Dominos sur grilles trouées

Année 2020 – 2021

Liviu-Ștefan NEACȘU-MICLEA, élève de classe 12.

Encadrés par GHIȚĂ Luminița, JILINSCHI Rebeca (professeur de langue)

Établissements : Colegiul Național „B. P. Hasdeu" Buzău, Roumanie

Chercheur Chercheuse : Bogdan ENESCU, Robert BROUZET

1. Présentation du sujet

In this paper we will study the possibilities of covering a square grid with 2x1-sized dominos. The grid may contain zero or an arbitrary number of missing cells. We want to cover this grid with dominos, without skipping, leaving blank spaces and without covering the supposed holes of the grid. Of course, two dominos cannot overlap.

2. Annonce des conjectures et résultats obtenus

The subject can be approached in different methods, based on the way we mathematically model the problem. Each of them helps in a better understanding of a certain aspect of the matter. At first, by visualizing the grid as a chess table, multiple laws and patterns can be formulated which are essential in deciding whether the grid can be or not covered with dominos. Then, by rewriting the problem in a more abstract manner which makes use of bipartite graphs, it is possible to solve it programatically. Whilst a brute force backtracking supplies us with crucial information about how an algorithm should behave in a number of particular cases, we imagined a computation method which refrains to finding the integer solutions of a linear system of equations. Although the algorithm finds the solution for minor scale tests in reasonable time, there is undeoubtedly room for optimization. That includes especially looking for a solution which runs in polynomial time.

3. Texte de l'article

3.1. Introduction

In the next pages, the following terms will be used:

1) $\boldsymbol{n \times n}$ **Grid** or **Type $\boldsymbol{n}$ grid** = a square grid of length n, $n \in \mathbb{N}^*$;
2) **Even grid** = an $n \times n$ grid, where n is even;
3) **Odd grid =** an $n \times n$ grid, where n is odd;
4) Because the missing cells of the grid prevent being covered by a domino piece, we will call them **obstacles**. Similarly, the remaining cells will be called **free cells/positions**;

5) **Holes** = a set containing one or more adjacent obstacles;
6) **Free grid** = a grid without holes;
7) **Pavement** or **covering** = the operation consisting of placing a number of dominos in valid positions of the grid. The set of dominos placed this way will be called **configuration**;
8) **Maximal covering (pavement)** = a grid covering which uses the largest number of dominos possible;
9) **Complete covering (pavement)** = a pavement which covers every free cell of the grid;
A complete covering is a maximal covering.
10) **Valid grid** = a grid for which there exists at least a complete pavement;

3.2. Grid validation criteria

In the first place, we will try to take some of the non-valid cases out of the equation, based on a handful of simple remarks. In the following, let $\mathcal{G}_n$ be the set of $n \times n$ grids, $n \in \mathbb{N}$, $n \geq 2$. Obviously, type 1 grids cannot be covered with domino pieces. We assign each row and column of an $n \times n$ grid numbers from 1 to n, as shown in figure 1.

We color each cell in white (A), respectively black (N), in a way similar to a chess board, where the top left corner is white. Obstacles will be marked with red (R). Let C_A, C_N be the number of white, respectively black cells in a grid $G \in \mathcal{G}_n$. This grid coloring convention will be available throughout this document.

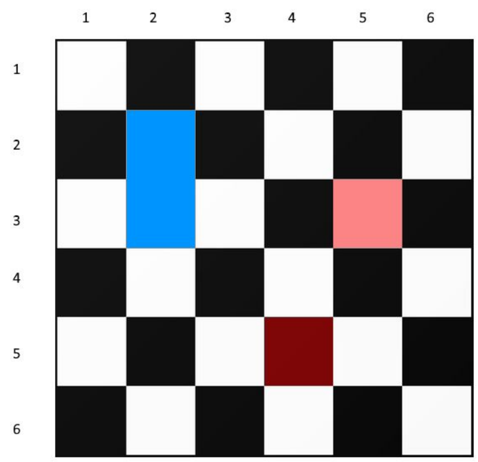


fig.1.

We call **Type-*A* obstacle** an obstacle at a position where, in a free grid, there would be a white cell. Let R_A be the number of type-A obstacles in a grid $G \in \mathcal{G}_n$. (fig. 1, position (3,5)).

We call **Type-*N* obstacle** an obstacle at a position where, in a free grid, there would be a black cell. Let R_N be the number of type-N obstacled in a grid $G \in \mathcal{G}_n$. (fig. 1, position (4,4))

We call **pair of obstacles** a set containing one type-A and one type-N obstacles.
Black and white cells are *free* positions.

In figure 1, the cell at coordinates (3,5) is a type-A obstacle, and the cell at coordinates (5,4) is a type-N obstacle. The blue rectangle represents a domino piece covering the free positions (2,2) and (3,2).

We define $C_R = R_A + R_N$ the total number of obstacles on the grid G. Moreover, the following relations take place:

$$C_A = \frac{n^2}{2} - R_A, \qquad C_N = \frac{n^2}{2} - R_N, \qquad if\ n\ is\ even;$$
$$C_A = \frac{n^2+1}{2} - R_A, \qquad C_N = \frac{n^2-1}{2} - R_N, \qquad if\ n\ is\ odd;$$

Let $G \in \mathcal{G}_n$ be some grid and D the number of dominos needed by a maximal covering of this grid. We observe that a domino covers exactly one white and one black cell. Obviously the D dominos cover D white cells and D black cells. For the grid to be valid, the maximal pavement needs to be a complete one, meaning that $C_A = D$ and $C_N = D$. Therefore, a first necessary condition for grid validity is $C_A = C_N$ (1). In other words, there must be the same number of black and white cells. Note that (1) holds not only for square grids, but also for any surface we need to cover with dominos.

Making the difference of the equalities above, we can find other relations equivalent to (1):

a) $R_A = R_N$, if n is even.
   A valid even grid has the same number of type-A and type-N obstacles.
   A valid even grid has an even number of obstacles.

b) $R_A = R_N + 1$, if n is odd.
   A valid odd grid has one type-A obstacles more than type-N obstacles.
   A valid odd grid has an odd number of obstacles.
   Observation: *A free odd grid is not valid.*

Another trivial case which cancels a grid's validity is when an off number of free positions is isolated from the rest of the grid by obstacles or the grid's sides, as can be seen in figure 2.

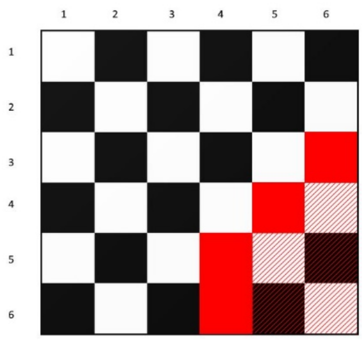
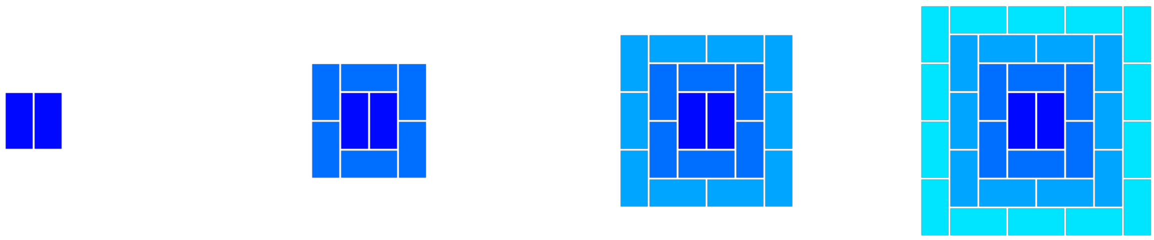


fig.2

3.3. Over the pavement method of an even grid with two obstacles

3.3.1. Conejcture. An even grid with a pair of obstacles is a valid grid.

Let $G \in \mathcal{G}_{2k}$, $k \in \mathbb{N}^\star$ ba a free even grid, completely covered by following this pattern:



k=1 *k=2* *k=3* *k=4*

fig.3

For each k, the configuration contains k square concentric „frames” built from domino pieces (we will call this a *„zero” configuration*). Let $\mathcal{F}_1, \mathcal{F}_2, \dots, \mathcal{F}_k \subset \mathcal{G}_{2k}$ be the sets of positions (i, j) covered by each frame. The frames count starts from the grid's center to the exterior.
Consider P, Q two positions on the grid G occupied by a pair of obstacles. We will try to change the „zero” configuration (by removing and adding dominos) until we have a complete covering.

