THE PERIODIC DOMINO PROBLEM REVISITED

EMMANUEL JEANDEL

NOTICE: This is the author's version of a work accepted for publication by Elsevier. Changes resulting from the publishing process, including peer review, editing, corrections, structural formatting and other quality control mechanisms, may not be reflected in this document. Changes may have been made to this work since it was submitted for publication. The definitive version has been published in Theoretical Computer Science, 411:4010-4016, 2010. doi:10.1016/j.tcs.2010.08.017

Abstract. In this article we give a new proof of the undecidability of the periodic domino problem. The main difference with the previous proofs is that this one does not start from a proof of the undecidability of the (general) domino problem but only from the existence of an aperiodic tileset.

The formalism of Wang tiles was introduced in [Wan61] to study decision procedures for the ∀∃∀ fragment of the first-order logic. The earliest and most fundamental question is the domino problem: decide, given a finite set of Wang tiles if it tiles the plane. It turns out that this is not possible, this so-called *domino problem* was proven undecidable [Ber64]. There are until now to the author's knowledge 6 different proofs of the undecidability of the domino problem. Five of them encode the halting problem [Ber64, Rob71, AL74, Oll08, DRS12] while the last one [Kar07] encode the immortality problem for Turing machines [Hoo66].

This problem is intimately linked with the existence of aperiodic tilesets. An aperiodic tileset is a tileset that can tile the plane, but cannot tile it periodically. Wang conjectured that no such tileset exists. Were the conjecture true, the domino problem would be decidable [Wan61]. As a consequence, every proof of the undecidability of the domino problem gives as a byproduct the existence of an aperiodic tileset. In fact, almost any known proof first builds an aperiodic tileset then explains how to code computation in its tilings. This is indeed the case in [Ber64, Rob71, AL74, Oll08, DRS12]. This is not the case in [Kar07]. However, we can still build an aperiodic tileset from the proof: The immortality problem being undecidable [Hoo66], there must exist by compactness a Turing machine with no periodic points. [BCN02] gives such a machine. Encoding this machine with the construction in [Kar07] will give an aperiodic tileset.

A proof of the undecidability of the domino problem gives a new aperiodic tileset. Is the converse true ? Can we use any aperiodic tileset as the first step in a proof of the undecidability of the domino problem ? In the constructions of [Ber64, Rob71, AL74, Oll08], each tileset is indeed handmade so that encoding of computation (by Turing machines) is easily done. However, can we do the same with any aperiodic tileset, not a specific one ? We do not know an answer to this question. As a specific example, we do not know how to encode a computation in the Ammann tileset [AGS92, GS87] or in the Kari-Culik tilesets [Kar96, II96].

A related problem is the *periodic domino problem*, where one asks whether a tileset can produce a periodic tiling. As aperiodic tilesets exist, this problem is not trivial. As a matter of fact, it is also undecidable. Interestingly, all known proofs are obtained by looking carefully to a proof of the undecidability of the domino problem and making some adjustements: [AL74, DRS12] already contain the two results, while [GK72] corresponds to [Ber64] and [AD01] to [Rob71]. One could also obtain a somewhat intricate proof tweaking [Kar07] using the methods of [KO08].

Similarly to the domino problem, any proof of the undecidability of the periodic domino problem gives a new aperiodic tileset. In this article, we will prove the converse: we will give a new proof of the undecidability starting from any aperiodic tileset. That is, an aperiodic tileset is all we need for this proof. The construction we use here is new. The main idea, which is quite simple, is exposed in section 2.1. The rest of the construction is textbook tilings.

This article is mostly self-contained. A superficial knowledge of Turing machines and finite automata is required.

1. Wang tilings

Wang tiles are square tiles with colored edges. A representation of a Wang tile can be found in Fig. 1.

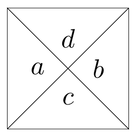

Figure 1. A Wang tile

Formally a Wang tile is a map t from the set $\{N, S, E, W\}$ to a finite set Q. In the figure, we have $t(N) = d, t(E) = b$, etc. A *tileset* τ is a finite set of Wang tiles.

