The domino problem for structures between $\mathbb{Z}$ and $\mathbb{Z}^2$.

Sebastián Barbieri
LIP, ENS de Lyon

Paris-Est-Créteil
November, 2015

G-subshifts

Consider a group G.

- $\mathcal{A}$ is a finite alphabet. Ex : $\mathcal{A} = \{0, 1\}$.
- $\mathcal{A}^G$ is the set of functions $x : G \to \mathcal{A}$.
- $\sigma : G \times \mathcal{A}^G \to \mathcal{A}^G$ is the shift action given by :

$$\sigma_g(x)_h = x_{g^{-1}h}.$$

G-subshifts

Consider a group G.

- $\mathcal{A}$ is a finite alphabet. Ex : $\mathcal{A} = \{0, 1\}$.
- $\mathcal{A}^G$ is the set of functions $x : G \to \mathcal{A}$.
- $\sigma : G \times \mathcal{A}^G \to \mathcal{A}^G$ is the shift action given by :

$$\sigma_g(x)_h = x_{g^{-1}h}.$$

Definition : G-subshift

$X \subset \mathcal{A}^G$ is a G-subshift if it is invariant under the action of σ and closed for the product topology on $\mathcal{A}^G$.

G-subshifts

Consider a group G.

- $\mathcal{A}$ is a finite alphabet. Ex : $\mathcal{A} = \{0,1\}$.
- $\mathcal{A}^G$ is the set of functions $x : G \to \mathcal{A}$.
- $\sigma : G \times \mathcal{A}^G \to \mathcal{A}^G$ is the shift action given by :

$$\sigma_g(x)_h = x_{g^{-1}h}.$$

Definition : G-subshift

$X \subset \mathcal{A}^G$ is a G-subshift if it is invariant under the action of σ and closed for the product topology on $\mathcal{A}^G$.

Alternative definition : G-subshift

X is a G-subshift if it can be defined as the set of configurations which avoid a set forbidden patterns : $\exists \mathcal{F} \subset \bigcup_{F \subset G, |F| < \infty} \mathcal{A}^F$ such that :

$$X = X_{\mathcal{F}} := \{x \in \mathcal{A}^G \mid \forall p \in \mathcal{F} : p \not\sqsubset x\}.$$

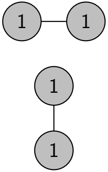
Example in $\mathbb{Z}^2$: Fibonacci shift

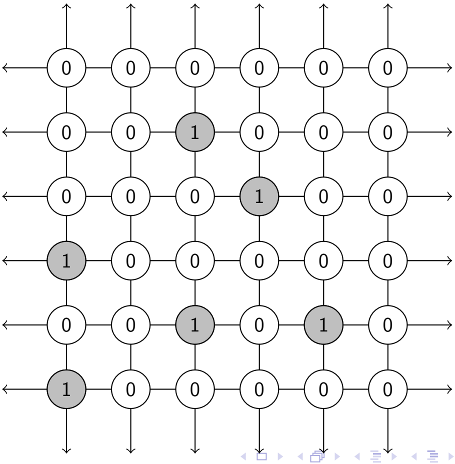
Example : Fibonacci shift. X_{Fib} is the set of assignments of $\mathbb{Z}^2$ to $\{0, 1\}$ such that there are no two adjacent ones.

Example in $\mathbb{Z}^2$: Fibonacci shift

Example : Fibonacci shift. X_{Fib} is the set of assignments of $\mathbb{Z}^2$ to $\{0,1\}$ such that there are no two adjacent ones.

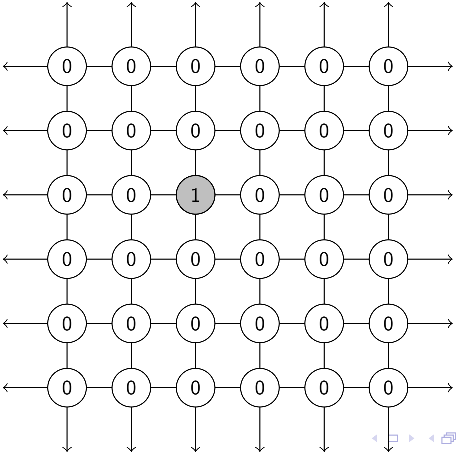
$\mathcal{F} =$



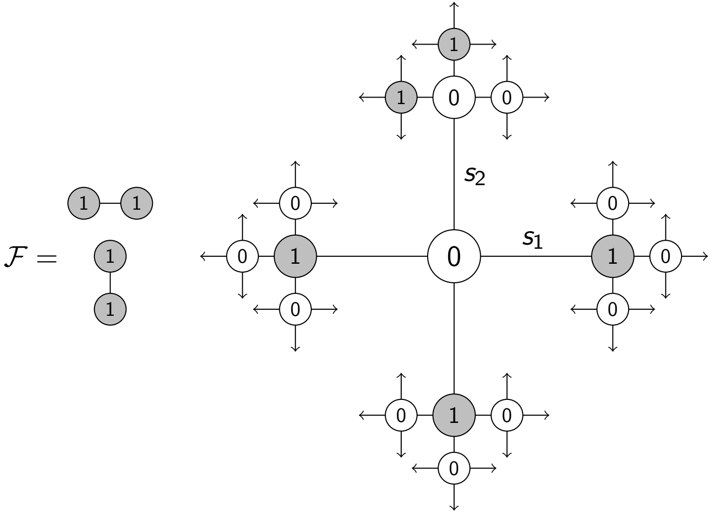


Example : one-or-less subshift.

$$X_{\leq 1} := \{x \in \{0,1\}^{\mathbb{Z}^d} \mid |\{z \in \mathbb{Z}^d : x_z = 1\}| \leq 1\}.$$



Fibonacci in F_2.



Subshifts of finite type.

- What about if we only consider local rules ?

Definition : subshift of finite type.

A G-subshift is of finite type (SFT) if it can be defined by a finite set $\mathcal{F}$ of forbidden patterns.

Example : Both Fibonacci subshifts shown before are of finite type. $X_{\leq 1}$ isn't.

Subshifts of finite type.

- What about if we only consider local rules ?

Definition : subshift of finite type.

A G-subshift is of finite type (SFT) if it can be defined by a finite set $\mathcal{F}$ of forbidden patterns.

Example : Both Fibonacci subshifts shown before are of finite type. $X_{\leq 1}$ isn't.

- Given a finite set of forbidden patterns, can we decide if the G-subshift produced by them is non-empty ?

The domino problem.

- Every finite alphabet can be identified as a finite subset of $\mathbb{N}$.

Domino problem.

$$\mathtt{DP}(G) = \{\mathcal{F} \subset \mathbb{N}_G^* \mid |\mathcal{F}| < \infty, X_{\mathcal{F}} \neq \emptyset\}.$$

The domino problem.

- Every finite alphabet can be identified as a finite subset of $\mathbb{N}$.