Frame rotation and segmentation. Suppose there exists $i \in \mathbb{N}^\star$ such that the obstacles $P, Q \in \mathcal{F}_i$ are on the same frame. If the obstacles' positions are covered in „zero” configuration by the same domino piece, then we reach our goal by removing that particular domino. If the obstacles „affect” different dominos, then, “walking on the frame”, we remove that sequence of dominos that begins and ends with each of the obstacles.

An interesting property of the above-defined frames is that they are not directly dependant on the „zero” configuration, thus is it possible to rotate them through 90°, which inverts the places dominos connect on that frame (figures 4.a, 4.b-left). Similarly, we remove from the „rotated” frame the sequence of dominos between those two obstacles.

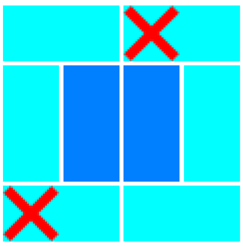


fig.4.a
Frame $\mathcal{F}_2$ is rotated through 90 degrees

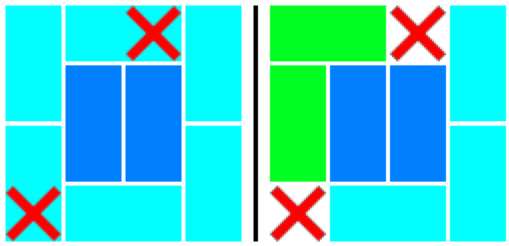


fig.4.b
left – „zero” configuration and the two obstacles
right – the „covering” induced by the rotated frame

Combining the dominos from the original and rotated frames results in creating a new, fragmented, „frame”, in which all cells, except for obstacles P and Q are covered by dominos. Consequently, we obstain a complete covering (figure 4.b-right).

Observation: The explained algorithm could be easily applied for a larger number of obstacles situated on the same frame (but not always!). Figure 5.c shows a complete covering created by the reunion of dominos sequences obtained from the original „zero” configuration frame (fig. 5.a) and the rotated frame (fig. 5.b). However, the grid in the figure 5.d, cannot be completely covered using this method.

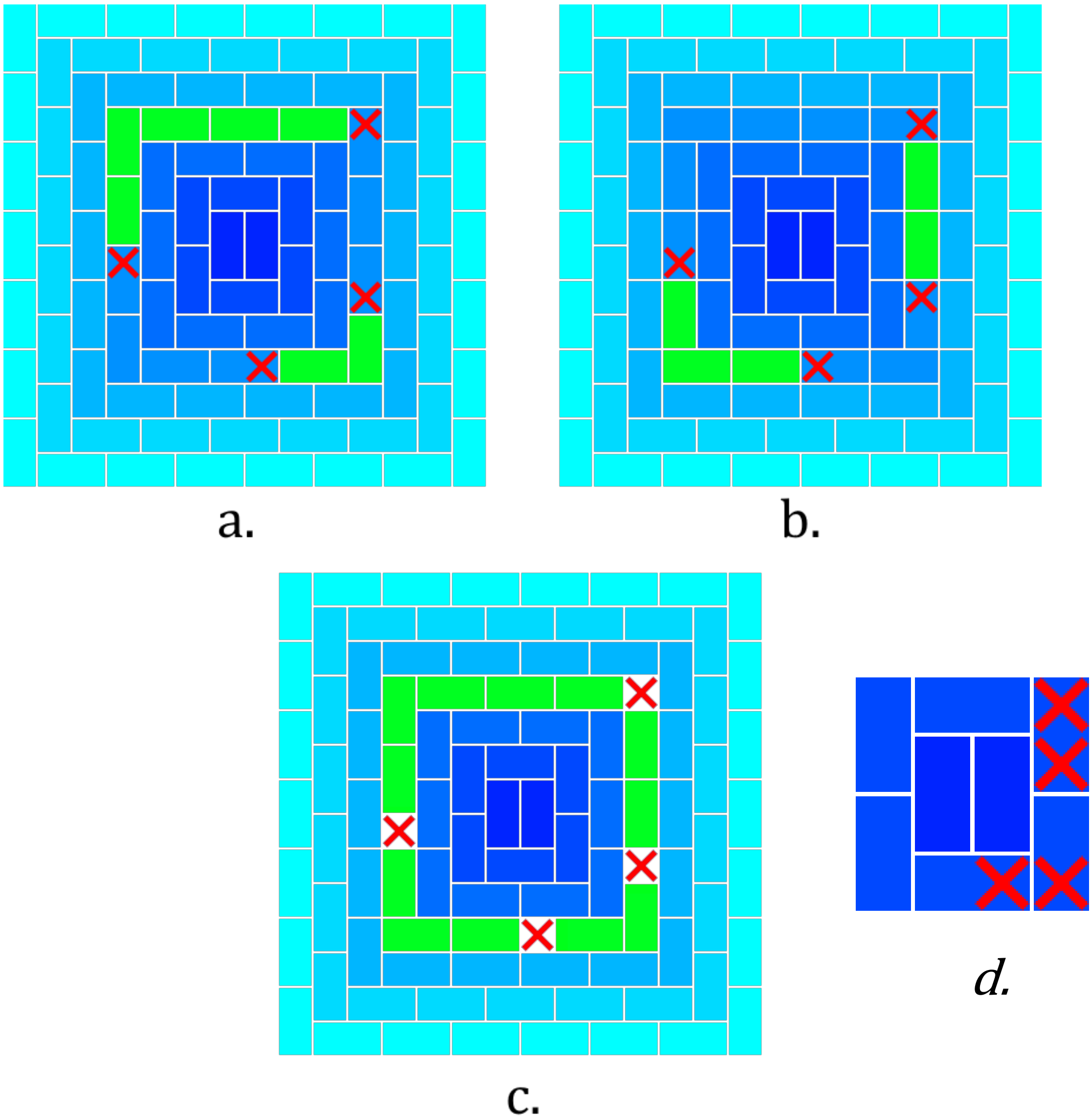


fig.5

We now suppose that the two obstacles belong to different frames ($P \in \mathcal{F}_i, Q \in \mathcal{F}_j, 1 \leq i < j \leq k$).

We define **obstacle injection into inferior frame** the process of replacing a domino of a frame r (either rotated or not) containing an obstacle with another domino, of opposite orientation, being as close as possible to the grid center (frame 1). The half of the new domino which intersects frame $r - 1$ will be seen as a new new obstacle for that frame (we will call it a **pseudo-obstacle**).

Observation: The pseudo-obstacole preserves the color/type (A/N) of the initial obstacle.

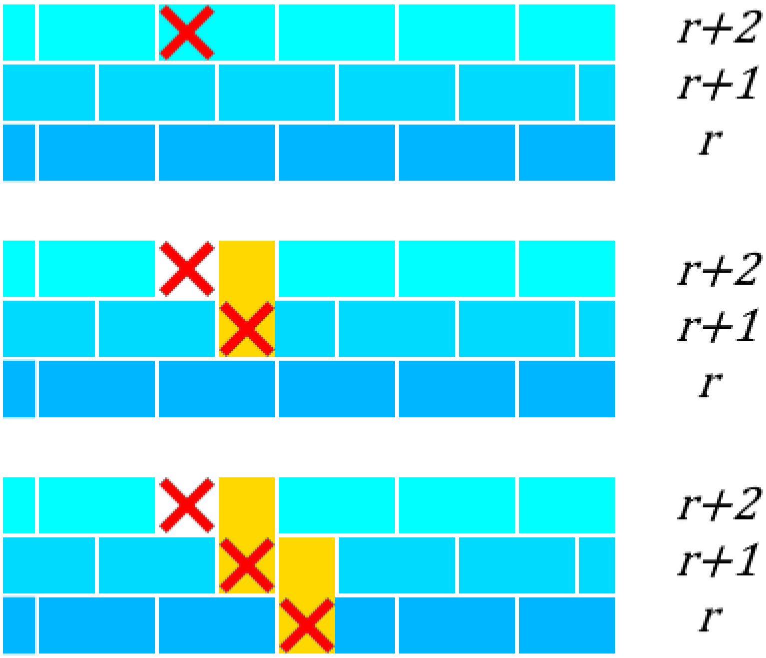


fig. 6 – pseudo-obstacle creation by succesive injection into frame r
Pseudo-obstacle is denoted by an „X" on the vertical dominos

We try, by rotating frames $i, \ldots, j$, to bring the dominos to a *convenable*[1] configuration which allows (succesive) obstacle injection from frame j to frame i. The obstale on frame i, along with the injected pseudo-obstacle creates the conditions which suggest performing *frame rotation and segmentation*. As a result, we obtain a fully covered even grid for each pair of type-A and type-N obstacles (fig. 7).