A *tiling* c of the plane by τ associates to each point of the discrete plane $\mathbb{Z}^2$ a tile of τ so that contiguous edges have the same color. If we denote by $c_{i,j}$ the tile at position (i, j), the condition becomes:

$$c_{i,j}(N) = c_{i,j+1}(S)$$
$$c_{i,j}(E) = c_{i+1,j}(W)$$

The *domino problem* is the following:

Problem 1 (Domino problem)**.** *Decide, given a tileset τ, whether there exists a tiling by τ.*

This problem was proven undecidable in [Ber64].

A tiling c by τ is *periodic* if there exist p so that for all i, j :

$$c_{i,j} = c_{i+p,j}$$
$$c_{i,j} = c_{i,j+p}$$

In this article we are interested in the following problem:

Problem 2 (Periodic Domino problem)**.** *Decide, given a tileset τ, whether there exists a periodic tiling by τ.*

A tileset is *aperiodic* if there exists tilings by τ, but no periodic tilings. Aperiodic tilesets exist [Ber64, Rob71, AGS92, Kar96, II96], so that in fact the periodic domino problem is a different problem from the domino problem. In fact, the main difficulty for the resolution of these problems is the existence of aperiodic tilesets. Did aperiodic tilesets not exist, would the two problems be decidable [Wan75].

To simplify the construction, we will mostly deal with horizontally periodic tilings. A tiling c is *horizontally periodic* if there exists p so that for all i, j,

$$c_{i,j} = c_{i+p,j}$$

The following lemma is folklore:

Lemma 3. *Let τ be a tileset. There exists a horizontally periodic tiling by τ if and only if there exists a periodic tiling by τ.*

The proof of the nontrivial implication is as follows. Let c be a horizontally periodic tiling by τ, of period p. Let $d_j : i \mapsto c_{i,j}$ denote the j-th line. Each d_j is periodic of period p. As there are at most $|\tau|^p$ different lines of period p, two d_j must be equal, say d_0 and d_q. As a consequence, the map c' defined by $c'_{i,j} = c_{i,(j \mod q)}$ is a tiling by τ and is periodic of period $\mathrm{lcm}(p, q)$.

The purpose of this article is to give an easy proof of the following theorem.

Theorem 4. *The periodic domino problem is undecidable.*

Our proof is as follows: Starting from any aperiodic tileset τ (for example the Ammann tileset [AGS92]) and a Turing machine M, we will build a tileset τ_M so that τ_M admits a periodic tiling if and only if M halts on the empty input. What is important to note is that our construction does not work at all if τ is not aperiodic; our construction does not build any new aperiodic tileset, but needs to start from one.

2. The construction

In this section, we give our new proof of the theorem. We first briefly discuss the key steps of the proof.

We start from any aperiodic tileset τ. The first part of the proof (section 2.1 and 2.2) builds a tileset τ_2 starting from τ. This tileset τ_2 wil be the (disjoint) union of three different tilesets of respectively *white*, *black* and *gray* tiles. The goal is to have *every* periodic tiling by τ_2 to look like a grid (see Fig. 2) delimited by black tiles vertically and gray tiles horizontally. The second step is then to encode a Turing computation inside each square of the grid, starting e.g. from each lower left corner.

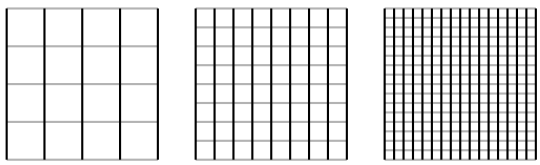

FIGURE 2. The grid shape for periodic tilings we would like to obtain

The second step is straightforward. The first step presents a slight difficulty. τ_2 produces arbitrary large all-white squares, and thus by compactness produces an all-white tiling. Were this tiling periodic, this would give us a periodic tiling without a grid. For the proof to work, the region inside a grid has to follow an aperiodic behaviour. This is where we use our aperiodic tileset τ: the grids in periodic tilings by τ_2 will actually be grids filled with squares of tilings by τ.

