Vertical-Horizontal Full Compatibility of one-dimensional Subshifts

Arthur Mittelstaedt[1] and Gaétan Richard[1]

Normandie Univ, UNICAEN, ENSICAEN, CNRS, GREYC, 14000 Caen, France
{arthur.mittelstaedt,gaetan.richard}@unicaen.fr

Abstract. We study the notion of full-compatibility: given two one-dimensional subshifts H and V, is there a two-dimensional subshift X such that $H = \{c_{|\mathbb{Z}\times\{0\}} \mid c \in X\}$ and $V = \{c_{|\{0\}\times\mathbb{Z}} \mid c \in X\}$? We show that this problem is decidable when both X and Y are nearest-neighbor SFTs but undecidable when at least one is allowed to be of slightly higher complexity. We also prove that the problem is undecidable for three nearest-neighbor SFTs combined in a three-dimensional subshift.

Keywords: Subshifts, Wang tiles, Undecidability, Domino Problem, Tilings, Subshifts of Finite Type

1 Introduction

In science, local constraints or rules are often well understood; however, their global implications are often difficult or impossible to determine. To better study the link between locality and globality, one can look at a formal simplified and regular case. One example of such a context consists of using a regular line (or grid) where each position is endowed with a state chosen among a finite set, and look at the implication of local constraints given by a set of forbidden words.

On the discrete line (dimension 1), this framework corresponds to an infinite word version of formal languages and has been heavily studied and its basics are now well understood [11]. Most problems relating locality and globality are decidable.

On the grid (dimension 2), this framework corresponds to *tilings*. In this case, one surprising thing is that simple local constraints can lead to very complex constructions. A first example is the existence of an aperiodic set of Wang tiles [3]. This kind of "strange" object has been proven fruitful when physicists have encountered quasicrystals. Further studies have found and constructed many other variants of such elements using self-similarity [10, 5], minimizing the size of constraints [4, 7], or using geometry [9, 12]. Depending on the construction, the understanding of the global structure is more or less complete.

With advances in dimension two, one current axis of research is to explore its links with the one dimensional case. One recent example of such result is the one from N. Aubrun and M. Sablik [2] that proves that any one dimensional infinite language from a large "complex" class (formally, whose forbidden patterns set is semi-computable), can be seen as the projection of a "simple" two dimensional one (with a finite set of forbidden patterns). One other example is the work of S. Esnay [6] and in particular [1] on the computational complexity of two-dimensional tilings when horizontal constraints are fixed.

In this paper, we study the following question: given **two** one-dimensional infinite languages, can both of them appear in a two dimensional tiling (horizontally and vertically). We also add one extra demand that **every** word appears: we call this problem *full compatibility*. The paper is divided as follows: In section 2, we give the formal definition of full compatibility and several basic and meaningful examples. Then, we present one case where full compatibility is decidable (section 3) and several cases where it is undecidable (section 4).

2 The Full Compatibility Problem

2.1 Subshift and traces

The elementary objects studied are the members of the set $\Sigma^{\mathbb{Z}^d}$, given d a dimension and Σ a finite alphabet. We call this set the full-shift over Σ and its elements are called *configurations*. When $d = 1$ we alternately call them bi-infinite words or just words. The elements of $\mathbb{Z}^d$ are called *positions* or vectors.

If c is a 2D configuration ($d = 2$), and (x, y) is a position, then $c_{(x,y)} = c(x, y)$ refers to the letter at position (x, y). This space is endowed with the distance defined by $d(c, c') = 2^{\min(\{|i| \ | c_i \neq c'_i\})}$. The resulting space is metric and compact. The *shift* is the action of an elementary vector $v : \sigma_v(x)_p = x_{p+v}$.

These objects are intuitively considered as tilings, and since they are formally defined using functions we will use some notations usually employed for functions. Especially, we often use the restriction notation to refer to a precise part of a configuration. Let us say that c is a 2D configuration, $c_{|\mathbb{Z}\times\{0\}}$ corresponds to the tiled horizontal line at height 0 (resp. $c_{|\{0\}\times\mathbb{Z}}$ is the central column).

A *pattern* is a function of P^Σ where P is a finite part of $\mathbb{Z}^d$, called the supporting set of the pattern. A pattern m *appears* in a configuration c if $\exists v \in \mathbb{Z}^d, \sigma_v(c)_{|P} = m$, a configuration avoids a set of forbidden patterns if none of its elements appears in it. For convenience, in dimension 2, we identify patterns and tuples.

