Self-simulation, substitutions and soficity of multidimensional shift spaces

Antonin Callard (LIP, ENS de Lyon)

Joint work with Léo Paviet Salomon and Pascal Vanier

Séminaire ESCAPE, LIRMM

03 February 2026

Shift spaces

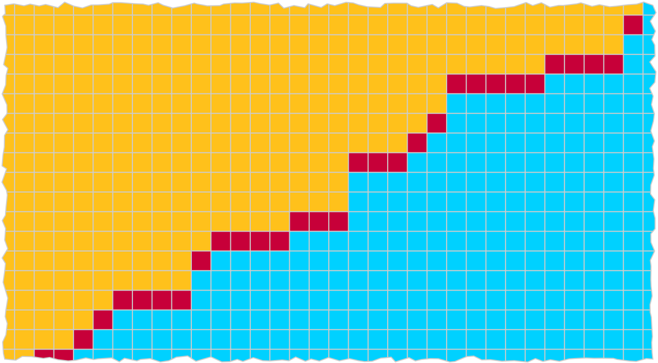
Definition

A *shift space* on $\mathbb{Z}^d$ is a set of colorings $\mathbb{Z}^d \to \mathcal{A}$ ("*configurations*") defined by forbidden patterns $\mathcal{F}$:

$$X_{\mathcal{F}} = \left\{ x \in \mathcal{A}^{\mathbb{Z}^d} : \forall w \in \mathcal{F},\ w \text{ does not appear in } x \right\}.$$

Example:

$\mathcal{F} = \Big\{$  $\Big\}$



Shift spaces

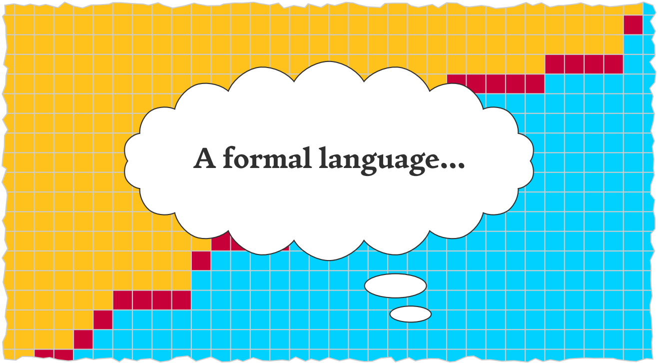
Definition

A *shift space* on $\mathbb{Z}^d$ is a set of colorings $\mathbb{Z}^d \to \mathcal{A}$ ("*configurations*") defined by forbidden patterns $\mathcal{F}$:

$$X_{\mathcal{F}} = \left\{ x \in \mathcal{A}^{\mathbb{Z}^d} : \forall w \in \mathcal{F},\ w \text{ does not appear in } x \right\}.$$

Example:

$\mathcal{F} = \{$  $\}$



Classifying shift spaces by computational expressive power

Definition

A *shift space* on $\mathbb{Z}^d$ is a set of colorings $\mathbb{Z}^d \to \mathcal{A}$ ("*configurations*") defined by forbidden patterns $\mathcal{F}$:

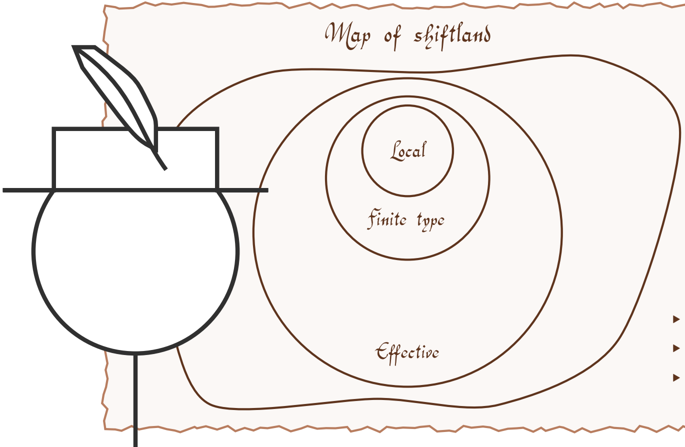
$$X_{\mathcal{F}} = \left\{ x \in \mathcal{A}^{\mathbb{Z}^d} : \forall w \in \mathcal{F},\, w \text{ does not appear in } x \right\}.$$

Classifying shift spaces by computational expressive power

Definition

A *shift space* on $\mathbb{Z}^d$ is a set of colorings $\mathbb{Z}^d \to \mathcal{A}$ ("*configurations*") defined by forbidden patterns $\mathcal{F}$:

$$X_{\mathcal{F}} = \left\{x \in \mathcal{A}^{\mathbb{Z}^d} : \forall w \in \mathcal{F},\ w \text{ does not appear in } x\right\}.$$



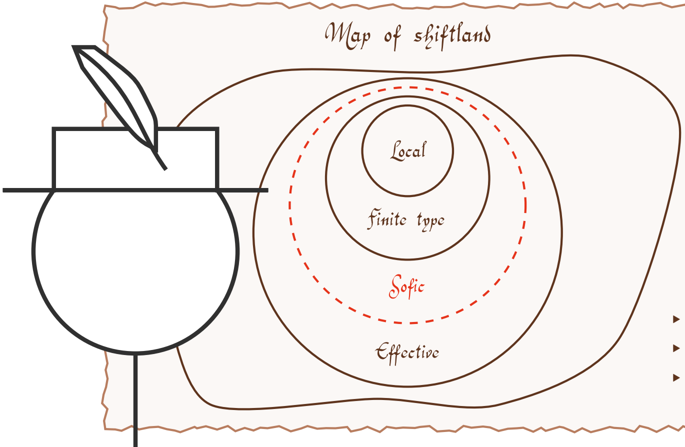
- Local: adjacency constraints
- Finite type: finite $\mathcal{F}$;
- Effective: computably enumerable $\mathcal{F}$.

Classifying shift spaces by computational expressive power

Definition

A *shift space* on $\mathbb{Z}^d$ is a set of colorings $\mathbb{Z}^d \to \mathcal{A}$ ("*configurations*") defined by forbidden patterns $\mathcal{F}$:

$$X_{\mathcal{F}} = \left\{x \in \mathcal{A}^{\mathbb{Z}^d} : \forall w \in \mathcal{F},\ w \text{ does not appear in } x\right\}.$$



- Local: adjacency constraints
- Finite type: finite $\mathcal{F}$;
- Effective: computably enumerable $\mathcal{F}$.

Classifying shift spaces by computational expressive power

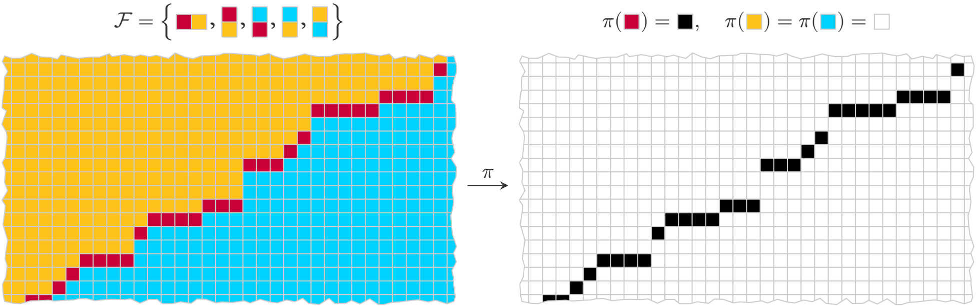
Definition

A *shift space* on $\mathbb{Z}^d$ is a set of colorings $\mathbb{Z}^d \to \mathcal{A}$ (“*configurations*”) defined by forbidden patterns $\mathcal{F}$:

$$X_{\mathcal{F}} = \left\{ x \in \mathcal{A}^{\mathbb{Z}^d} : \forall w \in \mathcal{F},\ w \text{ does not appear in } x \right\}.$$

Definition (Sofic space, ≃ 1973)

A shift space is *sofic* if it can be defined as the cell-by-cell projection of a local space.

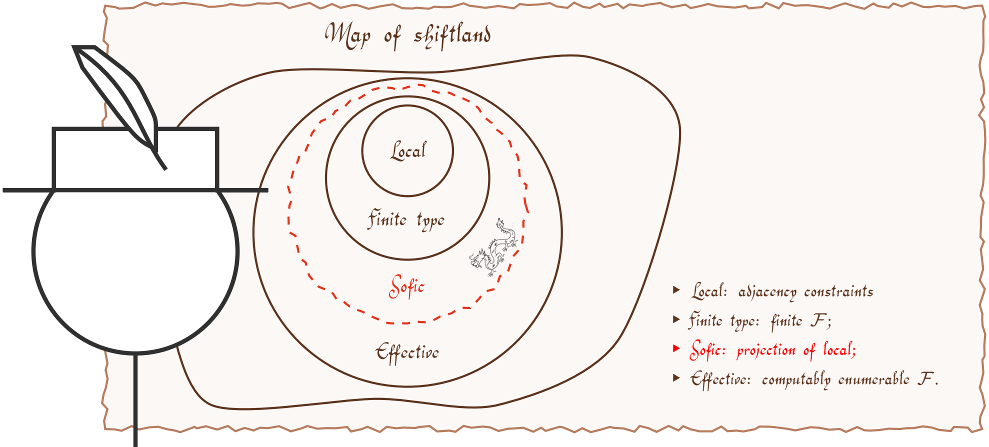


Classifying shift spaces by computational expressive power

Definition

A *shift space* on $\mathbb{Z}^d$ is a set of colorings $\mathbb{Z}^d \to \mathcal{A}$ ("*configurations*") defined by forbidden patterns $\mathcal{F}$:

$$X_{\mathcal{F}} = \left\{ x \in \mathcal{A}^{\mathbb{Z}^d} : \forall w \in \mathcal{F},\ w \text{ does not appear in } x \right\}.$$



Shift spaces on $\mathbb{Z}$

Shift space $X \subseteq \mathcal{A}^{\mathbb{Z}}$	**Language** $L \subseteq \mathcal{A}^*$
Local/finite type	Local
Sofic	Rational/regular
Effective	Comp. co-enumerable

Example

The sunny-side-up $X_{■} \subseteq \{□, ■\}^{\mathbb{Z}}$ is sofic.

Shift spaces on $\mathbb{Z}$

Shift space $X \subseteq \mathcal{A}^{\mathbb{Z}}$	**Language** $L \subseteq \mathcal{A}^*$
Local/finite type	Local
Sofic	Rational/regular
Effective	Comp. co-enumerable

Example

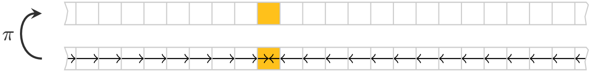
The sunny-side-up $X_{\blacksquare} \subseteq \{\square, \blacksquare\}^{\mathbb{Z}}$ is sofic.

Shift spaces on $\mathbb{Z}$

Shift space $X \subseteq \mathcal{A}^{\mathbb{Z}}$	**Language** $L \subseteq \mathcal{A}^*$
Local/finite type	Local
Sofic	Rational/regular
Effective	Comp. co-enumerable

Example

The sunny-side-up $X_{\blacksquare} \subseteq \{\square, \blacksquare\}^{\mathbb{Z}}$ is sofic.

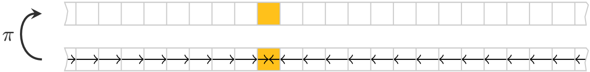


Shift spaces on $\mathbb{Z}$

Shift space $X \subseteq \mathcal{A}^{\mathbb{Z}}$	**Language** $L \subseteq \mathcal{A}^*$
Local/finite type	Local
Sofic	Rational/regular
Effective	Comp. co-enumerable

Example

The sunny-side-up $X_{■} \subseteq \{□, ■\}^{\mathbb{Z}}$ is sofic.



Example

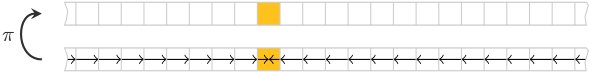
The space of all periods $X_p \subseteq \{□, ■\}^{\mathbb{Z}}$ is not sofic.

