Hardness of Conjugacy, Embedding and Factorization of multidimensional Subshifts of Finite Type

Emmanuel Jeandel, Pascal Vanier

Hardness of Conjugacy and Factorization of multidimensional Subshifts of Finite Type

Emmanuel Jeandel
LIRMM, CNRS UMR 5506 - CC 477
161 rue Ada, 34 095 Montpellier Cedex 5, France
Emmanuel.Jeandel@lirmm.fr

Pascal Vanier
Laboratoire d'informatique fondamentale de Marseille (LIF)
Aix-Marseille Université, CNRS
39 rue Joliot-Curie, 13453 Marseille Cedex 13, FRANCE
Pascal.Vanier@lif.univ-mrs.fr *

Abstract

We investigate here the hardness of conjugacy and factorization of subshifts of finite type (SFTs) in dimension $d > 1$. In particular, we prove that the factorization problem is Σ_3^0-complete and the conjugacy problem Σ_1^0-complete in the arithmetical hierarchy.
*Keywords:*Subshift of finite type, factorization, conjugacy, arithmetical hierarchy, computability, tilings.

Introduction

A d-dimensional Subshift of Finite Type (SFT) is the set of colorings of $\mathbb{Z}^d$ by a finite set of colors in which a finite set of forbidden patterns never appear. One can also see them as tilings of $\mathbb{Z}^d$, and in particular in dimension 2, they are equivalent to the usual notion of tilings introduced by Wang [Wan61]. SFTs are discrete dynamical systems, and as such, their factoring and conjugacy relations are of great importance. If two dynamical systems are conjugate, then they exhibit the same dynamics. When X factors on Y, then the dynamic of Y is a subdynamic of the one of X.

Conjugacy is an equivalence relation and separates SFTs into classes. Classifying SFTs is a long standing open problem in dimension one [Boy08] which has been proved decidable in the particular case of one-sided SFTs on $\mathbb{N}$, see [Wil73]. It has been known for a long time that the problem was undecidable when

*Both authors are partially supported by ANR-09-BLAN-0164

given two SFTs, since it can be reduced to the emptyness problem which is Σ_1^0-complete [Ber64]. However, we prove here a slightly stronger result: even by fixing the class in advance, it is still undecidable to decide whether some given SFT belongs to it:

Theorem 0.1. *For any fixed X, given Y as an input, it is Σ_1^0-complete to decide if X and Y are conjugate.*

The more general problem of knowing if some SFT exhibits subdynamics of another has also been studied intensively. In dimension one, it is only partly solved for the case when the entropies of the two SFTs X, Y verify $h(X) > h(Y)$, see [Boy83]. Factor maps have also been studied with the hope of finding universal SFTs: SFTs that can factor on any other and thus contain the dynamics of all of them. However it has been shown that such SFTs do not exist, see [Hoc09, BT10]. We prove here that it is harder to know if an SFT is a factor of another than to know if it is conjugate to it.

Theorem 0.2. *Given two SFTs X, Y as inputs, it is Σ_3^0-complete to decide if X factors onto Y.*

An interesting open question for higher dimensions that would probably help solve the one dimensional problem would be *is conjugacy of subshifts with a recursive language decidable?*. A positive answer to this question would solve the one dimensional case, even if the SFTs are considered on $\mathbb{N}^2$ instead of $\mathbb{Z}^2$.

The paper is organised as follows: first we give the necessary definitions and fix the notation is section 1, after what we give the proofs of theorems 0.1 and 0.2 in sections 2 and 3 respectively.

1 Preliminary definitions

1.1 Subshifts of finite type

We give here some standard definitions and facts about multidimensional subshifts, one may consult Lind [Lin04] or Lind/Markus [LM95] for more details.

