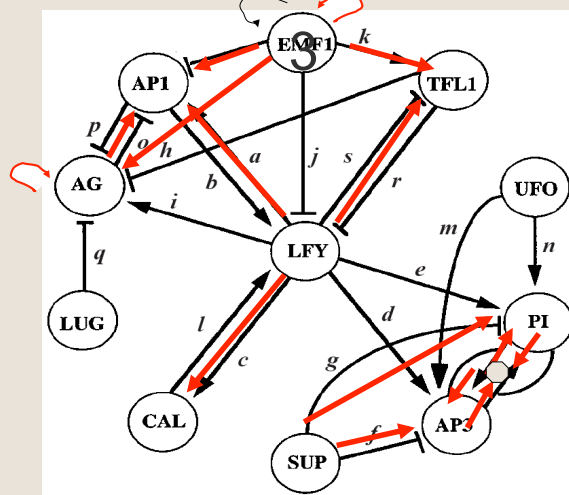
E. Goles, L.Salinas, Comparison between parallel and serial dynamics of boolean networks, in Theor. Comp. Sciences 347-53 (2008),

J.Aracena, E. Goles, A. Moreira, L.Salinas On the robustness of update schedules in boolean networks, in BioSystems, 97, 1-9, 2009.

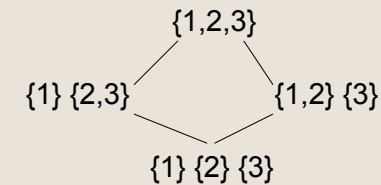
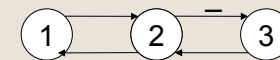
E.Goles, L.Salinas, Sequential Operator for filtering cycles in boolean networks, submitted to Advance in Applied Mathematics, 2009.

Actual motivation

bioinformatics



Genetic and Metabolic networks



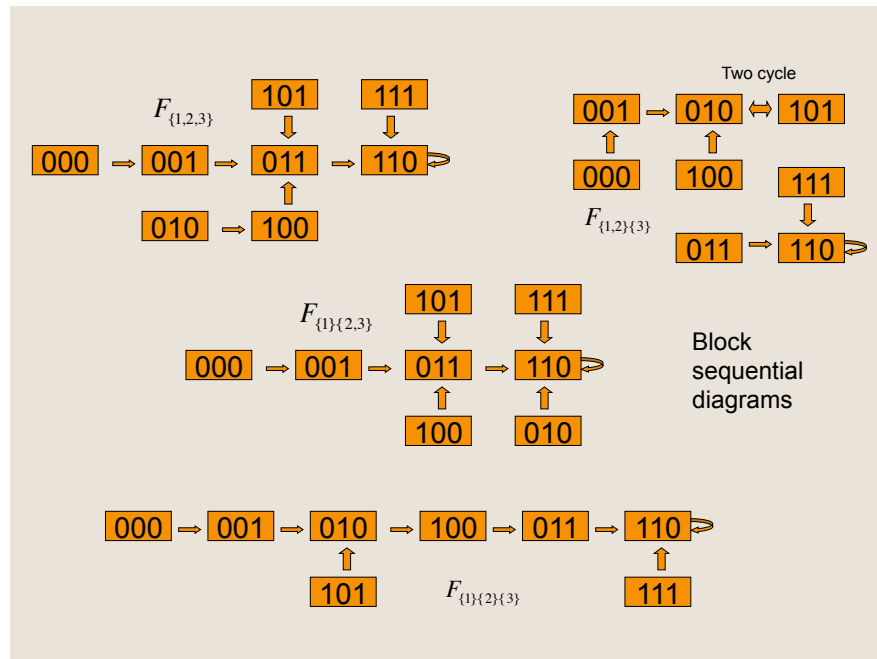
Block Sequential partitions for three elements

$$F_{\{1,2,3\}}(x_1, x_2, x_3) = (x_2, x_1 + x_3, \neg x_2)$$

$$F_{\{1,2\}\{3\}}(x_1, x_2, x_3) = (x_2, x_1 + x_3, (\neg x_1)(\neg x_3))$$

$$F_{\{1\}\{2,3\}}(x_1, x_2, x_3) = (x_2, x_2 + x_3, \neg x_2)$$

$$F_{\{1\}\{2,3\}}(x_1, x_2, x_3) = (x_2, x_2 + x_3, (\neg x_2)(\neg x_3))$$



Cycles in synchronous and serial Iterations

$$F : \{0,1\}^3 \rightarrow \{0,1\}^3$$

$$f_1(x_1, x_2, x_3) = x_2$$

$$f_2(x_1, x_2, x_3) = x_3$$

$$f_3(x_1, x_2, x_3) = x_1$$

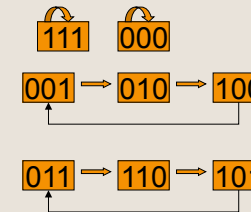


$$G : \{0,1\}^3 \rightarrow \{0,1\}^3$$

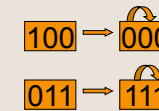
$$g_1(x_1, x_2, x_3) = x_2$$

$$g_2(x_1, x_2, x_3) = x_3$$

$$g_3(x_1, x_2, x_3) = x_2$$



Parallel update: 3-cycles



Serial update: 2-cycle

Theorem.

Consider a network with non-negative loops then the Cycles with period ≥ 2 , if there exists, are different for parallel and serial iteration.

i.e both iterations can not share non trivial cycles

Comparison between parallel and serial dynamics of Boolean networks
E. Goles, L. Salinas, T.C.S.

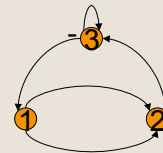
The hypothesis about loops is necessary. Consider:

$$F : \{0,1\}^3 \rightarrow \{0,1\}^3$$

$$f_1(x_1, x_2, x_3) = x_2$$

$$f_2(x_1, x_2, x_3) = x_1$$

$$f_3(x_1, x_2, x_3) = x_1 x_2 (\neg x_3)$$



It is a cycle for both iterations

There is an other way to encode different updates. Consider a network $N = (F, s)$; where F is the set of n local boolean functions and s is the "order" which nodes are updated.

That is to say s is a function from the set of nodes on itself.

