

# Tiling the Plane with a Fixed Number of Polyominoes

Nicolas Ollinger

Laboratoire d'informatique fondamentale de Marseille (LIF),  
Aix-Marseille Université, CNRS,  
39 rue Joliot-Curie, 13 013 Marseille, France  
`Nicolas.Ollinger@lif.univ-mrs.fr`

**Abstract.** Deciding whether a finite set of polyominoes tiles the plane is undecidable by reduction from the Domino problem. In this paper, we prove that the problem remains undecidable if the set of instances is restricted to sets of 5 polyominoes. In the case of tiling by translations only, we prove that the problem is undecidable for sets of 11 polyominoes.

## Introduction

Tiling the plane given a finite set of tiles is an old and fascinating problem. For an survey on tilings, the reader is invited to consult Grünbaum and Shephard [1]. A celebrated computability result by Berger [2] is the undecidability of the Domino problem: given a finite set of Wang tiles, unit squares with colored edges, decide if the Wang tiles can tile the whole plane so that matching edges share a same color. A polyomino is a simple kind of tile: it consists of rookwise connected unit squares. Golomb [3] studied tiling by polyominoes and proved in [4] that the Domino problem can be reduced to deciding if a finite set of polyominoes tiles the plane. The reduction can be achieved by a classical encoding of Wang tiles by polyominoes that preserves tilings. In this reduction, the number of polyominoes is equal to the initial number of Wang tiles. A natural question arises: what happens if we consider the tiling problem for a fixed number of polyominoes? From this previous result, two cases might happen: (1) the problem is undecidable starting from a certain fixed number of polyominoes (2) the problem is decidable for every fixed number of polyominoes but the family of decision procedures is not itself recursive. As case (1) is more likely to happen, the question is to find the frontier between decidability and undecidability. Such a study of decidability questions with respect to a parameter appears for example in the study of semi-Thue systems or for Post correspondence problem (PCP) where it is shown that  $PCP(2)$  is decidable and  $PCP(7)$  is undecidable [5,6,7].

Motivated by parallel computing, Wijshoff and van Leeuwen [8] proved that the tilability of the plane by translation of a unique polyomino is decidable. That result was further studied and understood by Beauquier and Nivat [9] who described precisely the tilings by translation generated by a unique polyomino.

This paper is organized as follows. In section 1, we introduce Wang tiles, polyominoes and dented polyominoes, a special variation of polyominoes used in

the constructions. In section 2, we prove that it is undecidable whether a set of 5 polyominoes tiles the plane. In section 3, we deduce from previous section that it is undecidable whether a set of 11 polyominoes tiles the plane by translation. In section 4, we discuss the case of smaller sets of tiles.

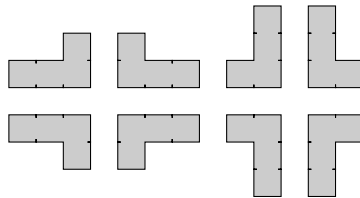
### 1 Definitions

**Polyominoes.** A *polyomino* is a simply connected tile obtained by gluing together rookwise connected unit squares. A *tiling*, of the Euclidian plane, by a set of polyominoes is a partition of the plane such that each element of the partition is the image by an isometry of a polyomino of the set. A *tiling by translation* is a tiling where isometries are restricted to translations. A tiling is *periodic* if it is invariant by translation, *biperiodic* if it is invariant by two non-colinear translations, *aperiodic* if it is not periodic. A set of polyominoes is *aperiodic* if it admits a tiling and all its tilings are aperiodic.

A tiling is *discrete* if all the vertices of the unit squares composing the polyominoes are aligned on the grid  $\mathbb{Z}^2$ . If a tiling is not discrete, the tiling can be split into two tilings of a half-plane along a line going through an edge along which two unit squares are not aligned. By shifting one half-plane to align the tiles, and iterating this process, one obtains the following lemma.

**Lemma 1.** *A set of polyominoes admits a tiling if and only if it admits a discrete tiling.*

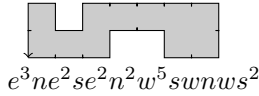
In this paper where we deal with tilability, we only consider discrete tilings, thanks to this lemma. Thus, a polyomino can be considered as a finite, simply connected, subset of  $\mathbb{Z}^2$  and a tiling by a set of polyominoes is a partition of  $\mathbb{Z}^2$  where each element is the image by an isometry of an element of the set. Each such isometry can be decomposed into a translation and one out of 8 elementary transformations obtained by composing right angle rotations and mirroring. A sample polyomino and its 8 transformations are represented in Fig. 1.



