
Pimp
My
~~Ride~~
Fixpoint
SEASON #04

Pimp my fixpoint

S-adic shifts in the self-simulation framework

Antonin Callard, with Léo Paviet Salomon, Pascal Vanier
and our special thanks to Pierre Guillon

Journées SDA2, June 2026

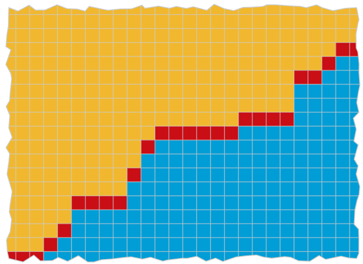
Welcome to shiftland

Definition (Shift spaces)

On a finite alphabet $\mathcal{A}$, a *shift space* is a set of configurations $X \subseteq \mathcal{A}^{\mathbb{Z}^d}$

- (**Dynamics**) that is closed and translation-invariant;
- (**Combinatorics**) that avoid the patterns from a forbidden family $\mathcal{F}$.

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

$X_{\mathcal{F}} = \{$  $, \ldots \}$

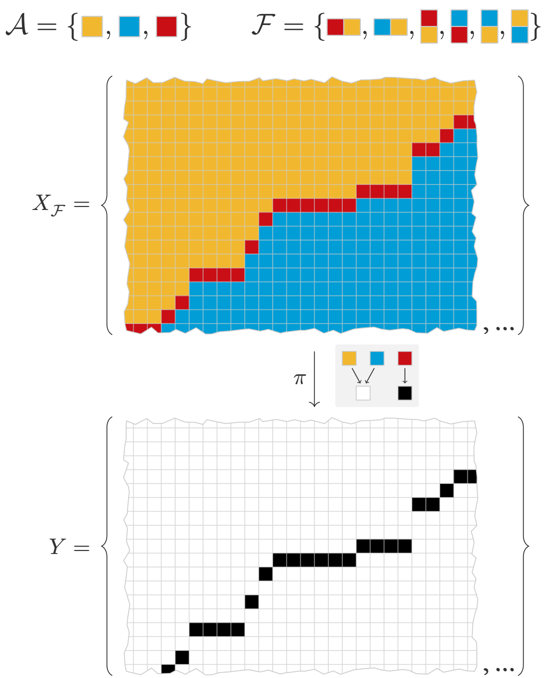
Shifts spaces are classified by their **presentations** $\mathcal{F}$:

- *Finite* $\implies$ *finite* family of forbidden patterns;
- *Effective* $\implies$ *comp. enumerable* forbidden patterns;

Definition (Shift spaces)

On a finite alphabet $\mathcal{A}$, a *shift space* is a set of configurations $X \subseteq \mathcal{A}^{\mathbb{Z}^d}$

- (**Dynamics**) that is closed and translation-invariant;
- (**Combinatorics**) that avoid the patterns from a forbidden family $\mathcal{F}$.



Shifts spaces are classified by their **presentations** $\mathcal{F}$:

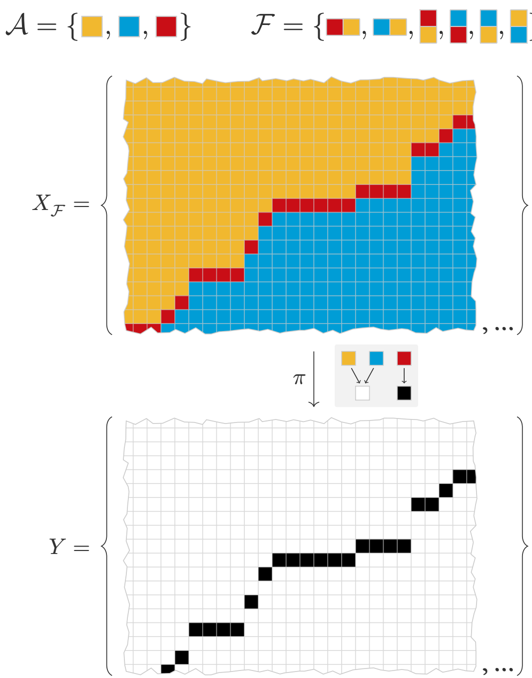
- *Finite* $\implies$ *finite* family of forbidden patterns;
- *Effective* $\implies$ *comp. enumerable* forbidden patterns;
- ***Sofic*** $\implies$ factor of a finite type shift;

$$\{\text{Finite type shifts}\} \subsetneq \{\text{Sofic shifts}\} \subsetneq \{\text{Effective shifts}\}$$

Definition (Shift spaces)

On a finite alphabet $\mathcal{A}$, a *shift space* is a set of configurations $X \subseteq \mathcal{A}^{\mathbb{Z}^d}$

- (**Dynamics**) that is closed and translation-invariant;
- (**Combinatorics**) that avoid the patterns from a forbidden family $\mathcal{F}$.



Shifts spaces are classified by their **presentations** $\mathcal{F}$:

- *Finite* $\implies$ *finite* family of forbidden patterns;
- *Effective* $\implies$ *comp. enumerable* forbidden patterns;
- ***Sofic*** $\implies$ factor of a finite type shift;

{Finite type shifts} $\subsetneq$ {Sofic shifts} $\subsetneq$ {Effective shifts}

⚠ In this talk, $\mathcal{F}$ is often implicit. If in doubt, ask!

Definition (Shift spaces)

On a finite alphabet $\mathcal{A}$, a *shift space* is a set of configurations $X \subseteq \mathcal{A}^{\mathbb{Z}^d}$

- (**Dynamics**) that is closed and translation-invariant;
- (**Combinatorics**) that avoid the patterns from a forbidden family $\mathcal{F}$.

- On $\mathbb{Z}$, sofic shifts essentially are *regular languages* of infinite words:

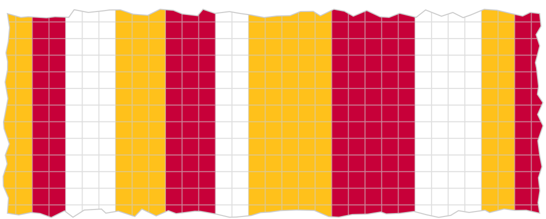
Lemma ("Myhill-Nerode, 1957")

A shift space $X \subseteq \mathcal{A}^{\mathbb{Z}}$ is sofic if and only if its factors define a regular language.

- On $\mathbb{Z}^d$ ($d \geq 2$), sofic shifts become wilder:

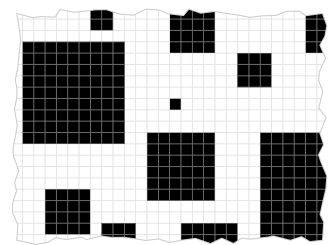
Example (Hochman, ..., '2009+)

Extensions of effective $\mathbb{Z}$ shifts:



Example (Westrick '2016)

Seas of squares of prime (resp....) sizes:



Motivation

How can I determine whether my favorite multidimensional shift is sofic?

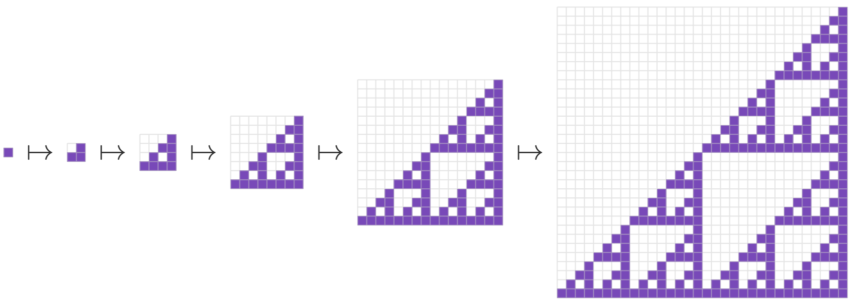
Definition (Shift spaces)

On a finite alphabet $\mathcal{A}$, a *shift space* is a set of configurations $X \subseteq \mathcal{A}^{\mathbb{Z}^d}$

- (**Dynamics**) that is closed and translation-invariant;
- (**Combinatorics**) that avoid the patterns from a forbidden family $\mathcal{F}$.

Substitution:

τ : 



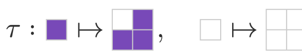
Two suspects: S-adic shifts and self-simulation

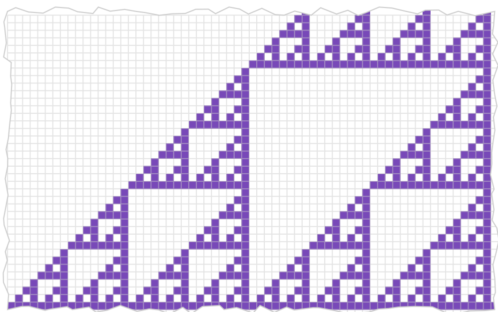
Definition (Shift spaces)

On a finite alphabet $\mathcal{A}$, a *shift space* is a set of configurations $X \subseteq \mathcal{A}^{\mathbb{Z}^d}$

- (**Dynamics**) that is closed and translation-invariant;
- (**Combinatorics**) that avoid the patterns from a forbidden family $\mathcal{F}$.

Substitution:





$$x = x_0 \xleftarrow{\tau} x_1 \xleftarrow{\tau} x_2 \xleftarrow{\tau} \dots$$

Lemma (Mozes, '1989)

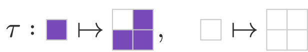
Substitutive shifts on $\mathbb{Z}^d$ ($d \geq 2$) are sofic.

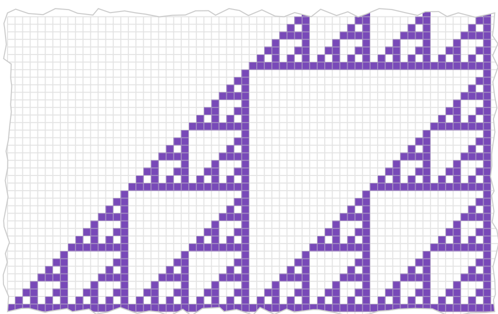
Definition (Shift spaces)

On a finite alphabet $\mathcal{A}$, a *shift space* is a set of configurations $X \subseteq \mathcal{A}^{\mathbb{Z}^d}$

- (**Dynamics**) that is closed and translation-invariant;
- (**Combinatorics**) that avoid the patterns from a forbidden family $\mathcal{F}$.

Substitution:





$$x = x_0 \overset{\tau_1}{\longleftarrow} x_1 \overset{\tau_2}{\longleftarrow} x_2 \overset{\tau_3}{\longleftarrow} \dots$$

Definition (S-adic shifts)

An S-adic shift is the set of limit configurations associated with a sequence of substitutions $\tau_{\ell+1} : \mathcal{A}_{\ell+1} \to \mathcal{A}_{\ell}^{[\![N_{\ell+1}]\!]^d}$.