Definition 1 (Subshift). *A* subshift *is the subset of configurations of* $\Sigma^{\mathbb{Z}^d}$ *avoiding* $\mathcal{F}$*, a certain set of forbidden patterns.*

The name subshift comes from the fact that these sets are stable under the shift operation. In fact subshifts are exactly the subsets of $\Sigma^{\mathbb{Z}^d}$ which are stable under the shift and topologically closed. Note that there is always several sets of forbidden patterns to describe a single subshift.

We also introduce some elementary notations to describe words. We use the symbol ω as an exponent on the left or on the right of a letter to symbolize an infinite repetition. For example if a and b are letters and x is some finite word then ${}^\omega a x b^\omega$ stands for a word with an infinite number of successive a followed by x, again followed by an infinite repetition of the letter b. Note that even if x is fixed, this description formally corresponds to an infinity of words that are just shifted versions of one another. We often abusively write $c = {}^{\omega}axb^{\omega}$ to say that c is one of the words described by ${}^{\omega}axb^{\omega}$. Note that the set of words corresponding to ${}^{\omega}axb^{\omega}$, despite being stable under the shift, is not a subshift as it is not necessarily closed, we should add the words ${}^{\omega}a^{\omega}$ and ${}^{\omega}b^{\omega}$ to obtain a subshift.

Definition 2. *We can distinguish several types of subshifts according to the complexity of their set of forbidden patterns:*

- *A subshift is* of finite type *(**SFT** for short) if it is defined by a finite set of forbidden patterns;*
- *Such a subshift is* nearest-neighbor SFT *(**NNSFT**) if it can be described by a set of forbidden patterns whose supporting sets consists of exactly two adjacent positions. For one-dimensional subshifts this is equivalent to $\mathcal{F} \subseteq \Sigma^{\{0,1\}}$.*
- *A subshift is* sofic *if it is a letter-to-letter projection of an SFT.*
- *A subshift is* effective *if its set of forbidden patterns is recursively enumerable.*

All these definitions are usual and widespread ones [8]. The nearest-neighbor notion is a bit more specific case but appears naturally [1]: on the line, NNSFT correspond to (bi-infinite) traces of states encountered marching along a finite graph (and also Rauzy graph of order one); on the plane, *Wang* tilings [13] are a classical example of NNSFT. In many cases, the additional size restriction of the size of patterns is not significant as any SFT is equivalent up to conjugacy to a NNSFT (up to scaling). In this paper, this scaling operation is significant and will impact results.

2.2 Full-Compatibility

One classical question on subshifts is the following: "Given a finite set of forbidden patterns, is the subshift avoiding it non-empty?". This problem is computable on the line but undecidable on the plane [3]. Due to the separation between these two cases, one can study the link between these two dimensions. A result by N. Aubrun and M. Sablik [2] states that any effective 1D subshift can be seen as the set of lines of a 2D sofic subshift. More recently, S. Esnay has studied the decidability of 2D subshift given horizontal constraints.

In this paper, our point of view is to give **both** the horizontal and the vertical languages and see if they can be achieved by a 2D subshift simultaneously. More formally, given two one-dimensional subshifts H and V is there a two-dimensional subshift X whose set of lines is exactly H and whose set of columns is exactly V.

Definition 3 (Fully-Compatible subshift).

Two one-dimensional subshifts H and V with common alphabet are fully-compatible *if there exists a two-dimensional subshift X such that $H = \{c_{|\mathbb{Z}\times\{0\}} \mid c \in X\}$ and $V = \{c_{|\{0\}\times\mathbb{Z}} \mid c \in X\}$.*

For following sections, where the decidability of the problem will be studied we will consider the necessary condition that all letters that appears in some configuration of H also appear in some configuration of V.

2.3 Basic properties and examples

This first formulation motivates the use of "compatibility" in the name of the problem and show the goal of the study but it hides a clearer way to present this question. Indeed, it is easy to verify that the set $X = \{c \in \Sigma^{\mathbb{Z}^2} \mid \forall i, c|_{\mathbb{Z}\times\{i\}} \in H, c|_{\{i\}\times\mathbb{Z}} \in V\}$ is a subshift, as it can be obtained by combining forbidden patterns from H and verticalized forbidden patterns from from V. We call X the *combiner* of H and V. The compatibility problem can then be stated as : given H and V, can all their configurations be found as lines and respectively columns of configurations from X. Note that X could also be empty.

