# Slopes of 3-dimensional Subshifts of Finite Type

Etienne Moutot[1] and Pascal Vanier[2]

[1] LIP, ENS de Lyon – CNRS – INRIA – UCBL – Université de Lyon,
6 allée d'Italie, 69364 Lyon Cedex, France
etienne.moutot@ens-lyon.org
[2] Laboratoire d'Algorithmique, Complexité et Logique
Université de Paris-Est, LACL, UPEC, France
pascal.vanier@lacl.fr

**Abstract.** In this paper we study the directions of periodicity of three-dimensional subshifts of finite type (SFTs) and in particular their slopes. A configuration of a subshift has a slope of periodicity if it is periodic in exactly one direction, the slope being the angles of the periodicity vector. In this paper, we prove that any $\Sigma_2^0$ set may be realized as a a set of slopes of an SFT.

A $d$-dimensional subshift of finite type (SFT for short) is a set of colorings of $\mathbb{Z}^d$ by a finite number of colors containing no pattern from a finite family of forbidden patterns. Subshifts may be seen as discretizations of continuous dynamical systems: if $X$ is a compact space and there are $d$ commuting continuous actions $\phi_1, \dots, \phi_d$ on $X$, one can partition $X$ in a finite number of parts indexed by an alphabet $\Sigma$. The orbit of a point $x \in X$ maps to a coloring $y$ of $\mathbb{Z}^d$ where $y(v)$ corresponds to the partition where $\phi^v(x)$ lies.

In dimension 1, most problems on SFTs are easy in a computational sense, since SFTs correspond to bi-infinite walks on finite automata. For instance, in dimension 1, detecting whether an SFT is non-empty is decidable since it suffices to detect if there exists a cycle in the corresponding automaton [13], which corresponds to the existence of a periodic configuration.

In higher dimensions however, the situation becomes more involved, and knowing whether an SFT is non-empty becomes undecidable [3,2]. The proof uses two key results on SFTs: the existence of an aperiodic SFT and an encoding of Turing machine's space time diagrams. The fact that there exists aperiodic SFTs is not straightforward, and the converse was first conjectured by Wang [20]. Had this conjecture been true, it would have meant the decidability of the emptiness problem for SFTs. Berger [3,2] proved however that there does exist SFTs containing only aperiodic configurations. Subsequently, many other aperiodic SFTs were constructed [18,12,16,17,8,4]. Note that the existence in itself of aperiodic SFTs does not suffice to prove that the emptiness problem is undecidable, one needs in addition to encode some computation in them, usually in the form of Turing machines.

Periodicity has thus been central in the study of SFTs from the beginning, and it has been proved very early that knowing whether an SFT is aperiodic is undecidable [6]. In fact, sets of periods constitute a classical conjugacy/isomorphism invariant for subshifts in any dimension. As such, they have been studied extensively and even characterized: algebraically in dimension 1, see [13] for more details, and computationally in dimension 2. In fact it seems that computability theory is the right tool to study dynamical aspects of higher dimensional symbolic dynamical systems [7,14,1,5].

In dimensions $d \geq 2$, one may investigate periodicity from different angles. Denote $\Gamma_x = \{v \in \mathbb{Z}^d \mid x(z+v) = x(z), \forall z \in \mathbb{Z}^d\}$ the lattice of vectors of periodicity of configuration $x$: $\Gamma_x$ may be of any dimension below $d$ and some cases are particularly interesting:

- When it is of dimension 0, then $x$ does not have any vector of periodicity and is hence aperiodic.
- When it is of dimension $d$, then $x$ is somehow finite, this case has been studied and partly characterized in terms of complexity classes by Jeandel and Vanier [10].
- When $d = 1$, then there exists some vector $v$ such that $\Gamma_x = v\mathbb{Z}$. In this case, one may talk about the direction or slope of the configuration.

In this paper, we are interested in this last case. In [9], this case was studied and characterized for 2-dimensional SFTs through the arithmetical hierarchy:

**Theorem** (Jeandel and Vanier)**.** The sets of slopes of 2-dimensional SFTs are exactly the $\Sigma_1^0$ subsets of $\mathbb{Q} \cup \{\infty\}$.