**Fig. 1.** A polyomino and its 8 transformations

Polyominoes are compactly described by their contour words. A *contour word* of a polyomino is a (finite) word on the alphabet  $\{e, w, n, s\}$  describing a walk along the outline of the polyomino starting from and ending to a vertex of the boundary of the polyomino where  $e$  is an east move  $(1, 0)$ ,  $w$  is a west move

$(-1, 0)$ ,  $n$  is a north move  $(0, 1)$  and  $s$  is a south move  $(0, -1)$ . A word is a contour word if and only if the associated path does not cross itself. A polyomino with a pointed contour word is represented in Fig. 2



**Fig. 2.** A polyomino with a pointed contour word

**Wang Tiles.** A *Wang tile* is a unit square with colored edges. A tiling by a set of Wang tiles is a discrete tiling by tiles of the set such that along each edge the colors match on both sides. The *Domino problem* is the following decision problem: given a finite set of Wang tiles, decide whether it admits a tiling.

**Theorem 1 (Berger [2]).** *The Domino problem is undecidable.*

The *Polyomino problem* is the following decision problem: given a finite set of polyominoes, decide whether it admits a tiling. By a reduction from the Domino problem to the Polyomino problem, Golomb [4] proved the undecidability of the Polyomino problem.

**Theorem 2 (Golomb [4]).** *The Polyomino problem is undecidable.*

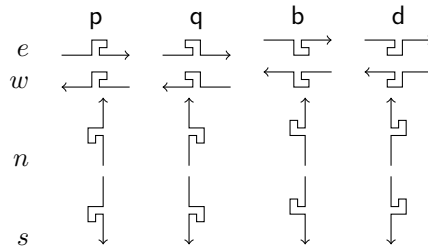
The reduction proceeds as follows. Given a finite set of Wang tiles, Golomb encodes each tile into a big squarish polyomino. Special bumps and dents are added to the corners of the tiles to force both alignment and orientation of the tiles: if one of the encoding polyominoes appears with an orientation, all the other tiles of the tiling have to use the same orientation. Special bumps and dents are used along the sides of the big polyominoes to encode the colors of the Wang tiles. Quotienting the set of tilings of the set of encoding polyominoes by isometries, it is in bijection with the set of tilings of the given set of Wang tiles.

**Dented Polyominoes.** A *dented polyomino* is a polyomino with edges labeled by a shape and an orientation. The four possible orientations  $\{p, q, b, d\}$  and their interpretation depending on the direction of the edge are depicted on Table 1 for a sample shape. On a contour word, inside shapes define bumps and outside shapes define dents. A tiling by a set of dented polyominoes is a tiling by the corresponding set of polyominoes where bumps and dents match along edges.

Dented polyominoes provide a convenient tool to construct complicated sets of polyominoes. These polyominoes with puzzle bumps and dents can be easily converted into polyominoes.

**Lemma 2.** *Every finite set of dented polyominoes can be encoded as a finite set of polyominoes such that their sets of tilings are in one-to-one correspondence.*

**Table 1.** Encoding of bumps and dents orientation



*Proof.* In order to guarantee that bumps and dents do not interfere with the matching conditions of polyominoes, the idea is to rescale the polyominoes. For all  $k \in \mathbb{Z}^+$ , a  $k$ -rescaling of a set of polyominoes consists into scaling the polyominoes by a factor  $k$ , *i.e.*, replacing each unit square by a square of  $k$  by  $k$  unit squares. Tilings are preserved by rescaling: the set of tilings of a set of polyominoes is in one-to-one correspondence with the set of tilings of its  $k$ -rescaling.

To encode a finite set of dented polyominoes into a finite set of polyominoes: first, rescale the set of polyominoes by a factor far bigger than the size of any shape of its bumps and dents; then, add bumps and dents in the middle of each rescaled edge. □

## 2 Tiling with a Fixed Number of Polyominoes

The  $k$ -Polyomino problem is the following decision problem: given a set of  $k$  polyominoes, decide whether it admits a tiling. This section is dedicated to the proof of the following theorem.

