Domino Tatami Covering Is NP-Complete

Alejandro Erickson and Frank Ruskey*

Department of Computer Science, University of Victoria, V8W 3P6, Canada

Abstract. A covering with dominoes of a rectilinear region is called *tatami* if no four dominoes meet at any point. We describe a reduction from planar 3SAT to Domino Tatami Covering. As a consequence it is therefore NP-complete to decide whether there is a perfect matching of a graph that meets every 4-cycle, even if the graph is restricted to be an induced subgraph of the grid-graph. The gadgets used in the reduction were discovered with the help of a SAT-solver.

1 Introduction

Imagine that you want to "pave" a rectilinear driveway on the integer lattice using 1 by 2 bricks. Sometimes this will be possible, but sometimes not, depending on the shape of the driveway. Abstractly, a rectilinear driveway D is a connected finite induced subgraph of the infinite planar grid-graph, and a paving with bricks corresponds to a perfect matching. Since D is bipartite, various network flow algorithms can be used to determine whether there is a paving in low-order polynomial time.

However, an examination of typical paving patterns reveals that another constraint is often enforced/desired, probably for both aesthetic reasons and engineering reasons. The constraint is that no four bricks meet at a point. In some recent papers, this restriction has come to be known as the tatami constraint, because Japanese tatami mat layouts often adhere to it. The question that we wish to address in this paper is: What is the complexity of determining whether D has a paving also satisfying the tatami constraint? We will show that the problem is NP-complete.

A *rectilinear region*, R, is a finite subset of the integer grid. We say R *is covered by dominoes* if it is covered exactly by non-overlapping dominoes. We describe a polynomial reduction from the NP-complete problem planar 3SAT to Domino Tatami Covering (DTC). The gadgets used in the reduction were discovered with the help of a SAT-solver.

Definition 1 (Domino Tatami Covering (DTC)).

INSTANCE: *A rectilinear region, R, represented as n grid squares.*
QUESTION: *Can R be covered exactly by non-overlapping dominoes such that no four of them meet at any one point?*

* This research was supported in part by an NSERC Discovery Grant

T. Lecroq and L. Mouchard (Eds.): IWOCA 2013, LNCS 8288, pp. 140–149, 2013.

Domino tatami coverings have an interesting combinatorial structure, elucidated for rectangles in [12] and further in [5]. The results in these papers, as well as [6, 4] are enumerative, whereas in this paper we explore tatami coverings from a computational perspective. There is no comprehensive structure theorem for tatami coverings of rectilinear grids, but evidently much of the structure is still there, as is illustrated in Fig. 1. In contrast with other tatami-related results, however, we make no attempt to characterise this structure. Instead, our reduction relies on the interactions between coverings of a few specific regions that are discovered using a SAT-solver.

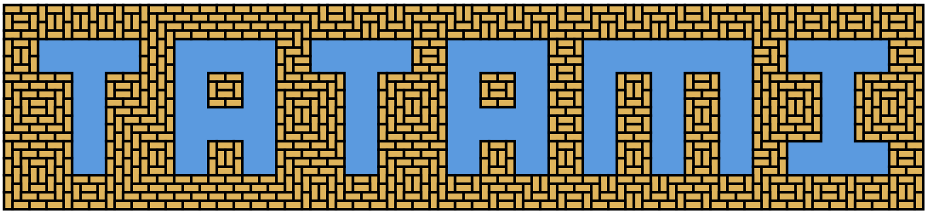


Fig. 1. A domino tatami covering of a rectilinear region, produced by a SAT-solver

There are some previous complexity results about tilings and domino coverings. Historically, perhaps the first concerned colour-constrained coverings, such as those of Wang tiles. It is well known, for example, that covering the $k \times k$ grid with Wang tiles is NP-complete ([8]). On the other hand tatami does not appear to be a special case of these, nor of similar colour restrictions on dominoes (e.g. [1, 13]).

A more closely related mathematical context is found, instead, among the graph matching problems discussed by Churchley, Huang, and Zhu, in [2]. In their paper, an *H-transverse matching* of a graph G, is a matching M, such that $G - M$ has no subgraph H. In a tatami covering of the rectilinear grid, G is a finite induced subgraph of the infinite grid-graph, H is a 4-cycle, and we require a perfect matching of the edges. In fact, if the matching is not required to be perfect, the problem is polynomial.