In the end of [9] it was conjectured that slopes of higher dimensional SFTs are the $\Sigma_2^0$ subsets of $(\mathbb{Q} \cup \{\infty\})^{d-1}$. This gap between dimension 2 and dimension 3 for decidability of periodicity questions is similar to the gap between dimension 1 and 2 for decidability of emptiness questions: the subset of periodic configurations of a $d$-dimensional subshift along some periodicity vector may be seen as a $(d-1)$-dimensional subshift (see *e.g.* [9]), hence the jump in complexity. This is the idea that led to the conjecture. However, in dimension higher than 2, the construction of [9] cannot be reused.

In this article, we prove one direction of the aforementioned conjecture: we show how to realize any $\Sigma_2^0$ subset of $(\mathbb{Q} \cup \{\infty\})^2$ as a set of slopes of a 3D subshift:

**Theorem 1.** *Any $\Sigma_2^0$ subset of $(\mathbb{Q} \cup \{\infty\})^2$ may be realized as the set of slopes of some 3D SFT.*

In order to do this, we introduce a new way to synchronize computations between different dimensions, inspired partly by what is done by Durand, Romashchenko and Shen [5]. Note that our construction can be easily generalized to realize any $\Sigma_2^0$ subset of $(\mathbb{Q} \cup \{\infty\})^{d-1}$ as a set of slopes of a $d$-dimensional subshift for $d \geq 3$.

However, we did not manage to prove the other part of the conjecture, that is the fact that the sets of slopes of $d$-dimensional SFTs are in $\Sigma_2^0$ (for $d \geq 3$).

The paper is organized as follows: in section 1 we recall the useful definitions about subshifts and the arithmetical hierarchy, and in section 2 we prove Theorem 1.

# 1 Definitions and properties

## 1.1 Subshifts and tilesets

We give here some standard definitions and facts about subshifts, one may consult [13] for more details.

Let $\Sigma$ be a finite alphabet, a *configuration* (or tiling) is a function $c : \mathbb{Z}^d \longrightarrow \Sigma$. A *pattern* is a function $p : N \longrightarrow \Sigma$, where $N \subseteq \mathbb{Z}^d$ is a finite set, called the *support* of $p$. A pattern $p$ *appears* in another pattern $p'$ if there exists $v \in \mathbb{Z}^d$ such that $\forall x \in N, p(x) = p'(x + v)$. We write then $p \sqsubseteq p'$. Informally, a configuration (or tiling) is a coloring of $\mathbb{Z}^d$ with elements of $\Sigma$. A *subshift* is a closed, shift-invariant subset of $\Sigma^{\mathbb{Z}^d}$, the $d$-dimensional *full shift*. For a subshift $X$ we will sometimes note $\Sigma_X$ its alphabet. The full shift is a compact metric space when equipped with the distance $d(x, y) = 2^{-\min\{\|v\|_\infty \mid v \in \mathbb{Z}^d, x(v) \neq y(v)\}}$ with $\|v\|_\infty = \max_i |v_i|$.

It is well known that subshifts may also be defined via collections of forbidden patterns. Let $F$ be a collection of forbidden patterns, the subset $X_F$ of $\Sigma^{\mathbb{Z}^d}$ defined by

$$X_F = \left\{ x \in \Sigma^{\mathbb{Z}^d} \mid \forall p \in F, p \not\sqsubseteq x \right\}$$

is a subshift. Any subshift may be defined via an adequate collection of forbidden patterns. A *subshift of finite type (SFT)* is a subshift which may be defined via a finite collection of forbidden patterns. A configuration of a subshift is also called a *point* of this subshift and is said to be valid with respect to the family of forbidden patterns $F$. Remark that $F$ being finite, we can define a subshift of finite type either by a set of forbidden or authorized patterns.

*Wang tiles* are unit squares with colored edges which may not be flipped or rotated, a *tileset* is a finite set of Wang tiles. Tiles of a tileset maybe placed side by side on the $\mathbb{Z}^2$ plane only when the matching borders have the same color, thus forming a tiling of the plane. The set of all tilings by some tileset is an SFT, and conversely, any SFT may be converted into an isomorphic tileset. From a computability point of view, both models are equivalent and we will use both indiscriminately. In 3D, Wang tiles can be straightforwardly generalized to Wang cubes.

A subshift is **North-West-deterministic** if, for any position, and for any two colors placed above it and to its left, there exists at most one valid color at this position. Likewise, we call a subshift **West-deterministic** if it is the case with the colors to its left and top-left.