Shift spaces on $\mathbb{Z}$

Shift space $X \subseteq \mathcal{A}^{\mathbb{Z}}$	**Language** $L \subseteq \mathcal{A}^*$
Local/finite type	Local
Sofic	Rational/regular
Effective	Comp. co-enumerable

Example

The sunny-side-up $X_{■} \subseteq \{□, ■\}^{\mathbb{Z}}$ is sofic.



Example

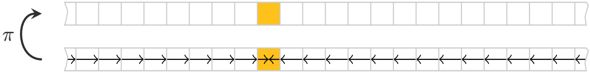
The space of all periods $X_p \subseteq \{□, ■\}^{\mathbb{Z}}$ is not sofic.

Shift spaces on $\mathbb{Z}$

Shift space $X \subseteq \mathcal{A}^{\mathbb{Z}}$	**Language** $L \subseteq \mathcal{A}^*$
Local/finite type	Local
Sofic	Rational/regular
Effective	Comp. co-enumerable

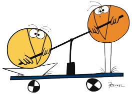
Example

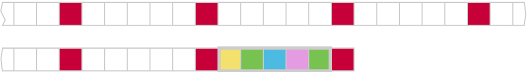
The sunny-side-up $X_{■} \subseteq \{□, ■\}^{\mathbb{Z}}$ is sofic.



Example

The space of all periods $X_p \subseteq \{□, ■\}^{\mathbb{Z}}$ is not sofic.



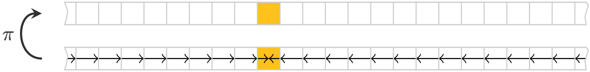


Shift spaces on $\mathbb{Z}$

Shift space $X \subseteq \mathcal{A}^{\mathbb{Z}}$	**Language** $L \subseteq \mathcal{A}^*$
Local/finite type	Local
Sofic	Rational/regular
Effective	Comp. co-enumerable

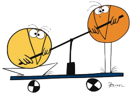
Example

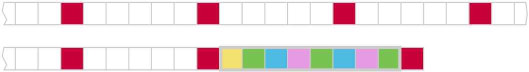
The sunny-side-up $X_{\blacksquare} \subseteq \{\square, \blacksquare\}^{\mathbb{Z}}$ is sofic.



Example

The space of all periods $X_p \subseteq \{\square, \blacksquare\}^{\mathbb{Z}}$ is not sofic.



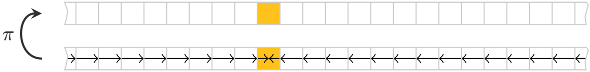


Shift spaces on $\mathbb{Z}$

Shift space $X \subseteq \mathcal{A}^{\mathbb{Z}}$	**Language** $L \subseteq \mathcal{A}^*$
Local/finite type	Local
Sofic	Rational/regular
Effective	Comp. co-enumerable

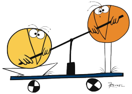
Example

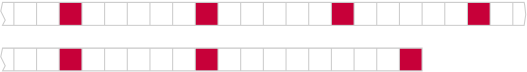
The sunny-side-up $X_{\blacksquare} \subseteq \{\square, \blacksquare\}^{\mathbb{Z}}$ is sofic.



Example

The space of all periods $X_p \subseteq \{\square, \blacksquare\}^{\mathbb{Z}}$ is not sofic.



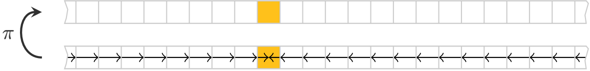


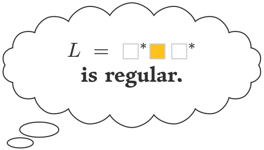
Shift spaces on $\mathbb{Z}$

Shift space $X \subseteq \mathcal{A}^{\mathbb{Z}}$	**Language** $L \subseteq \mathcal{A}^*$
Local/finite type	Local
Sofic	Rational/regular
Effective	Comp. co-enumerable

Example

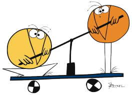
The sunny-side-up $X_{\blacksquare} \subseteq \{\square, \blacksquare\}^{\mathbb{Z}}$ is sofic.

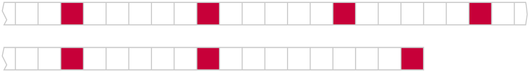


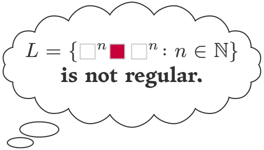


Example

The space of all periods $X_p \subseteq \{\square, \blacksquare\}^{\mathbb{Z}}$ is not sofic.





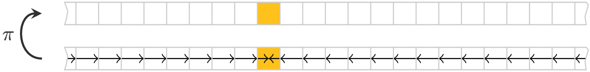


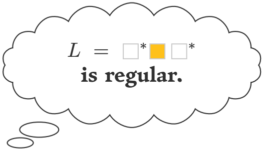
Shift spaces on $\mathbb{Z}$

Shift space $X \subseteq \mathcal{A}^{\mathbb{Z}}$	**Language** $L \subseteq \mathcal{A}^*$
Local/finite type	Local
Sofic	Rational/regular
Effective	Comp. co-enumerable

Example

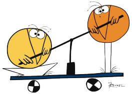
The sunny-side-up $X_{\blacksquare} \subseteq \{\square, \blacksquare\}^{\mathbb{Z}}$ is sofic.

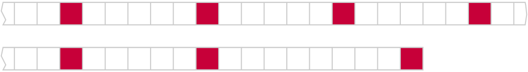


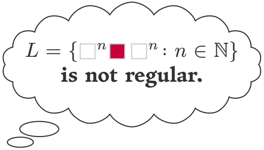


Example

The space of all periods $X_p \subseteq \{\square, \blacksquare\}^{\mathbb{Z}}$ is not sofic.







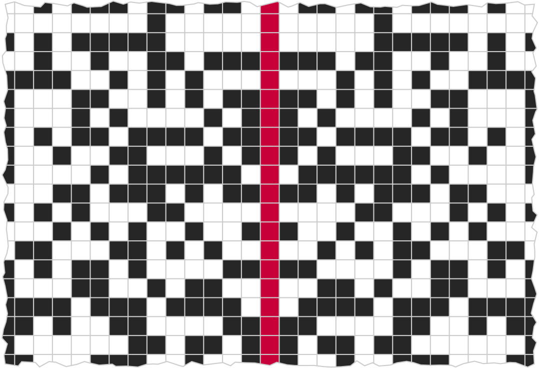
Lemma (Myhill-Nerode)

A shift space $X \subseteq \mathcal{A}^{\mathbb{Z}}$ is sofic if and only if it defines finitely many *extender sets* (*i.e.* “contexts”).

Soficity of shift spaces on $\mathbb{Z}^d$: information bounds

Example

The mirror shift space is **not** sofic on $\mathbb{Z}^d$ for any $d \in \mathbb{N}$:



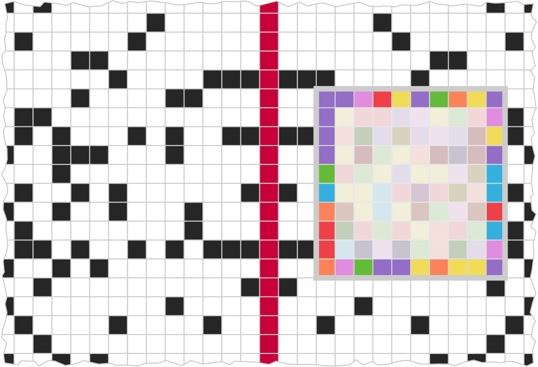
(folklore)

Soficity of shift spaces on $\mathbb{Z}^d$: information bounds

Example

The mirror shift space is **not** sofic on $\mathbb{Z}^d$ for any $d \in \mathbb{N}$:

Pattern $O(n^d)$;
Border $O(n^{d-1})$;



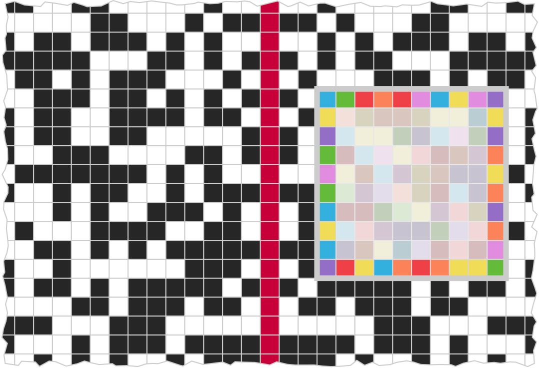
(folklore)

Soficity of shift spaces on $\mathbb{Z}^d$: information bounds

Example

The mirror shift space is **not** sofic on $\mathbb{Z}^d$ for any $d \in \mathbb{N}$:

Pattern $O(n^d)$;
Border $O(n^{d-1})$;



(folklore)

Soficity of shift spaces on $\mathbb{Z}^d$: information bounds

Example

The mirror shift space is **not** sofic on $\mathbb{Z}^d$ for any $d \in \mathbb{N}$:

Pattern $O(n^d)$;
Border $O(n^{d-1})$;

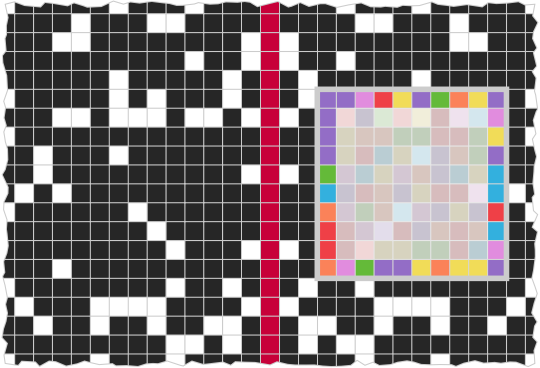
(folklore)

Soficity of shift spaces on $\mathbb{Z}^d$: information bounds

Example

The mirror shift space is **not** sofic on $\mathbb{Z}^d$ for any $d \in \mathbb{N}$:

Pattern $O(n^d)$;
Border $O(n^{d-1})$;



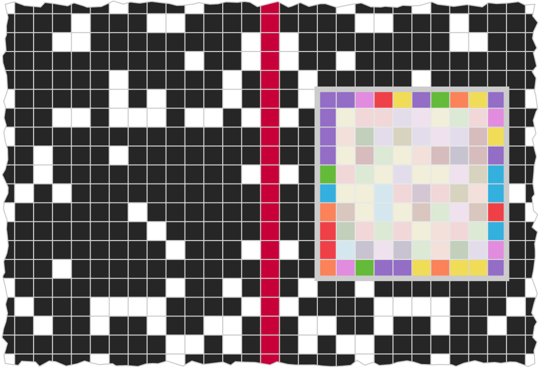
(folklore)

Soficity of shift spaces on $\mathbb{Z}^d$: information bounds

Example

The mirror shift space is **not** sofic on $\mathbb{Z}^d$ for any $d \in \mathbb{N}$:

Pattern $O(n^d)$;
Border $O(n^{d-1})$;



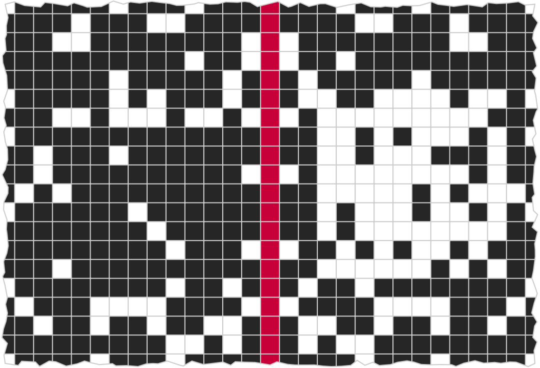
(folklore)

Soficity of shift spaces on $\mathbb{Z}^d$: information bounds

Example

The mirror shift space is **not** sofic on $\mathbb{Z}^d$ for any $d \in \mathbb{N}$:

Pattern $O(n^d)$;
Border $O(n^{d-1})$;