Definition 4 (combiner of H and V).
Given H and V two one-dimensional subshifts with alphabet Σ, the combiner *of H and V is the subshift X defined by :*

$$X = \{c \in \Sigma^{\mathbb{Z}^2} \mid \forall i, (c_{(j,i)})_{j\in\mathbb{Z}} \in H \text{ and } (c_{(i,j)})_{j\in\mathbb{Z}} \in V\}$$

Lemma 1. *Let H and V be both SFT (resp. nearest-neighbor SFT, resp. sofic-subshift, resp. effective subshift), their combiner is a subshift and can be chosen SFT (resp. nearest-neighbor SFT, resp. sofic, resp. effective)*

Proof. It is sufficient to take for the combiner the disjoint union of the horizontal forbidden patterns of H and the vertical ones of V.

Let us give some simple examples. The $H = V = {}^{\omega}(01)^{\omega}$ are fully-compatible (corresponding to the 2D checkboard). More generally, any subshift V is fully-compatible with $H = {}^{\omega}(01)^{\omega}$ if and only if V is closed under complement.

On the other hand, given a set of Wang tiles (2D nearest-neighbor SFT), it is undecidable to determine the set of its horizontal lines. In the case of [4], the set of horizontal lines is not sofic and partially understood (encodings of irrational numbers).

2.4 General direction

It can be noted that our paper deals with two orthogonal directions. Most of the results can be extended to cases where those directions are arbitrary.

Lemma 2. *Let U and V be two subshifts, u and v two non-colinear vectors. The following properties are equivalent:*

- *There is a two-dimensional subshift X such that: $U = \{c_{|\mathbb{Z}u} \mid c \in X\}$ and $V = \{c_{|\mathbb{Z}v} \mid c \in X\}$.*
- *U and V are fully-compatible.*

Proof. Let us assume the first property, We take the subshift X and construct C extracting point on the grid defined by (u, v) by:

$$C = \left\{c(ku + lv)_{(k,l)\in\mathbb{Z}^2} \mid c \in X\right\}$$

. It is clear that C is a combiner of U and V.

Conversely, let C be the combiner of U and V. As u and v are non-colinear, the regular grid can been seen as a finite disjoint union of (u, v) subgrids: $\mathbb{Z}^2 = \uplus_{0\leq i\leq N} \{ku + lv + d_i \mid k, l \in \mathbb{Z}\}$ with a fixed family of vectors $d_0, d_1, \cdots, d_N \in \mathbb{Z}^2$. Let us define the subshift obtained by taking arbitrary configuration of C on each subgrid:

$$X = \{c(x, y) = c_i(k, l) \text{ with } ku + lv + d_i = (x, y) \mid c_0, c_1, \ldots, c_N \in C\}$$

C satisfies the first property.

It can be noted that, in the previous proof X is a sofic subshift if and only if C is. However, the property is not true for SFT. In particular, the subshift X realising U and V along (u, v) could be of finite type while U and V are not

3 Decidability of the compatibility of two nearest-neighbor subshifts

In this section, we show that the compatibility problem is decidable for the case where H and V are nearest-neighbor SFT. Let H and V be one-dimensional subshifts of finite type described by the finite sets of forbidden patterns $\mathcal{F}_H$ and $\mathcal{F}_V$. As H and V are nearest-neighbor SFT we can consider that all their forbidden patterns only consists of two adjacent letters.

Lemma 3. *Let H and V be two nearest-neighbor SFT, they are compatible if and only if the 4 following propositions are verified:*

$$\begin{array}{l} 1)\ \forall a \in H\ \exists b \in H\ \forall i, (a_i, b_i) \notin \mathcal{F}_V \\ 2)\ \forall a \in H\ \exists b \in H\ \forall i, (b_i, a_i) \notin \mathcal{F}_V \\ 3)\ \forall a \in V\ \exists b \in V\ \forall i, (a_i, b_i) \notin \mathcal{F}_H \\ 4)\ \forall a \in V\ \exists b \in V\ \forall i, (b_i, a_i) \notin \mathcal{F}_H \end{array}$$

Theses formulas state that for any configuration a from H (resp. V), you can find configurations b, b' of H (resp. V) such that stacking b, a and b' on top (resp. at the right) of one another does not create a forbidden pattern of V (resp.H). Note that this lemma strongly use the fact that H and V are nearest-neighbor SFT and is not true for SFT.