Domino problem.

$$\mathtt{DP}(G) = \{\mathcal{F} \subset \mathbb{N}^*_G \mid |\mathcal{F}| < \infty, X_{\mathcal{F}} \neq \emptyset\}.$$

- if G is finitely generated by the set S, we can codify each pattern as a function from a finite set of words in $(S \cup S^{-1})^*$ to $\mathbb{N}$.
- Therefore, $\mathtt{DP}(G)$ can be written as a formal language. We say G has decidable domino problem if $\mathtt{DP}(G)$ is Turing-decidable.

The domino problem.

- Every finite alphabet can be identified as a finite subset of $\mathbb{N}$.

Domino problem.

$$\mathtt{DP}(G) = \{\mathcal{F} \subset \mathbb{N}^*_G \mid |\mathcal{F}| < \infty, X_{\mathcal{F}} \neq \emptyset\}.$$

- if G is finitely generated by the set S, we can codify each pattern as a function from a finite set of words in $(S \cup S^{-1})^*$ to $\mathbb{N}$.
- Therefore, $\mathtt{DP}(G)$ can be written as a formal language. We say G has decidable domino problem if $\mathtt{DP}(G)$ is Turing-decidable.

Question : Which groups have decidable domino problem ?

The easy case $G = \mathbb{Z}$.

Theorem :

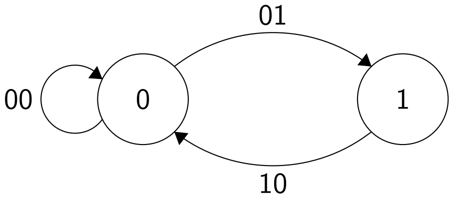
The set of configurations of a $\mathbb{Z}$-SFT can be characterized as the set of bi-infinite walks in a finite graph.

The easy case $G = \mathbb{Z}$.

Theorem :

The set of configurations of a $\mathbb{Z}$-SFT can be characterized as the set of bi-infinite walks in a finite graph.

Example : Consider the Fibonacci shift given by $\mathcal{F} = \{11\}$.

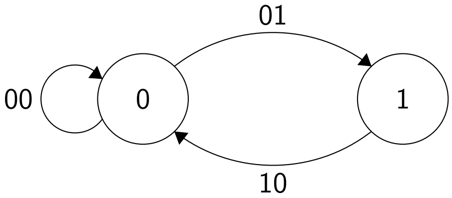


The easy case $G = \mathbb{Z}$.

Theorem :

The set of configurations of a $\mathbb{Z}$-SFT can be characterized as the set of bi-infinite walks in a finite graph.

Example : Consider the Fibonacci shift given by $\mathcal{F} = \{11\}$.



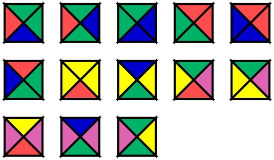
As the graph is finite, a $\mathbb{Z}$-SFT is non-empty if and only if its Rauzy graph contains a cycle, thus $\mathtt{DP}(\mathbb{Z})$ is decidable.

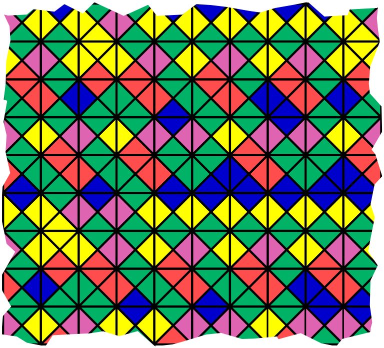
The not so easy case : $G = \mathbb{Z}^2$

The name "Domino problem" comes from the $G = \mathbb{Z}^2$ case.

The not so easy case : $G = \mathbb{Z}^2$

The name "Domino problem" comes from the $G = \mathbb{Z}^2$ case. Wang tiles are unit squares with colored edges, the forbidden patterns are implicit in the alphabet.





Wang's conjecture

Wang's conjecture (1961)

If a set of Wang tiles can tile the plane, then they can always be arranged to do so periodically.

Wang's conjecture

Wang's conjecture (1961)

If a set of Wang tiles can tile the plane, then they can always be arranged to do so periodically.

If Wang's conjecture is true, we can decide if a set of Wang tiles can tile the plane !

Semi-algorithm 1 :

1. Accept if there is a periodic configuration.
2. loops otherwise

Semi-algorithm 2 :

1. Accept if a block $[0, n]^2$ cannot be tiled without breaking local rules.
2. loops otherwise

Wang's conjecture

Theorem[Berger 1966]

Wang's conjecture is FALSE

Wang's conjecture

Theorem[Berger 1966]

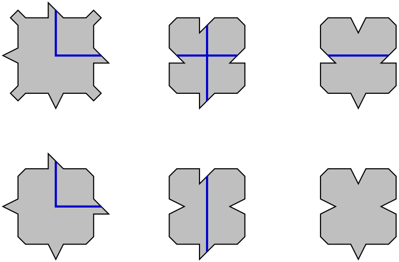
Wang's conjecture is FALSE

His construction encodes a Turing machine using an alphabet of size 20426.

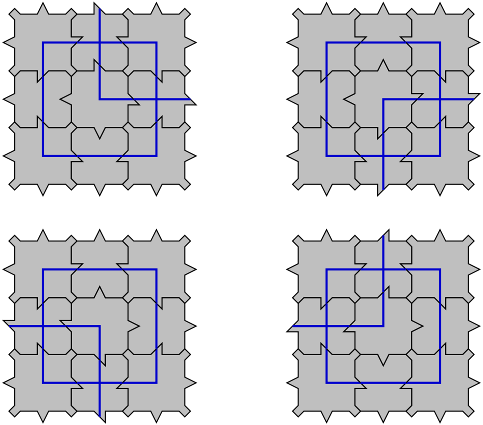
His proof was later simplified by Robinson[1971]. A proof with a different approach was also presented by Kari[1996].

Robinson tileset

The Robinson tileset, where tiles can be rotated.

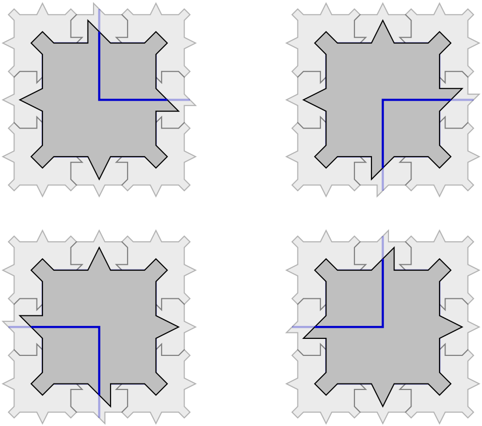


Macro-tiles of level 1.



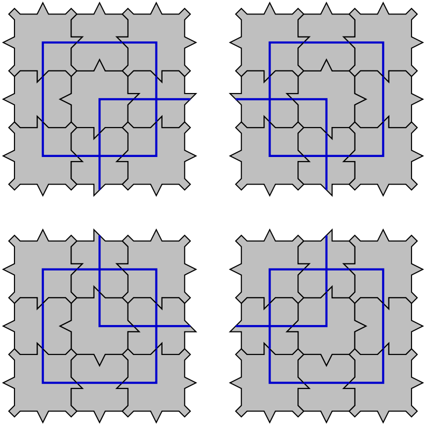
General structure of the Robinson tiling

Macro-tiles of level 1.