2.1. The black and white coloring.

We start from the aperiodic tileset τ and consider all tiles from this tileset to be *white*. We then introduce *black* tiles, depicted in Fig.3 which are tiles where N and S are black, and E and W are of any color that can appear horizontally (i.e as E or W) in a white tile. Our new tileset is then $\tau_1 = \tau \cup \tau_N$ where τ_N represents the new black tiles.

Now consider a (horizontally) periodic tiling by τ_1. As τ is an aperiodic tileset, this tiling must use tiles of τ_N. A tile of τ (resp. τ_N) must be surrounded vertically by tiles of τ (resp. τ_N). As a consequence, any horizontally periodic tiling by τ_1 of

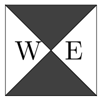

FIGURE 3. Black tiles

period p consists of vertical columns of either white or black tiles, and must contain *at least* one black column, as depicted in Fig. 4. Conversely, for every $n \geq 1$ there

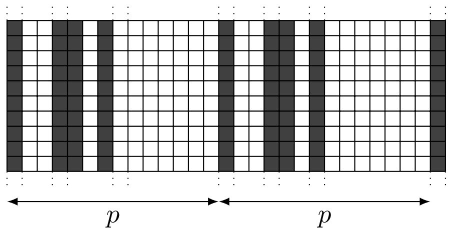

FIGURE 4. The generic shape of a periodic tiling by τ_1

exists a tiling by τ_1 of horizontal period n containing $n-1$ white columns and one black columns: Just take $n-1$ columns from any tiling by τ and add correctly a black column. Note that this tiling is typically not periodic, only horizontally periodic.

2.2. **The horizontal marker.** We now change the tileset τ_1 into a tileset τ_2 so that:

- Every periodic tiling by τ_2 consists of "squares"
- For every $n \geq 3$ there exists a periodic tiling by τ_2 where all squares are of size (at least) n.

We will see in the description of the tilings what we mean exactly by a square.

Consider the white space between two black lines. Represent the white as void and the black as walls. Suppose there is a "particle" in the void that goes from left to right and that teleports to the left every time it crosses the right wall, as depicted in Fig. 5.

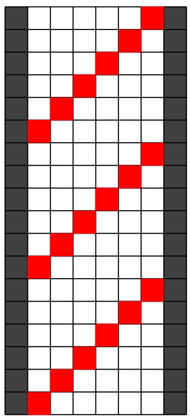

FIGURE 5. A particle that goes from left to right

Here is how we implement such a thing using Wang tiles. The particle will be represented by a 1 and the void by 0s. Hence each line must be in $0^\star 10^\star$.

Now, consider the transducer in Fig. 6 that takes every line to the following line[1].

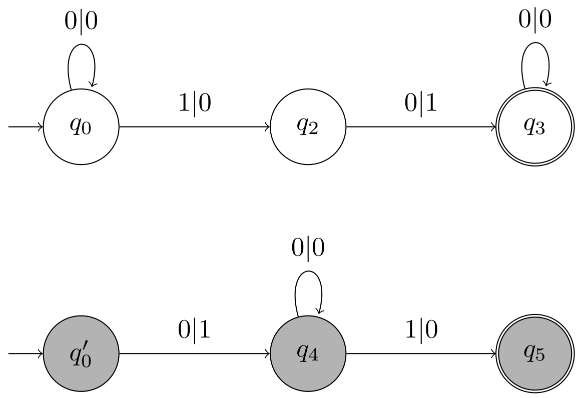

FIGURE 6. A transducer that maps 0^n10^m to $0^{n+1}10^{m-1}$ and 0^n1 to 10^n. The states q_0 and q'_0 are initial states, and the significance of the transitions is as follows: if the transducer is in state q_2 and reads a 0, then it emits a 1 and goes to state q_3. The first three states represent the most usual transitions, and the last three states represent the particular case of $0\cdots01 \mapsto 10\cdots0$.

We represent the transducer with Wang tiles: for each transition from q to q' reading a and outputting b, we create the tile where N, S, W, E is respectively b, a, q, q', see Fig. 7.

