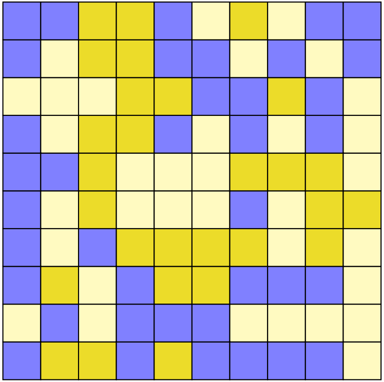
Hierarchy and expansiveness

Pierre Guillon & Charalampos Zinoviadis

CNRS & Institut de Mathématiques de Luminy
Turun yliopisto

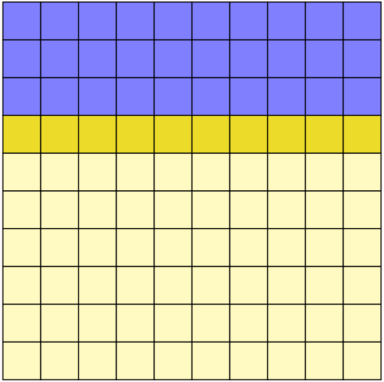
SFT

- **alphabet** : finite set A
  $(= \{\square, \square, \square\})$
- **configuration** : $x \in A^{\mathbb{Z}^d}$
- pattern : $x_{|F}$, $F \subset_{\text{finite}} A^{\mathbb{Z}^d}$
- **SFT** with forbidden patterns
  $\mathcal{F} \subset \{F \subset_{\text{finite}} A^{\mathbb{Z}^d}\}$:
  $X = \left\{ x \in A^{\mathbb{Z}^d} \middle| \forall i, F, x_{|F+i} \notin \mathcal{F} \right\}$



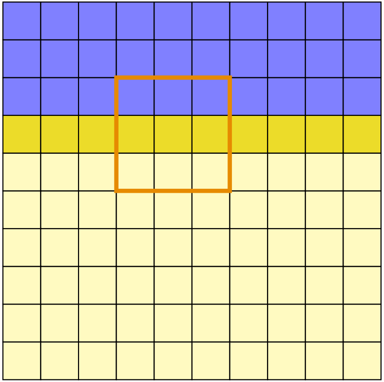
SFT

- **alphabet** : finite set A
  $(= \{\square, \square, \square\})$
- **configuration** : $x \in A^{\mathbb{Z}^d}$
- pattern : $x_{|F}$, $F \subset_{\text{finite}} A^{\mathbb{Z}^d}$
- **SFT** with forbidden patterns
  $\mathcal{F} \subset \{F \subset_{\text{finite}} A^{\mathbb{Z}^d}\}$:
  $X = \left\{ x \in A^{\mathbb{Z}^d} \middle| \forall i, F, x_{|F+i} \notin \mathcal{F} \right\}$



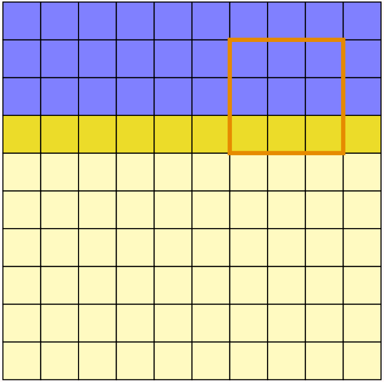
SFT

- **alphabet** : finite set A $(=\{\square,\square,\square\})$
- **configuration** : $x \in A^{\mathbb{Z}^d}$
- pattern : $x_{|F}$, $F \subset_{\text{finite}} A^{\mathbb{Z}^d}$
- **SFT** with forbidden patterns $\mathcal{F} \subset \{F \subset_{\text{finite}} A^{\mathbb{Z}^d}\}$:
  $X = \left\{ x \in A^{\mathbb{Z}^d} \middle| \forall i, F, x_{|F+i} \notin \mathcal{F} \right\}$



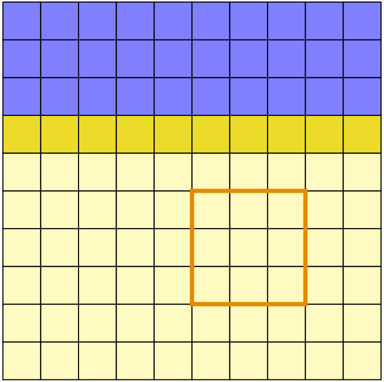
SFT

- **alphabet** : finite set A
  $(= \{\square, \square, \square\})$
- **configuration** : $x \in A^{\mathbb{Z}^d}$
- pattern : $x_{|F}$, $F \subset_{\text{finite}} A^{\mathbb{Z}^d}$
- **SFT** with forbidden patterns
  $\mathcal{F} \subset \{F \subset_{\text{finite}} A^{\mathbb{Z}^d}\}$:
  $X = \left\{ x \in A^{\mathbb{Z}^d} \middle| \forall i, F, x_{|F+i} \notin \mathcal{F} \right\}$



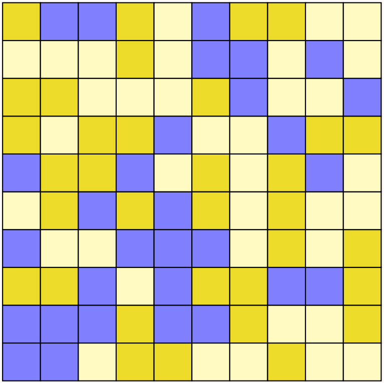
SFT

- **alphabet** : finite set A
  $(= \{\square, \square, \square\})$
- **configuration** : $x \in A^{\mathbb{Z}^d}$
- pattern : $x_{|F}$, $F \subset_{\text{finite}} A^{\mathbb{Z}^d}$
- **SFT** with forbidden patterns
  $\mathcal{F} \subset \{F \subset_{\text{finite}} A^{\mathbb{Z}^d}\}$:
  $X = \left\{ x \in A^{\mathbb{Z}^d} \middle| \forall i, F, x_{|F+i} \notin \mathcal{F} \right\}$



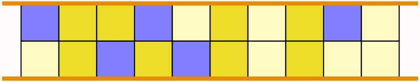
Expansiveness

- X is horizontally **expansive** if $\exists q, \forall x, y \in X$, $x_{|\mathbb{Z}\times[\![0,q[\![} = y_{|\mathbb{Z}\times[\![0,q[\![} \Rightarrow x = y$



