MATH 285N: Tiling Problems

Instructor: Igor Pak

Scribe: Glenn Sun [1]

University of California, Los Angeles — Fall 2022

[1] These notes may contain errors. If you notice an error, please report it to me at glennsun@ucla.edu.

Contents

I Methods 3

1 Domino tilings 4

2 Conway's tiling groups 11

3 Coloring arguments 17

4 Tiling by T-tetraminoes 20

5 Tiling of rectangles 24

II Complexity 28

6 An undecidable problem 29

7 NP-complete tilings 33

8 #P-complete tilings 39

9 Sequences of tiling counts 42

III Related problems 48

10 Partitions and rim hooks 49

11 Matchings 55

Part I

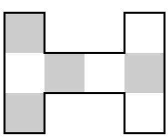
Methods

1 Domino tilings

A tile is a simply connected region $t \subset \mathbb{Z}^2$. Tiling problems ask variations of the following question: Given a finite region $\Gamma \subset \mathbb{Z}^2$ and a set of tiles T, is there a way to arrange translations of tiles from T, so that no two copies overlap and their union is exactly Γ? For example, after fixing a set of tiles T, can we come up with an efficient algorithm to determine if Γ is tileable by T? If the question is too hard, we can make it easier by imposing some restrictions on Γ, such as being simply connected or being a rectangle. On the other hand, we can also ask harder questions, like counting the number of distinct tilings of Γ by T.

In Part I, we will see many different methods to answer these questions positively. (Negative results will appear in Part II.) Let us begin by considering one of the simplest cases. Set $T = \{\square\!\square, \boxminus\}$. When is a region $\Gamma \subset \mathbb{Z}^2$ tileable by T? (When drawing tiles, each square represents one integer point.) This is the problem of domino tilings.

Introductory discrete math courses often cover the following exercise: given an 8×8 chessboard with two opposite corners removed, prove that the remaining board cannot be covered by dominoes. The solution is to color the chessboard in alternating black and white squares. Then opposite corners are of the same color, whereas each domino covers both a black and a white square, so a tiling is not possible. This coloring criterion is clearly necessary in general, but it is not sufficient. The following shape has no domino tiling, yet has an equal number of black and white squares.



A simple general algorithm to determine if Γ can be tiled by $\{\square\!\square, \boxminus\}$ involves a reduction to the perfect matching problem. Recall that the perfect matching problem asks: Given a graph $G = (V, E)$, is there a subset of edges $M \subset E$ such that every vertex appears in exactly one edge of M? The fastest known deterministic algorithm for perfect matching is due to Micali and Vazirani [MV80], and runs in $O(\sqrt{|V|} \cdot |E|)$ time.

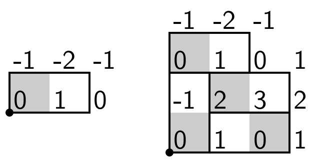
PROPOSITION 1.1. There is a polynomial time algorithm to decide tileability of Γ by $\{\square\!\square, \boxminus\}$. ⌟

Proof. Convert the region into a graph by taking $V = \Gamma$ and creating an edge between every two points with distance 1 from each other. Then a perfect matching in the graph is exactly a domino tiling (every edge in the matching is a domino). $\square$

Running the Micali–Vazirani algorithm, we get a runtime of $O(n^{1.5})$, where $n = |\Gamma|$. One might naturally wonder if this is best possible. For simply connected regions Γ, Thurston gave an algorithm to show that we can do better.

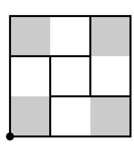
THEOREM 1.2 ([Thu90]). There is an $O(n \log n)$ time algorithm to decide tileability of simply connected Γ by $\{\square\!\square, \boxminus\}$. ⌟

Proof. Recalling that squares in our drawing denote integer points, denote $\Gamma^* = \Gamma + (\pm 0.5, \pm 0.5)$, the set of corners of the squares. Fix an origin in Γ^*. Given a tiling τ, define a function $h_\tau : \Gamma^* \to \mathbb{Z}$, called the height function as follows: h_τ is 0 at the origin, and travelling counterclockwise along tiling lines, add 1 around black squares and subtract 1 around white squares. Here are two examples. (The origin is the dot, and the h_τ is written to the top-left of each point.)



LEMMA 1.3. The function h_τ is well-defined. That is, computing h_τ using any path around the dominoes produces the same values. ⌟

Proof. First note that the region must be simply connected. For instance, if one tries to compute h_τ on the following tiling of an annulus-shaped region, one finds by tracing the outer perimeter that h_τ can be any multiple of 4 at the origin.



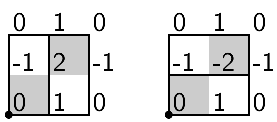
We prove by induction on the number of tiles. The base case of a single domino is clear by the calculation in the first example. For the inductive step, find a domino t whose removal keeps the region simply connected, then remove it and compute h_τ on the smaller region. (Take the existence of such a domino for granted, we'll come back to it in the next section.)

By our choice of domino, t may only touch $\Gamma \setminus t$ on a continuous path γ (i.e. not two segments). The computation of h_τ on points in γ is consistent with placing t back because clockwise around a black square (+1) is counterclockwise around the white square on the other side $(-(-1))$. The computation of h_τ on the remaining points of t is consistent with the two endpoints of γ because any two paths around a single domino produce the same values, by the base case. $\square$

One important fact to notice that is that by traversing the boundary $\partial\Gamma$, we find that h_τ has the same value on $\partial\Gamma$ for all τ. In fact, these values on the boundary can be found even if no tiling of Γ exists. When we only care about the boundary, we will hence write h instead of h_τ.

LEMMA 1.4. Assuming Γ has a tiling, there exists a height function $h_{\min}$ such that for all tilings τ, $h_{\min} \leq h_\tau$ pointwise. ⌟

Proof. Observe that by applying the following local transformation, called a flip, the value of h_τ at the center vertex decreases by 4.



Starting with any height function, apply such flips until no longer possible, implying a local minimum h^*. We will show that $h^* = h_{\min}$. It suffices to show that h^* can be computed using only the values of h on $\partial\Gamma$, because it implies that h^* is independent of the tiling one starts with, and hence has values strictly less than all other tilings.

Observe that h^* attains its maximum on $\partial\Gamma$. This is because all height functions increase in some direction at domino corners, and interior vertices that are not corners have the above flip applied, after which they are less than their neighbors. Therefore, we can compute h^* on the interior as follows: find a vertex where h^* is maximum. The vertex must be on the side of a domino, because corners are not maximums. This identifies a domino piece that must exist in the tiling of h^*, which allows the computation of h^* around that piece. Because it was a boundary piece, removing it leaves one or two simply connected regions, allowing the process to repeat. $\square$

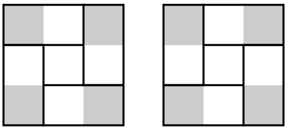
To conclude the proof, recall that the height function along $\partial\Gamma$ can be computed without knowing any tiling, even if there does not exist a tiling. Then, attempt to construct $h_{\min}$ using the tile-removing procedure described in the above lemma. If a tiling exists, then $h_{\min}$ will be found successfully. If no tiling exists, then because the above procedure finds a tiling by the end, an error can be detected.

Where $n = |\Gamma|$, this algorithm computes $h_{\min}$ on $O(n)$ points. Note that $h_{\min}$ may actually have values that are quite large (up to n) and hence this requires $O(n \log n)$ time. One also needs to find vertices where $h_{\min}$ attains maximum value about $O(n)$ times, but this can be done once and dynamically updated using a heap data structure, which again allows everything to be done in $O(n \log n)$ total time. $\square$

The proof of the above theorem gives the following corollary.

Corollary 1.5. Any two domino tilings of a simply connected region are connected by flips. ⌟

This is just because all domino tilings are connected to $h_{\min}$ by flips, and this is a transitive relation. Again, note that it is very important to be simply connected here, as the following example shows.