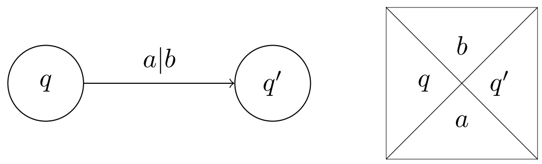

FIGURE 7. How to obtain a tile from a transition of the transducer

We obtain in this way the tileset τ_A of Fig. 8. To represent initial and final states, we add for each initial state q and final state q' a tile where E, W is respectively q, q', to obtain the tileset τ_B in Fig. 9.

We now add to our initial tileset τ_1 a new layer in the following way:

- Superimpose to each white tile of τ_1 one of the tiles τ_A. If the tile from τ_A is one of the three last ones, we will say that the tile is *gray*.
- Superimpose to each black tile of τ_1 one of the tiles of τ_B

Hence $\tau_2 = (\tau \times \tau_A) \cup (\tau_N \times \tau_B)$.

Now a typical tiling between two black lines will look on the second layer as squares delimited by black and gray lines as depicted in Fig. 10. As a consequence,

- Every tiling between two black lines consists of squares, delimited by the gray tiles.

[1]Note that the transducer works only if the line if of the form $0^\star 10^\star$

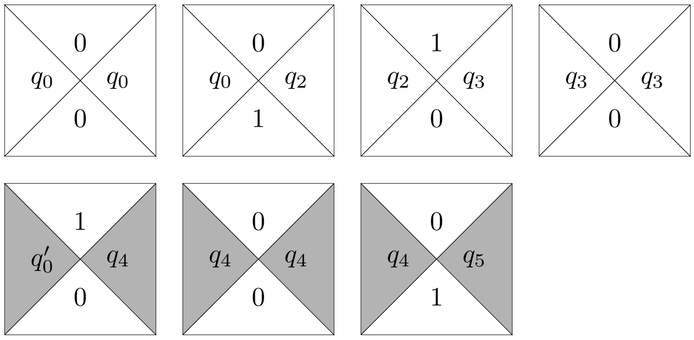

FIGURE 8. The tileset τ_A corresponding to the transducer in Fig. 6

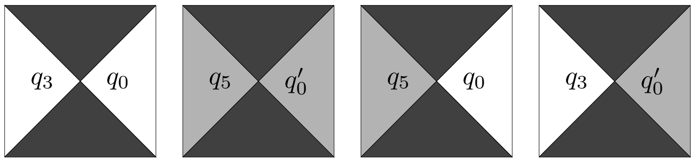

FIGURE 9. The tileset τ_B corresponding to the initial and final states of the transducer in Fig. 6

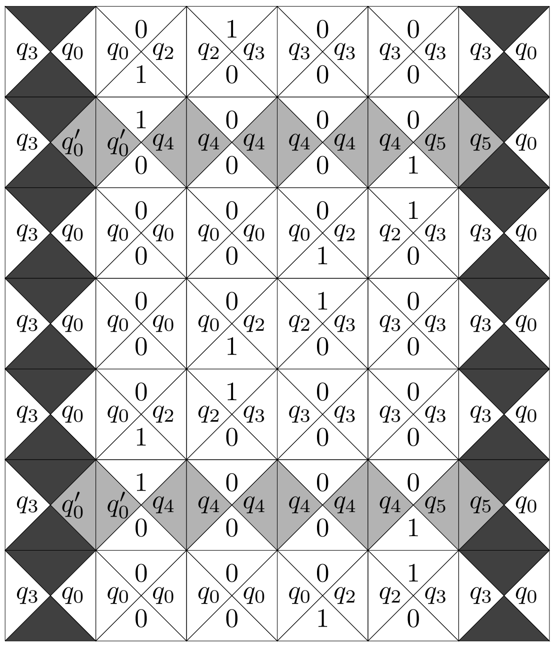

FIGURE 10. A typical periodic tiling by τ_2 (only the second layer is depicted)

- For every $n \geq 3$, there exists a periodic tiling where the distance between two black lines is exactly n, hence a tiling consisting entirely of $(n-1) \times (n-1)$ squares.