SAT-solvers have been applied to a broad range of industrial and mathematical problems in the last decade. Our reduction from planar 3SAT uses Minisat ([3]) to help automate gadget generation, as was also done by Ruepp and Holzer ([11]). It is easy to see that instances of other locally restricted covering problems can be expressed as satisfiability formulae, which suggests that SAT-solvers may provide a methodological applicability in hardness reductions involving those problems.

2 Preliminaries

Let ϕ be a CNF formula, with variables U, and clauses C. The formula is *planar* if there exists a planar graph $G(\phi)$ with vertex set $U \cup C$ and edges

$\{u, c\} \in E$, where one of the literals u or $\bar{u}$ is in the clause c. When the clauses contain at most three literals, ϕ is an instance of planar 3SAT (P3SAT), which is NP-complete ([9]).

We construct an instance of DTC which emulates a given instance, ϕ, of P3SAT, by replacing the vertices and edges of $G(\phi)$ with a rectilinear region, $R(\phi)$, that can be tatami-covered with dominoes if and only if ϕ is satisfiable. Let $n = |U \cup C|$. In Sec. 4 we show that $R(\phi)$ can be created in $\mathcal{O}(n)$ time, and that it fits in an $\mathcal{O}(n) \times \mathcal{O}(n)$ grid, by using Rosenstiehl and Tarjan's algorithm ([10]).

3 Gadgets

In this section we describe wire, NOT gates, and AND gates, which form the required gadgets. The functionality of our gadgets depends on the coverings of a certain 8×8 subgrid.

Lemma 1. *Let R be a rectilinear grid, with an 8×8 subgrid, S. If a domino crosses the boundary of S in a domino tatami covering of R, then at least one corner of S is also covered by a domino that crosses its boundary.*

Proof. Suppose R is covered by dominoes, and consider those dominoes which cover S. Such a cover may not be exact, in the sense that a domino may cross the boundary of S. If we consider all such dominoes to be monominoes within S, we obtain a monomino-domino covering of S. This covering inherits the tatami restriction from the covering of R, so it is one of the 8×8 monomino-domino coverings enumerated in [6] (and/or [5]).

The proof of Lemma 4 in [6] (third paragraph) states that there is a monomino in at least one corner of S if $0 < m < n$; Corollary 2 of [5] states that there is a monomino in at least one corner of S if $m = n$ (see examples in Fig. 2). This monomino corresponds to a domino which crosses the boundary in a corner of S, as required. □

The rectilinear region $R(\phi)$ incorporates a network of 8×8 squares, whose centres reside on a $16\mathbb{Z} \times 16\mathbb{Z}$ grid, and whose corners form part of the boundary of $R(\phi)$. Lemma 1 implies that no domino may cross their boundaries, and thus each one must be covered in one of the two ways shown in Fig. 3(a). (For proofs see [12] and Exercise 215, Sec. 7.1.4 in [7]).

The coverings of these squares are related to each other by connecting regions. The part of an 8×8 square which borders on a connector may be covered either by two tiles, denoted by F to signify "false", or three tiles, denoted by T to signify "true" (see Fig. 3(a)). Note that the covering of a square is not T or F by itself, because connectors below and beside it would meet the square at differing interfaces.

A connector, which imposes a relationship between the coverings of a set of 8×8 squares, is verified by showing that it can be covered if and only if the

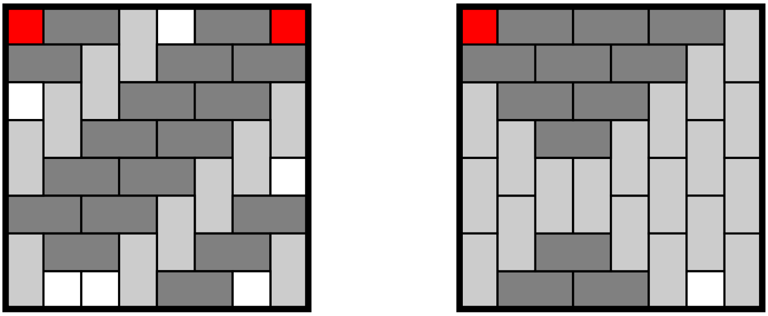
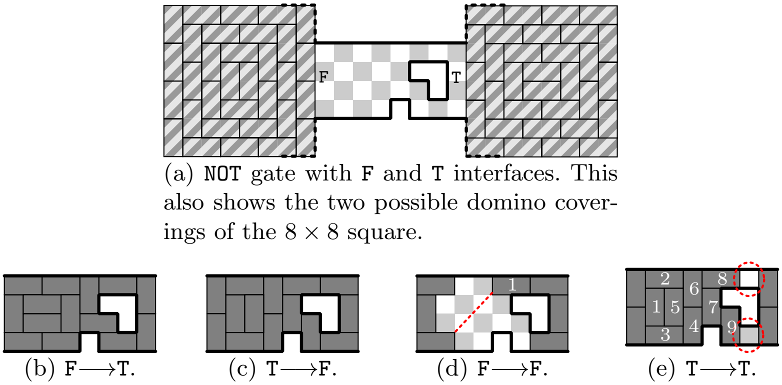