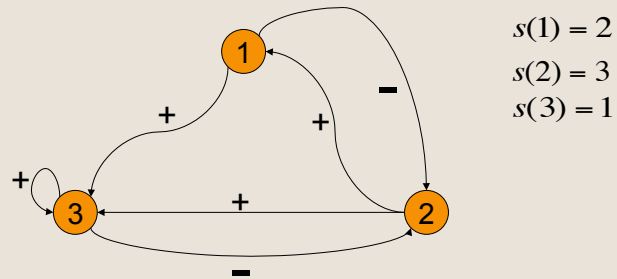
$$s : \{1, \dots, n\} \rightarrow \{1, \dots, n\}$$

Such that $s(i) < s(j)$ means node i -th is updated before node j -th

From that we may define a signed graph. To the graph G , associated to F we define $G(s)$ as follows:

$$\text{sgn}(i, j) = +1 \text{ if } s(i) \geq s(j)$$

$$= -1 \text{ if } s(i) < s(j)$$



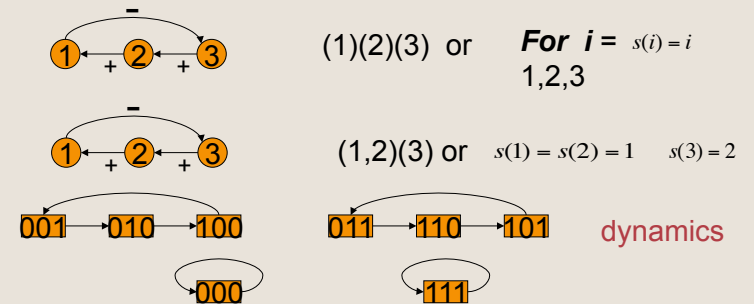
The 3-th node is updated first; the first is updated the second
And the second is the last to be updated. The iteration corresponds
to the serial update (3)(1)(2)

Given two iteration modes on a same
boolean function, i.e.

(F, s_1) and (F, s_2)

If they have the same signed graph : $G_{s_1}^F = G_{s_2}^F$

Then they have the same dynamics



FILTERS

A filter G associated to a boolean network F corresponds
to the recursive application of an iteration mode, S , to F :

$$G = \lim_{p \rightarrow \infty} S^p(F)$$

We will consider S the serial update:

$$S = \{1\} \{2\} \dots \{n\}$$

Since F is finite S converges to a network G
which we call the filter.

- Example: consider the function $F(x_1, x_2, x_3, x_4) = (x_2, x_3, x_4, x_1)$

and the serial update $S = \{1\} \{2\} \{3\} \{4\}$

$$F^0 = F(x_1, x_2, x_3, x_4) = (x_2, x_3, x_4, x_1)$$

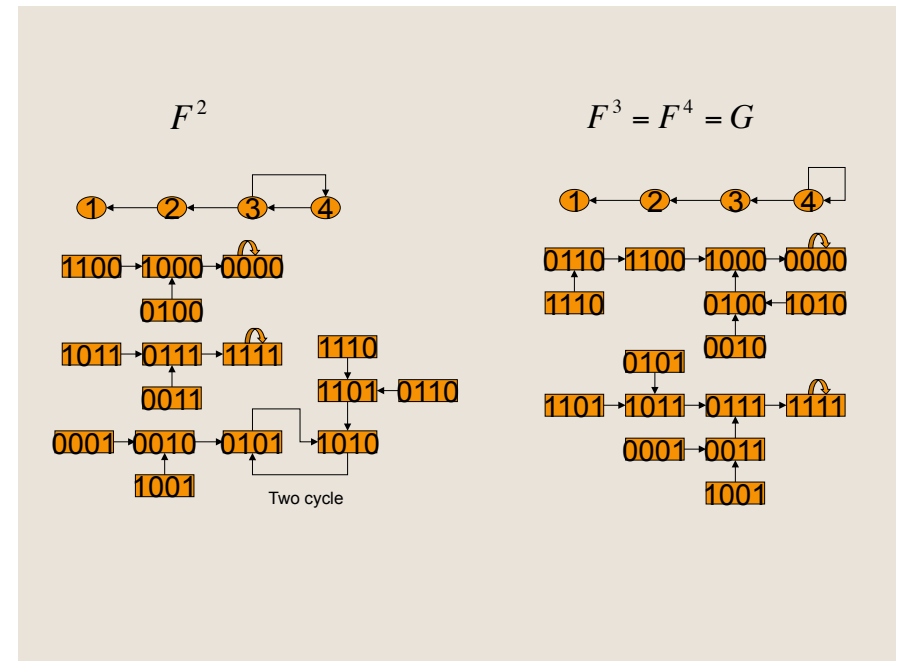
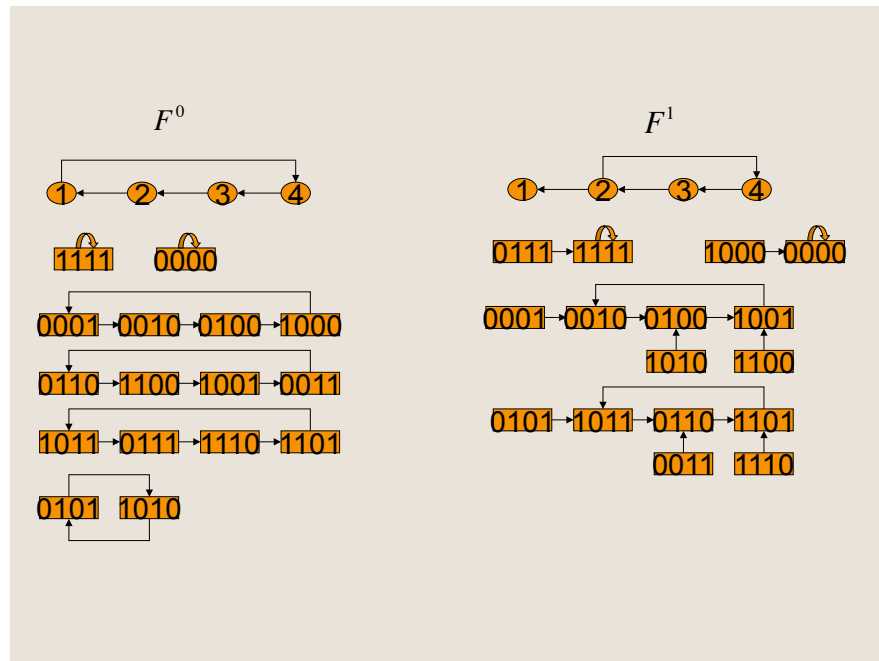
$$F^1 = S(F^0) = (x_2, x_3, x_4, x_2)$$

$$F^2 = S(F^1) = S^2(F^0) = (x_2, x_3, x_4, x_3)$$

$$F^3 = S(F^2) = S^3(F^0) = (x_2, x_3, x_4, x_4)$$

$$G = F^4 = S(F^3) = S^4(F^0) = (x_2, x_3, x_4, x_4)$$

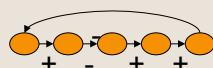
The function G is the filter and fixed point of the procedure



THEOREM

- Given a monotone boolean network then the serial filter converges in $o(n)$ to a network G without cycles in its dynamics

This result can be extended to networks such that its circuits are non-negative.



