Wang Tiles

May 2, 2006

Abstract

Suppose we want to cover the plane with decorated square tiles of the same size. Tiles are to be chosen from a finite number of types. There are unbounded tiles of each type available. Due to the decorations, however, there are local constraints on which tiles can be put next to each other, for the tiling to look appealing. Is it possible to cover the whole plane with tiles of given types? How if we require a certain tile to be used at least once? Can they be used to tile a finite rectangular area, with a certain boundary condition? It turns out that these problems—the way formulated by Hao Wang (1961)—are all undecidable.

1 Introduction

A Wang tile is a unit square with coloured edges. For a given tile t, the colour of the upper, lower, left, and right edges are denoted by $N(t)$, $S(t)$, $W(t)$, and $E(t)$ respectively.

Let T be a finite set of tiles, with edges coloured with colours from a finite set C. A configuration of the plane is simply an assignment $c : \mathbb{Z}^2 \to T$ of tiles to the points on the discrete plane. A partially defined configuration is referred to as a pattern. Finite patterns are often considered modulo translations. The set of finite patterns modulo translations is denoted by T^*.

A (valid) tiling of the plane is a configuration $t : \mathbb{Z}^2 \to T$, where adjacent tiles agree on the colour of their touching edges; i.e., for any position $(i, j) \in \mathbb{Z}^2$,

$$N(t(i,j)) = S(t(i,j+1)), \qquad S(t(i,j)) = N(t(i,j-1)),$$
$$W(t(i,j)) = E(t(i-1,j)), \qquad E(t(i,j)) = W(t(i+1,j)).$$

A valid pattern is defined accordingly. A finite valid pattern may also be called a patch. Note that, it might not be possible to extend a valid pattern to a valid tiling of the whole plane.

The Tiling problem, introduced by Hao Wang [9], asks if a tile set can be used to tile the whole plane:

Problem 1 (Plane Tiling)**.** Given a finite set T of Wang tiles, determine whether it admits a valid tiling of the plane.

The above is our main problem of interest. We are going to show that it is undecidable. The following variants will be also discussed. Quite simple though they are, their proof of undecidability is the basic ingredient of the main result.

Problem 2 (Plane Tiling with Restricted Origin)**.** Given a finite set T of Wang tiles and a distinct tile $\mathsf{t}_0 \in \mathsf{T}$, determine whether it admits a valid tiling of the plane which uses t_0 at least once.

Problem 3 (Finite Plane Tiling)**.** Given a finite set T of Wang tiles and a distinct colour \$, determine whether they can be used to validly cover a rectangular area whose boundary is all coloured by \$.

It is easy to see that the one-dimensional version of the above problems are all decidable. A 1d tile has only two edges, whose colours are denoted by L and R. In this case the tiling space is neatly represented by a finite directed graph: There is a vertex for each tile, and an edge from t to s, iff $\mathsf{R}(\mathsf{t}) = \mathsf{L}(\mathsf{s})$. The 1d tilings, then, correspond to the paths on this graph. Note that such a graph has bi-infinite path (i.e., a tiling of the line $\mathbb{Z}$), if and only if, it has a cycle.

The above also shows that, in 1d, any tile set that admits a tiling, also admits a tiling which is periodic. This is not true in 2d, as we shall see later. A weaker statement can however be extended to 2d. Let us stress that we call a tiling (or a configuration) of the plane periodic if it has two linearly independent periods. Using a similar argument as above one can easily see that,

Proposition 1. *A tile set can tile the plane periodically, if it admits a tiling that has a non-zero period.*

As we shall see, the undecidability of the non-restricted tiling problem in 2d (i.e., Problem 1) relies on the fact that there exist tile sets that can cover the whole plane validly, but not in any periodic way. Such a tile set is called aperiodic. In fact, when restricted to non-aperiodic tile sets, the non-restricted tiling problem becomes decidable: One tries to tile larger and larger rectangular areas, until she finds a block which can be used periodically, or a rectangle which cannot be tiled.