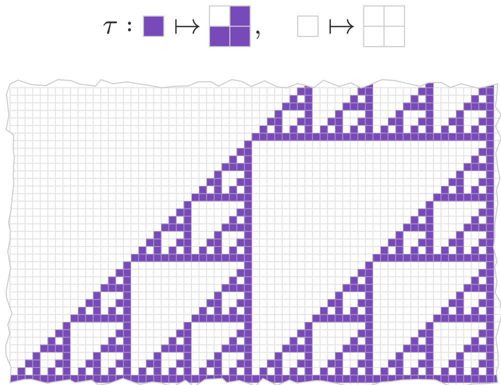
Two suspects: S-adic shifts and self-simulation

Definition (Shift spaces)

On a finite alphabet $\mathcal{A}$, a *shift space* is a set of configurations $X \subseteq \mathcal{A}^{\mathbb{Z}^d}$

- (**Dynamics**) that is closed and translation-invariant;
- (**Combinatorics**) that avoid the patterns from a forbidden family $\mathcal{F}$.

Substitution:

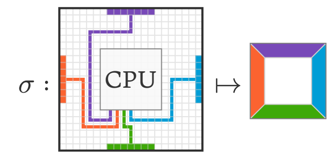


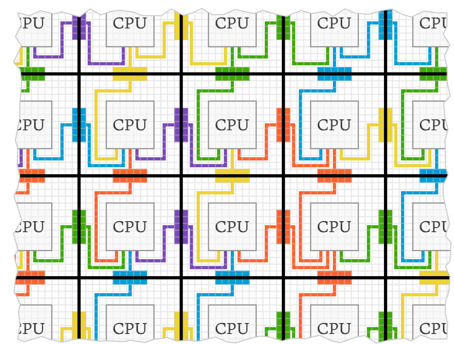
$$x = x_0 \xleftarrow{\tau_1} x_1 \xleftarrow{\tau_2} x_2 \xleftarrow{\tau_3} \dots$$

Definition (S-adic shifts)

An S-adic shift is the set of limit configurations associated with a sequence of substitutions $\tau_{\ell+1} : \mathcal{A}_{\ell+1} \to \mathcal{A}_\ell^{[\![N_{\ell+1}]\!]^d}$.

Simulation:





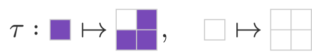
- Build computationally complex SFTs/sofic shifts [Durand, Romashchenko & Shen '2010+, Westrick '2017, Destombes '2021...]

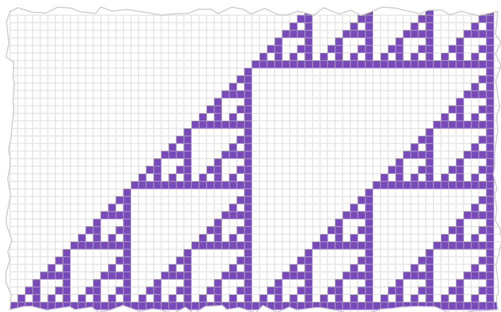
Remark

A simulation looks (a lot) like the inverse of a substitution...

Is there a connection?

Substitution:



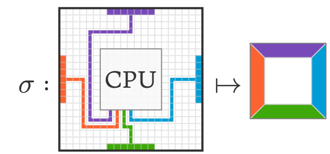


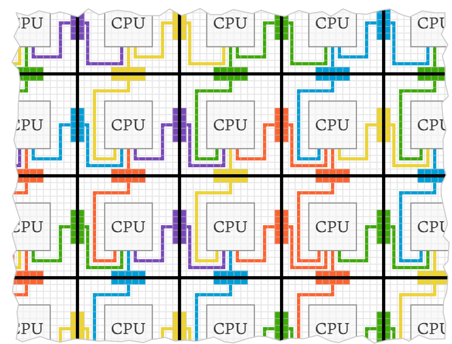
$$x = x_0 \overset{\tau_1}{\longleftarrow} x_1 \overset{\tau_2}{\longleftarrow} x_2 \overset{\tau_3}{\longleftarrow} \dots$$

Definition (S-adic shifts)

An S-adic shift is the set of limit configurations associated with a sequence of substitutions $\tau_{\ell+1} : \mathcal{A}_{\ell+1} \to \mathcal{A}_\ell^{[\![N_{\ell+1}]\!]^d}$.

Simulation:

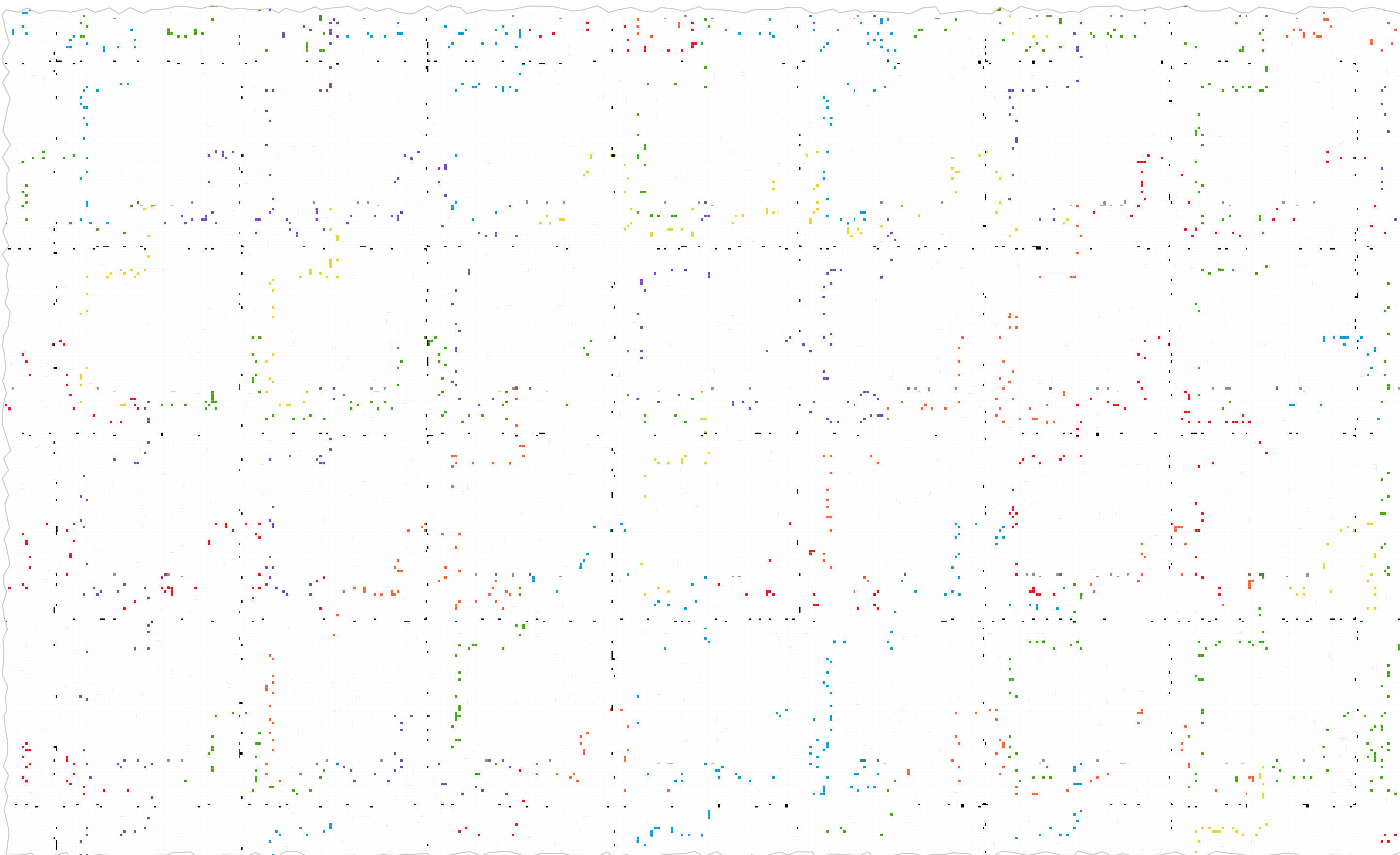




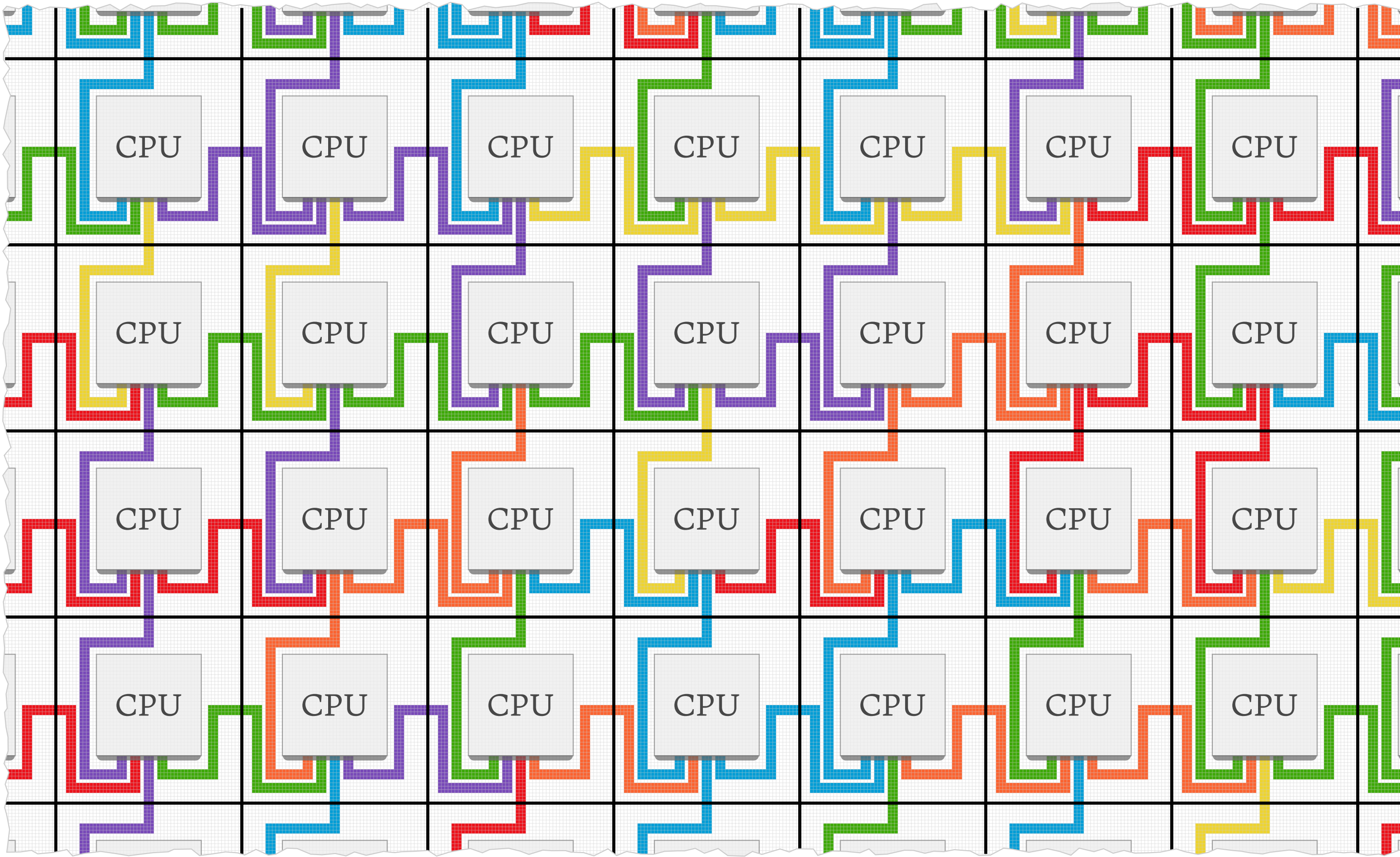
- Build computationally complex SFTs/sofic shifts [Durand, Romashchenko & Shen '2010+, Westrick '2017, Destombes '2021...]