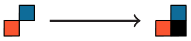


Fig. 1: NW-determinism.

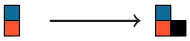


Fig. 2: W-determinism.

### 1.2 Periodicity and aperiodicity

The notion of periodicity being central in this paper, we will define it in this section.

**Definition 1 (Periodicity).** *A configuration $c$ is* periodic *of period $v$ if there exists $v \in \mathbb{Z}^d \setminus \{(0,0)\}$ such that $\forall x \in \mathbb{Z}^d, c(x) = c(x+v)$. If $c$ has no period, then it is said to be* aperiodic. *A subshift is* aperiodic *if all its points are aperiodic.*

From now on, we will focus on dimension 3 in this paper. As seen in the introduction, the lattice of vectors of periodicity may be of any dimension between 0 and $d$ and we are interested here in the case where it is 1-dimensional. In this case we can define the slope periodicity:

**Definition 2 (Slope of periodicity).** *Let $c$ be a configuration periodic along $v = (p, q, r)$. We call* slope of $v$ *the pair $\theta = (\theta_1, \theta_2)$ with $\theta_1 = \frac{p}{r}$ and $\theta_2 = \frac{p}{q}$. If all vectors of periodicity of $c$ have slope $\theta$, we say that $\theta$ is the* slope of periodicity *or* slope *of $c$. We write $Sl(X) = \{\theta \mid \exists x \in X, \theta$ is the slope of $x\}$ the set of slopes of $X$.*

### 1.3 Arithmetical hierarchy

We give now some basic definitions used in computability theory and in particular about the arithmetical hierarchy. More details may be found in [19].

Usually the arithmetical hierarchy is seen as a classification of sets according to their logical characterization. For our purpose we use an equivalent definition in terms of computability classes and Turing machines with oracles:

- $\Delta_0^0 = \Sigma_0^0 = \Pi_0^0$ is the class of recursive (or computable) problems.
- $\Sigma_n^0$ is the class of recursively enumerable (RE) problems with an oracle $\Pi_{n-1}^0$.
- $\Pi_n^0$ the complementary of $\Sigma_n^0$, or the class of co-recursively enumerable (coRE) problems with an oracle $\Sigma_{n-1}^0$.
- $\Delta_n^0 = \Sigma_n^0 \cap \Pi_n^0$ is the class of recursive (R) problems with an oracle $\Pi_{n-1}^0$.

In particular, $\Sigma_1^0$ is the class of recursively enumerable problems and $\Pi_1^0$ is the class of co-recursively enumerable problems.

## 2 Proof of Theorem 1

**Theorem 1.** *Let $R \in \Sigma_2^0 \cap \mathcal{P}((\mathbb{Q} \cup \{\infty\})^2)$, there exists a 3D SFT $X$ such that $Sl(X) = R$.*

*Proof.* Let $M$ be a Turing machine accepting $R$ with an oracle $O \in \Pi_1^0$. One can suppose that this machine takes as input 3 integers $(p, q, r) \in \mathbb{N}^3$ and that its output depends only on $\theta_1 = \frac{p}{r}$ and $\theta_2 = \frac{p}{q}$.

We only explain the case $0 < r < q < p$, the others are symmetric or quite similar and it suffices to take the disjoint union of the obtained SFTs to get the full characterization.

Let us construct a 3D SFT $X_M$ that has a periodic configuration along $\theta$ if and only if $\theta \in R$. To do so, $X_M$ will be such that "good" configurations (i.e. valid and 1-periodic) are formed of large cubes, shifted with an offset to allow periodicity along some slope. Then we encode $M$ inside all the cubes, and give to it the slope as input. The machine halts (i.e the slope is in $R$) implies that the cubes are of finite size. Which means that the configuration is 1-periodic only when the slope actually corresponds to some element of $R$.

For that, we separate the construction in different layers, in order to make it clearer. We define $X_M = B \times B' \times B'' \times C \times W \times P \times S \times T_O \times T_M \times A$, with the following layers:

- $B$ creates $(yz)$ black planes, separated by an aperiodic tiling.
- $B'$ and $B''$ create planes orthogonal to the ones of $B$, forming rectangular parallelepipeds.
- $C$ forces the parallelepipeds to become cubes.
- $W$ forces the aperiodicity vector to appear between cubes, and writes the input of the Turing machine in the cubes.
- $P$ reduces the size of the input.
- $S$ synchronizes the aperiodic backgrounds of the cubes.
- $T_M$ encodes the "$\Sigma^0_2$ " Turing machine $M$ in the cubes.
- $T_O$ encodes the $\Pi^0_1$ oracle $O$ that is used by $M$.
- $A$ ensures the existence of configurations with a unique direction of periodicity.