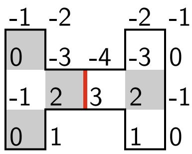


Flip connectivity is an important topic that we will revisit many times. For dominoes, there are many results that explore this even further. Let Γ be a square of side length $2k$ and H its graph of flip connectivity, whose vertices are domino tilings and two tilings are connected by an edge if and only if they differ by a flip. Then the diameter of H is $O(k^3)$. To sketch the argument, we can explicitly compute $h_{\min}$ and $h_{\max}$ for squares. For each, the gap between the boundary and the center is $O(k)$. A flip decreases the center value by 4 (a constant), and every tiling has $2k^2$ dominoes, so we should expect to need $O(k^3)$ flips to go from $h_{\max}$ to $h_{\min}$. A full proof can be found in [PZ17].

With Thurston's near-linear time algorithm, one might think that there are not many more improvements to be made. However, note that a simply connected region may be specified by listing only its boundary, which may be up to quadratically smaller. Thurston's algorithm is only linear in the area, and may be quadratic in the boundary length. Tassy began working in this direction by noticing the following phenomenon.

Theorem 1.6 ([Tas14]). Let $\alpha(x, y) = 2|x - y|_\infty + \delta(x, y)$, where $\delta(x, y)$ is a small correction factor ($|\delta(x, y)| \le 1$) whose formula is a bit complicated and irrelevant for us. (See [PST16] for a precise characterization.) Then a simply connected region Γ is tileable by dominoes if and only if for all $x, y \in \partial\Gamma$, we have $h(x) - h(y) \le \alpha(x, y)$. ⌟

As an example, consider our original example of a region not tileable by dominoes. The height function around the boundary is computed below.



Across the red line, $h(x) - h(y) = 7$, whereas $2|x - y|_\infty = 2$, so there is no way we can have $h(x) - h(y) \le \alpha(x, y)$. Hence Γ is not tileable by dominoes.

Note that this theorem does not immediately imply an improvement on Thurston's algorithm. By checking every pair of points on the perimeter, this theorem implies an algorithm to determine domino tileability in $O^*(p^2)$ time, where $p = |\partial\Gamma|$ is the perimeter and * hides logarithmic factors. However, further work in [PST16] (the T is Tassy) reduced this to $O^*(p)$ time by finding redundancies in checking every pair of boundary vertices: at a high-level, it suffices to add a few points on the interior in a Steiner tree-like way to speed things up.

The theorem is a bit tedious and not very enlightening to prove. However the intuition for the theorem follows from a classical result in geometry, known as Kirszbraun's theorem.

Theorem 1.7 ([Kir34]). Let $A = \{a_1, \ldots, a_n\}$ and $B = \{b_1, \ldots, b_n\}$ be points in $\mathbb{R}^d$ such that $|a_i - a_j| \ge |b_i - b_j|$ for all i, j. Then there exists a piecewise linear isometry $\varphi : \mathbb{R}^d \to \mathbb{R}^d$ such that $\varphi(a_i) = b_i$ for all i. ⌟

A piecewise linear isometry is a continuous function that is an isometry on each piece of a partition on $\mathbb{R}^d$ into a locally finite triangulation. A generic example is the act of crumpling a piece of paper and flattening it back down. The creases form a triangulation and lengths are preserved in each piece, while the entire motion is continuous. Note that piecewise linear isometries are always 1-Lipschitz functions.

Before proving the theorem, let us first connect this back to Tassy's tiling criterion. Think about A as the points on the perimeter, so that the distances $|a_i - a_j|$ are similar to $\alpha(x, y)$ (basically L_∞ distance). Then if $|b_i - b_j|$ corresponds to $h(x) - h(y)$, the existence of a piecewise linear isometry on the whole space that correctly maps these points is almost like the existence of a height function that extends to the interior, which exists if and only if Γ is tileable by dominoes.

Proof of Theorem 1.7. We follow a more recent constructive proof of the theorem by [AT08]. We also focus on the case of $d = 2$ for simplicity. We induct on n, and the base case of a single point is trivial.

For the inductive step, first note that by a translation, we can assume $a_n = b_n$ without loss of generality. Let ψ satisfy $\psi(a_i) = b_i$ for all $1 \leq i \leq n-1$. Let $\Omega = \{x \in \mathbb{R}^2 \mid |a_n - x| < |a_n - \psi(x)|\}$, the region of points that ψ moves away from a_n.

We make a few observations about Ω. First, if $a_n \notin \Omega$, that means $\psi(a_n) = a_n = b_n$ and we are done. So we can safely assume $a_n \in \Omega$. Additionally, $a_i \notin \Omega$ for $1 \leq i \leq n-1$ because such points move towards, not away from a_n by the hypotheses of the theorem. Lastly, Ω is an open set with piecewise linear boundary, since ψ is an isometry on each part of a locally finite triangulation.

LEMMA 1.8. Ω is a star-shaped domain with center a_n. ⌟

Proof. Let y be on the line segment $[a_n, x]$ for any $x \in \Omega$. We are to show $y \in \Omega$, meaning $|a_n - y| \leq |a_n - \psi(y)|$. We have

$$\begin{aligned} |a_n - y| &= |a_n - x| - |x - y| && \text{(def. of } y\text{)} \\ &< |a_n - \psi(x)| - |x - y| && (x \in \Omega) \\ &\leq |a_n - \psi(x)| - |\psi(x) - \psi(y)| && (\psi \text{ is 1-Lipschitz}) \\ &\leq |a_n - \psi(y)|. && \text{(triangle ineq.)} \end{aligned}$$

The lemma is proved. □

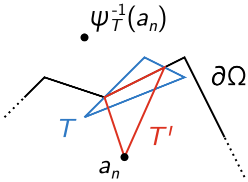
LEMMA 1.9. For every triangle T in the locally finite triangulation on which ψ is defined, denote $\psi_T = \psi|_T$ the isometry of ψ restricted to T. Then $T \cap \partial\Omega \subset \{x \in \mathbb{R}^2 \mid |x - a_n| = |x - \psi_T^{-1}(a_n)|\}$. ⌟

Proof. First note that because isometries are bijections, the point $\psi_T^{-1}(a_n)$ is certainly well-defined (though it may lie outside T). Next, for any $x \in T \cap \partial\Omega$,

$$\begin{aligned} |x - a_n| &= |\psi(x) - a_n| && (x \in \partial\Omega) \\ &= |\psi_T(x) - a_n| && (x \in T) \\ &= |\psi_T(x) - \psi_T(\psi_T^{-1}(x))| \\ &= |x - \psi_T^{-1}(x)|. && (\psi_T \text{ is isometry}) \end{aligned}$$

The lemma is proved. □

To construct the desired function φ, first we let $\varphi(x) = \psi(x)$ for all $x \notin \Omega$. This correctly maps $a_1, \ldots, a_{n-1}$ by the observation above. To define φ on Ω, we traverse $\partial\Omega$ and find every T that intersects it. By the above lemma, $T \cap \partial\Omega$ is a line segment, so let r_T denote the reflection across it. Because Ω is star-shaped, we also have a partition of Ω into triangles T' with bases $T \cap \partial\Omega$ and vertex a_n. On every such T', we define $\varphi(x) = \psi_T(r_T(x))$. The reflection maps a_n to $\psi_T^{-1}(a_n)$ by the above lemma, so $\varphi(a_n) = a_n$ as desired.



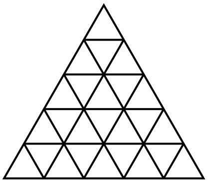
It remains to check that φ is continuous. On $\partial\Omega$, the reflection r_T is an identity map so we are left with ψ, which agrees with φ outside Ω. Inside Ω, the potentially conflicting points are line segments $[a_n, x]$ where x is an endpoint of some segment $T \cap \partial\Omega$. But regardless of T, because $x \in \partial\Omega$, again φ maps x to $\psi(x)$, so the entire segment $[a_n, x]$ is mapped to $[a_n, \psi(x)]$. So φ is continuous.

A detail missed here is the case where Ω is not bounded, so the triangles T' constructed above do not actually cover the entire set Ω. The interested reader may consult the original paper [AT08] to fill in this detail. $\square$

2 Conway's tiling groups

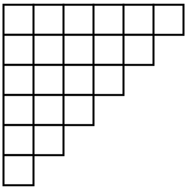
With dominoes sufficiently discussed, it is time to consider other sets of tiles T. In this section, we will discuss technique that generalizes parts of Thurston's algorithm, called Conway's tiling groups.