[1] It is not always necessary to rotate the frames in order to reach the result. The word *convenable* is used as a measure of efficiency and intuitive visualization of the algorithm (rotating the frame can minimize the number of altered dominos, like exposed in figure 8, where rotating frames 6 and 7 reduces the number of affected dominos by 88%). This happens due to the fact that *frame rotation affects the direction obstacles are injected in inferior frames*.

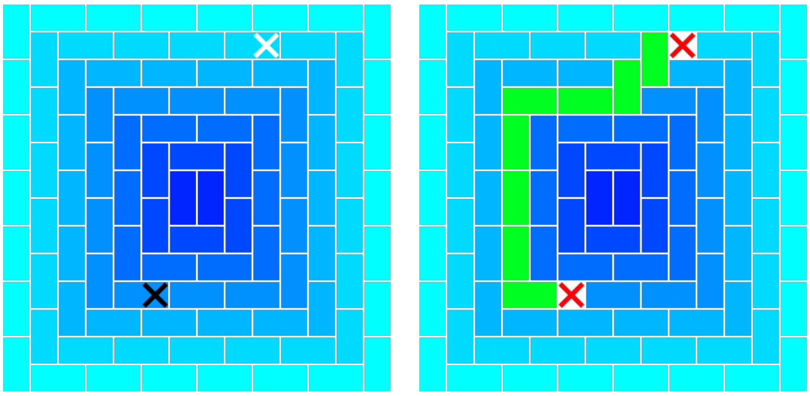
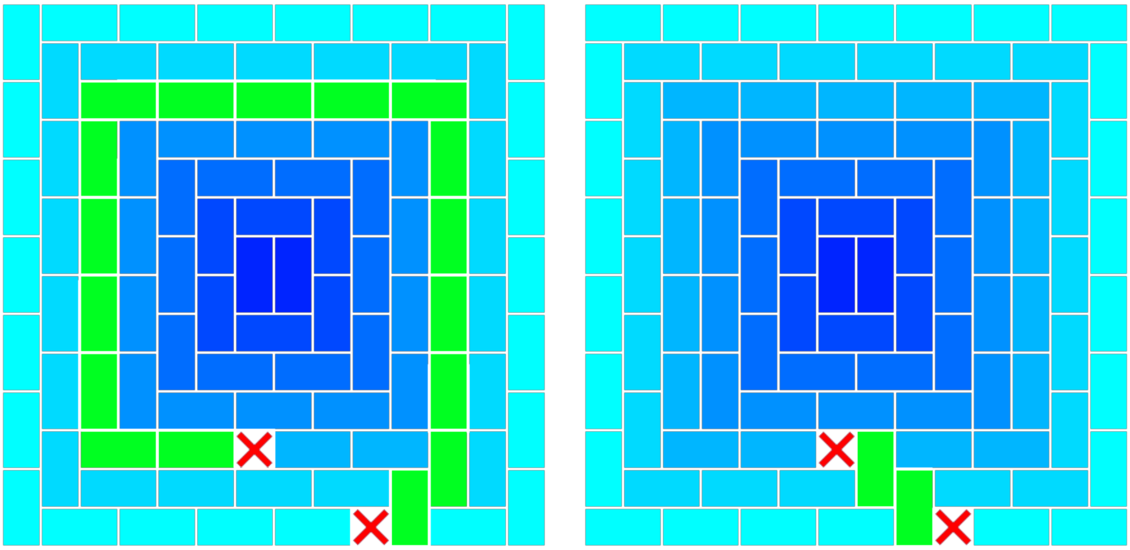


Fig. 7. *Exemple of complete grid pavement by injection and segmentation*

Consequently, the injection and rotation-segmentation methods applied on the „zero" configuration generated a complete covering of each even grid containing a single pair of obstacles, hence the **Conjecture 3.3.1** proves to be true.



a) complete covering by injection and segmentation

b) complete covering by injection and rotation-segmentation

fig. 8

3.3.2. Extensive pavement

Another approach for covering an even grid containing a pair of obstacles could be splitting the grid into a number of disjunctive surfaces and individually paving them following a certain pattern.
To start with, let's prove the following lemmas:

Lemma 1. A subset of grid G, *resembling a rectangle with corners of different colors, two of them being obstacles, can be completely covered using domino pieces.*

<u>*Proof:*</u> Corners not having the same color implies that at least a dimension of the given rectangle is even. More precisely, we can affirm that it is *an even-area rectangle*. We immediately deduce that number of white cells is equal to the number of black cells, which satisfies the relation (1) from chapter **2. ... Criteria**, thus the operation's validity is not contradicted. We will try to find a complete pavement of the rectangle with two missing corners. There are two cases to be treated:

a) Missing corners are on the same side of an even length. The rectangle is positioned such that the incomplete side is horizontally oriented and is exclusively paved with horizontal pieces, as shown in picture 9.a.
b) Missing corners are on the same diagonal (only possible when at least one dimension is odd). We place the rectangle with the odd side oriented horizontally. Fill the first and the last row

with horizontal pieces, then pave the remaining free cells with vertical dominos, as shown in picture 9.b.

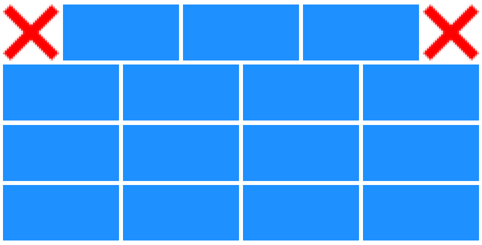


fig. 9.a

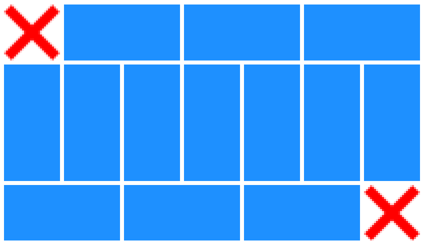


fig. 9.b

Lemma 2. *An even-area rectangle can be* *<u>borded</u>* *with domino pieces.*

<u>*Proof*:</u> We place the rectangle on a numbered square network such that the even side is horizontally oriented. Consider the top-left cell has coordinates $(1,1)$, whilst the bottom right cell has coordinates (w,h). We place $\frac{w}{2}$ dominos above the side $(1,1)-(w,1)$, without exceeding the side's extremities. Then, starting immediately below those pieces, we add vertical dominos, filling in the column 0, respectively $w+1$. The last domino placed on each column must reach (if rectangle height is odd) or surpass (if rectangle height is even) the below side of the rectangle, situated at row h. Finally, row $h+1$ is covered with $\frac{w}{2}$ horizontal pieces, if h is even, or $\frac{w}{2}+1$ pieces, if h is odd, such that the rectangle bordering is now complete.

Returning to our subject, we remind the notations P, Q for the two obstacles in an even type-n grid G. Let $\alpha=(i_P,j_P)$, $\beta=(i_Q,j_Q)$ be the coordinates of the obstacles. Suppose that $i_P\le i_Q$ and $j_P\le j_Q$. Consider now the rectangle U having P,Q as corners. Its dimensions are $w=|j_P-j_Q|+1$, and $h=|i_P-i_Q|+1$. Note that U can also be a *unit-rectangle* (one of its dimensions is 1). According to *Lemma 1*, rectangle U can be completely covered using dominos pieces. The problem refrains to showing that here is a complete pavement for $S=G\setminus U$.

If $\alpha=(1,1)$ and $\beta=(n,n)$, then $S=\emptyset$ and there is nothing to be proved.

If $\alpha=(1,1)$ and $\beta\neq(n,n)$, or $\alpha\neq(1,1)$ and $\beta=(n,n)$, then the surface S is *L-shaped*, and can be splitted in two rectangles. One of them, call it V, has either the width or the height equal to n, an even number. We prove that the other rectangle, W has at least one even side. We assume the opposite and try a proof by contradiction. Let $\mathcal{A}_U,\mathcal{A}_V,\mathcal{A}_W$ be the areas of the tree rectangles. The first two areas are even, whilst the last one is odd. Therefore, the sum $\mathcal{A}_U+\mathcal{A}_V+\mathcal{A}_W$ must be odd. But the reunion of these tree disjoint rectangles is exactly the type-n grid G which implies that $\mathcal{A}_U+\mathcal{A}_V+\mathcal{A}_W=n^2$ is odd. This results in a contradiction. It follows that the rectanles U and W have even areas, and can obviously be paved in a „boring" mode with wither horizontally or vertically oriented dominos, following (one of) the even sides.

