Two State Totalistic Freezing Cellular Automata

{diego.maldonado, pedro.montealegre and nicolas.ollinger }@univ-orleans.fr
eric.chacc@uai.cl

Cellular Automata (CA)

Is a function over the colored grid $\mathbb{Z}^2$ defined locally by the syncrhonous application of a local rule f. Formally, is $F : Q^{\mathbb{Z}^2} \to Q^{\mathbb{Z}^2}$:

$$\forall z \in \mathbb{Z}^2, F(c)_z = f(c_{N+z})$$

Where $N \subset \mathbb{Z}^2$ is called the neighborhood.

Freezing

A *freezing CA* (FCA) is a CA F compatible with a partial order $\geqslant$ on states:

$$F(c)_z \geqslant c_z \qquad \forall z \in \mathbb{Z}^2 \quad \forall c \in Q^{\mathbb{Z}^2}$$

Example: Life without death.

Freezing+Totalistic

The family of FTCA with 2 states and five neighbors is given by:

$$f\left(\begin{array}{|c|c|c|}\hline & n_1 & \\ \hline n_4 & c & n_2 \\ \hline & n_3 & \\ \hline\end{array}\right) = \begin{cases} 1 & \text{if } c = 1 \\ 1 & \sum_{j=1}^{4} n_j \in I \\ 0 & \text{otherwise} \end{cases}$$

Where $I \subseteq \{0, 1, 2, 3, 4\}$.

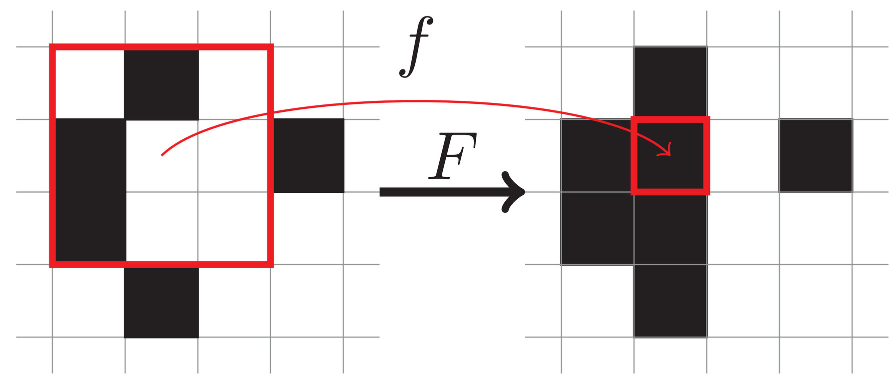
Notation: We call a TFCA by a number, representing the elements of I.

There are 32 possible TFCA.

Example The TFCA 24 is the TFCA that change to 1 with exactly 2 or 4 neighbors in state 1.

Example CA: Life without death

f: $\square \to \blacksquare$ with exactly 3 alive neighbors ($\blacksquare$).

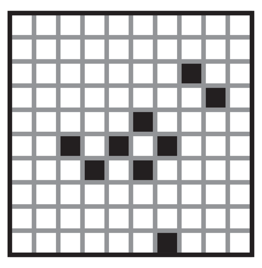


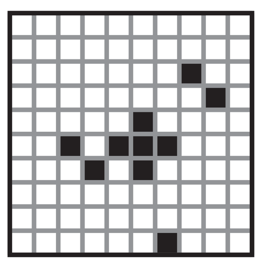
Totalistic

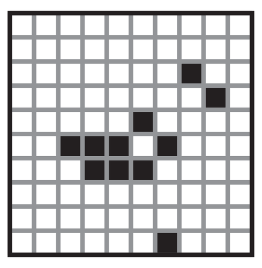
A *totalistic CA* (TCA) is a CA F where the LF only depend of the center cell and the total neighborhood:

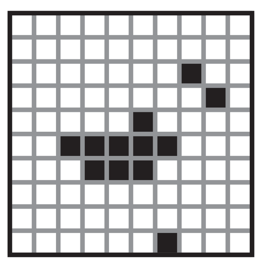
$$F(c)_z = f\left(c_z, \sum_{z \in N} c_z\right) \qquad \forall z \in \mathbb{Z}^d,\ \forall c \in Q^{\mathbb{Z}^d}$$

