

Block gluing intensity of bidimensional SFT: computability of the entropy and periodic points

Silvère Gangloff, Mathieu Sablik

June 7, 2017

Abstract

It is possible to define mixing properties for subshifts according to the intensity which allows to concatenate two rectangular blocks. We study the interplay between this intensity and computational properties. In particular we prove that there exists linearly block gluing subshift of finite type which are aperiodic and that all right-recursively enumerable positive number can be realized as entropy of linearly block gluing \mathbb{Z}^2 -subshift of finite type. Like linearly block gluing imply transitivity, this last point answer a question asked in [HM10] about the characterization of the entropy of transitive subshift of finite type.

1 Introduction

A subshift of finite type (or SFT) is a dynamical version of tilings defined by local rules. Given a finite set of forbidden pattern, the SFT associated is the set of coloring of \mathbb{Z}^d by a finite alphabet where no forbidden patterns appear. In dimension one, SFTs subshifts are quite well understood, as sets of biinfinite runs on finite automata [LM95]. In higher dimension they become more complex and almost all properties become undecidable. The first known is the domino problem: given a finite set of forbidden patterns, it is undecidable to say if the SFT associated is empty. The proof of this undecidability is related to a dynamical property: the existence of periodic orbit.

In fact, once the use of computability tools was accepted, it has appeared possible to characterize many dynamical concepts, the first of which being entropy [HM10]. This interplay between dynamical property and their computability can be find in a lot of recent works: characterization of subaction [Hoc09, AS13, DRS12], measure of the computationnally simplest configurations with Medvedev degrees [Sim11] and sets of Turing degrees [JV13], characterization of sets of periods in terms of complexity theory [JV14]... The proofs of these results, while sometimes technically involved, follow a common outline: show that the system is “rich enough” to simulate any Turing Machine and the undecidability comes from the halting problem. The importance of computability considerations in these models has been clearly established for decades now. A new direction is to see if dynamical properties can prevent to embed universal computing.

Mixing properties seems to be properties which simplify global behavior of multidimensional SFTs. In the case of the characterization of the entropy, a famous result states that the set of entropies of multi-dimensional SFTs is exactly the set of real numbers which are right-recursively enumerable (also called Π_1^0 -computable number in reference to the arithmetical hierarchy) [HM10]. When the SFT is strongly irreducible the entropy becomes computable. This means that there exists an algorithm which takes an integer n as input and gives back an approximation of the entropy up to 2^{-n} . Only some partial realization results are known [PS14] without full characterization. Thus the dynamics of the system involve restrictions on the power of realization of the entropy.

In [PS14] the authors study SFT which are block gluing: that is to say there exists a constant c such that the pattern obtained by the concatenation of two rectangular patterns in the langage separated by a distance c is also in the langage. We propose to study intensity of this mixing property, as done in [SG16] considering effective subshifts, for subshifts of finite type which allow rectangular patterns in the langage to be concatenated into a new patterns in the langage, given

certain gaps between them. All are defined in terms of an auxiliary gap function $f : \mathbb{N} \rightarrow \mathbb{N}$, which gives the minimum required gap length as a function of the lengths of the blocks on either side.

In this article, we are interested on two dynamical properties, existence of periodic orbits and realization of possible entropies, according to the intensity f of two dimensional f -block gluing SFT. We observe two regimes:

- **Strong block gluing:**

- if $f \in o(\log(n))$ then the set of periodic orbits of f -block gluing SFT is dense (Proposition 6) and the langage is decidable.
- if $f(n) \leq \frac{n^{1/\log_2(5)}}{\log(n)^{1+\epsilon}}$ the entropy of f -block gluing SFT is computable (Proposition 10).

- **Low block gluing:**

- if $f \in O(n)$ then there exists aperiodic f -block gluing SFT (Theorem 1).
- if $f \in O(n)$ then there exists f -block gluing SFT with non decidable langage (Proposition 7).
- if $f \in O(n)$, then the set of entropies of f -block gluing SFT is the set of right recursively enumerable real (Theorem 4). Since $O(n)$ -block gluing SFT are transitive, this result characterizes the set of possible entropy of transitive SFT which is an open question of [HM10].

The challenging question initiated here is to explore the limit of computability for this property.

The results in the strong block gluing regime extend to a larger class of intensities some known result about the density of the set of periodic orbits under block gluing condition (meaning finite block gluing considering intensity) obtained in [PS14].

The low block gluing regime is more interesting because it leads to more complex and structured constructions. The proof of Theorem 1 rely on the 'net gluing' property (notion introduced in this article) of the Robinson subshift and on transformations on subshift of finite type over some fixed alphabet that misshape the configuration of a subshift and permit to transform a net gluing subshift into a block gluing one. An important point is that the entropy of the image of a subshift by the transformation is a function of the entropy of this subshift which has a closed form.

The proof of Theorem 4 rely on the construction of [HM10], using the Robinson subshift to implement machines in computation zones defined by this subshift that control the frequency of some 'frequency bits' 0,1 that are identified in columns, and adding random bits 1,1' over the 1 symbols that generate the entropy. The two obstacles to the transitivity property in this construction are the identification of the frequency bits in columns, and that the behaviors occurring in infinite computation zones are very specific to these zones. We solve these problems identifying the frequency bits inside every computation zone to solve the first problem, and simulating machines having the 'bad behaviors' occurring in infinite computation zones in every finite one, aside machines that have the 'good behavior' to solve the second problem.

From considerations on the computability properties of subshifts, arise some 'natural' tools and principles of the organization, stocking, and exchanges of information of various 'objects' observable in the system. In the construction presented in this article, structures extracted from the Robinson subshift permit at the same time to attribute areas for computation and control agents (Turing machines), as this is done in [AS13], and allow various signals to propagate, so that the computation agents synchronize or communicate, without interfering between each other. These agents are organized as a hierarchy in [AS13] in the strong sense that the results of Turing machines at some level of the hierarchy will be transmitted to the Turing machine immediately above in the hierarchy for its proper computation (in our construction this hierarchy is only geographic).

This is noteworthy that this type of 'computation hierarchy' is present in neuroscience models for the visual system for instance [Ser14], where neurons are organized as a hierarchy which bottom is the set of sensitive receptors and the top is constituted with the highest cortical areas. In this

article, there is a strong analogy of this type between some objects arising in the construction for the proof of Theorem 4 and very simple cellular biology objects. This analogy is present in the words we use to describe the construction, that were useful to visualize and understand the construction. Moreover, it appears that the quest for block gluing property resulted in a natural centralization and fixation of information (what we called DNA) in the centers of the cells (we called them nuclei), with which the computation agents (the machines) communicate using error signals to 'have access' to this information telling it what behavior it should have. This leads to the intuition that this type of results could maybe read in a phenomenological way for the understanding of basic principle of 'information processing systems' (in particular cellular biology ones, at a local level), as 'transitivity implies centralization of the information'.

The article is organized as follows :

- In Section 2 we recall symbolic dynamics general definitions, define some "block gluing" notions, and recall the Robinson subshift definition and properties.
- In Section 3, we explore the property of periodic orbits.
- In Section 4, we explore the property of entropy.

2 Notion of block gluing with gap function

In this section we recall some definitions on symbolic dynamics and we introduce the notion of block gluing with intensity function. Then we give some examples of subshifts of finite type which are block gluing for various intensity functions.

2.1 Subshifts as dynamical systems

2.1.1 Subshifts and patterns

Let \mathcal{A} be a finite set (the alphabet). A **configuration** x is an element of $\mathcal{A}^{\mathbb{Z}^2}$. In this article we focus on two dimensional configurations but all the following definitions can be generalized to \mathbb{Z}^d , $d \geq 2$. The space $\mathcal{A}^{\mathbb{Z}^2}$ is endowed by the product topology derived from the discrete topology on \mathcal{A} . For this topology, $\mathcal{A}^{\mathbb{Z}^2}$ is a compact metric space on which \mathbb{Z}^2 acts continually by translation via the **shift** map, denoted σ , which is defined for all $\mathbf{i} \in \mathbb{Z}^2$ by:

$$\begin{aligned} \sigma^{\mathbf{i}} : \mathcal{A}^{\mathbb{Z}^2} &\longrightarrow \mathcal{A}^{\mathbb{Z}^2} \\ x &\longmapsto \sigma^{\mathbf{i}}(x) \quad \text{such that } \forall \mathbf{u} \in \mathbb{Z}^2, \sigma^{\mathbf{i}}(x)_{\mathbf{u}} = x_{\mathbf{i}+\mathbf{u}} \end{aligned}$$

Let \mathbb{U} be a finite subset of \mathbb{Z}^2 . Denote $x_{\mathbb{U}}$ the **restriction** of $x \in \mathcal{A}^{\mathbb{Z}^2}$ to \mathbb{U} . A **pattern** p on **support** \mathbb{U} , denoted $\text{supp}(p)$, is an element of $\mathcal{A}^{\mathbb{U}}$. Define $\mathbb{U}_n = \llbracket 0; n-1 \rrbracket^2$ the **elementary support** of size $n \in \mathbb{N}$. A pattern on support \mathbb{U}_n is a **n-block**. As well, a pattern with support $\llbracket 0; n-1 \rrbracket \times \llbracket 0; m-1 \rrbracket$ is a **n × m-rectangle**. A pattern p having support \mathbb{U} **appears** at position $\mathbf{i} \in \mathbb{Z}^2$ in a configuration $x \in \mathcal{A}^{\mathbb{Z}^2}$ if for all $\mathbf{j} \in \mathbb{U}$, $p_{\mathbf{j}} = x_{\mathbf{i}+\mathbf{j}}$, denote $p \sqsubset x$. A pattern p on support \mathbb{U} is a **sub-pattern** of a pattern q on support \mathbb{V} when $\mathbb{U} \subset \mathbb{V}$ and $q_{\mathbb{U}} = p$.

A **subshift** X is a closed subset of $\mathcal{A}^{\mathbb{Z}^2}$ which is invariant under the action of the shift, meaning $\sigma(X) \subset X$. The couple (X, σ) is a dynamical system. Any subshift X can be defined by a set of **forbidden patterns**, as the set of configurations where no element of this set appears. Formally there exists \mathcal{F} a set of patterns such that :

$$X = X_{\mathcal{F}} := \left\{ x \in \mathcal{A}^{\mathbb{Z}^2} : \text{for all } p \in \mathcal{F}, p \not\sqsubset x \right\}.$$

If the subshift can be defined by a finite set of forbidden patterns, it is called a **subshift of finite type** (SFT for short). The order of a SFT is the smallest r such that it can be defined by forbidden r -blocks.

A configuration $x \in \mathcal{A}^{\mathbb{Z}^2}$ is **periodic** if there exists $m, n > 0$ such that $\sigma^{(m,0)}(x) = \sigma^{(0,n)}(x) = x$. A subshift is **aperiodic** when none of its configurations is periodic.

A pattern **appears** in a subshift X if there is a configuration of X in which it appears. The set of patterns which appear in X is called the **language** of X , denoted $\mathcal{L}(X)$. Denote $\beta_n(X)$ the set of n -blocks that appears in X .

In this article the construction of subshifts is obtained on an alphabet \mathcal{A} which is a product of alphabets $\mathcal{A} = \mathcal{A}_1 \times \dots \times \mathcal{A}_k$. We call informally the i^{th} layer of this subshift the space of the projections of a configuration written on the i^{th} alphabet \mathcal{A}_i .

2.1.2 Morphisms

A **morphism** between two subshifts X and Y on alphabets \mathcal{A}_X and \mathcal{A}_Y is a continuous map $\varphi : X \rightarrow Y$ such that $\varphi \circ \sigma = \sigma \circ \varphi$. Equivalently by Hedlund's Theorem [Hed69], φ can be defined with a **local function** $\bar{\varphi} : \mathcal{A}_X^{\llbracket -r, r \rrbracket^2} \rightarrow \mathcal{A}_Y$ of **radius** $r \in \mathbb{N}$ by

$$\varphi(x)_{\mathbf{i}} = \bar{\varphi}(x_{\mathbf{i} + \llbracket -r, r \rrbracket^2}) \quad \text{for all } x \in X, \text{ and } \mathbf{i} \in \mathbb{Z}^2.$$

A **factor** is a morphism which is onto, and it is a **conjugacy** if it is invertible, the inverse map being also a morphism in this case. Two subshifts are **conjugated** if there exists a conjugacy between them. In this case we considerate that they have the same dynamical behavior.

2.2 Block gluing notions

We present here some mixing properties on multidimensional subshifts which extend the block-gluing notion introduced in [PS15]. Instead of an uniform condition of gluing, we specify the intensity by a non decreasing function.

2.2.1 Definitions

In this section, X is a subshift on the alphabet \mathcal{A} and $f : \mathbb{N} \rightarrow \mathbb{N}$ is a non decreasing function. Denote $\|\cdot\|_{\infty}$ the norm defined by $\|\mathbf{i}\|_{\infty} = \max\{i_1, i_2\}$ for all $\mathbf{i} \in \mathbb{Z}^2$.

Definition 1. Let $n \in \mathbb{N}$, the **gluing set** in the subshift X of some n -block p relatively to some other n -block q is the set of $\mathbf{u} \in \mathbb{Z}^2$ such that there exists a configuration in X where q appears in position $(0, 0)$, and p appears in position \mathbf{u} (see Figure 1). This set is denoted $\Delta_X(p, q)$. Formally

$$\Delta_X(p, q) = \{\mathbf{u} \in \mathbb{Z}^2 : \exists x \in X \text{ such that } x_{\llbracket 0, n-1 \rrbracket^2} = q \text{ and } x_{\mathbf{u} + \llbracket 0, n-1 \rrbracket^2} = p\}$$

When the intersection of these sets for all couples of n -blocks is non empty, we denote it $\Delta_X(n)$. This set is called the **gluing set of n -blocks** in X .

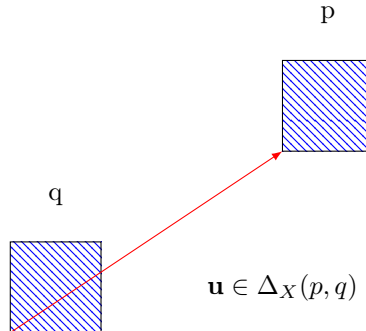


Figure 1: Illustration of Definition 1.

Definition 2. A subshift X is **f -block transitive** if for all $n \in \mathbb{N}$ one has

$$\Delta_X(n) \cap \{\mathbf{u} \in \mathbb{Z}^2 : \|\mathbf{u}\|_\infty \leq n + f(n)\} \neq \emptyset$$

Definition 3. A subshift X is **f -net gluing** if for all $n \in \mathbb{N}$ and for all n -blocks p and q , there exist some $\mathbf{u}(p, q) \in \mathbb{Z}^2$ and $\tilde{f}(p, q) \in \mathbb{N}$ such that

$$\mathbf{u}(p, q) + (n + \tilde{f}(p, q))(\mathbb{Z}^2 \setminus \{0\}) \subset \Delta_X(p, q),$$

with $\max_{p, q \in \beta_n(X)} \tilde{f}(p, q) \leq f(n)$. The function f is called the **gluing intensity**.

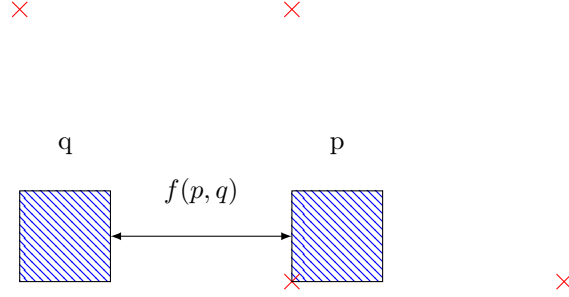


Figure 2: Illustration of Definition 3. Red crosses designate elements of the gluing set of p relatively to q in X .

Definition 4. A subshift X is **f -block gluing** if

$$\{\mathbf{u} \in \mathbb{Z}^2, \|\mathbf{u}\|_\infty \geq f(n) + n\} \subset \Delta_X(n).$$

For any function f , one has

$$f\text{-block gluing} \implies f\text{-net gluing} \implies f\text{-block transitive}$$

A subshift is said **$O(f)$ -block gluing** (resp. **$O(f)$ -net gluing**, **$O(f)$ -block transitive**) if it is g - (block gluing) (resp. g -net gluing, g -block transitive) for a function $g : \mathbb{N} \rightarrow \mathbb{N}$ such that there exists $C > 0$ such that $g(n) \leq C f(n)$ for all $n \in \mathbb{N}$. A property verified on the class of $O(f)$ -block gluing (resp. g -net gluing, g -block transitive) subshifts is **sharp** if the property is not verified for all $h \in o(f)$ (meaning that for all $\epsilon > 0$ there exists n_0 such that $h(n) \leq \epsilon f(n)$ for all $n \geq n_0$).

A subshift is **linearly block gluing** (resp. **linearly net gluing**, **linearly transitive**) if it is $O(n)$ -block gluing (resp. $O(n)$ -net gluing, $O(n)$ -block transitive).

2.2.2 Equivalent definition

The following proposition gives an equivalent definition for linear block gluing and net gluing subshifts using some exceptional values:

Proposition 1. A subshift $X \subset \mathcal{A}^{\mathbb{Z}^2}$ is linearly block gluing iff there exist a function $f \in O(n)$, $c \geq 2$ an integer and $m \in \mathbb{N}$ such that

$$\{\mathbf{u} \in \mathbb{Z}^2, \|\mathbf{u}\|_\infty \geq f(c^l + m) + c^l + m\} \subset \Delta_X(c^l + m) \quad \forall l \geq 0.$$

A similar assertion is true for net gluing.

Proof. Clearly a linear-block gluing subshift verifies this property. Reciprocally, let p and q two n -blocks, and consider $l(n) = \lceil \log_c(n - m) \rceil$, where $\lceil \cdot \rceil$ designates the smallest integer greater than ”.”. Consider p' and q' some $c^{l(n)} + m$ -blocks which restrictions on $\llbracket 0, n - 1 \rrbracket^k$ are respectively p and q . The set $\Delta_X(p', q')$ contains $\{\mathbf{u} \in \mathbb{Z}^2, \|\mathbf{u}\|_\infty \geq f(c^{l(n)} + m) + c^{l(n)} + m\}$. As a consequence, $\Delta_X(p, q)$ contains $\{\mathbf{u} \in \mathbb{Z}^2, \|\mathbf{u}\|_\infty \geq g(n) + n\}$, where $g(n) = f(c^{l(n)} + m) + c^{l(n)} - n + m$. Since $c^{l(n)} \leq c * (n + |m|)$, the function g is in $O(n)$, hence X is $O(n)$ -block gluing. \square

2.2.3 Gluing and morphisms

The following proposition shows that a factor of a block gluing (resp. net gluing) subshift is also block gluing (resp. net gluing) and precises the intensity function.

Proposition 2. *Let $\varphi : X \rightarrow Y$ be a factor between two bidimensional subshifts and $f : \mathbb{N} \rightarrow \mathbb{N}$ be a non decreasing function. If the subshift X is f -block gluing (resp. f -net gluing), then Y is g -block gluing (resp. g -net gluing) where $g : n \mapsto f(n + 2r) + 2r$.*

Proof. Let $\varphi : X \rightarrow Y$ a factor of radius r and local rule $\overline{\varphi} : \mathcal{A}_X^{\llbracket -r, r \rrbracket^2} \rightarrow \mathcal{A}_Y$.

Let p', q' be two n -blocks in the language of Y . There exist p and q two $(n + 2r)$ -blocks in the language of X such that p' and q' are respectively the image of p and q by $\overline{\varphi}$. Let $\mathbf{u} \in \Delta_X(p, q)$. There exist $x \in X$ such that $x_{\llbracket 0, n+2r-1 \rrbracket^2} = p$, and $x_{\mathbf{u} + \llbracket 0, n+2r-1 \rrbracket^2} = q$. Applying φ to $\sigma^{r(1,1)}(x)$, we obtain some $y \in Y$ such that $y_{\llbracket 0, n-1 \rrbracket^k} = p'$, and $y_{\mathbf{u} + \llbracket 0, n-1 \rrbracket^k} = q'$. We deduce that

$$\Delta_X(p, q) \subset \Delta_Y(p', q') \quad \text{so} \quad \Delta_X(n + 2r) \subset \Delta_Y(n).$$

Thus if X is f -block gluing then Y is g -block gluing where $g : n \mapsto f(n + 2r) + 2r$.

If X is f -net gluing, then the gluing set of two $(2n + r)$ -blocks p, q contains

$$\mathbf{u}(p, q) + (n + 2r + \tilde{f}(p, q))(\mathbb{Z}^2 \setminus \{(0, 0)\}),$$

such that $\tilde{f}(p, q) \leq f(n + 2r)$. Hence the gluing set of p' , image of p by $\overline{\varphi}$, relative to q' , image of q by $\overline{\varphi}$, in Z contains this set. One deduces that Y is g -net gluing where $g : n \mapsto f(n + 2r) + 2r$. \square

We deduce that the classes of subshifts defined by these properties are invariant of conjugacy under some assumption on f .

Corollary 1. *Let f be some non decreasing function. If for all $r \in \mathbb{N}$, there is a constant C such that for all $n \geq 0$, $Cf(n) \geq f(n + 2r)$ then the following classes of subshifts are invariant under conjugacy: $O(f)$ -block transitive, $O(f)$ -net gluing, $O(f)$ -block gluing, sharp $O(f)$ -net gluing and sharp $O(f)$ -block gluing subshifts.*

In particular it is verified when f is constant or $n \mapsto n^k$ with $k > 0$ or $n \mapsto e^n$ or $n \mapsto \log(n)$.