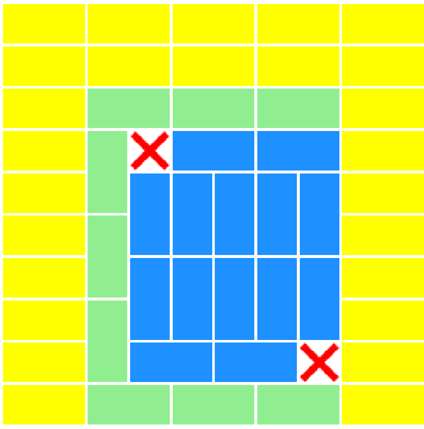


fig. 10 – Extensive pavement of a type-C surface

If $\alpha \neq (1,1)$ and $\beta \neq (n,n)$, then S takes the form of a rectangular ring. We create S' from S by removing 2×2 areas having the top-left coordinates of the form $(2i, 2j), i, j = \overline{1,k}, 2k = n$. These squares can be obviously paved with dominos. Depending on the parity of i_P, j_P, i_Q, j_Q, h, w the surface S' is whether the empty set, a *type-L* or *type-C*[2] structure, or a bordure. Using previous considerations or *Lemma 2*, depending on the case, we obtain that S' can be completely covered and hence S can be completely covered, which results in G being a valid grid.

[2]A type-C structure takes the form of the indicated letter and appears, for example, when i_P and i_Q are both odd, and j_P, j_Q have different parities. The pavement of such *incomplete borders* is made following the indication in the proof of *Lemma 2*, when the rectangle's height is off, ignoring the dominos on row $h+1$.

Although it proves its practicability in the case of even grids with a pair of obstacles, this pavement method could become difficult or even inefficient when talking about many pairs of obstacles. Since the pavement using injection and rotation-segmentation is easier to understand and perform, we will focus on it and try to find a generalization.

3.3.4. Injection in even grids with many pairs of obstacles

Consider the following grid:

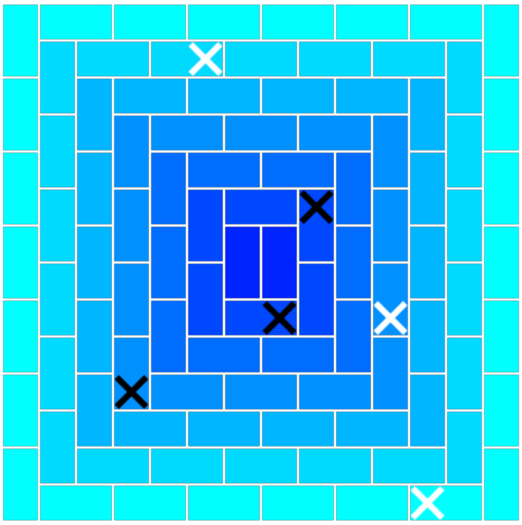
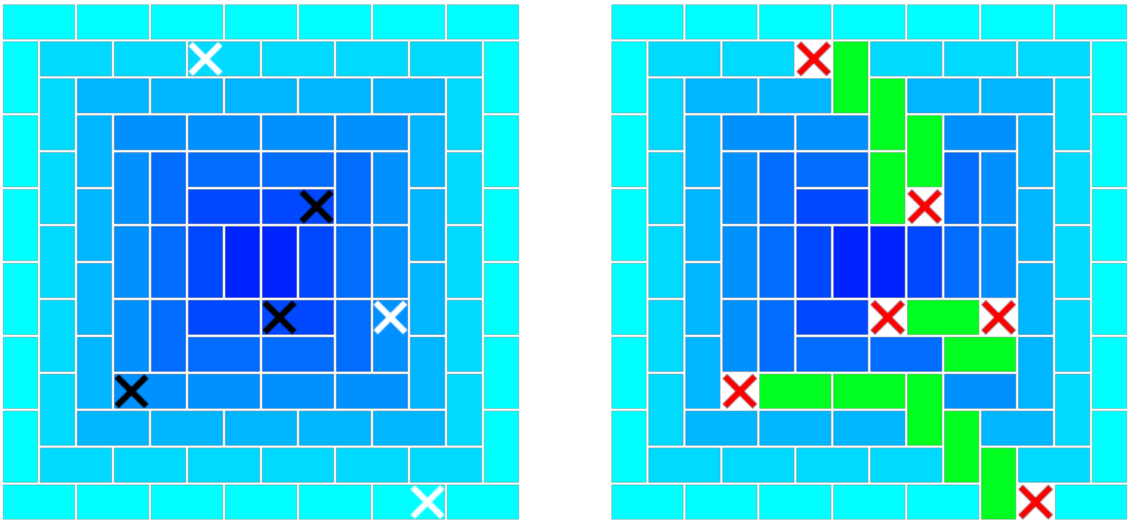


fig. 11 – Type-14 grid with 6 obstacles

We try to „connect" each type-A obstacle with a type-N using dominos, using frame rotation-segmentation and injection. The result is shown below:



a – after rotating frames 2,4,5,6,7 *b – after connecting obstacles via injection*

fig. 12

We ask ourselves whether *for any number of pairs of obstacles and any position they occupy on a grid G, there exist a domino configuration created by rotating the frames of the „zero" configuration, such that we can obtain a complete pavement by obstacles injection into inferior frames?*

We specify that *injection and segmentation operations lock the frame they act upon*. For this reason, this approach may meet limitations in case we try to cover a valid grid starting from a previous configuration, with a lower number of obstacles.

Consider the following example:

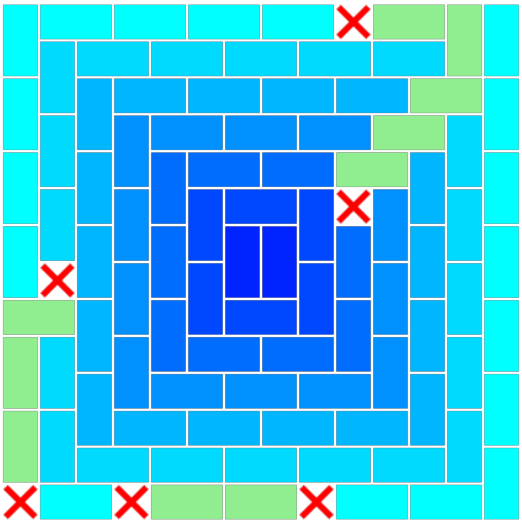
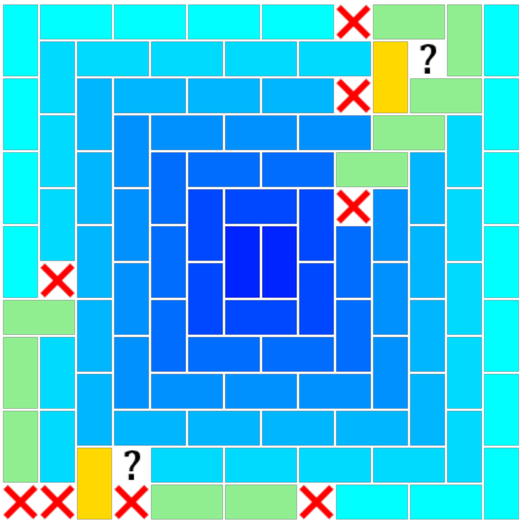


fig. 13 – Valid grid with 6 obstacles

Adding a type-N obstacle at position $(3,10)$ and a type-A obstacle at $(14,2)$, we observe that injection of the added type-N obstacles constraints us to perform injection into the superior frame, isolating the cell $(2,12)$ from the rest of the grid. Obviously, there is not a complete pavement in this case (fig. 14). Frame 7 rotation resolves this inconvenience.