Let Σ be a finite alphabet, its elements are called *symbols*, the d-dimensional full shift on Σ is the set $\Sigma^{\mathbb{Z}^d}$ of all maps (colorings) from $\mathbb{Z}^d$ to the Σ (the colors). For $v \in \mathbb{Z}^d$, the shift functions $\sigma_v : \Sigma^{\mathbb{Z}^d} \to \Sigma^{\mathbb{Z}^d}$, are defined locally by $\sigma_v(c_x) = c_{x+v}$. The full shift equipped with the distance $d(x, y) = 2^{-\min\{\|v\| | v \in \mathbb{Z}^d, x_v \neq y_v\}}$ is a compact metric space on which the shift functions act as homeomorphisms. An element of $\Sigma^{\mathbb{Z}^d}$ is called a *configuration*.

Every closed shift-invariant (invariant by application of any σ_v) subset X of $\Sigma^{\mathbb{Z}^d}$ is called a *subshift*. An element of a subshift is called a *point* of this subshift.

Alternatively, subshifts can be defined with the help of forbidden patterns. A *pattern* is a function $p : P \to \Sigma$, where P, the *support*, is a finite subset of $\mathbb{Z}^d$. Let $\mathcal{F}$ be a collection of *forbidden* patterns, the subset X_F of $\Sigma^{\mathbb{Z}^d}$ containing

the configurations having nowhere a pattern of F. More formally, $X_{\mathcal{F}}$ is defined by

$$X_{\mathcal{F}} = \left\{ x \in \Sigma^{\mathbb{Z}^d} \middle| \forall z \in \mathbb{Z}^d, \forall p \in F, x_{|z+P} \neq p \right\}.$$

In particular, a subshift is said to be a *subshift of finite type* (SFT) when the collection of forbidden patterns is finite. Usually, the patterns used are *blocks* or *r-blocks*, that is they are defined over a finite subset P of $\mathbb{Z}^d$ of the form $B_r = [\![-r, r]\!]^d$, r is called its *radius*. We may assume that all patterns of F are defined with blocks of the same radius r, and say the SFT has radius r.

Given a subshift X, a pattern p is said to be *extensible* if there exists $x \in X$ in which p appears, p is also said to be extensible to x. We also say that a pattern p_1 is extensible to a pattern p_2 if p_1 appears in p_2. A block or pattern is said to be *admissible* if it does not contain any forbidden pattern. Note that every extensible pattern is admissible but that the converse is not necessarily true. As a matter of fact, for SFTs, it is undecidable (in Π_1^0 to be precise) in general to know whether a pattern is extensible while it is always decidable efficiently (polytime) to know if a pattern is admissible.

As we said before, SFTs are compact spaces, this gives a link between admissible and extensible: if a pattern appears in an increasing sequence of admissible patterns, then it appears in a valid configuration and is thus extensible. More generally, if we have an increasing sequence of admissible pattern, then we can extract from it a sequence converging to some point of the SFT.

Note that instead of using the formalism of SFTs we could have used the formalism of Wang tiles, in which numerous results have been proved. In particular the undecidability of knowing whether a SFT is empty. Since we will be using a construction based on Wang tiles, we review their definitions.

Wang tiles are unit squares with colored edges which may not be flipped or rotated. A *tileset* T is a finite set of Wang tiles. A *coloring of the plane* is a mapping $c : \mathbb{Z}^2 \to T$ assigning a Wang tile to each point of the plane. If all adjacent tiles of a coloring of the plane have matching edges, it is called a tiling.

The set of tilings of a Wang tileset is a SFT on the alphabet formed by the tiles. Conversely, any SFT is isomorphic to a Wang tileset. From a recursivity point of view, one can say that SFTs and Wang tilesets are equivalent. In this paper, we will be using both terminologies indiscriminately.

1.2 Factorization and Conjugacy

In the rest of the paper, we will use the notation Σ_X for the alphabet of the subshift X.