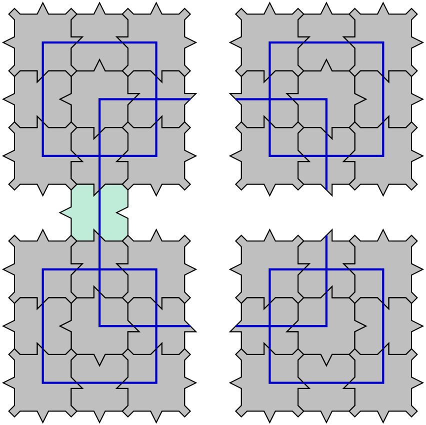


They behave like large .

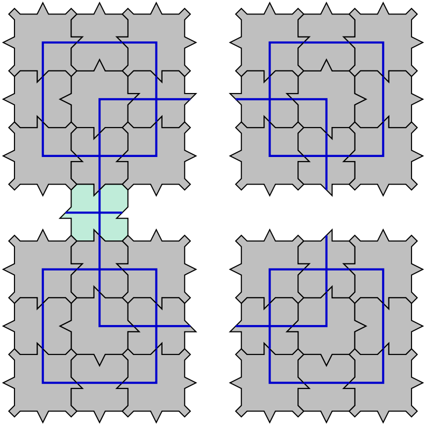
From macro-tiles of level 1 to macro-tiles of level 2



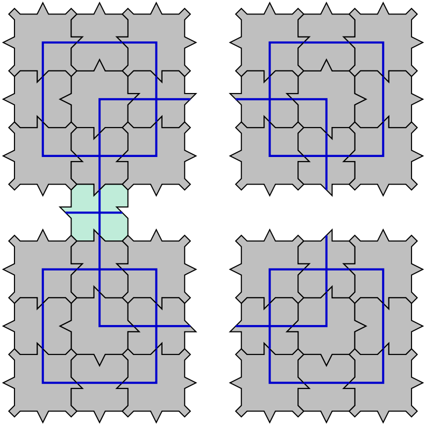
From macro-tiles of level 1 to macro-tiles of level 2



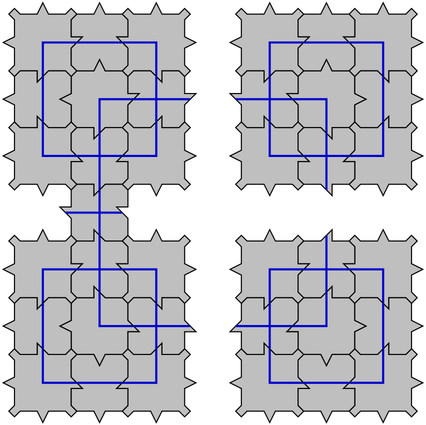
From macro-tiles of level 1 to macro-tiles of level 2



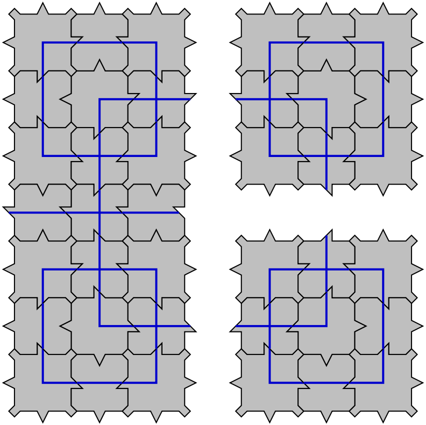
From macro-tiles of level 1 to macro-tiles of level 2



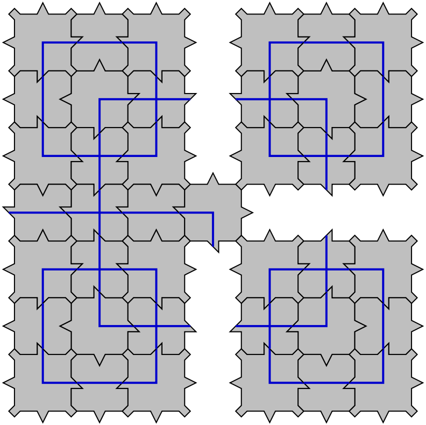
From macro-tiles of level 1 to macro-tiles of level 2



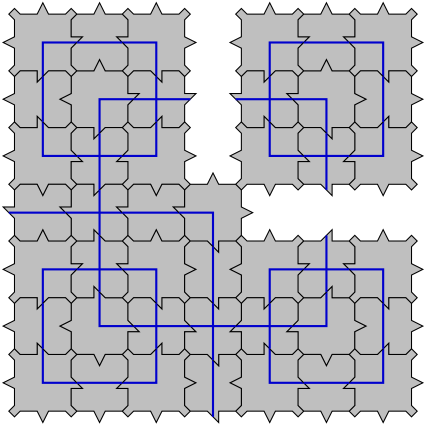
From macro-tiles of level 1 to macro-tiles of level 2



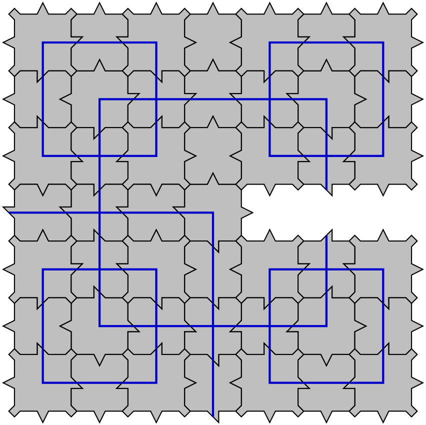
From macro-tiles of level 1 to macro-tiles of level 2



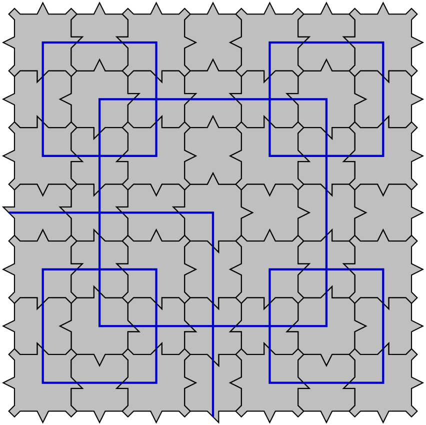
From macro-tiles of level 1 to macro-tiles of level 2



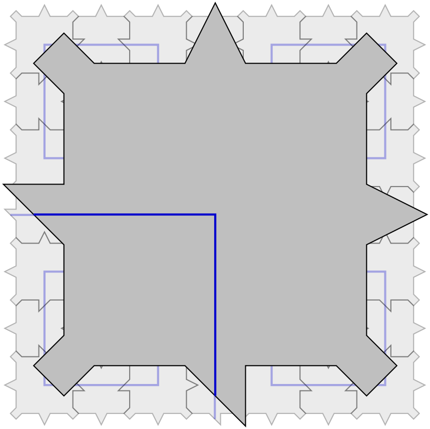
From macro-tiles of level 1 to macro-tiles of level 2