$P = 4$

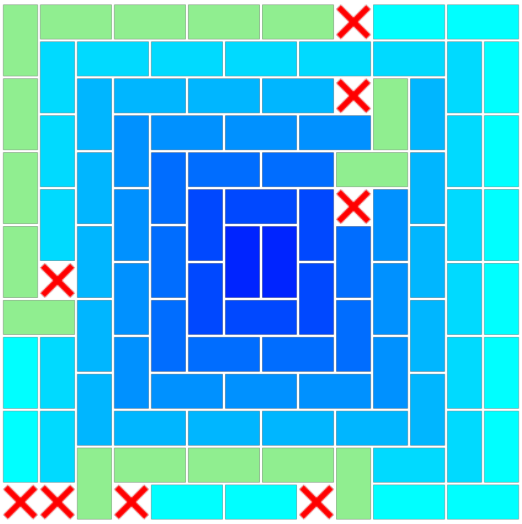
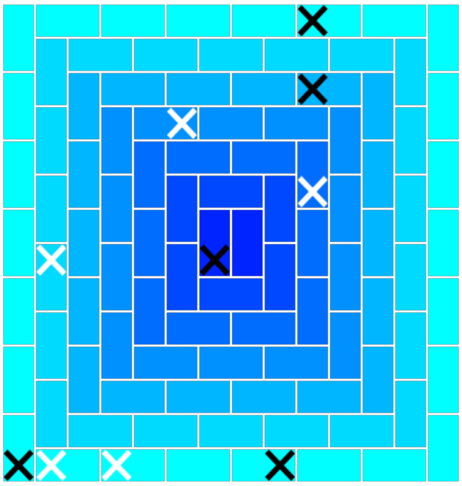
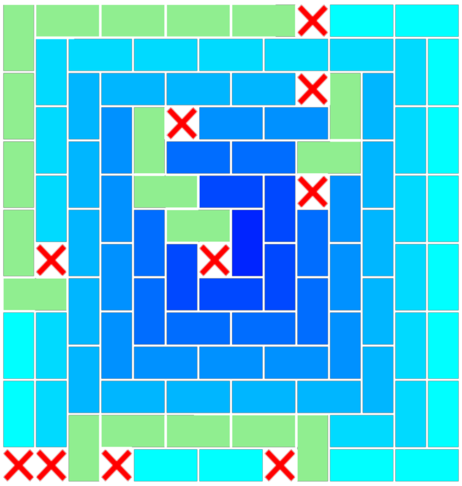


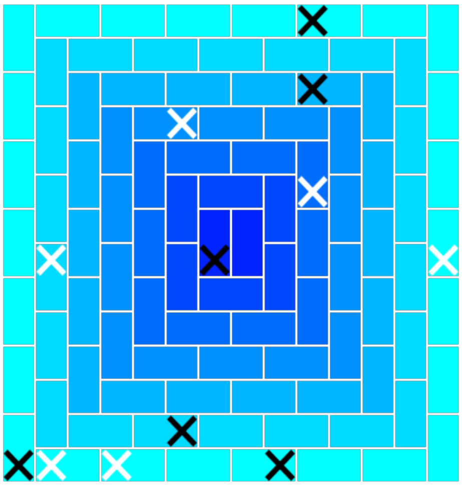
fig. 14 – Grid with 8 obstacles *fig. 15 – Complete pavement of this grid*

Extending the process, we can find complete coverings of grids with many pairs (P) of obstacles (fig. 16).

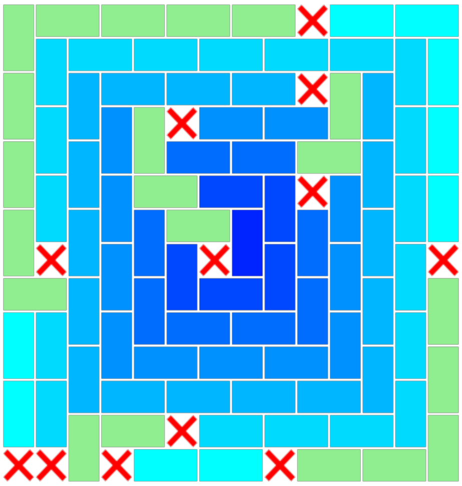


$P = 5$





$P = 6$



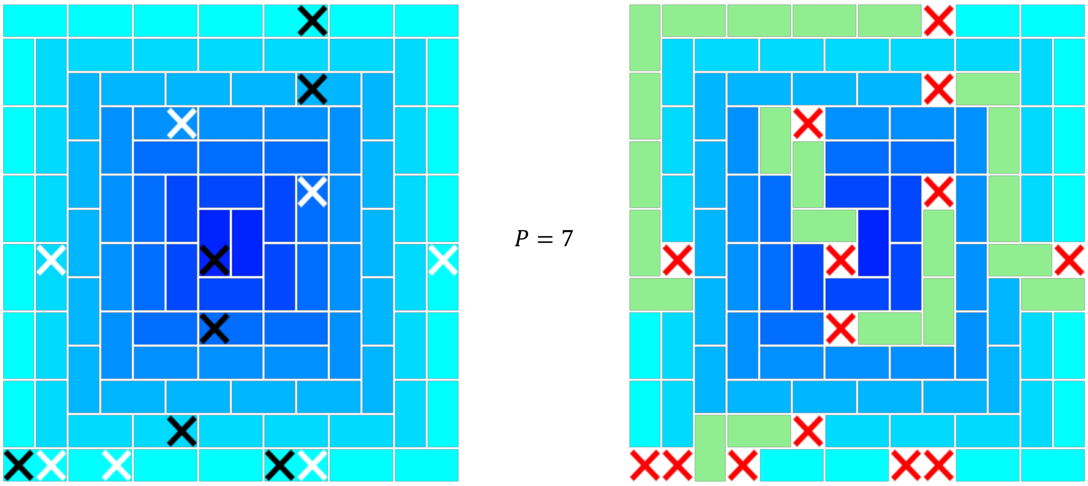


fig. 16 – Example of solving the fig.14 grid, but with 10, 12, respectively 14 obstacles

3.4. Rewriting the validation criteria using bipartite graphs

3.5.1. The grid's corresponding graph

In the following, for any $n \in \mathbb{N}^\star$, we note $[n] = \{1,2, \dots, n\}$.
Ley $Y \subseteq \mathbb{N}^\star \times \mathbb{N}^\star$ be a set of cells coordinates. Over $\mathbb{N}^\star \times \mathbb{N}^\star$ we define the order relation $\overset{n}{<}$, where n is a positive non-zero integer, between two points $P, Q \in \mathbb{N}^\star \times \mathbb{N}^\star$, in the following way:
$P \overset{n}{<} Q \Leftrightarrow x_P * (n+1) + y_P < x_Q * (n+1) + y_Q$.
We update the definition of a grid by calling it a mathematical object described by a non-zero natural number n, called the grid's length, and a set Y of obstacles. We note the grid $G = (n, Y) \in \mathcal{G}_n$ and represents the set $[n]^2 \setminus Y$.
We consider a grid $G = (n, Y)$. We call $A(G) = \{(i,j) \in G | i + j \; odd\}$ the set of white cells. Similarly, $N(G) = G \setminus A(G) = \{(i,j) \in G | i + j \; even\}$ is the set of black cells of the grid G. For ease, we write $N(G) = \{N_1, N_2, N_3, \dots\}$ and $A(G) = \{A_1, A_2, A_3, \dots\}$. The sets are ordered using the relation $\overset{n}{<}$ (e.g. $i < j \Leftrightarrow A_i \overset{n}{<} A_j$).

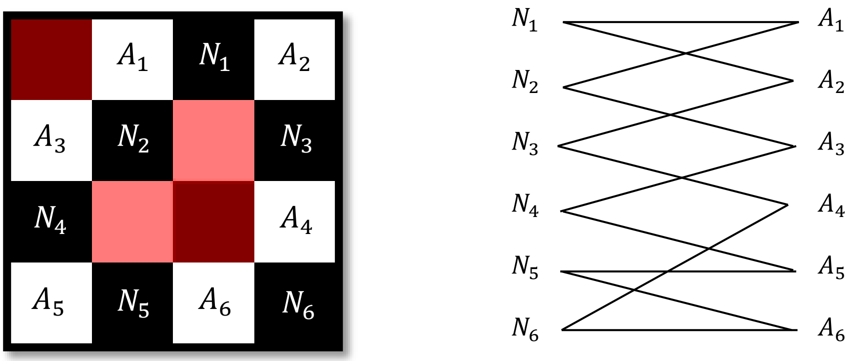


fig. 17 – Example of grid and its corresponding graph

Now let $\Gamma(G) = (A(G), N(G), E)$ be an unoriented bipartite graph from $A(G)$ to $N(G)$, where $E = \{(i,j) \in [n]^2 | A_i, N_j \; are \; adjacent\}$. $\Gamma(G)$ is the grid G's corresponding graph. An edge (X_a, X_n) in

graph encodes the fact that a domino can be placed on the grid covering the cells X_a and X_n.
Observation: There is an obvious bijective correspondence between the regions (sets of adjacent cells) $g \subset G$ bounded by grid G's sides and/or some obstacles, and the conex components $\gamma \subset \Gamma(G)$ in the corresponding graph $\Gamma(G)$.