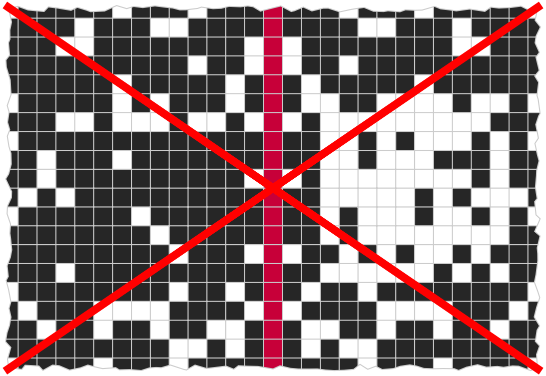
(folklore)

Soficity of shift spaces on $\mathbb{Z}^d$: information bounds

Example

The mirror shift space is **not** sofic on $\mathbb{Z}^d$ for any $d \in \mathbb{N}$:

Pattern $O(n^d)$;
Border $O(n^{d-1})$;



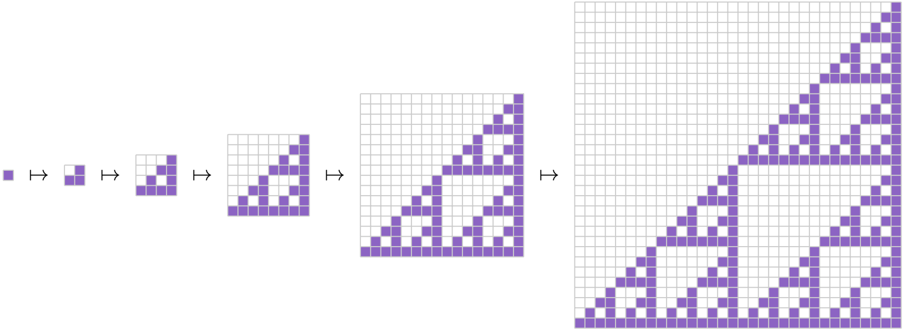
(folklore)

Intuition

Patterns of domain $[\![n]\!]^d$ in a sofic shift can only exchange $O(n^{d-1})$ bits with their exterior.

Soficity of shift spaces on $\mathbb{Z}^d$: examples

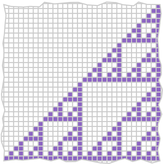
- Substitutive shifts;
  [Mozes, 1989]



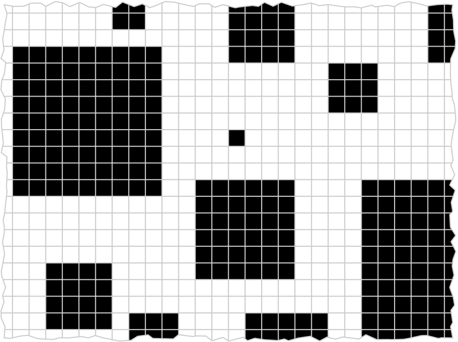
τ: 

Soficity of shift spaces on $\mathbb{Z}^d$: examples

- Substitutive shifts;
  [Mozes, 1989]

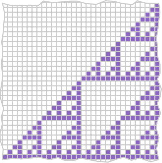


- Seas of squares;

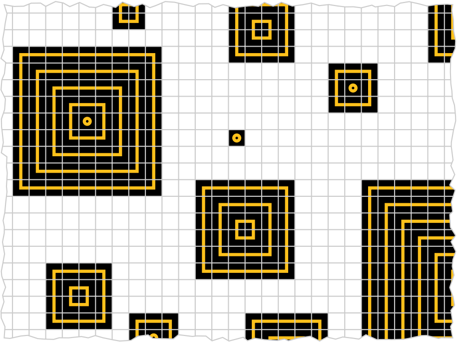


Soficity of shift spaces on $\mathbb{Z}^d$: examples

- Substitutive shifts;
  [Mozes, 1989]

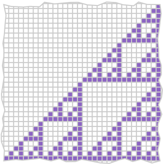


- Seas of squares;

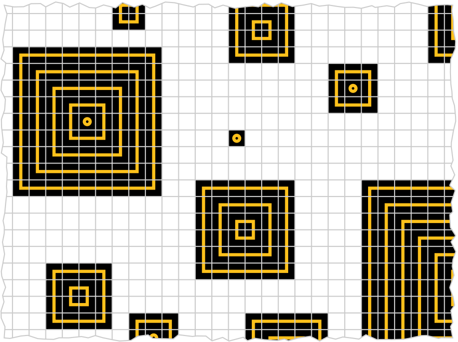


Soficity of shift spaces on $\mathbb{Z}^d$: examples

- Substitutive shifts;
  [Mozes, 1989]

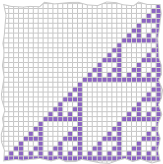


- Seas of squares (of Π_1^0 sizes);
  [Westrick, 2017]

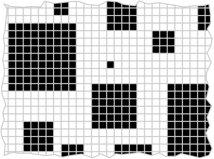


Soficity of shift spaces on $\mathbb{Z}^d$: examples

- Substitutive shifts;
  [Mozes, 1989]



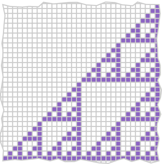
- Seas of squares (of Π^0_1 sizes);
  [Westrick, 2017]



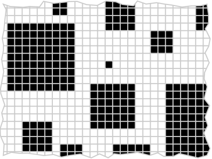
- Extensions of effective $\mathbb{Z}$ shifts;
  [Hochman, ... 2009+]

Soficity of shift spaces on $\mathbb{Z}^d$: examples

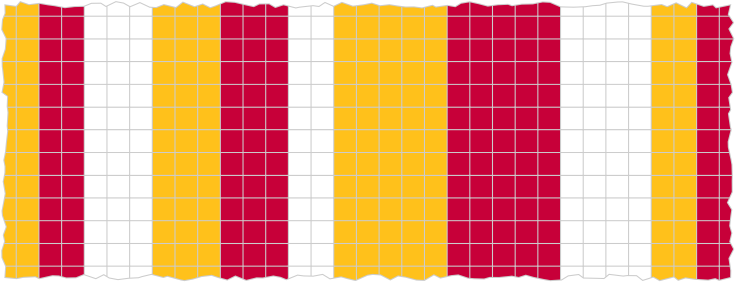
- Substitutive shifts; [Mozes, 1989]



- Seas of squares (of Π_1^0 sizes); [Westrick, 2017]

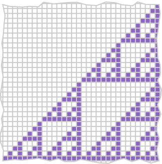


- Extensions of effective $\mathbb{Z}$ shifts; [Hochman, … 2009+]

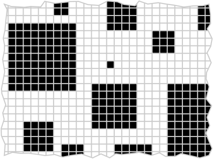


Soficity of shift spaces on $\mathbb{Z}^d$: examples

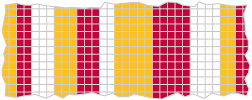
- Substitutive shifts;
  [Mozes, 1989]



- Seas of squares (of Π_1^0 sizes);
  [Westrick, 2017]



- Extensions of effective $\mathbb{Z}$ shifts;
  [Hochman, … 2009+]



Fact

All these shifts can be proved sofic on $\mathbb{Z}^d$ $(d \geq 2)$ using the *fixed point construction*.

[Durand, Romashchenko & Shen, 2008+]

Sofic realization of s-adic limit spaces by self-simulation

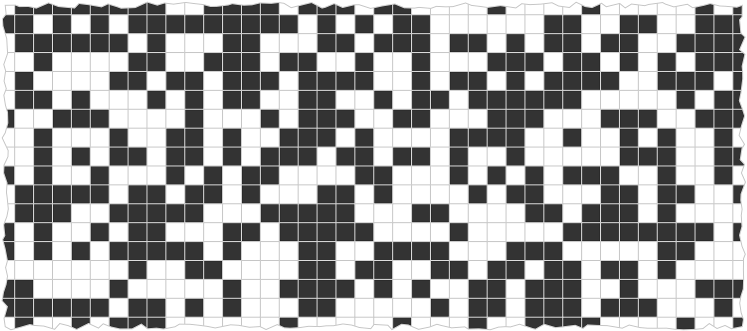
Substitutions

Definition

Given two alphabets $\mathcal{A}$, $\mathcal{B}$, and a zoom $N \in \mathbb{N}$, a *simulation* is an injective map

$$\mathcal{S}\colon \mathcal{A}^{[\![N]\!]^d} \rightrightarrows \mathcal{B}.$$

A simulation $\mathcal{S}$ induces a relation between configurations:



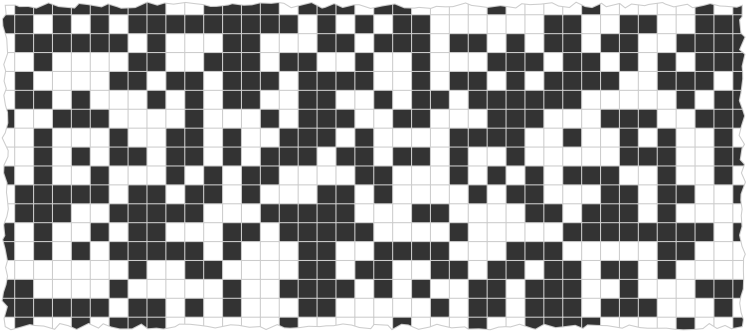
Substitutions

Definition

Given two alphabets $\mathcal{A}$, $\mathcal{B}$, and a zoom $N \in \mathbb{N}$, a *simulation* is an injective map

$$\mathcal{S} \colon \mathcal{A}^{[\![N]\!]^d} \rightrightarrows \mathcal{B}.$$

A simulation $\mathcal{S}$ induces a relation between configurations:



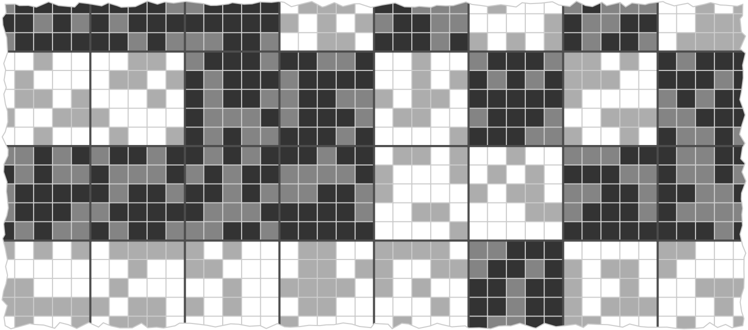
Substitutions

Definition

Given two alphabets $\mathcal{A}$, $\mathcal{B}$, and a zoom $N \in \mathbb{N}$, a *simulation* is an injective map

$$\mathcal{S} \colon \mathcal{A}^{[\![N]\!]^d} \rightrightarrows \mathcal{B}.$$

A simulation $\mathcal{S}$ induces a relation between configurations:



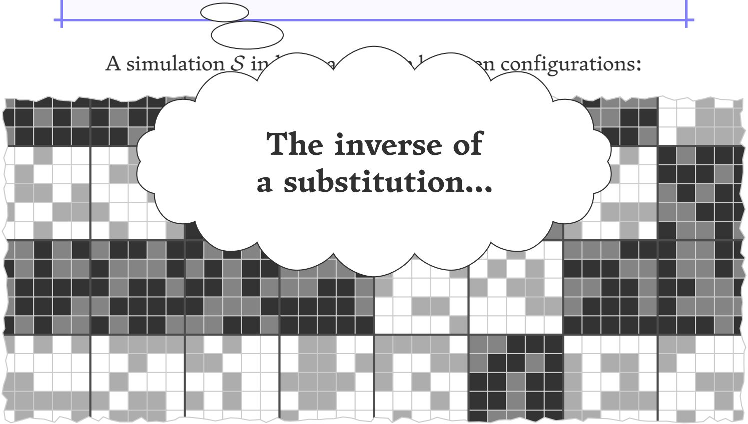
Substitutions

Definition

Given two alphabets $\mathcal{A}$, $\mathcal{B}$, and a zoom $N \in \mathbb{N}$, a *simulation* is an injective map

$$\mathcal{S} \colon \mathcal{A}^{[\![N]\!]^d} \rightrightarrows \mathcal{B}.$$

A simulation $\mathcal{S}$ in … n configurations:



Main statement

Definition

For a set of colors $C \subseteq \{0,1\}^*$, a *Wang tile* is some $t \in C^{2d+1}$:

A set of tiles $T \subseteq C^{2d+1}$ defines a local shift space X_T.