Fig. 2. All monomino-domino tatami coverings of the square have at least one monomino in their corners (see [5, 6]). The squares in $R(\phi)$ have isolate corners, so these must be covered in exactly one of the two ways given by Exercise 7.1.4.215 in [7], shown in the left and right-hand cross-hatched squares in Fig. 3(a).

relationship is satisfied. The connectors we describe were generated with SAT-solvers, but they are simple enough that we can verify them by hand, as is done below.

NOT gate. The NOT gate interfaces with two 8×8 squares (see Fig. 3(a)), and can be covered if and only if these squares are covered with differing configurations.

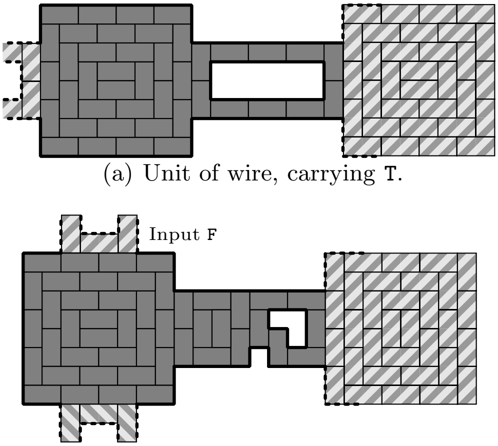


(a) NOT gate with F and T interfaces. This also shows the two possible domino coverings of the 8×8 square.

(b) F⟶T. (c) T⟶F. (d) F⟶F. (e) T⟶T.

Fig. 3. NOT gate can be covered if and only if the input differs from the output. Numbered tiles indicate the (non-unique) ordering in which their placement is forced. Red dotted lines indicate how domino coverings are impeded in *(d)* and *(e)*.

Wire gadget. Wire transmits T or F through a sequence of squares (see Fig. 4(a)). A turn may incorporate a NOT gate in order to maintain the same configuration (see Fig. 4(b)).



(a) Unit of wire, carrying T.

(b) Wire branch and turn, carrying F.

Fig. 4. Wire gadget

AND gate. The AND gate interfaces with two 8×8 input squares, and one output square (see Fig. 5). It can be covered with dominoes if and only if the output value is the AND of the inputs (see Figs. 6 and 7).

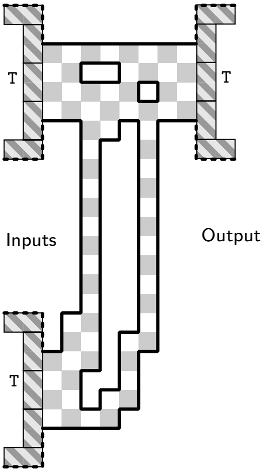
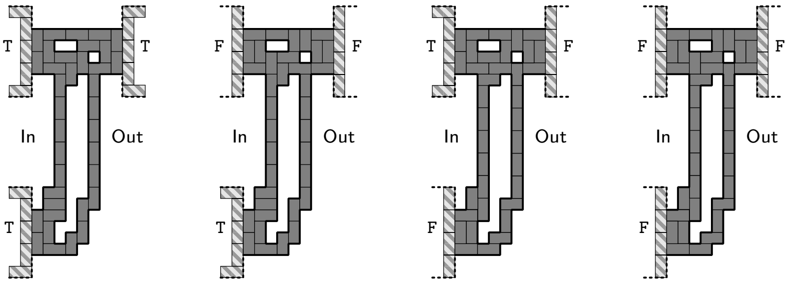