Note that periodic tilings of τ_2 are not exactly grids. Indeed, we can obtain a tiling where each column between two black lines contain squares, but the squares might be of different size or of different origin as in Fig. 11. To obtain grids, superimpose each black tile of τ_1 only with the first two tile of τ_B indicated above rather than the four tiles. This part is not strictly necessary to obtain our result.

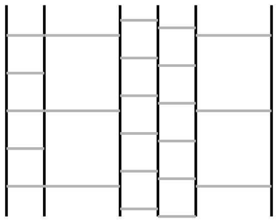

FIGURE 11. A typical periodic tiling by τ_2 (only black and gray tiles are depicted)

2.3. **Encoding computation.** Now that the tileset τ_2 is defined, it will be easy to prove the undecidability of the periodic domino problem.

Let M be a Turing machine over an alphabet Σ with a set of states Q. Let $Q_0 = Q \cup \{0\}$.

We will see a configuration of M of size n as a word w over $\Sigma \times Q_0$ that is $w_i = (u_i, q_i)$ where u_i is the symbol in position i of the tape, q_i denotes the state of the Turing machine if the head is in position i, $q_i = 0$ otherwise.

As in the previous section, we will use transducers to represent the evolution of the Turing Machine. The formal description of the transducer is given in Fig. 12.

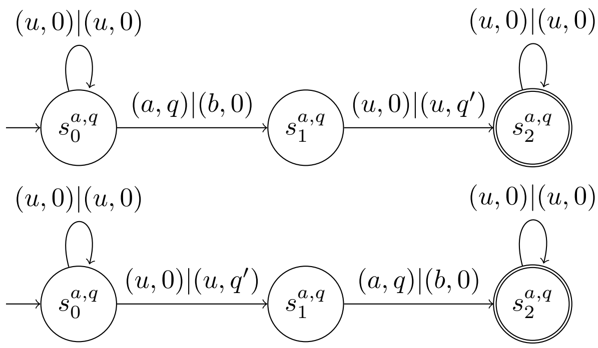

FIGURE 12. The skeleton of a transducer corresponding to a Turing Machine. For each pair (a, q) so that reading a in the state q makes the Turing Machine go *right*, writing b and going to state q', we add the first three states. For each pair (a, q) so that reading a in the state q makes the Turing Machine go *left*, writing b and going to state q', we add the last three states. u denotes any symbol. The transition $(u, 0)|(u, q')$ means for example that the first element of the pair is unchanged, and the second one becomes q'.

It should be clear now that the transducer we obtain this way takes any configuration of a Turing Machine to its successor configuration (except of course if the head of the Turing Machine is to the far left/right of the word and the Machine has to go left/right).

We now use the exact same technique as in the previous section to build our tileset τ_M on the black and white tiles. It remains to show how we code the

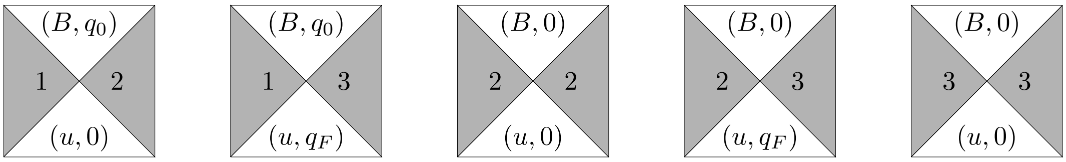

FIGURE 13. Encoding of the initial and final configurations of the Turing Machine. u denotes any symbol, B is the blank symbol, q_0 the initial state, q_F the final state of the Turing Machine

initial/final configuration. This will be done with the gray tiles, using the $5|\Sigma|$ tiles given in Fig. 13, that we superimpose with the gray tiles of τ_A. The following property is easy to verify: any tiling of a (finite) row starting from a 1 and finishing with a 3 has a word of $(B, q_0)(B, 0)^\star$ as his north side, and a word containing q_F in his south side.

To finish the construction of τ_M, superimpose with the gray sides of τ_B (the "corner" tiles) a 1 on the left, and a 3 on the right.