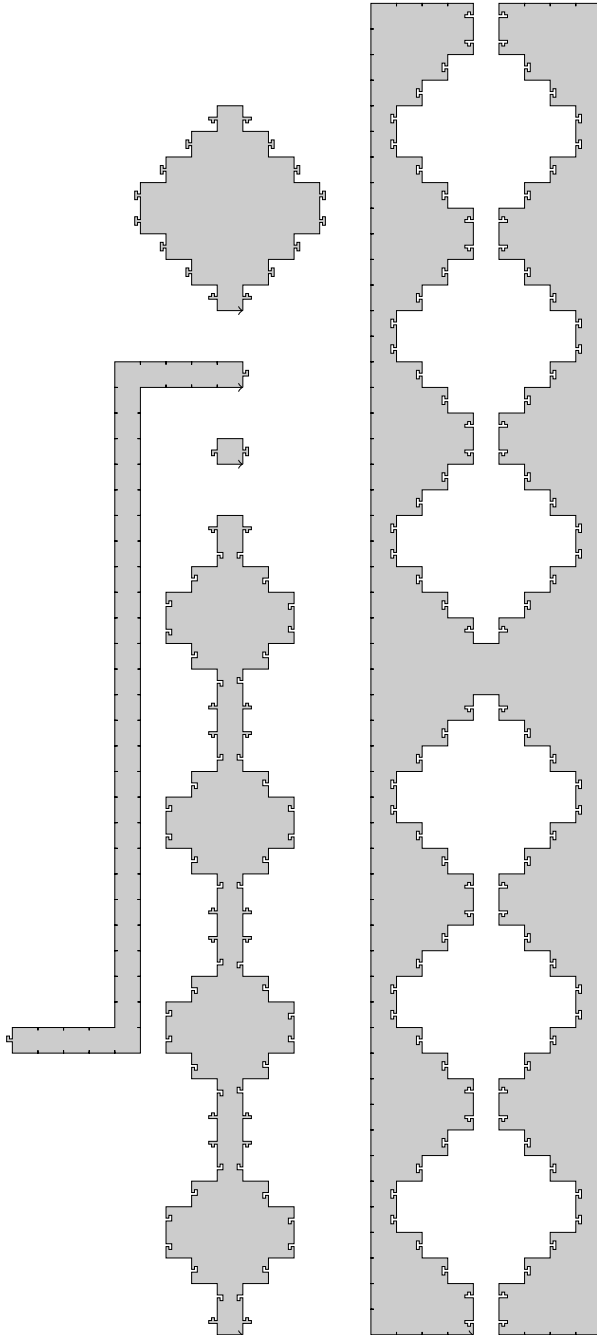
**Theorem 3.** *The 5-Polyomino problem is undecidable.*

We will proceed by reduction of the Domino problem. Given a finite set  $\tau$  of Wang tiles, we construct a set of 5 dented polyominoes  $P(\tau)$  such that, up to isometry, the set of tilings of  $\tau$  is in one-to-one correspondence with the set of tilings of  $P(\tau)$ . The proof goes as follows. First, we describe the construction of  $P(\tau)$ . Then, we explain how to encode any tiling of  $\tau$  by a tiling of  $P(\tau)$ . Finally, we show that any tiling of  $P(\tau)$  encodes a tiling of  $\tau$ .

### 2.1 Encoding a Set of Wang Tiles

Let  $\tau$  be a set of  $N$  Wang tiles. The set of dented polyominoes  $P(\tau)$  consists of the following 5 tiles, represented in Fig. 3:

- meat** encodes all tiles of the set  $\tau$  sequentially;
- jaw** acts as a selector to select exactly one tile of the *meat*;
- filler** is used for padding the blank leaved by the *meat* inside the *jaw*;
- tooth** erases the bits on the *meat* so that it fits inside the *jaw*;
- wire** links *meat* pieces together to verify tiling constraints.



From left to right and bottom to top : *meat, tooth, wire, filler, jaw*  
Notice that here  $N = 4$  and  $k = 3$  to fit the page (in the text  $k > 4$ ).

**Fig. 3.** Tiles (*rotated to fit in page*)

More formally, the dented polyominoes use 4 different shapes for bumps and dents, detailed on Table 2.