A subshift $X \subseteq \Sigma_X^{\mathbb{Z}^2}$ *factors* on a subshift $Y \subseteq \Sigma_Y^{\mathbb{Z}^2}$ if there exists a finite set $V = \{v_1, \ldots, v_k\} \subset \mathbb{Z}^2$, the *window*, and a local map $f : \Sigma_X^{|V|} \to \Sigma_Y$, such that for any point $y \in Y$, there exists a point $x \in X$ such that for all $z \in \mathbb{Z}^d, y_z = f(x_{z+v_1}, \ldots, x_{z+v_k})$. That is to say, the global map $F : X \to Y$ associated to f, the *factorization* or *factor map*, verifies $F(X) = Y$. Without

loss of generality, we may suppose that the window is an r-block, r being then called the radius of f and $(2r+1)$ its diameter.

When the map F is invertible and its inverse is also a factorization, the subshifts X and Y are said to be *conjugate*. In the rest of the paper, we will note with the same symbol the local and global functions, the context making clear which one is being used.

The entropy of a subshift X is defined as

$$h(X) = \lim_{n\to\infty} \frac{\log E_n(X)}{n^d}$$

where $E_n(X)$ is the number of extensible patterns of X of support $[\![0,n]\!]^d$ where d is the dimension. The entropy is a conjugacy invariant, that is to say, if X and Y are conjugate, then $h(X) = h(Y)$. It is in particular easy to see thanks to the entropy that the full shift on n symbols is not conjugate to the full shift with n' symbols when $n \neq n'$.

1.3 Arithmetical Hierarchy and computability

We give now some background in computability theory and in particular about the arithmetical hierarchy. More details can be found in Rogers [Rog87].

In computability, the arithmetical hierarchy is a classification of sets according to their logical characterization. A set A is Σ_n^0 if there exists a total computable predicate R such that $x \in A \Leftrightarrow \exists \overline{y_1}, \forall \overline{y_2}, \ldots, Q\overline{y_n} R(x, \overline{y_1}, \ldots, \overline{y_n})$, where Q is a $\forall$ or an $\exists$ depending on the parity of n. A set A is Π_n^0 if there exists a total computable predicate R such that $x \in A \Leftrightarrow \forall \overline{y_1}, \exists \overline{y_2}, \ldots, Q\overline{y_n} R(x, \overline{y_1}, \ldots, \overline{y_n}$, where Q is a $\forall$ or an $\exists$ depending on the parity of n. Equivalently, a set is Σ_n^0 iff its complement is Π_n^0.

We say a set A is many-one reducible to a set B, $A \leq_m B$ if there exists a computable function f such that for any x, $f(x) \in A \Leftrightarrow x \in B$. Given an enumeration of Turing Machines M_i with oracle X, the Turing *jump* X' of a set X is the set of integers i such that M_i halts on input i. We note $X^{(0)} = X$ and $X^{(n+1)} = (X^{(n)})'$. In particular $0'$ is the set of halting Turing machines.

A set A is Σ_n^0-hard (resp. Π_n^0) iff for any Σ_n^0 (resp. Π_n^0) set B, $B \leq_m A$. The problem $0^{(n)}$ is Σ_n^0-complete. Furthermore, it is Σ_n^0-complete if it is in Σ_n^0. The sets in Σ_1^0 are also called recursively enumerable and the sets in Π_1^0 are called the co-recursively enumerable or effectively closed sets.

2 Conjugacy

We prove here the Σ_1^0-completeness of the conjugacy problem in dimension $d \geq 2$, even for a fixed SFT.

Theorem 2.1. *Given two SFTs X, Y as an input, it is Σ_1^0 to decide whether X and Y are conjugate.*

Proof. To decide whether X and Y are conjugate, we have to check if there exists two local functions $F : \Sigma_X \to \Sigma_Y$, and $G : \Sigma_Y \to \Sigma_X$ such that $F \circ G = id_X$ and $G \circ F = id_Y$. To do this, we define two new functions F', G':

- Let r_X and r_F be the radiuses of X and F, then $F' : (\Sigma_X \sqcup \{\star\})^{B_{r_{F'}}} \to \Sigma_Y \sqcup \{\star\}$ is defined with a window of radius $r_{F'} = \max(r_F, r_X)$:

$$F' = \begin{cases} F & \text{if the window does not contain a forbidden pattern or a } \star \\ \star & \text{otherwise} \end{cases}$$