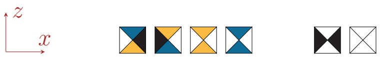
**Aperiodic background** We first need an aperiodic background in order to ensure that there is no other directions of periodicity that the one we create later on. We even make a 3D West-deterministic aperiodic background since some layers will need that deterministic property to work. For that we cross two 2D West-deterministic aperiodic backgrounds: the set of aperiodic cubes are the sets of cubes of the form shown in Figure 3. We also impose that all parallel planes are identical (Figure 4). The 2D aperiodic tiling is from Kari [11], which is a NW-aperiodic tiling, and can be easily transformed into a West-deterministic SFT. With such a superposition, one can easily show that the resulting 3D tiling is aperiodic.

**Layer** $B$ The first layer is made with two types of cubes: a white cube (□), which is a meta-cube that corresponds to any cube of the aperiodic background and a black cube (■) which will serve to break the aperiodicity brought by the white cubes. The rules of this layer are:

- In coordinates $z+1$ and $z-1$ of ■, only a ■ can appear.
- In coordinates $y+1$ and $y-1$ of ■, only ■ can appear.
- In coordinates $x+1$ and $x-1$ of ■, only □ can appear.

With only this layer, the valid periodic configurations are thick aperiodic $(yz)$ planes separated by infinite black $(yz)$ planes. At this stage, there may be several aperiodic planes "inside" a period.

**Layer** $B'$ For every Wang cube of this layer, we impose that the cube at $y+1$ is the same. so we can describe the layer $B'$ by a set of 2D Wang tiles in the $(xz)$ plane, duplicated on the $y$ axis. The tiles are:



The first four can only be superimposed with black tiles of layer $B$ and the last two only with white ones.

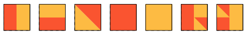
With layers $B$ and $B'$ the periodic configurations are formed of infinite planes along $(yz)$ linked by infinite $(xy)$ strips infinite along $y$, see Figure 5.

**Layer** $B''$ This layer is identical to $B'$ but tiles are duplicated along the $z$ axis. It creates portions of infinite planes along the $z$ axis, also delimited by the black planes the layer $B$.

With these three layers, periodic configurations are formed of parallelepipeds delimited by black cubes and with portions of aperiodic background inside them.

**Layer** $C$ This layer forces the parallelepipeds to be cubes, by forcing rectangles of $(xz)$ to be squares, and same for rectangles of $(xy)$.

Like the $B$ layer this layer is created by duplicating 2D Wang tiles along $y$ axis for rectangles of $(xz)$ and $z$ axis for rectangles of $(xy)$:



These tiles are superimposed once on the $B'$ tiles with rules on the $(xz)$ plane and on the $B''$ tiles with rules on the $(xy)$ plane. The superimpositions allowed are the following:

- can only be on and .
- can only be on
- can only be on and only on
- , and can only be on white tiles

Figure 6 shows how this layer forces squares to appear.

**Layer** $W$ This layer uses signals to synchronize the offsets of different cubes, and to force cubes to have the same size. In order to visualize the different offsets you can refer to Figure 7. The construction is done in several parts.

The first one forces the offsets along $y$ (denoted by $r$) to be the same in each cube. Here again, everything is duplicated along $z$. It creates signals (see Figure 8), that have to correspond with the extension of the neighboring

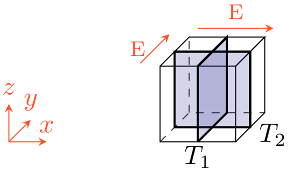


Fig. 3: $T_1$, $T_2$ are tiles of the 2D aperiodic tileset and form a Wang cube of the 3D aperiodic tileset. The "E" shows the east direction of the two planes.

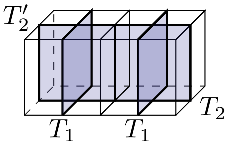


Fig. 4: Duplication of parallel backgrounds.