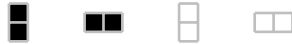
2.3 Some examples

We use the words **separated by distance k** , or **being glued at distance k** for two blocks p, q with support \mathbb{U}, \mathbb{V} , when $\max_{\mathbf{u} \in \mathbb{U}} \min_{\mathbf{v} \in \mathbb{V}} \|\mathbf{u} - \mathbf{v}\|_\infty \geq k$. This means that there are at least k column or at least k lines between the two blocks.

2.3.1 First examples

We present here some examples of block gluing SFT.

Example 1. Consider the SFT X_{chess} defined by the following set of forbidden patterns:



This subshift has two configurations (see Figure 3 for an example), which are periodic configurations. It is 1-net gluing, but not block gluing: the gluing set of the pattern \blacksquare relatively to itself is

$$\Delta_{X_{\text{chess}}}(\blacksquare, \blacksquare) = 2\mathbb{Z}^2 \setminus \{(0, 0)\} \cup (2\mathbb{Z}^2 + (1, 1))$$

Example 2. Consider the SFT X_{even} defined by the following set of forbidden patterns:



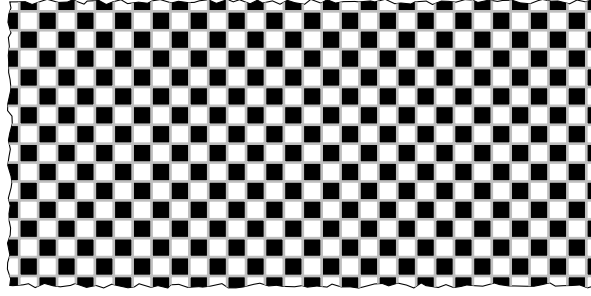


Figure 3: An example of configuration of X_{Chess} .

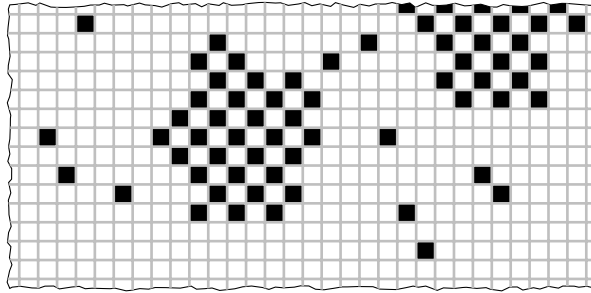


Figure 4: An example of configuration of X_{Even} .

An example of configuration in this subshift is given in Figure 4. This subshift is 1-block gluing since two blocks in its language can be glued with distance 1, filling the configuration with \square symbols.

Example 3. Consider the SFT X_{Linear} defined by the following set of forbidden patterns:



The local rules imply that if a configuration contains the pattern $\square \blacksquare^n \square$ then it contains $* \square \blacksquare^{n-2} \square *$ just above, where $*$ $\in \{\square, \blacksquare\}$. Thus a configuration of X_{Linear} can be seen as triangles of symbols \blacksquare on a background of \square symbols (an example of configuration is given on Figure 5).

This subshift is sharp linearly block gluing. Indeed consider two n -blocks in its language separated horizontally or vertically by $2n$ cells. They contain pieces of triangles that we complete with the smallest triangle possible, the other symbols of the configuration being all \square symbols. The worst case for gluing two n -blocks is when the blocks are filled with the symbol \blacksquare . In this case we can complete each of the two blocks by a triangle which base is constituted by \blacksquare^{3n} . Hence every couple of blocks can be glued horizontally and vertically with linear distance. To prove that X_{Linear} is not f -block gluing with $f(n) \in o(n)$, we consider the rectangle

$$\square \blacksquare^n \square$$

that we would like to glue above itself. To do that we need to separate the two copies of this pattern by about $\lceil \frac{n}{2} \rceil$ cells.

2.3.2 Linearly net gluing subshifts given by substitutions

Let \mathcal{A} be a finite alphabet, a **substitution** rule is a map $s : \mathcal{A} \rightarrow \mathcal{A}^{\mathbb{U}_m}$, for some $m \geq 1$. This function can be extended naturally on blocks in view to iterate it. The subshift X_s associated to

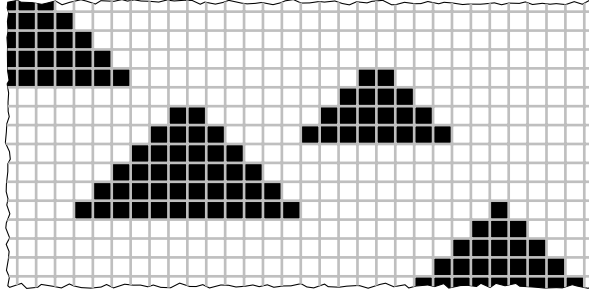


Figure 5: An example of configuration of X_{Linear} .

this substitution is the set of configurations such that any pattern appearing in it appears as a sub-pattern of some $s^n(a)$ with $n \geq 0$ and $a \in \mathcal{A}$.

Consider the following substitution s defined by

$$\square \mapsto \begin{array}{|c|c|} \hline \square & \square \\ \hline \blacksquare & \square \\ \hline \end{array} \quad \blacksquare \mapsto \begin{array}{|c|c|} \hline \blacksquare & \square \\ \hline \square & \blacksquare \\ \hline \end{array}$$

where an example of configuration is given in Figure 6. Since \blacksquare appears in the position $(0,0)$ in $s(\square)$ and $s(\blacksquare)$, we deduce that for any configuration x , there exist $\mathbf{i}_1 \in \llbracket 0, 1 \rrbracket^2$ such that $x_{\mathbf{i}_1 + 2\mathbb{Z}^2} = \blacksquare$. By induction, for all $n \geq 1$, there exist $\mathbf{i}_n \in \llbracket 0, 2^n - 1 \rrbracket^2$ such that $x_{\mathbf{i}_n + 2^n\mathbb{Z}^2} = s^n(\blacksquare)$. Since every pattern of X_s appears in $s^n(\blacksquare)$ for some $n \in \mathbb{N}$, we deduce that X_s has the linear net-gluing property, using Proposition 1.

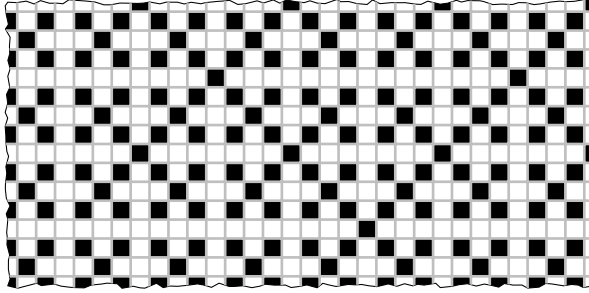


Figure 6: A part of a configuration of X_s .

This argument can be easily generalized for substitution s where there exist $i \in \mathbb{N}$, a subset $\mathcal{Z} \subset \llbracket 0, m^i - 1 \rrbracket^2$ and an invertible map $\nu : \mathcal{A} \rightarrow \mathcal{Z}$ such that $a \in \mathcal{A}$ appears in the same position $\nu(a)$ in any pattern $s^i(d)$ with $d \in \mathcal{A}$.

2.3.3 Intermediate intensities

Here we present an example of block gluing SFT with intensity strictly between linear and constant classes.

Consider the SFT X_{Log} having two layers, the first one with symbols \blacksquare and \square , and the second one the symbols :

$$\begin{array}{|c|c|} \hline 0 \\ \hline 0 & 0 \\ \hline 0 \\ \hline \end{array} \quad \begin{array}{|c|c|} \hline 1 \\ \hline 0 & 0 \\ \hline 1 \\ \hline \end{array} \quad \begin{array}{|c|c|} \hline 1 \\ \hline 1 & 0 \\ \hline 0 \\ \hline \end{array} \quad \begin{array}{|c|c|} \hline 0 \\ \hline 1 & 1 \\ \hline 1 \\ \hline \end{array} \quad \begin{array}{|c|} \hline \square \\ \hline \end{array}$$

The first four are thought as coding for the adding machine. Each one contains four symbols : the west one is the initial state of the machine, the east one the forward state, the south one the input letter, and the last symbol is the output.

The rules defining the SFT are the following ones : the following patterns are forbidden in the first layer :



The local rules imply that if a configuration contains the pattern $\square \blacksquare^n \square$ then it contains $\square \blacksquare^n \square$, $\square \blacksquare^{n+1}$, or $\square \blacksquare^{n+2}$ just above. Thus a configuration of the first layer of X_{Log} can be seen as triangular shapes of symbols \blacksquare on a background of \square symbols (an example of configuration is given in Figure 7).

The rules of the second layer are the following :

- the adding machine symbols are superimposed on black squares, the other one on blank squares.
- for two adjacent machine symbols, the symbols on the sides have to match.
- on a pattern $\blacksquare \square$, on the machine symbol over the black square, the east symbol have to be 0.
- on a pattern $\begin{smallmatrix} \blacksquare & \blacksquare \\ \blacksquare & \square \end{smallmatrix}$, the machine have a south symbol being 0 on the north west black square.

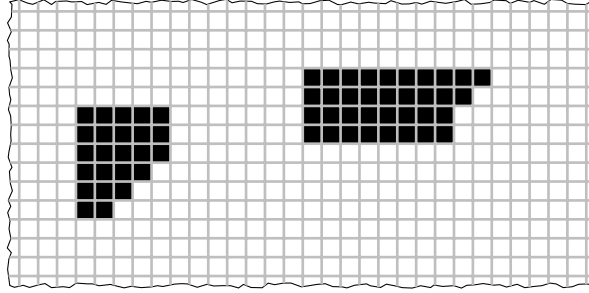


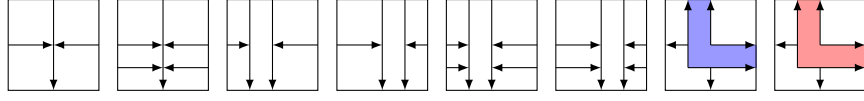
Figure 7: An example of configuration that respects the rules of the first layer of X_{Log} .

This subshift is sharp $O(\log)$ -block gluing. Indeed any two n -blocks of the language can be glued vertically with distance 1. For horizontal gluing, the worst case for gluing two n -blocks is when the two blocks are filled with black squares and the adding machine symbols on the leftmost column of the blocks are only 1 (thus maximizing the number of lines where the rectangular shape into which we complete the block have to be greater in length than the one just below). In this case, we can complete the block such that each line (from the bottom to the top) is extended from the one below, with one \blacksquare symbol on the right when the machine symbol have a 1 on its west side, and adding blank squares to obtain a rectangle. The number of columns added is less than the maximal number of bits added by the adding machine to a length n string of 0, 1 symbols in n steps, which is $O(\log(n))$. This means that two n -blocks can be glued horizontally with distance $O(\log(n))$. To see this property is sharp, consider the horizontal gluing of two $1 \times n$ rectangles of black squares, similarly as in the linear case.

2.4 A linearly net gluing version of the Robinson subshift

2.4.1 The Robinson subshift

The Robinson subshift X_{Rob} is a two dimensional SFT on the alphabet \mathcal{A}_{Rob} given by the following tiles and their rotations by $\frac{\pi}{2}$, π and $\frac{3\pi}{2}$:



The local rules are that :

1. the outgoing arrows and incoming ones correspond for two adjacent symbols.
2. in every 2×2 square there is a blue symbol and the presence of a blue symbol in position $\mathbf{u} \in \mathbb{Z}^2$ forces the presence of a blue symbol in the positions $\mathbf{u} + (0, 2)$, $\mathbf{u} - (0, 2)$, $\mathbf{u} + (2, 0)$ and $\mathbf{u} - (2, 0)$.

Robinson proved that this subshift is non empty and non-periodic. We present some properties of it in the following sections. The proofs of these properties can be found in [Rob71].

2.4.2 Supertiles

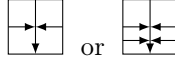
Define by induction the south west (resp. south east, north west, north east) **supertile of order** $n \in \mathbb{N}$. For $n = 0$, one has

$$St_{sw}(0) = \begin{array}{|c|c|} \hline \uparrow & \uparrow \\ \hline \leftarrow & \rightarrow \\ \hline \downarrow & \downarrow \\ \hline \end{array}, \quad St_{se}(0) = \begin{array}{|c|c|} \hline \uparrow & \uparrow \\ \hline \leftarrow & \rightarrow \\ \hline \downarrow & \downarrow \\ \hline \end{array}, \quad St_{nw}(0) = \begin{array}{|c|c|} \hline \uparrow & \uparrow \\ \hline \leftarrow & \rightarrow \\ \hline \downarrow & \downarrow \\ \hline \end{array}, \quad St_{ne}(0) = \begin{array}{|c|c|} \hline \uparrow & \uparrow \\ \hline \leftarrow & \rightarrow \\ \hline \downarrow & \downarrow \\ \hline \end{array}.$$

For $n \in \mathbb{N}$, the support of the supertile $St_{sw}(n+1)$ (resp. $St_{sw}(n+1)$, $St_{nw}(n+1)$, $St_{ne}(n+1)$) is $\mathbb{U}_{2^{n+2}-1}$. In position $\mathbf{u} = (2^{n+1} - 1, 2^{n+1} - 1)$ write

$$St_{sw}(n+1)_{\mathbf{u}} = \begin{array}{|c|c|} \hline \uparrow & \uparrow \\ \hline \leftarrow & \rightarrow \\ \hline \downarrow & \downarrow \\ \hline \end{array}, \quad St_{se}(n+1)_{\mathbf{u}} = \begin{array}{|c|c|} \hline \uparrow & \uparrow \\ \hline \leftarrow & \rightarrow \\ \hline \downarrow & \downarrow \\ \hline \end{array}, \quad St_{nw}(n+1)_{\mathbf{u}} = \begin{array}{|c|c|} \hline \uparrow & \uparrow \\ \hline \leftarrow & \rightarrow \\ \hline \downarrow & \downarrow \\ \hline \end{array}, \quad St_{ne}(n+1)_{\mathbf{u}} = \begin{array}{|c|c|} \hline \uparrow & \uparrow \\ \hline \leftarrow & \rightarrow \\ \hline \downarrow & \downarrow \\ \hline \end{array}.$$

Then complete the supertile such that the restriction to $\mathbb{U}_{2^{n+1}-1}$ is $St_{sw}(n)$, the restriction to $(2^{n+1}, 0) + \mathbb{U}_{2^{n+1}-1}$ is $St_{se}(n)$, the restriction to $(0, 2^{n+1}) + \mathbb{U}_{2^{n+1}-1}$ is $St_{nw}(n)$ and the restriction to $(2^{n+1}, 2^{n+1}) + \mathbb{U}_{2^{n+1}-1}$ is $St_{ne}(n)$. Then complete the cross uniquely by



in the south vertical arm with the first symbol when there is one incoming arrow, and the second when there are two. The other arms are completed in a similar way. For instance, Figure 8 shows the south west supertile of order two.

It is known that a supertile of order $n \in \mathbb{N}$ forces the presence of a supertile of order $n + 1$ containing it.

Let $x \in X_{\text{Rob}}$ and consider the equivalence relation \sim_x on \mathbb{Z}^2 defined by $\mathbf{i} \sim_x \mathbf{j}$ if there is a supertile in x which contains \mathbf{i} and \mathbf{j} . An **infinite order** supertile is an infinite pattern over an equivalence class of this relation. Each configuration is amongst the following types :

- (i) A unique infinite order supertile which covers \mathbb{Z}^2 .
- (ii) Two infinite order supertiles separated by a line or a column with only three-arrows symbols. In such a configuration finite supertiles of order n in an infinite order supertile are not necessary aligned with the supertiles of the same order in the other one (whereas in a configuration with a unique infinite supertile, all the supertiles with same order are aligned in a lattice).
- (iii) Four infinite order supertiles, separated by a cross, which center is a red symbol, and arms are filled with arrows symbols induced by the red one.

The supertiles of order $m \geq 1$ are repeated periodically in every supertile of order $n \geq m$ with period 2^{m+2} horizontally and vertically. This is also true inside an infinite supertile.

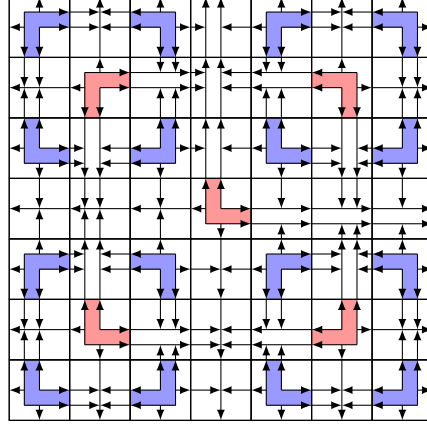


Figure 8: The south west order 2 supertile denoted $St_{sw}(2)$.

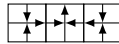
2.4.3 Alignment positioning

If a configuration of X_{Rob} has two infinite order supertiles, the two sides of this column or line which separates them are non dependent and the two infinite order supertiles of this configuration can be shifted vertically (resp. horizontally) one from each other, the configuration obtained staying an element of X_{Rob} . This is an obstacle to any mixing property since two patterns which appear at a position such that the support crosses the separating line can not be glued one to the other in X_{Rob} . We add a layer over the Robinson subshift in order to align all supertiles of same order and eliminate this phenomenon.

Here is a description of this new subshift of finite type denoted X'_{Rob} :

Symbols : The symbols of this subshift are $\mathcal{A}_{Rob} \times \{nw, ne, sw, se, \square\}$

Interaction rules : The orientation symbols (nw, ne, sw or se) are superimposed only on three arrows symbols in the Robinson layer. If a three arrows symbol is near a red or blue corner, we superimpose the orientation symbol corresponding to the orientation of the corner. This mark is transmitted to the next symbol in the direction of the arrow if the three or five arrows symbol has the same orientation. Where the pattern



occurs, on the two sides of the vertical three arrows symbol, the tiles must have complementary orientation symbols. We impose a similar rule rotating the pattern.

Global behavior: As for X_{Rob} , in an infinite supertile, supertiles of order n are repeated with period 2^{n+2} horizontally and vertically. In a type (iii) configuration, supertiles of order n are periodic in all the configuration since they are aligned by the cross. In a type (ii) configuration, around the line which separates the two infinite supertiles, it appears two toeplitz sequences which give the positions and types of supertiles on each infinite supertiles. By induction, the two toeplitz sequences have to be equal.

The consequence is that for a configuration $x \in X'_{Rob}$ for all $n \in \mathbb{N}$, there exists $\mathbf{i} \in \mathbb{Z}^2$ such that for all $\mathbf{j} \in 2^{n+2}\mathbb{Z}^2$ one has

$$\begin{aligned} \pi_{Rob}(x)_{\mathbf{i}+\mathbf{j}+\mathbf{U}_n} &= St_{sw}(n) & \pi_{Rob}(x)_{\mathbf{i}+\mathbf{j}+(2^{n+1},0)+\mathbf{U}_n} &= St_{se}(n) \\ \pi_{Rob}(x)_{\mathbf{i}+\mathbf{j}+(0,2^{n+1})+\mathbf{U}_n} &= St_{nw}(n) & \pi_{Rob}(x)_{\mathbf{i}+\mathbf{j}+(2^{n+1},2^{n+1})+\mathbf{U}_n} &= St_{ne}(n) \end{aligned}$$

where π_{Rob} is the projection according the first coordinate.

2.4.4 Linear net gluing

With this additional layer, we have the following properties.

Proposition 3. *Any $(2^{n+1} - 1)$ -block in the language of X'_{Rob} appears as a sub-pattern of a supertile of order $n + 2$.*

Proof. Consider some $(2^{n+1} - 1)$ -block $p \in \mathcal{L}(X'_{\text{Rob}})$ and let $x \in X'_{\text{Rob}}$ such that $p \sqsubset x$. If p is a supertile of order n , the proof is ended since it is included in a supertile of order $n + 2$ by the definition of supertiles. If not, according to the global behavior of X'_{Rob} , the pattern p in x is composed by parts of two supertiles of order n separated by one horizontal or one vertical line of width one, or parts of four supertiles separated by a cross (this is a consequence of 2.4.3). If there are two lines and in the intersection there is a red cross, the pattern can be completed in a supertile of order $n + 1$. If not, the lines of separation of supertiles of order n contain only three or five arrows symbols which constitute the arms of a supertile which can appear in a supertile of order $n + 2$. \square

Proposition 4. *The subshift X'_{Rob} is 16id-net gluing, hence linearly net gluing.*

Proof. Let p, q be two n -blocks in the language of X'_{Rob} with $n \geq 1$. There exists $m \in \mathbb{N}$ such that $2^{m+1} - 1 < n \leq 2^{m+2} - 1$. By the previous proposition, there is a supertile of order $m + 3$ denoted S where p appears. Consider a configuration $x \in X'_{\text{Rob}}$ which have the pattern q at position $(0, 0)$. The supertile S appears periodically in x with period $2^{m+5} = 16 \cdot 2^{m+1} \leq 16n$. Thus the gluing set of p relatively to q in X'_{Rob} contains a set $\mathbf{u} + 2^{m+5}(\mathbb{Z}^2 \setminus \{(0, 0)\})$ for some $\mathbf{u} \in \mathbb{Z}^2$. Thus X'_{Rob} is 16id-net gluing. \square

3 Existence of periodic points for f -block gluing SFT

In this section we study the existence of a periodic point in f -block gluing SFT according to the gluing intensity f .

3.1 'Strong' block gluing imply existence of periodic points

In [PS15] the authors show that any constant block gluing SFT admits periodic point. Using a similar argument, we obtain an upper bound on the gluing intensity to force the presence of periodic point.