Definition

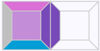
Given two alphabets $\mathcal{A}$, $\mathcal{B}$, and a zoom $N \in \mathbb{N}$, a *simulation* is an injective map

$$\mathcal{S} \colon \mathcal{A}^{[\![N]\!]^d} \rightrightarrows \mathcal{B}.$$

Main statement

Definition

For a set of colors $C \subseteq \{0,1\}^*$, a *Wang tile* is some $t \in C^{2d+1}$:



A set of tiles $T \subseteq C^{2d+1}$ defines a local shift space X_T.

Definition

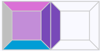
Given two alphabets $\mathcal{A}$, $\mathcal{B}$, and a zoom $N \in \mathbb{N}$, a *simulation* is an injective map

$$\mathcal{S} \colon \mathcal{A}^{[\![N]\!]^d} \rightrightarrows \mathcal{B}.$$

Main statement

Definition

For a set of colors $C \subseteq \{0,1\}^*$, a *Wang tile* is some $t \in C^{2d+1}$:



A set of tiles $T \subseteq C^{2d+1}$ defines a local shift space X_T.

Definition

Given two alphabets $\mathcal{A}$, $\mathcal{B}$, and a zoom $N \in \mathbb{N}$, a *simulation* is an injective map

$$\mathcal{S} \colon \mathcal{A}^{[\![N]\!]^d} \rightrightarrows \mathcal{B}.$$

Theorem

Let $(X_{T_\ell})_{\ell \in \mathbb{N}}$ be a sequence of tiling spaces on colors $(C_\ell)_{\ell \in \mathbb{N}}$ and

$$\mathcal{S}_\ell \colon T_\ell^{[\![N_\ell]\!]^d} \rightrightarrows T_{\ell+1}$$

be simulations between T_ℓ and $T_{\ell+1}$. Denote $L_\ell = \prod_{i=0}^{\ell-1} N_i$. If:

(i) All colors $c \in C_{\ell+1}$ are of size $O(L_\ell^\alpha)$ for some $\alpha < d-1$;

(ii) The simulations $\mathcal{S}_\ell$ are computable in time $O(L_\ell^\alpha)$;

(iii) $N_\ell \in [\![2, 2^{O(L_\ell^\delta)}]\!]$ for some $\delta < \frac{d-1}{2}$;

then

$$\left\{ x \in X_{T_0} : x = \underset{\in X_{T_0}}{x_0} \xrightarrow{\mathcal{S}_0} \underset{\in X_{T_1}}{x_1} \xrightarrow{\mathcal{S}_1} \underset{\in X_{T_2}}{x_2} \xrightarrow{\mathcal{S}_2} \dots \right\}$$

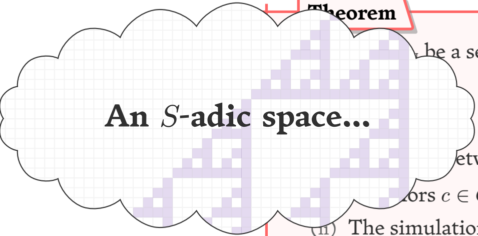
is a sofic shift.

Main statement

Definition

A sequence of substitutions $(\tau_\ell)_{\ell\in\mathbb{N}}$ (with $\tau_\ell\colon \mathcal{A}_{\ell+1} \rightrightarrows \mathcal{A}_\ell^{[\![N_\ell]\!]^d}$) defines an *$S$-adic limit space*

$$\left\{x \in \mathcal{A}_0^{\mathbb{Z}^d} : x = x_0 \xleftarrow{\tau_0} x_1 \xleftarrow{\tau_1} x_2 \xleftarrow{\tau_2} \dots\right\}$$



Theorem

be a sequence of tiling spaces on colors $(C_\ell)_{\ell\in\mathbb{N}}$ and

$$\mathcal{S}_\ell\colon T_\ell^{[\![N_\ell]\!]^d} \rightrightarrows T_{\ell+1}$$

etween T_ℓ and $T_{\ell+1}$. Denote $L_\ell = \prod_{i=0}^{\ell-1} N_i$. If:

ors $c \in C_{\ell+1}$ are of size $O(L_\ell^\alpha)$ for some $\alpha < d-1$;

(ii) The simulations $\mathcal{S}_\ell$ are computable in time $O(L_\ell^\alpha)$;

(iii) $N_\ell \in [\![2, 2^{O(L_\ell^\delta)}]\!]$ for some $\delta < \frac{d-1}{2}$;

then

$$\left\{x \in X_{T_0} : x = \underset{\in X_{T_0}}{x_0} \xrightarrow{\mathcal{S}_0} \underset{\in X_{T_1}}{x_1} \xrightarrow{\mathcal{S}_1} \underset{\in X_{T_2}}{x_2} \xrightarrow{\mathcal{S}_2} \dots\right\}$$

is a sofic shift.

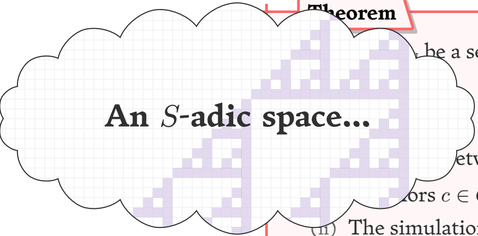
Main statement

Definition

A sequence of substitutions $(\tau_\ell)_{\ell\in\mathbb{N}}$ (with $\tau_\ell \colon \mathcal{A}_{\ell+1} \rightrightarrows \mathcal{A}_\ell^{[\![N_\ell]\!]^d}$) defines an *S-adic limit space*

$$\left\{x \in \mathcal{A}_0^{\mathbb{Z}^d} : x = x_0 \xleftarrow{\tau_0} x_1 \xleftarrow{\tau_1} x_2 \xleftarrow{\tau_2} \dots\right\}$$

Theorem



be a sequence of tiling spaces on colors (C_ℓ)

$$\mathcal{S}_\ell \colon T_\ell^{[\![N_\ell]\!]^d} \rightrightarrows T_{\ell+1}$$

etween T_ℓ and $T_{\ell+1}$. Denote $L_\ell = \prod_{i=0}^{\ell-1} N_i$. If:

ors $c \in C_{\ell+1}$ are of size $O(L_\ell^\alpha)$ for some $\alpha < d-1$;

(ii) The simulations $\mathcal{S}_\ell$ are computable in time $O(L_\ell^\alpha)$;

(iii) $N_\ell \in [\![2, 2^{O(L_\ell^\delta)}]\!]$ for some $\delta < \frac{d-1}{2}$;

then

$$\left\{x \in X_{T_0} : x = \underset{\in X_{T_0}}{x_0} \xrightarrow{\mathcal{S}_0} \underset{\in X_{T_1}}{x_1} \xrightarrow{\mathcal{S}_1} \underset{\in X_{T_2}}{x_2} \xrightarrow{\mathcal{S}_2} \dots\right\}$$

is a sofic shift.

Soficity: information $o(N^{d-1})$ in cubes $[\![N]\!]^d$

Example 1: seas of squares

Theorem

Let X_{T_ℓ} be a sequence of tiling spaces on colors $(C_\ell)_{\ell\in\mathbb{N}}$ and

$$\mathcal{S}_\ell\colon T_\ell^{[\![N_\ell]\!]^d} \rightrightarrows T_{\ell+1}$$

be simulations between X_{T_ℓ} and $X_{T_{\ell+1}}$. Denote $L_\ell = \prod_{i=0}^{\ell-1} N_i$. If:

(i) All colors $c \in C_{\ell+1}$ are of size $O(L_\ell^\alpha)$ for some $\alpha < d-1$;

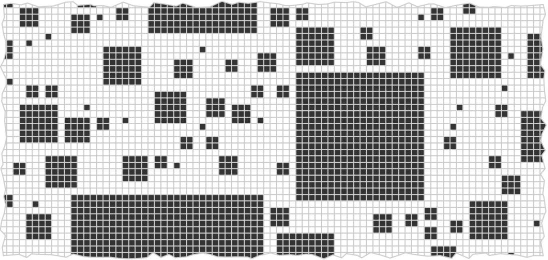
(ii) The simulations $\mathcal{S}_\ell$ are computable in time $O(L_\ell^\alpha)$;

(iii) $N_\ell \in [\![2, 2^{L_\ell^\delta}]\!]$ for some $\delta < \frac{d-1}{2}$;

then

$$\left\{x \in X_{T_0} : x = \underset{\in X_{T_0}}{x_0} \xrightarrow{\mathcal{S}_0} \underset{\in X_{T_1}}{x_1} \xrightarrow{\mathcal{S}_1} \underset{\in X_{T_2}}{x_2} \xrightarrow{\mathcal{S}_2} \dots\right\}$$

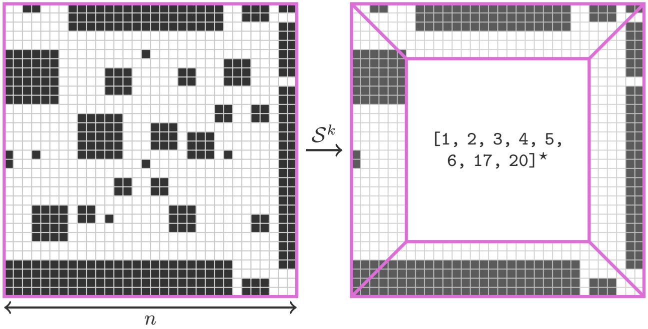
is a sofic shift.

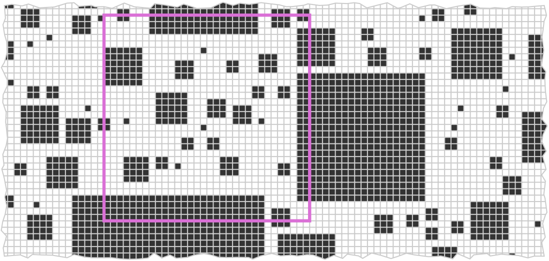


Seas of squares: [Westrick, 2017]

For $S \subseteq \mathbb{N}$ a Π_1^0-computable set, X draws independent ■ squares of sizes in S over a □ background.

Example 1: seas of squares





Theorem

Let X_{T_ℓ} be a sequence of tiling spaces on colors $(C_\ell)_{\ell\in\mathbb{N}}$ and

$$\mathcal{S}_\ell : T_\ell^{[\![N_\ell]\!]^d} \rightrightarrows T_{\ell+1}$$

be simulations between X_{T_ℓ} and $X_{T_{\ell+1}}$. Denote $L_\ell = \prod_{i=0}^{\ell-1} N_i$. If:

(i) All colors $c \in C_{\ell+1}$ are of size $O(L_\ell^\alpha)$ for some $\alpha < d-1$;

(ii) The simulations $\mathcal{S}_\ell$ are computable in time $O(L_\ell^\alpha)$;

(iii) $N_\ell \in [\![2, 2^{L_\ell^\delta}]\!]$ for some $\delta < \frac{d-1}{2}$;

then

$$\left\{ x \in X_{T_0} : x = \underset{\in X_{T_0}}{x_0} \overset{\mathcal{S}_0}{\longrightarrow} \underset{\in X_{T_1}}{x_1} \overset{\mathcal{S}_1}{\longrightarrow} \underset{\in X_{T_2}}{x_2} \overset{\mathcal{S}_2}{\longrightarrow} \dots \right\}$$

is a sofic shift.