The fixed point construction

Level $\ell = 0$

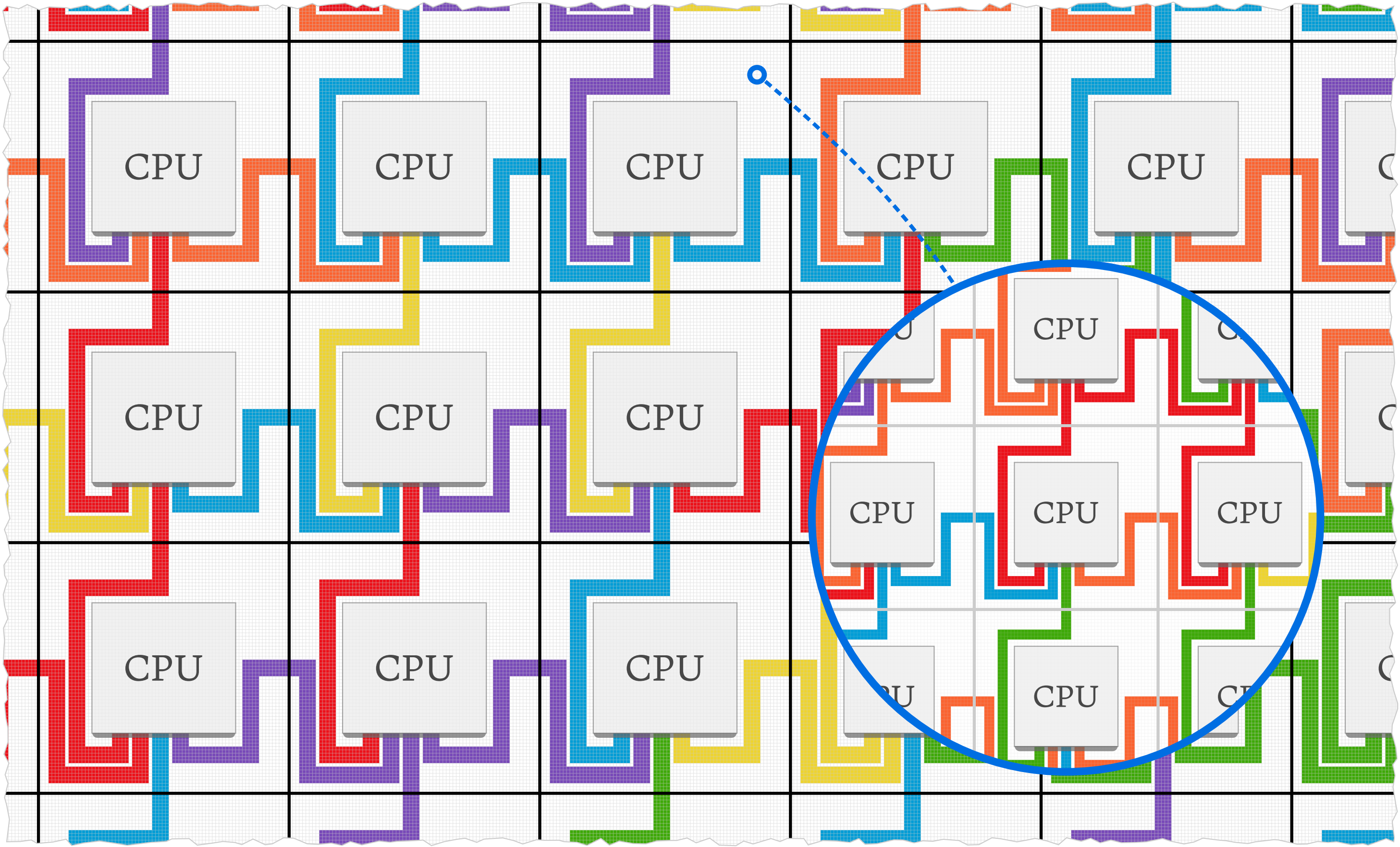


Level $\ell = 0$


CPU
CPU
CPU
CPU
CPU
CPU
CPU
CPU
CPU
CPU
CPU
CPU
CPU
CPU
CPU
CPU
CPU
CPU
CPU
CPU
CPU
CPU
CPU
CPU
CPU
CPU
CPU
CPU

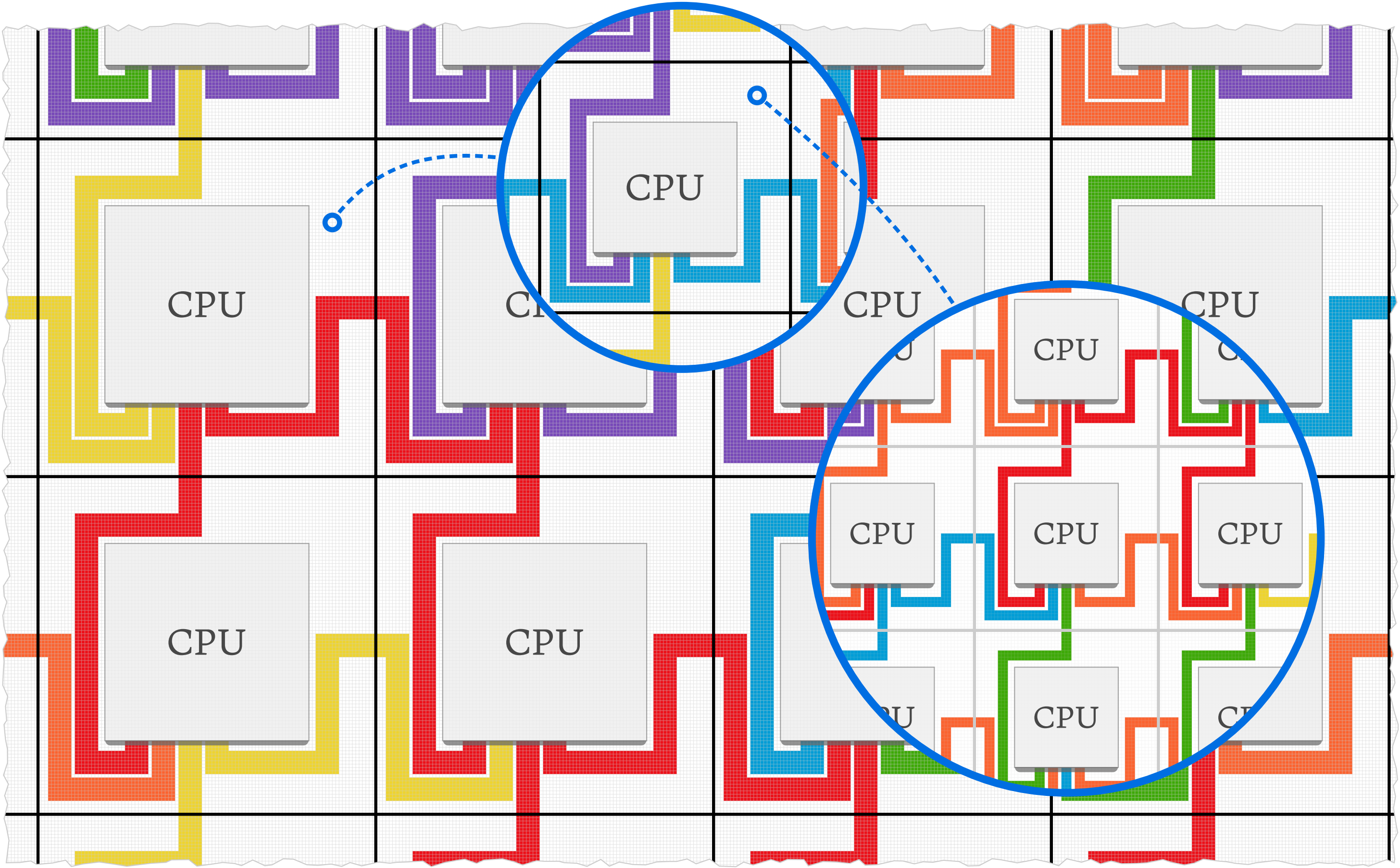
S-adicity of the fixed point construction

Level $\ell = 1$ (*i.e.* zoom $\times$(-55))



S-adicity of the fixed point construction

Level $\ell = 2$ $\quad$ (*i.e.* zoom $\times$(-5555))

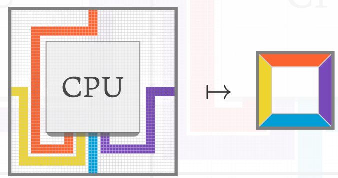


S-adicity of the fixed point construction

Level $\ell = 2$ ($i.e.$ zoom $\times$(-5555))

The tiling of level ℓ simulates (by some $\sigma_\ell : \mathcal{A}_\ell^{[\![N_\ell]\!]^d} \to \mathcal{A}_{\ell+1}$) the tiling of level $\ell + 1$.