Proposition 5. *Let $X \subset \mathcal{A}^{\mathbb{Z}^2}$ be some SFT defined by forbidden patterns in $\mathcal{A}^{\mathbb{U}_r}$ for some $r \geq 2$, which is f -block gluing. If there exists $n \in \mathbb{N}$ such that*

$$f(n) < \frac{\log_{|\mathcal{A}|}(n - r + 2)}{r - 1} - r + 2,$$

then X admits a periodic point.

Proof. Let w be a $n \times (r - 1)$ rectangle in the language of X . There exists $x \in X$ such that $x_{[0, n-1] \times [0, r-2]} = w = x_{[0, n-1] \times [f(n)+r-1, f(n)+2r-3]}$. Consider the sub-patterns of $x_{[0, n-1] \times [0, f(n)+r-2]}$ over supports $[k, k+r-2] \times [0, (f(n)+r-2)]$. There are $n - (r - 2)$ of them and the number of possibilities is $|\mathcal{A}|^{(r-1)(f(n)+r-2)}$. Since $f(n) < \frac{\log_{|\mathcal{A}|}(n-r+2)}{r-1} - r + 2$, by the pigeon hole principle, there exist $k, l \in [0, n - r + 1]$ such that $l > k$ and $x_{[k, k+r-2] \times [0, f(n)+r-2]} = x_{[l, l+r-2] \times [0, f(n)+r-2]}$ (see Figure 9). Thus the periodic configuration defined by

$$z_{(i(f(n)+r-2), j \min(r-1, l-k+1) + [0, r-2] \times [0, f(n)+r-2])} = x_{[k, \min(k+r-2, l)] \times [0, f(n)+r-2]}$$

for all $(i, j) \in \mathbb{Z}^2$ satisfies the local rules of X . We deduce that $z \in X$. \square

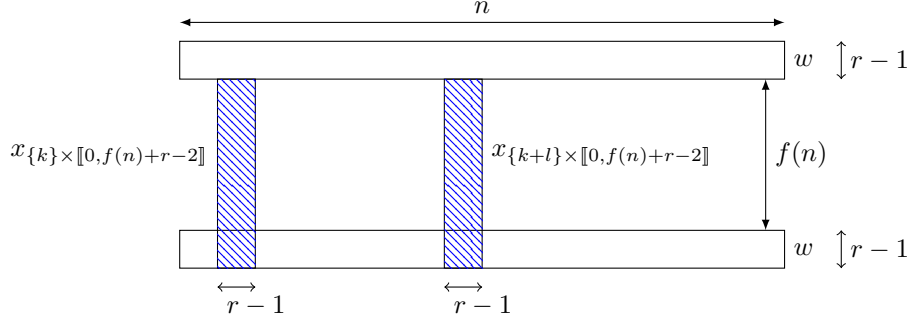


Figure 9: If $n - r + 2 > |\mathcal{A}|^{(r-1)(f(n)+r-2)}$, it is possible to find a pattern of x where the horizontal and vertical borders are similar.

3.2 Density of periodic points for 'very strong' block gluing

Using a similar argument as in [PS15], we obtain also an upper bound on the gluing intensity to force the density of periodic points.

Proposition 6. *Let X be some f -block gluing \mathbb{Z}^2 -SFT with f being a function such that $f(n) \in o(\log(n))$. Then X has a dense set of periodic points.*

Proof. Consider p some n -block in the language of X , and take 2^k copies of it. We group them by two and glue the couples horizontally, at distance $f(n)$. Then glue the obtained rectangles after grouping them by two, at distance $f(2n + f(n))$, and repeat this operation until having one rectangular block q , having length equal to $(2id + f)^{\circ k}(n)$. Then consider some $(2id + f)^{\circ k}(n) \times (r-1)$ pattern l , where r is the order of the SFT X . Glue it on the top of q with $f(\max((2id + f)^{\circ k}(n), n, r))$ lines between the two rectangles. Then glue the rectangle l under the obtained rectangle with

$$f(\max(f(\max((2id + f)^{\circ k}(n), n, r)) + r + n, (2id + f)^{\circ k}(n)))$$

lines between them. For k great enough (depending on n), these two last distances are equal to $f((2id + f)^{\circ k}(n))$, and $f(f((2id + f)^{\circ k}(n)) + n + r)$ respectively. By the gluing property, the obtained pattern (see Figure 10) is in the language of X .

Consider the $(r-1) \times (r + n + f((2id + f)^{\circ k}(n)) + f(f((2id + f)^{\circ k}(n)) + n + r))$ sub-patterns that appear on the bottom of the columns just on the right of each occurrence of the pattern p . There are 2^k of them, and there are at most $(|\mathcal{A}|)^{r+n+f((2id+f)^{\circ k}(n))+f(f((2id+f)^{\circ k}(n))+n+r))}$ different possibilities. From the fact that $f \leq id$, it follows that

$$(|\mathcal{A}|)^{r+n+f((2id+f)^{\circ k}(n))+f(f((2id+f)^{\circ k}(n))+n+r))} \leq (|\mathcal{A}|)^{2(r+n)+2f(3^k n)} \leq 2^k$$

for k great enough.

By the pigeon hole principle, two of these patterns are equal. Consider thus the rectangle between these two occurrences (including the second one). This rectangle can be repeated on the whole plane to get a periodic configuration which is in X .

The set of periodic configurations obtained by this method is dense in X (for every pattern in its language appear in such a configuration). \square

Corollary 2. *A subshift X verifying the conditions of proposition 6 has a decidable language, meaning that there is an algorithm that, taking as input any finite block, outputs 0 if this block is in the language of X and else 1.*

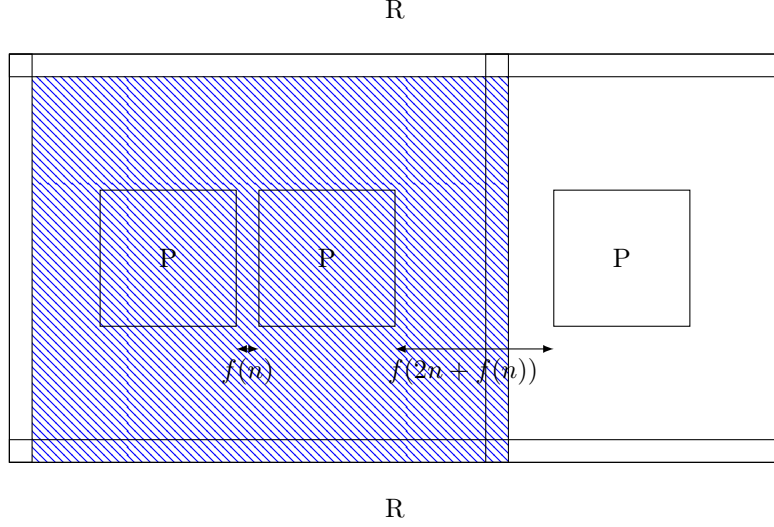


Figure 10: An illustration of the proof of Proposition 6. The result of the procedure is the colored rectangle.

3.3 An example of linearly block gluing SFT with non decidable language

In the previous section we have proven that a subshift of finite type with a strong block gluing properties. In fact this property is no longer true considering linearly block gluing subshifts of finite type. In this section we plan to give an example of linearly block gluing subshifts of finite type with non decidable language.

A Turing machine is an automaton with a finite number of internal states which reads and writes letters on an one-sided infinite tape. The computation begins with the machine in a special initial state and the head located over the leftmost symbol. Initially, the tape contains some data which is the input of the computation. The state of the data tape along with the location and internal state of the machine are called a configuration of the Turing machines. A configuration uniquely determines all the future configurations by a discrete time computation process. At each iteration the machine is located over some symbol of the tape, reads it and based on this data and on its internal state, performs the following actions : it replaces the current data symbol by a new one, updates its internal state and moves to the left or right. The computation may halt after a finite number of steps if the machine either moves off the tape or enters a halting state. A machine is formally some $\mathcal{M} = (Q, q_0, q_h, \mathcal{A}_{\mathcal{M}}, \#, \delta)$ where Q refers to the set of internal states of the machine, q_0 the initial state, q_h the halting state, $\mathcal{A}_{\mathcal{M}}$ the tape alphabet with a blank symbol $\#$ and $\delta : Q \times \mathcal{A}_{\mathcal{M}} \rightarrow Q \times \mathcal{A}_{\mathcal{M}} \times \{L, R\}$ the transition function (where L means left and R means right, and q_h the halting state).

The set of possible space-time diagrams of a machine (subset of $(A \times Q \times \{\leftarrow, \rightarrow\}) \cup A \times \{\leftarrow, \rightarrow\})^{\mathbb{Z}^2}$ where line n is the image of the line $n - 1$ after one step of computation, and the arrows symbols are used so that there is a unique machine head in a line) is of finite type, with constraints on 3×2 patterns as follows :

1. If the first line of the pattern contains no head, it has to be as follows :

u	v	w
u	v	w

2. Else, if for instance the machine head is in the $(1, 3)$ position with state q_1 and data w and

$\delta(q_1, w) = (q_2, x, L)$, we forbid another pattern than :

u	(q_2, v)	x
u	v	(q_1, w)

with similar rules for other local configurations.

- Moreover, the incoming/outgoing arrows have to match (this guarantees that there is a unique machine working in a configuration).

Using this, we build an example of some linearly block gluing SFT which language is non decidable.

Proposition 7. *There exists some $O(n)$ -block gluing \mathbb{Z}^2 -SFT with non decidable language.*

Proof. Consider the SFT X_{undec} which has two layers. The first one has alphabet :



The rules are the following :

- two horizontally adjacent non blank symbols have the same color.
- two vertically adjacent non blank symbols verify the following :
 - if the bottom symbol is black, the top symbol is red or purple.
 - if the bottom symbol is red, the top symbol is green.
 - if the bottom symbol is purple, the top symbol is orange.
 - if the bottom symbol is green or orange, the top symbol is black.

- the patterns



are forbidden, where the gray symbol stands for any non blank symbol.

- the patterns



are forbidden, where the gray symbol stands for any non blank symbol. Similar rules replacing the red symbol with a green one.

- the patterns



are forbidden, where the gray symbol stands for any non blank symbol. Similar rules replacing the red symbol with an orange one.

These rules imply that :

- above $\square \blacksquare^n \square$ there is $\square \text{red}^n \square$, or $\square \text{purple}^n \square$.
- above $\square \text{red}^n \square$ there is $\square \text{green}^n \square$.
- above $\square \text{purple}^n \square$ there is $\square \text{orange}^n \square$.
- above $\square \text{green}^n \square$ there is $\square \text{black}^n \square$.
- above $\square \text{orange}^n \square$ there is $\square \text{black}^n \square$.

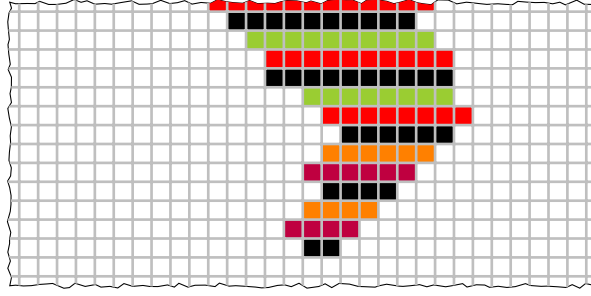


Figure 11: An example of configuration that respects the rules of the first layer of X_{Undec} .

All the configurations consist in shapes as in Figure 11 on a background of \square symbols.

The second layer consists in the implementation of a Turing machine over these shapes. This is done as follows : the blank symbols are superimposed with a blank symbol, and the Turing machine symbols are superimposed over non blank symbols. Moreover, considering a 3×2 pattern whose projection on the first layer is amongst the following :



where the gray symbol stands for any non blank symbol, then the rules of the turing machine apply in the second layer. considering a 3×2 pattern whose projection on the first layer is amongst the following :



where the gray symbol stands for any non blank symbol, in the second layer, every symbol in the top row is equal to the symbol just on the right in the bottom row. Similar rules replacing the red symbol by a green one. Replacing red or green by purple or orange, we have the rule that in the second layer, every symbol in the top row is equal to the symbol just on the left in the bottom row.

These rules imply that every line of a shape in the second layer is :

- the transformation of the line below by a step of the machine, if the line below is black.
- the line below shifted left if this line is red or green.
- the line below shifted right if this line is purple or orange.

The alphabet of the machine is $\{\#, 0, 1\}$. This is a universal Turing machine that has the following behavior when in initial state : it searches for the next symbol $\#$ on its left side, then the next symbol on its right side, reads the sequence of 0, 1 symbols that lies between the two $\#$ symbols. Then it simulates the n th turing machine with n the integer whose base two decomposition is the previous sequence of 0, 1. When the machine stops, it enters in the special state h .

We add the rule that the state h is forbidden.

This subshift is sharp linearly block gluing : the worst case for gluing being blocks filled with colored symbols, to glue them complete the two blocks in the first layer, as in example 3, into a shape surrounded with blank symbols but on the top. Then complete the trajectory of the machine if there is one. The two extended patterns can be glued horizontally without constraint on the

distance, because lines can be shifted towards opposite directions. For vertical gluing, one of the pattern is extended linearly so that its lines are sufficiently shifted.

The language of this subshift is undecidable, because if it was decidable, it would exist an algorithm to decide the halting problem. \square

3.4 Existence of non-periodic low mixing subshifts

In this section, we give a proof of the following theorem :

Theorem 1. *There exists a linearly block gluing aperiodic \mathbb{Z}^2 -SFT.*

3.4.1 A subshift inducing pseudo-coverings by curves

Definition Let us denote Δ the \mathbb{Z}^2 SFT on alphabet $\{\rightarrow, \downarrow\}$, defined by the forbidden patterns :

$$\begin{array}{ccc} \downarrow & \rightarrow & \downarrow \\ \downarrow & \rightarrow & \rightarrow \end{array} .$$

Pseudo-coverings by curves Let us introduce some words in order to talk about the global behavior induced by these rules :

- An (infinite) **curve** in \mathbb{Z}^2 is a set $\mathcal{C} = \varphi(\mathbb{Z})$ for some application $\varphi : \mathbb{Z} \rightarrow \mathbb{Z}^2$ such that for all $k \in \mathbb{Z}$, $\varphi(k+1) = \varphi(k) + (1, 0)$ or $\varphi(k+1) = \varphi(k) + (1, -1)$.
- We say that a curve is **shifted downwards** at position $\mathbf{j} \in \mathbb{Z}^2$ when there exists some $k \in \mathbb{Z}$ such that $\varphi(k) = \mathbf{j}$ and $\varphi(k+1) = \mathbf{j} + (1, -1)$.
- A **pseudo-covering of \mathbb{Z}^2 by curves** is a sequence of curves $(\mathcal{C}_k)_{k \in \mathbb{Z}}$ such that for every $\mathbf{j} \in \mathbb{Z}^2$, there exists some $k \in \mathbb{Z}$ such that $\{\mathbf{j}, \mathbf{j} + (0, 1)\} \cap \mathcal{C}_k \neq \emptyset$ (meaning that every element of \mathbb{Z}^2 is in a curve or the vector just above is), and for every $k \neq k'$, $\mathcal{C}_k \cap \mathcal{C}_{k'} = \emptyset$ (the curves do not intersect). We say that two curves in this pseudo-covering are **contiguous** when the area delimited by these two curves does not contain any third curve. The gap between two contiguous curves in some column is the distance between the intersection of these two curves with the column. This gap is 0 or 1 between two contiguous curves in a pseudo-covering.

A configuration in Δ induce a pseudo-covering by curves Let $\delta \in \Delta$. Let us consider the pseudo-covering of \mathbb{Z}^2 by curves $(\mathcal{C}_k(\delta))_{k \in \mathbb{Z}}$, such that $\mathcal{C}_k(\delta) = \varphi_{\delta,k}(\mathbb{Z})$, where $\varphi_{\delta,k}$ is as follows. For every $\mathbf{i} \in \mathbb{Z}^2$ such that $\mathbf{i} = \varphi_{\delta,k}(n)$ for some $k \in \mathbb{Z}, n \in \mathbb{Z}$:

- if $\delta_{\mathbf{i}} = \rightarrow$, and $\delta_{\mathbf{i}+(1,0)} = \downarrow$, then $\varphi_{\delta,k}(n+1) = \mathbf{i} + (1, -1)$ (the curve is shifted downwards in this column).
- else $\delta_{\mathbf{i}} = \rightarrow$ and $\delta_{\mathbf{i}+(1,0)} = \rightarrow$, then $\varphi_{\delta,k}(n+1) = \mathbf{i} + (1, 0)$.

This gives the construction of the curves from the knowledge of one point in it. Let us complete this description by attributing a point to the curve k for all $k \in \mathbb{Z}$. If $\delta_{(0,0)} = \rightarrow$, then $(0, 0) = \varphi_{\delta,0}(0)$. Else $(0, 1) = \varphi_{\delta,0}(0)$.

In addition, for x such that $\delta_{(x_k, 0)}$ is the k th \rightarrow in the column 0, counting from the previous considered one, then $(x_k, 0) = \varphi_{\delta,k}(0)$.

The first forbidden pattern induce that all the curves of this pseudo-covering can not be shifted downwards multiple times in the same column, and the second one that if a curve is shifted downwards at position $\mathbf{i} \in \mathbb{Z}^2$, then there is no curve going through position $\mathbf{i} - (1, 1)$.

Figure 12 gives an illustration of the definition of a curve in a configuration of Δ .

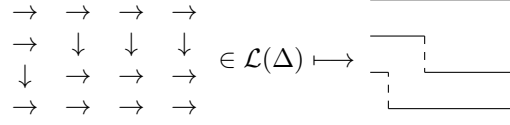


Figure 12: An example of a pattern admissible in Δ and the curves going through it.

3.4.2 Deformation operators on subshifts of finite type

Let \mathcal{A} be some alphabet. Denote $S_{\mathcal{A}}$ the set of SFTs over \mathcal{A} . We introduce operators $d_{\mathcal{A}} : S_{\mathcal{A}} \rightarrow S_{\tilde{\mathcal{A}}}$, with $\tilde{\mathcal{A}} = (\mathcal{A} \cup \{\square\}) \times \{\rightarrow, \downarrow\}$.

Pseudo-projection Consider the subshift $\Delta_{\mathcal{A}} \subset (\mathcal{A} \cup \{\square\})^{\mathbb{Z}^2} \times \Delta$, where the forbidden patterns are the ones defining Δ and the patterns where a symbol in \mathcal{A} is superimposed to a \downarrow symbol or where \square is superimposed to a \rightarrow symbol.

Define a **pseudo-projection** $\mathcal{P} : \Delta_{\mathcal{A}} \rightarrow \mathcal{A}^{\mathbb{Z}^2}$, as follows : for $(y, \delta) \in \Delta_{\mathcal{A}}$,

$$(\mathcal{P}(y, \delta))_{i,j} = y_{\varphi_{\delta,j}(i)}.$$

Notice that the function \mathcal{P} is continuous but not shift invariant.

We denote π_1 the projection on the first layer ($\pi_1(y, \delta) = y$), and π_2 the projection on the second layer.

Definition of the operators Let X be some SFT on the alphabet \mathcal{A} , and define $d_{\mathcal{A}}(X) = \mathcal{P}^{-1}(X)$. Denoting r the order of the SFT X , $d_{\mathcal{A}}(X)$ can be defined by imposing that, considering the intersection of a set of r contiguous curves with r consecutive columns, the corresponding r -block is not a forbidden pattern in X . Because the gap between two contiguous curves is bounded, $d_{\mathcal{A}}(X)$ is defined by a finite set of forbidden patterns, then is an SFT.

Properties of the operators $d_{\mathcal{A}}$ We use the following properties of the operators $d_{\mathcal{A}}$ in order to prove Theorem 1.

Proposition 8. *For an aperiodic SFT X on the alphabet \mathcal{A} , $d_{\mathcal{A}}(X)$ is also non-periodic.*

Proof. Assume that a configuration $y \in d_{\mathcal{A}}(X)$ is periodic : there exists $n > 0$ such that for all i, j , $y_{i+n,j} = y_{i,j+n} = y_{i,j}$. We will prove that the pseudo-projection of y on X , $x = \mathcal{P}(y)$ is periodic.

To each column k in y we associate the bi-infinite word ω^k in $(\mathbb{Z}/n\mathbb{Z})^{\mathbb{Z}}$ such that for all $i \in \mathbb{Z}$, ω_i^k is the element $\overline{m_i}$ of $\mathbb{Z}/n\mathbb{Z}$, class modulo n of m_i where $(0, m_i)$ is the intersection position of the i th curve of $\pi_1(y)$ with the column k . Following a curve (see Figure 13) from the column 0 to the column n , we get an application ψ from the set of possible $\overline{m_i}$ into itself (using the vertical periodicity of the projection of y on the second layer). The word ω^n is obtained from ω^0 applying ψ to all the letters in ω^0 . For ψ is a invertible function from a finite set into itself (indeed, we have an inverse map following the curve backwards), there exists some $c > 0$ integer such that $\psi^c = \text{Id}$, hence such that $\omega^{nc+j} = \omega^j$ for all integers j . That means that the column $cn + j$ is obtained by shifting kn times downwards, for some $k \geq 0$, the column j . Using the horizontal periodicity of y , we then have that $(x_{j,z})_{z \in \mathbb{Z}} = (x_{cn+j,z+kn})_{z \in \mathbb{Z}}$, and using the vertical periodicity, that $(x_{j,z})_{z \in \mathbb{Z}} = (x_{cn+j,z})_{z \in \mathbb{Z}}$, hence the configuration $x \in X$ is periodic, which can not be true.