Proof. Indeed, it is clear that these propositions are necessary if H and V are compatible. By using compactness, we can also show that it is sufficient : suppose that the propositions are verified, then for any configuration a of H you can construct a valid strip of any width with a in the middle. by alternately applying the first two formulas to stack enough lines to obtain a strip of any width with a in the middle. Similarly you can stack columns for a configuration of V.

We now should show how to computationally verify these four propositions.

Proposition 1. *Given two nearest-neighbor SFT U and V, it is decidable whether U and V are fully-compatible.*

Proof. H can also be seen as the set of bi-infinite walks on some digraph. Let Σ be the set of vertices with an edge between l and l' if and only if $(l, l') \notin \mathcal{F}_H$. Configurations of H then correspond to bi-infinite walks in this graph. Same goes for V. We can construct the graph (Σ, A_H) and (Σ, A_V) where: $A_H = \{(l_1, l_2) \notin \mathcal{F}_H\}$,$A_V = \{(l_1, l_2) \notin \mathcal{F}_V\}$.

Let us show how to verify the proposition 1), the other ones being similar.

Take the digraph $(\Sigma^2 \setminus \mathcal{F}_V, A)$ where $A = \{((l_1, l'_1), (l_2, l'_2)) \mid (l_1, l_2) \in A_H,\ (l'_1, l'_2) \in A_H\}$ (see example in figure 1). It is now sufficient to study the bi-infinite walks of this new graph, omitting the second element of each state. More explicitly, we are interested in $\{(l_i)_{i\in\mathbb{Z}} \mid \exists (l'_i)_{i\in\mathbb{Z}}, (l_i, l'_i)_{i\in\mathbb{Z}}$ is a bi-infinite walk $\}$. We can now directly identify this set with $\{a \in H \mid \exists b \in H\ \forall i, (a_i, b_i) \notin \mathcal{F}_V\}$.

All that is left is to verify that this set is H, we reduced the problem to testing the equality of the languages of two graphs (or automata), which is decidable.

The algorithm for the compatibility problem can be summed up as:

- Build (Σ, A_H) and (Σ, A_V)
- For each one of the four propositions :
  - Build (Σ^2, A)
  - Switch to (Σ, A) by erasing the second elements
  - Verify the equivalence of (Σ, A) and (Σ, A_H) (or (Σ, A_V) depending on the current proposition)
- H and V are compatible if all the propositions were found to be true.

4 Undecidability result

We have shown that for the class of pairs of nearest-neighbor SFTs, the problem is decidable. In this section we show that this problem is undecidable for the complementary class with respect to the class of pairs of SFT. In other words there is no algorithm to decide if two subshifts are compatible when at least one of them is allowed not to be a nearest-neighbor SFT.

In the previous section, we assumed that $\mathcal{F}_H$ and $\mathcal{F}_V$ contain only patterns consisting of two adjacent letters. But in the case of a subshift that is not a nearest-neighbor SFT, there are larger forbidden patterns, the propositions 1), 2) are no longer sufficient to build a configuration with any configuration of H as one of its lines.

We will proceed by reduction from the well-known Domino problem.

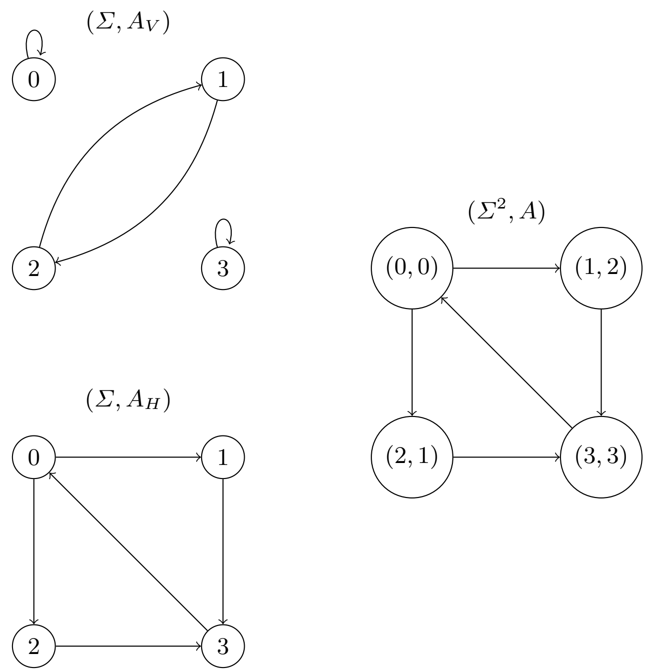

Fig. 1. Caption

Definition 5. *Domino problem Given Σ an alphabet and $\mathcal{F}$ a finite set of forbidden patterns on Σ. Is the subshift defined by $\mathcal{F}$ empty ?*