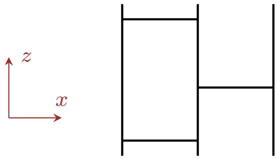


Fig. 5: Projection on the $(xz)$ plane of a valid configuration with layers $B$ and $B'$.

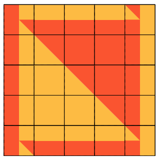


Fig. 6: Valid tiling with rules of layer $C$.

cubes. This also writes the number $r$ in unary in the border of each cube. This number will be used by the Turing machine encoded later in the tiling. The second part is identical to the first one, but on the $(xz)$ plane and rotated 90°. It forces the offset along $z$ (denoted by $q$) to be the same everywhere. Finally, the cubes are forced to be of same size. For that we add the two signals shown on Figure 9, which have to link a corner to the extension of a square. It has the effect to force each square (and hence each cube) to be of same size as its neighbors.

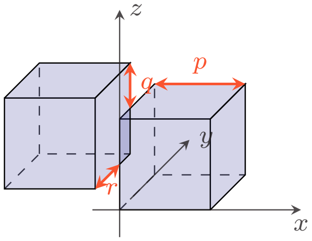


Fig. 7: Names of the offsets.

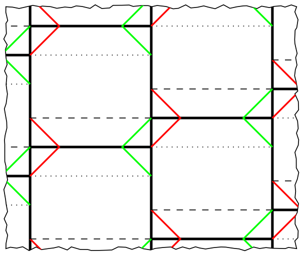


Fig. 8: Signals making the offsets identical.

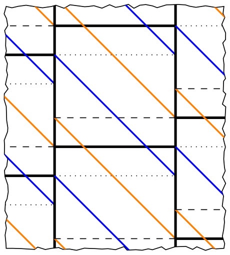
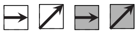


Fig. 9: Signals making the cubes of same size.

**Layer** $P$ This layer reduces the size of the input, in order to allow us to construct valid configurations as large as we want for the same input $(p', q', r')$. Starting from an unary input $(p, q, r)$, this layer writes into cubes what the input of the Turing machine will be: $(p', q', r')$, with $(p, q, r) = 2^k(p', q', r')$, and $\gcd(p', q', r')$ not divisible by 2.

For that, we use a transducer to convert numbers in binary. Such a transducer can be easily encoded into tilings (see for example [12] or [8]). Then it only remains to remove the final 0's they have in common, which can easily be done through local rules.

**Layer** $S$ Aperiodic backgrounds of different "slices" may be different ("slices" are the thick planes in the $(yz)$ plane). They must be synchronized in order to ensure the existence of a periodic configuration along $(p, q, r)$. To do this synchronization in 2D, we use the following arrow tiles:



with the following rules:

- To the left of ◩ (layer B') there is ⬈ and bottom left neighbor of ⬈ is ⬈ or a ➡.
- Bottom left tile of a square is ➡. On the right of ➡ there is only ➡ or ⧄.
- On the right, left and bottom of ⧄ there is only ⧄ or ⬈.
- The *breaking lines* can only have ⇥ on them.

We obtain the tiling shown on Figure 10. If we impose that the background is the same at the beginning and at the end of the arrow with gray background, its West-periodicity ensures that it is repeated along the global periodicity vector.

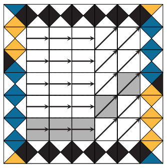


Fig. 10: Transmission of a 2D background.

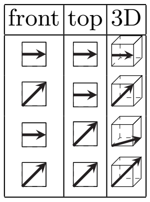
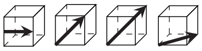


Fig. 11: Rules for layer S (transmission in 3D).

We now use this 2D construction to build the 3D transmission of the background in $X_M$. We create two layers of 2D arrows. One in the $(xy)$ plane, that are repeated along $z$ (*front* arrows) and the other in $(xz)$ and repeated along $y$ (*top* arrow). We then create our real layer using these two 2D layers, with 3D arrows:



The superimposition of two 2D arrows gives directly which 3D arrow is on each tile (see Figure 11). Like in 2D we impose that the background is the same at the beginning and at the end of the gray arrow. Thanks to the double West-periodicity of the background, this ensures that the background has a periodicity vector of $(p, q, r)$ in valid configurations.