As a consequence, no configuration in $d_{\mathcal{A}}(X)$ can be periodic, hence this subshift is non-periodic \square

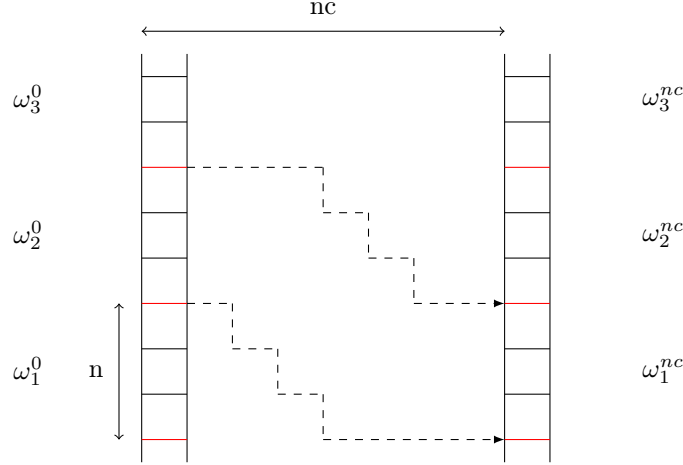


Figure 13: Illustration of the proof of Proposition 8.

The following proposition will be a useful tool in order to prove that the operators $d_{\mathcal{A}}$ transform linearly net gluing subshifts into block gluing ones.

Proposition 9 (Completing blocks). *There exists an algorithm \mathcal{T} that takes as input some locally admissible n -block p of Δ , and outputs a rectangular pattern $\mathcal{T}(p)$ which has p as a sub-pattern and such that :*

- *the number of curves in $\mathcal{T}(p)$ is equal to the number of its columns,*
- *the dimensions of $\mathcal{T}(p)$ are smaller than $5n$,*
- *the top and bottom rows of $\mathcal{T}(p)$ have only \rightarrow symbols. This means that all the curves crossing $\mathcal{T}(p)$ comes from its left side and go to the right side.*

Remark 1. *The properties of the pattern $\mathcal{T}(p)$ ensure that this is a globally admissible pattern. Hence every locally admissible pattern of the subshift Δ is globally admissible.*

Proof. : If p is a 1-block, and p is a single \rightarrow , then the result is direct. If p is a single \downarrow , then it can be completed in

$$\begin{array}{ccccc} \rightarrow & \rightarrow & \rightarrow \\ \rightarrow & \downarrow & \downarrow \\ \downarrow & \rightarrow & \rightarrow \\ \rightarrow & \rightarrow & \rightarrow \end{array}$$

that verifies the previous assertion.

If p is a n -block with $n \geq 2$:

First step : Completing the curves that enter in the block upside/downside.

1. While in the above line of the block p there is a pattern $\downarrow \rightarrow$ that appears, meaning that there is an incoming curve (which position of the \rightarrow is in this curve), consider the first incomplete one, from left to right.
2. Complete the incoming curve by adding a \downarrow over the \rightarrow , then \rightarrow symbols on the left until meeting another \downarrow , and repeat this operation until the left side of the block p .
3. Do similar operations on the bottom of the block.

Example 4. If we take p the following 4-block

```

→  →  ↓  →
→  ↓  →  →
↓  →  →  →
→  →  →  →

```

at this point, we obtain :

```

→  →  →  ↓
→  →  ↓  →
→  ↓  →  →
↓  →  →  →
→  →  →  →

```

Second step : Completing the pattern on the top and bottom until the top row and bottom row are straight.

While the top curve of the pattern is not straight, apply the following procedure :

1. On the top of the last column, add a \downarrow and keep adding \downarrow on the left until meeting on the left an already defined symbol.
2. Add another curve above by the following procedure. Add a \rightarrow on the top of the last column, and then add \rightarrow symbols on the left until meeting an already defined symbol on the left ; when that happens, add a \downarrow above and then add \rightarrow 's on the left until reaching a defined symbol. Repeat this operation until reaching the first column.

Do similar operations on the bottom.

Example 5. At this point, we obtain :

```

→  →  →  →
→  →  →  ↓
→  →  ↓  →
→  ↓  →  →
↓  →  →  →
→  →  →  →

```

Third step : Equalization of the number of curves and the number of columns.

If the number of columns is smaller than the number of curves, then add a number of columns equal to the difference, by adding copies of the last column. If the number of curves is smaller, then add lines of \rightarrow symbols on the top.

Example 6. After this last step we obtain :

```

→  →  →  →  →
→  →  →  ↓  ↓
→  →  ↓  →  →
→  ↓  →  →  →
↓  →  →  →  →
→  →  →  →  →

```

The dimensions of $\mathcal{T}(p)$ for p a n -block are smaller than the sum of the dimension of p (equal to n), two times the number of entering curves by the top and outgoing by the bottom (one for completing the curves (first step), and one for reducing the shifts (second step)), each one smaller than n . The third step does not make this bound greater, because in this pattern the number of curves is smaller than the number of lines. Hence the dimensions of $\mathcal{T}(p)$ are less than $5n$. \square

Let us denote ρ the transformation on subshifts that acts as a rotation by an angle $\pi/2$. If X is defined by a set \mathcal{F} of forbidden patterns, $\rho(X)$ is defined by the set of rotated patterns. Thus ρ transforms SFT into SFT.

Theorem 2. *The operator $d_{\tilde{\mathcal{A}}} \circ \rho \circ d_{\tilde{\mathcal{A}}} \circ \rho \circ d_{\mathcal{A}}$ transforms linear net-gluing subshifts of finite type into linear block gluing ones.*

Proof. : Let X be a subshift such that for all $n > 0$ and every couple of n -blocks p, q , the gluing set of p relative to q in X contains $\mathbf{u}(\mathbf{p}, \mathbf{q}) + (n + \tilde{f}(p, q))(\mathbb{Z}^2 - (0, 0))$, for some $\mathbf{u}(\mathbf{p}, \mathbf{q}) \in \mathbb{Z}^2$, and such that $\tilde{f}(p, q) \leq f(n)$ for all p, q n -blocks. We consider in this proof that $\mathbf{u}(\mathbf{p}, \mathbf{q}) = \mathbf{0}$, for this takes no great effort to adapt the proof to case $\mathbf{u}(\mathbf{p}, \mathbf{q}) \neq \mathbf{0}$.

Consider two n -blocks p, q of the subshift $d_{\mathcal{A}}(X)$. We complete $\pi_2(p)$ and $\pi_2(q)$ into $\mathcal{T}(\pi_2(q))$ and $\mathcal{T}(\pi_2(p))$. We can consider that these two patterns have the same number of curves. Then we complete these patterns with letters in \mathcal{A} into admissible patterns of $d_{\mathcal{A}}(X)$. The pseudo-projections of these patterns on X are m -block of X , where m is the number of curves in $\mathcal{T}(\pi_2(p))$ and $\mathcal{T}(\pi_2(q))$. We call them $X(p)$ and $X(q)$. For all $\mathbf{u} \in (m + \tilde{f}(X(p), X(q)))(\mathbb{Z}^2 - (0, 0))$, there exists a configuration $x^{\mathbf{u}} \in X$ such that $x^{\mathbf{u}}_{\llbracket 0, m-1 \rrbracket^2} = X(q)$, and $x^{\mathbf{u}}_{\mathbf{u} + \llbracket 0, m-1 \rrbracket^2} = X(p)$.

The gluing sets of $d_{\mathcal{A}}(X)$ contain periodic sets of infinite columns Consider some $\mathbf{u} = (m + \tilde{f}(X(p), X(q)))\mathbf{v}$ with $\mathbf{v} = (\mathbf{v}_1, \mathbf{v}_2)$, $\mathbf{v}_1 \geq 4$, and $\mathbf{v}_2 = 0$. We prove that the pattern p can be glued on every position of the infinite column containing \mathbf{u} relatively to q .

First step : Compactification of the outgoing curves : We extend $\mathcal{T}(\pi_2(q))$ using the following procedure : while in the last column of the pattern, there is some sub-pattern $\begin{smallmatrix} \rightarrow \\ \downarrow \end{smallmatrix}$ (meaning that there is a gap between two outgoing curves), do the following : on the right of the patterns $\begin{smallmatrix} \rightarrow \\ \downarrow \end{smallmatrix}$, write $\begin{smallmatrix} \downarrow \\ \rightarrow \end{smallmatrix}$ and write a copy of the other \rightarrow symbols on their right side.

Example 7. *Taking the same example as previously, the result is :*

$$\begin{array}{ccccccc} \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \downarrow \\ \rightarrow & \rightarrow & \rightarrow & \downarrow & \downarrow & \downarrow & \rightarrow \\ \rightarrow & \rightarrow & \downarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow \\ \rightarrow & \downarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow \\ \downarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow \\ \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow & \rightarrow \end{array}$$

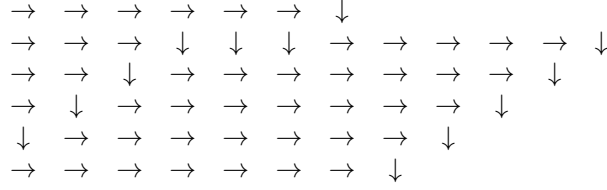
After that, we extend $\mathcal{T}(\pi_2(p))$ but with a similar procedure on the left side.

Second step : Making the curves shift

Let $\tilde{f}(X(p), X(q)) + m \geq k \geq 1$ be some integer. We add columns on the right of the extension of $\mathcal{T}(\pi_2(q))$ using the following procedure, in order to make all the curves of it shift k times :

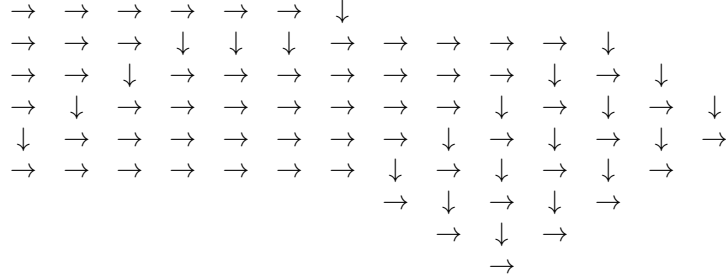
1. Consider the right part of the pattern constituted with \rightarrow symbols and add a triangle made of \rightarrow symbols except on the diagonal part where we write \downarrow symbols (this is the first shift).

Example 8. Taking the same example as previously, the result is :



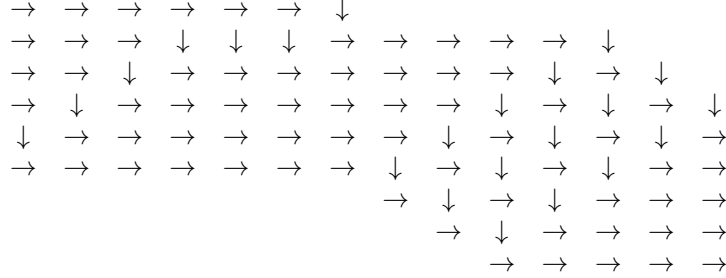
2. Then repeat $k - 1$ times the following operation : adding under each \downarrow on the right side a \rightarrow under, and after that a \downarrow on the right of the \rightarrow .

Example 9. Taking the same example as previously, with $k = 3$, the result is :



3. Complete the curves with \rightarrow symbols so that they end in the last column added.

Example 10. Taking the same example as previously, with $k = 3$, the result is :



4. Then continue the curves straightly on a number of columns so that the total number of columns added to $\mathcal{T}(\pi_2(q))$ and $\mathcal{T}(\pi_2(p))$ is equal to $\mathbf{u}_1 - m$. This is possible because until now the number of added columns is less than $m + m + f(X(p), X(q)) + m \leq \mathbf{u}_1 - m$ (because $\|\mathbf{v}\|_\infty = \mathbf{v}_1 \geq 4$).

Third step : We branch the obtained patterns.

1. Then branch the two patterns obtained such that the i th curve outgoing curves of the extension of $\mathcal{T}(\pi_2(q))$ is branched to the i th of incoming of the extension of $\mathcal{T}(\pi_2(p))$.
2. After that, complete the first column of $\mathcal{T}(\pi_2(q))$ with \rightarrow symbols and the left side of it by copying this infinite column until infinity. Then do the same thing to the last column of $\mathcal{T}(\pi_2(p))$ but copying it on the right side until infinity.
3. Complete the middle part by curves that have gap zero with the curve above and below.
4. We have now a configuration of $\pi_2(d_{\mathcal{A}}(X))$ which we can assume having the south west corner of $\pi_2(q)$ in position $(0, 0)$. We add letters in \mathcal{A} such that the pseudo-projection of it on X is $x^{\mathbf{u}}$.

From this construction we deduce that

$$\left\{ \mathbf{w} \in \mathbb{Z}^2 \mid \mathbf{w}_1 = \mathbf{v}_1(m + \tilde{f}(X(p), X(q))), |\mathbf{v}_1| \geq 4 \right\} \subset \Delta_{d_{\mathcal{A}}(X)}(p, q).$$

Because $m \leq 5n$, and f is non decreasing, this means that the gluing set of any n -block of $d_{\mathcal{A}}(X)$ relatively to another contains size 1 vertical stripes occurring with periodicity less than $4(f(5n) + 5n)$, which is linear in n .

The gluing sets of $d_{\mathcal{A}}(X)$ contain periodic positions in the column 0 Consider some $\mathbf{u} = (m + f(X(p), X(q)))\mathbf{v}$ with $\mathbf{v} = (\mathbf{v}_1, \mathbf{v}_2)$, $\mathbf{v}_2 \neq 0$, and $\mathbf{v}_1 = 0$. The pattern p can be glued relatively to q in $d_{\mathcal{A}}(X)$ in position \mathbf{u} in a configuration $x^{\mathbf{u}}$.

Indeed, the pattern $\mathcal{T}(\pi_2(q))$ can be glued relatively to $\mathcal{T}(\pi_2(p))$ in Δ with relative position \mathbf{u} : to prove that, we glue the two patterns with this relative position, and complete straightly the curves that go through the two patterns, and fulfill \mathbb{Z}^2 with straight curves. Shift it so that $\pi_2(q)$ appears in position $(0, 0)$. Then complete this configuration with letters in \mathcal{A} so that the pseudo-projection is $x^{\mathbf{u}}$.

This means that

$$\left\{ \mathbf{w} \in \mathbb{Z}^2 \mid \mathbf{w}_2 \in (m + \tilde{f}(X(p), X(q))) (\mathbb{Z} \setminus \{0\}), \mathbf{w}_1 = 0 \right\} \subset \Delta_{d_{\mathcal{A}}(X)}(p, q).$$

In other words, the patterns p, q can be glued vertically one relatively to the other in positions with periodicity less than $5n + f(5n)$, and in particular with periodicity less than $4(f(5n) + 5n)$.

The Figure 14 shows the set of positions we proved to be in the gluing set of the pattern p relatively to q .

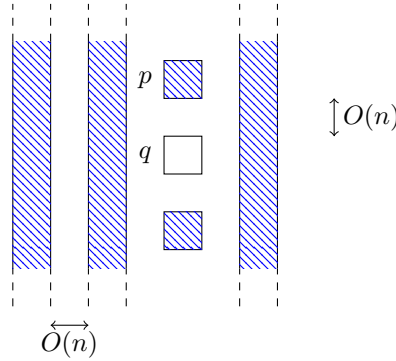


Figure 14: Schematic representation of a set of positions included in the gluing set of some couple of n -blocks in $d_{\mathcal{A}}(X)$.

Proof of the linear block gluing For ρ acts as a $\pi/2$ rotation over patterns, and thus on configurations, the gluing set of some n -block p relatively to another, q , in $\rho \circ d_{\mathcal{A}}(X)$ contains the positions shown by Figure 15. Using the same procedure as in the beginning of this proof, we get that the gluing set of two n -blocks contains some set of positions as in Figure 16 : two half planes, with linear distance to position 0, and periodic positions in column 0. In the end, the gluing sets of two n -blocks in $d_{\tilde{\mathcal{A}}} \circ \rho \circ d_{\tilde{\mathcal{A}}} \circ \rho \circ d_{\mathcal{A}}(X)$ some set of positions as in Figure 17

This means that the subshift $d_{\tilde{\mathcal{A}}} \circ \rho \circ d_{\tilde{\mathcal{A}}} \circ \rho \circ d_{\mathcal{A}}(X)$ is linearly net gluing.

□

Considering the action of this operator on X'_{Rob} , we have a proof of Theorem 1.

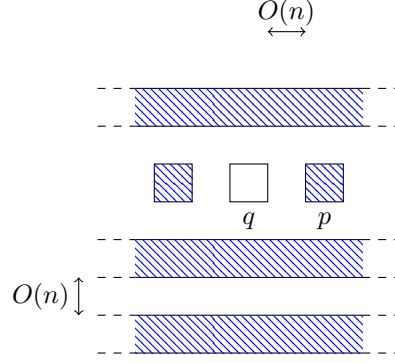


Figure 15: Schematic representation of a set of positions included in the gluing set of some couple of n -blocks in $\rho \circ d_{\mathcal{A}}(X)$.

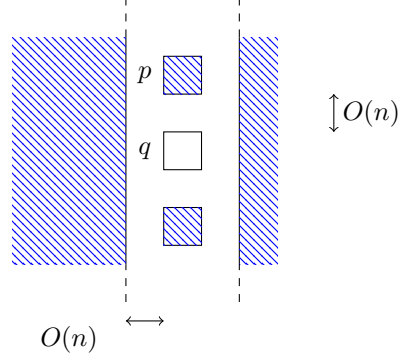


Figure 16: Schematic representation of a set of positions included in the gluing set of some couple of n -blocks in $d_{\tilde{\mathcal{A}}} \circ \rho \circ d_{\mathcal{A}}(X)$.

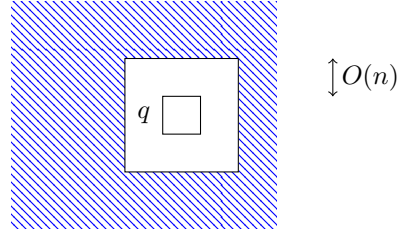


Figure 17: Schematic representation of a set of positions included in the gluing set of some couple of n -blocks in $d_{\tilde{\mathcal{A}}} \circ \rho \circ d_{\tilde{\mathcal{A}}} \circ \rho \circ d_{\mathcal{A}}(X)$.

4 Entropy of block gluing \mathbb{Z}^2 -SFTs

4.1 Computability of the entropy

We recall that a pattern **appears** in X if it appears in a configuration $x \in X$. Denote $\beta_n(X)$ the set of n -blocks that appears in X . Given a SFT defined by a finite set of forbidden patterns \mathcal{F} , a pattern p is **locally admissible** if no pattern of \mathcal{F} appears in p and a pattern p is **globally admissible** if it appears in $X_{\mathcal{F}}$.

Let us denote $N_n(X) = \#\beta_n(X)$. The (topological) **entropy** of the subshift X is defined as :

$$h(X) = \inf_n \frac{\log_2(N_n(X))}{n^2}$$

Notice that we use base two logarithm instead of usual neperian base logarithm, for it is more convenient in the context of this article.

The following notions constitute the basis of an **arithmetical hierarchy of real numbers** from the point of view of computability, introduced by Weihrauch and Zheng [XZ01] :

Definition 5 (Computable numbers). *A real number $h \in \mathbb{R}$ is said to be (Δ_0) -computable when there exists some turing machine that given as input an integer $n \in \mathbb{N}$ outputs some rational number h_n such that $|h_n - h| \leq 2^{-n}$. A real number h is said to be Π_1 -computable when there exists some algorithm that taking as input an integer n outputs a rational number h_n such that $h = \inf_n h_n$.*

The name Π_1 -computable means that such a number is computable with an oracle turing machine with a Π_1 oracle. The following theorem gives the class of numbers which are entropy of a multidimensional SFT :

Theorem 3 ([HM10]). *The entropies of the \mathbb{Z}^k -SFT ($k \geq 2$) are the Π_1 -computable numbers.*

4.2 Strong block gluing implies computability of the entropy

In this section we show that if some SFT is block gluing with an intensity function sufficiently small, the entropy of this subshift is a computable number. In the Proposition 3.3 of [PS15] the authors shows that any \mathbb{Z}^2 -SFT that is constant-block gluing has a computable entropy. Let us generalize this statement for a larger set of functions f :

Proposition 10. *Let X be some f -block gluing \mathbb{Z}^2 -SFT on some alphabet A , with f a non decreasing function that verifies for some $\epsilon > 0$:*

$$\forall n \in \mathbb{N}, f(n) \leq \frac{n^{1/\log_2(5)}}{\log(n)^{1+\epsilon}}.$$

If the complexity function $(N_n(X))_n$ is computable, then the entropy of X is computable.

Proof. : Consider $k \geq 1$, $n \geq 1$, and a number 4^k of 4^n -blocks in the language of X . We group them by two and glue the two elements of each group horizontally, at distance $f(4^n)$ (which is possible, from the block gluing property). Then make groups of two new formed patterns (see Figure 18) and glue them vertically with distance $f(2 \cdot 4^n + f(4^n))$ (completing them into blocks before gluing). Repeat these two operations until there is a unique block left. Denote $l_k(n)$ and $h_k(n)$ its length and height, which verify :

$$\begin{aligned} l_0 &= 4^n \\ h_0 &= 4^n \\ l_{k+1} &= 2l_k + f(h_k) \\ h_{k+1} &= 2h_k + f(l_{k+1}) \end{aligned}$$

This comes from the fact that $l_{k+1} \geq h_k$ and $h_k \geq l_k$, for all $k \geq 0$. This fact is true for $k = 0$ and if true for k , then $h_{k+1} \geq 2l_k + f(h_k) = l_{k+1}$ (for f is non decreasing), and $l_{k+2} = 2l_{k+1} + f(h_{k+1}) \geq 2h_k + f(l_{k+1}) = h_{k+1}$.