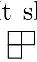
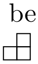
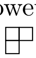
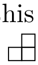
Before introducing the main event, consider another example. Call the following graph Δ_n, with n vertices on each side. (Below pictured is Δ_6.)



We ask the question: for what n does there exist a vertex cover of Δ_n by triangles $\Delta_2 = K_3$? That is, can we choose some triangles (copies of Δ_2) in Δ_n such that every vertex in Δ_n is contained in exactly one triangle? For instance, Δ_3 has no vertex cover by Δ_2. The three vertices of degree 2 each force a Δ_2 in the cover, leading to overlap.

Of course, we are studying tilings of regions in $\mathbb{Z}^2$, so let us translate the problem first. Tiling Δ_n with triangles is equivalent to tiling the following region, call it Γ_n, with $T = \{$, $\}$. (Just tilt your head 45 degrees.)



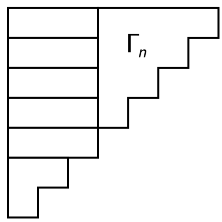
It should be quickly observed that because Γ_n has $\frac{1}{2}n(n+1)$ squares and  and  have 3 each, we must have $3 \mid n(n+1)$, and hence $n \equiv 0, 2 \pmod 3$. However, this is not sufficient, since as mentioned before, Γ_3 has no tiling by  and . Conway and Lagarias prove the following.

THEOREM 2.1 ([CL90]). The region Γ_n is tileable by $T = \{\ ,\ \}$ if and only if $n \equiv 0, 2, 9, 11 \pmod{12}$.

Proof. First, note the following lemma:

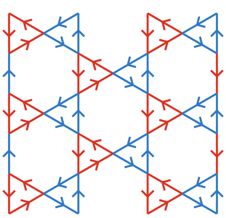
LEMMA 2.2. The region Γ_n can be tiled by $\{\ ,\ ,\ ,\ \}$ if and only if $n \equiv 0, 2 \pmod 3$.

Proof. The $\Longrightarrow$ direction is the same trivial divisiblity argument above. The $\Longleftarrow$ direction is by induction. $\Gamma_2 =$ is trivially tiled. Γ_3 can be tiled by stacking and . For the inductive step, given a tiling for Γ_n, one can tile Γ_{n+3} as in the following picture.



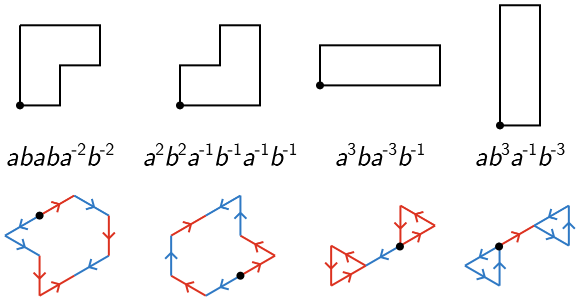
□

Our approach is to define a sort of height function, similar to the proof of domino tilings. Like before, the height function will depend on the tiling τ using tronimoes in T, which exist for all n that we care about by the above lemma, and will assign something to every vertex in Γ_n. First, consider the following infinite graph, denoted R.



For a precise characterization, alluding to our forthcoming discussion of Conway's tiling groups, R is the Cayley graph of the group $\langle a, b \mid a^3 = b^3 = (ab)^3 = 1\rangle$. (That is, the free group generated by a and b under the relations $a^3 = b^3 = (ab)^3 = 1$.) For our purposes now, just note that at every vertex has one each of red in-edge, red out-edge, blue in-edge, and blue out-edge.

Fix a tiling τ of Γ_n by $\{\ ,\ ,\ ,\ \}$, and fix an origin in both Γ_n^* and R. The height function $h_\tau : \Gamma_n^* \to R$ is defined by walking along tiling lines, then having right steps correspond to following red out-edges (a steps), up steps correspond to following blue out-edges (b steps), and left/down steps correspond to following in-edges (a^{-1} or b^{-1} steps). For example, here is how h_τ would map the 4 tronimoes , , , and . (The sequences of a and b are written by walking counterclockwise, but it doesn't matter for the path in R.)



LEMMA 2.3. The height function h_τ is well-defined. That is, any two paths between the same points, following tiles, end at the same vertex in R. ⌟

Proof. A similar induction argument to the corresponding lemma in the dominoes case. In particular, the above four drawings show that the height function is well-defined around single tiles, since each path in R loops back to its original vertex. In the inductive step, we would once again add tiles one by one, noting that we can continue to define h_τ nicely. □

As before, we observe that h_τ restricted to the boundary of Γ_n is the same function for all τ. We denote the image by $h(\partial\Gamma_n)$. The key lemma of our main argument is the following:

LEMMA 2.4. For all tilings τ of Γ_n by T, the number of pieces minus the number of pieces is constant. (Denote this quantity by $\rho(\Gamma_n)$). ⌟

Proof. Place a point z_i inside each hexagon in R. Denote wind(γ, z) to be the winding number of a curve γ around a point z (the number of full counterclockwise rotations γ makes around z). We will show that

$$\rho(\Gamma_n) = \sum_i \mathrm{wind}(h(\partial\Gamma_n), z_i),$$

which is a quantity that does not depend on the tiling.

Note from the previous figure that pieces loop clockwise around one hexagon (winding number -1), pieces loop counterclockwise around one hexagon (winding number 1), and and pieces do not loop around any hexagons (winding number 0). Hence any tiling τ,

$$\rho(\Gamma_n) = \sum_i \sum_{t\in\tau} \mathrm{wind}(h_\tau(\partial t), z_i).$$

Now, we just need to observe that $\partial\Gamma_n$ is exactly the boundaries of the pieces ∂t for $t \in \tau$ joined together, and winding number is additive when joining two cycles, so the result is exactly as claimed. □

To finish the proof, we first find example tilings of Γ_2, Γ_9, Γ_{11}, and Γ_{12} by brute force or computer search. A simple induction argument gives the periodicity with a picture nearly identical to Lemma 2.2.

To show that the remaining n are not possible, do the following simple calculation for each n. The area of Γ_n is $\frac{1}{2}n(n+1)$. Using a tiling of Γ_n by T, find $\rho(\Gamma_n)$. Then if Γ_n has a tiling by p  tiles and q  tiles, we have $p+q = \frac{1}{6}n(n+1)$ and $p-q = \rho(\Gamma_n)$. A computation will find that there will not be integer solutions for p and q. For instance, for $n = 15$, the area is 120 (meaning $p+q = 40$) and $p - q = \rho(\Gamma_n) = -5$, for which there are no integer solutions. □

There were many similarities between this proof and Thurston's algorithm, differing really only at the end. We also note that all the lemmas hold for simply connected Γ in general, and we only use the shape of Γ at the end. The general idea here is known as Conway's tiling group.

Definition 2.5. Let $T = \{t_1, \ldots, t_n\}$ be a set of tiles. Denote $\langle a, b\rangle$ the free group on two generators. Fixing an arbitrary origin for each t_i on its boundary, let $w_i \in \langle a, b\rangle$ be the word obtained by tracing the boundary of t_i counterclockwise, with a representing moving right and b representing moving up.

Notice that the normal closure N_T of the tile words, which by definition is the group generated by $\{\alpha^{-1}w_i\alpha \mid \alpha \in \langle a, b\rangle,\ 1 \le i \le n\}$, is agnostic to the choice of origin, because any two origin choices differ by an appropriate α. The tiling group of T is then defined as the quotient $G_T = \langle a, b\rangle / N_T$. ⌟

With the above triangular tronimo tilings, our notation was already quite suggestive. We already calculated $w_1 = ababa^{-2}a^{-2}$, etc. in an example. However, G_T has some very complicated structure. To simplify the problem, we noticed that in order for $\alpha^{-1}w_1\alpha = 1$, etc., it suffices for $a^3 = b^3 = (ab)^3 = 1$. So we used the natural map $\varphi : G_T \to \langle a, b \mid a^3 = b^3 = (ab)^3 = 1\rangle$ defined by $\varphi(a) = a$ and $\varphi(b) = b$ to simplify the problem, recalling that R is exactly the Cayley graph of $\langle a, b \mid a^3 = b^3 = (ab)^3 = 1\rangle$.