Proposition 2. *There is an algorithm, that given a finite set of Wang tiles, halts on any instance which is not aperiodic, and outputs* `YES` *or* `NO`*, according to whether they can tile the plane validly.*

In the above argument, we implicitly used the fact that if the plane cannot be tiled by a set of tiles, there is already a finite rectangle which cannot be tiled by those tiles. Let us make this statement clearer.

We say a set $\mathsf{L} \subseteq \mathsf{T}^*$ of patterns is factorial if it contains all sub-patterns of its elements. The following lemma will come handy in our discussion and follows from the König's Lemma.

Lemma 1 (Extension Lemma)**.** *Let* $\mathsf{L} \subseteq \mathsf{T}^*$ *be a factorial set of finite patterns which contains arbitrarily large patterns. Then there is a (total) configuration* $\mathsf{c} \in \mathsf{T}^{\mathbb{Z}^2}$ *whose sub-patterns are all from* L.

In topological terms, the above lemma simply says that the space of configurations under Cantor topology is compact. Cantor topology is the one in which convergence is defined as pointwise eventual agreement.

Corollary 1. *A finite tile set can tile the whole plane, if and only if, it can tile arbitrarily large finite areas of the plane.*

Note that the above immediately tells that the main tiling problem is co-recursively enumerable. That is, we can recognize, algorithmically, if a tile set does not admit a valid tiling.

2 Simulating Turing Machines

In this section we see how to encode the space-time computation table of a Turing machine into a tiling of the plane. More clearly, given a Turing machine, we construct a tile set, so that if we translate the initial configuration of the machine into a row of tiles on the plane, the upper rows are forced to be translations of the consecutive configurations of the machine. As a consequence we can immediately reduce the Halting Problem to either of Problems 2 or 3 and establish their undecidability.

The encoding is most natural and intuitive. Let $M = (\Sigma, Q, \delta, q_0, q_F, \#)$ be a Turing machine with a one-way-infinite tape, where Σ is the alphabet, Q the set of states, $\delta : Q - \{q_F\} \times \Sigma \to \Sigma \times \{L, R\} \times Q$ the transition function, $q_0 \in Q$ the initial state, $q_F \in Q$ the final state, and $\# \in \Sigma$ the blank symbol. We construct a set T_M of Wang tiles as shown in Figure 1 and 2 and described bellow.

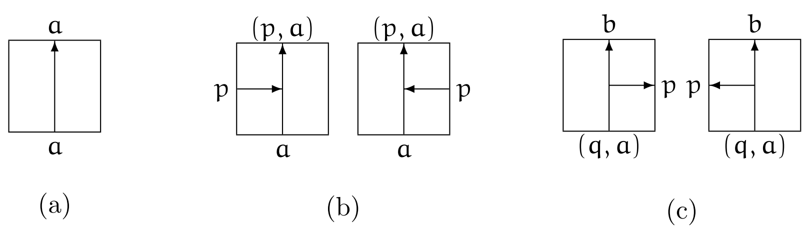


Figure 1: Tiles for encoding a Turing machine. (a) Alphabet tile. (b) Merging tiles. (c) Action tiles.

- For any letter $a \in \Sigma$, we have a tile of the form depicted in Figure 1(a). These are to pass the content of an inactive cell of the tape one row upward (which corresponds to one step later in the computation process of the machine).
- For any state $q \in Q$ and letter $a \in \Sigma$, where $\delta(q, a) = (b, D, p)$, there is a tile like one of those in Figure 1(c), depending on whether $D = R$

or $D = L$. These correspond the action of the transition function, and passing the new state to a neighbor cell.

- The state p received from a neighbor cell, is combined with the current content of the cell, by a tile of the form shown in Figure 1(b) and passed to the upper tile to be processed in next step.