- Let r_Y and r_G be the radiuses of Y and G, then $G' : (\Sigma_Y \sqcup \{\star\})^{B_{r_{G'}}} \to \Sigma_X \sqcup \{\star\}$ is defined with a window of radius $r_{G'} = \max(r_F, r_X)$:

$$G' = \begin{cases} G & \text{if the window does not contain a forbidden pattern or a } \star \\ \star & \text{otherwise} \end{cases}$$

This definition may look gruesome, but is actually simple: F' and G' act exactly as F and G, except they keep track of forbidden patterns. That is, if for some pattern p, $F \circ G(p)$ does not contain a $\star$, then $G(p)$ does not contain a forbidden pattern. This will allow us to prove at the same time that $F(X) \subseteq Y$ (resp. $G(Y) \subseteq X$) and that $F \circ G = id_X$ (resp. $G \circ F = id_Y$).

We want to prove that X and Y are conjugate if and only if *there exists* F, G *and* $k > (r_{F'} + r_{G'})$ *such that for any admissible block* b *of size* k *of* X *(resp.* Y*),* $F' \circ G'(b)$ *(resp.* $G' \circ F'(b)$*) is equal to* b *at 0.*

We prove this by contraposition for $F' \circ G'$, the proof is identical for the other composition. Suppose there is a configuration $x \in X$ such that $F' \circ G'(x) \neq x$, we may suppose they differ in 0 by shifting x. So for any k there is an admissible block b of size k such that $F' \circ G'(b)$ differs from b at 0. Conversely, if there is a sequence of blocks b_k of size k, for $k > (r_{F'} + r_{G'})$ such that $F' \circ G'(b_k)$ differs from b_k in 0, then by compactness we can extract a sequence converging to some configuration $x \in X$, and by construction $F' \circ G'(x)$ differs from x in 0. □ □

Theorem 2.2. *For any* X*, given* Y *as an input, it is* Σ_1^0*-hard to decide if* X *and* Y *are conjugate.*

Proof. We reduce the problem to $0'$, the halting problem. Given a Turing machine M we construct a SFT Y_M such that Y_M is conjugate to X iff M halts.

Let R_M be Robinson's SFT [Rob71] encoding computations of M: R_M is empty iff M halts[1].

Now take the full shift on one more symbol than X, note it F. Y_M is now the disjoint union of X and $R_M \times F$.

If M halts, $Y_M = X$ and hence is conjugate to X. In the other direction, suppose M does not halt, then $R_M \times F$ has entropy strictly greater than that of X and hence Y_M is not conjugate to X. □ □

[1]Robinson's SFT is in dimension 2 of course, for higher dimensions, we take the rules that the symbol in $x \pm e_i$ equals the symbol in x, for $i > 2$.

3 Factorization

We start with two small examples to see why factorization is more complex than conjugacy. Here the examples are the simplest ones possible: we fix the SFT to which we factor in a very simple way, thus making the factor map known in advance.

Theorem 3.1. *Let Y be the SFT containing exactly one configuration, a uniform configuration. Given X as an input, it is Π_1^0-complete to know whether X factors onto Y.*

Proof. In this case the factor map is forced: it has to send everything to the only symbol of Σ_Y. And the problem is hence equivalent to knowing whether a SFT is *not* empty, which is Π_1^0-complete. □

Theorem 3.2. *Let Y be the empty SFT. Given X as an input, it is Σ_1^0-complete to know whether X factors onto Y.*

Proof. Here any factor map is suitable, the problem is equivalent to knowing whether X is empty, which is Σ_1^0-complete. □

We study now the hardness of factorization in the general case, that is to say when two SFTs are given as inputs and we want to know whether one is a factor of the other. We prove here with theorems 3.3 and 3.8 the Σ_3^0-completeness of the factorization problem.