This construction leads to $N_{h_k(n)} \geq (N_{4^n})^{4^k}$ (we can choose the 4^k blocks independently).

Moreover, because $h_k \geq l_k$, and f is non decreasing, $h_{k+1} \leq (2id + f \circ (2id + f))(h_k)$, hence $h_k \leq (2id + f \circ (2id + f))^{\circ k}(4^n)$. Let us denote $d_k(n)$ this last number. We have

$$\frac{\log_2(N_{h_k(n)})}{(h_k(n))^2} \geq 4^k \frac{\log_2(N_{4^n})}{(h_k(n))^2} \geq \frac{4^{2n} 2^{2k}}{(d_k(n))^2} \frac{\log_2(N_{4^n})}{4^{2n}} \quad (1)$$

Let us denote g the function defined for all integer n by $g(n) = f(2n + f(n))$. Hence, $d_k(n) = (2id + g)^{\circ k}(n)$, and by induction, using $(2id + g)^{\circ k} = g \circ (2id + g)^{\circ k-1} + 2(2id + g)^{\circ k-1}$:

$$d_k(n) = 2^k 4^n + \sum_{j=0}^{k-1} 2^{k-1-j} g((2id + g)^j(4^n))$$

Using the second condition of the statement, $f \leq id$ so $g \leq f \circ (3id) \leq 3id$. As a consequence we get :

$$\frac{d_k(n)}{2^k 4^n} \leq 1 + \sum_{j=0}^{k-1} 2^{-(j+1)} \frac{g(5^j 4^n)}{4^n}.$$

As a consequence (the first inequality coming from the definition of the entropy), the last sum converges and taking $k \rightarrow +\infty$ we get :

$$\frac{\log_2(N_{4^n})}{(4^n)^2} \geq h(X) \geq \frac{\log_2(N_{4^n})}{(4^n)^2} * \frac{1}{(1 + \sum_{j=0}^{\infty} 2^{-(j+1)} \frac{g(5^j 4^n)}{4^n})}$$

Using $g \leq f(3id)$ we have :

$$\begin{aligned} \frac{g(5^j 4^n)}{4^n} &\leq \frac{3^{1/\log_2(5)} 2^j}{(\log_2(3) + j \log_2(5) + 2n)^{1+\epsilon} 4^{n(1-1/\log_2(5))}} \\ \frac{\log_2(N_{4^n})}{(4^n)^2} &\geq h(X) \geq \frac{\log_2(N_{4^n})}{(4^n)^2} * \frac{1}{(1 + 2 * 4^{n(1-1/\log_2(5))}) \sum_{j=1}^{\infty} \frac{1}{j^{(1+\epsilon)}})} \end{aligned}$$

Thus, if $(N_n(X))_n$ is a computable sequence, the entropy is a computable number. \square

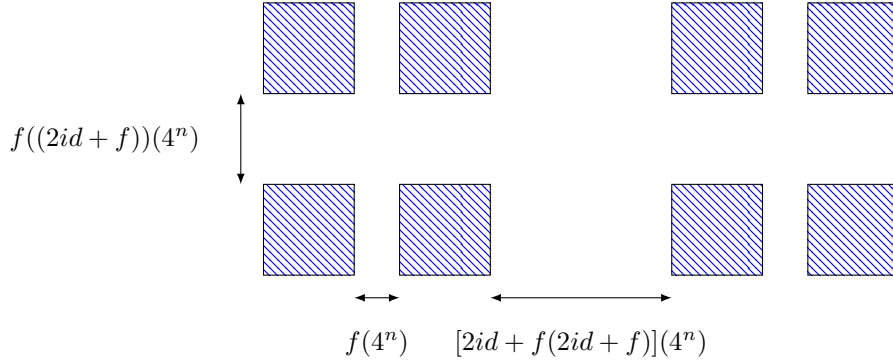


Figure 18: An illustration of the proof of Proposition 10. First three steps of the gluing process of 4^n -blocks.

Corollary 3. For X an SFT which is $o(\log(n))$ block gluing, the entropy $h(X)$ is computable.

4.3 Theorem of realization: outline of the proof

The aim of the following sections is to draw a gap between two behaviors for the entropy regarding the gluing property (low and strong block gluing). We prove here a theorem of realization which characterize the possible entropies of sufficiently low (meaning with an intensity function being great) block gluing subshifts of finite type:

Theorem 4. *The entropies of linearly block gluing \mathbb{Z}^2 -SFTs (and in particular transitive ones) are exactly Π_1 -computable numbers.*

Hochman and Meyerovitch (in [HM10]) used the type of construction we make in this part to prove that every Π_1 -computable number is the entropy of a \mathbb{Z}^2 -SFT. They expressed the question about the realization of every Π_1 -computable number as the entropy of a \mathbb{Z}^2 -SFT that would be transitive (**Problem 9.1** of [HM10]). **We answer here to this question proving that every Π_1 number is the entropy of a linearly block gluing \mathbb{Z}^2 -SFT** (this provides a proof to Theorem 4).

Let h be such a number. We construct a linearly net-gluing SFT X which has entropy h by superposition of various layers [Figure 19]. After that we apply a combination of ρ and operators adapted from the operators d_A to obtain the block gluing property. We describe the global behavior here and will precise in the following subsections the alphabet and local rules of each layer :

- *Basis layer* [Section 4.8] : this layer has symbols 0, 1, and has no internal local rule. The function of the other layers is to control the frequency of positions where the symbol 1 (the symbol 0 being always authorized) can appear so that the entropy of the total subshift will be h . It has interaction rules with the structure and cells coding/synchronization layers.
- *Structure layer* [Section 4.4] : This second layer is derived from the Robinson subshift X_{Rob} , so that every configuration in this layer has a description as a hierarchy of 'cells' (we define this notion later) of order $n \geq 1$ (including ∞), that occur periodically (vertically and horizontally) in each configuration, and containing other cells of smaller order. All the configurations are decomposed in two subsets of \mathbb{Z}^2 : one for the structure, hierarchy and information exchanges between cells (having frequency 3/4), and the other one corresponds to the inside of the cells (having frequency 1/4). Denoting i the integer part of $4h$, we impose in the basis layer, under the first set, a frequency of 1 symbols equal to $i/3$. This means that we adjust the entropy due to the bits in the basis layer corresponding to the first subset according to the position of h in the interval $[0, 1]$. The machines will control the frequency of 1 symbols in the second subset.

Let us fix an integer $N > 0$. This integer separates cells of order $n \leq N$ and order $n > N$ cells that have different behaviors. In each cell, we program four types of machines, each will use a fixed quarter of the cell.

In a cell of order greater than N , two of them will control the frequency of positions where the two symbols 0, 1 in the second subset appear, so that the total frequency of 1 symbols will be h (as a consequence the entropy will be h). The others two will be used to simulate the behaviors that occur in infinite cells, which are not taken into account in [HM10] for those behaviors occur on a set that have frequency zero in every configuration, hence the entropy was not affected. However, in order to have the net gluing property, we have to take them into account.

If the cell order is smaller than N , then all the machines control the frequency. The set of the insides of the cells is decomposed in every configuration into two sets : the first one is the set of the insides of quarters used for simulation, and the other is the set of the insides of

quarters used for correct computations. For the simulation of infinite cells behaviors induce parasitic entropy, we choose N such that the frequency of this first set together with the 'structure' set is smaller than h . Then we program the machines to control the frequency of the second set such that the total frequency will be h .

Each cell is informed in its center (called nucleus) which quarters are used in it for the simulation of the infinite cells machines behaviors, and which ones are elected for correct computations.

- *Cells coding and synchronization layer* [Section 4.5] : This third layer is used to allow synchronization of some bits used in the control of basis layer bits, called frequency bits, of the cells having the same order that lie in a same greater one, and inhibit it between cells that do not lie in a same greater one. The inhibition is important for net gluing. The synchronization is for the action of the machines : the consequence is that for every integer $k > 0$, there exists $N(k)$ such that in every cell of order greater than $N(k)$, the machines have time to control the data of every cell of order less than k that lie in this one. This synchronization is between correct computation quarters on one hand and each of the simulation quarters on the other hand. This fact permits the simulation of the data which occur in infinite areas.
- *Computation areas* [Section 4.6] : This layer is used to constitute, inside each cell, the computation areas, that is to say the signalization telling the machine where it will execute one step of its computation, and where it will transfer the information of its state to where it will execute the next computation step. This is done for all the four machines. In the two quarters used to simulate behaviors that occur specifically in infinite areas, we simulate the possible behaviors of these computation areas, whereas in the other two quarters, the computation areas are well constituted. For this we use a signal that is triggered and propagates through the walls and reticle of the cell with the information of the quarter in it if there is an error, and we forbid the coexistence of this signal with the information in the center of the cell that this quarter is elected to correct computation.
- *Machines layer* [Section 4.7] : This final layer supports the computations of the machines. Each quarter of a cell have its proper direction of time and space, and the tapes of the machines are not linear : there is space between the data, which will be the set of bits corresponding to the two quarters elected to correct computation, in each cell that lie in the considered one. The information is transmitted through the level of this cell (we call level of order n the union of columns and lines that contains a corner of a cell of order n), hence the machine have access to it looking in the column directly on its right in the two right quarters, and on its left on the two left quarters. Moreover, this data is arranged in the following order : 121312141213121.. where i designates the information that comes from cells of order i . This order does not depend on the order of the cell, so the machine can compute the successive positions where it has to write. The behavior of the machines is to write a Π_1 computable sequence of $\{0, 1\}^{\mathbb{N}}$ (thought as the base two decomposition of a Π_1 -computable number) after computing an expression of the entropy depending on it in section 4.9.3.

We allow the machines to enter in halting state, but in this case, it transmits a signal through its trajectory back in time until initialization, and is forbidden the coexistence of this signal with the information of correct computation for this quarter. Hence, if there is an error signal, the computations are not taken into account.

In [HM10], the obstacles for transitivity comes from two facts : rigidity of the frequency conditions (solved here with the immanence of identification of the bits in a colored area), and infinite

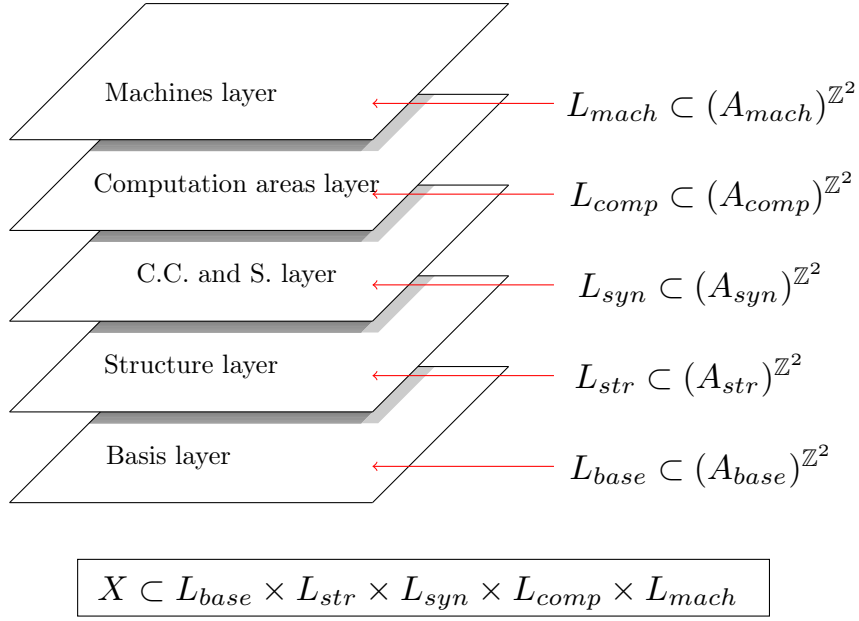


Figure 19: The decomposition of the subshift X into layers.

cells where a machine could have a different behavior than a machine initialized in a finite cell. We solve this problem by splitting every computation area into four sub-areas, and simulating in two of them non well initialized computations as it could happen in an infinite computation area.

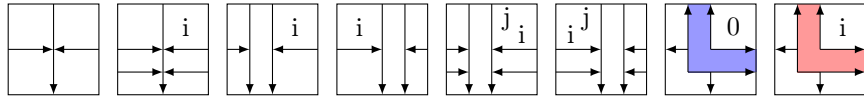
Let us make explicit the local rules that induce these global behaviors. We first present the layers from the structure to the machines, then we talk about the basis. The Figure 19 make a summary of the layer structure of X , with the notations we use for the alphabet and corresponding subshift of each layer.

4.4 Description of the structure layer

In this section, we present the basic structure constituting cells, and the information paths between them.

The structure layer has two sub-layers, the first one being constructed over the Robinson subshift X_{Rob} :

Symbols : the same as X_{Rob} , but with additional counters i, j being 0 or 1, and $i \neq j$ in the same symbol, as follows.



The counter i and j corresponds respectively to incoming arrows and outgoing arrows.

Internal rules : The additional rules are that the counters i and j are transmitted through corresponding arrows.

Global behavior : In a configuration x , we distinguish subsets of \mathbb{Z}^2 that we will designate with the following words :

- The positions of blue symbols are the 0 order cells. In an order $2m$ supertile, with $m \geq 1$, the centers of the four order $2m - 1$ supertiles used to construct it have counter one. An order $m \geq 1$ **cell** is the support of the $(2 \cdot (2^{2m-2+1} - 1) + 3) = (4^m + 1)$ -block having these four points as extremities (thus a finite subset of \mathbb{Z}^2).
- The red symbols with counter zero are at the center of a cell, and in a configuration, their positions are called **nuclei**.
- In a cell, the union of the column and the line which contain the center is called the **reticle** of the cell.
- We call **walls** the frontier of a cell (these are the positions with one counter 1 symbols proper to the cell).
- The other positions that are proper (not included in another smaller cell) to the cell constitute the **cytoplasm**.
- Each of the four parts in an order n cell without walls and reticle is called a **quarter**. We call **simulation quarter** a quarter designated for simulations, and **control quarter** a quarter elected for correct computation.

This is the structure of all the configurations : a hierarchy of cells included one in another. Figure 29 gives an example of a pattern over an order 2 cell in the structure layer, with the corresponding pattern in the synchronization layer.

The cells have the following properties :

- By a recurrence argument, each order m cell contains properly $4 \cdot 12^{i-1}$ order $m - i$ cells, for all $i \in \{1, \dots, m - 1\}$ (in Figure 29, the cell contains 4 cells of order 1).
- As a consequence, each order m cell contains properly $\frac{12^m}{3}$ blue symbols.
- Moreover, in every configuration each order m cell (and in particular the nuclei) repeats periodically with period $(4^m - 1) + 4^m + 1 = 2 \cdot 4^m$.

4.5 Description of the cells coding and synchronization layer

This layer supports the material that permits to synchronize the computations of the machines of the cells included in a same greater one, and codes for the position of simulation quarters.

This layer has also various sub-layers, described as follows.




4.5.1 Defining synchronization areas





In this section, the synchronization areas are defined, that is to say the areas that share the same 'frequency bits' on which the machines have control. These informations are already contained in the structure layer, but we make it visual in this one.

The first sublayer has seven *symbols* :



Each symbol corresponds to a part of the cells :

-  corresponds to the reticle.
-  corresponds to the walls.
-  corresponds to the north east synchronization area.

-  corresponds to the north west synchronization area.
-  corresponds to the south east synchronization area.
-  corresponds to the south west synchronization area.
-  corresponds to the inhibition area.

This layer interacts with the structure layer with the following *rules* :

1. **Specification of the walls** : The symbols in the Robinson subshift with at least a counter 1 are superimposed with dark gray.
2. **Specification of the reticle** :
 - The red symbols in the Robinson subshift with a counter 0 are superimposed with light gray.
 - The non-blue symbols with a unique counter 0 or without counter propagates color through outgoing arrows if this color is light gray, until meeting a dark gray symbol.
3. **Coloring synchronization/inhibition areas** :
 - The south west corners with counter 1 induce red to its north east symbol (with similar rules for the other corners with counter 1), a two counters (1,0) symbol with its long arrows directed to the left induce yellow to its south east symbol and purple to the north east one (with similar rules for other similar symbols). Moreover, a red symbol in the Robinson subshift with counter 0 will induce purple to its north west, red to its south west, orange to north east, and yellow to south east.
 - A colored symbol in this layer transmits its color in all the four directions (unless the near symbol in the direction is gray).
 - A gray symbol can not have another gray symbol on its north east, north west, south west or south east (this induces that the positions of the cytoplasm are colored and not gray).
4. **Managing infinite areas** : In a 3×1 pattern, if the symbol at the center is colored with light gray, then there is only two possibilities for the two others (with similar conditions for 1×3 patterns) :
 - The left one is purple, the right one is orange.
 - The left one is red, the other one is yellow.

This means that on the two sides of an arm of an infinite reticle, it cannot appear a couple of colors that we cannot find around finite reticles.

The *global behavior* induced by these rules is the following : every finite cell has dark gray walls, and is splitted in four parts by its reticle, colored with the lighter gray. The north west part which is specific to this cell is colored with purple, the north east one with orange, the south west one with red and the south east one with yellow (the connected parts having the same color are called **synchronization areas**, and the blank ones (always infinite) are called **inhibition areas** ; notice that ∞ order supertiles correspond, when they are colored, to parts of the same infinite cell, and we make reference to the corresponding synchronization areas when we talk about 'infinite areas'). The Picture 20 shows an exemple of the synchronization areas for a three order cell. When a configuration has an infinite synchronization areas which is blank, we call this type of configuration **inhibition configuration**.

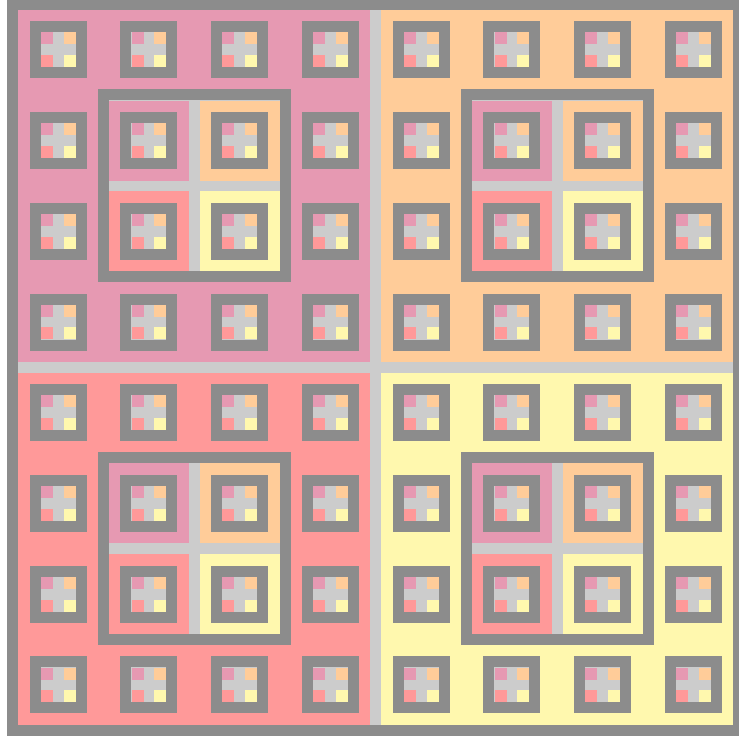


Figure 20: An example of the pattern in the synchronization layer over an order 3 cell.

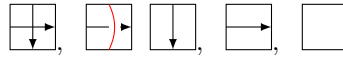
4.5.2 Frequency bits

To the cytoplasm position, we associate a symbol in $\{0,1\}$ called the **frequency bit**, which by local rules is imposed to be the same inside a synchronization area. These are the bits controlled by the machines in the quarters elected for correct computation. In the basis layer, to the 1 frequency symbols will correspond 'random bits' generating entropy according to the frequency of the 1 frequency bits. **The frequency bit is imposed to be 0 over the inhibition areas.**

4.5.3 Synchronization net

In this section, we build the network that permits transfers of information between cells.

The *symbols* of this second sub-layer are the following :



with the following *interaction rules* :

- Crosses are superimposed on the nuclei.
- Simple arrows are transmitted on the two sides in the direction of the arrow, unless the next symbol is blank in the previous sublayer.
- Arrows can not meet dark gray corners (red symbols in the structure layer with counter one), or a symbol near to a dark gray corner, or be superimposed on blank symbols in first sublayer.

Global behavior : These rules induce a synchronization net that permits synchronization between nuclei of the cells of the same order $n \geq 1$ that lie in a same greater cell. The first two rules

build the wires of the net, the other one forbid the presence of wires at wrong places. The Figure 21 shows the net in an order two cell.