For domino tilings, a horizontal domino has word $w_1 = a^2ba^{-2}b^{-1}$ and a vertical domino has word $w_2 = ab^2a^{-1}b^{-2}$. Again, we simplified the problem by considering a map $\varphi : G_T \to \langle a, b \mid a^2 = b^2 = 1\rangle \cong D_\infty$, the infinite dihedral group, whose Cayley graph is an infinite line.

After fixing origins of both Γ and the Cayley graph R of the groups we were interested in, we defined height functions $h_\tau : \Gamma \to R$. (Fixing the origin of D_∞ turned it into $\mathbb{Z}$.) Height functions were defined by travelling along tiling lines, with a representing moving right and b representing moving up. For triangular tronimo tilings, this is exactly what we did.

For domino tilings, we originally defined h_τ to increase counterclockwise around black tiles and decrease around white tiles. But because black and white tiles alternate, this exactly captures the relation $a^2 = b^2 = 1$, so the two definitions are really equivalent.

In each case, we then had to show that h_τ is well-defined. We relied on the ability to remove tiles while leaving the region simply connected. We now give a reference: the following general version was sketched in [CL90] and proven in full in [MP99].

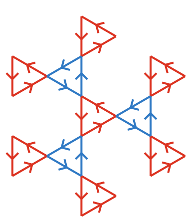
LEMMA 2.6. Given a tiling τ of a finite simply connected region Γ, there exists a tile t such that $\Gamma \setminus t$ is simply connected. ⌟

The proofs diverged from this point, as we used the height function in different ways. With dominoes, we found a way to construct a tiling (in fact, the minimum height tiling) by building pieces from the boundary. With triangular tronimoes, we used the height function to differentiate $\begin{smallmatrix}\square&\square\\\square&\end{smallmatrix}$ and $\begin{smallmatrix}&\square\\\square&\square\end{smallmatrix}$ from $\square\!\square\!\square$ and $\begin{smallmatrix}\square\\\square\\\square\end{smallmatrix}$, which gave a constraint on the number of $\begin{smallmatrix}\square&\square\\\square&\end{smallmatrix}$ and $\begin{smallmatrix}&\square\\\square&\square\end{smallmatrix}$ tiles in a tiling.

Now, let's see how Kenyon and Kenyon applied these ideas to a new problem. The two actually proved a little more than what we state here: they showed the result for $1 \times k$ and $\ell \times 1$ rectangles, but we'll focus on $k = 3$ for simplicity.

THEOREM 2.7 ([KK92]). If $\Gamma \subset \mathbb{Z}^2$ is simply connected, then there is a $O(n \log n)$ algorithm to determine if it is tileable by $T = \{\square\!\square\!\square, \begin{smallmatrix}\square\\\square\\\square\end{smallmatrix}\}$. ⌟

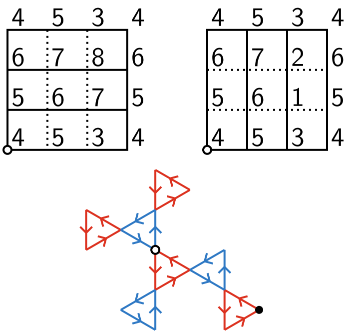
Proof sketch. As previously computed, the tile words are $w_1 = a^3ba^{-3}b^{-1}$ and $w_2 = ab^3a^{-1}b^{-3}$. Hence, the map $\phi : G_T \to \langle a, b \mid a^3 = b^3 = 1\rangle$ is a homomorphism and simplifies the problem. Its Cayley graph can be visualized in the following picture, similar to R in the previous section but without hexagons, expanding outwards in a tree-like structure.



We now get height functions $h_\tau : \Gamma \to R$, where R is the Cayley graph above. They are well-defined because the previous lemma gives us tiles that can be safely removed, and placing them back, it is easy to see that they locally keep heights well-defined.

The innovation in this proof is to consider a function $d_\tau : \Gamma \to \mathbb{N}$, where $d_\tau(x)$ is the distance in the graph R from the origin to $h_\tau(x)$. (The original paper and the lecture called this the height function.) This distance function serves the same purpose as the height function in the domino case.

Choose the origin (below as the filled black circle) to be outside the entire region $h_{\mathcal{T}}(\Gamma)$. Then, like in the domino case, flips can decrease the distance. (On each vertex in the tiling, the distance between its corresponding vertex in the Cayley graph to the origin is written. The empty circle shows where one vertex is mapped, the rest follow.)



The origin matters because whether the horizontal or vertical orientation has smaller distance depends on the relative positions of the region and origin. More specifically, call the lower left corner of the square x. Then the vertical tiling has smaller distances if and only if the red triangle adjacent to $h_{\mathcal{T}}(x)$ is closer to the origin than the blue triangle. One is always closer than the other because $h_{\mathcal{T}}(x)$ is not the origin, and R has a tree-like structure, so there is a unique acyclic path to follow.

In any case, the rest of the proof proceeds identically to the domino case. This idea of flips gives produces a tiling for which the maximum distance is found on the boundary. The point with maximum distance produces a unique compatible tronimo, and this can be iterated to produce a tiling, if one exists. □

Corollary 2.8. All tilings of simply connected regions by bars are flip-connected, by the kind of flip drawn above. ⌟

Rémila later generalized this argument to T consisting of any two rectangles of side length at least 3, also using tiling groups, in [Rém05].

One final thing to note is that tiling group arguments work best when the set of tiles is small. If there are too many tiles, the quotient $G_T = \langle a, b\rangle / N_T$ may become too small or too complicated to be interesting. However, there are exceptions. In [Kor04, Chapter 7], Korn used tiling group arguments to give algorithms and flip-connectivity results for tiling simply connected regions by the set of $2^k \times 2^{k-i}$ blocks over all i. The arguments work because although k may be large, the tiles are all quite simple.

3 Coloring arguments

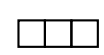
Recall that in the original domino problem, we colored the grid black and white, and started with the simple observation that for a region to be tileable, the number of black and white tiles must be the same. Compared to Conway's tiling group, this is an extremely simple idea, but it is still worth looking into a little deeper.

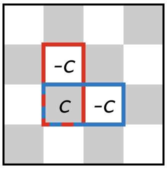
Definition 3.1. Let $T = \{t_1, \ldots, t_k\}$ be a set of tiles and let G be an abelian group written additively, the set of colors. A coloring function for T is a function $f : \mathbb{Z}^2 \to G$ such that for all translations t of any t_i,

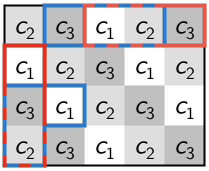
$$\sum_{x \in t} f(x) = 0.$$

The coloring group O_T is the set of coloring functions (under addition). It is the quotient of the free group on generators indexed by $\mathbb{Z}^2$ by the relations given by the above equations. ⌟

As an example, let $T = \{$, $\}$ and $G = \mathbb{Z}$. In the below picture (left), we translated two tiles (red and blue) to overlap. Because f must have the same sum on both tiles, we get that the top square and right square must be the same color, in particular, $-c$. By repeating this argument for neighboring squares, we get the checkerboard pattern that we expected. The coloring group O_T is isomorphic to $\mathbb{Z}$ because there is one coloring for every choice of c.

As a second example (right), consider $T = \{$, , , $\}$. In this case, we overlap  and  to get that all diagonals must be constant, then observe that by overlapping  with itself, there are three colors, the third of which is fixed by the first two. The coloring group is hence isomorphic to $\mathbb{Z}^2$ because there are two free colors.

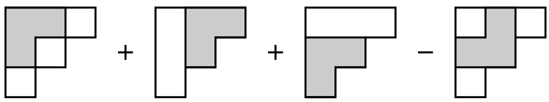




Colorings are useful as a quick first method to prove that a tiling does not exist. However, their power is limited. Our goal is to briefly explore just how limited their power is.

Definition 3.2. Denote χ_X the characteristic function of X. A signed tiling of Γ by T is a set of translated tiles τ and coefficients $a_t \in \{\pm 1\}$ such that $\chi_\Gamma = \sum_{t \in \tau} a_t \chi_t$. ⌟

As an example, recall that Γ_3, the triangular region with side length 3, cannot be tiled with  and . However, a signed tiling exists.