**Table 2.** Types of bumps and dents

	<i>blank</i>	<i>bit</i>	<i>marker</i>	<i>inside</i>
shape				
notation		<i>b</i>	<i>m</i>	<i>i</i>
order	1	4	4	2
bump		<i>wire, tooth</i>	<i>meat, filler</i>	<i>tooth, filler</i>
dent		<i>meat</i>	<i>jaw</i>	<i>jaw</i>

Let  $k$  be a large enough integer and choose an encoding on  $k - 4$  bits of the colors of the set of Wang tiles (horizontal and vertical colors can use different encodings). Let  $(a_j^i), (b_j^i), (c_j^i), (d_j^i)$  be respectively the north, east, south, and west binary encoding of the tiles where  $i$  is the tile index from 1 to  $N$  and  $j$  is the bit index from 1 to  $k - 4$ . Let  $(a_j^i)$  be the encoding of the  $k$  bits, by adding prefix 00 and suffix 01, of  $(a_j^i)$  on the alphabet  $\{b, d\}$ . Let  $(b_j^i)$  be the encoding of the  $k$  bits, by adding prefix 10 and suffix 11, of  $(b_j^i)$  on the alphabet  $\{p, q\}$ . Let  $(c_j^i)$  be the encoding of the  $k$  bits, by adding prefix 00 and suffix 01, of  $(c_j^i)$  on the alphabet  $\{b, d\}$ . Let  $(d_j^i)$  be the encoding of the  $k$  bits, by adding prefix 10 and suffix 11, of  $(d_j^i)$  on the alphabet  $\{p, q\}$ . The dented polyominoes are given by their contour words on Table 3.

**Table 3.** Contour encoding of the tiles

*tooth:*  $e_b^b n w_b^q s$

*wire:*  $e_b^b n^5 w^{2N(k+1)+1} n^4 w_b^p s^5 e^{2N(k+1)+1} s^4$

*filler:*  $e_m^b (s e_b^b)^k (e_b^b n)^k e_m^d n w_m^q (n w_b^p)^k (w_b^p s)^k w_m^p s$

*jaw:*  $e_m^q \left( e_m^p (n e_b^p)^k (e_b^p s)^k e_m^q \right)^{N-1} s \left( w_m^d (s w_b^d)^k (w_b^d n)^k w_m^b \right)^{N-1} w_m^d s^4 e^{2(N-1)(2k+2)+4} n^4$

$w_m^b \left( w_m^d (s w_b^d)^k (w_b^d n)^k w_m^b \right)^{N-1} n \left( e_m^p (n e_b^p)^k (e_b^p s)^k e_m^q \right)^{N-1} e_m^p n^4 w^{2(N-1)(2k+2)+4} s^4$

*meat:*  $\prod_{i=1}^N \left( e_m^b \prod_{j=1}^{k-1} \left( e_b^{a_j^i} s \right) e_b^{a_k^i} \prod_{j=1}^{k-1} \left( e_b^{b_j^i} n \right) e_b^{b_k^i} e_m^d \right) n$

$\prod_{i=N}^1 \left( w_m^q \prod_{j=k}^2 \left( w_b^{c_j^i} n \right) w_b^{c_1^i} \prod_{j=k}^2 \left( w_b^{d_j^i} s \right) w_b^{d_1^i} w_m^p \right) s$

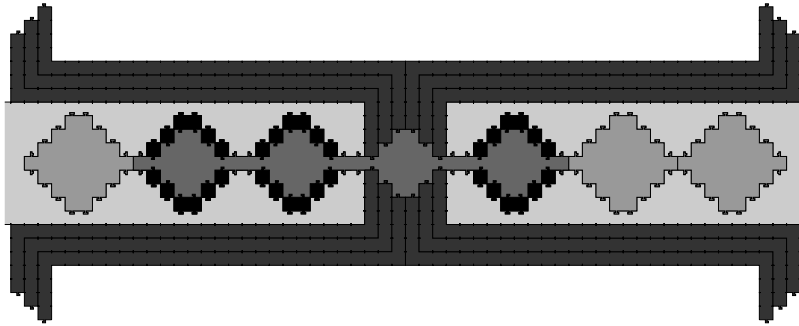
Notice the following important properties of these tiles. The *meat* consists of a sequence of  $N$  diamonds decorated with *bit* shapes with a prefix and a suffix *marker* shape pointing to the diamond. If one connects a *tooth* in each *bit* dent of a diamond of the *meat*, the diamond becomes similar to the *filler*. Moreover, both inside parts of a *jaw* consists of  $N - 1$  places to put a *filler* plus a *marker* shape at the entry of the *jaw*, pointing outside.