This problem is known to be undecidable [13], even when the patterns are those of a set of wang tiles, a special case of NNSFT. In the following part we will consider that the Domino problem instances are NNSFT described by forbidden patterns whose supporting sets consists of two adjacent cells.

Theorem 1. *When V is a nearest-neighbor SFT and H is SFT, there is no algorithm to decide whether they are compatible.*

Proof. Let X be a two-dimensional NNSFT on the alphabet Σ, described by its finite set of forbidden patterns $\mathcal{F}$. Write $\mathcal{F} = \mathcal{F}_h \cup \mathcal{F}_v$ by separating vertical and horizontal constraints. Without loss of generality, we can assume that there are words in $\Sigma^{\mathbb{Z}}$ avoiding $\mathcal{F}_h$, as well as words avoiding $\mathcal{F}_v$. Indeed, this can be computationally verified and X would be trivially empty otherwise.

Let $\Sigma' = \Sigma \cup \{\#\}$ with $\# \notin \Sigma$. Now let H be the SFT with alphabet Σ' and forbidden pattern set $\{(a, \#) \mid a \in \Sigma\} \cup \{(a, b, c) \mid a \in \Sigma, (b, c) \in \mathcal{F}_h\}$. Thus a word of H is either the word ${}^{\omega}\#^{\omega}$, or a word on Σ avoiding $\mathcal{F}_h$, or word of the form ${}^{\omega}\#ax$ where a is any letter from Σ and x is the half of a word on Σ avoiding $\mathcal{F}_h$. Intuitively, in H, consecutive letters from Σ must respect constraints of X except when they are preceded by the special symbol $\#$. We choose V to be the nearest-neighbor subshift on Σ' avoiding $\mathcal{F}_v$. It's the set of configurations consisting of sequences of letters from Σ avoiding $\mathcal{F}_v$, interrupted by sequences of #.

The claim is that H and V are compatible if and only if X is not empty.

Suppose that H and V are compatible. Let x be a word in $\Sigma^{\mathbb{Z}}$ avoiding $\mathcal{F}_v$, x is in V and there must be a configuration c of the combiner with its column 0 being x. As patterns of the set $\{(a, \#) \mid a \in \Sigma\}$ are forbidden in H, all the columns on the left of column 0 do not contain any occurrence of the symbol #. Then the half plane $c\,|_{\mathbb{Z}\times\mathbb{N}^*}$ avoids both $\mathcal{F}_h$ and $\mathcal{F}_v$ so by compactness we can extract a valid configuration of X from it.

Reciprocally suppose that X is not empty, and let x be one of its configurations. We must find configurations of the combiner for each of the words of H and V. We first deal with H. Let h be a word of H, define c such that $c_{(i,0)} = h_i$ and $c_{(i,j)} = \#$ otherwise, then c is a configuration of the combiner. Now let v be a configuration of V. v consists of an alternation of (possibly infinite) sequences of # and of letters from Σ avoiding $\mathcal{F}_v$. We construct the 2D configuration c as depicted in figure 2 :

- $c_{(0,j)} = v_j$ for all j
- $c_{(i,j)} = \#$ for all $i < 0$ and all j
- $c_{(i,j)} = \#$ for all if $i > 0$ all all j such that $c_{(0,j)} = \#$
- $c_{(i,j)} = x_(i,j)$ for all if $i > 0$ all all j such that $c_{(0,j)} \in \Sigma$

c is indeed a configuration of the combiner as adjacent Σ letters from column 0 and column 1 do not have to avoid $\mathcal{F}_v$ as they are always preceded by a # on column -1.