Communication Complexity on Cellular Automata

- We will present some results about communication complexity for one dimensional C.A.

This work was done with the colaboration:

P.E. Meunier (Ph.D student ENSL- France)

I. Rapaport (DIM, U. De Chile)

G. Theyssier (Univ. de Savoie, CNRS, France)

E:G

1. Communication Complexity in CA

3. Examples

3. Problems PRED and INV

4 Application to rule 218

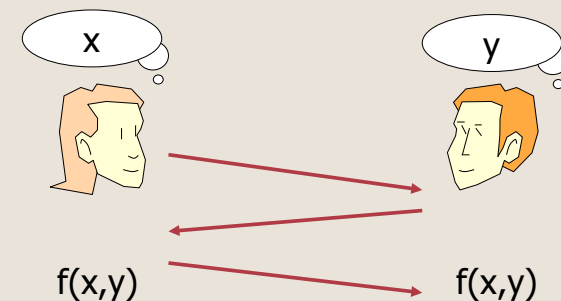
5 Intrinsic Universality and C.C

6 Applications

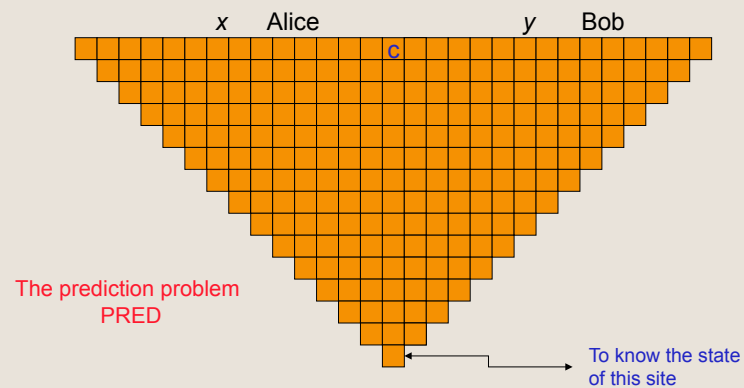
• References:

- C.Durr, I Rapaport, G.Theyssier, C.A and Communication Complexity, TCS 290/3,355-368, 2003
- E. Goles, C. Little,I. Rapaport, Understanding a non-trivial C.A. by finding its simplest underlaying communication protocol, in S.H Hong, H Nagamochi (eds) Oprocc 19th. Int. Symposium in Computer Science (ISAAC2008), LNCS 5369, vol 2380, Springer, 2008.
- E. Goles, P.E. Meunier, I. Rapaport, G. Theyssier, communication complexity and intrinsic universality in cellular automata, to appear in TCS,2009

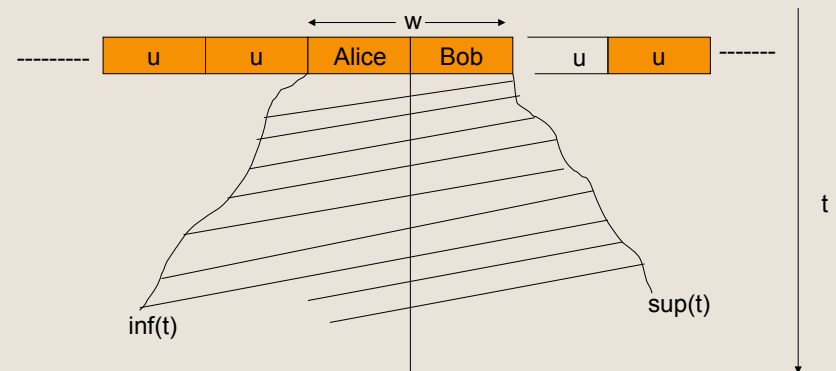
Def: necessary number of communication bits in order to compute a function when each party knows only part of the input



We will present two communication complexity problems related with CA:
The prediction problem (PRED) and the Invasion problem (INV).

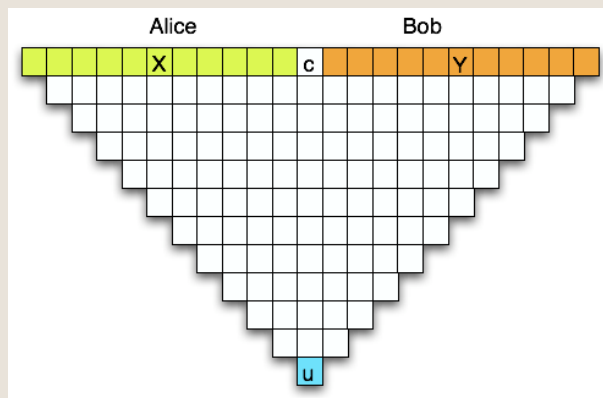


The invasion problem: INV(u)

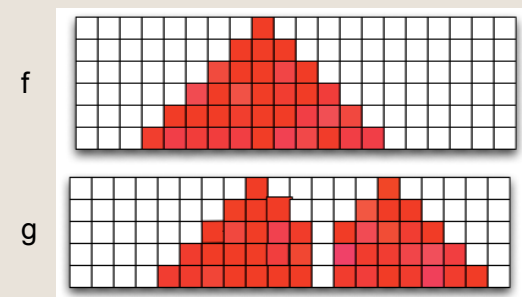


Return to the PRED problem

Information to be shared by Alice and Bob
X, Y are binary vectors, c is 0 or 1



f	g
000 0	000 0
001 1	001 1
010 1	010 1
011 1	011 1
100 1	100 1
101 1	101 0
110 1	110 1
111 1	111 1



One way protocole for a rule f :
The minimum bit information to be send by A (B) to B (A)

Def: Let $M_n(c)$ the $2^n \times 2^n$ matrix

$$(M_n(c)) = (f^n(x, c, y))$$

Theorem: the number of different rows or columns is a lower bound for the size of the one way protocole

In Communication Complexity E. Hushilevitz, N. Nisan,
Cambridge University Press 1997

The communication complexity for additive rules is $O(1)$

In fact, given the vector x, y and c
Since f is additive:

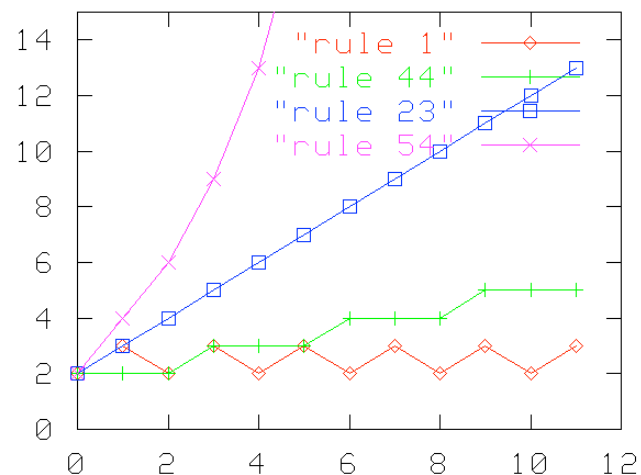
It suffices that Alice send the bit:

$$f^n(x, c, \vec{0})$$

From additivity:

$$f^n(x, c, y) = f^n(x, c, \vec{0}) + f^n(\vec{0}, 0, y)$$

Matrix behavior of different rules



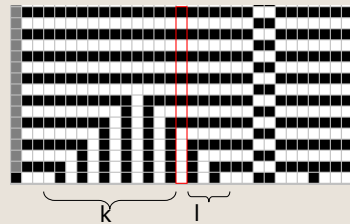
Rule 23

000 1
001 1
010 1
011 0
100 1
101 0
110 0
111 0

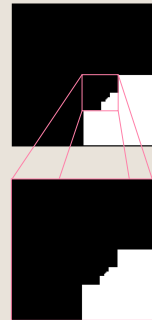
The idea for the protocol is to remark that more than
Two consecutives zeros or ones ALTERNATES !!!

00 11
11 00

Linear grow of the non-linear Rule 23



Matrix



Protocol:

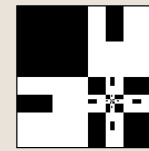


communicates k



knows that after $\min(k, l)$ the cell alternates

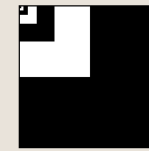
Others examples



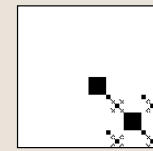
Rule 33



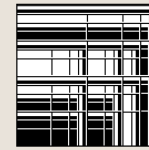
Rule 44



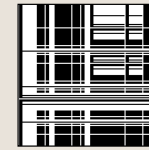
Rule 50



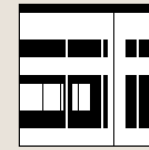
Rule 164



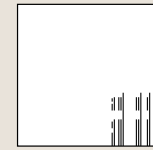
Rule 184



Rule 14



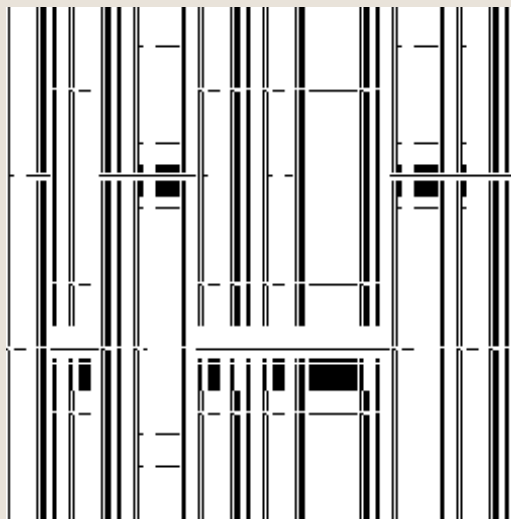
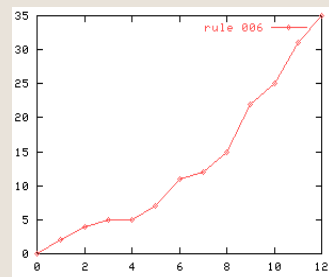
Rule 35



Rule 168

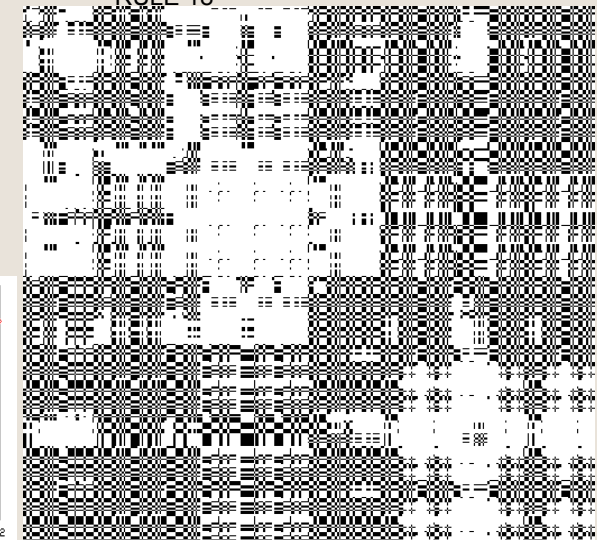
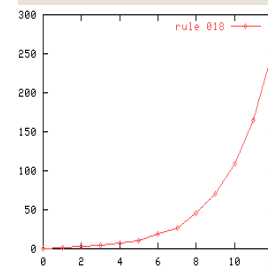
RULE 6

000 0
001 1
010 1
011 0
100 0
101 0
110 0
111 0



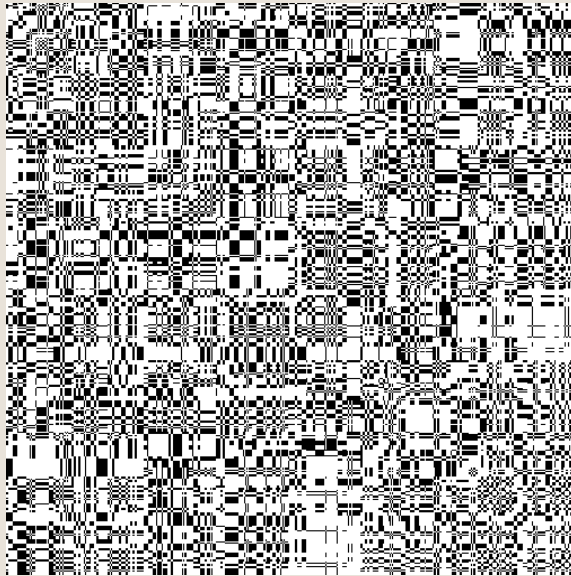
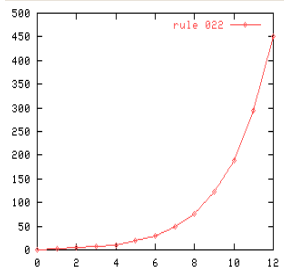
RULE 18