**Layer** $T_M$ This layer encodes the Turing machine $M$ in the tiling. In our definition of the arithmetical hierarchy, the machine $M$ being in $\Sigma_2^0$, it has access to a $\Pi_1^0$ oracle. This oracle will be represented by a tape $R_O$ filled with zeros and ones, such that position $i$ of $R_O$ is a 1 if and only if the oracle $O$ accepts $i$ (i.e. the Turing machine of index $i$ runs indefinitely). For the moment, we will encode $M$ with an additional read-only arbitrary tape, and the next layer will ensure that the content of the tape is valid for $O$. This additional tape is supposed to be infinite, but since $M$ has to halt in the periodic configurations, we can restrict the construction to a finite but arbitrarily large portion of it. The tape is a line along axis $y$ duplicated along axes $x$ and $z$ (see Figure 12). We add the two rules:

1. Inside a cube, a number at position $x$ is the same as the number at position $x-1$ and a number at position $z$ must be equal to the number at position $z-1$.
2. The first line of the tape is transmitted through black cubes like the aperiodic background.

The first rule duplicates the first line everywhere inside a cube, and the second one ensures that the same $R_O$ tape is duplicated along the direction of periodicity.

Then, we encode $M$ in the $(xz)$ plane using the usual encoding of Turing machines in tilings. Let us say that the time is along the $z$ axis and the working tape along $x$. In order to access the entire $R_O$ tape, we add the spacial dimension $y$ to the TM encoding: while doing a transition, the machine can move its head along the $y$ axis and read the value of the $R_O$ tape in it; rule 1 above prevents $M$ to modify this extra tape.

Note that because $O$ is a $\Pi_1^0$ oracle, it can only ensure that the 1s of $R_O$ are correct. $M$ has to check that the 0s are correct. But checking the 0s, i.e. checking if a TM halts in a computation does not add any complexity to the problem, because we are only interested in the periodic configurations, where $M$ actually halts (and so do all its checks).

**Layer** $T_O$ This layer is the core of this proof, and it is where 3D actually comes to play: in the thick aperiodic $(yz)$ planes we will compute the $\Pi_1^0$ oracle by encoding an infinite computation that checks simultaneously all possible inputs of $M_O$, the Turing machine checking the $\Pi_1^0$ oracle $O$ ($M_O$ halts if and only if there is a wrong 1 in the portion of $R_O$ written in all the cubes).

The key idea of this layer is the use of the previously constructed cubes as *macro-tiles* in order to encode computations of $M_O$. Each cube will thus represent one tile and the thick aperiodic planes will contain, more sparsely, another 2D tiling. See Figure 14 to see how the cubes store this macro-tileset. For this macro-tileset, we may use a construction of Myers [15] which modifies Robinson's aperiodic tileset in order to synchronize the input tapes on all of the partial computations. So each of our cubes contains/represents one tile of Myer's tileset, and the thick aperiodic planes thus also contain a Myers tiling checking some input that for the moment is not synchronized with the oracle written inside these cubes.

We now have a valid macro-tiling for the cubes if and only if the machine $M_O$ never halts on $R_O$.

The one remaining thing to do is to explain the $R_O$ tape that $M_O$ accesses is synchronized with the $R_O$ which is stored inside the large cubes. We add to the set of numbered tiles the same tiles, but in red, representing the head of the $M_O$ Turing machine on the tape $R_O$. We impose that there is only one red number in every large cube (see Figure 13).

The red tile of a cube must be synchronized with the cell of the oracle $R_O$ currently contained in the Myers tile. Every time a new partial computation is started in the macro-tiles, the red tile must be placed at the beginning of

$R_O$, whenever the macro-tile moves the head to the right/left, the red tile must also be moved, if the red tile reaches the border of the cube, in which case it reaches a special state of non-synchronization, since the beginning has already been synchronized.
To do that, we must allow two new transitions. These new transitions do not change the state of the working tape, thus we only move to the next time along $z$. But in the new position, the red-marked cell in the large cube must have changed. To do this, we again use signals between the bottom-cube (previous state), the cell doing the transition and the upper-cube (next state), see Figure 15.

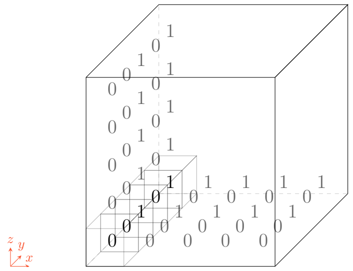


Fig. 12: Tape $R_O$ of the oracle in a cube.