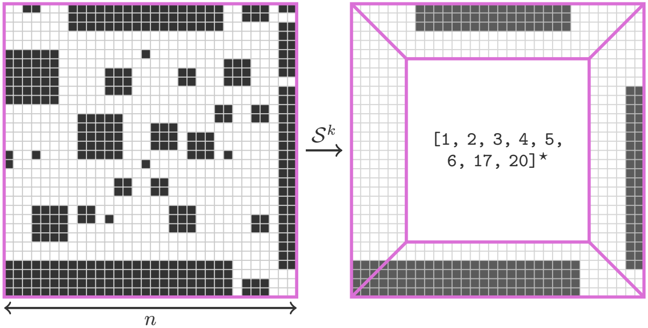
* Check the lengths against an enumeration of S.

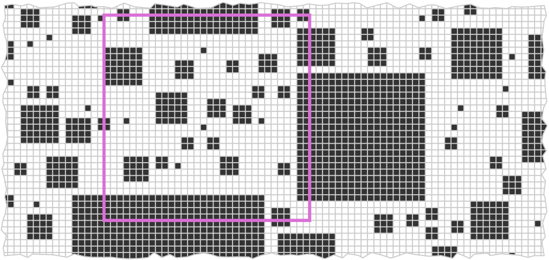
Seas of squares: [Westrick, 2017]

For $S \subseteq \mathbb{N}$ a Π_1^0-computable set, X draws independent ■ squares of sizes in S over a □ background.

For a pattern of domain $[\![n]\!]^2$:
colors of size $O(n^{2/3} \cdot \log n)$.

Example 1: seas of squares





Theorem

Let X_{T_ℓ} be a sequence of tiling spaces on colors $(C_\ell)_{\ell \in \mathbb{N}}$ and

$$\mathcal{S}_\ell : T_\ell^{[\![N_\ell]\!]^d} \rightrightarrows T_{\ell+1}$$

be simulations between X_{T_ℓ} and $X_{T_{\ell+1}}$. Denote $L_\ell = \prod_{i=0}^{\ell-1} N_i$. If:

(i) All colors $c \in C_{\ell+1}$ are of size $O(L_\ell^\alpha)$ for some $\alpha < d-1$;

(ii) The simulations $\mathcal{S}_\ell$ are computable in time $O(L_\ell^\alpha)$;

(iii) $N_\ell \in [\![2, 2^{L_\ell^\delta}]\!]$ for some $\delta < \frac{d-1}{2}$;

then

$$\left\{ x \in X_{T_0} : x = \underset{\in X_{T_0}}{x_0} \xrightarrow{\mathcal{S}_0} \underset{\in X_{T_1}}{x_1} \xrightarrow{\mathcal{S}_1} \underset{\in X_{T_2}}{x_2} \xrightarrow{\mathcal{S}_2} \dots \right\}$$

is a sofic shift.

* Check the lengths against an enumeration of S.

Seas of squares: [Westrick, 2017]

For $S \subseteq \mathbb{N}$ a Π_1^0-computable set, X draws independent ■ squares of sizes in S over a □ background.

For a pattern of domain $[\![n]\!]^2$:
colors of size $O(n^{2/3} \cdot \log n)$.

Example 2: density

Theorem

Let X_{T_ℓ} be a sequence of tiling spaces on colors $(C_\ell)_{\ell\in\mathbb{N}}$ and

$$\mathcal{S}_\ell \colon T_\ell^{[\![N_\ell]\!]^d} \rightrightarrows T_{\ell+1}$$

be simulations between X_{T_ℓ} and $X_{T_{\ell+1}}$. Denote $L_\ell = \prod_{i=0}^{\ell-1} N_i$. If:

(i) All colors $c \in C_{\ell+1}$ are of size $O(L_\ell^\alpha)$ for some $\alpha < d-1$;

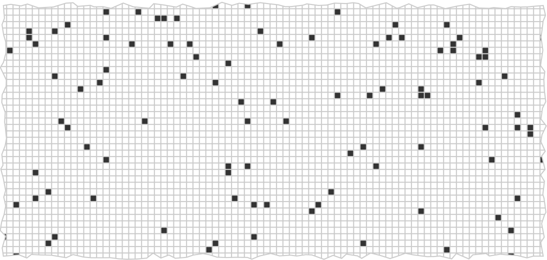
(ii) The simulations $\mathcal{S}_\ell$ are computable in time $O(L_\ell^\alpha)$;

(iii) $N_\ell \in [\![2, 2^{L_\ell^\delta}]\!]$ for some $\delta < \frac{d-1}{2}$;

then

$$\left\{x \in X_{T_0} : x = \underset{\in X_{T_0}}{x_0} \xrightarrow{\mathcal{S}_0} \underset{\in X_{T_1}}{x_1} \xrightarrow{\mathcal{S}_1} \underset{\in X_{T_2}}{x_2} \xrightarrow{\mathcal{S}_2} \dots\right\}$$

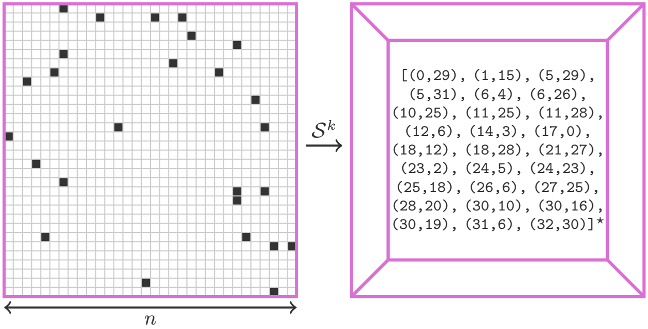
is a sofic shift.

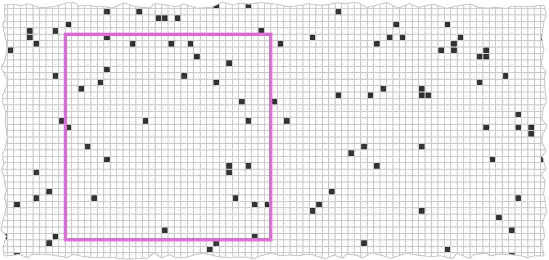


Density shifts: [Destombes, 2021]

$X \subseteq \{■, □\}^{\mathbb{Z}^d}$ effective such that patterns of domain $[\![n]\!]^d$ contain $O(n^\alpha)$ ■ symbols for some $\alpha < d-1$.

Example 2: density





Theorem

Let X_{T_ℓ} be a sequence of tiling spaces on colors $(C_\ell)_{\ell\in\mathbb{N}}$ and

$$\mathcal{S}_\ell : T_\ell^{[\![N_\ell]\!]^d} \rightrightarrows T_{\ell+1}$$

be simulations between X_{T_ℓ} and $X_{T_{\ell+1}}$. Denote $L_\ell = \prod_{i=0}^{\ell-1} N_i$. If:

(i) All colors $c \in C_{\ell+1}$ are of size $O(L_\ell^\alpha)$ for some $\alpha < d-1$;
(ii) The simulations $\mathcal{S}_\ell$ are computable in time $O(L_\ell^\alpha)$;
(iii) $N_\ell \in [\![2, 2^{L_\ell^\delta}]\!]$ for some $\delta < \frac{d-1}{2}$;

then

$$\left\{x \in X_{T_0} : x = \underset{\in X_{T_0}}{x_0} \xrightarrow{\mathcal{S}_0} \underset{\in X_{T_1}}{x_1} \xrightarrow{\mathcal{S}_1} \underset{\in X_{T_2}}{x_2} \xrightarrow{\mathcal{S}_2} \dots \right\}$$

is a sofic shift.

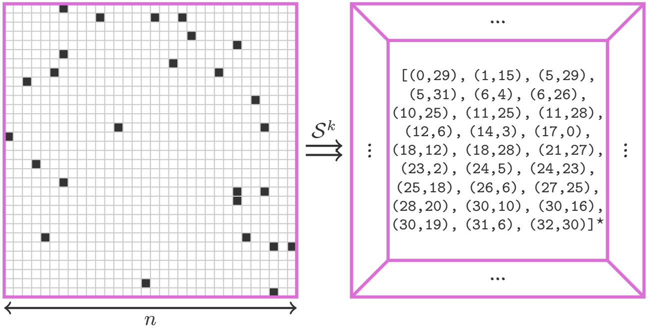
* Check ■ positions against an enumeration of $\mathcal{F}$.

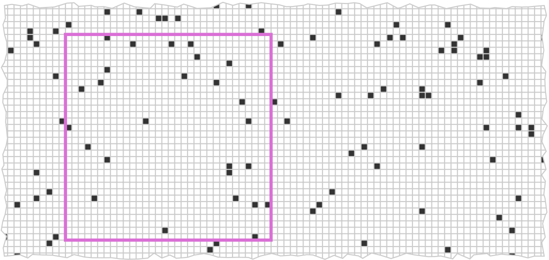
Density shifts: [Destombes, 2021]

$X \subseteq \{■, □\}^{\mathbb{Z}^d}$ effective such that patterns of domain $[\![n]\!]^d$ contain $O(n^\alpha)$ ■ symbols for some $\alpha < d-1$.

For a pattern of domain $[\![n]\!]^d$:
colors of size $O(n^\alpha)$.

Example 2: density





Theorem

Let X_{T_ℓ} be a sequence of tiling spaces on colors $(C_\ell)_{\ell\in\mathbb{N}}$ and

$$\mathcal{S}_\ell \colon T_\ell^{[\![N_\ell]\!]^d} \rightrightarrows T_{\ell+1}$$

be simulations between X_{T_ℓ} and $X_{T_{\ell+1}}$. Denote $L_\ell = \prod_{i=0}^{\ell-1} N_i$. If:

(i) All colors $c \in C_{\ell+1}$ are of size $O(L_\ell^\alpha)$ for some $\alpha < d-1$;

(ii) The simulations $\mathcal{S}_\ell$ are computable in time $O(L_\ell^\alpha)$;

(iii) $N_\ell \in [\![2, 2^{L_\ell^\delta}]\!]$ for some $\delta < \frac{d-1}{2}$;

then

$$\left\{x \in X_{T_0} : x = \underset{\in X_{T_0}}{x_0} \xrightarrow{\mathcal{S}_0} \underset{\in X_{T_1}}{x_1} \xrightarrow{\mathcal{S}_1} \underset{\in X_{T_2}}{x_2} \xrightarrow{\mathcal{S}_2} \dots\right\}$$

is a sofic shift.

* Check ■ positions against an enumeration of $\mathcal{F}$.

Density shifts: [Destombes, 2021]

$X \subseteq \{■, □\}^{\mathbb{Z}^d}$ effective such that patterns of domain $[\![n]\!]^d$ contain $O(n^\alpha)$ ■ symbols for some $\alpha < d-1$.

For a pattern of domain $[\![n]\!]^d$:
colors of size $O(n^\alpha)$.

Example 3: periodic extensions

Theorem

Let X_{T_ℓ} be a sequence of tiling spaces on colors $(C_\ell)_{\ell\in\mathbb{N}}$ and

$$\mathcal{S}_\ell \colon T_\ell^{[\![N_\ell]\!]^d} \rightrightarrows T_{\ell+1}$$

be simulations between X_{T_ℓ} and $X_{T_{\ell+1}}$. Denote $L_\ell = \prod_{i=0}^{\ell-1} N_i$. If:

(i) All colors $c \in C_{\ell+1}$ are of size $O(L_\ell^\alpha)$ for some $\alpha < d-1$;

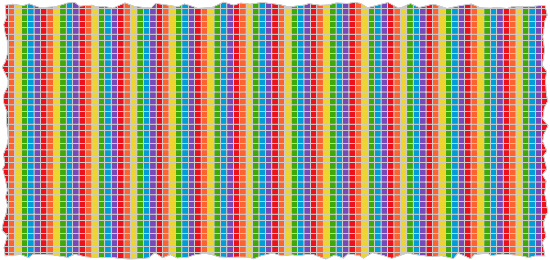
(ii) The simulations $\mathcal{S}_\ell$ are computable in time $O(L_\ell^\alpha)$;

(iii) $N_\ell \in [\![2, 2^{L_\ell^\delta}]\!]$ for some $\delta < \frac{d-1}{2}$;

then

$$\left\{x \in X_{T_0} : x = \underset{\in X_{T_0}}{x_0} \xrightarrow{\mathcal{S}_0} \underset{\in X_{T_1}}{x_1} \xrightarrow{\mathcal{S}_1} \underset{\in X_{T_2}}{x_2} \xrightarrow{\mathcal{S}_2} \dots\right\}$$

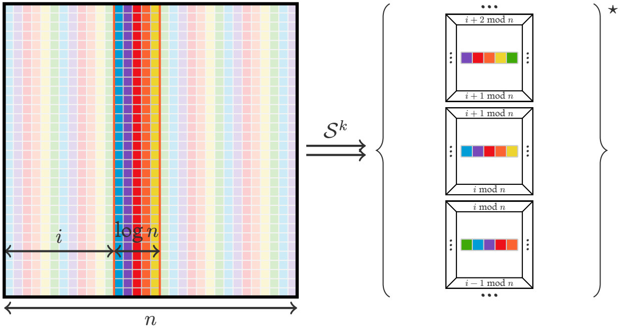
is a sofic shift.