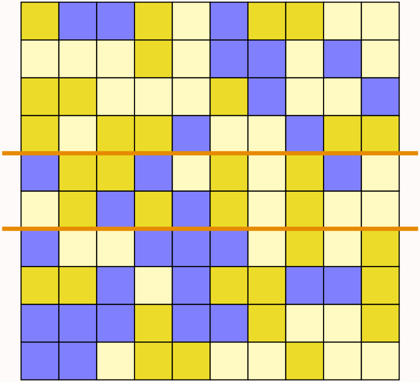
Expansiveness

- X is horizontally **expansive** if $\exists q, \forall x, y \in X,$ $x_{|\mathbb{Z}\times[\![0,q[\![} = y_{|\mathbb{Z}\times[\![0,q[\![} \Rightarrow x = y$



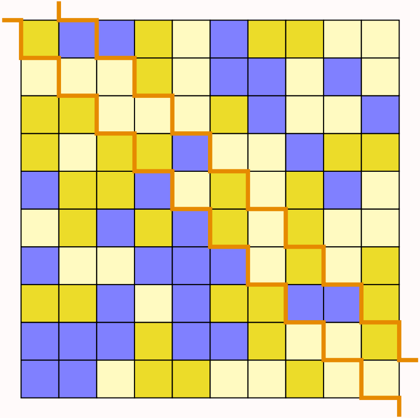
Expansiveness

- X is horizontally **expansive** if $\exists q, \forall x, y \in X$, $x_{|\mathbb{Z}\times[\![0,q[\![} = y_{|\mathbb{Z}\times[\![0,q[\![} \Rightarrow x = y$



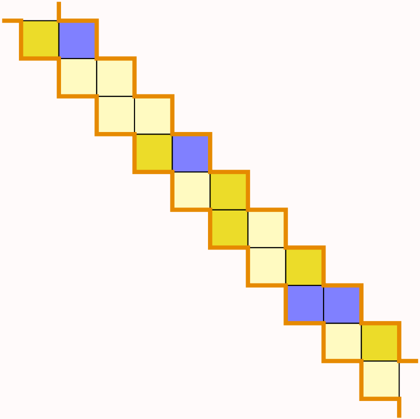
Expansiveness

- X is **expansive** in direction $\overrightarrow{d}$ if $\exists F, \forall x, y \in X,$
  $x_{\big|F+\overrightarrow{d}\mathbb{Z}} = y_{\big|F+\overrightarrow{d}\mathbb{Z}} \Rightarrow x = y$



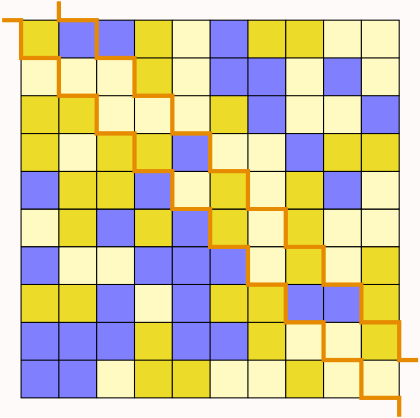
Expansiveness

- X is **expansive** in direction $\overrightarrow{d}$ if $\exists F, \forall x, y \in X$, $x_{|F+\overrightarrow{d}\mathbb{Z}} = y_{|F+\overrightarrow{d}\mathbb{Z}} \Rightarrow x = y$



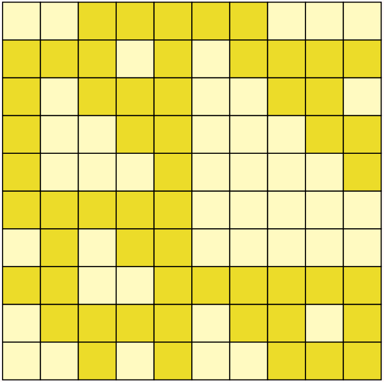
Expansiveness

- X is **expansive** in direction $\overrightarrow{d}$ if $\exists F, \forall x, y \in X,$ $x_{|F+\overrightarrow{d}\mathbb{Z}} = y_{|F+\overrightarrow{d}\mathbb{Z}} \Rightarrow x = y$



Expansiveness

- X is **expansive** in direction $\overrightarrow{d}$ if $\exists F, \forall x, y \in X$, $x_{|F+\overrightarrow{d}\mathbb{Z}} = y_{|F+\overrightarrow{d}\mathbb{Z}} \Rightarrow x = y$
- **partial cellular automaton** (PCA) (with radius 1) : partial map $f : A^3 \to A$.



Expansiveness

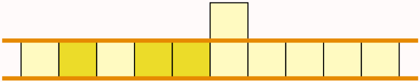
- X is **expansive** in direction $\overrightarrow{d}$ if
  $\exists F, \forall x, y \in X,$
  $x_{\big|F+\overrightarrow{d}\mathbb{Z}} = y_{\big|F+\overrightarrow{d}\mathbb{Z}} \Rightarrow x = y$
- **partial cellular automaton** (PCA)
  (with radius 1) :
  partial map $f : A^3 \rightarrow A$.
  horizontally expansive SFT $\leftrightarrow$
  reversible PCA

Expansiveness

- X is **expansive** in direction $\overrightarrow{d}$ if $\exists F, \forall x, y \in X,$ $x_{|F+\overrightarrow{d}\mathbb{Z}} = y_{|F+\overrightarrow{d}\mathbb{Z}} \Rightarrow x = y$
- **partial cellular automaton** (PCA) (with radius 1) :
  partial map $f : A^3 \to A$.
  horizontally expansive SFT $\leftrightarrow$ reversible PCA

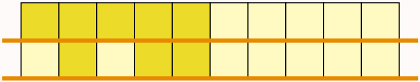
Expansiveness

- X is **expansive** in direction $\overrightarrow{d}$ if $\exists F, \forall x, y \in X$, $x_{\big|F+\overrightarrow{d}\mathbb{Z}} = y_{\big|F+\overrightarrow{d}\mathbb{Z}} \Rightarrow x = y$
- **partial cellular automaton** (PCA) (with radius 1) : partial map $f : A^3 \to A$. *horizontally expansive SFT* $\leftrightarrow$ *reversible PCA*



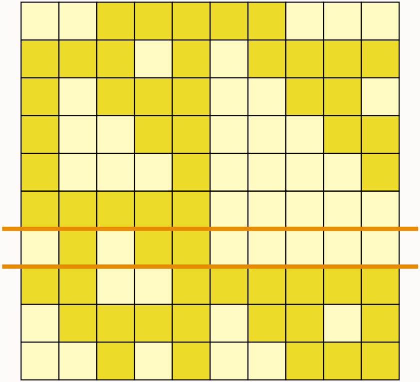
Expansiveness

- X is **expansive** in direction $\overrightarrow{d}$ if $\exists F, \forall x, y \in X,$
  $x_{|F+\overrightarrow{d}\mathbb{Z}} = y_{|F+\overrightarrow{d}\mathbb{Z}} \Rightarrow x = y$
- **partial cellular automaton** (PCA) (with radius 1) :
  partial map $f : A^3 \to A$.
  horizontally expansive SFT ↔ reversible PCA