σ_ℓ :



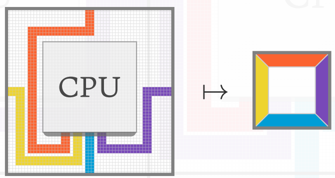
Conclusion

Self-simulating tiling $\Longrightarrow$ S-adic configuration

S-adicity of the fixed point construction

Level $\ell = 2$ (*i.e.* zoom $\times$(-5555))

The tiling of level ℓ simulates (by some $\sigma_\ell : \mathcal{A}_\ell^{[\![N_\ell]\!]^d} \to \mathcal{A}_{\ell+1}$) the tiling of level $\ell + 1$.

$\sigma_\ell :$ 

Main question

Self-simulating tiling $\overset{?}{\Longleftarrow}$ S-adic configuration

Fix substitutions $(\tau_\ell)_{\ell\geq 1}$ with $\tau_{\ell+1} : \mathcal{A}_{\ell+1} \mapsto \mathcal{A}_\ell^{[\![N_{\ell+1}]\!]^d}$.

Simulation

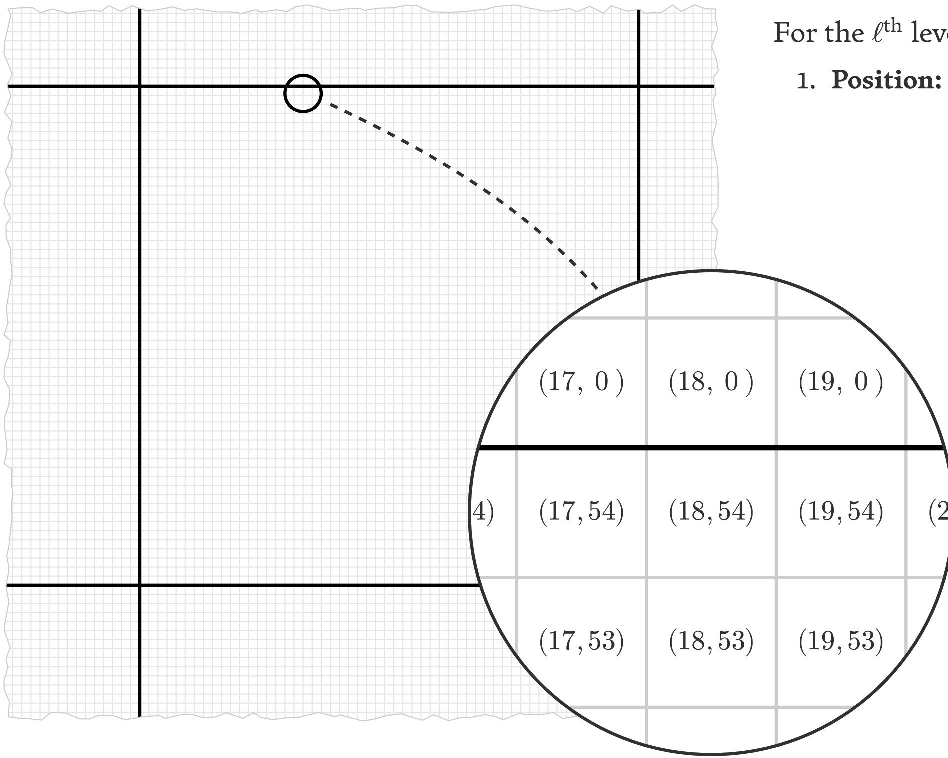
Realize the corresponding S-adic configurations as self-simulating tilings.

For the ℓ^{th} level, define tiles T_ℓ drawing $[\![N_{\ell+1}]\!]^d$ *macro-tiles*:

Fix substitutions $(\tau_\ell)_{\ell\geq 1}$ with $\tau_{\ell+1} : \mathcal{A}_{\ell+1} \mapsto \mathcal{A}_\ell^{[\![N_{\ell+1}]\!]^d}$.

Simulation

Realize the corresponding S-adic configurations as self-simulating tilings.



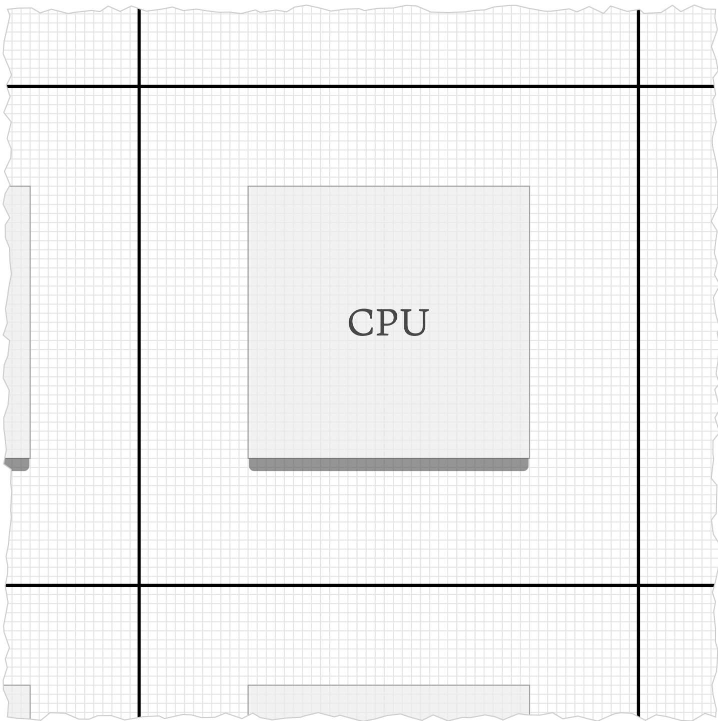
For the ℓ^{th} level, define tiles T_ℓ drawing $[\![N_{\ell+1}]\!]^d$ *macro-tiles*:

1. **Position:** Draw the position of each tile in its block $[\![N_{\ell+1}]\!]^d$;

Fix substitutions $(\tau_\ell)_{\ell\geq 1}$ with $\tau_{\ell+1} : \mathcal{A}_{\ell+1} \mapsto \mathcal{A}_\ell^{[\![N_{\ell+1}]\!]^d}$.

Simulation

Realize the corresponding S-adic configurations as self-simulating tilings.



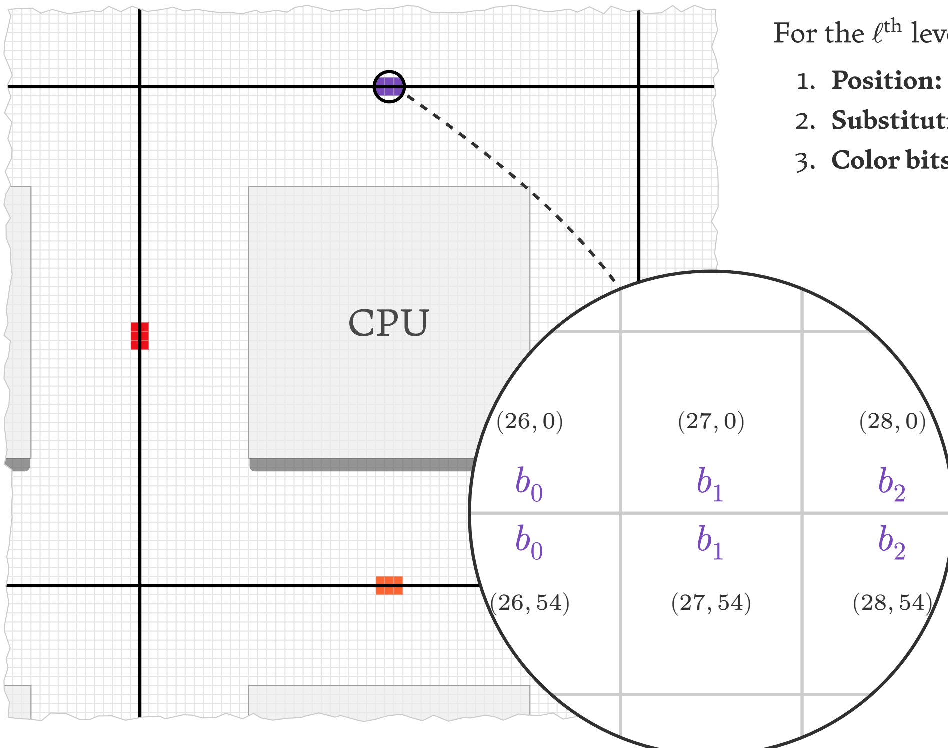
For the ℓ^{th} level, define tiles T_ℓ drawing $[\![N_{\ell+1}]\!]^d$ *macro-tiles*:

1. **Position:** Draw the position of each tile in its block $[\![N_{\ell+1}]\!]^d$;
2. **Substitution:** Embed a computation of $\tau_{\ell+1}$;

Fix substitutions $(\tau_\ell)_{\ell\geq 1}$ with $\tau_{\ell+1} : \mathcal{A}_{\ell+1} \mapsto \mathcal{A}_\ell^{[\![N_{\ell+1}]\!]^d}$.

Simulation

Realize the corresponding S-adic configurations as self-simulating tilings.