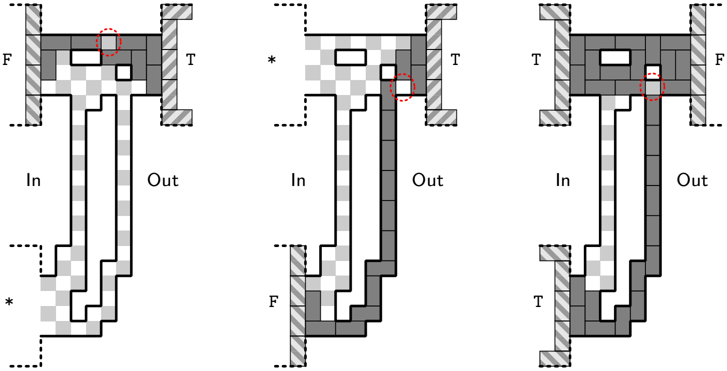
Fig. 5. AND gate with input (T,T)

Variable gadget. We use a vertical segment of wire. The variable gadget is set to T or F by choosing the appropriate covering of one of its 8×8 squares. Its value (or its negation) is propagated to clause gadgets via horizontal wire gadgets, representing edges.



(a) TT⟶T. (b) TF⟶F. (c) FT⟶F. (d) FF⟶F.

Fig. 6. AND gate coverings



(a) *F⟶T. (b) F*⟶T. (c) TT⟶F.

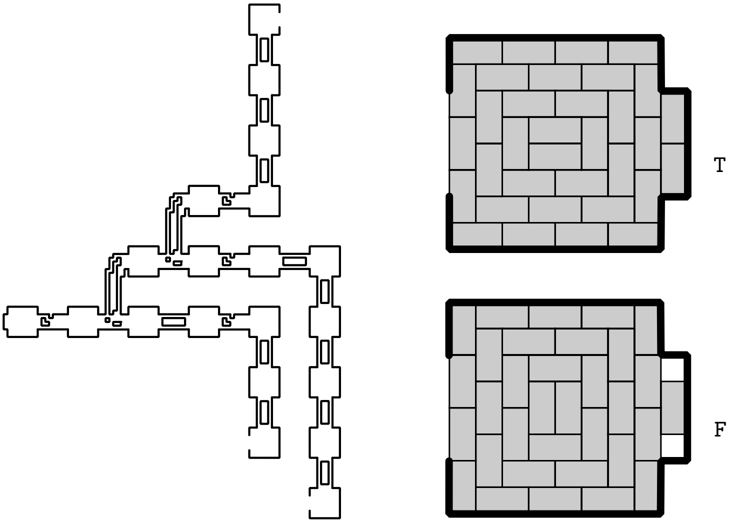
Fig. 7. Impossible AND gate coverings, where * denotes F or T

Clause gadget. The clause gadget is a circuit for $\neg(\bar{a} \wedge (\bar{b} \wedge \bar{c}))$, or the equivalent with fewer inputs, ending in a configuration that can be covered if and only if the output signal of the circuit is T. To satisfy the layout requirements, the inputs to the clause are vertically translated by wire (see Fig. 8).

4 Layout

Let $G(\phi)$ be a planar embedding of the Boolean 3CNF formula ϕ, using Rosenstiehl and Tarjan's ([10]) algorithm, so that each vertex is represented by a vertical line segment, and each edge is represented by a horizontal line segment. All parts lie on integer grid lines, inside of a $\mathcal{O}(n) \times \mathcal{O}(n)$ grid, where $n = |U \cup C|$, and the embedding is found in $\mathcal{O}(n)$ time.

There exists a constant K, which is the same for any planar 3CNF formula, such that $G(\phi)$ can be scaled to fit on the $nK \times nK$ grid, and its parts replaced by the gadgets described above. Each gadget has a constant number of grid squares, which ensures that $R(\phi)$ has $\mathcal{O}(n^2)$ grid squares altogether.



(a)

(b) End of clause.

Fig. 8. A three input clause gadget from the circuit $\neg(\bar{a} \wedge (\bar{b} \wedge \bar{c}))$. Vertical wire translates horizontal inputs without changing the signal. The end of the clause is coverable if and only if its signal is `T`.

The variable gadget is connected to edges by branches. The layout of $G(\phi)$ prevents conflicts between edges meeting the variable gadget on the same side, while two edges can meet the left and right sides of the variable gadget without interfering with each other. The inputs of the clause gadget are symmetric, so there are no conflicts when connecting these to horizontal edges (see Fig. 8(a)).