### 2.2 Encoding Tilings by Wang Tiles

Let us first prove the following lemma.

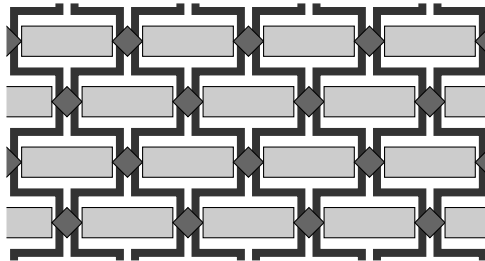
**Lemma 3.** *Every tiling by  $\tau$  can be encoded as a tiling by  $P(\tau)$ .*

The set of dented polyominoes is designed to encode a Wang tile by selecting one diamond of a *meat*, hiding the other diamonds using two *jaws* on the left and the right, padding inside the *jaws* with *teeth* and *fillers*, as represented in Fig. 4. The colors of the tile are propagated to the four neighbor tiles using *wires*.



**Fig. 4.** Encoding of a Wang tile including inter-tiles wires

Let  $T \in \tau^{\mathbb{Z}^2}$  be a tiling by  $\tau$ . Each north-west diagonal ( $x + y = i$  for the  $i$ th diagonal) is encoded as a line of Wang tile encodings where a *jaw* connects tile  $T(x, i - x)$  to tile  $T(x + 1, i - x - 1)$ . These lines of encoding are put on top of each other with a slight translation so that tile  $T(x, i - x)$  is connected by *wires* to  $T(x + 1, i - x)$ ,  $T(x, i - x + 1)$ ,  $T(x - 1, i - x)$  and  $T(x, i - x - 1)$ , as represented in Fig. 5. Notice that the choices made for **a**, **b**, **c**, and **d** permits such a connection only if the Wang tiling is valid. Thus, one obtains a tiling of the dented polyominoes.



**Fig. 5.** Wiring of Wang tiles

### 2.3 Every Tiling Encodes a Tiling

Now, we prove the following lemma.

**Lemma 4.** *Every tiling by  $P(\tau)$  encodes a tiling by  $\tau$ .*

Consider a tiling by the dented polyominoes. We first show that it must contain a *jaw*. Consider any tile of the tiling. If it is a *jaw*, we are done. Examine Table 2. If it is a *meat* or a *filler*, it has a *marker* bump that should be linked to a dent only found on a *jaw*. If it is a *tooth*, it has an *inside* bump that should be linked to a dent only found on a *jaw*. Finally, if it is a *wire*, it has *bit* bump that should be linked to a *meat*, itself connected to a *jaw*.

Consider a *jaw* tile of the tiling. To fill all the *marker* bumps, only *filler* and *meat* tiles can be used. As *fillers* have *inside* bumps, they can only be used completely inside the *jaw*. Thus, the *markers* on the extremities of the *jaw* have to be filled by the dents of a *meat*. Consider the *meat* that fills a *marker* at the extremity of the *jaw*. As *marker* bumps only appear inside *jaws*, the *marker* dent on the other side of the diamond of the *meat* next to the *jaw* has to be at the extremity of a next *jaw*. The only possibility to fill the gap in between the *jaw* and the *meat* locking its extremity is to use *fillers* and *teeth*. By now, we have proved that each *jaw* of the tiling appears in a biinfinite line (or column if rotated) of alternating *jaws* and *meats* where each *meat* has exactly one selected diamond outside the *jaws*.

Consider now the diamond of each *meat* appearing outside the *jaws*: it has *bit* bumps. A *bit* dent is found only on *teeth* and *wires*. As a *tooth* cannot appear outside a *jaw* (it as an *inside* dent), only *wires* can be connected to these bumps. Consider the two *bit* bumps side by side at the top of the diamond hill. The only possibility for two *wires* to appear side by side is to have the left one point to the left and the right one point to the right. This enforces all the *wires* in between the *jaw* and the top hill *wire* to point in the same direction: left ones to the left and right ones to the right. Thus, every tiling by dented polyominoes consists of biinfinite lines (or columns) of selected *meat* diamonds connected by *wires* in a lattice way as on Fig. 5.