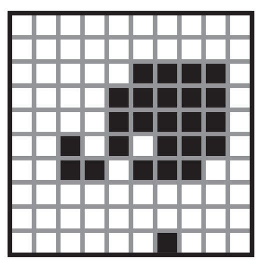
Dynamics, fifth iteration

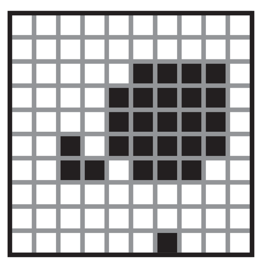

Initial

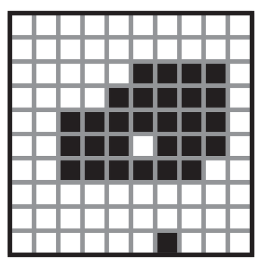

Rule 4

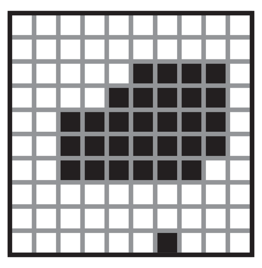

Rule 3

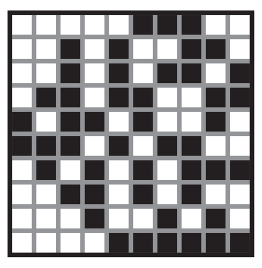

Rule 34

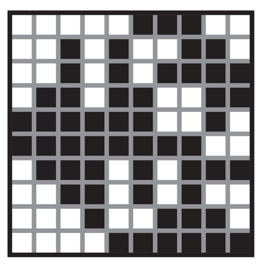

Rule 2

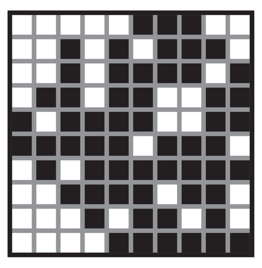

Rule 24

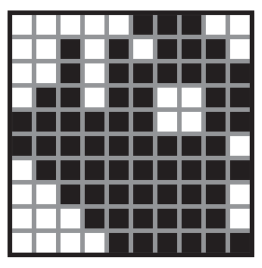

Rule 23

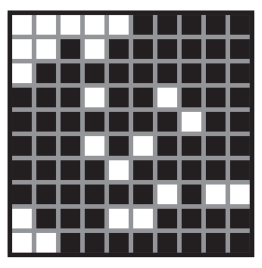

Rule 234

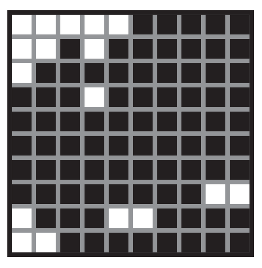

Rule 1

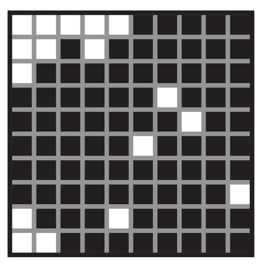

Rule 14

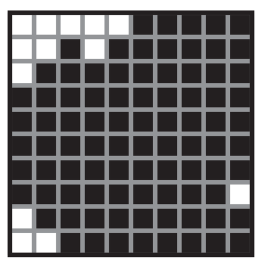

Rule 13


Rule 134

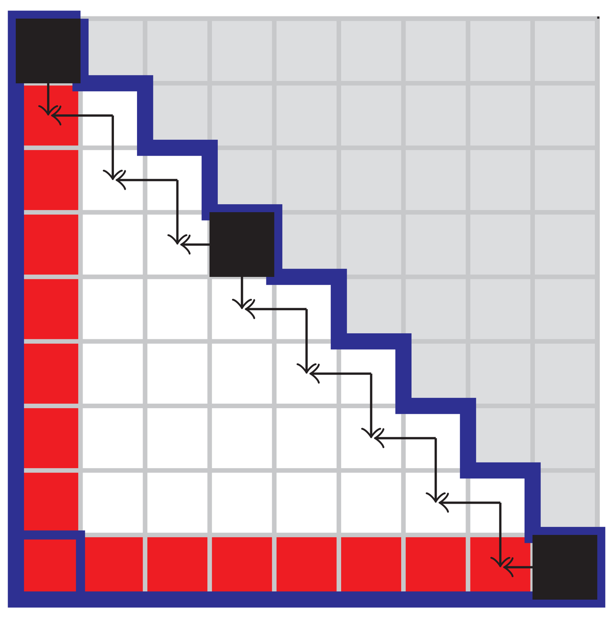

Rule 12


Rule 124

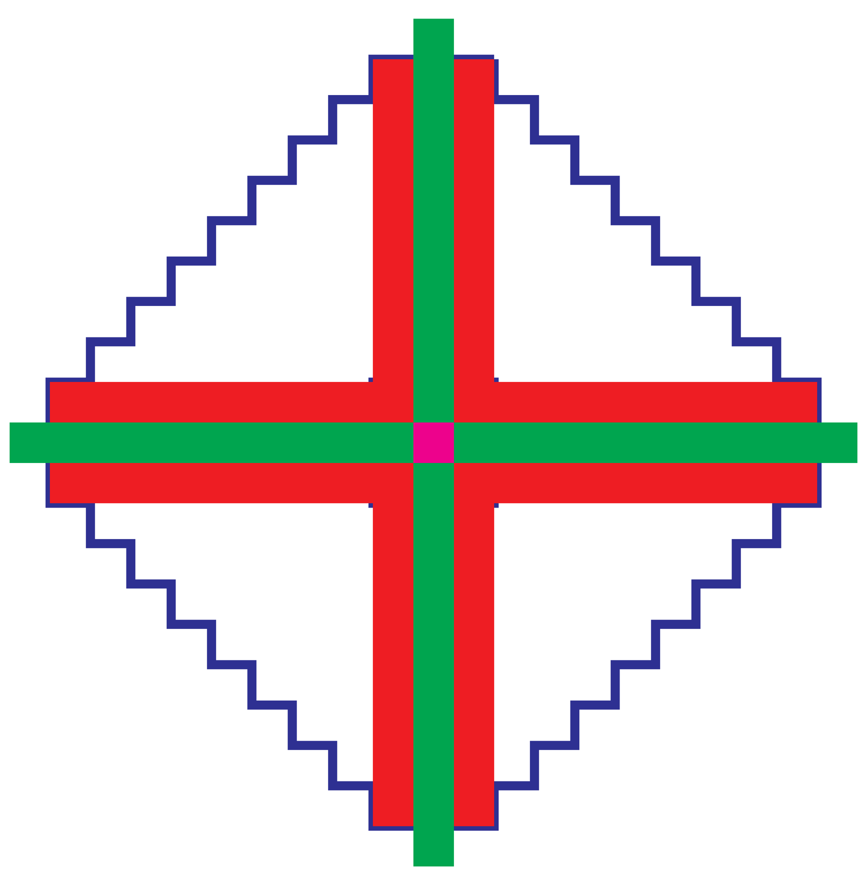

Rule 123


Rule 1234

Cases 12, 123, 124