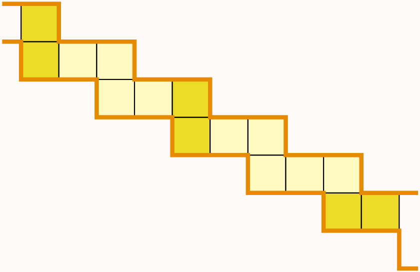
Expansiveness

- X is **expansive** in direction $\overrightarrow{d}$ if $\exists F, \forall x, y \in X,$ $x_{\big|F+\overrightarrow{d}\mathbb{Z}} = y_{\big|F+\overrightarrow{d}\mathbb{Z}} \Rightarrow x = y$
- **partial cellular automaton** (PCA) (with radius 1) :
  partial map $f : A^3 \to A$.
  horizontally expansive SFT $\leftrightarrow$ *reversible PCA*



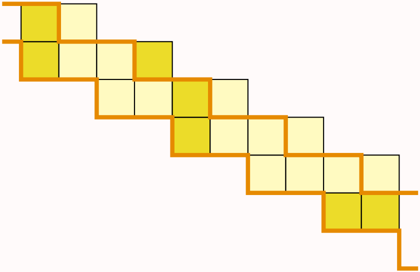
Expansiveness

- X is **expansive** in direction $\overrightarrow{d}$ if $\exists F, \forall x, y \in X$, $x_{\big|F+\overrightarrow{d}\mathbb{Z}} = y_{\big|F+\overrightarrow{d}\mathbb{Z}} \Rightarrow x = y$
- **partial cellular automaton** (PCA) (with radius 1) : partial map $f : A^3 \to A$. *horizontally expansive SFT* $\leftrightarrow$ *reversible PCA*



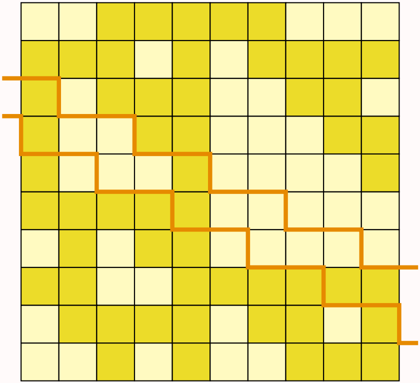
Expansiveness

- X is **expansive** in direction $\overrightarrow{d}$ if $\exists F, \forall x, y \in X$, $x_{\big|F+\overrightarrow{d}\mathbb{Z}} = y_{\big|F+\overrightarrow{d}\mathbb{Z}} \Rightarrow x = y$
- **partial cellular automaton** (PCA) (with radius 1) : partial map $f : A^3 \to A$. *horizontally expansive SFT* $\leftrightarrow$ *reversible PCA*



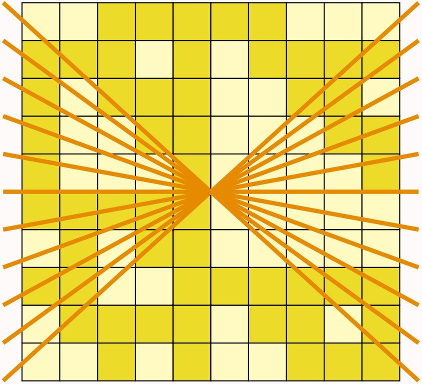
Expansiveness

- X is **expansive** in direction $\overrightarrow{d}$ if $\exists F, \forall x, y \in X$, $x_{|F+\overrightarrow{d}\mathbb{Z}} = y_{|F+\overrightarrow{d}\mathbb{Z}} \Rightarrow x = y$
- **partial cellular automaton** (PCA) (with radius 1) : partial map $f : A^3 \to A$.
  horizontally expansive SFT $\leftrightarrow$ *reversible PCA*
  $\Rightarrow$ *expansive in every slope* $\in]-1, 1[$.



Expansiveness

- X is **expansive** in direction $\overrightarrow{d}$ if $\exists F, \forall x, y \in X$, $x_{\left|F+\overrightarrow{d}\mathbb{Z}\right.} = y_{\left|F+\overrightarrow{d}\mathbb{Z}\right.} \Rightarrow x = y$
- **partial cellular automaton** (PCA) (with radius 1) : partial map $f : A^3 \to A$. *horizontally expansive SFT $\leftrightarrow$ reversible PCA* *$\Rightarrow$ expansive in every slope $\in]-1, 1[$.*
- *set of expansive directions: effectively open*



Expansiveness and computation

	universality	(directional) entropy	subactions	
	Berger 1966	Meyerovitch-Hochman 2007	Aubrun-Sablik 2010	
	Kari 1992	G-Z 2012		
	Lukkarila-Kari 2008			
	Lukkarila-Kari 2009			

Expansiveness and computation

	universality	(directional) entropy	subactions	
	Berger 1966	Meyerovitch-Hochman 2007	Aubrun-Sablik 2010	
	Kari 1992	G-Z 2012		
	Lukkarila-Kari 2008			
	Lukkarila-Kari 2009			
	$\perp$	0	periodic	

Expansiveness and computation

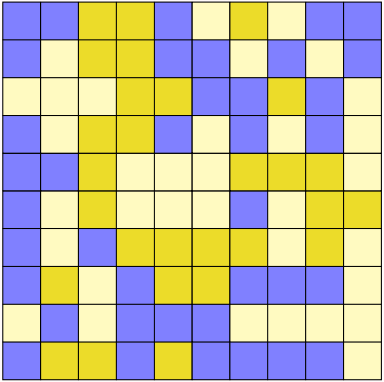
	universality	(directional) entropy	subactions	self-simula
	Berger 1966	Meyerovitch-Hochman 2007	Aubrun-Sablik 2010	
	Kari 1992	G-Z 2012		
	Lukkarila-Kari 2008			
	Lukkarila-Kari 2009			
	✓	✓	✓	
	$\perp$	0	periodic	

Expansiveness and computation

	universality	**(directional) entropy**	**subactions**	**self-simula**
	Berger 1966	Meyerovitch-Hochman 2007	Aubrun-Sablik 2010	Romashche Durand-Sher
	Kari 1992	G-Z 2012		Gács 2001
	Lukkarila-Kari 2008			
	Lukkarila-Kari 2009			
	✓	✓	✓	✓
	$\perp$	0	periodic	trivial

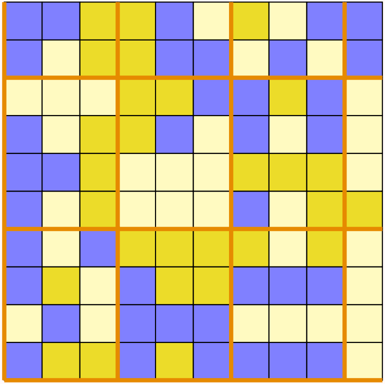
Simulation

- $\phi : \{\square, \square, \square\}^{p \times q} \to \{\square, \square, \square, \square\}$
  x **(ϕ, p, q, j)-simulates** y:
  $y \preceq_{\phi,p,q} x$ if