Periodic extensions: [DRS, 2010]

Given $X \subseteq \mathcal{A}^{\mathbb{Z}^d}$ effective, the configurations of $\mathcal{A}^{\mathbb{Z}^{d+1}}$ repeating some facet $x \in X$ vertically.

Example 3: periodic extensions



Theorem

Let X_{T_ℓ} be a sequence of tiling spaces on colors $(C_\ell)_{\ell\in\mathbb{N}}$ and

$$\mathcal{S}_\ell\colon T_\ell^{[\![N_\ell]\!]^d} \rightrightarrows T_{\ell+1}$$

be simulations between X_{T_ℓ} and $X_{T_{\ell+1}}$. Denote $L_\ell = \prod_{i=0}^{\ell-1} N_i$. If:

(i) All colors $c \in C_{\ell+1}$ are of size $O(L_\ell^\alpha)$ for some $\alpha < d-1$;

(ii) The simulations $\mathcal{S}_\ell$ are computable in time $O(L_\ell^\alpha)$;

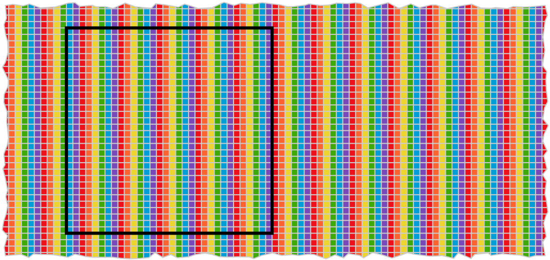
(iii) $N_\ell \in [\![2, 2^{L_\ell^\delta}]\!]$ for some $\delta < \frac{d-1}{2}$;

then

$$\left\{x \in X_{T_0} : x = \underset{\in X_{T_0}}{x_0} \xrightarrow{\mathcal{S}_0} \underset{\in X_{T_1}}{x_1} \xrightarrow{\mathcal{S}_1} \underset{\in X_{T_2}}{x_2} \xrightarrow{\mathcal{S}_2} \dots\right\}$$

is a sofic shift.

★ Check middle patterns against an enumeration of $\mathcal{F}$.



Periodic extensions: [DRS, 2010]

Given $X \subseteq \mathcal{A}^{\mathbb{Z}^d}$ effective, the configurations of $\mathcal{A}^{\mathbb{Z}^{d+1}}$ repeating some facet $x \in X$ vertically.

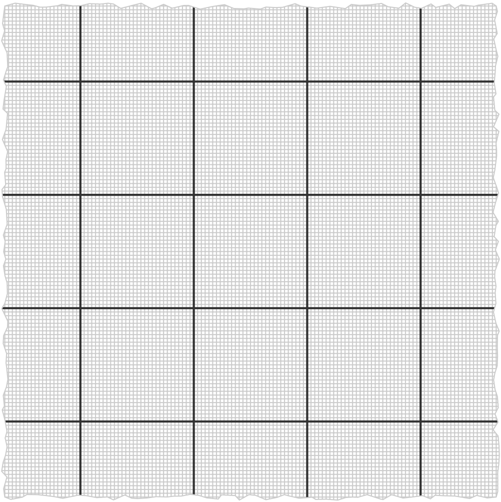
For a pattern of domain $[\![n]\!]^d$:
colors of size $O(\log n)$.

Proof outline

The fixpoint construction:

Define a sequence of tilesets $(T_\ell)_\ell$ as follows:

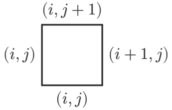
Proof outline



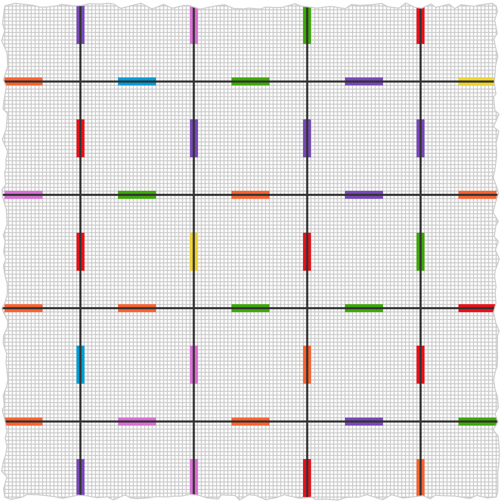
The fixpoint construction:

Define a sequence of tilesets $(T_\ell)_\ell$ as follows:

1. Encode the position of each tile in a block $[\![N_\ell]\!]^d$



Proof outline



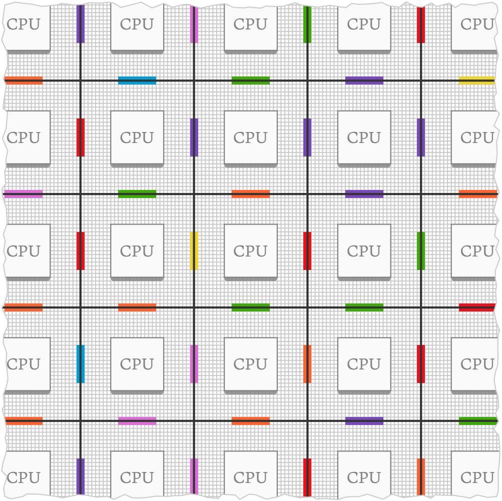
The fixpoint construction:

Define a sequence of tilesets $(T_\ell)_\ell$ as follows:

1. Encode the position of each tile in a block $[\![N_\ell]\!]^d$
2. Some border tiles contain a bit $b \in \{0,1\}$: their concatenation forms a *macro-color* $c \in \{0,1\}^*$;

If $i = N_\ell - 1$ and $j \simeq N_\ell/2$:

Proof outline

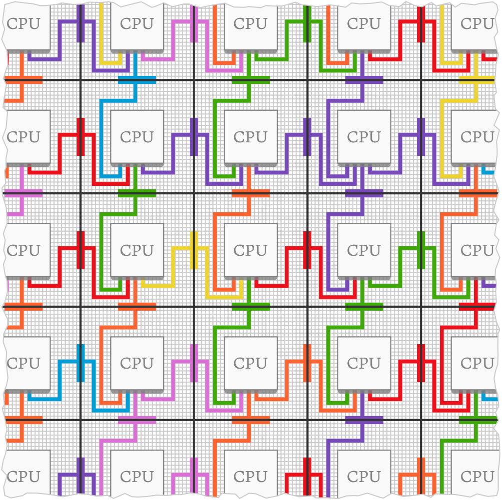


The fixpoint construction:

Define a sequence of tilesets $(T_\ell)_\ell$ as follows:

1. Encode the position of each tile in a block $[\![N_\ell]\!]^d$
2. Some border tiles contain a bit $b \in \{0,1\}$: their concatenation forms a *macro-color* $c \in \{0,1\}^*$;
3. Embed computations to “program” $T_{\ell+1}$;
   role: check that a tuple of macro-colors emulates a tile of $T_{\ell+1}$

Proof outline

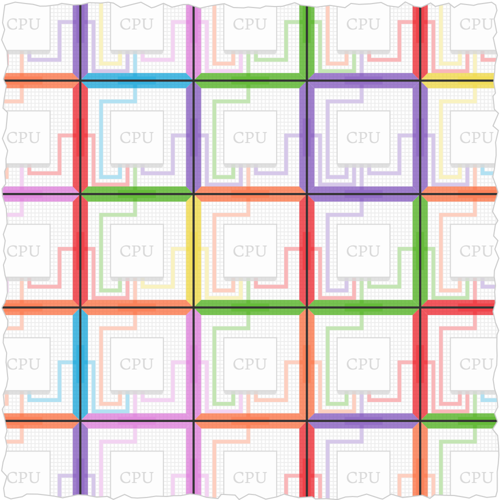


The fixpoint construction:

Define a sequence of tilesets $(T_\ell)_\ell$ as follows:

1. Encode the position of each tile in a block $[\![N_\ell]\!]^d$
2. Some border tiles contain a bit $b \in \{0,1\}$: their concatenation forms a *macro-color* $c \in \{0,1\}^*$;
3. Embed computations to "program" $T_{\ell+1}$;
   role: check that a tuple of macro-colors emulates a tile of $T_{\ell+1}$

Proof outline

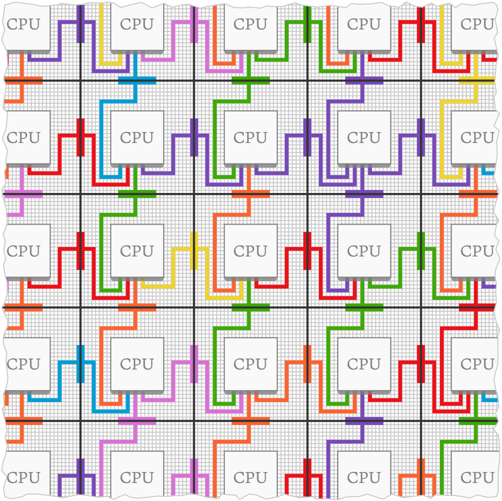


The fixpoint construction:

Define a sequence of tilesets $(T_\ell)_\ell$ as follows:

1. Encode the position of each tile in a block $[\![N_\ell]\!]^d$
2. Some border tiles contain a bit $b \in \{0,1\}$: their concatenation forms a *macro-color* $c \in \{0,1\}^*$;
3. Embed computations to "program" $T_{\ell+1}$;
   role: check that a tuple of macro-colors emulates a tile of $T_{\ell+1}$

Proof outline



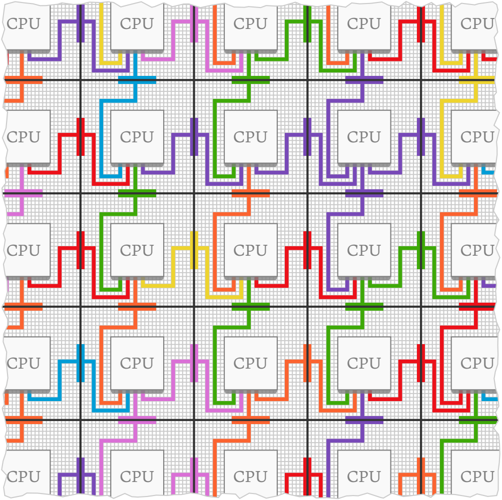
The fixpoint construction:

Define a sequence of tilesets $(T_\ell)_\ell$ as follows:

1. Encode the position of each tile in a block $[\![N_\ell]\!]^d$
2. Some border tiles contain a bit $b \in \{0,1\}$: their concatenation forms a *macro-color* $c \in \{0,1\}^*$;
3. Embed computations to "program" $T_{\ell+1}$;

   role: check that a tuple of macro-colors emulates a tile of $T_{\ell+1}$

Proof outline

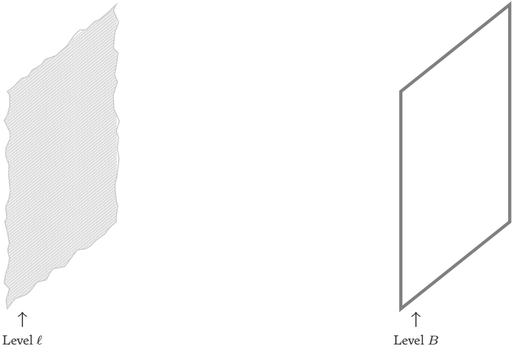


The fixpoint construction:

Define a sequence of tilesets $(T_\ell)_\ell$ as follows:

1. Encode the position of each tile in a block $[\![N_\ell]\!]^d$
2. Some border tiles contain a bit $b \in \{0,1\}$: their concatenation forms a *macro-color* $c \in \{0,1\}^*$;
3. Embed computations to "program" $T_{\ell+1}$;
   role: check that a tuple of macro-colors emulates a tile of $T_{\ell+1}$
4. The programming of $T_{\ell+1}$ is required to build T_ℓ: solve this dependency by a fixed point argument!
   🚨 This only works if computations have enough room to terminate properly...