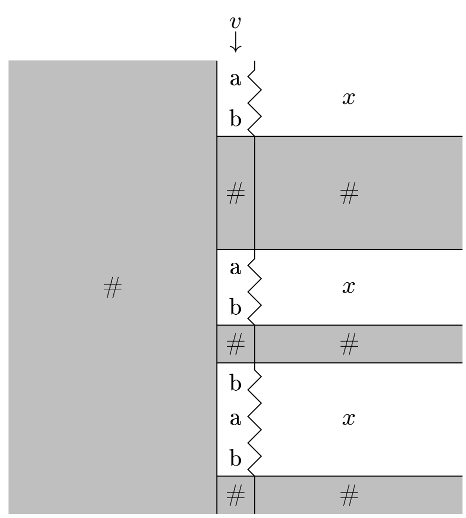

Fig. 2. Construction of a configuration with a given v as one of its columns

The undecidability extends trivially to the general case (SFT, sofic, or effective subshift).

5 Undecidability of 3D compatibility

The 3D compatibility of a 2D SFT and 1D SFT is undecidable, as the emptiness of 2D SFT already is , A better question is whether three 1D subshifts are fully-compatible — the definition extends naturally requiring each subshift being the projection along any orthogonal direction.

Proposition 2. *Given X, Y and Z three one-dimensional nearest-neighbor SFT, it is undecidable to know whether X, Y, Z are fully compatible.*

Proof. The reduction is done again starting from the domino problem. However, this reduction is one-to-many. More formally, we reduce one instance of the domino problem to several instances of 3D compatibility instances.

Given X an instance of the domino problem, the intuitive idea is to build a 3D subshift where the domino appear as a diagonal plane, this requires to thicken the plane to ensure transmission and achieving that along any orthogonal direction, we do cross at most two cells of the plane. As drawing and thinking in 3D is not so easy, we first give the sketch of the proof in 2D (which by itself does not prove anything as 1D-SFT emptiness is decidable).

Let us take a 1D nearest-neighbor SFT language L over alphabet Σ defined by the set of forbidden words $\mathcal{F}$. We want to construct two 1D-SFTs X and Z such that each non-uniform 2D configuration in the combiner corresponds exactly to one configuration of L (up to a shift). For this, we construct the alphabet $\Sigma \cup \Sigma' \cup \{\#, -\}$ where Σ' is a copy of Σ. The encoding is as follows (see figure 3): for a word $(a_i)_{i \in \mathbb{Z}}$, we places a_i at position (i, i), a copy a'_i at position $(i+1, i)$, $-$ at position (i, j) for $j < i$ and $\#$ at position (i, j) for $j > i$.

To achieve this encoding, we define the (copy) 1D-SFT X as the closure of the set $\{{}^{\omega}-\ a\ a'\ \#^{\omega} \mid \forall a \in \Sigma\}$. And the (check) 1D-SFT Z as the closure of $\{{}^{\omega}-\ a'\ b\ \#^{\omega} \mid \forall (a, b)$ admissible pattern of size two$\}$.

It is clear that any encoding of a word of $a \in L$ satisfy that every line is in X and every column is in Z. It is also clear that any configuration of the combiner is either composed only of $-$, only of $\#$, or is an encoding of a word of L.

The last point is to note that X and Z may not be fully compatible even though L is non-empty (that is some case of consecutive letters could be allowed but never occur). If they are not compatible, there is still two fully compatible subshifts $X' \subset X$ and $Z' \subset Z$ such that their combiner is exactly the combiner of X and Z . The key point is that since both X and Z are finite up to shift, there is only a finite number of possibilities for X' and Z' and if ever L is not empty, X' and Z' are distinct from ${}^{\omega}-^{\omega} \cup {}^{\omega}\#^{\omega}$

Thus, the algorithm for deciding if L is empty is the following:

- Construct X and Z

$-$	$-$	$-$	$-$	$-$	$-$	$-$	$-$	a_8
$-$	$-$	$-$	$-$	$-$	$-$	$-$	a_7	a'_7
$-$	$-$	$-$	$-$	$-$	$-$	a_6	a'_6	#
$-$	$-$	$-$	$-$	$-$	a_5	a'_5	#	#
$-$	$-$	$-$	$-$	a_4	a'_4	#	#	#
$-$	$-$	$-$	a_3	a'_3	#	#	#	#
$-$	$-$	a_2	a'_2	#	#	#	#	#
$-$	a_1	a'_1	#	#	#	#	#	#
a_0	a'_0	#	#	#	#	#	#	#

Fig. 3. Encoding of a 1D-nearest-neighbor SFT in two 1D-nearest-neighbor SFT

- For any $S_X \subsetneq \Sigma$ and any $S_Y \subsetneq \Sigma' \times \Sigma$, take X' be X with S_X as additional forbidden patterns and Z' be Z with S_Z as additional forbidden patterns, test whether X' and Z' are fully compatible.
- If there at least one pair verifying this, L is not empty.