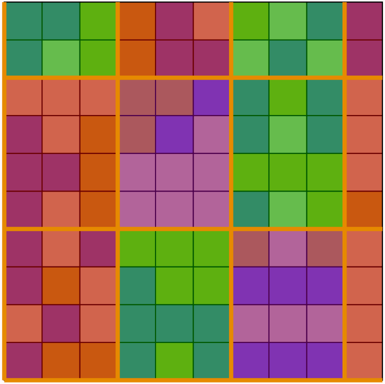
Simulation

- $\phi : \{\square, \square, \square\}^{p \times q} \to \{\square, \square, \square, \square\}$
  x (ϕ, p, q, j)**-simulates** y:
  $y \preceq_{\phi,p,q} x$ if



Simulation

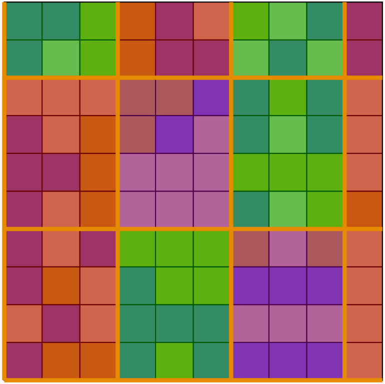
- $\phi : \{\square, \square, \square\}^{p\times q} \to \{\square, \square, \square, \square\}$
  x **(ϕ, p, q, j)-simulates** y:
  $y \preceq_{\phi,p,q} x$ if
  $\forall i, \phi(x_{|(p,q)i+j+[\![0,p[\![\times[\![0,q[\![}) = y_i$



Simulation

- $\phi : \{\square, \square, \square\}^{p\times q} \to \{\square, \square, \square, \square\}$
  x **(ϕ, p, q, j)-simulates** y:
  $y \preceq_{\phi,p,q} x$ if
  $\forall i, \phi(x_{|(p,q)i+j+[\![0,p[\![\times[\![0,q[\![}) = y_i$
- X **(ϕ, p, q)-simulates** Y:
  $Y \preceq_{\phi,p,q} X$ if

$$\begin{cases} \forall x \in X, \exists! y \in Y, \exists! j, y \preceq_{\phi,p,q} x \\ \forall y \in Y, \exists! x \in X, y \preceq_{\phi,p,q,0} x \end{cases}$$

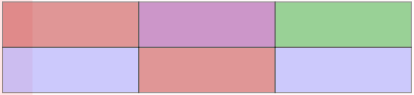


Simulation of a partial CA

$\forall X, \exists Y \succeq X,$

Y over alphabet :

- $\{\square, \blacksquare\}$ (simulated state)
- address (from 0 to $S-1$)

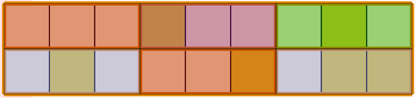


Simulation of a partial CA

$\forall X, \exists Y \succeq X,$

Y over alphabet :

- $\{\square, \blacksquare\}$ (simulated state)
- address (from 0 to $S-1$)

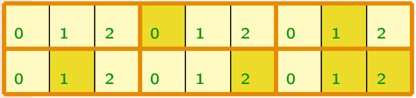


Simulation of a partial CA

$\forall X, \exists Y \succeq X,$

Y over alphabet :

- $\{$□, ■$\}$ (simulated state)
- address (from 0 to $S-1$)



Simulation of a partial CA

$\forall X, \exists Y \succeq X,$

Y over alphabet :

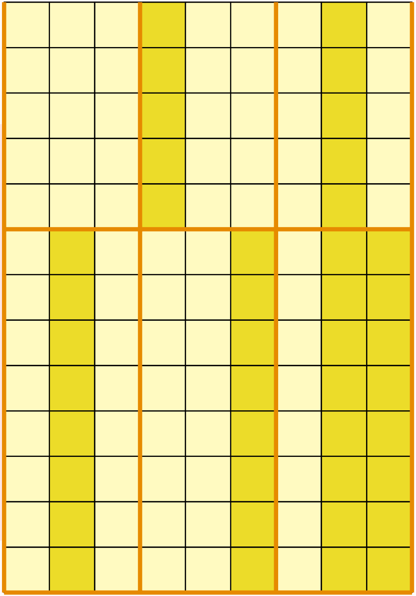
- $\{\square, \blacksquare\}$ (simulated state)
- address (from 0 to $S-1$)
- clock (from 0 to $T-1$)

4 0	4 1	4 2	4 0	4 1	4 2	4 0	4 1	4 2
3 0	3 1	3 2	3 0	3 1	3 2	3 0	3 1	3 2
2 0	2 1	2 2	2 0	2 1	2 2	2 0	2 1	2 2
1 0	1 1	1 2	1 0	1 1	1 2	1 0	1 1	1 2
0 0	0 1	0 2	0 0	0 1	0 2	0 0	0 1	0 2
7 0	7 1	7 2	7 0	7 1	7 2	7 0	7 1	7 2
6 0	6 1	6 2	6 0	6 1	6 2	6 0	6 1	6 2
5 0	5 1	5 2	5 0	5 1	5 2	5 0	5 1	5 2
4 0	4 1	4 2	4 0	4 1	4 2	4 0	4 1	4 2
3 0	3 1	3 2	3 0	3 1	3 2	3 0	3 1	3 2
2 0	2 1	2 2	2 0	2 1	2 2	2 0	2 1	2 2
1 0	1 1	1 2	1 0	1 1	1 2	1 0	1 1	1 2
0 0	0 1	0 2	0 0	0 1	0 2	0 0	0 1	0 2

Simulation of a partial CA

$\forall X, \exists Y \succeq X,$
Y with radius 1 over alphabet :

- $\{\square, \blacksquare\}$ (simulated state)
- address (from 0 to $S-1$)
- clock (from 0 to $T-1$)

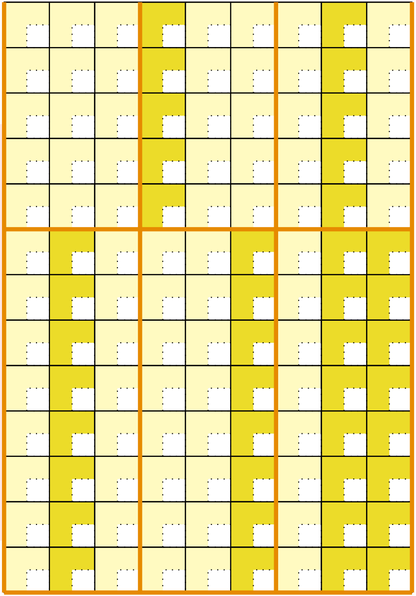


Simulation of a partial CA

$\forall X, \exists Y \succeq X$,

Y with radius 1 over alphabet :