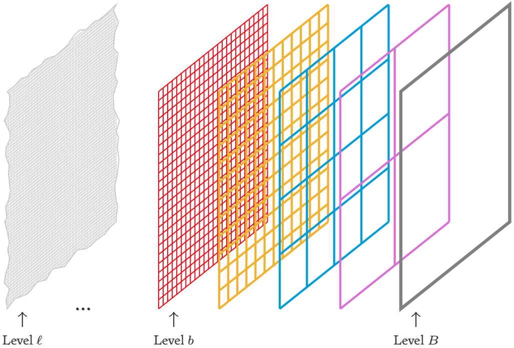
Proof outline (more details)



New simulation hierarchy:

Assume we are given some level ℓ:

Proof outline (more details)

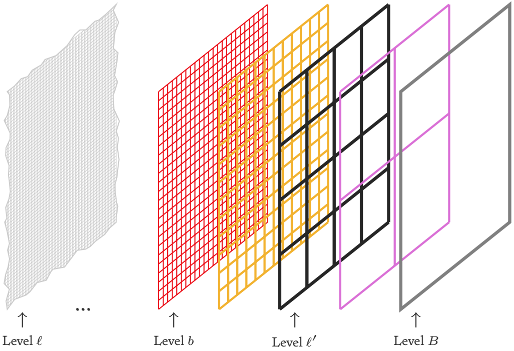


New simulation hierarchy:

Assume we are given some level ℓ:

1. Each T_ℓ-tile is involved in *many* macro-tiles (levels $b \leq j < B$ for some $b > \ell$);

Proof outline (more details)

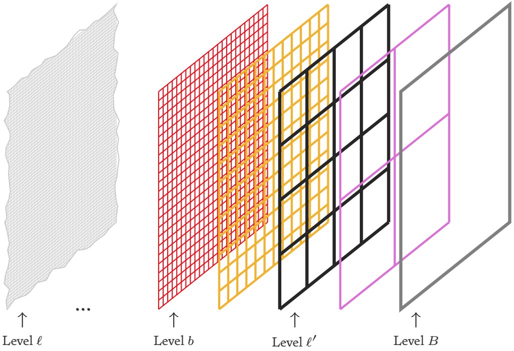


New simulation hierarchy:

Assume we are given some level ℓ:

1. Each T_ℓ-tile is involved in *many* macro-tiles (levels $b \leq j < B$ for some $b > \ell$);
2. Tiles in T_ℓ "*pick*" the next level of simulation ℓ' (such that $b \leq \ell' < B$);

Proof outline (more details)



New simulation hierarchy:

Assume we are given some level ℓ:

1. Each T_ℓ-tile is involved in *many* macro-tiles (levels $b \leq j < B$ for some $b > \ell$);
2. Tiles in T_ℓ "*pick*" the next level of simulation ℓ' (such that $b \leq \ell' < B$);
3. Macro-tiles resp. embed the simulation steps $\mathcal{S}_j$ (for $b \leq j < B$).

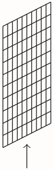
   🚨 This only works if computations have enough room to terminate properly...

Drawing computation embeddings

Question

How many computation steps can one embed in a cube $[\![N]\!]^d$?

Proposition (Cube wiring theorem)



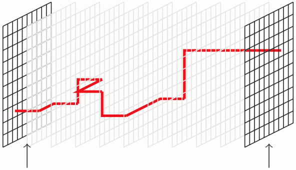
An $[\![N]\!]^{d-1}$ array of integers

Drawing computation embeddings

Question

How many computation steps can one embed in a cube $[\![N]\!]^d$?

Proposition (Cube wiring theorem)



For any $[\![N]\!]^{d-1}$ array, there exists a wiring of depth N and $O(1)$ crossings sorting the cube.

Proof sketch [Thompson-Kung, 1976]. Sort rows (alternating order) and columns $\log N$ times:

47	46	65	56	12	66	41	49	61	2
57	60	55	19	52	13	89	17	94	97
42	91	85	74	10	4	11	77	44	67
82	73	18	86	20	87	23	38	90	0
78	1	35	43	29	7	37	84	14	5
76	39	54	22	21	9	15	48	51	26
79	72	40	96	6	24	75	3	27	80
58	31	45	98	50	25	83	59	68	53
70	28	69	33	64	8	93	32	99	16
30	88	92	36	63	71	34	81	95	62

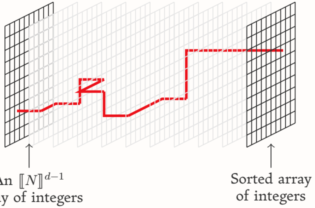
□

Drawing computation embeddings

Question

How many computation steps can one embed in a cube $[\![N]\!]^d$?

Proposition (Cube wiring theorem)



For any $[\![N]\!]^{d-1}$ array, there exists a wiring of depth N and $O(1)$ crossings sorting the cube.

Proof sketch [Thompson-Kung, 1976].
Sort rows (alternating order) and columns log N times:

47	46	65	56	12	66	41	49	61	2
57	60	55	19	52	13	89	17	94	97
42	91	85	74	10	4	11	77	44	67
82	73	18	86	20	87	23	38	90	0
78	1	35	43	29	7	37	84	14	5
76	39	54	22	21	9	15	48	51	26
79	72	40	96	6	24	75	3	27	80
58	31	45	98	50	25	83	59	68	53
70	28	69	33	64	8	93	32	99	16
30	88	92	36	63	71	34	81	95	62

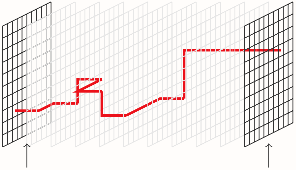
□

Drawing computation embeddings

Question

How many computation steps can one embed in a cube $[\![N]\!]^d$?

Proposition (Cube wiring theorem)



For any $[\![N]\!]^{d-1}$ array, there exists a wiring of depth N and $O(1)$ crossings sorting the cube.

Proof sketch [Thompson-Kung, 1976].
Sort rows (alternating order) and columns $\log N$ times:

2	12	41	46	47	49	56	61	65	66
57	60	55	19	52	13	89	17	94	97
42	91	85	74	10	4	11	77	44	67
82	73	18	86	20	87	23	38	90	0
78	1	35	43	29	7	37	84	14	5
76	39	54	22	21	9	15	48	51	26
79	72	40	96	6	24	75	3	27	80
58	31	45	98	50	25	83	59	68	53
70	28	69	33	64	8	93	32	99	16
30	88	92	36	63	71	34	81	95	62

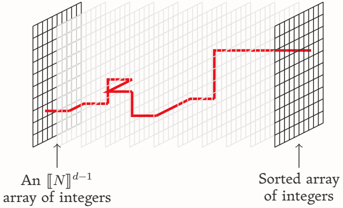
□

Drawing computation embeddings

Question

How many computation steps can one embed in a cube $[\![N]\!]^d$?

Proposition (Cube wiring theorem)



For any $[\![N]\!]^{d-1}$ array, there exists a wiring of depth N and $O(1)$ crossings sorting the cube.

Proof sketch [Thompson-Kung, 1976].
Sort rows (alternating order) and columns $\log N$ times:

2	12	41	46	47	49	56	61	65	66
57	60	55	19	52	13	89	17	94	97
42	91	85	74	10	4	11	77	44	67
82	73	18	86	20	87	23	38	90	0
78	1	35	43	29	7	37	84	14	5
76	39	54	22	21	9	15	48	51	26
79	72	40	96	6	24	75	3	27	80
58	31	45	98	50	25	83	59	68	53
70	28	69	33	64	8	93	32	99	16
30	88	92	36	63	71	34	81	95	62

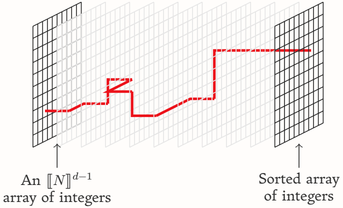
□

Drawing computation embeddings

Question

How many computation steps can one embed in a cube $[\![N]\!]^d$?

Proposition (Cube wiring theorem)



For any $[\![N]\!]^{d-1}$ array, there exists a wiring of depth N and $O(1)$ crossings sorting the cube.

Proof sketch [Thompson-Kung, 1976].
Sort rows (alternating order) and columns $\log N$ times:

2	12	41	46	47	49	56	61	65	66
97	94	89	60	57	55	52	19	17	13
42	91	85	74	10	4	11	77	44	67
82	73	18	86	20	87	23	38	90	0
78	1	35	43	29	7	37	84	14	5
76	39	54	22	21	9	15	48	51	26
79	72	40	96	6	24	75	3	27	80
58	31	45	98	50	25	83	59	68	53
70	28	69	33	64	8	93	32	99	16
30	88	92	36	63	71	34	81	95	62

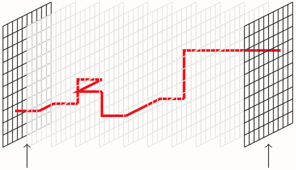
□

Drawing computation embeddings

Question

How many computation steps can one embed in a cube $[\![N]\!]^d$?

Proposition (Cube wiring theorem)



For any $[\![N]\!]^{d-1}$ array, there exists a wiring of depth N and $O(1)$ crossings sorting the cube.

Proof sketch [Thompson-Kung, 1976].
Sort rows (alternating order) and columns $\log N$ times:

2	12	41	46	47	49	56	61	65	66
97	94	89	60	57	55	52	19	17	13
4	10	11	42	44	67	74	77	85	91
90	87	86	82	73	38	23	20	18	0
1	5	7	14	29	35	37	43	78	84
76	54	51	48	39	26	22	21	15	9
3	6	24	27	40	72	75	79	80	96
98	83	68	59	58	53	50	45	31	25
8	16	28	32	33	64	69	70	93	99
95	92	88	81	71	63	62	36	34	30

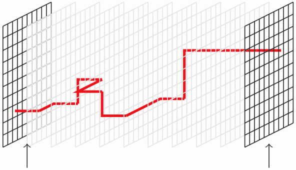
□

Drawing computation embeddings

Question

How many computation steps can one embed in a cube $[\![N]\!]^d$?

Proposition (Cube wiring theorem)



For any $[\![N]\!]^{d-1}$ array, there exists a wiring of depth N and $O(1)$ crossings sorting the cube.

Proof sketch [Thompson-Kung, 1976].
Sort rows (alternating order) and columns $\log N$ times:

2	12	41	46	47	49	56	61	65	66
97	94	89	60	57	55	52	19	17	13
4	10	11	42	44	67	74	77	85	91
90	87	86	82	73	38	23	20	18	0
1	5	7	14	29	35	37	43	78	84
76	54	51	48	39	26	22	21	15	9
3	6	24	27	40	72	75	79	80	96
98	83	68	59	58	53	50	45	31	25
8	16	28	32	33	64	69	70	93	99
95	92	88	81	71	63	62	36	34	30

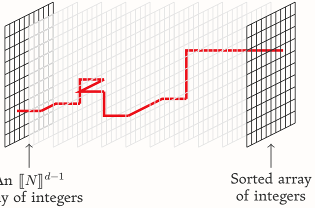
□

Drawing computation embeddings

Question

How many computation steps can one embed in a cube $[\![N]\!]^d$?