The real 3D case is very similar to the previous one except we add one additional subshift Y. Let us take a 2D-nearest-neighbor SFT S and a configuration $a_{(i,j)} \in \Sigma^{\mathbb{Z}}$. The encoding is done by putting: $a_{(i,j)}$ at position $(i, j, i+j)$, $a'_{(i,j)}$ at position $(i+1, j, i+j)$, $a''_{(i,j)}$ at position $(i, j+1, i+j)$, and as previously, preceded by ${}^{\omega}-$ and followed by $+^{\omega}$. It can be noted that this encoding places two symbols $a'_{(i-1,j)}$ and $a''_{(i,j-1)}$ at position $(i, j, i+j-1)$.

We construct similarly over the set $\Sigma \cup (\Sigma' \times \Sigma'') \cup \{\#, -\}$ the 1D-SFT:

$$X = \{{}^{\omega}-\ \ a\ (a', b'')\ \#^{\omega} \mid a, b \in \Sigma\}$$
$$Y = \{{}^{\omega}-\ \ b\ (a', b'')\ \#^{\omega} \mid a, b \in \Sigma\}$$
$$Z = \{{}^{\omega}-\ \ (a', b'')\ c\ \#^{\omega} \mid (a, c) \text{ admissible horizontal pattern, } (b, c) \text{ vertical one } \}$$

The properties and algorithm are the same as in the 1D case but here, since tiling by 2D-nearest-neighbor SFT is undecidable, it implies that checking full compatibility of three 1D-nearest-neighbor SFT X, Y and Z is undecidable.

References

1. Aubrun, N., Esnay, S., Sablik, M.: Domino Problem Under Horizontal Constraints. In: Paul, C., Bläser, M. (eds.) STACS 2020. vol. 154, pp. 26:1–26:15.

Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl, Germany (2020). https://doi.org/10.4230/LIPIcs.STACS.2020.26
2. Aubrun, N., Sablik, M.: Simulation of effective subshifts by two-dimensional subshifts of finite type. Acta Appl. Math. **126**(1), 35–63 (2013). https://doi.org/10.1007/s10440-013-9808-5
3. Berger, R.: The undecidability of the domino problem. Ph.D. thesis, Harvard University (1964)
4. Čulík II, K., Kari, J.: An aperiodic set of wang cubes. In: Puech, C., Reischuk, R. (eds.) STACS. Lecture Notes in Computer Science, vol. 1046, pp. 137–146. Springer (1996)
5. Durand, B., Levin, L.A., Shen, A.: Complex tilings. In: STOC. pp. 732–739 (2001)
6. Esnay, S.: Limitation de la complexité de certains invariants des sous-décalages par contraintes dynamiques et structurelles. Ph.D. thesis, Université Toulouse~III, Paul Sabatier (2022)
7. Jeandel, E., Rao, M.: An aperiodic set of 11 wang tiles. Advances in Combinatorics (2021). https://doi.org/10.19086/aic.18614
8. Lind, D., Marcus, B.H.: An Introduction to Symbolic Dynamics and Coding. Cambridge University Press, 2nd edn. (2021)
9. Penrose, R.: The role of aesthetics in pure and applied mathematical research. Bulletin of the Institute of Mathematics and Its Applications **10**(2), 266–271 (1974)
10. Robinson, R.: Undecidability and nonperiodicity for tilings of the plane. Inventiones Mathematicae **12** (1971)
11. Rozenberg, G., Salomaa, A. (eds.): Handbook of Formal Languages, Volume 3: Beyond Words. Springer (1997). https://doi.org/10.1007/978-3-642-59126-6, https://doi.org/10.1007/978-3-642-59126-6
12. Smith, D., Myers, J.S., Kaplan, C.S., Goodman-Strauss, C.: An aperiodic monotile. CoRR **abs/2303.10798** (2023). https://doi.org/10.48550/ARXIV.2303.10798, https://doi.org/10.48550/arXiv.2303.10798
13. Wang, H.: Proving theorems by pattern recognition — ii. The Bell System Technical Journal **40**(1), 1–41 (1961). https://doi.org/10.1002/j.1538-7305.1961.tb03975.x