Now we prove that the construction works:

- Consider a (horizontally) periodic tiling by τ_M. This tiling consists of squares, bordered by black columns and gray lines. Now we examine any square. This square codes an execution of the Turing Machine M. The north side of the lower gray line contains a word in $(B, q_0)(B, 0)^\star$ hence, the execution of the Turing Machine starts from the initial state for an empty input. The south side of the upper gray line contains a word containing the final state. Hence the execution of the Turing Machine reaches the final state. As a consequence, if there is a (horizontally) periodic tiling by τ_M, then M halts on the empty input.
- Conversely, suppose that M halts on the empty input in $n \geq 2$ steps. Then we can easily build a tiling by τ_M which is horizontally periodic of period $n + 1$.

As a consequence, τ_M admits a (horizontally) periodic tiling (hence a periodic tiling) if and only if M halts on the empty input. This ends the proof.

3. SOME REMARKS

We conclude this article with some remarks about the proof:

- Sections 2.2, 2.3 is actually a result about (letter-to-letter, deterministic) transducers : there is no algorithm to decide given a (letter-to-letter) transducer f whether there exists a word x so that $f^i(x)$ is defined for all i or equivalenty (by compactness) whether there exists a sequence of words $(x_i)_{i\in\mathbb{Z}}$ of the same length so that $f(x_i) = x_{i+1}$ for all $i \in \mathbb{Z}$.
- Note that the tileset τ_M we obtain at the end always tiles the plane: the second and third step both contain one tile that tiles the plane periodically (for example, take the sixth tile of τ_A, and the last tile of Fig.13). As a consequence, if the tiling does not contain any black tile in the first layer, there is no way to control what happens in the second and third step, and a trivial tiling can appear. The trick is that periodicity forces black tiles to appear, hence the second and third layer to behave correctly.
- By a similar argument, note that if our original tileset τ tiles the plane periodically, there will always be a periodic tiling, and the construction fails dramatically. Hence the need for aperiodic tilesets.

- Our result can be formulated as follows. The two following statements are equivalent: (i) there exists an aperiodic tileset (ii) the periodic domino problem is undecidable. $(i) \to (ii)$ follows from the proof, $(ii) \to (i)$ follows from compactness (see e.g. [Wan75]). It is important to note that what we have done here is not a proof of (ii) but rather a proof of $(i) \to (ii)$, as illustrated by the two previous remarks. We may ask how this result generalizes. Suppose we are trying to tile other objects (the hyperbolic plane, a finitely generated group...), is the undecidability of the periodic domino problem equivalent to the existence of an aperiodic tileset ?

  The situation is quite different for the domino problem. Let (ii'): the domino problem is undecidable. Then again, $(ii') \to (i)$ follows directly from compactness. We do not know however of any direct proof of $(i) \to (ii')$: all known proofs of (ii') start indeed by constructing an *ad hoc* aperiodic tileset. Formally speaking, this means we do not know a proof of $(i) \to (ii')$ without first proving (i). In particular, we do not know how to encode computation in specific tilesets like e.g. the Ammann tileset [AGS92].
- We can use our proof together with the undecidability[2] of the domino problem [Ber64] to prove the following result [GK72]:

  Theorem 5. *There exists an algorithm that given a Turing Machine M produces a tileset τ_M so that*
  - *If M does not halt on the empty input, τ_M is an aperiodic tileset*
  - *If M halts on state q_0, τ_M admits a periodic tiling*
  - *If M halts on state q_1, there is no tiling by τ_M*

  The main idea is as follows. By co-r.e.-completeness of the domino problem, there is an algorithm that given a Turing Machine N produces a tileset τ'_N so that
  - If N does not halt on the empty input, there is a tiling by τ'_N
  - If N halts, there is no tiling by τ'_N