Proposition (Cube wiring theorem)



For any $[\![N]\!]^{d-1}$ array, there exists a wiring of depth N and $O(1)$ crossings sorting the cube.

Proof sketch [Thompson-Kung, 1976].
Sort rows (alternating order) and columns log N times:

1	12	41	46	47	49	56	61	65	66
2	94	89	60	57	55	52	19	17	13
3	10	11	42	44	67	74	77	85	91
4	87	86	82	73	38	23	20	18	0
8	5	7	14	29	35	37	43	78	84
76	54	51	48	39	26	22	21	15	9
90	6	24	27	40	72	75	79	80	96
95	83	68	59	58	53	50	45	31	25
97	16	28	32	33	64	69	70	93	99
98	92	88	81	71	63	62	36	34	30

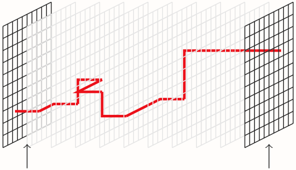
□

Drawing computation embeddings

Question

How many computation steps can one embed in a cube $[\![N]\!]^d$?

Proposition (Cube wiring theorem)



For any $[\![N]\!]^{d-1}$ array, there exists a wiring of depth N and $O(1)$ crossings sorting the cube.

Proof sketch [Thompson-Kung, 1976].
Sort rows (alternating order) and columns $\log N$ times:

1	5	7	14	29	26	22	19	15	0
2	6	11	27	33	35	23	20	17	9
3	10	24	32	39	38	37	21	18	13
4	12	28	42	40	49	50	36	31	25
8	16	41	46	44	53	52	43	34	30
76	54	51	48	47	55	56	45	65	66
90	83	68	59	57	63	62	61	78	84
95	87	86	60	58	64	69	70	80	91
97	92	88	81	71	67	74	77	85	96
98	94	89	82	73	72	75	79	93	99

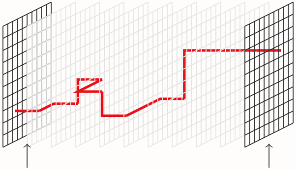
□

Drawing computation embeddings

Question

How many computation steps can one embed in a cube $[\![N]\!]^d$?

Proposition (Cube wiring theorem)



For any $[\![N]\!]^{d-1}$ array, there exists a wiring of depth N and $O(1)$ crossings sorting the cube.

Proof sketch [Thompson-Kung, 1976].
Sort rows (alternating order) and columns $\log N$ times:

0	1	5	7	14	15	19	22	26	29
35	33	27	23	20	17	11	9	6	2
3	10	13	18	21	24	32	37	38	39
50	49	42	40	36	31	28	25	12	4
8	16	30	34	41	43	44	46	52	53
76	66	65	56	55	54	51	48	47	45
57	59	61	62	63	68	78	83	84	90
95	91	87	86	80	70	69	64	60	58
67	71	74	77	81	85	88	92	96	97
99	98	94	93	89	82	79	75	73	72

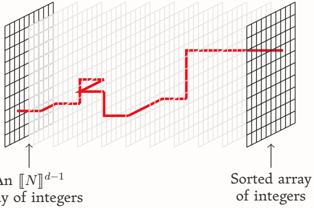
□

Drawing computation embeddings

Question

How many computation steps can one embed in a cube $[\![N]\!]^d$?

Proposition (Cube wiring theorem)



For any $[\![N]\!]^{d-1}$ array, there exists a wiring of depth N and $O(1)$ crossings sorting the cube.

Proof sketch [Thompson-Kung, 1976].
Sort rows (alternating order) and columns $\log N$ times:

0	1	5	7	14	15	11	9	6	2
3	10	13	18	20	17	19	22	12	4
8	16	27	23	21	24	28	25	26	29
35	33	30	34	36	31	32	37	38	39
50	49	42	40	41	43	44	46	47	45
57	59	61	56	55	54	51	48	52	53
67	66	65	62	63	68	69	64	60	58
76	71	74	77	80	70	78	75	73	72
95	91	87	86	81	82	79	83	84	90
99	98	94	93	89	85	88	92	96	97

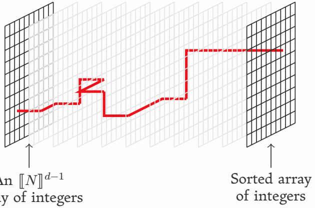
□

Drawing computation embeddings

Question

How many computation steps can one embed in a cube $[\![N]\!]^d$?

Proposition (Cube wiring theorem)



For any $[\![N]\!]^{d-1}$ array, there exists a wiring of depth N and $O(1)$ crossings sorting the cube.

Proof sketch [Thompson-Kung, 1976].
Sort rows (alternating order) and columns $\log N$ times:

0	1	2	5	6	7	9	11	14	15
22	20	19	18	17	13	12	10	4	3
8	16	21	23	24	25	26	27	28	29
39	38	37	36	35	34	33	32	31	30
40	41	42	43	44	45	46	47	49	50
61	59	57	56	55	54	53	52	51	48
58	60	62	63	64	65	66	67	68	69
80	78	77	76	75	74	73	72	71	70
79	81	82	83	84	86	87	90	91	95
99	98	97	96	94	93	92	89	88	85

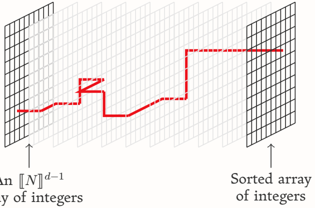
□

Drawing computation embeddings

Question

How many computation steps can one embed in a cube $[\![N]\!]^d$?

Proposition (Cube wiring theorem)



For any $[\![N]\!]^{d-1}$ array, there exists a wiring of depth N and $O(1)$ crossings sorting the cube.

Proof sketch [Thompson-Kung, 1976].
Sort rows (alternating order) and columns $\log N$ times:

0	1	2	5	6	7	9	10	4	3
8	16	19	18	17	13	12	11	14	15
22	20	21	23	24	25	26	27	28	29
39	38	37	36	35	34	33	32	31	30
40	41	42	43	44	45	46	47	49	48
58	59	57	56	55	54	53	52	51	50
61	60	62	63	64	65	66	67	68	69
79	78	77	76	75	74	73	72	71	70
80	81	82	83	84	86	87	89	88	85
99	98	97	96	94	93	92	90	91	95

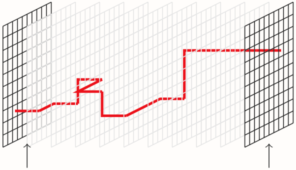
□

Drawing computation embeddings

Question

How many computation steps can one embed in a cube $[\![N]\!]^d$?

Proposition (Cube wiring theorem)



For any $[\![N]\!]^{d-1}$ array, there exists a wiring of depth N and $O(1)$ crossings sorting the cube.

Proof sketch [Thompson-Kung, 1976].
Sort rows (alternating order) and columns $\log N$ times:

0	1	2	3	4	5	6	7	9	10
19	18	17	16	15	14	13	12	11	8
20	21	22	23	24	25	26	27	28	29
39	38	37	36	35	34	33	32	31	30
40	41	42	43	44	45	46	47	48	49
59	58	57	56	55	54	53	52	51	50
60	61	62	63	64	65	66	67	68	69
79	78	77	76	75	74	73	72	71	70
80	81	82	83	84	85	86	87	88	89
99	98	97	96	95	94	93	92	91	90

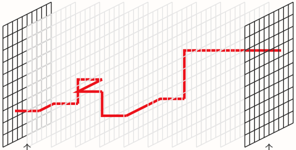
□

Drawing computation embeddings

Question

How many computation steps can one embed in a cube $[\![N]\!]^d$?

Proposition (Cube wiring theorem)



For any $[\![N]\!]^{d-1}$ array, there exists a wiring of depth N and $O(1)$ crossings sorting the cube.

Proof sketch [Thompson-Kung, 1976].
Sort rows (alternating order) and columns $\log N$ times:

0	1	2	3	4	5	6	7	9	8
19	18	17	16	15	14	13	12	11	10
20	21	22	23	24	25	26	27	28	29
39	38	37	36	35	34	33	32	31	30
40	41	42	43	44	45	46	47	48	49
59	58	57	56	55	54	53	52	51	50
60	61	62	63	64	65	66	67	68	69
79	78	77	76	75	74	73	72	71	70
80	81	82	83	84	85	86	87	88	89
99	98	97	96	95	94	93	92	91	90

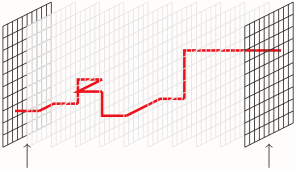
□

Drawing computation embeddings

Question

How many computation steps can one embed in a cube $[\![N]\!]^d$?

Proposition (Cube wiring theorem)



For any $[\![N]\!]^{d-1}$ array, there exists a wiring of depth N and $O(1)$ crossings sorting the cube.

Proof sketch [Thompson-Kung, 1976].
Sort rows (alternating order) and columns $\log N$ times:

0	1	2	3	4	5	6	7	8	9
19	18	17	16	15	14	13	12	11	10
20	21	22	23	24	25	26	27	28	29
39	38	37	36	35	34	33	32	31	30
40	41	42	43	44	45	46	47	48	49
59	58	57	56	55	54	53	52	51	50
60	61	62	63	64	65	66	67	68	69
79	78	77	76	75	74	73	72	71	70
80	81	82	83	84	85	86	87	88	89
99	98	97	96	95	94	93	92	91	90

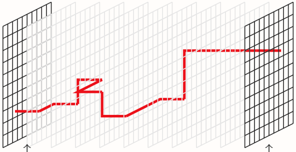
□

Drawing computation embeddings

Question

How many computation steps can one embed in a cube $[\![N]\!]^d$? $\implies$ $O(N^{d-1})$.

Proposition (Cube wiring theorem)



For any $[\![N]\!]^{d-1}$ array, there exists a wiring of depth N and $O(1)$ crossings sorting the cube.

Proof sketch [Thompson-Kung, 1976].
Sort rows (alternating order) and columns log N times:

0	1	2	3	4	5	6	7	8	9
19	18	17	16	15	14	13	12	11	10
20	21	22	23	24	25	26	27	28	29
39	38	37	36	35	34	33	32	31	30
40	41	42	43	44	45	46	47	48	49
59	58	57	56	55	54	53	52	51	50
60	61	62	63	64	65	66	67	68	69
79	78	77	76	75	74	73	72	71	70
80	81	82	83	84	85	86	87	88	89
99	98	97	96	95	94	93	92	91	90

□

RAM machines

- 32 variables $\mathtt{var}_0$, $\mathtt{var}_1$...
- An infinite memory $(\mathtt{M}[i])_{i\in\mathbb{N}}$.

Instructions: arithmetic operations on variables, and memory I/O.

Proof (folklore).
Given a trace of computation in chronological order,

$$\ldots(\mathtt{address}=6, \mathtt{time}=15, \mathtt{READ}, m)\,;\ (\mathtt{address}=42, \mathtt{time}=16, \mathtt{READ}, n)\ldots$$

check its correctness by sorting it lexicographically:

$$\ldots(\mathtt{address}=42, \mathtt{time}=3, \mathtt{WRITE}, n)\,;\ (\mathtt{address}=42, \mathtt{time}=16, \mathtt{READ}, n)\ldots$$ □

Conclusion

Sofic shifts generalize regular languages to infinite and higher-dimensional words.

In this talk

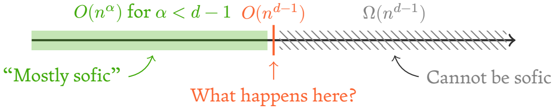
Proving soficity by quantifying the information in infinite sequences of substitutions.

Remarks:

- Block sizes anywhere in the interval $[\![2, 2^{O(L_\ell^\delta)}]\!]$;
- Generalizes to rectangular substitutions;
- Generalizes to non-uniform substitutions.

Questions:

- How much computations can be embedded in a rectangle $[\![n_1, \dots, n_d]\!]$?
- Generalization to other geometries (*i.e.* Cayley graphs of groups);
- Formalization of the $O(n^{d-1})$ argument.




That's all Folks!