000 0
001 1
010 0
011 0
100 1
101 0
110 0
111 0



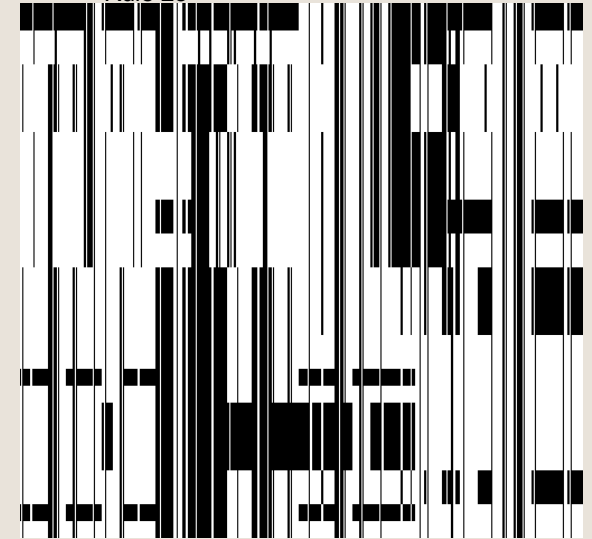
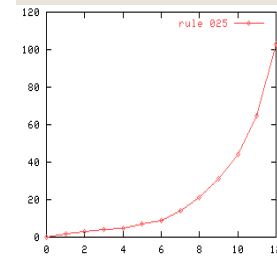
Rule 22

000 0
 001 1
 010 1
 011 0
 100 1
 101 0
 110 0
 111 0



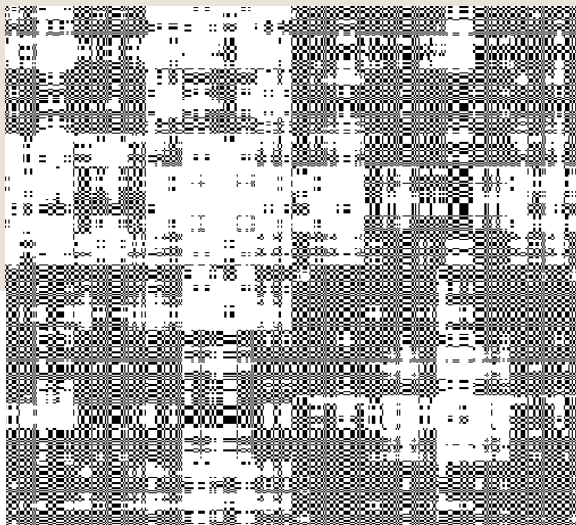
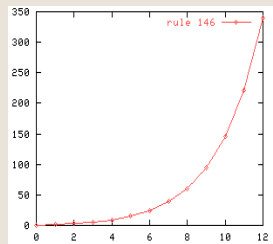
Rule 25

000 1
 001 1
 010 0
 011 0
 100 1
 101 0
 110 0
 111 0



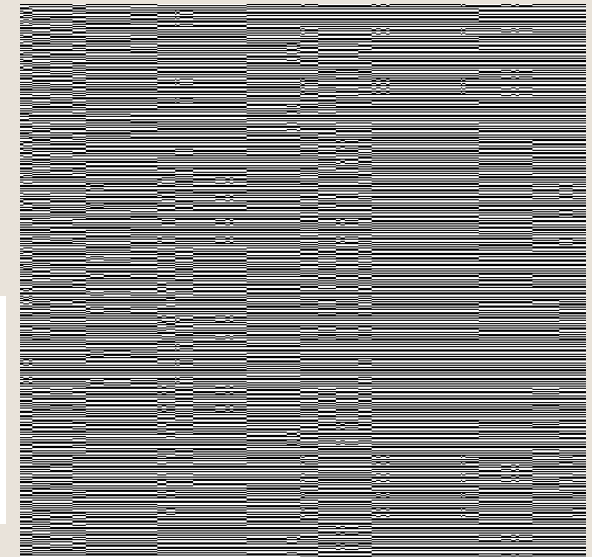
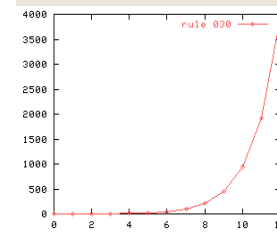
Rule 146

000 0
 001 1
 010 0
 011 0
 100 1
 101 0
 110 0
 111 1



Rule 30

000 0
 001 1
 010 1
 011 1
 100 1
 101 0
 110 0
 111 0



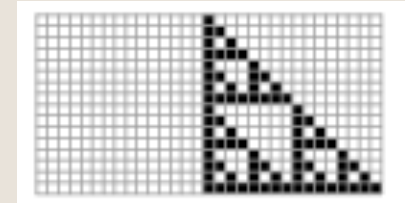
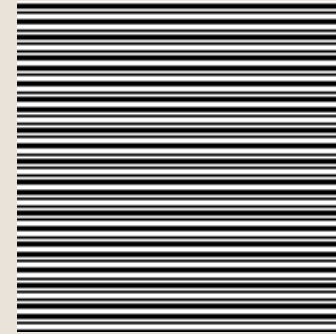
CC1 = constant number of different rows or columns.

0, 1, 2, 3, 4, 5, 7, 8, 10, 12, 13, 15, 16, 17, 19, 21, 24,
27, 28, 29, 31, 32, 34, 36, 38, 39, 40, 42, 46, 48, 51, 52, 53, 55,
60, 63, 64, 66, 68, 69, 70, 71, 72, 76, 78, 79, 80, 83, 85, 87, **90**,
93, 95, 96, 102, 105, 108, 112, 116, 119, 127, 128, 130, 132,
136, 138, 139, 140, 141, 144, 150, 152, 153, 154, 155, 156,
157, 160, 162, 165, 166, 170, 171, 172, 174, 175, 176, 180,
186, 187, 189, 190, 191, 192, 194, 195, 196, 198, 199, 200,
201, 202, 204, 205, 206, 207, 208, 209, 210, 211, 216, 219,
220, 221, 223, 228, 231, 236, 237, 238, 239, 240, 241, 242,
243, 244, 245, 246, 247, 250, 251, 252, 253, 254, 255.