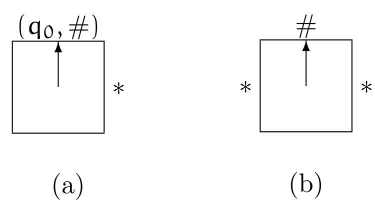


Figure 2: Tiles for fixing the initial configuration of a Turing machine. (a) Head position. (b) Empty cells.

- The initial configuration of the machine is fixed by the tiles in Figure 2. Here $*$ is a new colour and used to ensure that only one head appears on the tape!

Note that the arrows on tiles of Figures 1 and 2 are just for clarity, and the colours used are all from $\Sigma \cup Q \cup (Q \times \Sigma) \cup \{*, \texttt{WHITE}\}$. It should be easy to see how the above construction works. As a result we have the following lemma. Let t_0 be tile in Figure 2(a).

Lemma 2. *The infinite upper-right quarter of the plane can be tiled with the tile set T_M and the tile t_0 in the lower-left corner, if and only if, the Turing machine M, on blank input, never halts.*

We are now ready to prove the undecidability of Problem 2.

Theorem 1. *Plane Tiling with Restricted Origin is undecidable.*

Proof. Let M be an instance of the Halting Problem. Construct the tile set T_M as described above. Let t_w be a tile having all edges coloured with `WHITE`. The machine M never halts, if and only if, the tile set $T_M \cup \{t_w\}$ can tile the plane, provided that we require the tile t_0 to be used at least once.[1] □

To prove the undecidability of Problem 3 we need to add some more tiles and modify the tile t_0, but we skip the details, because it is straightforward.

Theorem 2. *Finite Plane Tiling is undecidable.*

[1]We should have further required that the initial state q_0 never appear in later steps, but that is easily achieved by properly modifying the machine M.

We only comment that the undecidability of Problems 2 and 3 are different in the sense that in the latter, the halting instances are mapped to the tile sets that *admit* a tiling of interest. In fact, Finite Plane Tiling is recursively enumerable, while Plane Tiling with Restricted Origin is co-recursively enumerable.

3 Non-restricted Tiling Problem

We now turn our attention to proving the undecidability of the main problem; i.e., the Plane Tiling Problem. We would like to use the construction in the previous section to reduce the Halting Problem to our problem. However, there are certain difficulties in this respect.

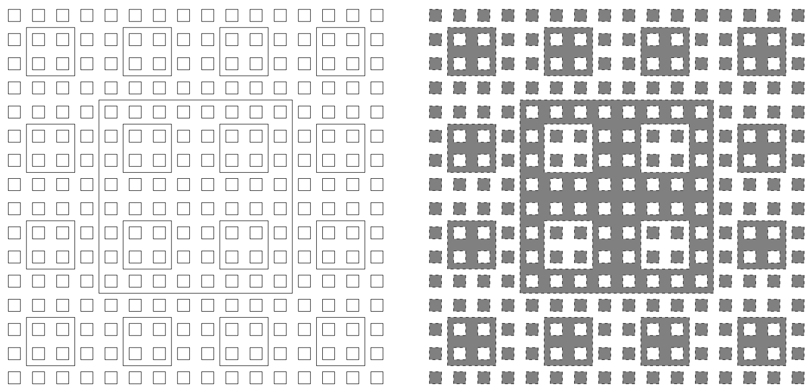
The following interesting result (basically of a topological nature) may shed some light on the difficulties we face. A tiling (or more generally, a configuration) $c : \mathbb{Z}^2 \to T$ is called quasi-periodic, if for any sub-pattern u of c, there is an integer $p_u > 0$, such that u appears on any $p_u \times p_u$ block of c.