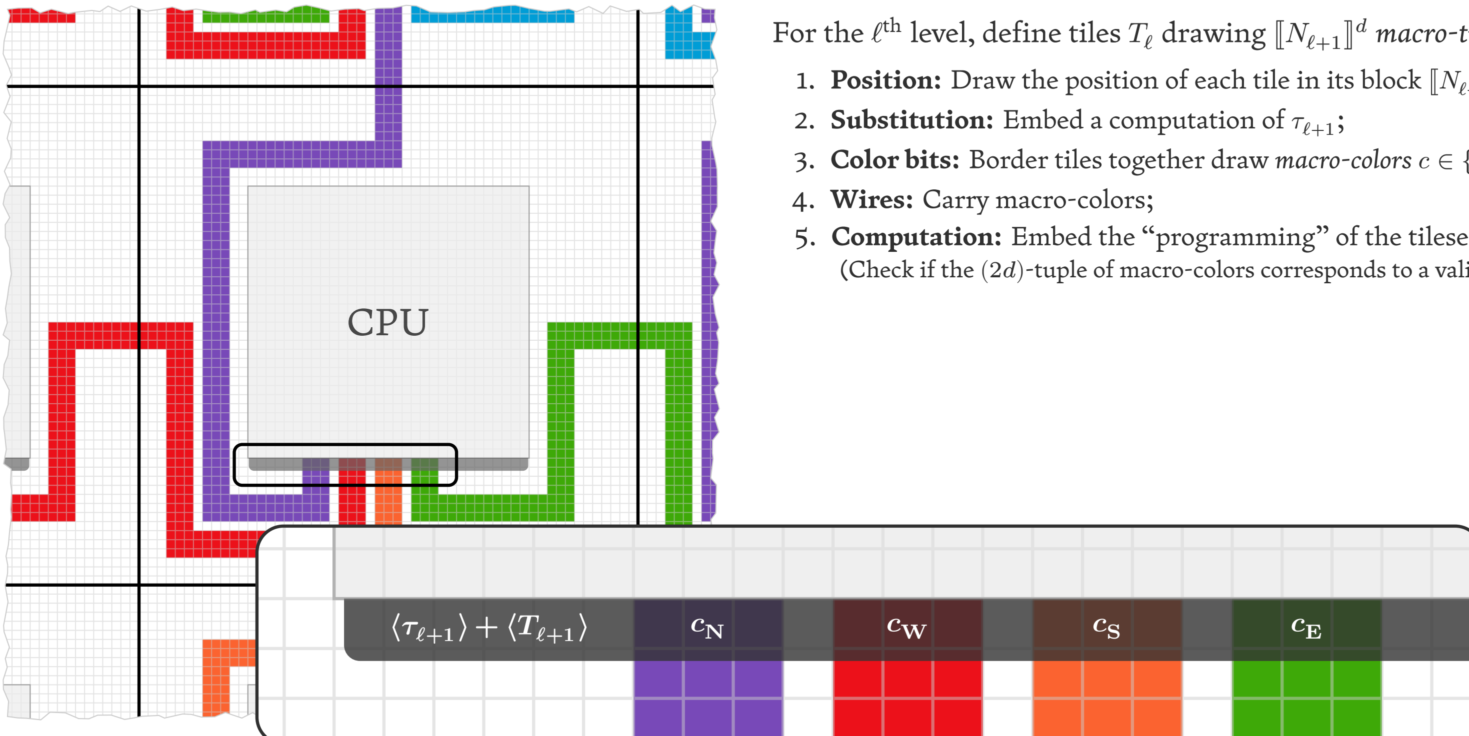
For the ℓ^{th} level, define tiles T_ℓ drawing $[\![N_{\ell+1}]\!]^d$ *macro-tiles*:

1. **Position:** Draw the position of each tile in its block $[\![N_{\ell+1}]\!]^d$;
2. **Substitution:** Embed a computation of $\tau_{\ell+1}$;
3. **Color bits:** Border tiles together draw *macro-colors* $c \in \{0,1\}^*$;

Fix substitutions $(\tau_\ell)_{\ell\geq 1}$ with $\tau_{\ell+1} : \mathcal{A}_{\ell+1} \mapsto \mathcal{A}_\ell^{[\![N_{\ell+1}]\!]^d}$.

Simulation

Realize the corresponding S-adic configurations as self-simulating tilings.



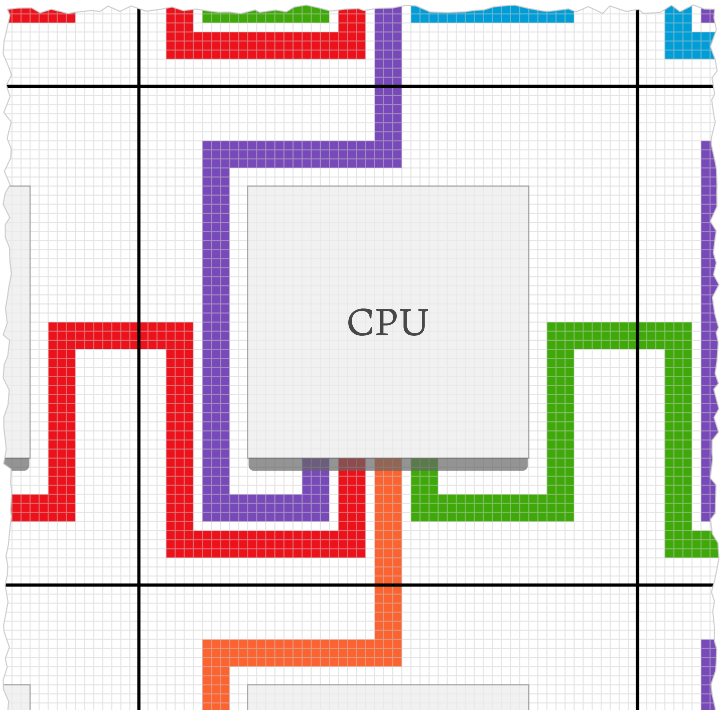
For the ℓ^{th} level, define tiles T_ℓ drawing $[\![N_{\ell+1}]\!]^d$ *macro-tiles*:

1. **Position:** Draw the position of each tile in its block $[\![N_{\ell+1}]\!]^d$;
2. **Substitution:** Embed a computation of $\tau_{\ell+1}$;
3. **Color bits:** Border tiles together draw *macro-colors* $c \in \{0,1\}^*$;
4. **Wires:** Carry macro-colors;
5. **Computation:** Embed the "programming" of the tileset $T_{\ell+1}$;
   (Check if the $(2d)$-tuple of macro-colors corresponds to a valid tile)

Fix substitutions $(\tau_\ell)_{\ell\geq 1}$ with $\tau_{\ell+1} : \mathcal{A}_{\ell+1} \mapsto \mathcal{A}_\ell^{[\![N_{\ell+1}]\!]^d}$.

Simulation

Realize the corresponding S-adic configurations as self-simulating tilings.



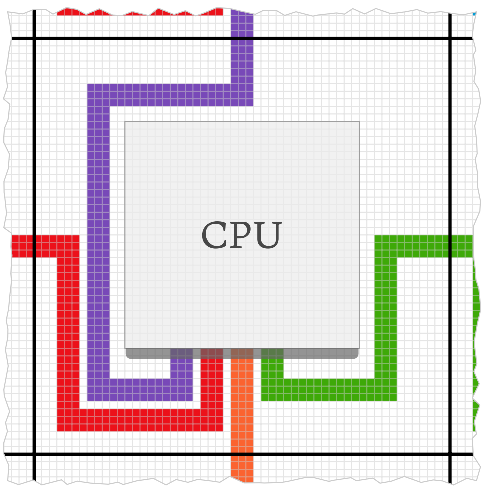
For the ℓ^{th} level, define tiles T_ℓ drawing $[\![N_{\ell+1}]\!]^d$ *macro-tiles*:

1. **Position:** Draw the position of each tile in its block $[\![N_{\ell+1}]\!]^d$;
2. **Substitution:** Embed a computation of $\tau_{\ell+1}$;
3. **Color bits:** Border tiles together draw *macro-colors* $c \in \{0,1\}^*$;
4. **Wires:** Carry macro-colors;
5. **Computation:** Embed the "programming" of the tileset $T_{\ell+1}$;
   (Check if the $(2d)$-tuple of macro-colors corresponds to a valid tile)

To simulate the next level $\ell + 1$:

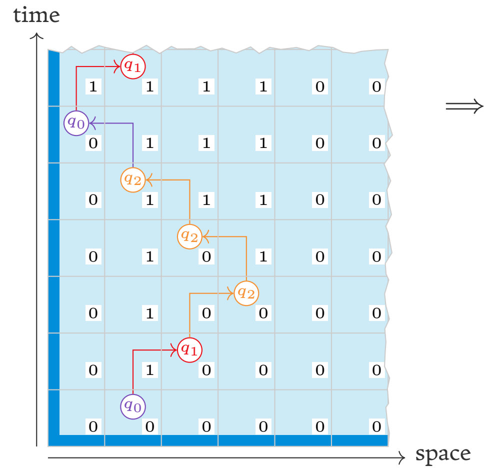
- ⚠ **Problem:** we need the program for $T_{\ell+1}$ to define T_ℓ;
- We solve this dependency using a *computational* fixed point.

Tilings of T_0 inductively simulate valid tilings of T_1, T_2...



1. Embedding of computations:

How much computation can we embed in a d-dimensional cube $[\![N]\!]^d$?



$\Longrightarrow$

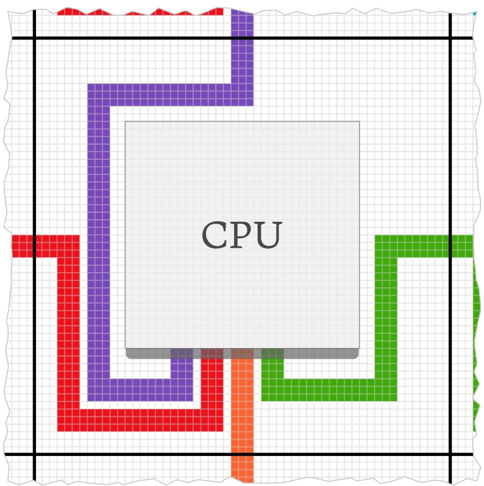
```
Time t = 0: pos 1, state q0, symbol 0
Time t = 1: pos 2, state q1, symbol 0
Time t = 2: pos 3, state q2, symbol 0
Time t = 3: pos 2, state q2, symbol 0
Time t = 4: pos 1, state q2, symbol 1
                etc...
```

$\Downarrow$

```
Time t = 0: pos 1, state q0, symbol 0
Time t = 4: pos 1, state q2, symbol 1
Time t = 1: pos 2, state q1, symbol 0
Time t = 3: pos 2, state q2, symbol 0
Time t = 2: pos 3, state q2, symbol 0
                etc...
```

$\Longrightarrow$ Use traces instead of space-time diagrams.

At least N^{d-1} steps (size of $(d-1)$-facet).



1. **Embedding of computations:**

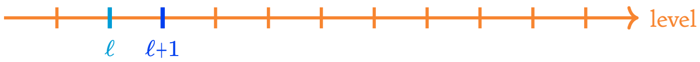
How much computation can we embed in a d-dimensional cube $[\![N]\!]^d$?

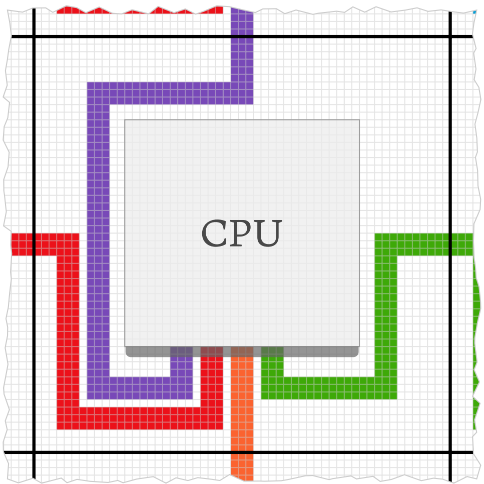
> At least N^{d-1} steps (size of $(d-1)$-facet).