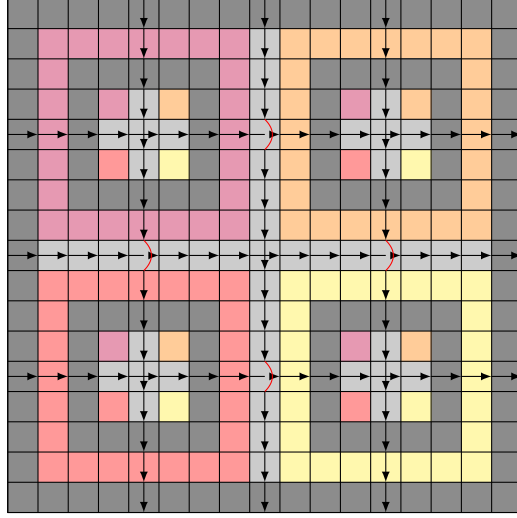


Figure 21: Example of a synchronization net in an order 2 cell, presented in section 4.5.3

The arrows will transmit information, synchronized at crosses. This way, only cells with the same order inside the same greater cell will synchronize.

4.5.4 Cells coding : the DNA

This sublayer has symbols :

$$\{\text{•••}, \text{••}, \text{•}, \text{••••}, \text{•••••}\},$$

called the **DNA** of the cell and a blank symbol.

The rules are the following :

- The DNA symbols are superimposed to the nuclei, the blank symbols to other positions.
- The DNA symbol •• is and can only be over a nucleus of an order $\leq N$ cell.
- The others DNA symbols are over $> N$ order cells.

The colors in the DNA symbols give the information of which quarters are elected to do correct computations.

4.5.5 Synchronization signals

Here we specify the signals going through the synchronization net, permitting the synchronization of the frequency bits of the same order cells that lie in another greater one. For order $n \leq N$ cells, one bit is transmitted. For order $n > N$ cells, there are three bits : one for the correct computation quarters, and two for the simulation quarters.

The *symbols* of this second sub-layer are the following : elements of $\{0,1\}$, $\{0,1\}^2$, $\{0,1\}^3$, $\{0,1\}^4$ or $\{0,1\}^6$, and a blank symbol.

The *interaction rules* are these ones :

1. **Localization of the signals** :

- The non-blank symbols are superimposed only on the net.
- Crosses with synchronization (first symbol of the alphabet of the synchronization net) have symbols in $\{0,1\}$ if the DNA symbol is red, yellow and in $\{0,1\}^3$ if not. In the first case, the bit is equal to the frequency bit of all the quarters. In the second case, the first bit has to be equal to the frequency bits of control quarters. The second bit is equal to the first simulation quarter, the third to the second one, with order being $NE < SE < SW < NW$.
- Simple arrows are superimposed with symbols in $\{0,1\}^3$ or $\{0,1\}$.
- Crosses without synchronization (second symbol of the alphabet of the synchronization net) have symbols in $\{0,1\}^6$, $\{0,1\}^3 \times \{0,1\}$, or $\{0,1\} \times \{0,1\}$. The second case happens when a wire linking order $\leq N$ cells cross a wire linking order $> N$ cells.

2. Information transfer rules :

- The symbols over the simple arrows in the net are transmitted through arrows.
- If a symbol over a cross without synchronization are in $\{0,1\}^3 \times \{0,1\}^3$, the first triple of bits is transmitted horizontally, the other one vertically. If in $\{0,1\} \times \{0,1\}^3$, the first bit is transmitted horizontally and the triple of other bits is transmitted vertically. If in $\{0,1\} \times \{0,1\}$, the first bit is transmitted horizontally and the second one vertically.

Global behavior : This way, in a cell, if $n > N$, all the order n cells contained in it share the same three bits, the first one corresponding to simulation quarters of these cells, the other two to the simulation quarters, ordered with the order $NE < SE < SW < NW$. If $n \leq N$, the order n cells share the same unique bit, corresponding to all the quarters.

4.6 Computation areas

This layer specifies the computation areas of the Turing machines that will work in the cytoplasm, that is to say each of the symbols of this layer will say if in this position, the machine have to transfer information, horizontally or vertically, or execute a step of its computation. This is done as in [Rob71]. However, in this work the constitution of the computation areas in infinite cells is not well controlled. In order to have a net gluing property for the subshift, we have to simulate in finite cells the behaviors that occur in the infinite cells. The simulation is done in the simulation quarters specified by the DNA.

The *symbols* : elements of $\{\text{in}, \text{out}\}^2 \times \{\text{in}, \text{out}\}^2$ (the destination bit ; the origin bit), a blank symbol, elements of

$$\{\text{red}, \text{yellow}, \text{red, yellow}, \text{red, red}, \text{red, yellow}, \text{yellow, yellow}\}$$

these one being called the **error signals** symbols, and elements of

$$\{\text{red}, \text{yellow}, \text{red, yellow}, \square\} \times \{\downarrow, \uparrow, \rightarrow, \leftarrow, \nearrow, \nwarrow, \searrow, \swarrow, \downarrow, \uparrow, \rightarrow, \leftarrow, \nearrow, \nwarrow, \searrow, \swarrow\},$$

these ones being called the **origin error signals** symbols.

The *interaction rules* are as follows :

1. Localization :

- Blank symbols in the synchronization layer are superimposed with blank.
- **Cytoplasm** : Cytoplasm is superimposed with symbols in $\{\text{in}, \text{out}\}^4$. The first bit of each couple is transmitted horizontally to same color symbols, the second vertically.
- **Reticle** : On the reticle, the symbols are blank or in $\{\text{red}, \text{yellow}, \text{red, yellow}, \text{red, red}, \text{red, yellow}, \text{yellow, yellow}\}$.

- The walls symbols that are not near a reticle symbol, and are not corners have a symbol in

$$\{\cdot, \circ, \bullet, \circ\} \times \{\downarrow, \uparrow, \rightarrow, \leftarrow\} \cup \{\square\},$$

with the direction (vertical or horizontal) corresponding to the direction of the wall. Wall position near a reticle one have a symbol in

$$\{\cdot, \circ, \bullet, \circ, \square\} \times \{\downarrow, \uparrow, \rightarrow, \leftarrow\},$$

with an arrow corresponding to the direction of the reticle arm, and oriented towards the nucleus. Corners are superimposed with symbols in

$$\{\cdot, \circ, \bullet, \circ\} \times \{\rightarrow, \leftarrow, \downarrow, \uparrow, \swarrow, \searrow, \nwarrow, \nearrow\} \cup \{\square\},$$

with arrows corresponding to the orientation of the corner.

The following rules are presented in the red cytoplasm. There are similar rules for the other ones.

2. **Propagation of the origin and destination bits inside the cytoplasm** : Notice that the orientation of walls (inside/outside) is differentiated by Robinson symbols. A west (resp. north) wall induce a first (resp. second) bit being **out** to its outside if the symbol is in a red cytoplasm ; An east (resp. south) wall induce a third (resp. fourth) bit being **out** to its outside if the symbol is in a red cytoplasm.

The following rules are for the propagation of error signals inside the reticle and walls.

3. **Triggering errors** : Considering a reticle symbol, if the symbol on the left has first bit **out** (horizontal signal, destination bit) and the color is red, then it contains a signal error symbol containing \cdot . Considering an wall symbol which is not near an inhibition area symbol, if the symbol on the right has third bit **out**, then it contains a signal error symbol containing \cdot . Similar rules for vertical direction.
4. **Propagation of error signals.** An error signal is propagated in the direction of the arrow (until the nucleus). When in a position near the reticle, the signal loose the arrow.
5. **Forbidding 'wrong' error signals.** In the four reticle symbols around the nucleus, there can not be a symbol that contains a color which is in the DNA. There can not be an error signal on walls near inhibition areas symbols.

These rules induce the following *global behavior* :

We can consider that in a finite synchronization area which corresponds to the information in the DNA, for cells that lie in another one, the walls transmit signals to the reticle such that signal 0 (0 corresponds to (**in, in**)) can not meet obstacles, and signal 1 (other couples of destination/origin bits) meets obstacles, namely the smaller cells included in it. This way the positions marked with (0,0) constitute the **computation area** of the Turing machine that will work in this synchronization area. Those marked with (1,0) are **vertical transfers of information**, and (0,1) are **horizontal ones**.

In the other two, the only condition is that 0 signals can not meet obstacles. If a 1 signal does not meet obstacles (this is specified by the symbols in the second and third sets of the alphabet) then an error signal is sent through walls and reticle to the nucleus. This signal permits to well constitute the computations areas in the two DNA synchronization areas of a cell. The Figure 22 shows a well constituted computation areas in the red synchronization area of order three. The Figure 23 shows possible errors signals for the computation areas of an order two cell.

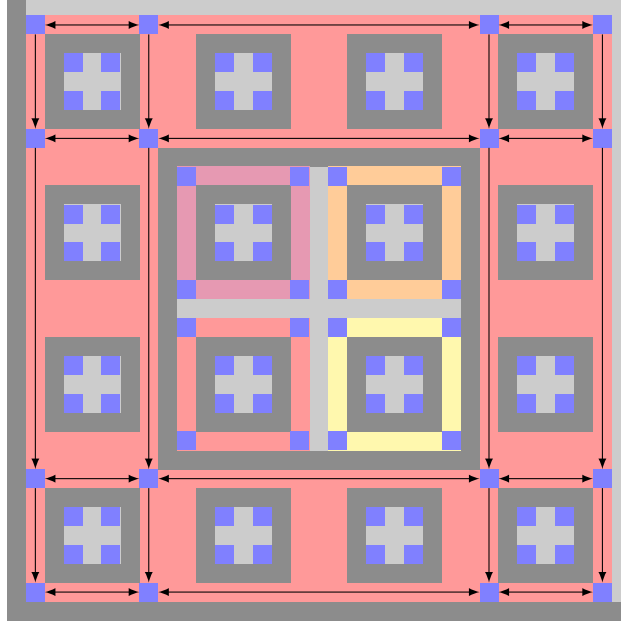


Figure 22: Example of a computation area on an order 3 cell, as presented in 4.6. The computation places are colored in blue, and the arrows designate zones where there is vertical/horizontal transfer.

4.7 The machines ('RNA')

We describe the machine layer as three sub-layers : one for the computations of the machines, one for the data, and one for error signals.

4.7.1 Initialization of the machines

In this section, we describe how and where are initialized the various machines.

The direction of time and space will depend on the color of the cytoplasm where the machine evolve :

- In the red one, time goes downwards and space leftwards.
- In the yellow one, time goes downwards and space rightwards.
- In the orange one : upwards, rightwards.
- In the purple one : upwards, leftwards.

We describe the rules of this **computation sub-layer** in the **orange cytoplasm**, for the directions of time and space are usual in this one. The rules (and symbols) for other colors are obtained by symmetry. The *symbols* are in the set :

$$(\mathcal{Q} \times \mathcal{A} \times \{\leftrightarrow\} \times \{i, ni\}) \cup (\mathcal{A} \times \{\leftarrow, \leftrightarrow, \rightarrow\} \times \{i, ni\}) \cup (\mathcal{Q} \times \mathcal{A} \times \{\leftrightarrow\}) \cup (\mathcal{A} \times \{\leftarrow, \leftrightarrow, \rightarrow\}) \cup \{\square\}$$

the set \mathcal{A} being the alphabet and \mathcal{Q} the state set of the machine. The arrows serve to initialize a unique machine in a quarter.

The *rules* are :

- **Localization** : The walls and the reticle positions are superimposed with the blank symbol.

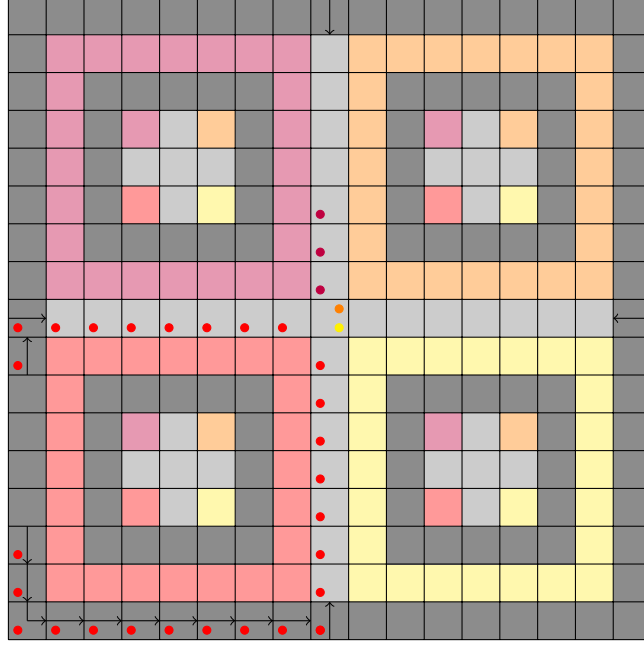


Figure 23: Examples of possible computation areas error signals on an order two cell, as presented in 4.6

- The symbols in the cytoplasm which are not information transfer places or computation places are superimposed with symbols in $\{\leftarrow, \rightarrow\} \times \{\mathbf{i}, \mathbf{ni}\}$.
- The transfer places and computation places are superimposed with symbols in $(\mathcal{Q} \times \mathcal{A} \times \{\leftrightarrow\}) \times \{\mathbf{i}, \mathbf{ni}\}) \cup (\mathcal{A} \times \{\leftarrow, \leftrightarrow, \rightarrow\} \times \{\mathbf{i}, \mathbf{ni}\})$ or in $(\mathcal{Q} \times \mathcal{A} \times \{\leftrightarrow\}) \cup (\mathcal{A} \times \{\leftarrow, \leftrightarrow, \rightarrow\})$.
- **Initialization in control quarters** : the nucleus induces the symbol $(q_0, \square, \leftrightarrow, \mathbf{i})$ in its north east if the color orange is in the DNA symbol.
- The symbols \mathbf{i}, \mathbf{ni} are present over computation places only if above a wall symbol.
- These symbols propagate horizontally inside the cytoplasm (this way only the first line of a cytoplasm can be marked with this type of symbol).
- A computation place which is marked with \mathbf{i} has a letter in \mathcal{A} being \square .
- **Loneliness of the machine in a synchronization area** : The directions of the outgoing/incoming arrows have to match inside a line.

The *Global behavior* : In the two quarters represented in the DNA, there is a machine initialized in the position near to the nucleus, and the first line of computation places have \mathcal{A} symbols only blank. This allows the initialization of a machine at some arbitrary position, or no machine at all, in the simulation quarters, and with arbitrary \mathcal{A} symbols in the first line.

4.7.2 A non connected tape Turing machine model

We keep going on the description of the rules, with dynamic rules of the machine which take into account the topology of the computation area. We use the symbolism of [Rob71]. They are as follows :

- **Loneliness** : The double arrow transmitted through the trajectory of a machine.
- On the computation places, there are $\mathcal{Q} \times \mathcal{A} \times \{\leftrightarrow\}$ symbols or $\mathcal{A} \times \{\leftarrow, \rightarrow\}$ symbols. According to the symbol there are the following behaviors (**dynamics of the machines**) :
 1. If the symbol is in $\mathcal{A} \times \{\leftarrow, \rightarrow\}$, or in $\{q_f\} \times \mathcal{A} \times \{\leftarrow, \rightarrow\}$, the \mathcal{A} symbol is transmitted upwards with the arrow, and the machine stops (if there is one).
 2. If the symbol is $(q, a, f) \in (\mathcal{Q} - \{q_f\}) \times \mathcal{A} \times \{\leftrightarrow\}$ and the \mathcal{Q} symbol comes from the sides or is initialization state, (q, a) is transmitted upwards without change.
 3. If the $(\mathcal{Q} - \{q_f\})$ symbol comes from below, then the new state, according to the transition rule of the machine, is transmitted in the direction specified by the transition function, and the new symbol is transmitted upwards (until meeting a new computation place).

Over the computation places of a quarter, one can see a parts of the space-time diagram of the Turing machine. This model allows a machine to disappear at some point and re-appear after. This phenomenon can not happen when the machine is well initialized (as in the quarters elected for correct computation).

4.7.3 Data

The machine we program here will write bits on its tape that will be compared to the frequency bits (specific to a synchronization area) - which color corresponds to the DNA information - of the cells included in the one in which the machine is initialized. As we described it, it does not have direct access to it. This layer serves to transfer this data from the nuclei so that the machine can have access to it. This **data sub-layer** have the following **symbols** : 0, 1 and a blank symbol, and the **rules** are the following :



- **Localization** : Only colored symbols in L_{syn} with second bit 0 in the L_{comp} , walls and reticle symbols are superimposed with 0 or 1 (meaning the columns which do not intersect a smaller cell).
- **Areas sharing the same bit** : The bits are transmitted vertically inside a synchronization area, and through walls and reticle.
- **Nature of the information** : The bit on the nucleus of each cell is equal to the maximum of the two bits corresponding to synchronization areas of this cell which color does not correspond to the DNA information.
- **Data access for the machine** : When a color symbol has on its right a wall symbol and further on the right a reticle symbol, or directly a reticle symbol, its bit is equal to the one of these two (resp this one).

The *Global behavior* : the data consists in frequency bits written on 'free' columns in the cytoplasm (those which do not intersect a smaller cell), which are arranged as follows : the i th of these columns, from the reticle to the walls, contains the supremum of the two DNA synchronization areas of the cells of order c_i included in the one where the machine is initialized which are intersected by the column just on the right, and $(c_i)_{i \geq 1} = (1213121412...)$. As a consequence, the machine finds the bits of the cells of order i in the column 2^i of the cytoplasm. The Figure 30 shows the localization of the data in a quarter of an order 3 cell.

4.7.4 The machine

Let (s_k) be a computable sequence of infinite words in $\{0, 1\}^{\mathbb{N}}$. The machine will work as follows : it writes successively the k first bits of s_k at positions 2^j , $j = 0..k$ for k from 0 to infinity. It has no halting state, and two tapes, one for computations, and one for writing. In the writing tape, the blank symbol corresponds to the symbol 1. Hence in a correct computation quarter, the writing tape is initialized with only 1 symbols.

4.7.5 Error signals

The error signal sub-layer has two *symbols* :  and .

The *rules* :

- The green symbol can be superimposed only on the location of the head of a Turing machine, and propagate to posterior and anterior location of the machine head (this is the error signal).
- If the machine is about to write a symbol that is smaller than the corresponding symbol in the data sub-layer, then it is in error state.
- Near the nucleus, there can not be an error signal in an area corresponding to DNA information.

This way in the two areas of cell corresponding to the DNA information, all the bits in the data tape have to be smaller than the bits written by the machine. The error signal propagate through all the trajectory of a machine. Note that the parasitic entropy it causes does not depend on the particular machine we use.

Example 11. *Here is a simple example of a possible trajectory of a machine in a red order two synchronization area. The machine is as follows : it has internal states set $Q = \{a, b\}$, the symbols in A are 0, 1 and a blank symbol (corresponds to no symbol on the picture). The transition function is simply $a \mapsto b$ and $b \mapsto a$ with the machine going only to the left and when in state a writes 1 and 0 when in state b . The symbol a is the initial state (it writes 101010...). There is no halting state. There is an illustration on Figure 24.*

4.8 Basis layer

We can assume in the following that $h \in [0, 1]$, for the reason that if we can realize Π_1 computable numbers as entropies of net gluing \mathbb{Z}^2 -SFTs, then it is also true for all Π_1 -computable numbers, taking the product of any of a subshift constructed for a number in $[0, 1]$ with a full shift over 2^k symbols, for some $k \geq 1$.

Symbols in this layer are 0, 1 and 1'. The rules are that

- Over blue symbols, the bit is 0 if the frequency bit is 0 and 1 or 1' if the frequency bit is 1.
- Over an inhibition area, the bit is 0.
- Let i the integer such that $i/4 \leq h < (i+1)/4$. The non-blue symbols are marked with symbols in $\{0, 1, 1'\}$ such that over a 2-block that does not intersect an inhibition area there are at most i times a symbol 1 or 1'.

We call **controlled bits** the ones of this layer that are over blue symbols in the structure. We call **structure bits** the ones over the other structure symbols.

4.9 Entropy

We now turn to entropy considerations. The entropy is not only due to structure and control bits of the basis layer but also various other phenomena : error signals, possible initial positions of the machines, etc. Hence the entropy of X is the sum of a 'frequency entropy' and a 'parasitic entropy'. In this section, we choose N such that this parasitic entropy, which is computable, is smaller than h , and then choose the sequence $(s_k)_k$ such that the total entropy is h .

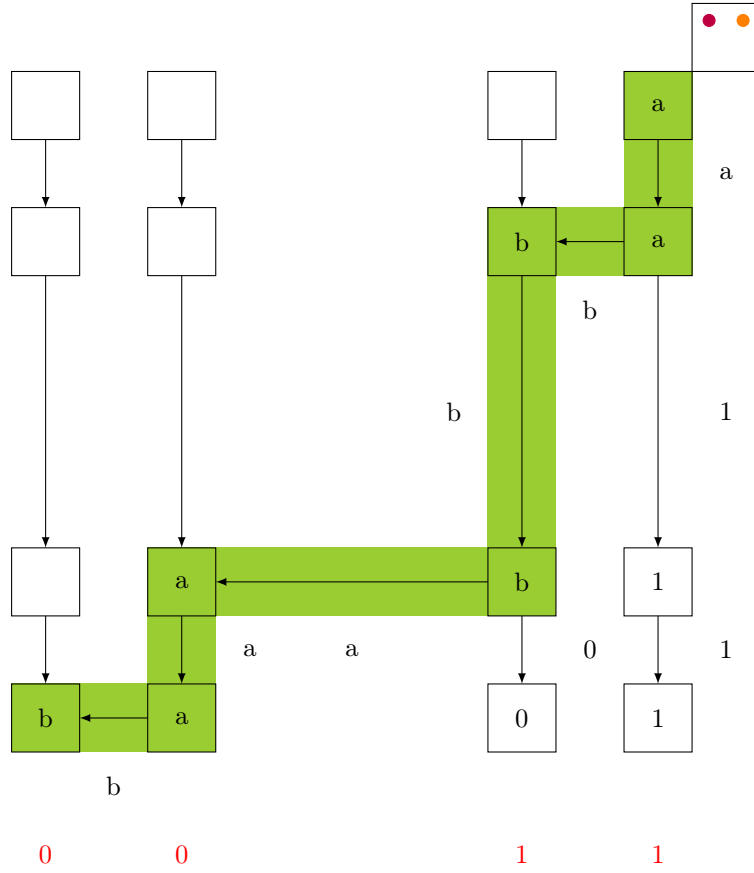


Figure 24: An example of an admissible trajectory of a machine (11). Squares designate the computation places, arrows information transfers. The data are written under.