The diagonals with 1s spread to the south west. We can compute this as a prefix sum ($\vee$) in **NC**.



Algorithm 1 STABLE for cases 12, 123, 124.

Find the smallest **blue** square containing cells in state 1 in its boundary.
Compute **red** cells
Compute **green** cells
Output: Value in the **center** cell.

Complexity

F	$STABLE$
4	$\mathcal{O}(1)$
3	in NC
34	in NC
2	P-Complete
24	P-Complete
23	?
234	in NC
1	?
14	?
13	?
134	?
12	in NC
124	in NC
123	in NC
1234	in NC

• A **stable cell** (SC) is a cell that never changes.

• Decision problem **STABLE**:

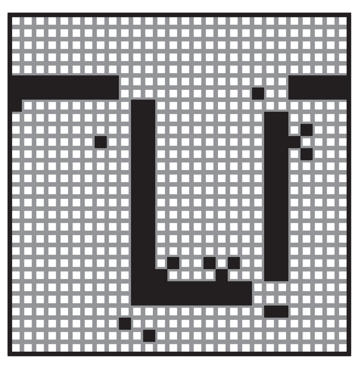
Inputs: TFCA F and $c \in \{0, 1\}^{\mathbb{Z}^2}$ a finite configuration.

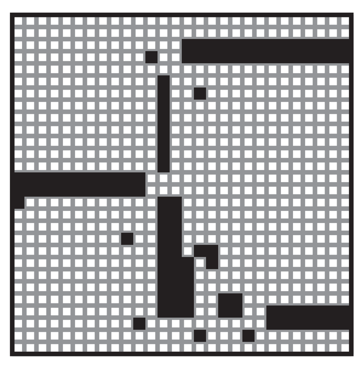
Question: Is $(0, 0)$ a SC?

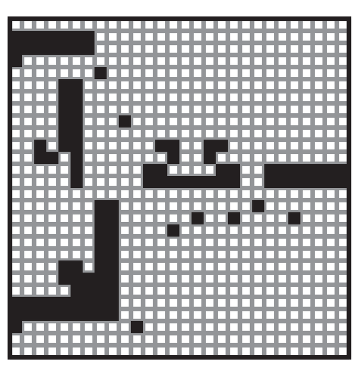
• **P**: Problems solvable in polynomial time in a sequential machine.