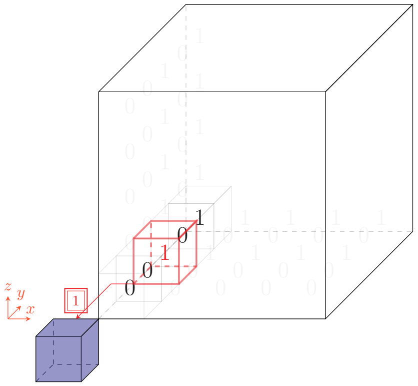


Fig. 13: The red tiles on $R_O$ and the transmission of its value.

**Layer** $A$ This last layer forces the apparition of 1-periodic configurations. Using two cubes (🟥 and 🟦), superimposed only with □ and borders of big cubes. We impose that blue/red neighbors have the same color. It is easy to see that the color is uniform inside a cube and spread to two opposite corners of cubes. Thus all the cubes along $(p, q, r)$ have the same color and there exists at least one 1-periodic configuration.

Now we prove that this construction does what we claim, finishing the proof of Theorem 1.

### 2.1 Every slope $\theta$ of $X_M$ is accepted by $M$.

Let $\theta = (\frac{p}{r}, \frac{q}{r})$ be a slope, by construction every periodic configuration along this slope is formed with cubes of the same size $p$, shifted with the same offset $(q, r)$. Every cube has the same content, which corresponds to an execution of $M$. Cubes being of finite size, every execution is a halting execution of $M$. Let's take $(p, q, r) = 2^k(p', q', r')$, with $p', q', r'$ odds. Thanks to the layer P, the input of $M$ is $(p', q', r')$, then $M$ accepts $(\frac{p'}{r'}, \frac{q'}{r'}) = (\frac{p}{r}, \frac{q}{r}) = \theta$.

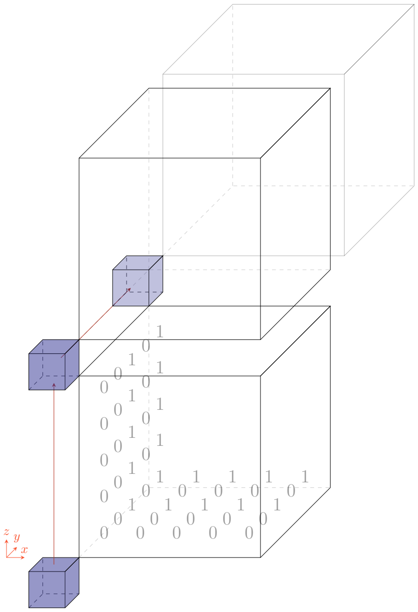


Fig. 14: Meta-tiles of large cubes, with the adjacency rules represented by the arrows. Myers' tiles are placed in the darker cubes, in the $(yz)$ plane.

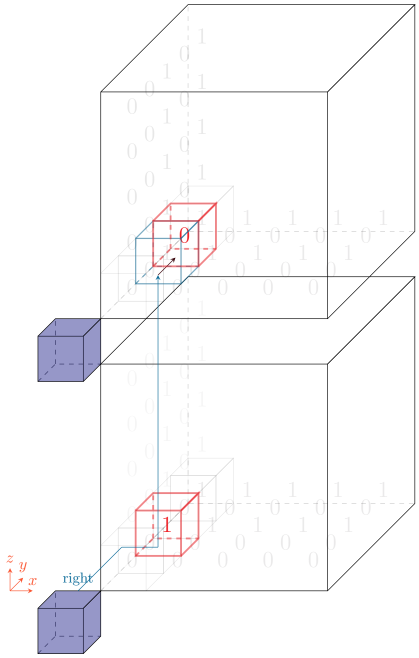


Fig. 15: Moving the red cube when the head of $M_O$ moves.