Proposition 3.3. If a region $\Gamma \subset \mathbb{Z}^2$ (possibly not simply connected) has a signed tiling by T and $f \in O_T$, then $\sum_{x \in \Gamma} f(x) = 0$. ⌟

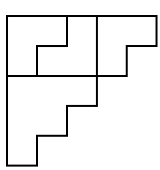
Proof. Split the sum over a signed tiling of Γ. The sum over each tile is just 0 by definition of f. □

Before, we noted that Γ must have equal numbers of black and white tiles to be tileable by dominoes. But in fact, the above proposition says that this condition must hold for the region to even have just a *signed* tiling. In other words, coloring arguments are so weak that they cannot differentiate signed and unsigned tilings.

As another example of signed tilings, we have the following result, to be contrasted with the analogous result for unsigned tilings (Theorem 2.1), where there were only tilings when $n \equiv 0, 2, 9, 11 \pmod{12}$.

Theorem 3.4 ([CL90]). The region Γ_n has a signed tiling with  and  if and only if $n \equiv 0, 2 \pmod 3$. ⌟

Proof. One direction trivial as we noted before: the total number of squares must be divisible by 3, so a tileable Γ_n has $n \equiv 0, 2 \pmod 3$. For the other direction, we induct from n to $n + 12$ by the same argument as the original proof. Hence, the only thing left to show are the base cases. The cases of 0 and 2 are trivial, 3 was shown just above, and 5 is shown below.



The cases of 6, 8, 9, and 11 are similar and left to the reader to verify. □

We note that more specifically, the proof of the Conway–Lagarias result (Theorem 2.1) fails with coloring arguments because such arguments cannot prove that the number of  tiles minus the number of  is constant. Indeed, for the Γ_3 region, this difference is 4 with signed tilings, but 1 for unsigned tilings.

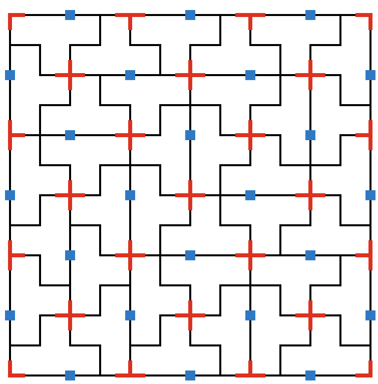
4 Tiling by T-tetraminoes

Let's consider another basic tiling problem: When can an $a \times b$ rectangle Γ be tiled by $T = \{\ ,\ ,\ ,\ \}$? An obvious first necessary condition is that $4 \mid ab$, and by noting that after applying a checkerboard coloring, each tile covers an imbalance of colors, we can quickly get that $8 \mid ab$. The full story is not that much more complicated and was resolved by Walkup.

THEOREM 4.1 ([Wal65]). An $a \times b$ rectangle is tileable by $\{\ ,\ ,\ ,\ \}$ if and only if $4 \mid a$ and $4 \mid b$. ⌟

Proof. The proof is actually quite elementary, not using any of the techniques we have seen so far. It relies on the following structural observation.

LEMMA 4.2. Any tiling of a rectangle by T-tetraminoes conforms to the following diagram. Red lines cannot be in the interior of any tile and blue squares cannot be the corner of any tile.

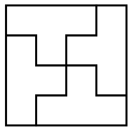


⌟

Proof sketch. By induction, starting from the lower-left corner, and inducting in diagonals towards to the upper-right. The base case involves the 2 short red lines in the lower-left and the two blue squares next to them. For the red, there is nothing to prove, and for the blue, only and can cover the lower-left square, and both leave the blue square cornerless. The actual inductive step is completely elementary but painful to write down, we refer to the reader to the original article for a full proof. □

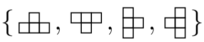
Simple casework remains. Align the above pattern to the lower-left corner of the rectangle. Suppose that the number of rows a is 1, 2, or 3 (mod 4). By considering the upper-left corner of the rectangle, it is obvious in every case that there is no valid tiling that respects the red lines. Similarly, the same is true for b.

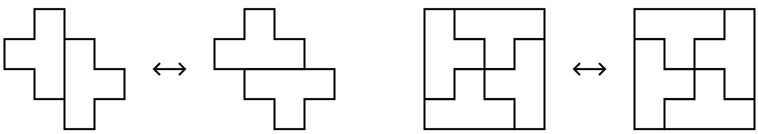
The other direction, which says that such rectangles do have tilings, is easy to see as every 4×4 block can be tiled as follows:



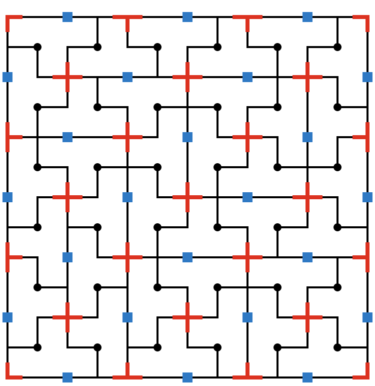
□

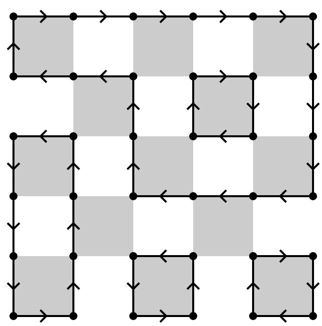
Recall for dominoes and bars, the answer to flip-connectivity came directly out of the algorithms for determining if a tiling exists. For this problem, however, the question of flip-connectivity is more involved, and was not resolved until more recently by Korn and Pak.

THEOREM 4.3 ([KP04]). All tilings of an $a \times b$ rectangle by  are connected by the following two kinds of flips.



Proof. Note that the above structural lemma implies for following: if one divides the rectangle into 2×2 boxes delimited by the red lines, then every such box has 3 squares from one tetramino and 1 square from another tetramino. Following this observation, given a tiling, define a directed graph with each box as a vertex, and an edge from box A to box B if from the same tetramino, A has 3 squares and B has 1. With our previous example, we get the following graph.

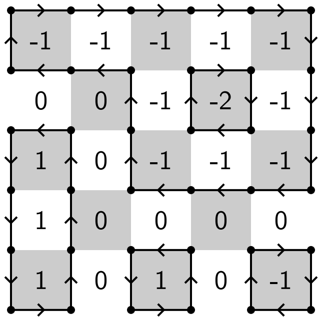




We make the following observations:

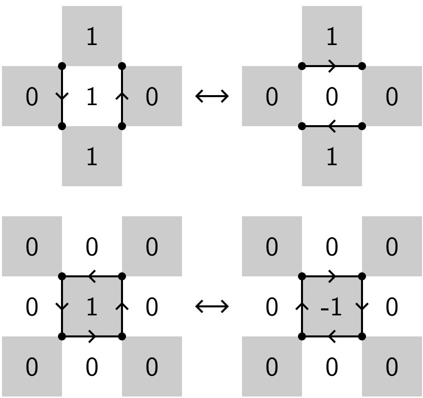
1. Every vertex has indegree and outdegree 1, so the graph is a union of disjoint directed cycles.
2. By the defining property of blue squares, the white boxes on the right must be bounded by two parallel edges (which may or may not go in the same direction).
3. A subgraph of $\mathbb{Z}^2$ satisfying the above two properties can be used to reconstruct a tiling.

Because of (1), we can define a "height function" that increases by 1 inside counterclockwise loops and decreases by 1 inside clockwise loops.



We notice that the parity of the height function on shaded squares is constant along rook moves (i.e. not diagonal). This is generally true because of (2): when passing by a white square, we pass either no edges (height unchanged), two edges in opposite directions (height unchanged), or two edges in the same direction (± 2 height). Also, half the shaded squares are odd and the other half are even, by looking at a corner.

By translating the original claim to these graphs, which is valid by (3), it remains to show that such graphs are connected by the following flips:



We will show that all graphs are flip-connected to the graph of a bunch of little counterclockwise loops around every other shaded square, which has height 0 on the outside and 1 inside every little loop. (This corresponds to repeating a tiling of 4×4 across the entire rectangle.)