  We can change τ'_N so that if N does not halt, τ'_N is an aperiodic tileset: to do this, add an aperiodic layer to τ'_N. Now use this tileset τ'_N as the basis tileset τ of the construction of the previous section. This is only the main idea and we leave all details to the reader.
- Our result is slightly weaker than the previous ones in the following sense. All other proofs build a tileset τ_M so that if M halts on the empty input, τ_M produces only *one* tiling (up to translation), which is periodic. As stated above, in our case τ_M always produce nonperiodic tilings, hence a weaker result. For all common uses (e.g. the conservative reduction for $\forall\exists\forall$ [BGG01], who relies on the previous theorem) our result is sufficient.
- Our result is stronger in the following sense. The least (horizontal) period of a periodic tiling by τ_M is *exactly* the number of steps n of M before it halts. For other proofs, the period is typically $O(n^2)$ or $O(2^n)$. This tighter bound is not coincidental. In fact, a refinement of our techniques can be used to give an answer to the following problem [JV10]: characterize which sets of positive numbers are periods of tilesets.

[2] More accurately, we need the fact that the domino problem is co-r.e-complete

References

[AD01] Cyril Allauzen and Bruno Durand. Tiling Problems. In *The Classical Decision Problem* [BGG01], chapter A, pages 407–420.

[AGS92] Robert Ammann, Branko Grunbaum, and G.C. Shephard. Aperiodic tiles. *Discrete and Computational Geometry*, 8(1):1–25, December 1992.

[AL74] Stål O. Aanderaa and Harry R. Lewis. Linear Sampling and the ∀∃∀ Case of the Decision Problem. *Journal of Symbolic Logic*, 39(3):519–548, September 1974. See also [Lew79].

[BCN02] Vincent D. Blondel, Julien Cassaigne, and Codrin Nichitiu. On the presence of periodic configurations in Turing machines and in counter machines. *Theoretical Computer Science*, 289(1):573–590, 2002.

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

[BGG01] Egon Börger, Erich Grädel, and Yuri Gurevich. *The Classical Decision Problem*. Perspectives in Mathematical Logic. Springer, 2001.

[DRS12] Bruno Durand, Andrei Romashchenko, and Alexander Shen. Fixed-point tile sets and their applications. *Journal of Computer and System Sciences*, 78(3):731–764, May 2012.

[GK72] Yuri Gurevich and I Koryakov. Remarks on Berger's paper on the domino problem. *Siberian Math. Journal*, 1972.

[GS87] Branko Grunbaum and G.C. Shephard. *Tilings and Patterns*. W.H. Freeman, 1987.

[Hoo66] Philip K. Hooper. The Undecidability of the Turing Machine Immortality Problem. *Journal of Symbolic Logic*, 31(2):219–234, June 1966.

[II96] Karel Culik II. An aperiodic set of 13 Wang tiles. *Discrete Mathematics*, 160:245–251, 1996.

[JV10] Emmanuel Jeandel and Pascal Vanier. Periodicity in Tilings. In *Developments in Language Theory (DLT)*, volume 6224 of *Lecture Notes in Computer Science*, pages 243–254. Springer, 2010.

[Kar96] Jarkko Kari. A small aperiodic set of Wang tiles. *Discrete Mathematics*, 160:259–264, 1996.

[Kar07] Jarkko Kari. The Tiling Problem Revisited . In *Machines, Computations, and Universality (MCU)*, number 4664 in Lecture Notes in Computer Science, pages 72–79, 2007.

[KO08] Jarkko Kari and Nicolas Ollinger. Periodicity and Immortality in Reversible Computing. In *MFCS 2008*, LNCS 5162, pages 419–430, 2008.

[Lew79] Harry R. Lewis. *Unsolvable Classes of Quantificational Formulas*. Addison-Wesley, 1979.

[Oll08] Nicolas Ollinger. Two-by-Two Substitution Systems and the Undecidability of the Domino Problem. In *CiE 2008*, number 5028 in Lecture Notes in Computer Science, pages 476–485, 2008.

[Rob71] Raphael M. Robinson. Undecidability and Nonperiodicity for Tilings of the Plane. *Inventiones Mathematicae*, 12(3), 1971.

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

[Wan75] Hao Wang. Notes on a class of tiling problems. *Fundamenta Mathematicae*, 82:295–305, 1975.