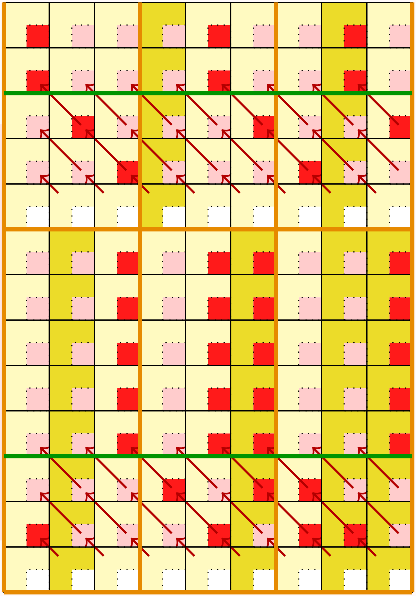
- $\{\square, \blacksquare\}$ (simulated state)
- address (from 0 to $S-1$)
- clock (from 0 to $T-1$)
- $\{\square, \blacksquare\}$ (state of the neighbor)



Simulation of a partial CA

$\forall X, \exists Y \succeq X,$
Y with radius 1 over alphabet :

- $\{\square, \blacksquare\}$ (simulated state)
- address (from 0 to $S-1$)
- clock (from 0 to $T-1$)
- $\{\square, \blacksquare\}$ (state of the neighbor)

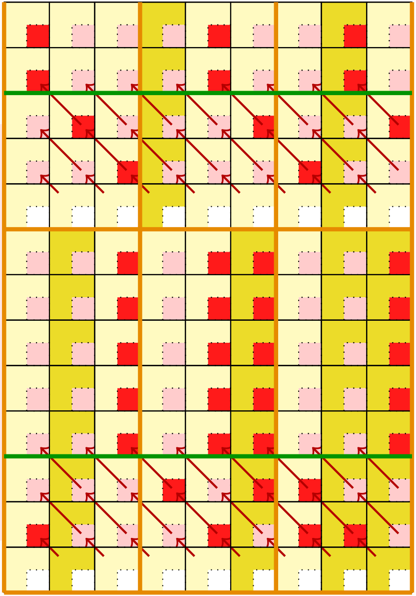


Simulation of a partial CA

$\forall X, \exists Y \succeq X,$

Y with radius 1 over alphabet :

- $\{\square, \blacksquare\}$ (simulated state)
- address (from 0 to $S-1$)
- clock (from 0 to $T-1$)
- $\{\square, \blacksquare\}^2$ (state of the neighbors)

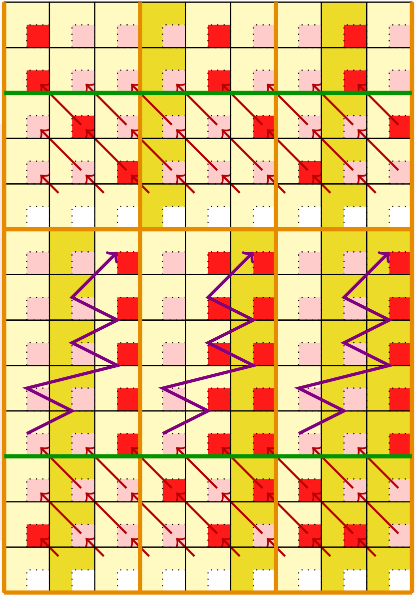


Simulation of a partial CA

$\forall X, \exists Y \succeq X,$

Y with radius 1 over alphabet :

- $\{\square, \blacksquare\}$ (simulated state)
- address (from 0 to $S-1$)
- clock (from 0 to $T-1$)
- $\{\square, \blacksquare\}^2$ (state of the neighbors)
- $Q \sqcup \{\emptyset\}$ (state of the universal Turing machine)

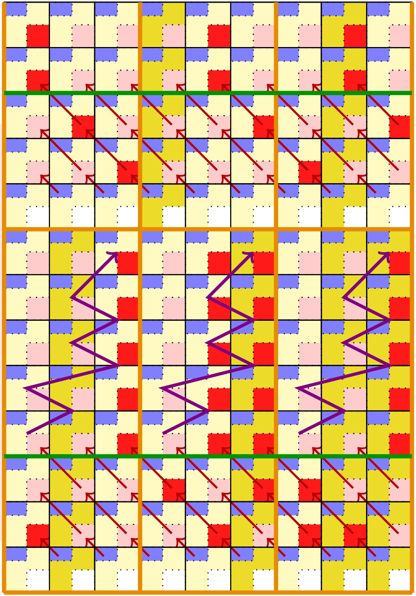


Simulation of a partial CA

$\forall X, \exists Y \succeq X,$

Y with radius 1 over alphabet :

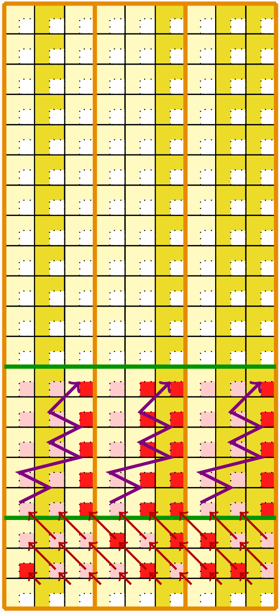
- $\{\square, \blacksquare\}$ (simulated state)
- address (from 0 to $S-1$)
- clock (from 0 to $T-1$)
- $\{\square, \blacksquare\}^2$ (state of the neighbors)
- $Q \sqcup \{\emptyset\}$ (state of the universal Turing machine)
- P (program)
- $\ldots$ (worktapes)



Simulation of a reversible partial CA

$\forall X$ reversible, $\exists Y \succeq X$ reversible,
Y with radius 1 over alphabet :

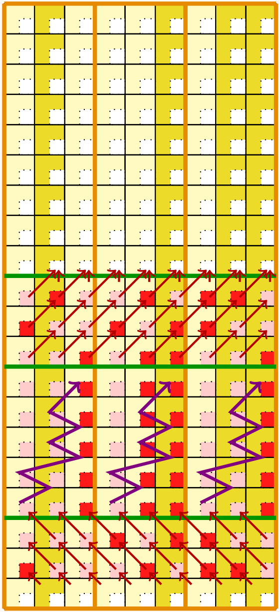
- $\{\square, \blacksquare\}$ (simulated state)
- address , clock
- $\{\square, \blacksquare\}^2$ (state of the neighbors)
- $Q \sqcup \{\emptyset\}$ (state of the universal reversible oblivious Turing machine)
- P (program for X (+inverse))
- $\ldots$ (worktapes)