3.5.2. A valid grid's corresponding graph

Let $G \in \mathcal{G}_n$ and $\gamma_i \subset \Gamma(G), i = \overline{1,k}, k \in \mathbb{N}^\star$ the graph's conex components. Pavement of the whole grid G implies independent pavement of each one of its regions, g_i, which means that each regions can be seen as a separate grid, only surrounded by obstacles: $g_i = (n, [n]^2 \setminus g_i)$. (fig. 18)
Threfore, it is sufficient to study the pavement of a single region. In this regard, let $g \in \mathcal{G}_n$ be a grid with a single region and $\gamma = \Gamma(g)$ its (conex) corresponding graph. The validation condition found in chapter

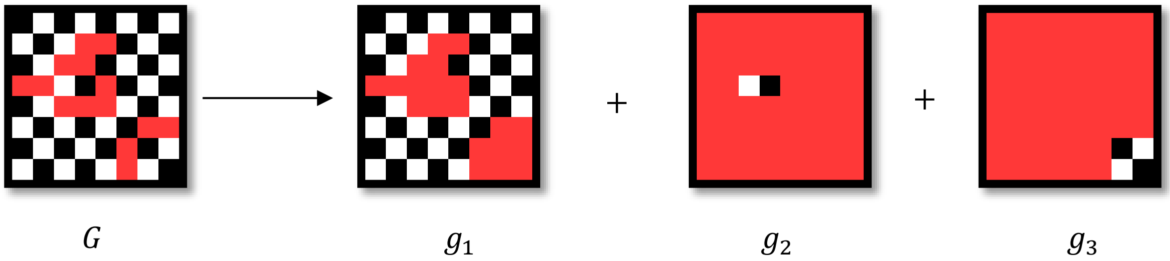


fig. 18 – Example of decomposing a grid in regions of adjacent cells

2 now becomes $|N(g)| = |A(g)|$ (number of free white cells equals the number of free black cells – we call it an **echichrome grid**). We will try to remove edges from γ such that **the grade of each node is equal to 1**. In this case, every edge represents a domino covering the grid's two adjacent cells (fig. 17). We look for a validity condition which must be imposed to the edges of γ in order to make this possible.
To start with, let's observe that $1 \leq \deg(x) \leq 4, (\forall)\, x \in N(g) \cup A(g)$ (except for the non-valid

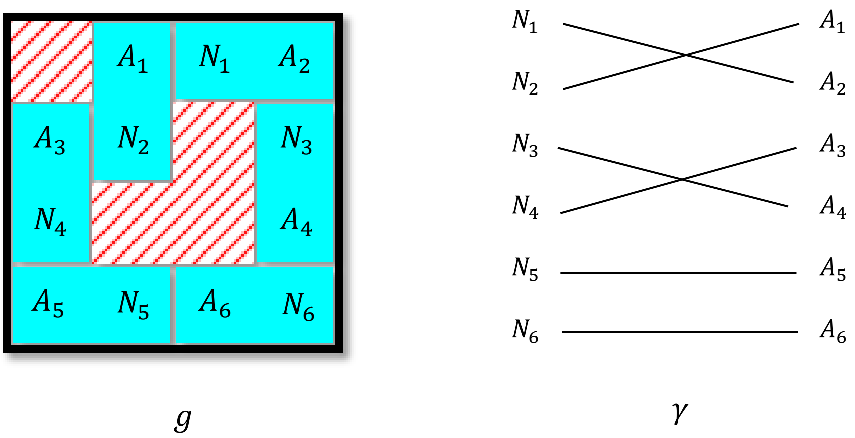


fig. 19 – Solution of the grid at fig. 17

type-1 grid, each cell has between one and four neighbors). Moreover, two cells of the same color must have at most two common neighbors.

3.5.2.1 Lemma. Let $g = (n, Y)$ be a region-grid and x, y, z nodes in the graph $\gamma = \Gamma(g)$. If there exist edges (x, y), (x, z), and $\deg y = \deg z = 1$, then the grid g is not valid.

Proof: Suppose the contrary holds, so that grid g is valid. We try to build $A_\# - N_\#$ connections in bipartite graph γ. Suppose that $x = N_0 \in N(G)$. Then $y, z \in A(G), y = A_0, z = A_1$. The other case is treated analogously. The grid is valid, so after removing the edges from γ we obtain a new graph γ_0 in which there exists an edge from N_0 to a neighbor A_x. We build $g' = (n, Y \cup \{N_0, A_x\})$ a grid similar to g, but the domino N_0A_x was replaced by obstacles. Because there is no change in the other dominos, results that the grid g' is valid, and $\gamma' = \Gamma(g') = (N(g) \setminus \{N_0\}, A(g) \setminus \{A_x\}, E')$ is a subgraph of γ, without the nodes corresponding to the cells covered by the domino N_0A_x. If $A_x \notin \{A_0, A_1\}$, then $\deg A_0 = \deg A_1 = 0$. In other words, cells A_0 and A_1 are isolated, so the grid g' contains regions of area 1, which means it is non-valid, contradiction. If $A_x = A_0$, then $\deg A_1 = 0$, which results in the same contradiction. Similarly for $A_x = A_1$. Consequently, grid g is not valid.

This lemma warrants the non-validity of grids containing type-„T" and type-„L" terminals (a black cell surrounded by three or four white cells, or the opposite, at least two of them having a single neighbor), even it is echichrome. (fig. 20, 21)

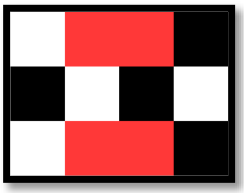


fig. 20– Type „T" terminals

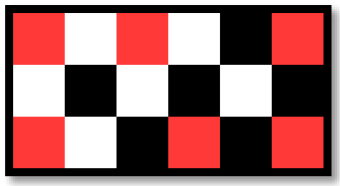


fig. 21 – Type „L" terminals (top-left, bottom-right)

3.5.3. Generating solution using Branch and Bound method

We can create a state tree of all possible pavements, in the way described below:

- root has depth 0
- children of root have depth 1, and children of a node with depth i have depth $i + 1$.
- each depth level, except for 0, represents the id of a white cell
- each node with depth i is assigned the id of a black cell, adjacent to A_i

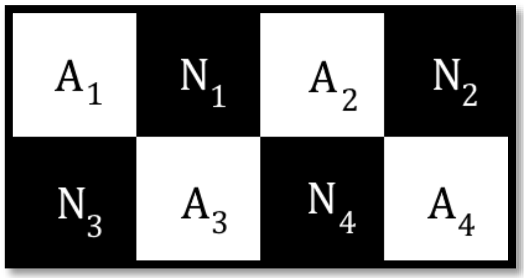


fig. 22 – A 4x2 grid

- the meaning of the node N_j at depth A_i is: *„there exists a domino covering the cells N_j, A_i"*
- All nodes which are part of an elementary chain connecting the root to one of the leaves have to be distinct.
- In figure 21, meaning of the chain $\dotplus\ (root) - N_1 - N_2 - N_4$ is *„we place dominos over the cells A_1N_1, A_2N_2, A_3N_4"*. Note that non-adjacent cells N_3 and A_4 remain uncovered, therefore $\dotplus - N_1 - N_2 - N_4$ is an incomplete pavement.
- In a valid grid, a complete pavement is encoded in the state tree by a chain of length equal to the number of all white cells in the grid.