3.1 Factorization is in Σ_3^0

Theorem 3.3. *Given two SFTs X, Y as an input, deciding whether X factors onto Y is in Σ_3^0.*

Proof. X factors onto Y iff there exists a factor map F, a local function, such that $F(X) = Y$. This is the first existential quantifier.The result follows from the two next lemmas. □ □

Lemma 3.4. *Given F, X, Y as an input, deciding if $Y \subseteq F(X)$ is Π_2^0.*

Proof. We prove here that the statement $Y \subseteq F(X)$, that is to say, *for every point $y \in Y$, there exists a point $x \in X$ such that $F(x) = y$*, is equivalent to the following Π_2^0 statement: *for any admissible pattern m of Y, if m is extensible, then $F^{-1}(m)$ contains an admissible pattern.* This statement is Π_2^0 since checking that m is *not* extensible is Σ_1^0, that is to say: there exists a radius r such that all r-blocks containing m are not admissible.

We now prove the equivalence. Suppose that $Y \subseteq F(X)$, then any extensible pattern m of Y appears in a configuration $y \in Y$ which has a preimage $x \in X$. A preimage of m being extensible, it is also admissible. This proves the first direction.

Conversely, suppose all extensible patterns m of Y have an admissible preimage. Let y be a point of Y, then we have an increasing sequence m_i of extensible

patterns converging to y. All of them have at least one admissible preimage m'_i. By compactness, we can extract from this sequence a converging subsequence, note x its limit. By construction x is a point of X and a preimage of y.

□ □

Lemma 3.5. *Given F, X, Y as an input, deciding if $F(X) \subseteq Y$ is Σ_1^0.*

Proof. It is clear that $F(X) \subseteq Y$ if and only if $F(X)$ does not contain any configuration where a forbidden patterns of Y appears. We now show that this is equivalent to the following Σ_1^0 statement: *there exists a radius r such that for any admissible r-block M of X, $F(M)$ does not contain any forbidden pattern in its center.*

We prove the result by contraposition, in both directions. Suppose there is a configuration $x \in X$ such that $F(x)$ contains a forbidden pattern. Then for any radius $r > 0$, there exists an extensible, hence admissible, pattern M of size r such that $F(M)$ contains a forbidden pattern in its center.

Conversely, if for any radius $r > 0$, there exists an admissible pattern M of X of size r such that $F(M)$ contains a forbidden pattern in its center, then by compactness, there exists a configuration $x \in X$ such that $F(x)$ contains a forbidden pattern in its center. □ □

3.2 Factorization is Σ_3^0-hard

To prove the hardness, we use the base construction that we introduced in [JV11]: we note it T. This construction introduces a new way to put Turing machine computations in SFTs, in particular, the base construction has exactly one point (up to shift) in which computations may be encoded. We call this point *configuration α*, its schematic view is shown in figure 1a. The computation is encoded in the inner grid which is sparse. Each crossing between a horizontal line and a vertical one forms a *cell*. The constraints are carried along the vertical and horizontal lines, so that we may view the encoding of the Turing machine as a tiling on the grid. For each time step, the tape of the Turing machine is encoded in the NW-SE diagonals and the size of the diagonal steadily increases in size when going north-east. At each growth of the diagonal size, it gains two cells.

Configuration α is made of two layers: one producing the horizontal lines and the other the vertical ones. The layer producing the vertical lines is shown in figure 2, the vertical lines are the black vertical lines. The configuration producing the horizontal lines is its exact symetric along the south-west/north-east diagonal. The key property of these layers is that when a corner tile (the tile in the lower left corner of the first square) appears, then the point is necessarily of this form.

In the original construction, corner tiles of the horizontal and vertical layers could only be superimposed to each other. We just change this so that instead, the corner tile of the vertical layer has to be at position $(1, -1)$ relative to the corner tile of the horizontal one. This change does not impact any of the properties of T, but simplifies a bit the proof of lemma 3.7.

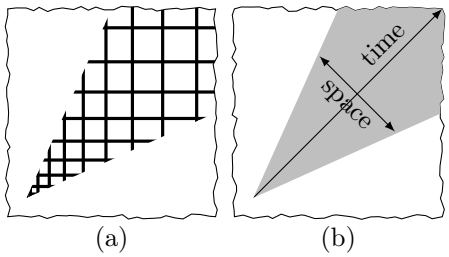

Figure 1: (a) The skeleton of configuration α. (b) How the computation is superimposed to α.