CC2: linear number of different rows or columns:

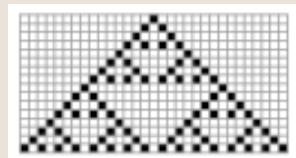
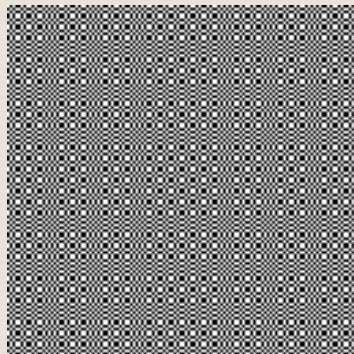
11, 14, 23, 33, 35, 43, 44, 47, 49, 50, 56, 58, 59, 77, 81, 84, 98,
100, 113, 114, 115, 117, 142, 143, 164, 168, 177, 178, 184,
185, 188, 197, 203, 212, 213, 217, 222, 227, 232, 234, 235,
248, 249.

RULE 60



CLASS 1
CC1

RULE 90



CLASS 1
CC1

CLASS 3
WOLFRAM

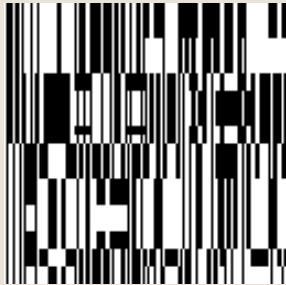
CC3: other polynomials

6, 9, 18, 20, 22, 25, 26, 37, 41, 54, 57, 61, 62, 65, 67, 73, 74,
82, 86, 88, 89, 91, 94, 97, 99, 101, 103, 104, 106, 107, 109,
110, 111, 118, 121, 122, 123, 124, 125, 126, 129, 131,
133, 134, 137, 145, 146, 147, 148, 149, 151, 158, 159,
161, 163, 167, 169, 173, 179, 181, 182, 183, 193, 214,
215, 218, 226, 229, 230, 233.

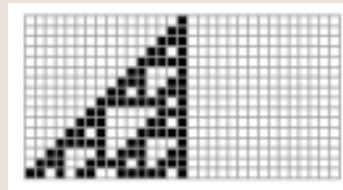
CC4: exponential:

30, 45, 75, 120, 135, 225

RULE 110

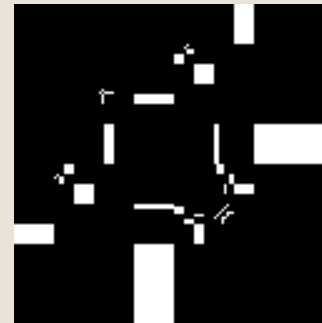


CLASS 3

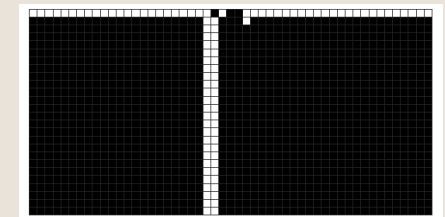


CLASS 4
(WOLFRAM)

RULE 233

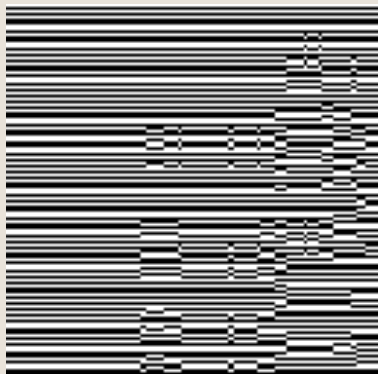


CLASS 3

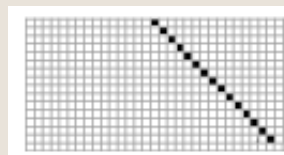


CLASS 1
(Wolfram)

Rule 120



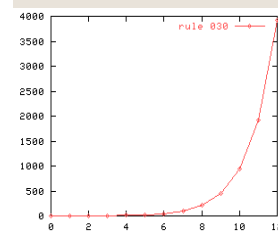
Class 4



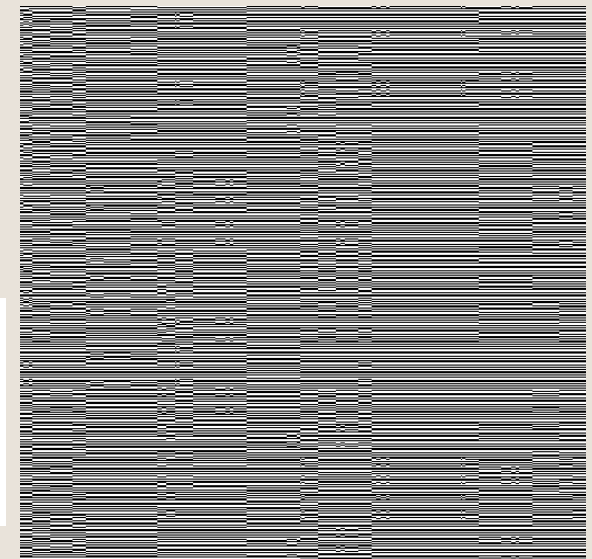
000 0
001 0
010 0
011 1
100 1
101 1
110 1
111 0

Rule 30

000 0
001 1
010 1
011 1
100 1
101 0
110 0
111 0



Class 4



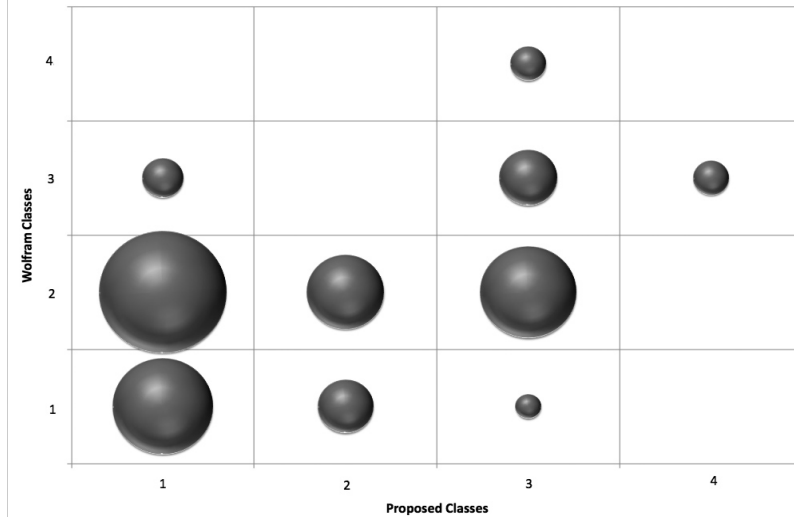
Wolfram's Classification (aprox)