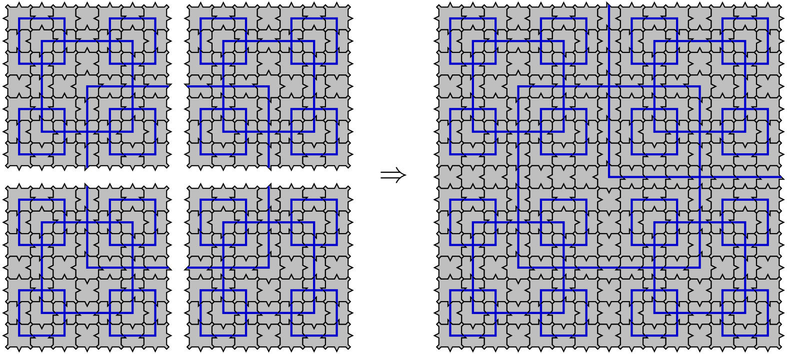
From macro-tiles of level 1 to macro-tiles of level 2



From macro-tiles of level 1 to macro-tiles of level 2



From macro-tiles of level n to macro-tiles of level $n + 1$



Some recent results and facts in f.g. groups

- If a group G has undecidable word problem $\Rightarrow \mathtt{DP}(G)$ is undecidable.
- Virtually free groups have decidable domino problem.
- For virtually nilpotent groups : $\mathtt{DP}(G)$ is decidable if and only if it has two or more ends (2013 Ballier, Stein).
- Every virtually polycyclic group which is not virtually $\mathbb{Z}$ has undecidable domino problem (work in progress by Jeandel).
- The domino problem is a quasi-isometry invariant for finitely presented groups (2015 Cohen).

Going between $\mathbb{Z}$ and $\mathbb{Z}^2$ (Joint work with M. Sablik)

So far we have :

- $\mathtt{DP}(\mathbb{Z})$ is decidable.
- $\mathtt{DP}(\mathbb{Z}^2)$ is undecidable.

And if $H \leq \mathbb{Z}^2$, then either $H \cong 1, H \cong \mathbb{Z}$ or $H \cong \mathbb{Z}^2$.

Going between $\mathbb{Z}$ and $\mathbb{Z}^2$ (Joint work with M. Sablik)

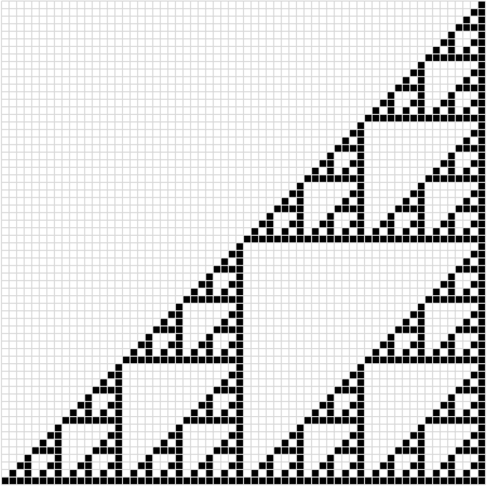
So far we have :

- $\mathtt{DP}(\mathbb{Z})$ is decidable.
- $\mathtt{DP}(\mathbb{Z}^2)$ is undecidable.

And if $H \leq \mathbb{Z}^2$, then either $H \cong 1$, $H \cong \mathbb{Z}$ or $H \cong \mathbb{Z}^2$.

We need to lose the group structure if we want to study intermediate structures.

Toy case : Sierpiński triangle



Coding subsets of $\mathbb{Z}^2$ as configurations.

Let $F \subset \mathbb{Z}^2$ and define the configuration $x_F \in \{0,1\}^{\mathbb{Z}^2}$:

$$(x_F)_z = \begin{cases} 1 \text{ if } z \in F \\ 0 \text{ if not.} \end{cases}$$

And let $Y = \overline{\bigcup_{z \in \mathbb{Z}^2} \{\sigma_z(x_F)\}}$.

Coding subsets of $\mathbb{Z}^2$ as configurations.

Let $F \subset \mathbb{Z}^2$ and define the configuration $x_F \in \{0,1\}^{\mathbb{Z}^2}$:

$$(x_F)_z = \begin{cases} 1 \text{ if } z \in F \\ 0 \text{ if not.} \end{cases}$$

And let $Y = \overline{\bigcup_{z \in \mathbb{Z}^2} \{\sigma_z(x_F)\}}$.
Given a set of forbidden patterns $\mathcal{F}$ we can define colorings of F as the configurations of $X_{\mathcal{F}}$ over an alphabet $\mathcal{A} \ni 0$ such that the application $\pi : \mathcal{A}^{\mathbb{Z}^2} \to \{0,1\}^{\mathbb{Z}^2}$:

$$\pi(x)_z = \begin{cases} 1 \text{ if } x_z \neq 0 \\ 0 \text{ if } x_z = 0 \end{cases}$$

yields an element of Y.

Formally...

- Let $Y \ni 0^{\mathbb{Z}^2}$ be a $\mathbb{Z}^2$-subshift over the alphabet $\{0,1\}$.
- Let $\mathcal{F}$ be a set of forbidden patterns over an alphabet $\mathcal{A} \ni 0$ which does not forbid any pattern consisting only of 0.

Formally...

- Let $Y \ni 0^{\mathbb{Z}^2}$ be a $\mathbb{Z}^2$-subshift over the alphabet $\{0, 1\}$.
- Let $\mathcal{F}$ be a set of forbidden patterns over an alphabet $\mathcal{A} \ni 0$ which does not forbid any pattern consisting only of 0.

Definition : Y-based subshift

The Y-based subshift defined by $\mathcal{F}$ is the set :

$$X_{Y,\mathcal{F}} := \pi^{-1}(Y) \cap X_{\mathcal{F}}.$$

Formally...

- Let $Y \ni 0^{\mathbb{Z}^2}$ be a $\mathbb{Z}^2$-subshift over the alphabet $\{0,1\}$.
- Let $\mathcal{F}$ be a set of forbidden patterns over an alphabet $\mathcal{A} \ni 0$ which does not forbid any pattern consisting only of 0.

Definition : Y-based subshift

The Y-based subshift defined by $\mathcal{F}$ is the set :

$$X_{Y,\mathcal{F}} := \pi^{-1}(Y) \cap X_{\mathcal{F}}.$$

Definition : Y-based domino problem

$$\mathtt{DP}(Y) := \{\mathcal{F} \subset \mathbb{N}^*_{\mathbb{Z}^2} \mid |\mathcal{F}| < \infty \text{ and } X_{Y,\mathcal{F}} \neq \{0^{\mathbb{Z}^2}\}\}.$$

Back to the fractal structures...