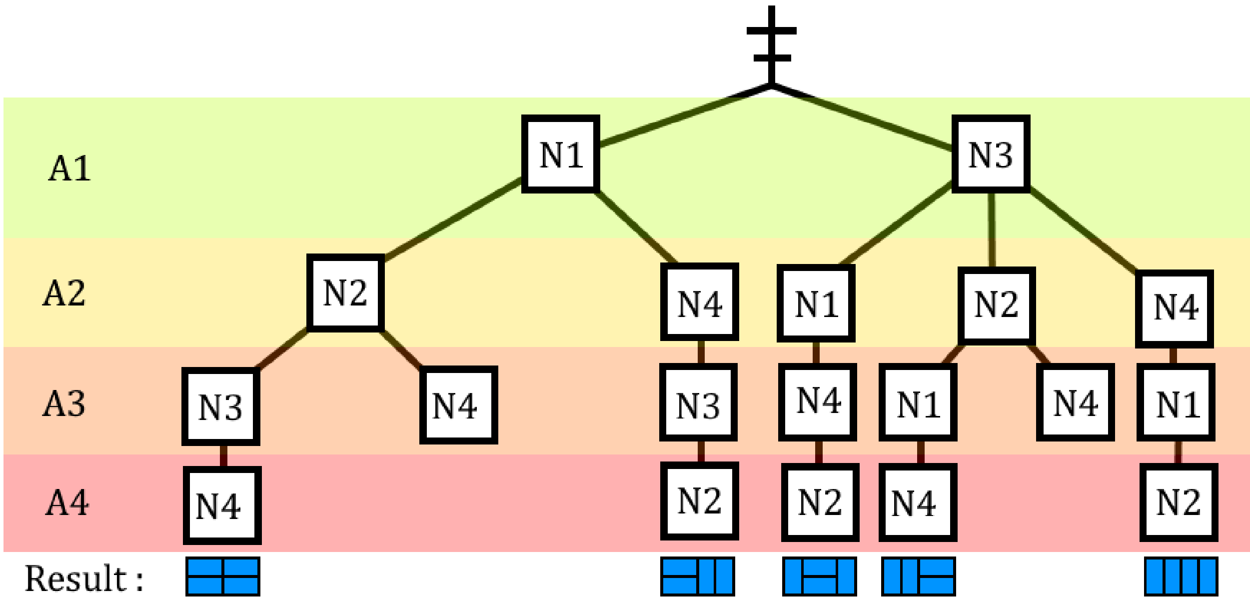


fig. 23 – Example of state tree for the grid in fig. 22

A depth-first search of such a tree is a brute-force method which offers all the grid's possible pavements (both valid and non-valid).

Possible optimizations:

- check the grid's validity using the criteria
- White cells with low number of neighbors get a low id (this way, terminal cells would be visited first hoping to reduce the tree's complexity)
- If in a chain there exists the cells N_i, N_j, N_k, N_l, all neighbors of A_x, having the depth lower than x, then that particular chain can be ommited (no matter how far we continue the pavement, we cannot cover the cell A_x with a domino)
- the same holds for a black cells surrounded by covered white positions
- Avoiding type-L and type-T terminals
- After obtaining a *promising* partial solution, we can initiate a *local search* strategy to finish the pavement.
- Memory optimizations (use the fact that each node has at most 4 children, don't generate the nodes we are certain we won't visit)

Complexity: $O\left(4^{\frac{n^2}{2}}\right) = O\left(2^{n^2}\right)$

3.5.4. Linear algebra approach of the problem

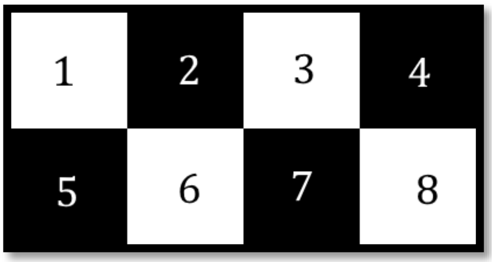


fig. 24 – 4x2 grid, where cells are numbered from 1 to 8

For simplicity, we will get rid of the notations N_x, A_x of the squared and start counting them from 1. For each two adjacent cells i, j, we define:

$$d_{ij} = \begin{cases} 1, & if\ a\ domino\ covers\ cells\ i, j \\ 0, & in\ the\ other\ case \end{cases}$$

(in the bipartite graph, $d_{ij} = 1$ *is equivalent to „there exists an edge from i to j”).*

We admit that $d_{ij} = d_{ji}\ (\forall) i, j\ neighbors$. Hence, for any pavement of the grid, there is a bijective correspondence between the domino configuration and the set $\mathcal{D} = \{d_{ij} | d_{ij} = 1, i, j\ neighbor\ cells\}$. A valid pavement satisfies the following conditions:

- a cell cannot be covered by two or more domino pieces:

$$\sum_{i\ neighbor\ of\ k} d_{ki} = 1\ (\forall) k = \overline{1, 2C_A}, considering\ C_A = C_N \qquad (1.i)$$

- number of all dominos must be equal to half of the number of all cells in the grid:

$$\sum_{i=1}^{2C_A} \sum_{\substack{k=i \\ k\ neighbor\ of\ i \\ k<i}} d_{ik} = C_A \qquad (2)$$

We obtain a linear system with $2C_A + 1$ equations and $|\mathcal{D}|$ variables, whose solution defines a complete pavement of the grid in question.

Consider the same 4×2 grid, whose cells are labelled as can be seen in figure 24. The system associated with this grid is:

$$(S) \begin{cases} d_{12} + d_{15} = 1 \\ d_{12} + d_{23} + d_{26} = 1 \\ d_{23} + d_{34} + d_{37} = 1 \\ d_{34} + d_{48} = 1 \\ d_{15} + d_{56} = 1, \\ d_{26} + d_{56} + d_{67} = 1 \\ d_{37} + d_{67} + d_{78} = 1 \\ d_{48} + d_{78} = 1 \\ d_{12} + d_{23} + d_{34} + d_{15} + d_{26} + d_{37} + d_{48} + d_{56} + d_{67} + d_{78} = 4 \end{cases}$$

where $d_{12}, d_{23}, d_{34}, d_{15}, d_{26}, d_{37}, d_{48}, d_{56}, d_{67}, d_{78} \in \{0,1\}$.

Or, in matrix form, the sistem becomes:

$$\begin{pmatrix} 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} \begin{pmatrix} d_{11} \\ d_{23} \\ d_{34} \\ d_{15} \\ d_{26} \\ d_{37} \\ d_{48} \\ d_{56} \\ d_{67} \\ d_{78} \end{pmatrix} = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 4 \end{pmatrix}.$$

After performing Gaussian elimination, the matrix is in the following reduced row echelon form:

$$\begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} d_{11} \\ d_{23} \\ d_{34} \\ d_{15} \\ d_{26} \\ d_{37} \\ d_{48} \\ d_{56} \\ d_{67} \\ d_{78} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \\ 1 \\ 1 \\ 1 \\ 0 \\ 0 \end{pmatrix}.$$

As the matrix rank is 7, we find the solutions in terms of parameters d_{56}, d_{67}, d_{78} (the other variables are constrained variables):

$$(S)\begin{cases} d_{11} = d_{56} \\ d_{23} = d_{67} \\ d_{34} = d_{78} \\ d_{15} = 1 - d_{56} \\ d_{26} = 1 - d_{56} - d_{67} \\ d_{37} = 1 - d_{67} - d_{78} \\ d_{48} = 1 - d_{78} \end{cases}.$$

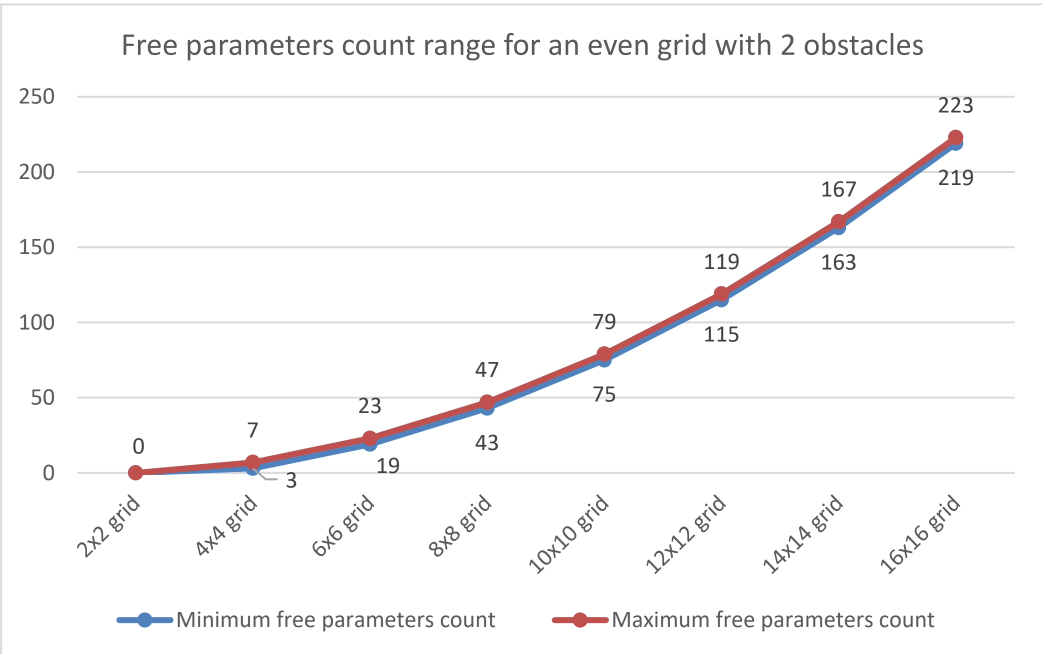
Keepind in mind that the variables are integers ranging from 0 to 1, we find additional conditions for the parameters d_{56}, d_{67}, d_{78}:

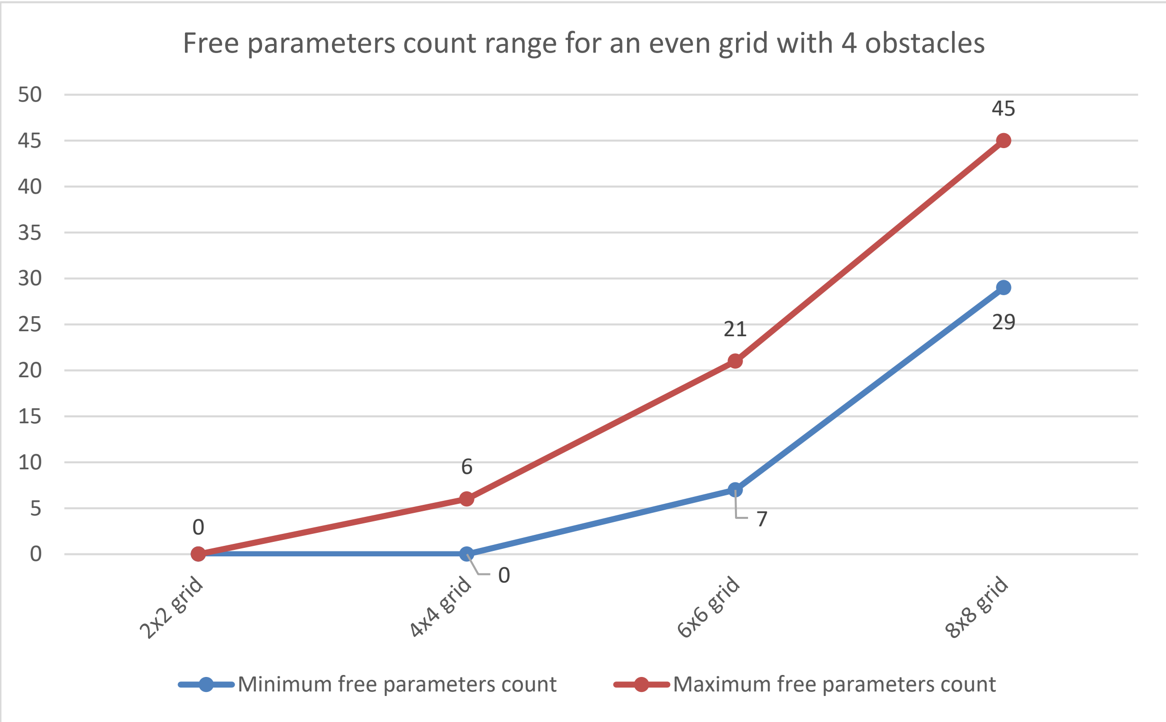
$$(C)\begin{cases} d_{56} + d_{67} \in \{0,1\} \\ d_{67} + d_{78} \in \{0,1\} \end{cases}.$$

The conditions (C) impose $(d_{56}, d_{67}, d_{78}) \in \{(0,0,0), (0,0,1), (0,1,0), (1,0,0), (1,0,1)\}$. Returning to (S), we find the exact 5 solutions we discovered by expanding the state tree of all possible pavements.

This method's difficulty consists in solving the conditional system (C), because the system's number of parameters grows with the number of grid's free cells and decreases with the density of the contained obstacles.

Complexity: $O(n^3)$ for Gaussian ellimination and $O\left(2^{(number\ of\ free\ vars)}\right)$ for computing solutions





Although the complexity of this algorithm seems to be theoretically worse than the branch-and-bound one (the number of constrained variables is always less than $n^2 + 1$ while the total maximum number of variables can reaches $2(n^2 - n)$ in a free grid), in practice it solves a large set of grids in reasonable time, because there are solutions in which a good deal of the free variables are can be 0, meaning that they can be found in faster computation time. There are also some cases, like the grid shown in figure 25, where the yellow regions are the only areas that can be covered in more than one way, resulting in 64 different complete pavements (solutions) of the same grid out of 2^{23} total configurations. This means the chance of finding a complete pavement is 0.000007%! A backtracking algorithm performed on the conditional system (C) found all of the possible solutions in the last quarter of its whole execution time.

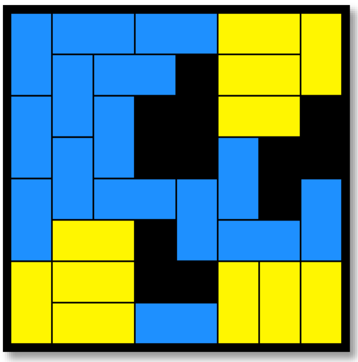
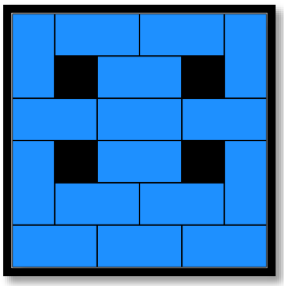
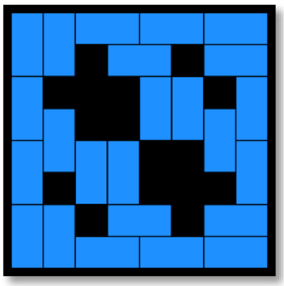


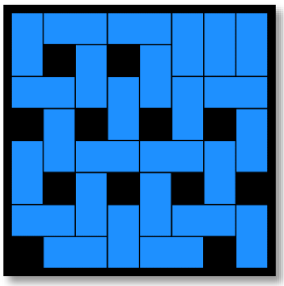
fig. 25



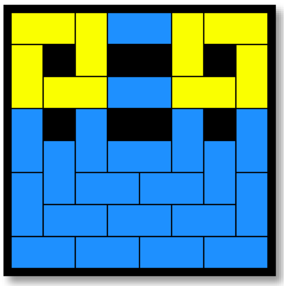
28 total solutions
0.003%



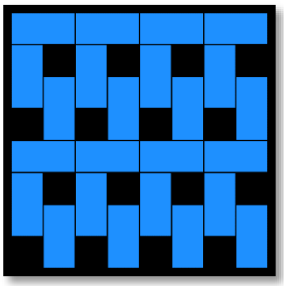
64 total solutions
0.031%



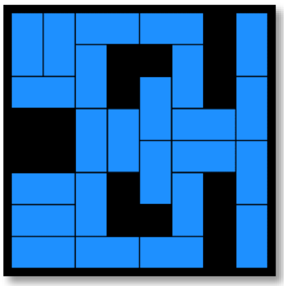
423 total solutions
0.0016%



4 total solutions
0.00000003%



1 solution
0.15%



256 total solutions
0.004%

fig. 26
Examples of complete pavements and chance to find solution for certain valid grids

Hypothetsis: The following result was issued experimentally by analyzing a number of patterns, no proof is provided yet:

> *A grid is not valid if the associated system's matrix contains a value different from* -1, 0 *or* 1 *after the Gaussian elimination in reduced row echelon form.*

4. Conclusion

The domino covering problem is simple-looking yet a complex and beautiful subject, with a plenitude of ways to approach it. Having a first visual representation of the problem provided valuable information and eventually proved mandatory in defining the mathematical model using bipartite graphs, which made designing algorithms to solve this problem possible. However, in terms of difficulty, it is worth to note that the domino covering problem fades out relative to other covering problems with many variants and constraints, which could imply rectangular pieces of various sizes, irregularly shaped pieces, or even using more than one type of shape in our pavement. Not to mention the three dimensional variant of this problem, in which we want to fill a certain space with $2 \times 1 \times 1$ blocks. This particular problem can be trated similar to the current subject whether by finding a match in a bipartite graph or by solving a linear system.