Due to isometries, it remains to prove that all the selected diamonds have the same orientation. This part is enforced by the prefix/suffix trick in the **a**, **b**, **c**, and **d** encoding: the only possibility for a diamond to be connected to another diamond is that prefix and suffix match, thus both should be oriented in the same way. Thus, the tiling is the image by an isometry of a tiling by Wang tiles. We have proved that every tiling by  $P(\tau)$  encodes a tiling by  $\tau$ , achieving the reduction.

## 3 Tiling by Translation

The  $k$ -Polyomino translation problem is the following decision problem: given a set of  $k$  polyominoes, decide if it admits a tiling by translation. Using the result of previous section, one obtains the following.



**Theorem 4.** *The 11-Polyomino translation problem is undecidable.*

To prove this theorem, for any finite set of Wang tiles  $\tau$ , construct a set of 11 polyominoes as follows. Consider the set of 5 dented polyominoes  $P(\tau)$ . To encode any tiling by  $\tau$  into a tiling by  $P(\tau)$  as done in the previous section, we use exactly:

- 1 transformation of *meat*;
- 1 transformation of *jaw*;
- 1 transformation of *filler*;
- 4 transformations of *wire*;
- 4 transformations of *tooth*.

The set of 11 polyominoes tiling by translation consists exactly of these tile transformations. These dented polyominoes admit a tiling if and only if the set of Wang tiles admits a tiling.

## 4 Going Further

What can be said about tilability for sets of less than 5 polyominoes? or less than 11 polyominoes for tilings by translation? In the case of 1 polyomino, it is decidable for tiling by translation and still open for tiling with isometries.

**Theorem 5 (Wijshoff and van Leeuwen [8], Gambini and Vuillon [10]).** *The 1-Polyomino translation problem is decidable in time quadratic in the size of the contour word.*

**Open Problem 1.** *Is the 1-Polyomino problem decidable?*

To prove the undecidability of the Polyomino problem, one has to be able to construct aperiodic sets of polyominoes. Ammann *et al* provide a set of 2 polygonal tiles with colors and 3 polygonal tiles with bumps and dents that admits only aperiodic tilings. This set is convertible into polyominoes.

**Theorem 6 (Ammann *et al.* [11]).** *There exists an aperiodic set of 3 polyominoes.*

**Theorem 7 (Ammann *et al.* [11]).** *There exists an aperiodic set of 8 polyominoes for tiling by translation.*

**Open Problem 2.** *Is the  $k$ -Polyomino problem decidable for  $3 \leq k < 5$ ?*

**Open Problem 3.** *For  $8 \leq k < 11$ , is the  $k$ -Polyomino translation problem decidable?*

## Acknowledgement

The author thanks Bruno Durand for challenging him with the decision problem of tiling the plane with a fixed number of polyominoes.

## References

1. Grünbaum, B., Shephard, G.C.: Tilings and patterns. A Series of Books in the Mathematical Sciences. W. H. Freeman and Company, New York (1989)
2. Berger, R.: The undecidability of the domino problem. *Memoirs American Mathematical Society* 66 (1966)
3. Golomb, S.W.: Tiling with polyominoes. *Journal of Combinatorial Theory* 1(2), 280–296 (1966)
4. Golomb, S.W.: Tiling with sets of polyominoes. *Journal of Combinatorial Theory* 9(1), 60–71 (1970)
5. Post, E.L.: A variant of a recursively unsolvable problem. *Bulletin of the American Mathematical Society* 52, 264–268 (1946)
6. Claus, V.: Some remarks on PCP(k) and related problems. *Bulletin of the EATCS* 12, 54–61 (1980)
7. Matiyasevich, Y., Sénizergues, G.: Decision problems for semi-thue systems with a few rules. *Theoretical Computer Science* 330(1), 145–169 (2005)
8. Wijshoff, H.A.G., van Leeuwen, J.: Arbitrary versus periodic storage schemes and tessellations of the plane using one type of polyomino. *Information and Control* 62(1), 1–25 (1984)
9. Beauquier, D., Nivat, M.: On translating one polyomino to tile the plane. *Discrete and Computational Geometry* 6(1), 575–592 (1991)
10. Gambini, I., Vuillon, L.: An algorithm for deciding if a polyomino tiles the plane. *Theoretical Informatics and Applications* 41(2), 147–155 (2007)
11. Ammann, R., Grünbaum, B., Shephard, G.: Aperiodic tiles. *Discrete and Computational Geometry* 8(1), 1–25 (1992)