2. **Sizes of substitutions:**

What sizes of substitutions can we accommodate?

- Large sizes: $N_{\ell+2}$ must fit in the macro-colors (*i.e.* $\log N_{\ell+2} \ll N_{\ell+1}^{d-1}$);
- Small sizes: $N_{\ell+1}$ must be large enough to embed all valid computations.





1. **Embedding of computations:**

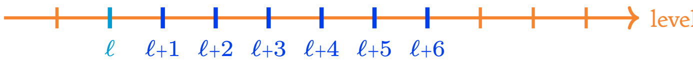
   How much computation can we embed in a d-dimensional cube $[\![N]\!]^d$?

   At least N^{d-1} steps (size of $(d-1)$-facet).

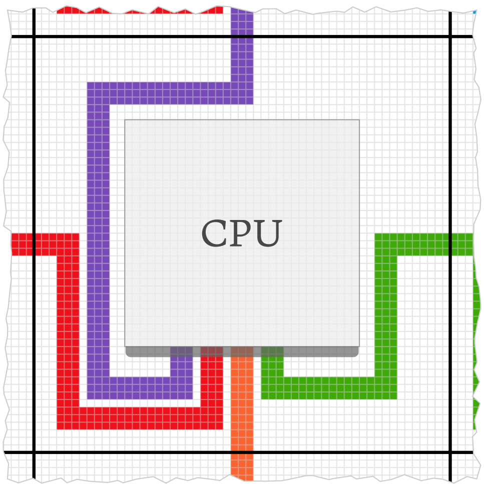
2. **Sizes of substitutions:**

   What sizes of substitutions can we accommodate?

   - Large sizes: $N_{\ell+2}$ must fit in the macro-colors (*i.e.* $\log N_{\ell+2} \ll N_{\ell+1}^{d-1}$);
   - Small sizes: $N_{\ell+1}$ must be large enough to embed all valid computations.



$\Longrightarrow$ Simulations by blocks to allow $N_\ell = 2$;



1. **Embedding of computations:**

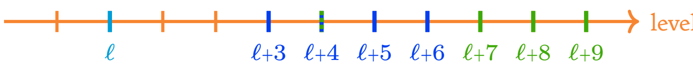
   How much computation can we embed in a d-dimensional cube $[\![N]\!]^d$*?*

   At least N^{d-1} steps (size of $(d-1)$-facet).

2. **Sizes of substitutions:**

   What sizes of substitutions can we accommodate?

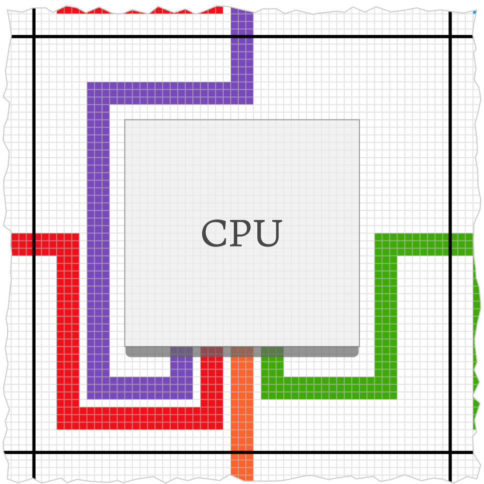
   - Large sizes: $N_{\ell+2}$ must fit in the macro-colors (*i.e.* $\log N_{\ell+2} \ll N_{\ell+1}^{d-1}$);
   - Small sizes: $N_{\ell+1}$ must be large enough to embed all valid computations.



$\Longrightarrow$ Simulations by blocks to allow $N_\ell = 2$;

$\Longrightarrow$ Alternative hierarchy for better numeric bounds.

$$2 \leq N_{\ell+1} \ll 2^{N_\ell^{d-1}}$$



1. **Embedding of computations:**

How much computation can we embed in a d-dimensional cube $[\![N]\!]^d$?

At least N^{d-1} steps (size of $(d-1)$-facet).

2. **Sizes of substitutions:**

What sizes of substitutions can we accommodate?

$$2 \leq N_{\ell+1} \ll 2^{N_\ell^{d-1}}$$

3. **Tiling conditions:**

Macro-tiles can exchange information with their neighbors through their colors.

$$x = \underset{\in X_0}{x^{(0)}} \overset{\tau_1}{\longleftarrow} \underset{\in X_1}{x^{(1)}} \overset{\tau_2}{\longleftarrow} \underset{\in X_2}{x^{(2)}} \overset{\tau_3}{\longleftarrow} \underset{\in X_3}{x^{(3)}} \ \dots$$

Substitutions with SFT “filters”.

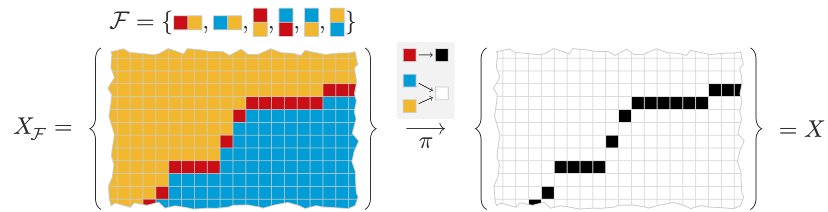
Realization of D-adic shifts

Definition (Dill map)

A *dill map* is the composition of a block map and a substitution.

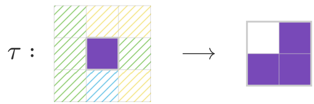
Definition (Sofic shift)

A shift $X \subseteq \mathcal{A}^{\mathbb{Z}^d}$ is *sofic* if it is the factor of an SFT.



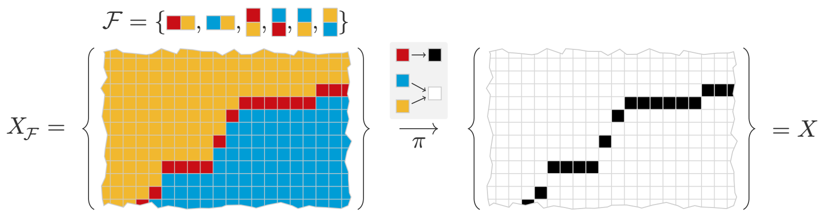
Definition (Dill map)

A *dill map* is the composition of a block map and a substitution.



Definition (Sofic shift)

A shift $X \subseteq \mathcal{A}^{\mathbb{Z}^d}$ is *sofic* if it is the factor of an SFT.

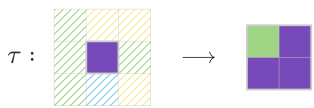


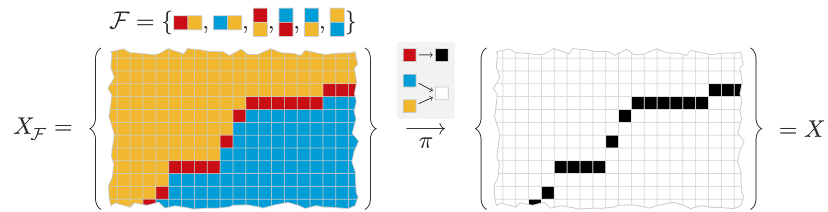
Definition (Dill map)

A *dill map* is the composition of a block map and a substitution.

Definition (Sofic shift)

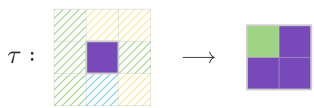
A shift $X \subseteq \mathcal{A}^{\mathbb{Z}^d}$ is *sofic* if it is the factor of an SFT.





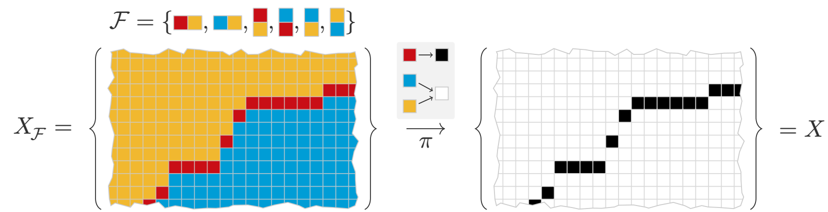
Definition (Dill map)

A *dill map* is the composition of a block map and a substitution.



Definition (Sofic shift)

A shift $X \subseteq \mathcal{A}^{\mathbb{Z}^d}$ is *sofic* if it is the factor of an SFT.

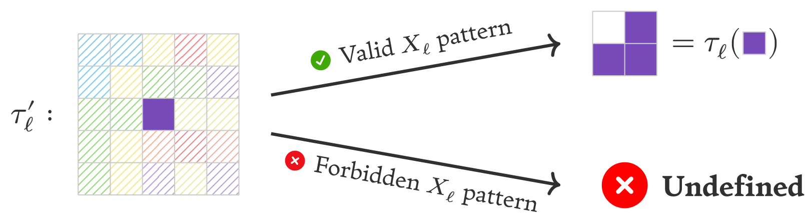


Substitutions + SFTs:

$$x = \underset{\in X_0}{x^{(0)}} \xleftarrow{\tau_1} \underset{\in X_1}{x^{(1)}} \xleftarrow{\tau_2} \underset{\in X_2}{x^{(2)}} \xleftarrow{\tau_3} \underset{\in X_3}{x^{(3)}} \ \dots$$

Dill maps:

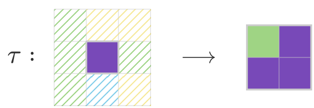
$$x = x^{(0)} \xleftarrow{\tau'_1} x^{(1)} \xleftarrow{\tau'_2} x^{(2)} \xleftarrow{\tau'_3} x^{(3)} \ \dots$$



3 – Realization of D-adic shifts — The statement

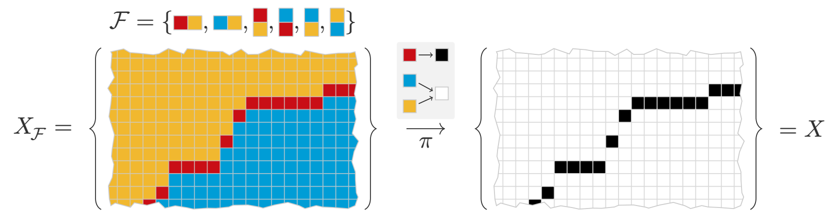
Definition (Dill map)

A *dill map* is the composition of a block map and a substitution.



Definition (Sofic shift)

A shift $X \subseteq \mathcal{A}^{\mathbb{Z}^d}$ is *sofic* if it is the factor of an SFT.