Example. The planar Boolean formula from Fig. 1 in [9] gives the `DTC` instance in Fig. 9.

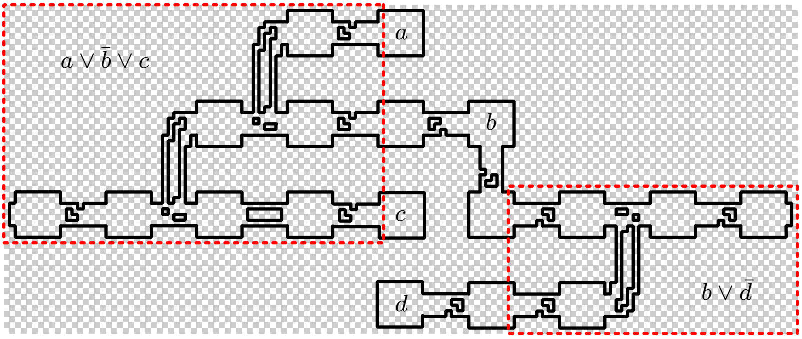


Fig. 9. An instance of `DTC` for the formula $(a \vee \bar{b} \vee c) \wedge (b \vee \bar{d})$

5 SAT-Solver

The search for logical gates required fast testing of small DTC instances. We reduced DTC to SAT in order to use the SAT-solver, Minisat ([3]), and efficiently test candidate regions connecting 8×8 squares while satisfying the conditions of the desired gate. The DTC solver was also allowed to make certain decisions about the region, rather than simply testing regions generated by another program.

Our search algorithm requires the following inputs:

- an $r \times c$ rectangle of grid squares, partitioned into pairwise disjoint sets K, X, A, C; and,
- a set of partial (good) coverings, G, and partial (bad) coverings, B, of C.

The output, R, is the region $A' \cup K$, where $A' \subseteq A$, which satisfies the following constraints.

(g) There exist coverings of R which form partial tatami domino coverings with each element of G.
(b) There exists no covering of R which forms a partial tatami domino covering with an element of B.

The outer loop of the search algorithm calls the SAT-solver to find a region that satisfies all elements of G, and avoids a list of forbidden regions, which is initially empty. Upon finding such a region, the inner loop checks whether the region satisfies any element of B. The search succeeds when *(g)* and *(b)* are both satisfied, and fails if the outer loop's SAT instance has no satisfying assignment.

The search space grows very quickly for several reasons, not least of which is the fact that 2^{160} regions are possible within the 20×8 rectangle occupied by our AND gate (if corners are allowed to meet one another). In addition, the list of forbidden regions, L, becomes too large for the SAT-solver to handle efficiently.

We used two heuristics on the inputs to obtain a feasible search. The first was searching for a smaller AND gate, which we modified to fit the placement of the 8×8 squares. The second was choosing forbidden squares, X, and required squares, K, to reduce the number of trivially useless regions that are tested.

5.1 DTC as a Boolean Formula

The SAT instances used above are modifications of a formula which is satisfiable if and only if a given region has a domino tatami covering.

Let R be the region we want to cover, and consider the graph whose vertices are the grid squares of R, and whose edges connect vertices of adjacent grid squares. Let H be the set of horizontal edges and let V be the set of vertical edges. The variables of the SAT instance are $H \cup V$, and those variables set to true in a satisfying assignment are the dominoes in the covering. The clauses are as follows, where $h, h' \in H$ and $v, v' \in V$.

1. Ensure a matching: For each pair of incident horizontal edges (h, h'), require the clause $\bar{h} \vee \bar{h}'$, and similarly for (v, v'), (h, v).

2. Ensure the matching is perfect: For each set of edges $\{h, h', v, v'\}$, which are incident to a vertex, require the clause $h \vee h' \vee v \vee v'$.
3. Enforce the tatami restriction: For each 4-cycle, $hvh'v'$, require the clause $h \vee h' \vee v \vee v'$.

6 Variations and Future Work

There are other locally constrained covering problems that are easily represented as Boolean formulae. Some of these are obviously polynomial, such as monomino-domino tatami covering, but others may be NP-complete. SAT-solvers can sometimes be used in such problems to create elaborate gadgets, which may help find a hardness reduction.