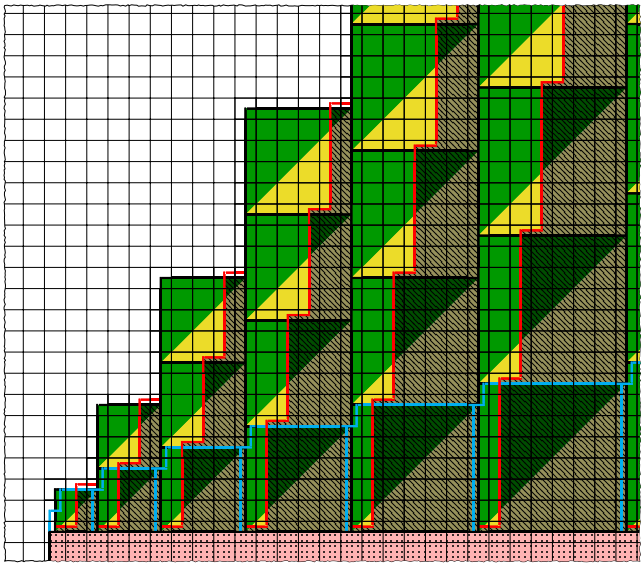

Figure 2: Point α, the meaningful point of X_T. The corner tile may be seen on the first non all-white column: it is the lower left corner of the square.

Our reduction will use two SFTs based on this construction, both of them will be feature a different tiling on its grid. We will say that an SFT which is basically T with a tiling on its grid as having T-structure.

Definition 3.6 (T-structure)**.** *We say an SFT X has* T-structure *it is a copy of T to which we superimposed new symbols only on the symbols representing the horizontal/vertical lines and their crossings.*

Note that an SFT may have T-structure while having no α-configuration: for instance if you put a computation of a Turing machine that produces an error whenever it halts.

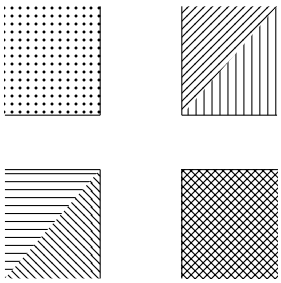

Figure 3: Uniform quarter- and eighth-planes in non-α-configurations.

The next lemma states a very intuitive result, that will be used later, namely that if an SFT with T-structure factors to another one, then the structure of each point is preserved by factorization. Furthermore, it shows that the factor map can only send a cell to its corresponding one, that is to say cell of the preimage has to be in the window of the image.

Lemma 3.7. *Let X, Y be two SFTs with T-structure, such that X factors onto Y. Let r be the radius of the factorization, then any α-configuration of Y is factored on by an α-configuration of X, and can be shifted by v iff $\|v\|_\infty \leq r$.*

Proof. By [JV11, lemma 1], we know that non-α-configurations have at most one vertical line and one horizontal line. And therefore that they have two uniform (same symbols everywhere) quarter-planes and four uniform eighth-planes, as seen on figure 3. The two north east eighth-planes are not uniform in α. thus they cannot factor on α.

It remains to prove the second part: that in the factorization process the α-structure is at most shifted by the radius of the factorization. We do that by contradiction, suppose that an α-configurations x of X factors on an α-configuration y of Y and shifts it by $v = (v_x, v_y)$, with $\|v\|_\infty > r$. Without loss of generality we may suppose that $v_x > r$ and $v_y > 0$ and that the vertical and horizontal corner tiles of the preimage are at positions $(0, 1)$ and $(1, 0)$ respectively. We are now going to show that this is not possible.