We focus on subshifts Y with a self-similar structure generated by substitutions.

Back to the fractal structures...

We focus on subshifts Y with a self-similar structure generated by substitutions.

- If Y contains a strongly periodic point which is not $0^{\mathbb{Z}^2}$ then $\mathtt{DP}(Y)$ is undecidable.
- It is easy to calculate a Hausdorff dimension (in this case box-counting dimension). Is there a threshold in the dimension which enforces undecidability ?
- These subshifts can be defined by local rules (sofic subshifts) according to Mozes Theorem.

Back to the fractal structures...

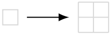
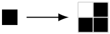
We focus on subshifts Y with a self-similar structure generated by substitutions.

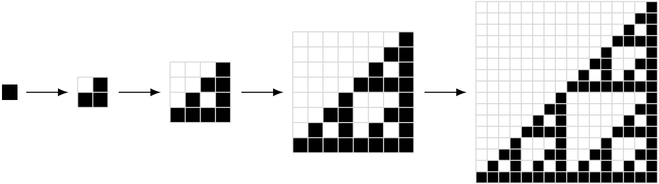
- If Y contains a strongly periodic point which is not $0^{\mathbb{Z}^2}$ then $\mathtt{DP}(Y)$ is undecidable.
- It is easy to calculate a Hausdorff dimension (in this case box-counting dimension). Is there a threshold in the dimension which enforces undecidability ?
- These subshifts can be defined by local rules (sofic subshifts) according to Mozes Theorem.

In particular we consider : substitutions over $\{0, 1\}$ such that the image of 0 is a rectangle of zeros.

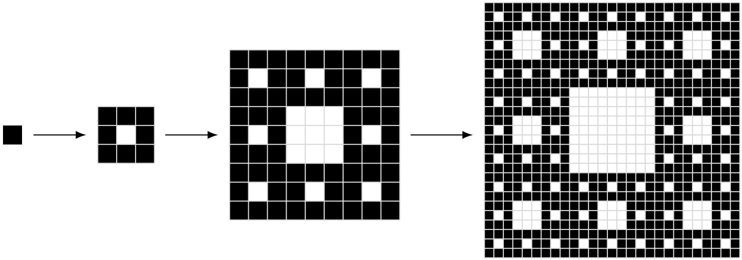
Example 1 : Sierpiński triangle

Consider the alphabet $\mathcal{A} = \{\square, \blacksquare\}$ and the self-similar substitution s such that :

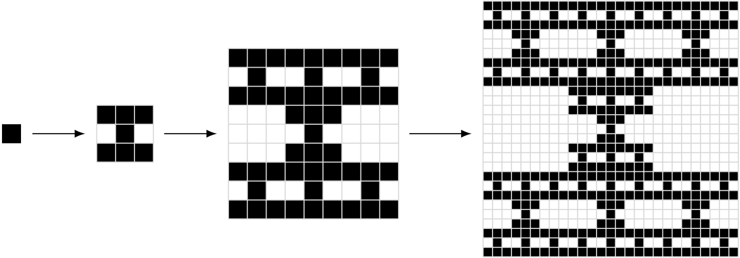
 and 



Example 2 : Sierpiński carpet



Example 3 : The Bridge.



Toy case 1 : Sierpiński triangle.

Theorem :

The domino problem is decidable in the Sierpiński triangle.

Toy case 1 : Sierpiński triangle.

Theorem :

The domino problem is decidable in the Sierpiński triangle.

Proof strategy :

- Consider a rectangle containing the union of the support of all forbidden patterns.

Toy case 1 : Sierpiński triangle.

Theorem :
The domino problem is decidable in the Sierpiński triangle.

Proof strategy :

- Consider a rectangle containing the union of the support of all forbidden patterns.
- Suppose we can tile locally an iteration n of the substitution without producing forbidden patterns.

Toy case 1 : Sierpiński triangle.

Theorem :

The domino problem is decidable in the Sierpiński triangle.

Proof strategy :

- Consider a rectangle containing the union of the support of all forbidden patterns.
- Suppose we can tile locally an iteration n of the substitution without producing forbidden patterns.
- To construct a tiling of the next level, it suffices to "paste" three tilings of the iteration n without producing forbidden patterns.

Toy case 1 : Sierpiński triangle.

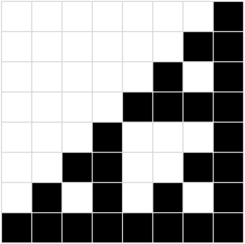
Theorem :

The domino problem is decidable in the Sierpiński triangle.

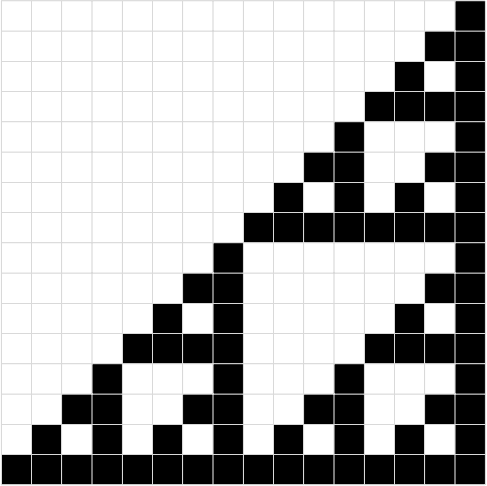
Proof strategy :

- Consider a rectangle containing the union of the support of all forbidden patterns.
- Suppose we can tile locally an iteration n of the substitution without producing forbidden patterns.
- To construct a tiling of the next level, it suffices to "paste" three tilings of the iteration n without producing forbidden patterns.

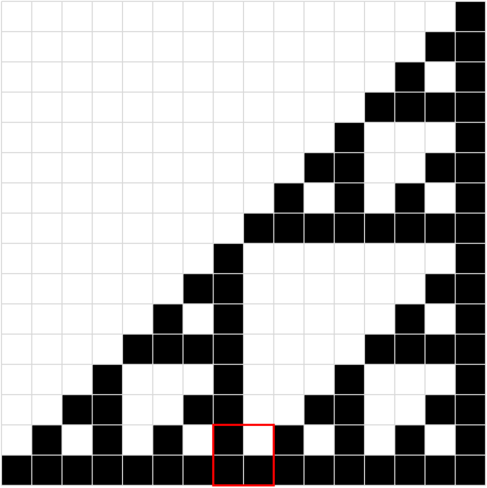
Toy case 1 : Sierpiński triangle.



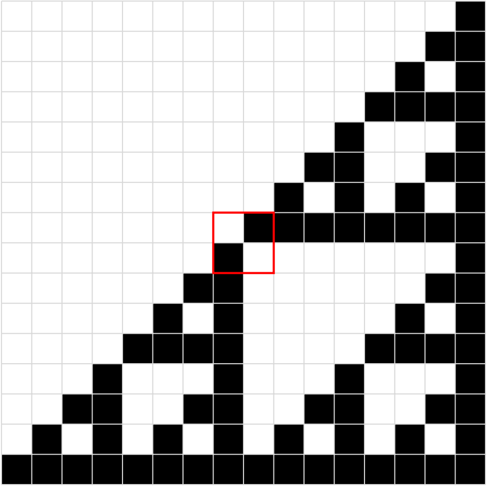
Toy case 1 : Sierpiński triangle.