Simulation of a reversible partial CA

$\forall X$ reversible, $\exists Y \succeq X$ reversible,
Y with radius 1 over alphabet :

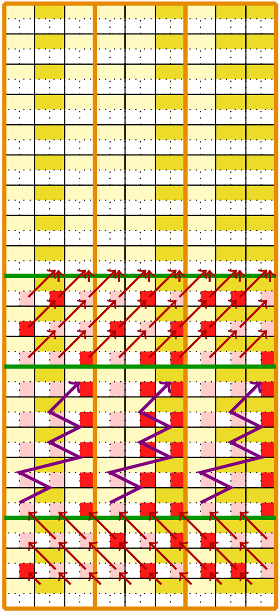
- $\{\square, \blacksquare\}$ (simulated state)
- address , clock
- $\{\square, \blacksquare\}^2$ (state of the neighbors)
- $Q \sqcup \{\emptyset\}$ (state of the universal reversible oblivious Turing machine)
- P (program for X (+inverse))
- $\ldots$ (worktapes)



Simulation of a reversible partial CA

$\forall X$ reversible, $\exists Y \succeq X$ reversible,
Y with radius 1 over alphabet :

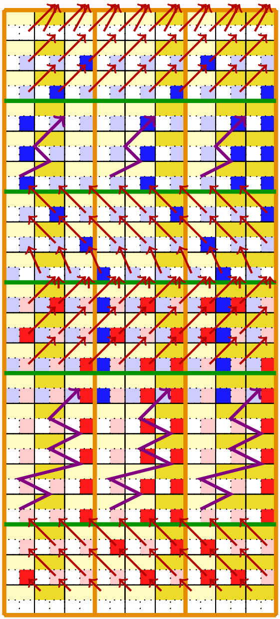
- $\{\square, \blacksquare\}$ (simulated state)
- address , clock
- $\{\square, \blacksquare\}^2$ (state of the neighbors)
- $Q \sqcup \{\emptyset\}$ (state of the universal reversible oblivious Turing machine)
- P (program for X (+inverse))
- $\ldots$ (worktapes)
- $\{\square, \blacksquare\}$ (future simulated state)



Simulation of a reversible partial CA

$\forall X$ reversible, $\exists Y \succeq X$ reversible,
Y with radius 1 over alphabet :

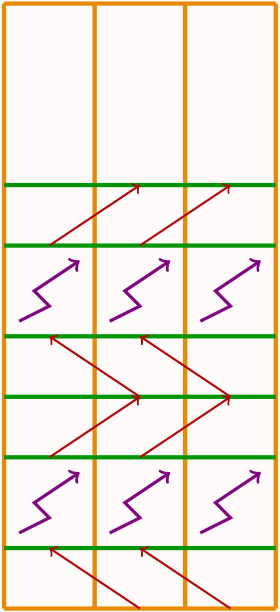
- $\{\square, \blacksquare\}$ (simulated state)
- address , clock
- $\{\square, \blacksquare\}^2$ (state of the neighbors)
- $Q \sqcup \{\emptyset\}$ (state of the universal reversible oblivious Turing machine)
- P (program for X (+inverse))
- ... (worktapes)
- $\{\square, \blacksquare\}$ (future simulated state)



Self-simulation

For all S, T, we can construct $Y_{S,T}$ that simulates all X that are relatively "small" and "easy". We can do more:

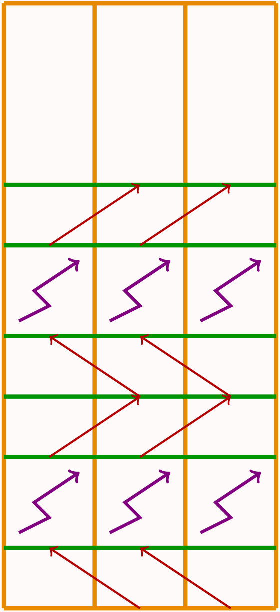
- $A_{S,T,n} = A$ has a fixed number of **fields**.
- $A.Prog$ contains the **program** of the simulated rule.
  $A.Level$ contains the **level**.



Self-simulation

For all S, T, we can construct $Y_{S,T}$ that simulates all X that are relatively "small" and "easy". We can do more:

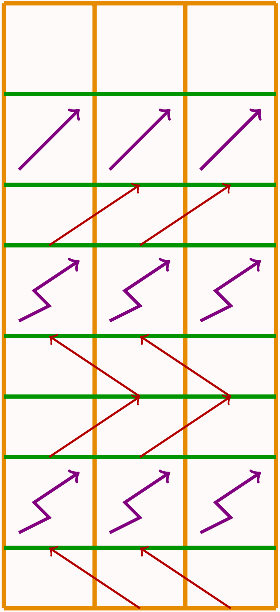
- $A_{S,T,n} = A$ has a fixed number of **fields**.
- *A.Prog* contains the **program** of the simulated rule.
  A.Level contains the **level**.



Self-simulation

For all S, T, we can construct $Y_{S,T}$ that simulates all X that are relatively "small" and "easy". We can do more:

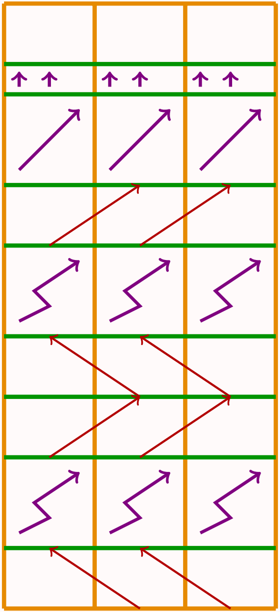
- $A_{S,T,n} = A$ has a fixed number of **fields**.
- *A.Prog* contains the **program** of the simulated rule.
  A.Level contains the **level**.
- Check that B has the fields of A.



Self-simulation

For all S, T, we can construct $Y_{S,T}$ that simulates all X that are relatively "small" and "easy". We can do more:

- $A_{S,T,n} = A$ has a fixed number of **fields**.
- $A.Prog$ contains the **program** of the simulated rule.
  $A.Level$ contains the **level**.
- Check that B has the fields of A.
- $B.Prog = A.Prog$ (bit by bit).