Theorem (C., Paviet Salomon and Vanier, '2026+)

Let $X \subseteq \mathcal{A}^{\mathbb{Z}^d}$ be a sofic shift, and $(\tau_\ell)_{\ell \geq 1}$ a computable sequence of dill maps $\tau_{\ell+1} : \mathcal{A}_{\ell+1}^{V_\ell} \rightrightarrows \mathcal{A}_\ell^{[\![N_{\ell+1}]\!]^d}$ and $\alpha < d - 1$. Denote $L_\ell = \prod_{k \leq \ell} N_k$. If:

1. **Growth:** $2 \leq N_{\ell+1} \leq 2^{L_\ell^\alpha}$;
2. **Alphabet:** $\log |\mathcal{A}_{\ell+1}| = \mathcal{O}(L_\ell^\alpha)$;
3. **Computability:** $\tau_{\ell+1}$ is computable in time $\mathcal{O}(L_\ell^\alpha)$;

then the resulting *D-adic limit space* is a sofic shift:

$$\overleftarrow{X} = \{x \in X : \exists (x_\ell)_{\ell \in \mathbb{N}},\ x = x_0 \overset{\tau_1}{\longleftarrow} x_1 \overset{\tau_2}{\longleftarrow} x_2 \ldots\}.$$

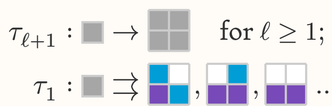
! Non-deterministic dill maps

✓ Rectangular dill maps ✓ Non-uniform dill maps

Claim

All full shifts $Y = \mathcal{A}^{\mathbb{Z}^d}$ are sofic.

Substitution



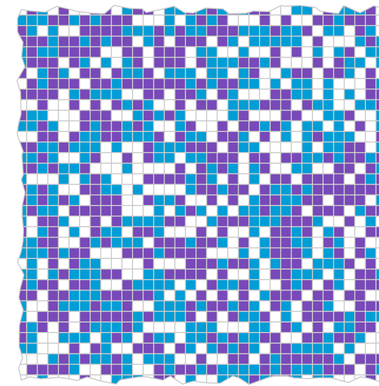
Level $\ell = 1$:

- Non-deterministically substitute to any 2×2 pattern;

Conditions

✓ **Growth:** $N_{\ell+1} = 2$; ✓ **Alphabet:** $\log |\mathcal{A}_{\ell+1}| = \mathcal{O}(1)$;

✓ **Computability:** $\tau_{\ell+1}$ is computable in time $\mathcal{O}(1)$;

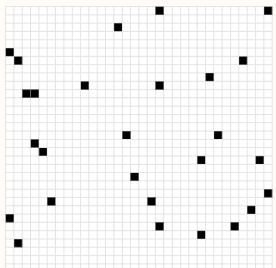
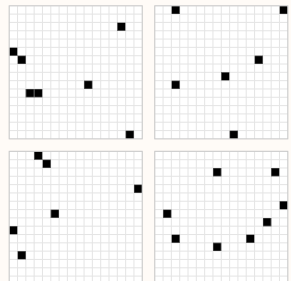


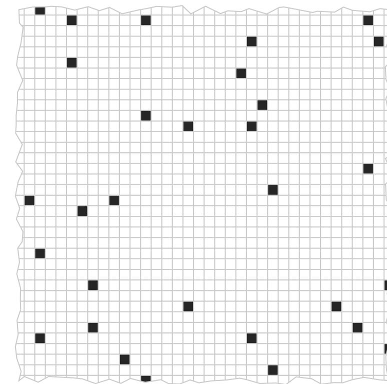
A full shift $\mathcal{A}^{\mathbb{Z}^d}$: all configurations are valid.

Claim

All effective shifts $Y \subseteq \{■, □\}^{\mathbb{Z}^d}$ of density $\mathcal{O}(n^\alpha)$ $(\alpha < d-1)$ are sofic.

Substitution

$\tau_{\ell+1}$:  $\mapsto$ 



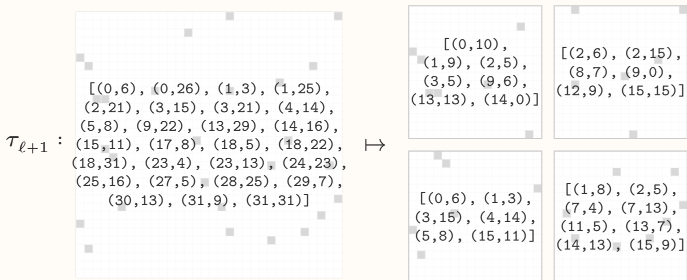
A density shift: configurations with $\mathcal{O}(n^\alpha)$ *black cells per* $[\![n]\!]^d$ *pattern.*

Claim

All effective shifts $Y \subseteq \{■, □\}^{\mathbb{Z}^d}$ of density $\mathcal{O}(n^\alpha)$ $(\alpha < d-1)$ are sofic.

Substitution

Alphabet $\mathcal{A}_\ell \subseteq \mathcal{P}(⟦0, 2^\ell{-}1⟧^d)$ Radius $V_\ell = \{\text{-}1, 0, 1\}^d$

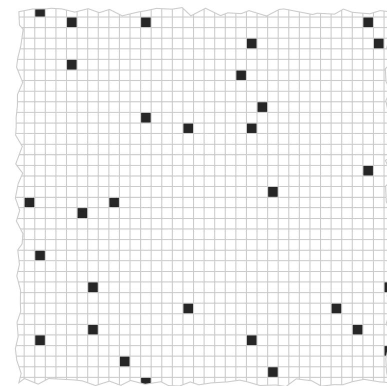


1. Split the list $\{0\}^d$ into 2^d quadrants (+ coordinate shifts);
2. Use radius to check against $\mathcal{O}(\ell)$ forbidden patterns;

✅ Output the quadrants ❌ No output

Conditions

✅ **Growth:** $N_{\ell+1} = 2$; ✅ **Alphabet:** $\log |\mathcal{A}_{\ell+1}| = \tilde{\mathcal{O}}(L_\ell^\alpha)$;
✅ **Computability:** $\tau_{\ell+1}$ is computable in time $\tilde{\mathcal{O}}(L_\ell^\alpha)$;



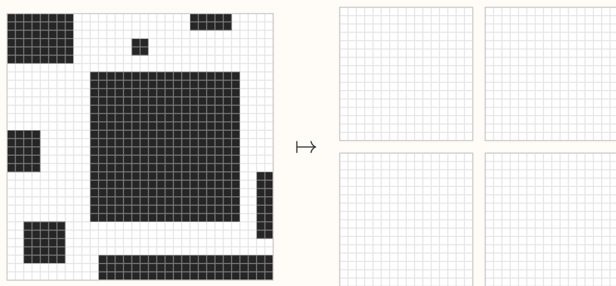
A density shift: configurations with $\mathcal{O}(n^\alpha)$ black cells per $⟦n⟧^d$ pattern.

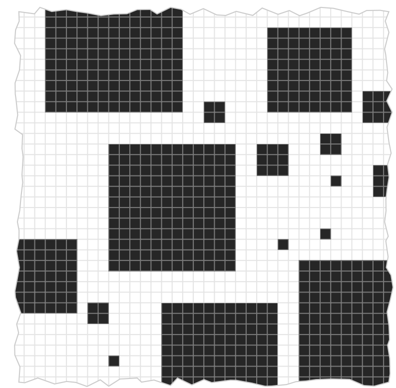
Claim

Let $S \subseteq \mathbb{N}$ be a (Π^0_1)-computable set. Then the S-square shift Y_S is sofic.

Substitution

$\tau_{\ell+1}$:





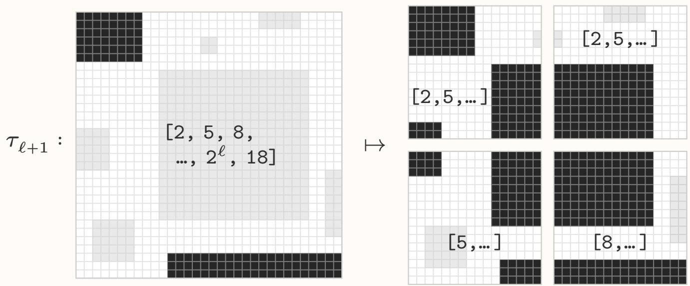
The S-square shift: black squares of sizes S over a white background.

Claim

Let $S \subseteq \mathbb{N}$ be a (Π_1^0)-computable set. Then the S-square shift Y_S is sofic.

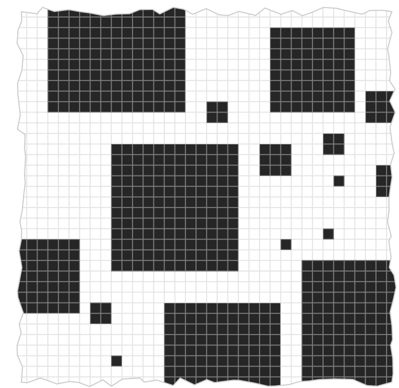
Substitution

Alphabet $\mathcal{A}_\ell \subseteq \mathcal{P}(\llbracket 0, 2^\ell{-}1 \rrbracket) \times \{\text{corners}\}$ Radius $V_\ell = \{\text{-}1, 0, 1\}^d$



1. Split into 2×2 quadrants + introduce internal corners;
2. Verify the alignment of corners in the neighborhood V_ℓ;
3. Verify each size (for $\mathcal{O}(\ell)$ steps) against S;

✅ Output the image ❌ No output



The S-square shift: black squares of sizes S over a white background.

Conditions

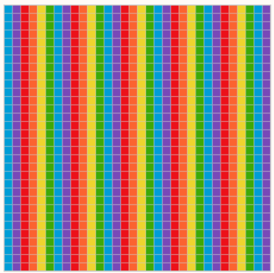
✅ **Growth:** $N_{\ell+1} = 2$; ✅ **Alphabet:** $\log |\mathcal{A}_{\ell+1}| = \tilde{\mathcal{O}}(L_\ell^{2/3})$;

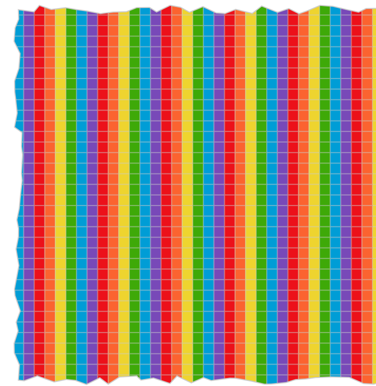
✅ **Computability:** $\tau_{\ell+1}$ is computable in time $\tilde{\mathcal{O}}(L_\ell^{2/3})$;

Claim

A shift $Y \subseteq \mathcal{A}^{\mathbb{Z}^d}$ is effective iff its periodic lift $Y^{\uparrow} \subseteq \mathcal{A}^{\mathbb{Z}^{d+1}}$ is sofic.