4.9.1 Computation of the entropy and patterns frequencies

We present in this section a technical proposition used for the computation of $h(X)$.

Let us define the \mathbb{U} -frequency of a subset of \mathbb{Z}^2 , for \mathbb{U} a finite subset of \mathbb{Z}^2 :

Definition 6. Given a set $\Lambda \subset \mathbb{Z}^2$, \mathbb{U} a finite subset of \mathbb{Z}^2 , the \mathbb{U} -frequency of Λ in \mathbb{Z}^2 is the number :

$$f^{(\mathbb{U})}(\Lambda) = \limsup_M \frac{\#\{u \in \Lambda \mid u + \mathbb{U} \subset \llbracket 0, M-1 \rrbracket^2\}}{M^2}$$

Remark 2. Notice that for $\mathbb{U} = \{0\}$, this notion coincides with the usual notion of frequency of a subset of \mathbb{Z}^2 .

Proposition 11. Let be T some subshift on \mathbb{Z}^2 such that :

1. $T \subset U \times V$, with U a zero entropy subshift on alphabet \mathcal{A} and $V = \mathcal{B}^{\mathbb{Z}^2}$.
2. There exists a sequence $(\mathbb{U}_n)_n$ of finite subsets of \mathbb{Z}^2 , a sequence of positive numbers (f_n) , and a sequence $G_n \subset D^{\mathbb{U}_n}$ such that for all $x \in U$, there exists a sequence $(\mathcal{N}_n(x))_n$ of infinite subsets of \mathbb{Z}^2 , such that
 - For all x , the sets $u + \mathbb{U}_n$, where $n \in \mathbb{N}$ and $u \in \mathcal{N}_n(x)$ are disjoint.

- For all x and n , $f_n = f^{\mathbb{U}_n}(\mathcal{N}_n(x))$, and for all $n \in \mathbb{N}$, the sequence

$$\frac{\#\{u \in \mathcal{N}_n(x) \mid u + \mathbb{U}_n \subset \llbracket 0, M-1 \rrbracket^2\}}{M^2}$$

converges uniformly w.r.t. x towards f_n . (f1)

- The following sequence converges towards 1 uniformly w.r.t. x :

$$\frac{\sum_n \#\{u \in \mathcal{N}(x)_n \mid u + \mathbb{U}_n \subset \llbracket 0, M-1 \rrbracket^2\} \# \mathbb{U}_n}{M^2}$$

(f2)

- $T = \{(x, z) \in U \times V \mid \forall n \in \mathbb{N}, \forall u \in \mathcal{N}_n, z_{u+\mathbb{U}_n} \in G_n\}$.

Then the entropy of T is equal to $h(T) = \sum_{n \in \mathbb{N}} f_n \log_2(\#G_n)$.

Proof. The number t_M of M -blocks in the language of T verifies

$$\prod_n (\#G_n)^{\min_x \#\{u \in \mathcal{N}_n(x) \mid u + \mathbb{U}_n \subset \llbracket 0, M-1 \rrbracket^2\}} \leq t_M$$

$$\begin{aligned} t_M &\leq u_M 2^{M^2 - \min_x \sum_n \#\{u \in \mathcal{N}(x)_n \mid u + \mathbb{U}_n \subset \llbracket 0, M-1 \rrbracket^2\} \# \mathbb{U}_n} \\ &\quad * \prod_n (\#G_n)^{\max_x \#\{u \in \mathcal{N}_n(x) \mid u + \mathbb{U}_n \subset \llbracket 0, M-1 \rrbracket^2\}} \end{aligned}$$

where u_M designates the number of M -blocks in the language of U . Hence :

$$\min_x \sum_n \frac{\#\{u \in \mathcal{N}(x)_n \mid u + \mathbb{U}_n \subset \llbracket 0, M-1 \rrbracket^2\}}{M^2} \log_2(\#G_n) \leq \frac{\log_2(t_M)}{M^2}$$

and

$$\begin{aligned} \frac{\log_2(t_M)}{M^2} &\leq \frac{\log_2(u_M)}{M^2} + \max_x \sum_n \frac{\#\{u \in \mathcal{N}(x)_n \mid u + \mathbb{U}_n \subset \llbracket 0, M-1 \rrbracket^2\}}{M^2} \log_2(\#G_n) \\ &\quad + \left(1 - \frac{\min_x \sum_n \#\{u \in \mathcal{N}(x)_n \mid u + \mathbb{U}_n \subset \llbracket 0, M-1 \rrbracket^2\} \# \mathbb{U}_n}{M^2}\right) \end{aligned}$$

Because $\sum_n f_n \log(\#G_n) < +\infty$, we can use the dominated convergence theorem, we have equality $h(T) = \sum_{n \in \mathbb{N}} f_n \log_2(\#G_n)$. Indeed, $\sum_n f_n \log(\#G_n) \leq \sum_n f_n \# \mathbb{U}_n = 1$, because of the uniform convergence condition. \square

4.9.2 Frequency properties of the Robinson subshift

For each x a configuration of the Robinson subshift, denote $\mathcal{N}_n(x) = u(x) + 2.4^n \mathbb{Z}^2$ for some $u(x) \in \mathbb{Z}^2$, the set of $z \in \mathbb{Z}^2$, such that x_z is the nucleus of an order n cell, and \mathbb{U}_n the set of the relative positions to the nucleus of an order n cells of its proper elements. Figure 25 shows the set \mathbb{U}_2 . Moreover, define $f_n = 1/4.16^n$.

Let us prove that these sets verify the frequency conditions (f1) and (f2) of the Proposition 11.

First, for every $M \in \mathbb{N}$, and every k (in particular $k = 2.4^n$), the square $\llbracket 0, M-1 \rrbracket^2$ contains i^2 disjoint squares $k \times k$, and is contained in the union of $(i+1)^2$ such squares, where i is the greater integer smaller than M/k . Hence,

$$\frac{(M/(2.4^n) - 1)^2}{M^2} \leq \frac{\#\{u \in \mathcal{N}_n(x) \mid u + \mathbb{U}_n \subset \llbracket 0, M-1 \rrbracket^2\}}{M^2} \leq \frac{(M/2.4^n + 1)^2}{M^2}$$

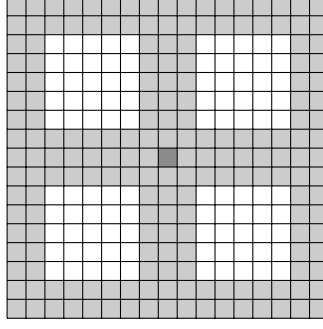


Figure 25: The set \mathbb{U}_2 is colored in gray. The $(0,0)$ position is colored dark.

for all $n \in \mathbb{N}$. Hence this number converges uniformly in x towards $1/(2.4^n)^2 = 1/4.16^n$.

The second condition to verify is that the fraction of the total area occupied by cells contained in the square $\llbracket 0, M-1 \rrbracket^2$ converges towards 1 uniformly in x . This is sufficient to prove that for all $\epsilon > 0$, for all $n > 0$, there exists some M_n such that for all $M \geq M_n$, and for all x , the fraction of the area occupied by cells of order $\leq n$ in the square $\llbracket 0, M-1 \rrbracket^2$ is greater than $1 - \epsilon$. This number is greater than $\frac{(M/(4^n+1)-1)^2 * (4^n+1)^2}{M^2}$ for all x , hence we have directly this property.

From this we deduce also that the fraction of the area occupied by order $\geq m$ cells for a fixed m tends uniformly to 1, hence the frequency of the set of positions of the nuclei of cells not included in a another that have order $\geq m$ is zero in every configuration, for all m .

4.9.3 Computing the entropy of X

Let $m > N$ be some integer, and s a Π_1 -computable sequence in $\{0, 1\}^{\mathbb{N}}$. Let us denote X_m obtained from X by the following modification : the bits of the basis layer verify only the condition that the bits in an order $i \leq m$ cell's synchronization area represented in the DNA information are 0 if $s_i = 0$. There exists some $C(m) > 0$ such that all these conditions are verified for cells that lie in an order $\geq C(m)$ cells.

We will apply the proposition of the previous section to the subshift X_m and with the following parameters :

- U is the structure layer, and \mathcal{B} is the product of the alphabets of the other layers.
- The sets \mathbb{U}_n , $n \geq 1$ are equal the set of relative positions to the nucleus of the proper elements of an order n cell.
- The sets $\mathcal{N}_n^{(m)}(x)$, $n \geq 1$, are equal to the set of the positions of the nuclei of cells lying in an order $\geq C(m)$ cell.
- The frequencies : $f_n = 1/4.16^n$. They verify the conditions of Proposition 11 because the frequency of the nuclei positions of cells that don't lye in an order $\geq C(m)$ cell is 0.
- The sets $G_n^{(m)}$ corresponds to the possible patterns in the non-structure layers over some $u + \mathbb{U}_n$, $u \in \mathcal{N}_n^{(m)}(x)$.

We have, using the previous proposition, that :

$$h(X_m) = \sum_{n \geq 1} f_n \log(|G_n^{(m)}|)$$

Because $X = \bigcap_m X_m$, and by the dominated convergence theorem, the entropy of X is given by :

$$h(X) = \sum_{n \geq 1} f_n \log(|G_n|)$$

where G_n is the set of possible patterns over some $u + \mathbb{U}_n$, corresponding to a cell which is included into arbitrary order cells.

The numbers $|G_n|$ can be decomposed in the following way :

$$|G_n| = N_n^1 * N_n^2 * N_n^3 * N_n^4,$$

where N_n^1 is the number of possible patterns of structure bits over some $u + \mathbb{U}_n$ corresponding to an order n cell, N_n^2 is the number of possible patterns of controled bits in correct quarters, N_n^3 the number of possible patterns of structure bits, and N_n^4 is the number of possible patterns in the non structure and non basis layers.

$$h(X) = \sum_{n \geq 1} f_n \log(N_n^1) + \sum_{n \geq 1} f_n \log(N_n^2) + \sum_{n \geq 1} f_n \log(N_n^3) + \sum_{n \geq 1} f_n \log(N_n^4)$$

The first term of the second member of this equality is the frequency of 1 structure bits in a configuration, and this frequency is $\frac{i}{4}$.

For the second term, we have $N_n^2 = 2^{\frac{1}{2}s_n 4.12^{n-1}}$ if $n > N$ and $N_n^2 = 2^{s_n 4.12^{n-1}}$ if $n \leq N$. The third one is given by $N_n^3 = 2^{\frac{1}{2}4.12^{n-1}}$, and as a consequence is less than $\sum_{n > N} \frac{2.12^{n-1}}{4.16^n} = \sum_{n > N} \frac{1}{3} \left(\frac{3}{4}\right)^n$.

The last one is a computable number (this comes from the fact that G^n is computable) which is less than $\sum_{n > N} f_n \log(|\mathcal{B}^{\mathbb{U}_n}|)$. These numbers tends towards 0 when N tends to infinity.

Thus, the entropy verifies :

$$h(X) = \frac{i}{4} + \frac{1}{3} \sum_{n=1}^N \left(\frac{3}{4}\right)^n s_n + h_N$$

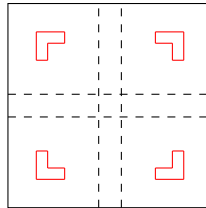
such that h_N is a computable sequence of numbers that tends to 0. Choose N such that $\frac{i}{4} + h_N < h$, and $\frac{1}{3} \sum_{n=1}^N \left(\frac{3}{4}\right)^n$ is greater than $h - \frac{i}{4} - h_N$, which is possible because this sum tends towards $1/4$, and $h - \frac{i}{4} < 1/4$. For h_N is computable and h is a Π_1 -computable number, we can choose a Π_1 -computable sequence s such that $h(X) = h$.

4.10 Linear net gluing

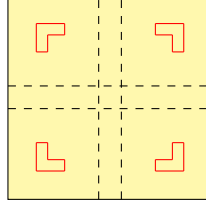
4.10.1 Completing blocks

We derive from the completion of blocks in the subshift X_{Rob} (given by proposition 3), a block completion in the subshift X . Let p be a $(2.4^n - 1)$ -block in X . All the possibilities are as follows :

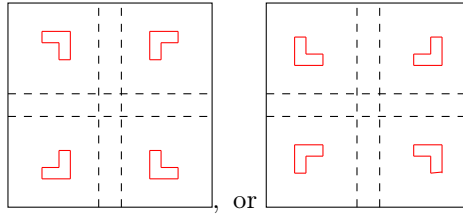
- The projection of p on the structure layer can be completed in this configuration be the union of four order $2n + 1$ supertiles separated by two segments, as :



Here the pattern p can be completed inside x , a configuration where it appears, by a quarter of an order $n + 1$ cell. There are five possibilities to color what corresponds to the cytoplasm of this $n + 1$ order cell in this quarter : blank, purple, orange, red, yellow. For instance ((i)) :



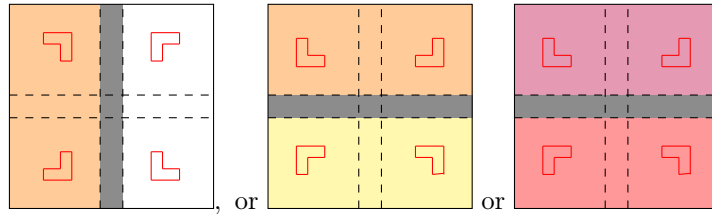
- The projection of p on the structure layer can be completed into the projection of the same configuration on the structure layer in the union of four order $2n + 1$ supertiles separated by two segments, as :



Here there are two possibilities :

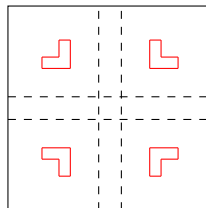
1. The north, south, east or west two supertiles are in the same order $n + 1$ or more cell, and the others two are outside. The two first have the same color (non blank) in their cytoplasm , and the other two have the same color, or the two are blank.
2. The four are in the same order $n + 1$ or more cell, but separated by an arm of the reticle. Two of them have the same color, the other two another color, such that the orientations of these color match.

For instance ((ii),iii,iv) :



These can be completed in the same configuration by two quarters with same colors separated by an arm.

- The projection of p on the structure layer can be completed in this layer by the union of four order $2n + 1$ supertiles separated by two segments, as :



In this case, there are three possibilities

1. The four supertiles are in a same order $n + 1$ or more cell, and they are separated by a reticle, and colored according to the orientation.
2. One is in a cell of order $n + 1$ or more and the others are outside. The first is colored according to orientation, and the others have the same color, or are all blank.
3. Two of the supertiles are in a same order $n + 1$ or more cell, and they are separated by a reticle arm, and colored according to the orientation. The other two are outside, and separated from the other two by walls.

These can be completed in the same configuration in four quarters with same colors.

Before proving the net gluing property, let us prove that the colored parts of each of the previous completed patterns, can be extended by an order $n + 1$ or $n + 2$ cell, or a quarter of some $n + 2$ order cell, colored blank. :

1. For the first type (for instance (i)), we can complete it with reticle, and walls : the initial positions of the machines are those in the top row (or bottom row, depending on the color), we put computation areas error signal according to the computation area, and choose DNA that do not represents the considered quarter. Hence all the completed patterns of this type can be completed into a cell.
2. For the second type, we distinguish the patterns that have a color and blank, the ones that have two colors corresponding to the limit between two synchronization area of an order $n + 1$ cell, and the others that have two colors not corresponding to these types of limits (for instance (iv)). For the formers :
 - (a) If the limit is an horizontal reticle arm, then we complete the quarters with well formed computation areas, and continuing the trajectories of the machines. Then add walls with error signals according to the quarters, and DNA symbol such that the two quarters are not represented in the DNA, and then add the other quarters.
 - (b) If the limit is an horizontal wall, the we complete the quarters with only 1 bits horizontal lines in the computation areas layer, and further walls with errors signals, DNA and other quarters. Moreover, if there is an origin error signal to add, the propagation of this signal is oriented through the wall which was not present is the initial pattern.
 - (c) If the limit is vertical, we can complete the quarters by adding machines in the initialization row, and then complete the trajectories of the machines, and well formed computation areas. Then the walls,DNA and the other quarters.

For the lateres, we have to complete one of the quarters in an order $n + 1$ cell, and the other in an order $n + 2$ cell. In the case of a wall separating a colored quarter and a blank one, we complete it into a quarter of an order $n + 2$ cell, colored blank.

3. For the third type, the completion is similar to the second type. Just notice that when the four supertiles are separated by a complete reticle, then the pattern is completed such that the walls contains no error signal.

As a consequence, every $(2.4^n - 1)$ -block in X can be completed in the language of X into a quarter of an order $n + 3$ cell, colored purple, red, yellow, orange or blank (because any $n + 1$ order cell can be completed into a quarter of some $n + 2$ cell), meaning some $2n + 5$ supertile.

4.10.2 Inhibition configurations and net gluing

All the previous patterns can be glued on a lattice relatively to each other in an inhibition configuration. We use this fact and Proposition 1 to prove the linear net gluing property for X .

The Figure 26 shows a representation of an inhibition configuration. Possible positions of the different types of completed patterns are represented in the figure, for $n = 1$ and $n=0$ for (iv).

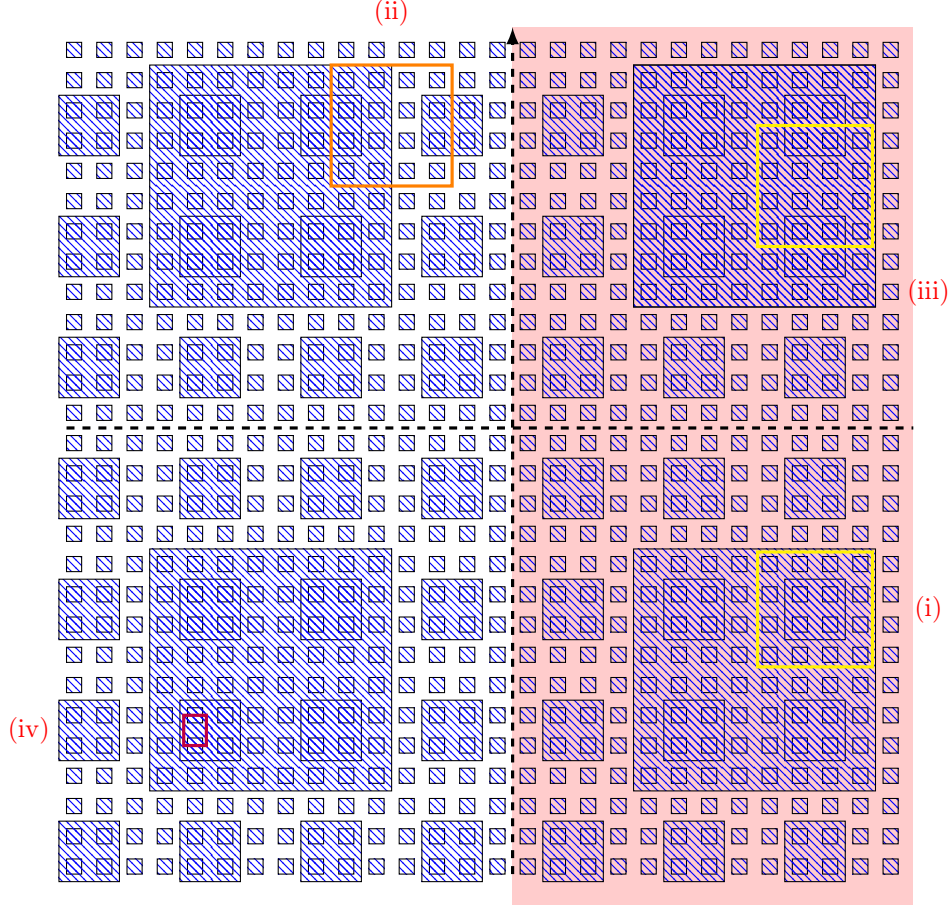


Figure 26: Examples of localisation of various completed patterns in an inhibition configuration. The dashed arrow designate the infinite column of arrows, the dashed line the line 0, and the squares designate cells. The color represent an infinite synchronization area. (Section 4.10.2)

Consider $2n + 5$ supertiles, as in Section 4.10.1. Let k, l be some integers. Let us show that there exists an inhibition configuration in X where the first cell quarter appears on $\llbracket 0, 2^{2n+6} - 2 \rrbracket^2$ and the second on $(k2^{2n+5}, l2^{2n+5}) + \llbracket 0, 2^{2n+6} - 2 \rrbracket^2$. Consider the $2n + 2m + 7$ supertiles with $4^m \geq k, l$, which do not appear in another cell. They contain centered in it an order $m + n + 3$ cell with order $2n + 5$ supertiles surrounding it with period 2^{2n+6} as in the Figure 27. Because of the choice of m , the number of these supertiles on one side of the order $m + n + 3$ cell is more than k and l .