Note that the two flips change the height function only at the middle square. We apply flips starting where the height is greatest. First, note that a white square where the height is greatest cannot be bounded by parallel lines in the same direction, so we flip all of these. Then a shaded square where the height is greatest must now have edges on all sides, so we flip all of these. Now the maximum height is smaller, so we repeat, and similarly increase the height where it is negative. Because of what we noted about the parity of shaded squares before, this finishes the proof. □

One unusual aspect of tilings by T-tetraminoes is that the number of tilings is related to another commonly studied object in combinatorics: the Tutte polynomial T_G of an undirected graph G. We allow G to have loops and multiple edges. Denote $G \setminus e$ the graph obtained by simply deleting the edge e, and G/e the graph obtained by contracting e: merging the endpoints before removing e. Then T is defined recursively as

$$T_G = T_{G\setminus e} + T_{G/e},$$

where e is any edge that is not a loop or cut edge (an edge whose deletion disconnects its connected component). The base case is a graph with i cut edges and j loops, for which $T_G(x, y) = x^i y^j$. An alternate definition is the formula

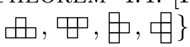
$$T_G(x, y) = \sum_{H \subset G} (x-1)^{c(H)-c(G)} (y-1)^{c(H)+|E_H|-|V_G|},$$

where $c(G)$ is the number of connected components of G, and the sum is over all spanning subgraphs.

The Tutte polynomial is able to count many things related G, such as the following:

- $T_G(x, 0)$ is the chromatic polynomial, the number of proper colorings of the vertices of G using x colors.
- $T_G(1, 1)$ is the number of spanning forests of G.
- $T_G(2, 1)$ is the number of forests that are a subgraph of G.
- $T_G(1, 2)$ is the number of spanning subgraphs of G.

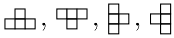
In the same paper as before, Korn and Pak showed the following.

THEOREM 4.4. [KP04] The number of tilings of a $4a \times 4b$ rectangle by {  } is

$$2T_G(3, 3) = \sum_{H \subset G} 2^{2c(H)-c(G)+|E_H|-|V_G|+1},$$

where G is the $a \times b$ grid. ⌟

5 Tiling of rectangles

In the previous section, we saw an ad hoc explanation of which rectangles can be tiled by $\{$  $\}$. Now, we turn our attention to tiling rectangles by arbitrary sets of tiles T. In general, we are interested in the set $S_T = \{(a,b) \in \mathbb{N}^2 \mid a \times b \text{ is tileable by } T\}$, and the question is whether or not there exists a polynomial time algorithm to decide whether or not $(a,b) \in S_T$. It will be useful for us to have a bit of algebraic language for this purpose. In this section, $\mathbb{N} = \{1, 2, \dots\}$.

Definition 5.1. An d-dimensional Klarner system (we will only use $n = 1, 2$) is a set $S \subset \mathbb{N}^d$ such that each coordinate is closed under addition. That is, when $d = 2$, we have $(a,b), (a',b) \in S$ implies $(a+a', b) \in S$ and $(a,b), (a,b') \in S$ implies $(a, b+b') \in S$. A Klarner system S is generated by P, denoted $S = \langle P \rangle$, if S is the smallest Klarner system containing P. ⌟

Equivalently, $\langle P \rangle$ is the set of all elements that can be obtained by P from applying addition in either coordinate as above. The set S_T is a Klarner system, because one can easily construct a tiling of $(a+a') \times b$ by stacking tilings of $a \times b$ and $a' \times b$, and similarly for the second coordinate.

The main theorem is the following result, first proven and used with similar language by de Bruijn and Klarner, but first explicitly stated in this form by Reid. Both papers proved for general d, but we focus on the case $d = 2$ for simplicity. Our proof follows Reid.

Theorem 5.2 ([dBK75, Rei05]). Let $S \subset \mathbb{N}^2$ be a 2-dimensional Klarner system and $R(y) = \{x \in \mathbb{N} \mid (x,y) \in S\}$ be the set of elements in row y. Then $R(y)$ is eventually periodic. ⌟

Proof. We first make a few easy observations about $R(y)$.

1. Each $R(y)$ is a 1-dimensional Klarner system. This is obvious.
2. A 1-dimensional Klarner system is finitely generated. To briefly show this, let $m = \min(S)$ and $x_i = \min\{x \in S \mid x \equiv i \pmod{m}\}$, if it exists. Then $S = \langle m, x_0, \dots, x_{m-1} \rangle$ (omitting ones that don't exist).
3. For all $y, y' \in \mathbb{N}$, we have $R(y) \cap R(y') \subset R(y+y')$. This is obvious.
4. If $y \mid y'$, then $R(y) \subset R(y')$. This is also obvious.

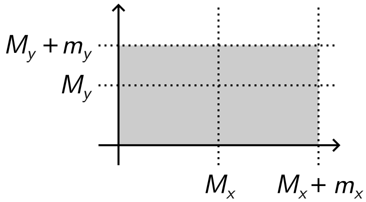
Next, we claim that there exists a maximal $R(y)$. Define the set $R_\infty = \bigcup_{y=1}^{\infty} R(y!)$. By observations (1) and (4), this is an increasing chain of Klarner systems. Note that the union of an increasing chain of Klarner systems is Klarner, so R_∞ is a (1-dimensional) Klarner system. By observation (2), R_∞ is finitely generated. Write $R_\infty = \langle P \rangle$. Because P is finite and $P \subset R_\infty$, there exists $y_0 = (y_0')!$ such that $P \subset R(y_0)$. Then $R_\infty = R(y_0)$, and $y \mid y!$ for all $y \in \mathbb{N}$, so this is maximal by observation (4).

Consider $R_y = \bigcup_{n=0}^{\infty} R(y + ny_0)$ for $1 \le y \le y_0$. By observation (3), $R(y+ny_0) \cap R(y_0) \subset R(y+(n+1)y_0)$, and $R(y_0)$ is the maximal one, so R_y is the union of an increasing chain. By the same argument as before, R_y is finitely generated, and it equals $R(y + n_y y_0)$ for some $n_y \in \mathbb{N}$. In other words, all of the increasing chains stabilize. Hence, after they all stabilize, i.e. for all $y > \max\{n_y y_0 \mid 1 \le y \le y_0\}$, we have $R(y) = R(y + y_0)$. □

We can now deduce the answer to our original question as an immediate corollary. We have the best case scenario: a logarithmic time algorithm for all T. The result was first noticed in the unpublished manuscript [LMP05].

Corollary 5.3. For any set of tiles T, there is a $O(\log ab)$ time algorithm to determine if $(a, b) \in S_T$. ⌟

Proof. By the theorem, $R(y) = \{x \in \mathbb{N} \mid (x, y) \in S\}$ is eventually periodic. So there exist $m_y, M_y \in \mathbb{N}$ such that for all $y > M_y$, we have $R(y + m_y) = R(y)$. By symmetry, $C(x) = \{y \in \mathbb{N} \mid (x, y) \in S\}$ is also eventually periodic, and there exist $m_x, M_x \in \mathbb{N}$ such that for all $x > M_x$, we have $R(x + m_x) = R(x)$.



Memorize the points in S in the shaded rectangle. As the rectangle depends only on T and not the input, it is just a large constant in the complexity of the algorithm. Then, given $(a, b) \in \mathbb{N}^2$, one uses the periodicity to do some easy casework and division (by m_x and m_y) to determine which point (a', b') in the rectangle the input corresponds to. We output whether or not (a', b') belongs to our memory. Division takes logarithmic time. □

An important note is that this is a non-constructive proof. How does one actually compute the shaded rectangle? The issue lies in algorithmically finding a finite generating set for R_∞ and R_y in a way that is guaranteed to terminate. However, no constructive proof can actually exist, because as the next section will show, this all leads to a computationally undecidable problem. That is, we know that an extremely fast algorithm always exists, but by an unfortunate twist of fate, we may not always be able to find it.

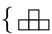
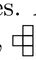
We can further investigate the structure of Klarner systems to reveal more structure about tilings of rectangles. In fact, what follows is the original purpose of the two aforementioned papers by de Bruijn and Klarner, and by Reid.

PROPOSITION 5.4. Given a Klarner system S and $p \in S$, the following are equivalent:

1. There does not exist $(a, b), (a', b) \in S$ such that $(a + a', b) = p$, nor the second coordinate.
2. The set $S \setminus \{p\}$ is a Klarner system.