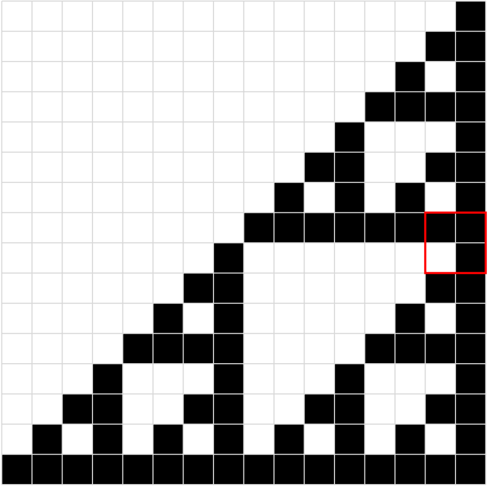
Toy case 1 : Sierpiński triangle.



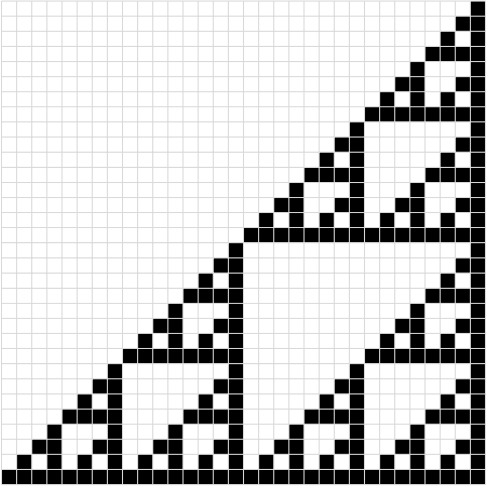
Toy case 1 : Sierpiński triangle.



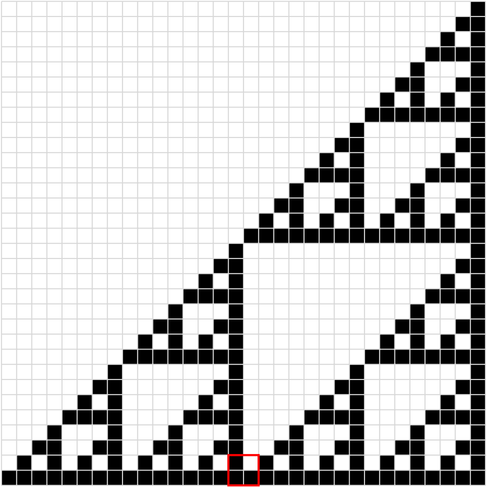
Toy case 1 : Sierpiński triangle.



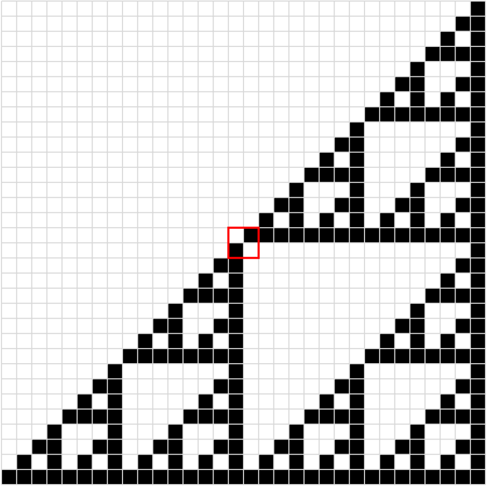
Toy case 1 : Sierpiński triangle.



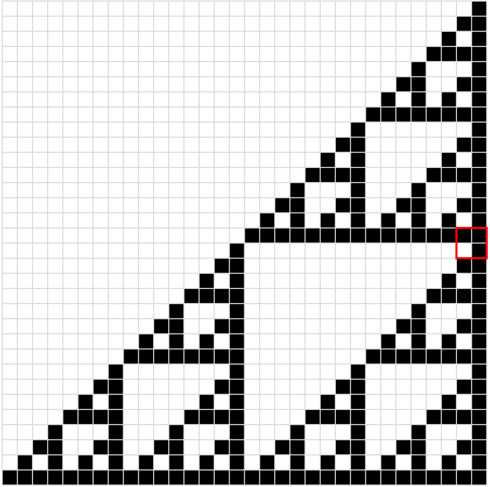
Toy case 1 : Sierpiński triangle.



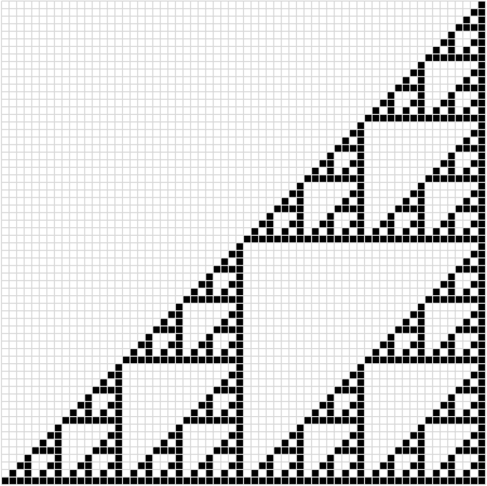
Toy case 1 : Sierpiński triangle.



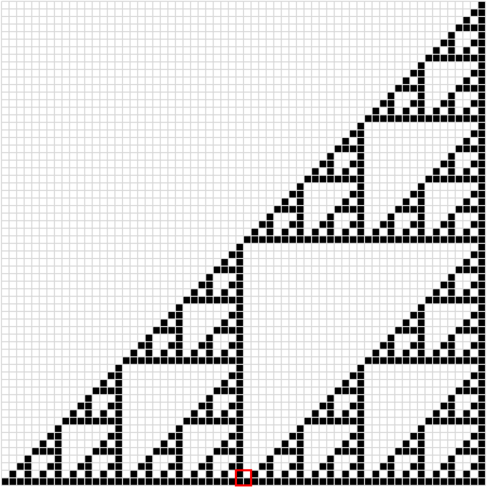
Toy case 1 : Sierpiński triangle.



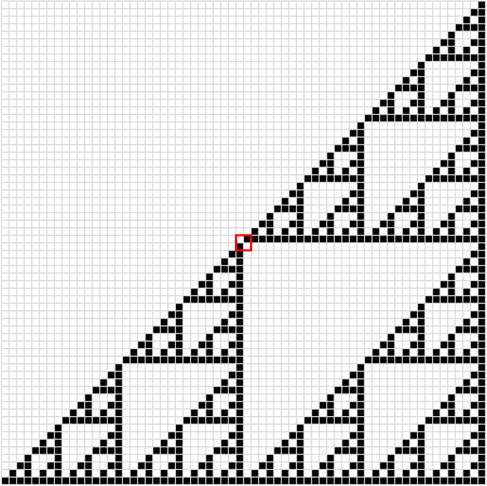
Toy case 1 : Sierpiński triangle.