Substitution

$\tau_{\ell+1}:$  $\mapsto$



The shift $Y^{\uparrow}$: configurations of $Y \subseteq \mathcal{A}^{\mathbb{Z}^d}$ repeated vertically.

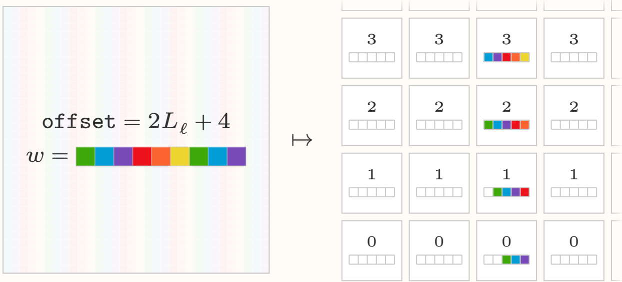
Claim

A shift $Y \subseteq \mathcal{A}^{\mathbb{Z}^d}$ is effective iff its periodic lift $Y^{\uparrow} \subseteq \mathcal{A}^{\mathbb{Z}^{d+1}}$ is sofic.

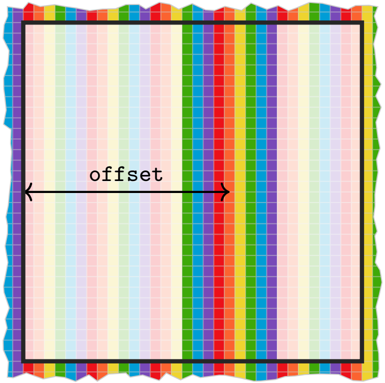
Substitution

Alphabet $\mathcal{A}_\ell = [\![0, L_\ell{-}1]\!] \times \mathcal{A}^{\log L_\ell + 1}$ Radius $V_\ell = \{\text{-}1, 0, 1\}^d$

$\tau_{\ell+1}$:



1. Verify the offsets in the neighborhood V_ℓ;
2. Attribute offset $j \in [\![0, L_\ell{-}1]\!]$ to all image cells of height j;
3. Distribute the word w to image cells ;



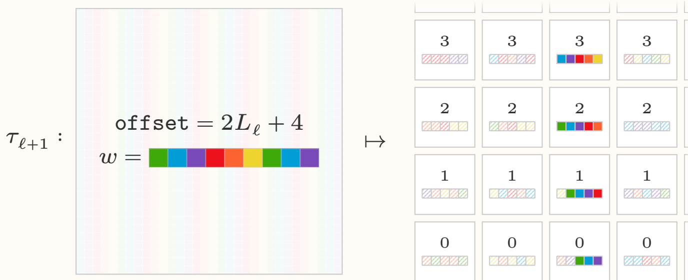
The shift $Y^{\uparrow}$: configurations of $Y \subseteq \mathcal{A}^{\mathbb{Z}^d}$ repeated vertically.

Claim

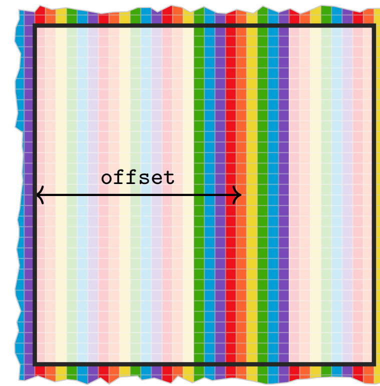
A shift $Y \subseteq \mathcal{A}^{\mathbb{Z}^d}$ is effective iff its periodic lift $Y^{\uparrow} \subseteq \mathcal{A}^{\mathbb{Z}^{d+1}}$ is sofic.

Substitution

Alphabet $\mathcal{A}_\ell = [\![0, L_\ell{-}1]\!] \times \mathcal{A}^{\log L_\ell + 1}$ Radius $V_\ell = \{-1, 0, 1\}^d$



1. Verify the offsets in the neighborhood V_ℓ;
2. Attribute offset $j \in [\![0, L_\ell{-}1]\!]$ to all image cells of height j;
3. Distribute the word w to image cells (+ non-det. filling);



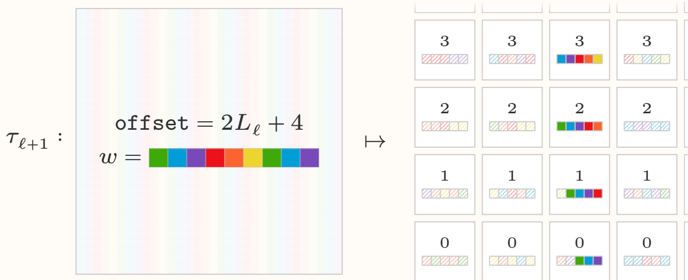
The shift $Y^{\uparrow}$: configurations of $Y \subseteq \mathcal{A}^{\mathbb{Z}^d}$ repeated vertically.

Claim

A shift $Y \subseteq \mathcal{A}^{\mathbb{Z}^d}$ is effective iff its periodic lift $Y^{\uparrow} \subseteq \mathcal{A}^{\mathbb{Z}^{d+1}}$ is sofic.

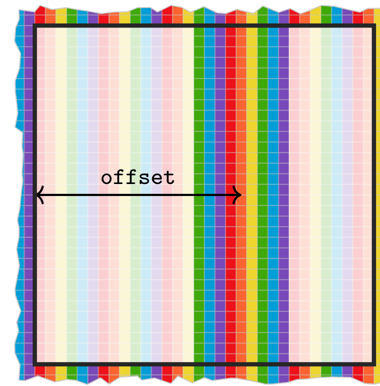
Substitution

Alphabet $\mathcal{A}_\ell = [\![0, L_\ell{-}1]\!] \times \mathcal{A}^{\log L_\ell + 1}$ Radius $V_\ell = \{\text{-}1, 0, 1\}^d$



1. Verify the offsets in the neighborhood V_ℓ;
2. Attribute offset $j \in [\![0, L_\ell{-}1]\!]$ to all image cells of height j;
3. Distribute the word w to image cells (+ non-det. filling);
4. Check w against Y-forbidden patterns (for $\mathcal{O}(\log L_\ell)$ steps)

✓ Output the image   ✗ No output



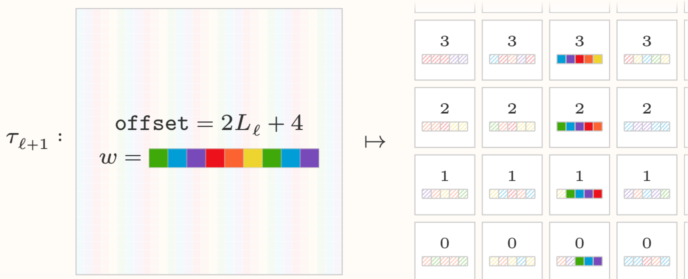
The shift $Y^{\uparrow}$: configurations of $Y \subseteq \mathcal{A}^{\mathbb{Z}^d}$ repeated vertically.

Claim

A shift $Y \subseteq \mathcal{A}^{\mathbb{Z}^d}$ is effective iff its periodic lift $Y^{\uparrow} \subseteq \mathcal{A}^{\mathbb{Z}^{d+1}}$ is sofic.

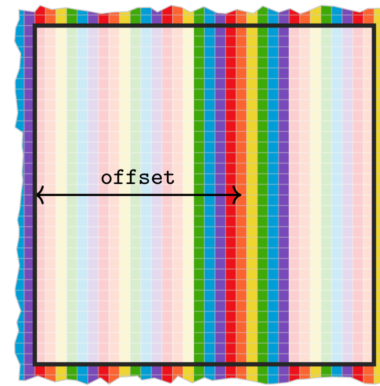
Substitution

Alphabet $\mathcal{A}_\ell = [\![0, L_\ell{-}1]\!] \times \mathcal{A}^{\log L_\ell + 1}$ Radius $V_\ell = \{\text{-}1, 0, 1\}^d$



1. Verify the offsets in the neighborhood V_ℓ;
2. Attribute offset $j \in [\![0, L_\ell{-}1]\!]$ to all image cells of height j;
3. Distribute the word w to image cells (+ non-det. filling);
4. Check w against Y-forbidden patterns (for $\mathcal{O}(\log L_\ell)$ steps)

✓ Output the image ✗ No output



The shift $Y^{\uparrow}$: configurations of $Y \subseteq \mathcal{A}^{\mathbb{Z}^d}$ repeated vertically.

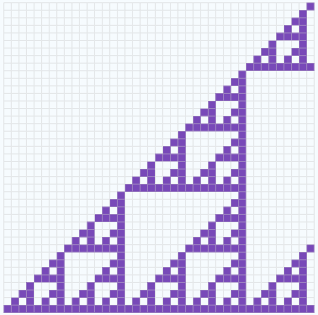
Conditions

✓ **Growth:** $N_{\ell+1} = L_\ell$; ✓ **Alphabet:** $\log|\mathcal{A}_{\ell+1}| = \mathcal{O}(\log L_\ell)$;

✓ **Computability:** $\tau_{\ell+1}$ is computable in time $\mathcal{O}(\log^2 L_\ell)$;

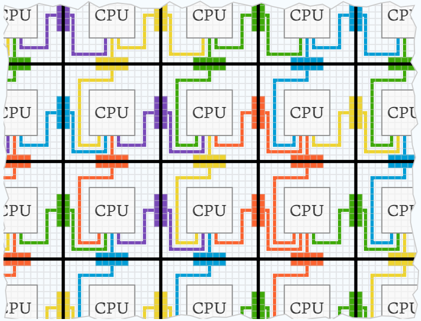
Take-home message

S-adic shifts: $x = x_0 \xleftarrow{\tau_1} x_1 \xleftarrow{\tau_2} x_2 \xleftarrow{\tau_3} \dots$



$\Longleftrightarrow$

Self-simulation



*: up to computational conditions and change of hierarchy;

Soficity

- Prove the soficity of shifts with information bounds $O(n^{d-1})$.

Future work

- Applications: directions of expansivity ([Zinoviadis, 2016]), invariant measures?
- Cover of finite type (entropy, determinism...);
- Other (Cayley graphs of) groups.


That's all Folks!