Proposition 3. *Any tile set that admits a valid tiling, also admits a quasi-periodic tiling.*

Proof. Let T be a finite set of Wang tiles that admits valid tilings. Our main tool is the Extension Lemma. Let us take a set $L \subseteq T^*$ of patches (i.e., valid patterns) which is factorial and contains arbitrarily large patterns, and is *minimal*, in the sense that it has no proper subset with those properties. The Extension Lemma guarantees that there exist a tiling $t : \mathbb{Z}^2 \to T$ whose sub-patterns are all from L. We claim that t is quasi-periodic. For, let it be not. That implies that a sub-pattern u of t exists, such that larger and larger patterns with no occurrence of u can be found in t. Let L' be a collection of such patterns together with all their factors. Clearly $L' \subsetneq L$, and yet L' is factorial and contains arbitrarily large patterns. This contradicts the minimality of L. Therefore, t must be quasi-periodic. □

Back to our troubles, first of all, how can we make sure that, in any valid tiling, the tile t_0 is used at least once? A basic idea would be to use tiles that have two layers. The simulation is done on the upper layer. The lower layer is there to generate a certain fixed decoration on the background. If the decoration has suitable reference points (occurrences of a certain tile, e.g.), we can use them to trigger the start of computation in the upper layer. Unfortunately, the existence of quasi-periodic tilings means that, no matter how it is designed, the reference tile may inevitably occur every so often. How can we now tackle the interaction between computations triggered in different places?

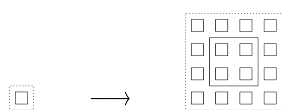
The intriguing idea in Robinson's proof [7] (also in Berger's [2]?) is the use of a *self-similar* decoration in the background. A fragment of such a decoration is shown in Figure 3(a). The whole (infinite) decoration is the fixed point of the



(a) (b)

Figure 3: A self-similar decoration of the plane. (a) The decoration. (b) Regions cloured alternatively.

following substitution:



Fortunately, such a decoration can be forced by a finite set of Wang tiles. Actually, Robinson constructed a set of 56 tiles that does so, and we shall see how to do that in the next section.

For clarity, in Figure 3(b), the alternate regions are coloured with two different colours. The idea is to simulate the computation of the Turing machine on each connected region independently. That is, once on each small square, once on each of the 2×2 regions, once on each of the 4×4 regions, and so forth. There are two crucial points in this approach:

- Obviously, the whole computation table might not be mapped to a finite region. However, if larger and larger initial segments of the table are mapped on the plane, we can argue that the computation never ends!
- Observe that the connected regions we are using have so many holes that are distributed densely over the region. That is also crucial. Note that if we could map larger and larger bulks of the computation table on a tiling, we would also have larger and larger bulks that have no beginning, and the Extension Lemma would then imply that a meaningless computation without a beginning could also be mapped on a tiling.

Assuming the existence of a tile set that generates the decoration in Figure 3, we can prove the undecidability of the Plane Tiling Problem.

Theorem 3 ([2]). *Plane Tiling is undecidable.*

Proof. Let M be a Turing Machine—an instance of the Halting Problem. We use the tile set $\mathsf{T_M}$, constructed in Section 2, and a tile set $\mathsf{T_R}$, that forces the self-similar decoration shown in Figure 3, to construct a new set of tiles T that can tile the plane, if and only if, M on empty input never halts.

Each tile in T is basically a pair (s, t), where $\mathsf{s} \in \mathsf{T_R}$, and t is either in $\mathsf{T_M}$ or is of an auxiliary type, to pass information along a vertical or horizontal line. The two components are interpreted as layers of the tile. Let $\mathsf{s_0} \in \mathsf{T_R}$ be any tile that represents the lower-left corner of a connected region. For any such tile, we place a tile $(\mathsf{s_0}, \mathsf{t_0})$ in T, where $\mathsf{t_0}$ is the marked tile of $\mathsf{T_M}$ as defined in Section 2. This is the only pair in which $\mathsf{t_0}$ appears, and is supposed to trigger the start of simulation in each connected region. Any other tile $\mathsf{t} \in \mathsf{T_M} - \{\mathsf{t_0}\}$ is paired with tiles $\mathsf{s} \in \mathsf{T_R}$ that represent a cell fully inside a connected region.