Self-simulation

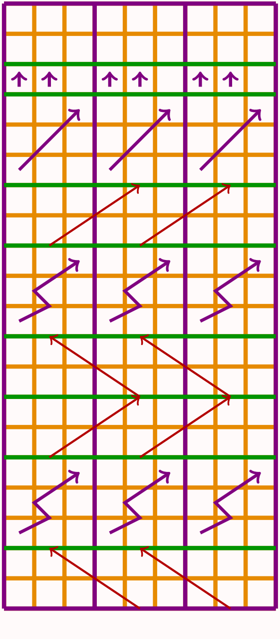
For all S, T, we can construct $Y_{S,T}$ that simulates all X that are relatively "small" and "easy". We can do more:

- $A_{S,T,n} = A$ has a fixed number of **fields**.
- $A.Prog$ contains the **program** of the simulated rule.
  $A.Level$ contains the **level**.
- Check that B has the fields of A.
- $B.Prog = A.Prog$ (bit by bit).

Self-simulation

For all S, T, we can construct $Y_{S,T}$ that simulates all X that are relatively "small" and "easy". We can do more:

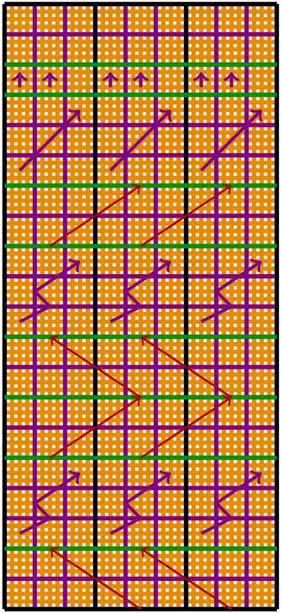
- $A_{S,T,n} = A$ has a fixed number of **fields**.
- $A.Prog$ contains the **program** of the simulated rule.
  $A.Level$ contains the **level**.
- Check that B has the fields of A.
- $B.Prog = A.Prog$ (bit by bit).



Self-simulation

For all S, T, we can construct $Y_{S,T}$ that simulates all X that are relatively "small" and "easy". We can do more:

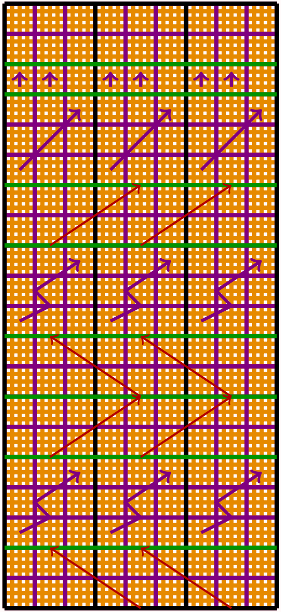
- $A_{S,T,n} = A$ has a fixed number of **fields**.
- $A.Prog$ contains the **program** of the simulated rule.
  $A.Level$ contains the **level**.
- Check that B has the fields of A.
- $B.Prog = A.Prog$ (bit by bit).



Self-simulation

For all S, T, we can construct $Y_{S,T}$ that simulates all X that are relatively "small" and "easy". We can do more:

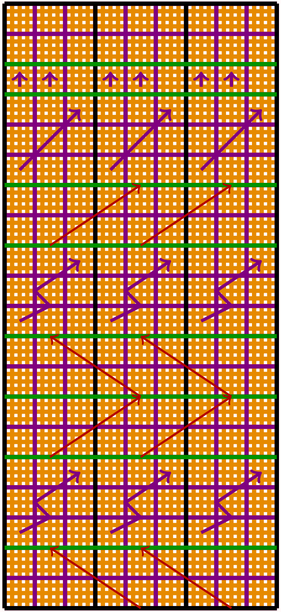
- $A_{S,T,n} = A$ has a fixed number of **fields**.
- $A.Prog$ contains the **program**. $A.Level$ contains the **level**.
- Check that B has the fields of A.
- $B.Prog = A.Prog$ (bit by bit).
- $B.Level = A.Level + 1$ (bit by bit).
- $B.Address$ and $B.Clock$ have maximal values $S' = S_{n+1}$ and $T' = T_{n+1}$.



Self-simulation

For all S, T, we can construct $Y_{S,T}$ that simulates all X that are relatively "small" and "easy". We can do more:

- $A_{S,T,n} = A$ has a fixed number of **fields**.
- *A.Prog* contains the **program**. *A.Level* contains the **level**.
- Check that B has the fields of A.
- $B.Prog = A.Prog$ (bit by bit).
- $B.Level = A.Level + 1$ (bit by bit).
- *B.Address* and *B.Clock* have maximal values $S' = S_{n+1}$ and $T' = T_{n+1}$.
- simulate another machine $\mathcal{M}_n$ for n steps.



Self-simulation

There exists sequences $(S_n, T_n)_{n\in\mathbb{N}}$ such that $Y_{S_{n+1},T_{n+1},n+1}$ is "small" and "easy" relative to $Y_{S_n,T_n,n}$, for all n.

Self-simulation

There exists sequences $(S_n, T_n)_{n\in\mathbb{N}}$ such that $Y_{S_{n+1},T_{n+1},n+1}$ is "small" and "easy" relative to $Y_{S_n,T_n,n}$, for all n.
Let $\mathcal{M}$ be a TM.

Then we can build (X_n) RPCA,
- empty if $\mathcal{M}$ halts;
- otherwise $\forall n, X_n$ simulates X_{n+1}.

Cor:

Emptiness of RPCA is *undecidable*.

Expansiveness and self-simulation

Let $\mathcal{M}$ be a TM.

Then we can build (X_n) RPCA,
- empty if $\mathcal{M}$ halts ;
- otherwise $\forall n, X_n$ (ϕ_n, p_n, q_n)-simulates X_{n+1}.

Expansiveness and self-simulation

Let $\mathcal{M}$ be a TM.

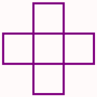
Then we can build (X_n) RPCA,
- empty if $\mathcal{M}$ halts ;
- otherwise $\forall n, X_n$ (ϕ_n, p_n, q_n)-simulates X_{n+1}.

Expansiveness and self-simulation

Let $\mathcal{M}$ be a TM.

Then we can build (X_n) RPCA,
- empty if $\mathcal{M}$ halts ;
- otherwise $\forall n, X_n$
(ϕ_n, p_n, q_n)-simulates X_{n+1}.

Then X_n is expansive in the directions of X_{n+1} expanded by (p_n, q_n).

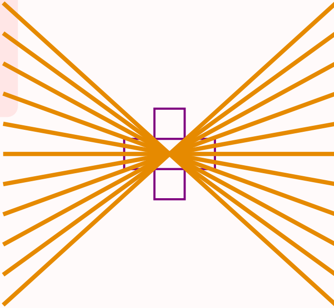


Expansiveness and self-simulation

Let $\mathcal{M}$ be a TM.

Then we can build (X_n) RPCA,
- empty if $\mathcal{M}$ halts ;
- otherwise $\forall n, X_n$ (ϕ_n, p_n, q_n)-simulates X_{n+1}.