On the horizontal layer, for all $k \in \mathbb{N}^*$ there is a square with lower left corner at $(2k^2+k, 2k^2+k)$, see figure 2. Inside this square, there are two $(k-1)\times(k-1)$ uniform smaller squares, see figure 4. This being also true for the vertical layer, these squares remain uniform when they are superimposed. Now take k such that $k > (\|v\|_\infty + 2r + 1)$. By hypothesis, there is a vertical line symbol t at $z_p = (2k^2+2k+1, 2k^2+k)$ on x, and thus at $z_i = (2k^2+2k+1+v_x, 2k^2+k+v_y)$ on y. We know $x_{|z_i+B_r}$ has image t, and by what precedes that $x_{|z_i+B_r} = x_{|z_i+(1,0)+B_r}$ since they are both uniform, therefore, there should be two t symbols next to eachother in y at z_i and $z_i + (1, 0)$. This is impossible.

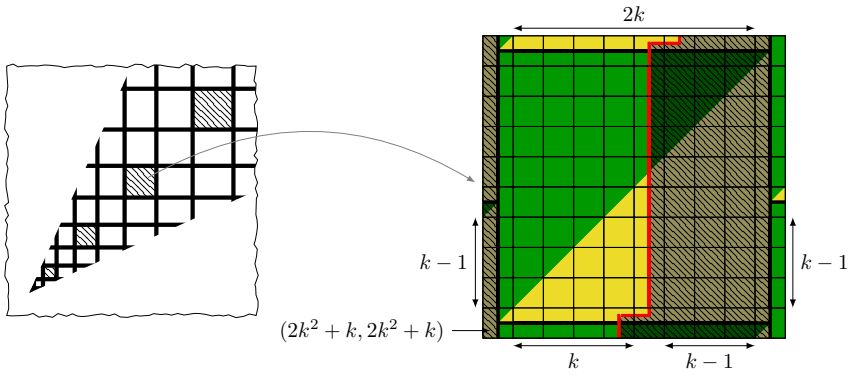

Figure 4: For every $k \in \mathbb{N}^*$, the square starting at position $(2k^2+k, 2k^2+k)$ is of the form on the right on the component producing the vertical lines (and is the symetric along the diagonal for the one producing the horizontal lines). We can see that there are two uniform $(k-1)\times(k-1)$ squares at $(2k^2+2k+2, 2k^2+k+1)$ and $(2k^2+k+1, 2k^2+2k+2)$ respectively.

□ □

Theorem 3.8. *Given two SFTs X, Y as an input, deciding whether X factors onto Y is Σ_3^0-hard.*

For this proof, we will reduce to the problem **COFINITE**, which is known to be Σ_3^0-complete, see Kozen [Koz06]. **COFINITE** is the set of Turing machines which run infinitely only on a finite set of inputs.

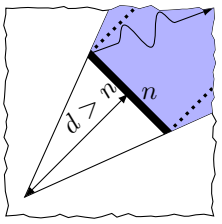

Figure 5: Computation on input n in the SFT Z, the number of white diagonals d preceeding the computation is strigtly greater than the input n.

Proof. Given a Turing machine M, we construct two SFTs X_M and Y_M such that X_M factors on Y_M iff the set of inputs on which M does not halt is finite. We first introduce an SFT Z_M on which both will be based. It will of course have T structure. Above the T base, we allow the cells of the grid to be either white or blue according to the following rules:

- All cells on a NW-SE diagonal are of the same color.

- A blue diagonal may follow (along direction SW-NE) a white diagonal, but not the contrary.
- A transition from white diagonal to blue may only appear when the grid grows.

We now allow computation on blue cells only. Only the diagonals after the growth of the grid may contain computations. The Turing machine M is launched on the input formed by the size of the first blue line (in number of cells of course). We forbid the machine to halt

So for each n on which M does not halt, there is a configuration with white cells until the first blue diagonal appears, then computation occurs inside the blue cone, see figure 5 for a schematic view. If M halts on n, then there is no tiling where the first blue line codes n. By compactness, there is of course a configuration with only white diagonals. If M is total, then the only α-configuration in Z_M is the one with only white diagonals.