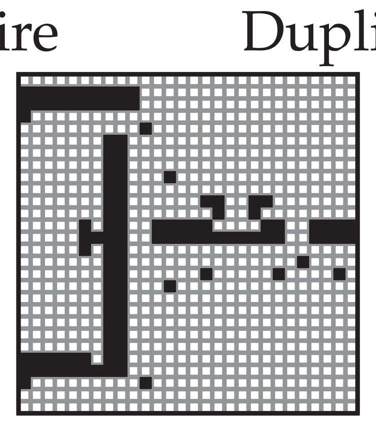
• **NC**: Problems solvable in poly-log time in a parallel machine.

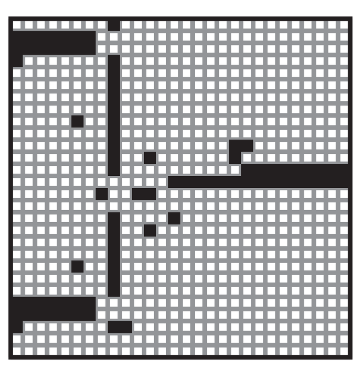
Cases 2, 24


Wire


Duplicator

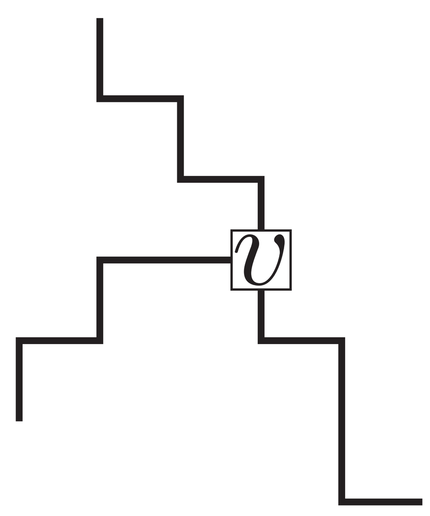

AND gate


OR gate

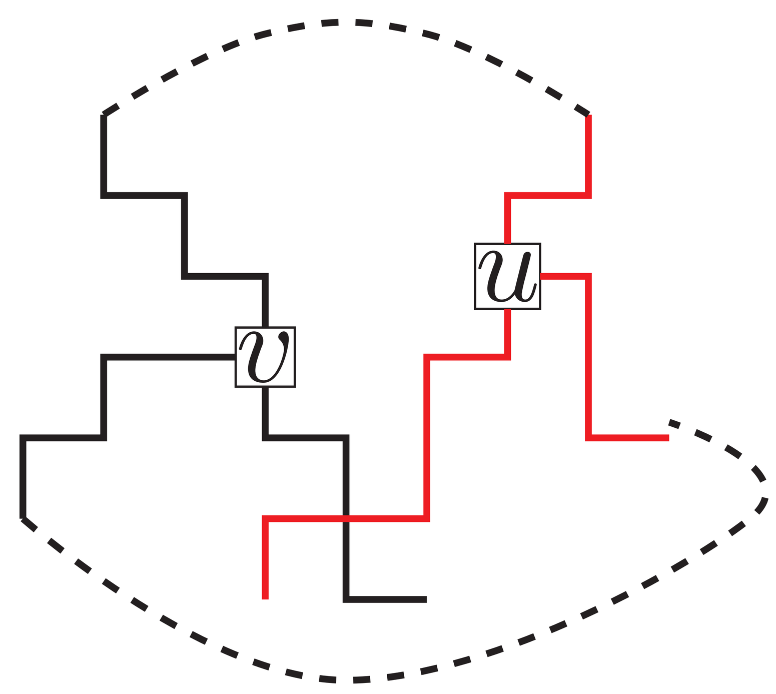

XOR gate

The problem of to know the output in a monotone circuit is **P**-complete (MCVP). Using this logic gates is possible to reduce MCVP to STABLE.

Case 234



A stable cell (SC) has 3 stable neighbors in state 0, then has 3 paths to the border.



A pair of SC have 3 paths joining them, then they are a triply connected component (TCC) (computable in **NC**).

Algorithm 2 STABLE for case 234.

Find tri-connected components in the graph induced by cells initially in state 0.
if $(0, 0)$ is in a TCÂžC component **then**
  return no change
else
  return change
end if