Then X_n is expansive in the directions of X_{n+1} expanded by (p_n, q_n).
If $\forall n, p_n > 2q_n$,
then X_n is expansive in every direction but 1.



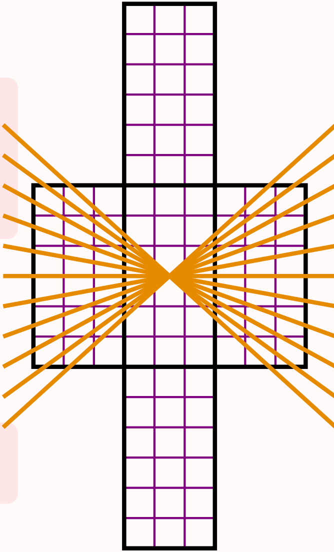
Expansiveness and self-simulation

Let $\mathcal{M}$ be a TM.

Then we can build (X_n) RPCA,
- empty if $\mathcal{M}$ halts ;
- otherwise $\forall n, X_n$
(ϕ_n, p_n, q_n)-simulates X_{n+1}.

Then X_n is expansive in the directions of X_{n+1} expanded by (p_n, q_n).
If $\forall n, p_n > 2q_n$,
then X_n is expansive in every direction but 1.
Cor:

Emptiness of RPCA expansive in all directions but 1 is *undecidable*.



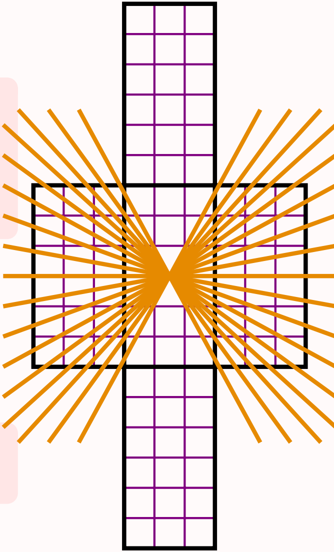
Expansiveness and self-simulation

Let $\mathcal{M}$ be a TM.

Then we can build (X_n) RPCA,
- empty if $\mathcal{M}$ halts ;
- otherwise $\forall n, X_n$ (ϕ_n, p_n, q_n)-simulates X_{n+1}.

Then X_n is expansive in the directions of X_{n+1} expanded by (p_n, q_n).
If $\forall n, p_n > 2q_n$,
then X_n is expansive in every direction but 1.
Cor:

Emptiness of RPCA expansive in all directions but 1 is *undecidable*.



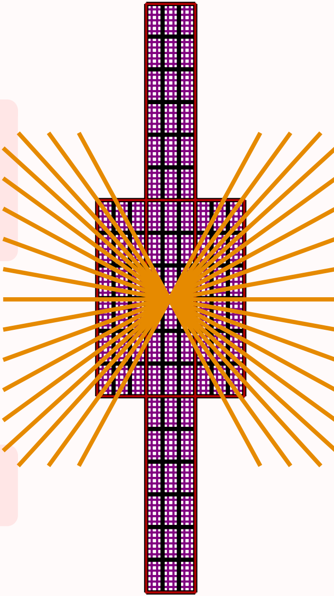
Expansiveness and self-simulation

Let $\mathcal{M}$ be a TM.

Then we can build (X_n) RPCA,
- empty if $\mathcal{M}$ halts ;
- otherwise $\forall n, X_n$
(ϕ_n, p_n, q_n)-simulates X_{n+1}.

Then X_n is expansive in the directions of X_{n+1} expanded by (p_n, q_n).
If $\forall n, p_n > 2q_n$,
then X_n is expansive in every direction but 1.
Cor:

Emptiness of RPCA expansive in all directions but 1 is *undecidable*.



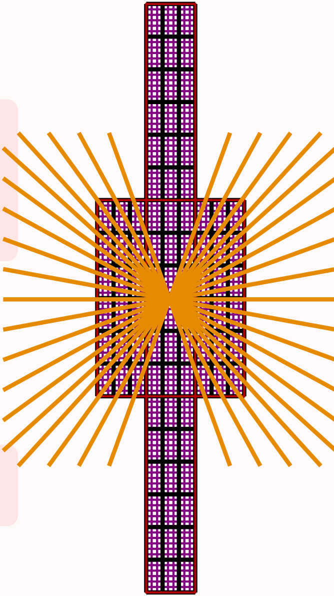
Expansiveness and self-simulation

Let $\mathcal{M}$ be a TM.

Then we can build (X_n) RPCA,
- empty if $\mathcal{M}$ halts ;
- otherwise $\forall n, X_n$
(ϕ_n, p_n, q_n)-simulates X_{n+1}.

Then X_n is expansive in the directions of X_{n+1} expanded by (p_n, q_n).
If $\forall n, p_n > 2q_n$,
then X_n is expansive in every direction but 1.
Cor:

Emptiness of RPCA expansive in all directions but 1 is *undecidable*.



Minimality and self-similarity

Let $\mathcal{M}$ be a TM.

Then we can build (X_n) RPCA,
- empty if $\mathcal{M}$ halts ;
- otherwise $\forall n, X_n$ (ϕ_n, p_n, q_n)-simulates X_{n+1}.

Minimality and self-similarity

Let $\mathcal{M}$ be a TM.

Then we can build (X_n) RPCA,
- empty if $\mathcal{M}$ halts ;
- otherwise $\forall n, X_n$ (ϕ_n, p_n, q_n)-simulates X_{n+1}.

- We can assume that the pattern is a line of cells with $clock = 0$
- By de-substituting a finite number of times, we can assume that the pattern is a letter of X_m with $clock = 0$
- These letters appear as frequently as $p_{m+1}p_m$.
- By susbstituting back, we get that the initial pattern also apperas periodically.

Then X_n is minimal (even Toeplitz).

Minimality and self-similarity

Let $\mathcal{M}$ be a TM.

Then we can build (X_n) RPCA,
- empty if $\mathcal{M}$ halts ;
- otherwise $\forall n, X_n$ (ϕ_n, p_n, q_n)-simulates X_{n+1}.

- We can assume that the pattern is a line of cells with $clock = 0$
- By de-substituting a finite number of times, we can assume that the pattern is a letter of X_m with $clock = 0$
- These letters appear as frequently as $p_{m+1}p_m$.
- By susbstituting back, we get that the initial pattern also apperas periodically.

Then X_n is minimal (even Toeplitz). Cor:

Emptiness of minimal RPCA expansive in all directions but 1 is *undecidable*.

Other applications

Characterization of realizability for RPCA:

- substitutions
- subactions
- entropies
- directions of expansiveness

Other applications

Characterization of realizability for RPCA:

- substitutions
- subactions
- entropies
- directions of expansiveness

Future work:

- entropy, subactions of reversible CA?
- undecidability of CA expansiveness?