We consider that $k > 0$ and $l \geq 0$, for the other cases are similar to prove. Let us choose to place the first $2n + 5$ supertile (containing the first pattern) at the k th position from the first one on the right under the $m + n + 3$ cell, and the second one at position l from the one on the bottom at the right of the great cell (see an illustration on Figure 27), then we complete the inhibition configuration. Hence the gluing set of the two starting $2^{2n} - 1$ block in X relatively one to the other contains some $u(p, q) + 2^{2n+6}(\mathbb{Z}^2 - (0, 0))$. Hence the subshift X is $O(id)$ -net gluing.

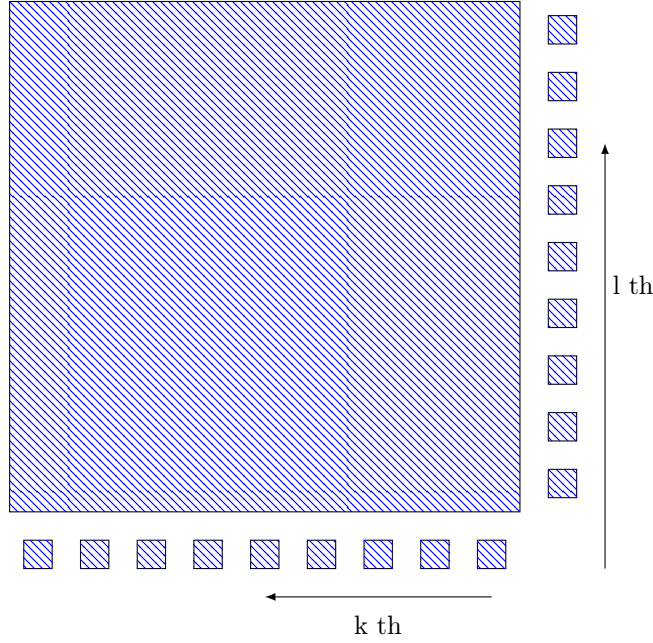


Figure 27: An illustration of the proof of the Net gluing property of X (Section 4.10.2)

4.10.3 Linear block gluing

The operators d'_A In this section, we prove that every Π_1 -computable non-negative real number is the entropy of some linearly block gluing \mathbb{Z}^2 -SFT.

In order to prove this assertion, we use a modified version of the operator d_A , denoted d'_A , so that there is an explicit expression of the transformed subshift $d'_A(Z)$ as a function of the entropy of Z , where Z is any SFT on alphabet \mathcal{A} . Let us denote $\tilde{\mathcal{A}}'$ the alphabet of $d'_A(Z)$.

Let Z be some \mathbb{Z}^2 -SFT on alphabet \mathcal{A} . We think of the subshift d'_A as having three layers. The two first ones are the same as in the definition of $d_A(Z)$, with the same interaction rules between them. We add a third layer that has alphabet $\{0, 1\}$ with the rule that a symbol 1 can appear only over a symbol \rightarrow .

The consequences on the behavior of the operator is that the patterns that have large 'blocks' of straight curves are more numerous, and this fact simplifies the computation of the entropy.

Let us denote Δ' the subshift that consists in the second and third layer together, with their interaction rules. In order to compute the entropy of $d'_A(Z)$ from the entropy of Z , we first prove bounds on the number $N_n(\Delta')$ of n -blocks in the language of Δ' .

Let us consider, given any n -block that appears in Δ' , the number of ways to extend it into some $(n+2)$ -block. Counting it will be done in four steps (which are illustrated in the example of Figure 28) :

1. Choice of adding curves of \downarrow symbols (defined in a similar way as the curves of \rightarrow in the analysis of the operators d_A) (zero, one, or two curves) : this corresponds to choose two length for block of \downarrow symbols on the north west and south east of the block (illustrated in red in the example). The number of these choices is then less than n^2 .

2. All the curves of \downarrow are extended. To a curve corresponds some $p \times 1$ (for some p) and some 1×2 patterns where the choice of how to extend the curve is made (they are colored with green in the example). For all these curves there are $3(1 + \dots + 2^{p-1}) \leq 3 \cdot 2^p$ way to extend it, where 3 stands for the number of choices for the 1×2 pattern, and where p is the maximal length of a \downarrow line which extend the \downarrow curve from above. For instance if $p = 2$, the choices are $\downarrow \rightarrow$ and $\downarrow \downarrow$, with a choice of 0 or 1 in the third layer for the first one. For this reason the number of ways to extend the sides of a block (except the corners) is less than 2^{4n} .
3. Extend all the other side positions except the corner, with \rightarrow symbols. Then choose 0 or 1 in the third layer.
4. Extend the corners.

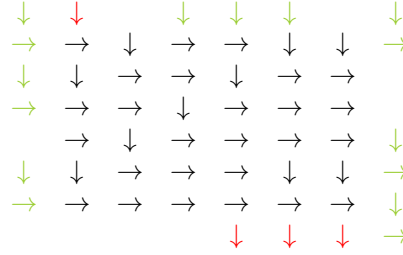


Figure 28: Illustration of three first steps of the extension of some n -block into some $(n+2)$ -block (here $n = 6$).

We deduce the inequality :

$$N_{n+2}(\Delta') \leq n^2 3^4 2^{4n} N_n(\Delta')$$

As a consequence,

$$N_{2p}(\Delta') \leq (p!)^2 3^{4p} 2^{4(2p+2(p-1)+\dots+2)} \leq (p!)^2 3^{4p} 2^{2 \cdot (2p+1)^2}$$

Furthermore, for the patterns that consists only on \rightarrow symbols in the second layer are in Δ' , we have

$$N_{2p}(\Delta') \geq 2^{(2p)^2}.$$

At this point we will need the following lemma :

Lemma 1. *Let P be some n -block in the language of Δ (first layer of Δ'). The number of curves (of \rightarrow symbols) crossing P is smaller than n .*

Proof. This is sufficient to prove that changing some symbols in a n -block in the language of Δ to add another curve of \downarrow symbols can not increase the number of \rightarrow curves that cross this pattern (for the pattern that consists in only \rightarrow symbols has exactly n curves). To see this, consider that such an addition acts as pushing some curves to the top, and this can only preserve or decrease the number of curves. \square

It follows that :

$$2^{(2p)^2} N_{2p}(Z) \leq N_{2p}(d(Z)) \leq (p!)^2 2^{5p+4} 2^{(2p)^2} N_{2p}(Z)$$

Hence the entropy of $d'_{\mathcal{A}}(Z)$ is equal to $1 + h(Z)$, hence the entropy of $d'_{\tilde{\mathcal{A}}} \circ \rho \circ d'_{\tilde{\mathcal{A}}} \circ \rho \circ d'_{\mathcal{A}}(Z)$ is $3 + h(Z)$, and this subshift is linearly block gluing. We get that every Π_1 number which is greater than 3 is the entropy of a linearly block gluing SFT.

In order to realize the numbers greater than $1/p$, for any integer $p > 1$ (hence to realize every positive number) as the entropy of a linearly block gluing SFT, we consider some non-deterministic substitutions, and how the entropy is changed by the application of it.

Some non-deterministic substitutions Let $p > 1$ some integer, and X some SFT on alphabet \mathcal{A} . Consider the subshift \tilde{X}_m which alphabet is $\mathcal{A} \times \{1, \dots, m+1\}^2$, and imposing the 1×2 patterns to be one of the following (with similar rules for the 2×1 patterns) :

$$\begin{pmatrix} (a, i, l) \\ (b, k, l) \end{pmatrix}$$

with $k = i + 1$ if $i < m$, $k = 1$ or $m + 1$ if $k = m$, $a = b$ if $k > 1$, and the pattern $\begin{pmatrix} a \\ b \end{pmatrix}$ being in the language of X if $i = m$ or $m + 1$ and $k = 1$.

This way \tilde{X}_m is obtained from X by applying a non-deterministic substitution that consists in replacing letters in A by $m \times m$, $(m + 1) \times (m + 1)$, $m \times (m + 1)$, or $(m + 1) \times m$ rectangle made of the same letter (such that in a configuration, the length or height of adjacent blocks have to be the same).

Proposition 12. *If the subshift X is f -block gluing, the subshift \tilde{X}_m is $mf(id/m)$ -block gluing.*

Proof. Let p, q two n -blocks in the language of \tilde{X} . These two blocks are sub-patterns of a $i \times j$ rectangular patterns and some $i' \times j'$ one with i, j, i', j' between n and $n + 4$, which are images of rectangular patterns in X by the non-deterministic substitution, which length and height are some $i'' + j'', i^{(3)} + j^{(3)}$ (assume $i'' + j'' > i^{(3)} + j^{(3)}$) such that $mi'' + (m + 1)j'' = i$ and $mi^{(3)} + (m + 1)j^{(3)} = j$. The second of these two patterns can be glued with distance to the first being greater than $f(i'' + j'')$ (so the number of columns or lines between the two blocks is more than $f(i'' + j'')$). Hence the two patterns p, q can be glued (applying the substitution) with a number of columns or lines between them being $mi^{(4)} + (m + 1)j^{(4)}$, with $i^{(4)} + j^{(4)} \geq f(i'' + j'')$. Hence this number can be any number greater than $mf(i'' + j'') \leq mf(i/m)$ (because $mi'' + (m + 1)j'' = i$ and f is non decreasing). \square

Let us compute the entropy of \tilde{X}_m given the entropy of X .

Proposition 13. *The entropy of \tilde{X}_m is $\frac{h(X)}{m}$.*

Proof. Let us count the number of n blocks in the language of \tilde{X}_m . Such a block appear as a sub-pattern of some $i \times j$ rectangular pattern with i and j between n and $n + 4$, which is the image by the substitution of some $(i + j) \times (i' + j')$ rectangular pattern in X , such that $mi + (m + 1)j = i$ and $mi' + (m + 1)j' = j$. We denote $C_{k,l}(Z)$ the number of $k \times l$ rectangular patterns in the language of Z . Hence we have the following inequalities :

$$\sum_{\substack{mi + (m+1)j = n \\ mi' + (m+1)j' = n}} N_{i+j, i'+j'}(X) \leq N_n(\tilde{X}_m)$$

$$N_n(\tilde{X}_m) \leq \sum_{\epsilon \in \{0, \dots, 4\}^2} (1 + \epsilon_1)(1 + \epsilon_2) \sum_{\substack{mi + (m+1)j = n + \epsilon_1 \\ mi' + (m+1)j' = n + \epsilon_2}} N_{i+j, i'+j'}(X)$$

As a consequence, for the reason that if $mi' + (m + 1)j' = n$ then $n/m \geq (i' + j') \geq n/(m + 1)$,

$$\frac{n}{m+1} N_{\lfloor n/(m+1) \rfloor}(X) \leq N_n(\tilde{X}_m) \leq 5^4 \frac{n}{m} N_{\lceil (n+4)/m \rceil}(X)$$

Thus,

$$\frac{\log_2(\frac{n}{m+1})}{n} + \frac{\lfloor n/(m+1) \rfloor}{n} \frac{\log_2(N_{\lfloor n/(m+1) \rfloor}(X))}{\lfloor n/(m+1) \rfloor} \leq N_n(\tilde{X}_m) \leq \frac{\log_2(5^4 \frac{n}{m})}{n} + \frac{\lceil (n+4)/m \rceil}{n} \frac{N_{\lceil (n+4)/m \rceil}(X)}{\lceil (n+4)/m \rceil}$$

This means that the entropy of \tilde{X}_m is $h(X)/m$. \square

Hence, by applying this operator, we get that every number greater than $1/m$ is the entropy of some linearly block gluing \mathbb{Z}^2 -SFT, for every integer $m > 1$. This ends the proof of theorem 4.

References

- [AS13] Nathalie Aubrun and Mathieu Sablik. Simulation of Effective Subshifts by Two-dimensional Subshifts of Finite Type. *Acta Appl. Math.*, 126:35–63, 2013.
- [DRS12] Bruno Durand, Andrei Romashchenko, and Alexander Shen. Fixed-point tile sets and their applications. *J. Comput. System Sci.*, 78(3):731–764, 2012.
- [Hed69] G.A. Hedlund. Endomorphisms and automorphisms of the shift dynamical system. *Mathematical systems theory*, 3, 1969.
- [HM10] Michael Hochman and Tom Meyerovitch. A characterization of the entropies of multidimensional shifts of finite type. *Annals of Mathematics*, 171(3):2011–2038, 2010.
- [Hoc09] Michael Hochman. On the dynamics and recursive properties of multidimensional symbolic systems. *Invent. Math.*, 176(1):131–167, 2009.
- [JV13] E. Jeandel and P. Vanier. Turing degrees of multidimensional SFTs. *Theor. Comput. Sci.*, 505:81–92, 2013.
- [JV14] Emmanuel Jeandel and Pascal Vanier. Characterizations of periods of multidimensional shifts. *Ergodic Theory and Dynamical Systems*, *accepted.*, 2014.
- [LM95] Douglas A. Lind and Brian Marcus. *An Introduction to Symbolic Dynamics and Coding*. Cambridge University Press, New York, NY, USA, 1995.
- [PS14] Ronnie Pavlov and Michael Schraudner. Entropies realizable by block gluing shifts of finite type. *Journal d’Analyse Mathématique*, to appear, 2014.
- [PS15] Ronnie Pavlov and Michael Schraudner. Entropies realizable by block gluing zd subshifts of finite type. *Journal d’Analyse Mathématique*, 126(1):113–174, 2015.
- [Rob71] R. Robinson. Undecidability and nonperiodicity for tilings of the plane. *Inventiones Math.*, 12:177–209, 1971.
- [Ser14] Thomas Serre. Hierarchical models of the visual system. In *Encyclopedia of Computational Neuroscience*, 2014.
- [SG16] B. Hellouin S. Gangloff. Computing the entropy of mixing subshift. 2016.
- [Sim11] S. G. Simpson. Medvedev degrees of 2-dimensional subshifts of finite type. *Ergodic Theory and Dynamical Systems*, 2011.
- [XZ01] K. Weihrauch X. Zheng. The arithmetical hierarchy of real numbers. *Mathematical Logic Quarterly*, 47:51–66, 2001.

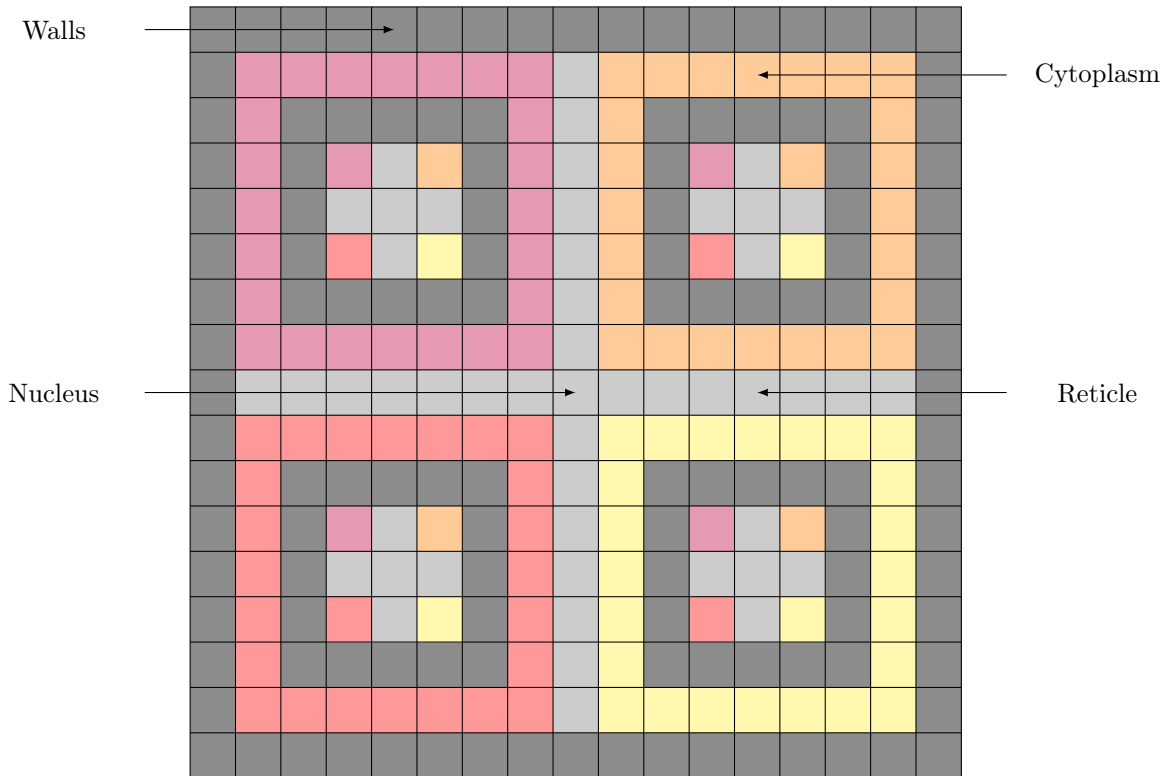
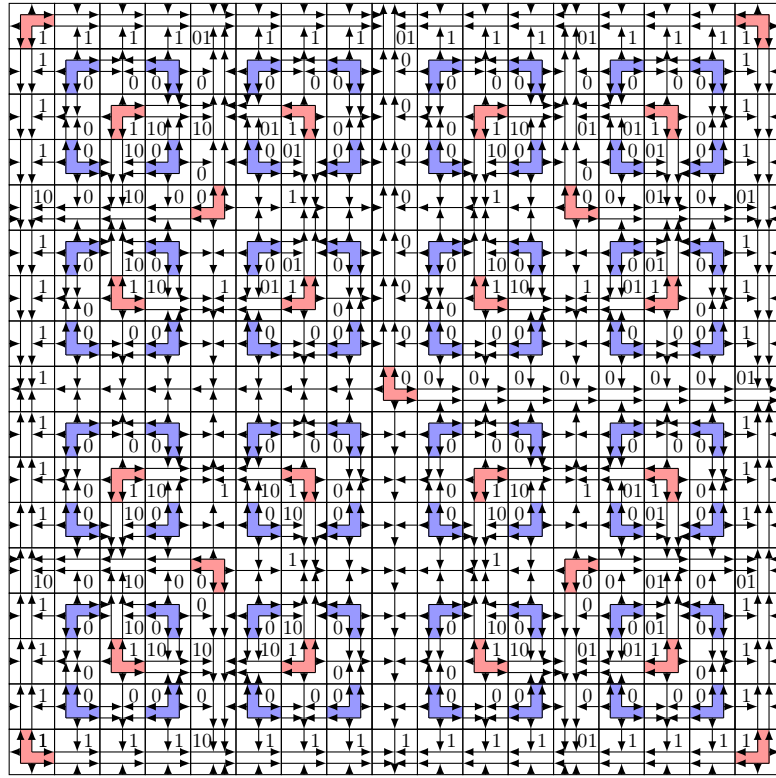


Figure 29: Example of the projection over the structure (first picture) and the synchronization (second one) layers of a pattern over an order 2 cell, presented in 4.4.

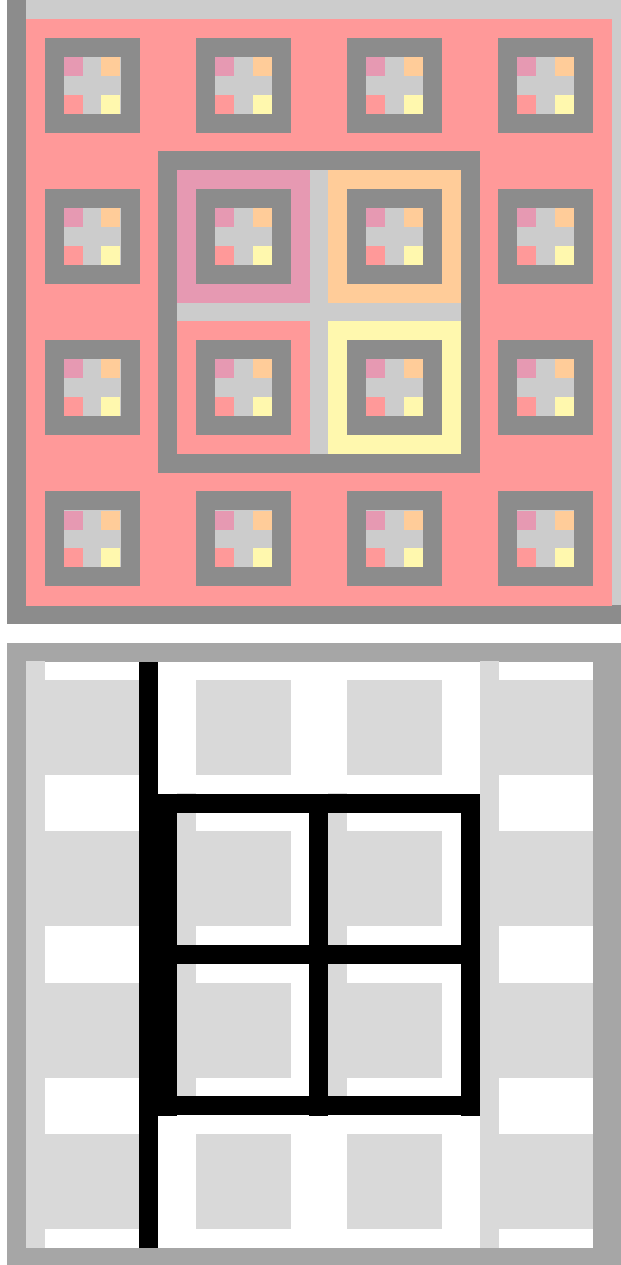


Figure 30: Example of localization of the machine data on an order 2 cell. The dark symbol stands for the order three cells data, the black symbol for order two cells data, and the light gray for order one cells data.