If either of the above hold, the element p is called a prime. Furthermore, the set of all primes P is the minimal generating set of S. ⌟

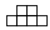
Proof. The equivalence (1) $\iff$ (2) is immediate from the definition. To show that P generates S, first order S by the sum of the coordinates, then lexicographically. This is a well-order. Let q be the least element of $S \setminus \langle P \rangle$. Then q is not prime, so it can be decomposed by (1) into $r, s \in \langle P \rangle$ because $r, s < q$. But then $q \in \langle P \rangle$, a contradiction. Lastly, P is clearly minimal, since by definition, a removed prime cannot be generated from other primes. □

In tiling language, a rectangle is prime if no tiling of it can be split into two of tilings of two subrectangles. As an example, recall from the previous section that if $T = \{$, , , $\}$, then $S_T = 4\mathbb{N} \times 4\mathbb{N}$. The only prime of this Klarner system is $(4, 4)$. In general, as another corollary of eventual periodicity, we have the following result.

COROLLARY 5.5. For any Klarner system S, the set of primes is finite. ⌟

Proof. It suffices to show that S is finitely generated, as one can simply remove elements until it is minimal to find the set of primes. With the same picture as the previous corollary, we claim that the shaded rectangle generates S.

Recall that the periods m_x and m_y are such that $C(m_x)$ and $R(m_y)$ are maximal. Hence, for all (a, b) outside the shaded rectangle, we first map it to some $(a', b') = (a - t_x m_x, b - t_y m_y)$ inside the shaded rectangle by periodicity. Because $R(m_y)$ is maximal, $a' \in R(b') \subset R(m_y)$. So $(a', m_y) \in S$, and we can construct (a', b) by putting together (a', b') and t_y copies of (a', m_y). We similarly do the other coordinate to generate (a, b) from the shaded rectangle. □

The paper [Rei05] gives many examples to compute the primes of S_T for small T. (Each computation must use ad hoc methods, since as previously noted, we gave a nonconstructive proof of the existence of the shaded rectangle.) For example, let T contain  and its rotations. Then the primes are $(4,6)$, $(6,4)$, $(5,12)$, and $(12,5)$. The proof involves a coloring argument and a couple specific permutations from S_{32}.

Part II

Complexity

6 An undecidable problem

In Part II, we will begin to investigate the computability and complexity of tiling in general, including many hardness results. Recall from the previous chapter that for every set of tiles T, there is a fast algorithm to determine if a given rectangle $a \times b$ is tileable by T. Let us ask a related question: Given T, is there a rectangle that it can tile? In other words, does $S_T = \emptyset$?

The answer is, surprisingly, that is no algorithm to solve this. We say that the problem is undecidable, much like the famous Halting problem. But to those unfamiliar with computer science, this could seem mysterious: how do you prove the lack of an algorithm?

In general, such hardness results in the field of computer science are proven using a technique called reduction. Knowing that a certain problem L is already hard, one can show that L' is also hard by somehow encoding L into the language of L'. Then, if one could solve L', it would also solve L, but L is hard, so L' must also be hard. We will employ this method throughout the next few chapters.

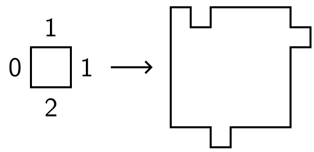
Let us warm up to the idea of reductions by showing a reduction in both directions between normal tiling and something called Wang tiling. One can think about this as meaning that normal tiling and Wang tiling have the same expressive power, and are equally difficult.

Definition 6.1. A Wang tile is a 1×1 square with sides labeled by a finite set of colors (for example, a subset of $\mathbb{Z}^k$). We may sometimes write $c\overset{d}{\underset{b}{\cdot}}a$. Given a set of Wang tiles T, a region $\Gamma \subset \mathbb{Z}^2$ is said to be tileable by T if the tiles in T can be arranged in Γ so that every edge is colored the same on both sides, and the boundary of Γ is colored by 0. ⌟

For example, one may tile a 1×4 rectangle with $T = \{0\overset{0}{\underset{0}{\cdot}}1, 1\overset{0}{\underset{0}{\cdot}}0\}$ like $0\overset{0}{\underset{0}{\cdot}}1\,1\overset{0}{\underset{0}{\cdot}}0\,0\overset{0}{\underset{0}{\cdot}}1\,1\overset{0}{\underset{0}{\cdot}}0$, but not like $0\overset{0}{\underset{0}{\cdot}}1\,0\overset{0}{\underset{0}{\cdot}}1\,1\overset{0}{\underset{0}{\cdot}}0\,1\overset{0}{\underset{0}{\cdot}}0$, as the colors on the inside don't match.

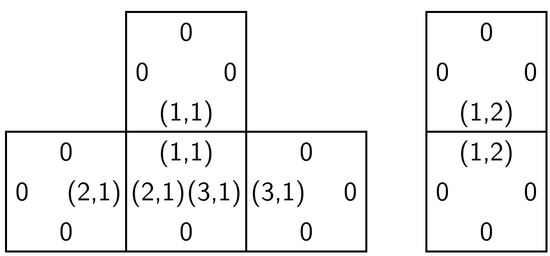
Lemma 6.2. Whether or not $S_T = \emptyset$ is undecidable for Wang tiling if and only if it is undecidable for normal tiling. ⌟

Proof. ($\Longrightarrow$) To prove that normal tiling is undecidable, we will show that if (towards a contradiction) there is an algorithm for it, then there is also an algorithm for Wang tiling. Here is the algorithm for Wang tiling: First, blow up the Wang tiles to become sufficiently large squares. For every color, create a new small edge pattern (like a puzzle piece), and have it protrude out of the right and bottom and inset into the left and top. Set the boundary color 0 to be a flat edge. In the below example, the number n becomes a bump or indent on the nth square of every edge, but there are many different ideas that all work.



Then there is a Wang tiling of the original rectangle if and only if there is a tiling with these puzzle piece tiles of the rectangle appropriately magnified, so we can answer Wang tiling using an algorithm for normal tiling.

($\Longleftarrow$) Here is the algorithm for normal tiling, assuming Wang tiling: For every tile t, first break it into 1×1 squares. To assign colors to make these into Wang tiles, let 0 be the color of the exterior edges. Create new colors for every interior edge, specific to tile t. This way, any one of these Wang tiles forces the others to be placed accordingly, as the only color on more than 2 Wang tiles is 0. Here is an example of how to transform $T = \{$, $\}$.



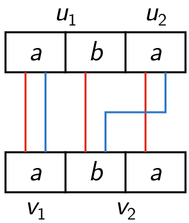
$\square$

To prove that rectangle tiling is undecidable, we will first use the above reduction to Wang tiling. Then, we will reduce again to a problem called Post's correspondence problem (PCP). That is, we will show how to solve PCP using Wang tiling. We follow a proof given by Yang.

Recall that A^* denotes all finite sequences of elements in A. The problem asks: given words $u_1, \ldots, u_k, v_1, \ldots, v_k \in \{a, b\}^*$, is there a word $w \in \{x_1, \ldots, x_k\}^*$ such that substituting u_i for x_i (denoted $w(u)$) produces the same word as v_i for x_i (denoted $w(v)$)? For example, if $u_1 = ab$, $u_2 = a$, $v_1 = a$, and $v_2 = ba$, then the answer is yes, with $w = x_1 x_2$. A proof that this problem is undecidable can be found on Wikipedia.

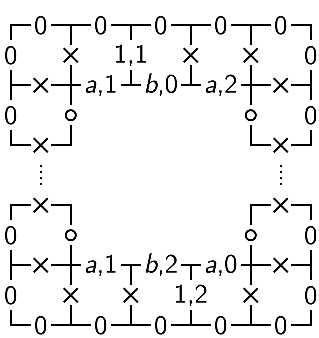
Theorem 6.3 ([Yan13]). Given a finite set of tiles T, in general it is undecidable if there exists a rectangle that can be tiled by T. ⌟

Proof. We give an algorithm based on Wang tiles to solve Post's correspondence problem. The approach is based on a few important abstractions that illuminate the general power of Wang tiles: border tiles, transmitting wires, and layering. With the above example of a PCP instance, we're aiming for a construction that captures the following picture.