Let us say a row (or column) of a region is free if it does not pass through a hole. Otherwise, the row (or column) is blocked by the hole. The intersection of a free row and free column is a free cell. Only free cells are used for simulation. A cell that belongs to a blocked row (resp. a blocked column) simply passes the information along the edge of the blocking hole. More clearly, a cell whose lower (or upper) edge touches a hole simply passes the colour of its left edge to the right, and asks its upper (resp. lower) neighbor to do the same. Similarly, a cell whose left (resp. right) edge touches a hole passes the colour of its lower edge to the upper edge and asks its right (resp. left) neighbor to do the same.

Now, it should be clear that the above-described tiles can tile a connected region C of the decoration, if and only if, $\mathsf{T_M}$ can tile a square with the same area as C, provided $\mathsf{t_0}$ is used in the lower-left corner. Since the entire decoration provides connected regions of arbitrarily large net area, using the Extension Lemma, we conclude that the plane can be tiled by the tiles in T, if and only if, $\mathsf{T_M}$ can tile an upper-right quarter of the plane with $\mathsf{t_0}$ in the lower-left corner. This completes the proof. □

4 Forcing a Self-Similar Decoration

In this section we see Robinson's construction of a tile set that forces the decoration in Figure 3. Robinson's tiles, in fact, force a slightly different decoration shown in Figure 4. Incidentally, this is a 2d variant of the celebrated *paper-folding* sequence. One should readily see how to change these tiles to obtain the former decoration.

Two tile sets $\mathsf{T_1}$ and $\mathsf{T_2}$ shown in Figure 5 are used in the construction. The rotations and reflections of the tiles in Figure 5(b) are also included in $\mathsf{T_2}$. The arrows are understood to be followed by an arrow heading the same direction. This can be easily implemented in terms of colouring of the edges.

The Robinson's tiles are made of two interacting layers. The tiles in the lower layer are chosen from $\mathsf{T_1}$, and those in the upper layer from $\mathsf{T_2}$. Let x

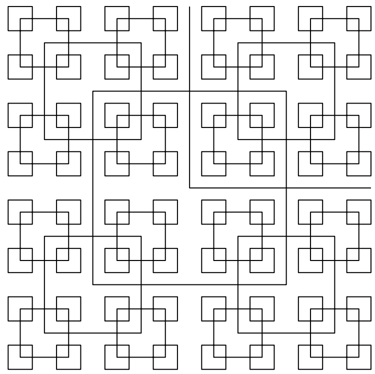
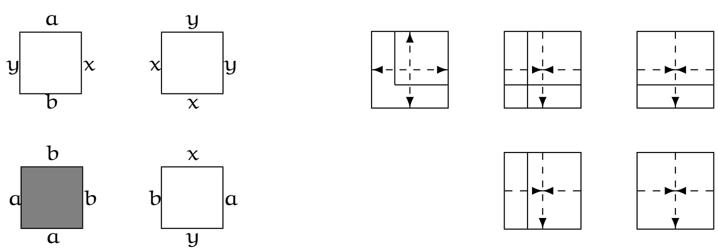


Figure 4: The decoration generated by Robinson's tiles.



(a) (b)

Figure 5: The basic tiles used in Robinson's construction. (a) The set T_1. (b) The set T_2 (rotations and reflections omitted).

be the dark tile in Figure 5(a), and y_1 the one singled out on the left side of Figure 5(b), and y_2, y_3 and y_4 its rotations. The set T of tiles is defined as bellow:

$$T \triangleq \{x\} \times \{y_1, y_2, y_3, y_4\} \cup (T_1 - \{x\}) \times T_2$$

The lower layer is only there to fix the place of the smallest squares on the plane (squares of level 0). Note that the pattern of these squares is periodic. Clearly, any periodic decoration can be forced by a tile set.

To understand how the construction of T_2 works, note that each edge of a $2^k \times 2^k$ square ($k > 1$) crosses edges of two $2^{k-1} \times 2^{k-1}$ squares and one or zero $2^{k+1} \times 2^{k+1}$ square. The dashed arrows, together with the displacement of the filled lines, are used to keep track of the edges already crossed, and to synchronize the position of the squares.