An example problem, whose computational complexity is open, is Lozenge-only Tatami Covering. This problem is the decision about whether or not a finite sub-grid of the triangular lattice can be covered with lozenges, such that no 5 lozenges meet at any point. A structure similar to that of tatami coverings occurs for this constraint (see Fig. 10).

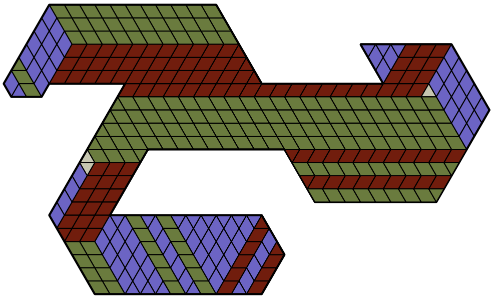


Fig. 10. A triangle-lozenge tatami covering

Our main question about DTC is the complexity of the case where the region is simply connected (no holes). It seems likely that the problem is still NP-complete, but a completely new approach will be required.

Secondarily, we are interested in H-transverse perfect matchings for H and G other than C_4 and grid-graphs. Are there other H-transverse perfect matchings of interest which induce a tatami-like global structure in the containing graph?

Another variant, mildly advocated by Don Knuth (personal communication), concerns inner corners of the coverings, such as occurs at the upper left in the letter T in Figure 1. If corners such as these, where a + occurs, are forbidden but corners such as the upper right one in the I are allowed (a $\perp$ shape or one of its rotations), then the nature of tatami coverings changes. The complexity of such coverings is unknown.

Acknowledgement. We thank Bruce Kapron for useful conversations about this problem, and Don Knuth for his comments on an earlier draft. Part of this research was conducted at the 9th McGill–INRIA Workshop on Computational Geometry.

References

[1] Biedl, T.: The complexity of domino tiling. In: Proceedings of the 17th Canadian Conference on Computational Geometry (CCCG), pp. 187–190 (2005)

[2] Churchley, R., Huang, J., Zhu, X.: Complexity of cycle transverse matching problems. In: Iliopoulos, C.S., Smyth, W.F. (eds.) IWOCA 2011. LNCS, vol. 7056, pp. 135–143. Springer, Heidelberg (2011)

[3] Eén, N., Sörensson, N.: An extensible sat-solver. In: Giunchiglia, E., Tacchella, A. (eds.) SAT 2003. LNCS, vol. 2919, pp. 502–518. Springer, Heidelberg (2004)

[4] Erickson, A., Ruskey, F.: Enumerating maximal tatami mat coverings of square grids with v vertical dominoes. Submitted to a Journal (2013), http://arxiv.org/abs/1304.0070

[5] Erickson, A., Ruskey, F., Schurch, M., Woodcock, J.: Monomer-dimer tatami tilings of rectangular regions. The Electronic Journal of Combinatorics 18(1) #P109, 24 pages (2011)

[6] Erickson, A., Schurch, M.: Monomer-dimer tatami tilings of square regions. Journal of Discrete Algorithms 16, 258–269 (2012)

[7] Knuth, D.E.: The Art of Computer Programming: Combinatorial Algorithms, Part 1, 1st edn., vol. 4A. Addison-Wesley Professional (January 2011)

[8] Lewis, H.R.: Complexity of solvable cases of the decision problem for the predicate calculus. In: 19th Annual Symposium on Foundations of Computer Science, pp. 35–47 (October 1978)

[9] Lichtenstein, D.: Planar formulae and their uses. SIAM Journal on Computing 11(2), 329–343 (1982)

[10] Rosenstiehl, P., Tarjan, R.E.: Rectilinear planar layouts and bipolar orientations of planar graphs. Discrete & Computational Geometry 1(1), 343–353 (1986)

[11] Ruepp, O., Holzer, M.: The computational complexity of the Kakuro puzzle, revisited. In: Boldi, P. (ed.) FUN 2010. LNCS, vol. 6099, pp. 319–330. Springer, Heidelberg (2010)

[12] Ruskey, F., Woodcock, J.: Counting fixed-height tatami tilings. The Electronic Journal of Combinatorics 16(1) #R126, 20 pages (2009)

[13] Worman, C., Watson, M.D.: Tiling layouts with dominoes. In: Proceedings of the 16th Canadian Conference on Computational Geometry (CCCG), pp. 86–90 (2004)