W1: 0,4,8,12,13,32,36,40,44,64,,68,69,72,76,77,78,,79,92,93,96,100,104,128,132,136,140,141,160,164,168,172,192,196,197,200,202,203,204,205,206,207,216,219,219,220,221,222,223,224,228,232,233,234,235,236,237,238,239,248,249,250,251,252,253,254,255

W2: 1,2,3,5,7,9,10,11,14,15,16,17,19,20,21,23,24,25,26,27,28,29,31,33,34,35,37,38,39,41,42,43,46,47,48,49,50,51,52,53,55,56,57,58,59,61,62,63,65,66,67,70,71,73,74,80,81,82,83,84,85,87,88,91,94,95,97,98,99,103,107,108,109,111,112,113,114,115,116,117,118,119,121,123,125,127,130,131,133,134,138,139,142,143,144,145,148,152,154,155,156,157,158,159,162,163,166,167,171,173,174,175,176,177,178,179,180,181,184,185,186,187,188,189,190,191,194,198,199,201,208,209,210,211,212,213,214,215,226,227,229,230,231,240,241,242,243,244,245,246,247

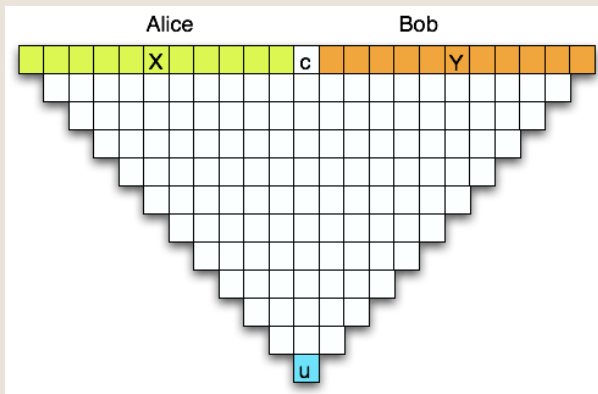
W3: 18,22,30,45,60,75,86,89,90,101,102,105,106,120,122,126,129,135,146,149,150,151,153,161,165,169,182,183,195,218,225
W4: 54,110,124,137,147,193

Class Criteria Comparison Table



Proposed classes for the pred

The prediction problem for rule 218

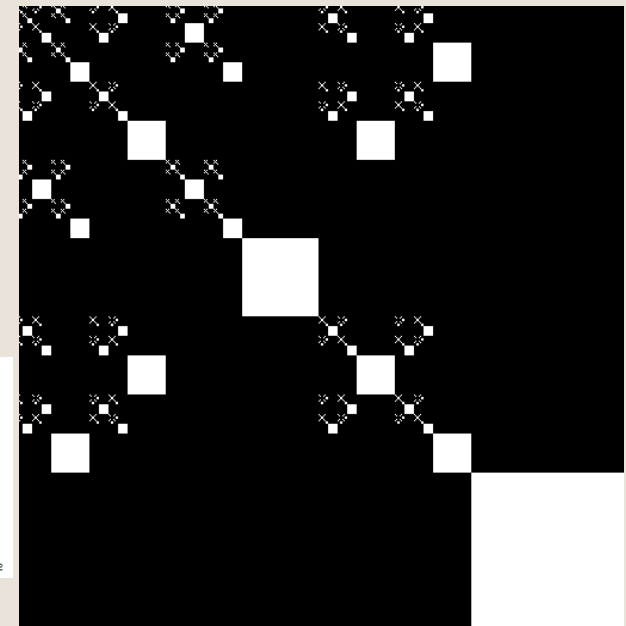
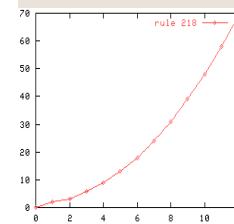


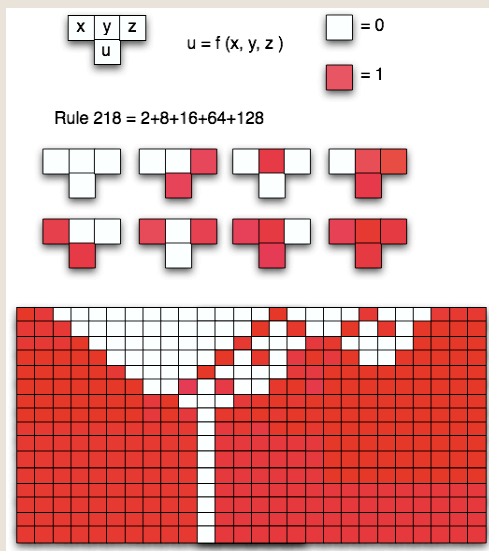
Minimun information send by Alice such that Bob computes

$$u = f_{218}^n(x, c, y)$$

Rule 218

000 0
 001 1
 010 0
 011 1
 100 1
 101 0
 111 1
 112 1

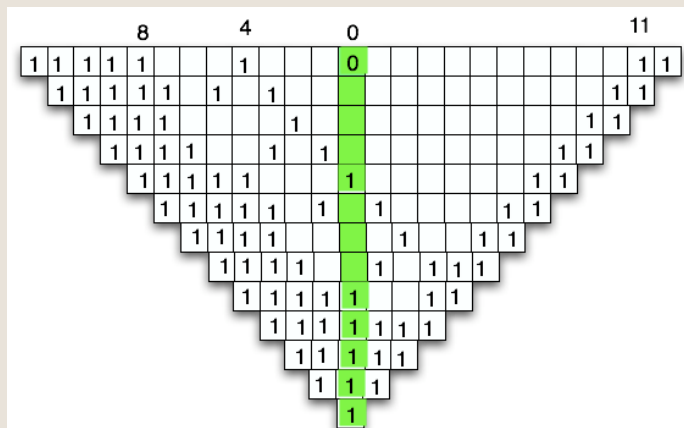




218's dynamics



Example for rule 218



Remarks

If the ones are isolated and every couple is separated by an odd number of zeros the rule 218 becomes additive.