Figure 6 shows a part of a valid tiling using the tiles in T. To keep the figure look less messy, only the upper layer (i.e., the T_2 component) is shown.

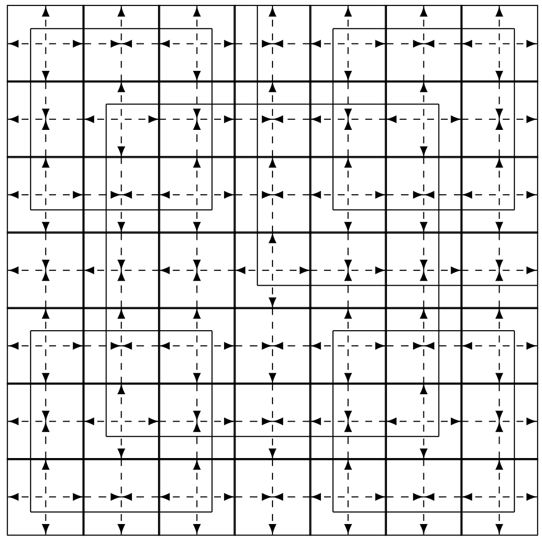


Figure 6: A part of the plane tiled by Robinson's tiles.

Note that the dark tiles (i.e., those with x in the lower layer) occur in every other row, and every other column. Each dark tile has a corner (i.e., either of tiles y_1 or its rotations) in the upper layer. (Corners may also occur on light tiles.) The dark corners may either be face-to-face or back-to-back. The face-to-face dark corners form 3×3 blocks, representing the 2×2 squares of the decoration (squares of level 1). It can be verified that such 3×3 blocks may only be arranged in a synchronized fashion, generating the periodic pattern of the occurrence of squares of level 1. The centre of each such block is, again, a corner.

Let us now consider the set of 3×3 valid blocks, having dark tiles on the corners, as *super-tiles*, where two super-tiles may sit next two each other if the corresponding blocks have an overlapping row or column. One may verify that the set T' of such super-tiles is in fact *isomorphic* to T (in the natural sense). Therefore, following an inductive reasoning, we may conclude that the tile set T generates the required decoration.

Proposition 4. *The Robinson's tile set generates the self-similar decoration in Figure 4.*

Corollary 2. *The Robinson's tile set is aperiodic.*

5 Non-recursive Tilings

In this section, we will investigate tile sets with such exotic property that they provably admit valid tilings, yet none of such tilings can be constructed algorithmically!

Theorem 4 ([4, 6])**.** *There is a finite set of Wang tiles that can tile the plane, but not in any recursive way.*

Given a finite set of tiles, one may follow a trial and error strategy to tile the plane: She puts the tiles one by one on the plane. Whenever there are several choices, she simply follows the first one. Once she got stuck, she goes back and follows the next choice, and so forth. This process gives a valid tiling of the plane in the limit, if such a tiling exists. The theorem, however, says that for the above-mentioned tile set there is no recursive bound on the length of the branches she might need to track back! She might never become confident about the tile that should be used in a certain position.

We shall not prove this result. Rather, we comment on its basic ideas. The main ingredient of the construction comes from the following lemma.

Lemma 3. *There exist a disjoint recursively inseparable pair of recursively enumerable sets.*

Proof. Let $\{\lambda_i\}_i$ be an effective enumeration of all one-variable recursive functions. Define

$$P \triangleq \{x \in \mathbb{N} \mid \lambda_x(x) = 0\} \ ,$$

$$Q \triangleq \{x \in \mathbb{N} \mid \lambda_x(x) = 1\} \ .$$

Clearly P and Q are recursively enumerable and disjoint. They are also recursively inseparable. For, suppose that C is a recursive set such that $P \subseteq C$ and $Q \subseteq \mathbb{N} - C$. Since C is recursive, there exists an index i such that

$$\lambda_i(x) = \begin{cases} 1 & \text{if } x \in C, \\ 0 & \text{otherwise.} \end{cases}$$

Examining the value of $\lambda_i(i)$ reveals the contradiction. □

As a corollary we can make Turing machines that on any infinite input, but possibly on some non-recursive ones, halt.