### 2.2 Accepting inputs of $M$ are slopes of $X_M$.

If $M$ accepts the input $(p, q, r)$, there exists a time $t$ and a space $a$ on the working tape, $b$ on the oracle tape, in which the machine $M$ halts. Then, the cube of size $m = 2^{\lceil \log t \rceil} p \geq a, b, t$ can contain the computation of $M$. The configuration formed by cubes of size $m$ and of offset $(n, o) = 2^{\lceil \log t \rceil}(q, r)$ is of slope $(\frac{m}{o}, \frac{n}{o}) = (\frac{p}{r}, \frac{q}{r})$. □

## 3 Open Problems

The problem of deciding if all configurations of a 2D SFT are aperiodic is well-known to be $\Pi_1^0$. Proving the other direction of the conjecture would require the study of a very similar problem: deciding if there **exists** a periodic configuration in a given SFT. Four our purpose, one needs to prove that the problem of the existence of an aperiodic configuration is $\Pi_1^0$ or $\Sigma_2^0$. However, we aren't aware of any study of this problem, not even a simpler bound like $\Pi_2^0$. Our quick look at it suggests that this could be a very challenging problem to tackle. Yet, it seems interesting by itself, as it would likely lead to a better understanding of periodicity and aperiodicity in SFTs.

## Acknowledgements

The authors would like to thank anonymous reviewers who pointed out a mistake in a previous version of the paper.

This work was supported by grant TARMAC ANR 12 BS02 007 01.

## References

1. N. Aubrun and M. Sablik. Simulation of effective subshifts by two-dimensional subshifts of finite type. *Acta Applicandae Mathematicae*, 2013.
2. R. Berger. *The Undecidability of the Domino Problem.* PhD thesis, Harvard University, 1964.
3. R. Berger. *The Undecidability of the Domino Problem.* Number 66 in Memoirs of the American Mathematical Society. The American Mathematical Society, 1966.
4. K. Culik II and J. Kari. An aperiodic set of Wang cubes. *Journal of Universal Computer Science*, 1(10):675–686, 1995.
5. B. Durand, A. Romashchenko, and A. Shen. Fixed-point tile sets and their applications. *Journal of Computer and System Sciences*, 78(3):731–764, May 2012.
6. Y. Gurevich and I. Koryakov. Remarks on Berger's paper on the domino problem. *Siberian Math. Journal*, pages 319–320, 1972.
7. M. Hochman and T. Meyerovitch. A characterization of the entropies of multidimensional shifts of finite type. *Annals of Mathematics*, 171(3):2011–2038, May 2010.
8. E. Jeandel and M. Rao. An aperiodic set of 11 wang tiles. *CoRR*, abs/1506.06492, 2015.
9. E. Jeandel and P. Vanier. Slopes of tilings. In J. Kari, editor, *JAC*, pages 145–155. Turku Center for Computer Science, 2010.
10. E. Jeandel and P. Vanier. Characterizations of periods of multi-dimensional shifts. *Ergodic Theory and Dynamical Systems*, 35:431–460, 4 2015.
11. J. Kari. The Nilpotency Problem of One-Dimensional Cellular Automata. *SIAM Journal on Computing*, 21(3):571–586, 1992.
12. J. Kari. A small aperiodic set of Wang tiles. *Discrete Mathematics*, 160:259–264, 1996.
13. D. A. Lind and B. Marcus. *An Introduction to Symbolic Dynamics and Coding.* Cambridge University Press, New York, NY, USA, 1995.
14. T. Meyerovitch. Growth-type invariants for $\mathbb{Z}^d$ subshifts of finite type and arithmetical classes of real numbers. *Inventiones Mathematicae*, 184(3), 2010.
15. D. Myers. Non Recursive Tilings of the Plane II. *Journal of Symbolic Logic*, 39(2):286–294, June 1974.
16. N. Ollinger. Two-by-Two Substitution Systems and the Undecidability of the Domino Problem. In *CiE 2008*, number 5028 in Lecture Notes in Computer Science, pages 476–485, 2008.
17. V. Poupet. Yet Another Aperiodic Tile Set . In *Journées Automates Cellulaires (JAC)*, TUCS, pages 191–202, 2010.
18. R. M. Robinson. Undecidability and Nonperiodicity for Tilings of the Plane. *Inventiones Mathematicae*, 12(3):177–209, 1971.
19. H. Rogers, Jr. *Theory of Recursive Functions and Effective Computability.* MIT Press, Cambridge, MA, USA, 1987.
20. H. Wang. Proving Theorems by Pattern Recognition I. *Communications of the ACM*, 3(4):220–234, Apr. 1960.