In this case its behavior is like the rule 90.

$$f(a, b, c) = f_{90}(a, b, c) = a \oplus c$$

For additivity we have $f^n(x, c, y) = f^n(x, c, \vec{0}) \oplus f^n(\vec{0}, 0, y)$

$$f^n(x, c, y) = f^n(x, c, \vec{0}) \oplus f^n(\vec{0}, 0, y)$$

So Alice on send the bit

$$f^n(x, c, \vec{0})$$

So the protocol is constant for additive rules

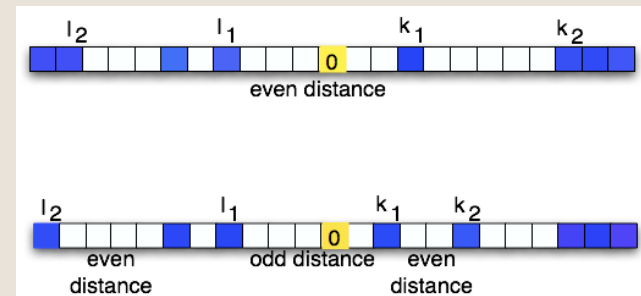
Definition of indexes for the 218 protocol

A crucial observation

Two or more ones remain invariant by the rule application

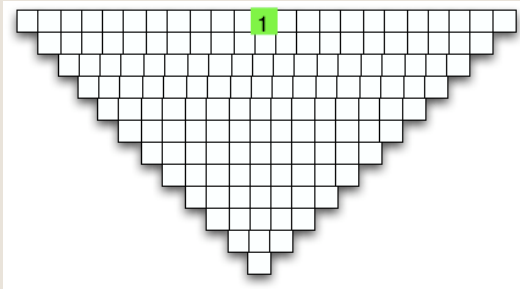
11
11

Definition of indexes for the protocol



l_1 Is the first one from the center cell

l_2 Is the first 1 such that the distance with the previous 1 from the center is even OR the next position is also a 1



Clearly in this situation one does not need the position of the first one in order to send it to Bob. Since the center is 1 the parity (even or odd) only depends of each side

So for the protocol It is enough to send only the second index l_2

Also in this case the protocol is optimal. One may exhibit A linear number of different rows in the $M(1,n)$ matrix.

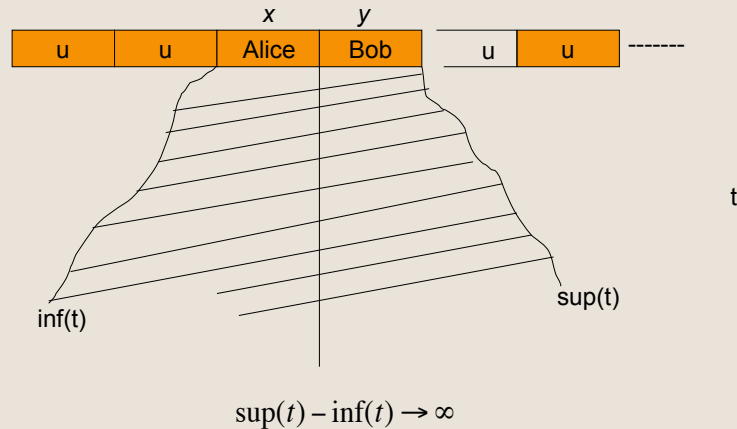
So, for the PRED problem $CC(218) = 2\log(n) + Cte$

Rule 218 is the first to have a quadratic number of rows (i.e a 2 indexes protocol) when the center is 0

and a linear one (1 index protocol) when the center is 1.

Now let see the communication complexity Of CA 218 for the INVASION problem

We have the configuration



Some remarks

1. A configuration is additive iff it does not contains walls.
or 1's at even distance. Further, additivity is preserver by the rule..

$$f(a,b,c) \neq f(1-a,b,c)$$

2. For abc additive: and

The protocol idea: $f(a,b,c) \neq f(a,b,1-c)$

if u is non-additive then Bob or Alice decide alone (nothing to communicate) because the dynamics of xy remains bounded between the walls in u

If u is additive either: $\dots u u u u u u u u \dots = \dots u u u u u u x y u u u u u u \dots$

Which can be done with 1 bit of communication.

Or They are different. In this case (directly from (2)):

$$\inf(t) = \inf(0) - t$$

and

$$\sup(t) = \sup(0) + t$$

Intrinsic Universality

$F \triangleleft G$: F is a subautomata of G If and only if

There exists an injective map $\phi : Q_F \rightarrow Q_G$

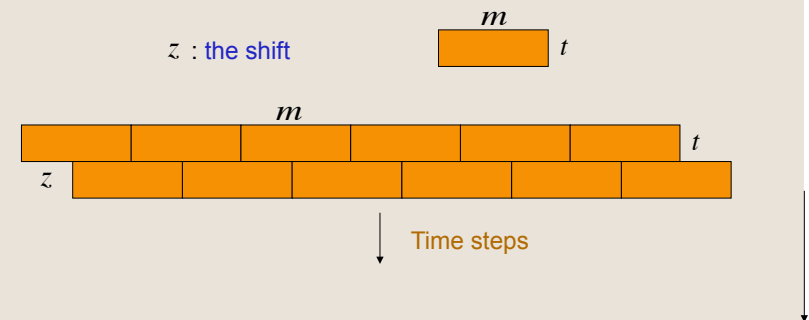
$$\phi_{ext} : Q_F^Z \rightarrow Q_G^Z \text{ such that } \phi_{ext} \circ F = G \circ \phi_{ext}$$

$$Q_F = \{a,b,c\} \text{ and } Q_G = \{d,e,h,l,m\} \quad \begin{array}{l} a \xrightarrow{\phi} d \\ b \rightarrow e \\ c \rightarrow h \end{array}$$

$$\begin{array}{cc} a b c a b & d e h d e \\ \downarrow F & \downarrow G \\ b a a c a & e d d h d \end{array}$$

$F \leq G$: F is simulated by G

$$\exists m_1, t_1, z_1; m_2, t_2, z_2 \text{ such that } F^{(m_1, t_1, z_1)} \triangleleft G^{(m_2, t_2, z_2)}$$



An automaton U is intrinsically Universal iff

$$\forall F; F \leq U$$

Some facts:

1. The maximum communication complexity $\Omega(n)$ corresponds to Alice sending the whole vector:
2. There exist automata such that for the PRED and INV problems have maximum communication complexity: $\Omega(n)$
3. The relation \leq preserves the communication complexity

Theorem: F intrinsically Universal implies
CC(PRED) and CC(INV) are

$$\Omega(n)$$

Some consequences

Additive CA's are not I.U

Positive Expansive CA's are not I.U

Rule 218 is not I.U

As well as rule 94, 184, 33 etc