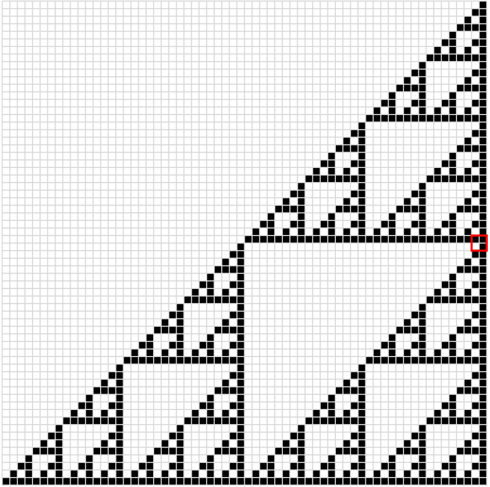
Toy case 1 : Sierpiński triangle.



Toy case 1 : Sierpiński triangle.



Toy case 1 : Sierpiński triangle.



Toy case 1 : Sierpiński triangle.

Theorem :

The domino problem is decidable in the Sierpiński triangle.

Proof strategy (continued) :

- Keep the information about the pasting places (finite tuples) and build pasting rules $(T_1, T_2, T_3) \to T_4$.

Toy case 1 : Sierpiński triangle.

Theorem :

The domino problem is decidable in the Sierpiński triangle.

Proof strategy (continued) :

- Keep the information about the pasting places (finite tuples) and build pasting rules $(T_1, T_2, T_3) \to T_4$.
- For each iteration n, construct the set of tuples observed in the pasting places. Construct the next set using this one.

Toy case 1 : Sierpiński triangle.

Theorem :

The domino problem is decidable in the Sierpiński triangle.

Proof strategy (continued) :

- Keep the information about the pasting places (finite tuples) and build pasting rules $(T_1, T_2, T_3) \rightarrow T_4$.
- For each iteration n, construct the set of tuples observed in the pasting places. Construct the next set using this one.
- This process either cycles (arbitrary iterations can be tiled) or ends up producing the empty set (the only valid tiling is $0^{\mathbb{Z}^2}$).

Toy case 1 : Sierpiński triangle.

Theorem :

The domino problem is decidable in the Sierpiński triangle.

Proof strategy (continued) :

- Keep the information about the pasting places (finite tuples) and build pasting rules $(T_1, T_2, T_3) \to T_4$.
- For each iteration n, construct the set of tuples observed in the pasting places. Construct the next set using this one.
- This process either cycles (arbitrary iterations can be tiled) or ends up producing the empty set (the only valid tiling is $0^{\mathbb{Z}^2}$).

This technique can be extended to a big class of self-similar substitutions !

Toy case 2 : Sierpiński carpet.

Theorem :

The domino problem is undecidable in the Sierpiński carpet.

Proof strategy :

- Suppose we can simulate substitutions over the Sierpiński carpet (using a bigger alphabet).

Toy case 2 : Sierpiński carpet.

Theorem :

The domino problem is undecidable in the Sierpiński carpet.

Proof strategy :

- Suppose we can simulate substitutions over the Sierpiński carpet (using a bigger alphabet).

Toy case 2 : Sierpiński carpet.

$$\bullet \xrightarrow{s'} \begin{matrix} \bullet & \leftrightarrow & \bullet \\ \updownarrow & 0 & \updownarrow \\ \bullet & \leftrightarrow & \bullet \end{matrix}, \qquad \updownarrow \xrightarrow{s'} \begin{matrix} \updownarrow & \bullet & \updownarrow \\ \updownarrow & 0 & \updownarrow \\ \updownarrow & \bullet & \updownarrow \end{matrix}, \qquad \leftrightarrow \xrightarrow{s'} \begin{matrix} \leftrightarrow & \leftrightarrow & \leftrightarrow \\ \bullet & 0 & \bullet \\ \leftrightarrow & \leftrightarrow & \leftrightarrow \end{matrix}.$$

Toy case 2 : Sierpiński carpet.

$$
\begin{array}{ccccccccc}
\bullet & \leftrightarrow & \bullet & \leftrightarrow & \leftrightarrow & \leftrightarrow & \bullet & \leftrightarrow & \bullet \\
\updownarrow & & \updownarrow & \bullet & & \bullet & \updownarrow & & \updownarrow \\
\bullet & \leftrightarrow & \bullet & \leftrightarrow & \leftrightarrow & \leftrightarrow & \bullet & \leftrightarrow & \bullet \\
\updownarrow & \bullet & \updownarrow & & & & \updownarrow & \bullet & \updownarrow \\
\updownarrow & & \updownarrow & & & & \updownarrow & & \updownarrow \\
\updownarrow & \bullet & \updownarrow & & & & \updownarrow & \bullet & \updownarrow \\
\bullet & \leftrightarrow & \bullet & \leftrightarrow & \leftrightarrow & \leftrightarrow & \bullet & \leftrightarrow & \bullet \\
\updownarrow & & \updownarrow & \bullet & & \bullet & \updownarrow & & \updownarrow \\
\bullet & \leftrightarrow & \bullet & \leftrightarrow & \leftrightarrow & \leftrightarrow & \bullet & \leftrightarrow & \bullet
\end{array}
$$

Toy case 2 : Sierpiński carpet.

$$
\begin{array}{cccc}
a_1 & a_2 & a_3 & a_4 \\
b_1 & b_2 & b_3 & b_4 \\
c_1 & c_2 & c_3 & c_4 \\
d_1 & d_2 & d_3 & d_4
\end{array}
\longleftrightarrow
\begin{array}{ccccccccc}
\bullet & \leftrightarrow & \bullet & \leftrightarrow & \leftrightarrow & \leftrightarrow & \bullet & \leftrightarrow & \bullet \\
\updownarrow & & \updownarrow & \bullet & & \bullet & \updownarrow & & \updownarrow \\
\bullet & \leftrightarrow & \bullet & \leftrightarrow & \leftrightarrow & \leftrightarrow & \bullet & \leftrightarrow & \bullet \\
\updownarrow & \bullet & \updownarrow & & & & \updownarrow & \bullet & \updownarrow \\
\updownarrow & & \updownarrow & & & & \updownarrow & & \updownarrow \\
\updownarrow & \bullet & \updownarrow & & & & \updownarrow & \bullet & \updownarrow \\
\bullet & \leftrightarrow & \bullet & \leftrightarrow & \leftrightarrow & \leftrightarrow & \bullet & \leftrightarrow & \bullet \\
\updownarrow & & \updownarrow & \bullet & & \bullet & \updownarrow & & \updownarrow \\
\bullet & \leftrightarrow & \bullet & \leftrightarrow & \leftrightarrow & \leftrightarrow & \bullet & \leftrightarrow & \bullet
\end{array}
\quad
\begin{array}{ccccccccc}
a_1 & a_2 & a_2 & a_3 & a_3 & a_3 & a_3 & a_4 & a_4 \\
b_1 & & b_2 & \star & & \star & b_3 & & b_4 \\
b_1 & b_2 & b_2 & b_3 & b_3 & b_3 & b_3 & b_4 & b_4 \\
c_1 & \star & c_2 & & & & c_3 & \star & c_4 \\
c_1 & & c_2 & & & & c_3 & & c_4 \\
c_1 & \star & c_2 & & & & c_3 & \star & c_4 \\
c_1 & c_2 & c_2 & c_3 & c_3 & c_3 & c_3 & c_4 & c_4 \\
d_1 & & d_2 & \star & & \star & d_3 & & d_4 \\
d_1 & d_2 & d_2 & d_3 & d_3 & d_3 & d_3 & d_4 & d_4
\end{array}
$$

Toy case 2 : Sierpiński carpet.