Corollary 3. *There is a Turing machine with a read-only input tape and a work tape that has the following properties:*

i) On any infinite, but recursive input the machine halts.

ii) There exist some non-recursive infinite inputs on which the machine does not halt.

Proof. Let P and Q be two disjoint recursively inseparable recursively enumerable sets, and g and h be some effective enumerations of them. The machine M on an infinite binary input works as follows: First it verifies if the $g(0)$'th bit of the input is 1 and halts if it is not. Then it verifies if the $h(0)$'th bit of the input is 0 and halts if it is not. Then again sees if the $g(1)$'th bit is 1; then if the $h(1)$'th bit is 0, and so forth. If the input is recursive, machine will inevitably stop at some point, because P and Q are inseparable. On the other hand, since P and Q are disjoint, there is a (non-recursive) set C that includes P and does not intersect Q. Feeding the characteristic sequence of C as input, the machine never halts. □

It is easy to modify the construction in Section 2 so that the machine has two tapes on top of each other. It can also be made so that every cell of each row is aware of the state of the machine, and the position of the two heads are distinguished separately. The modified tiles would be designed so that arbitrary binary sequence can be fed to the machine on its input tape.

Given the Turing machine M constructed in Corollary 3, we may construct the tile set T'_M as above. Note, however, that this, per se, does not satisfy the property we are seeking. Namely, the input sequence fed to different simulation blocks, as discussed in Section 3, can be different. Myers [6] has developed a way to ensure that the first row of all simulation blocks are the same.

6 Conclusion

Problem 4 (Tiling of Cayley Graphs)**.** The Wang tilings can be investigated in a more general setting; i.e., on Cayley graphs, where there is a natural meaning of translation and adjacency. One may wonder, on which Cayley graphs the Tiling Problem is undecidable, and on which it becomes decidable.

Problem 5 (Bi-infinite PCP)**.** A problem, from a different context, but having similar kind of difficulties as in the Plane Tiling, is the Bi-infinite Post Correspondence Problem: Given two morphisms $g, h : \Sigma^+ \to \Gamma^+$, determine if there is a bi-infinite word $w \in {}^{\omega}\Sigma^{\omega}$ so that $g(w) = h(w)$. Here, the equality of two bi-infinite sequences should be considered modulo translations. Can we tackle this problem using the ideas used in Robinson's proof?

Bibliography

[1] C. Allauzen and B. Durand. Tiling problems. In E. Börger, E. Grädel, and Y. Gurevich, editors, *The Classical Decision Problem*. Springer, 1997.

[2] R. Berger. The undecidability of the domino problem. *Memoirs of the American Mathematical Society*, 66, 1966.

[3] B. Grünbaum and G. C. Shephard. *Tilings and Patterns*. Freeman, 1987.

[4] W. Hanf. Nonrecursive tilings of the plane I. *The Journal of Symbolic Logic*, 39(2), 1974.

[5] J. Kari. Recent results on aperiodic Wang tilings. In A. Carbone, M. Gromov, and P. Prusinkiewicz, editors, *Pattern formation in biology, vision and dynamics*. World Scientific, 2000.

[6] D. Myers. Nonrecursive tilings of the plane II. *The Journal of Symbolic Logic*, 39(2), 1974.

[7] R. M. Robinson. Undecidability and nonperiodicity for tilings of the plane. *Inventiones Mathematicae*, 12:177–209, 1971.

[8] H. Rogers. *Theory of Recursive Functions and Effective Computability*. McGraw-Hill, 1967.

[9] H. Wang. Proving theorems by pattern recognition—II. *Bell Systems Technical Journal*, 40:1–41, 1961.