Now from Z_M, we can give X_M and Y_M:

- X_M: Let Z'_M be a copy of Z_M to which we add two decorations 0 and 1 on the blue cells only, and all blue cells in a configuration must have the same decoration. Now X_M is Z'_M to which we add a third color, red, that may only appear alone, instead of white and blue. No computation is superimposed on red.
- Y_M is a copy of Z_M where we decorated only the horizontal corner tile with two symbols 0 and 1.

We now check that X_M factors onto Y_M iff M does not halt on a finite set of inputs:

$\Rightarrow$ Suppose M does not halt on a finite set of inputs: there exists N such that M halts on every input greater than N. The following factor map F works:

  - F is the identity on Z_M. Note that the additional copy of T is also sent onto Z_M.
  - F has a radius big enough so that when its window is centered on the corner tile, it would cover the beginning of the computation on input N.
  - An α-configuration x of X_M is sent on the same α-configuration y in Y_M. For the decorations, when there is a computation on x, the factor map can see it and gives the same decoration to the corner tile of y. When there is no computation, the factor map doesn't see a computation zone and gives decoration 0 to the corner tile. The configuration with only white diagonals and decoration 1 of Y_M is factored on by the α-configuration colored in red contained in X_M.

  Note that this also works when M is total.

$\Leftarrow$ Conversely, suppose M does not halt on an infinite set of inputs, and that there exists a factor map F with radius r: lemma 3.7 states that all α-configurations of Y_M are factored on by α-configurations of X_M. Now, there is an infinite number of α-configurations with corner tile decorated with 0 (resp. 1) in Y_M, they all must be factored on by some α-configuration of X_M. Still by lemma 3.7, the corner tile of the preimage must be in the window of the corner tile of the image. However, there can only be a finite number of configurations in which the symbols in this window differ. So the α-configurations of X_M factor to a finite number of α-configurations of Y_M with one of the decorations. This is impossible.

Note that the construction of X_M and Y_M from the description of M is computable and uniform. The reduction is thus many-one. □ □

References

[Ber64] Robert Berger. *The Undecidability of the Domino Problem*. PhD thesis, Harvard University, 1964.

[Boy83] Mike Boyle. Lower entropy factors of sofic systems. *Ergodic Theory and Dynamical Systems*, 3:541–551, 1983.

[Boy08] Mike Boyle. Open Problems in Symbolic Dynamics. *Contemporary Mathematics*, 469:69–118, 2008.

[BT10] Laurent Boyer and Guillaume Theyssier. On factor universality in symbolic spaces. In *MFCS*, pages 209–220, 2010.

[Hoc09] Michael Hochman. A note on universality in multidimensional symbolic dynamics. *Discrete and Continuous Dynamical Systems S*, 2(2), 2009.

[JV11] Emmanuel Jeandel and Pascal Vanier. Π_1^0 sets and tilings. In *Theory and Applications of Models of Computation (TAMC)*, volume 6648 of *Lecture Notes in Computer Science*, pages 230–239, 2011.

[Koz06] Dexter Kozen. *Theory of Computation*. Springer, New York, 2006.

[Lin04] Douglas A. Lind. Multi-Dimensional Symbolic Dynamics. In Susan G. Williams, editor, *Symbolic Dynamics and its Applications*, number 60 in Proceedings of Symposia in Applied Mathematics, pages 61–79. American Mathematical Society, 2004.

[LM95] Douglas Lind and Brian Marcus. *An introduction to symbolic dynamics and coding*. Cambridge University Press, New York, NY, USA, 1995.

[Rob71] Raphael M. Robinson. Undecidability and Nonperiodicity for Tilings of the Plane. *Inventiones Math.*, 12, 1971.

[Rog87] Hartley Rogers, Jr. *Theory of recursive functions and effective computability*. MIT Press, Cambridge, MA, USA, 1987.

[Wan61] Hao Wang. Proving theorems by Pattern Recognition II. *Bell Systems technical journal*, 40:1–41, 1961.

[Wil73] R. F. Williams. Classification of subshifts of finite type. *Annals of Mathematics*, 98:120–153, 1973.