Theorem :
The domino problem is undecidable in the Sierpiński carpet.

Proof strategy :

- Suppose we can simulate substitutions over the Sierpiński carpet (using a bigger alphabet).
- Use the substitution shown above to simulate arbitrarily big patterns of a $\mathbb{Z}^2$-subshift

Toy case 2 : Sierpiński carpet.

Theorem :

The domino problem is undecidable in the Sierpiński carpet.

Proof strategy :

- Suppose we can simulate substitutions over the Sierpiński carpet (using a bigger alphabet).
  - Use the substitution shown above to simulate arbitrarily big patterns of a $\mathbb{Z}^2$-subshift
  - $\mathtt{DP}(\mathbb{Z}^2)$ is reduced to the domino problem in the carpet.

Toy case 2 : Sierpiński carpet.

Theorem :

The domino problem is undecidable in the Sierpiński carpet.

Proof strategy :

- Suppose we can simulate substitutions over the Sierpiński carpet (using a bigger alphabet).
  - Use the substitution shown above to simulate arbitrarily big patterns of a $\mathbb{Z}^2$-subshift
  - $\mathtt{DP}(\mathbb{Z}^2)$ is reduced to the domino problem in the carpet.

It only remains to show that we can simulate substitutions with local rules.

Toy case 2 : Sierpiński carpet and Mozes

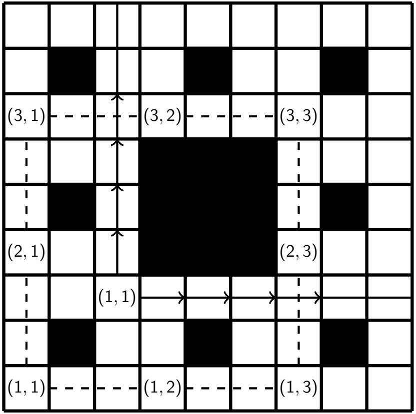
We need to prove a modified version of Mozes' theorem :

Theorem : Mozes.

The subshifts generated by $\mathbb{Z}^2$-substitutions are sofic (are the image of an SFT under a cellular automaton)

We can prove a similar version for some Y-based subshifts. Among them the Sierpiński carpet.

Toy case 2 : Sierpiński carpet and Mozes



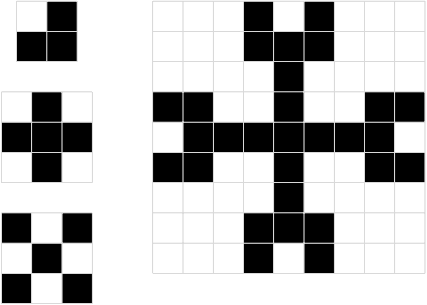
Conclusion

We can generalize the ideas in the previous toy problems to attack classes of substitutions :

Conclusion

We can generalize the ideas in the previous toy problems to attack classes of substitutions :

Bounded Connectivity

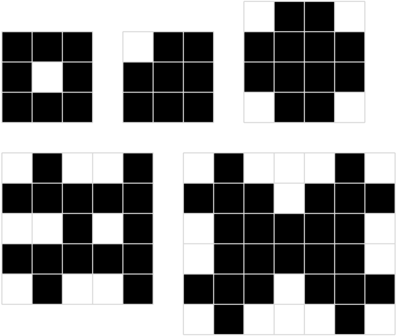


Decidable domino problem

Conclusion

We can generalize the ideas in the previous toy problems to attack classes of substitutions :

Strong grid



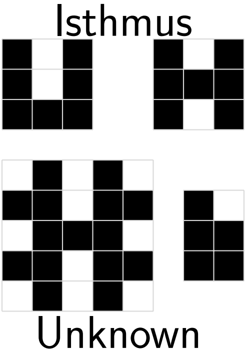
Undecidable domino problem

Conclusion

And separate the substitutions which we cannot classify into two groups :

Conclusion

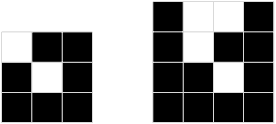
And separate the substitutions which we cannot classify into two groups :

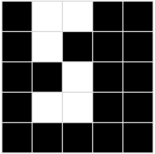


Conclusion

And separate the substitutions which we cannot classify into two groups :

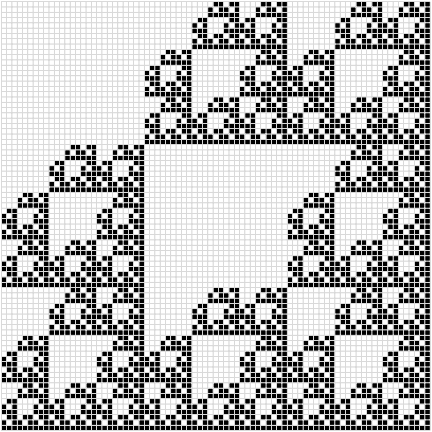
Weak grid





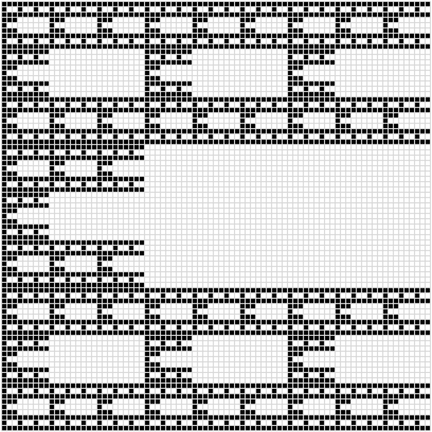
Unknown

Weak grid



We got some ideas of how it might be...

Isthmus



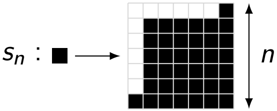
We don't know anything about this one.

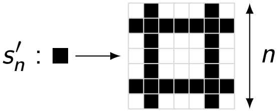
Conclusion

And about the Hausdorff dimension ?...

Conclusion

And about the Hausdorff dimension ?...





There is no threshold.

Thank you for your attention !