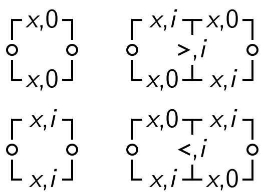
To explain the picture a bit, the top and bottom are the final word after substitution, keeping in mind which u and v words they originally come from. These are our border tiles, and we think of them as initial signals or initial colors. We will have two layers of transmitting wires. First, the red layer must always go straight up and down, ensuring that the first part of the signal (a or b) is the same. Second, the blue layer can turn, but cannot have crossings, and must pair up the first letter of each u and v word. The blue layer ensures that the order of indices of the u's and v's is the same, as is required by the problem.

Now, let's formalize. To do layering, the colors that represent signals on our Wang tiles will be ordered pairs. The first component will be a or b (the red wire), and the second component will be $i \in \{0, 1, \ldots, k\}$ (the blue wire, with 0 denoting no blue wire). Then, we can create new border tiles to set up the initial signals. In the below picture, the border tiles are set up to match our running example.



More specifically, we will have the four corner tiles above, the left and right side tiles above, and for every $i \in \{1, \ldots, k\}$, $|u_i|$ tiles for u_i and $|v_i|$ tiles for v_i. These word tiles are forced to appear in blocks using the same technique as the reduction before (introducing new colors for each internal edge), and u tiles have the signal on the bottom, whereas v tiles have the signal on the top.

Continuing the construction, we have tiles for transmitter wires. For each $x \in \{a, b\}$ and $i \in \{1, \ldots, k\}$, we will introduce the following six tiles. The two on the left allow the signal to pass through, while the four on the right allow the blue wire to turn.



Finally, we are ready to state our claim. Call the above set of Wang tiles T. Then there exists a rectangle tileable by T if and only if there exists a solution to the PCP problem. (This would complete the reduction.)

($\implies$) Consider the smallest rectangle tileable by T. Because only border tiles have 0, starting at the edge of the rectangle, one finds a subrectangle delimited by border tiles. Because we started with the smallest rectangle tileable by T, the subrectangle is the whole rectangle. In particular, we can read off a solution to the PCP problem by looking at the order in which word tiles appear in the top and bottom. This is a valid solution because the transmitter tiles provide the proof.

($\impliedby$) We essentially ran through the construction of a rectangle tileable by T when motivating our choice of tiles, so there is nothing left to prove here. $\square$

To give a bit of historical context on this problem, it was a classic result by Berger in [Ber66] that given a set of tiles T, it is undecidable whether or not the entire plane is tileable by T. His original proof gave a construction of a set of Wang tiles that only admitted aperiodic tilings: tilings of the plane that do not repeat periodically. A reduction to a problem about affine maps (the immortality problem) shows the undecidability. Recently, after a long line of work reducing the size of Berger's original tile set, it was shown by Jeandel and Rao in [JR15] that the smallest aperiodic Wang tile set contains exactly 11 tiles.

7 NP-complete tilings

In the previous section, we contrasted the existence of polynomial-time algorithms for tiling of rectangles with the nonexistence of any algorithm for determining if a tileable rectangle exists. But this was a kind of fundamentally different problem: we have previously been studying problems where the region Γ is part of the input, not something quantified over. So it remains to ask: we have seen many algorithms, but is there a set of tiles T for which it is computationally hard to determine if an input region Γ can be tiled? We review some basic definitions from computational complexity.

DEFINITION 7.1. A decision problem $L \subset \{0,1\}^*$ (encoded as the set of all true instances) belongs to the class NP if there exists a polynomial-time algorithm A such that:

- For all $x \in L$, there exists a polynomial-length string y such that $A(x, y)$ outputs true (accepts).
- For all $x \notin L$, $A(x, y)$ outputs false (rejects) for all strings y. ⌟

Informally, true instances have proofs y that can be quickly checked. Clearly, tiling an input $x = \Gamma$ by fixed T belongs to the class NP, as an example tiling (y above) can easy be checked to be valid in polynomial time. So, tiling cannot be too hard. In particular, with Wang tiles, there is always at least an exponential-time, polynomial-space algorithm by just enumerating every possibility and checking each one.

DEFINITION 7.2. The problem $L \in \mathsf{NP}$ is NP-complete if an oracle that answers queries to L can be used to answer all other problems in NP in polynomial time (i.e. a reduction exists). ⌟

Informally, NP-complete problems are the hardest problems in NP. It is initially not clear that NP-complete problems even exist, but the hugely important Cook–Levin theorem proved that a problem called SAT (satisfiability) is NP-complete. Most people do not believe that polynomial time algorithms exist for NP-complete problems (i.e. they believe $\mathsf{P} \neq \mathsf{NP}$).

A CNF (conjunctive normal form) formula in the Boolean variables $x_1, \ldots, x_n$ is a conjunction ($\wedge$ or "and") of disjunctions ($\vee$ or "or") of literals $x_1, \neg x_1, \ldots, x_m, \neg x_n$. Each disjunction of literals is called a clause. The

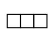
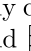
SAT problem asks: given a CNF formula, does there exist an assignment $\alpha : \{x_1, \ldots, x_n\} \to \{0, 1\}$ that makes the formula true? For example, $(x_1 \vee \neg x_3) \wedge (\neg x_1 \vee x_2 \vee x_4)$ is a CNF formula, and setting all the variables to true is a satisfying assignment.

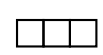
Cook's proof (Levin was an independent discovery) takes an input x and the Turing machine (algorithm) A that checks membership in L, and writes a SAT formula in variables y that is true if and only if the Turing machine accepts on (x, y). Therefore, the SAT formula has a satisfying assignment if and only if the Turing machine accepts for some proof y.

Knowing that SAT is NP-complete, it is much easier to prove that other problems are NP-complete by reducing them to SAT. For example, it is easy to see that 3SAT, which restricts each clause to at most 3 literals, is NP-complete. For every long clause $(\ell_1 \vee \cdots \vee \ell_n)$, introduce new variables $z_3, \ldots z_{n-1}$ and replace the long clause with

$$(\ell_1 \vee \ell_2 \vee z_3) \wedge (\neg z_3 \vee \ell_3 \vee z_4) \wedge \cdots \wedge (\neg z_{n-2} \vee \ell_{n-2} \vee z_{n-1}) \wedge (\neg z_{n-1} \vee \ell_{n-1} \vee \ell_n).$$

The meaning of z_i is "we rely on $\ell_i, \ldots, \ell_n$ to satisfy the clause," from which it is clear that the new clauses have a satisfying assignment if and only if the original long clause does. Many further variants of SAT are also NP-complete, such as planar 3SAT: the problem where the CNF formula, when drawn as a logical circuit in the plane, has no wire crossings. One can do this reduction by replacing every crossing with a few new variables and clauses, see Wikipedia for details.

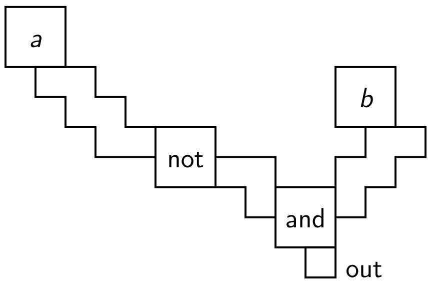
The existence of NP-complete tiling problems has been known since the beginning of NP-completeness theory. In fact, it was one of six problems mentioned in Levin's original paper, although Levin did not provide a proof. Since then, several papers have come out giving examples of tile sets T for which tiling is NP-complete. Recall the linear-time (with some logarithms) Kenyon–Kenyon algorithm for determining if a simply connected region Γ is tileable with bars $1 \times k$ and $\ell \times 1$ (we showed  and ). It turns out that if one removes the simply connected requirement, the problem becomes NP-complete. This result should also be contrasted with recalling that running a perfect matching algorithm solves domino tiling in polynomial time, even when the region is not simply connected.

Theorem 7.3 ([BNRR95]). Tiling $\Gamma \subset \mathbb{Z}^2$ by $T = \{$, $\}$ (or any $1 \times k$ and $\ell \times 1$ bars, for $k, \ell \geq 2$ and $k + \ell \geq 5$) is NP-complete. ⌟

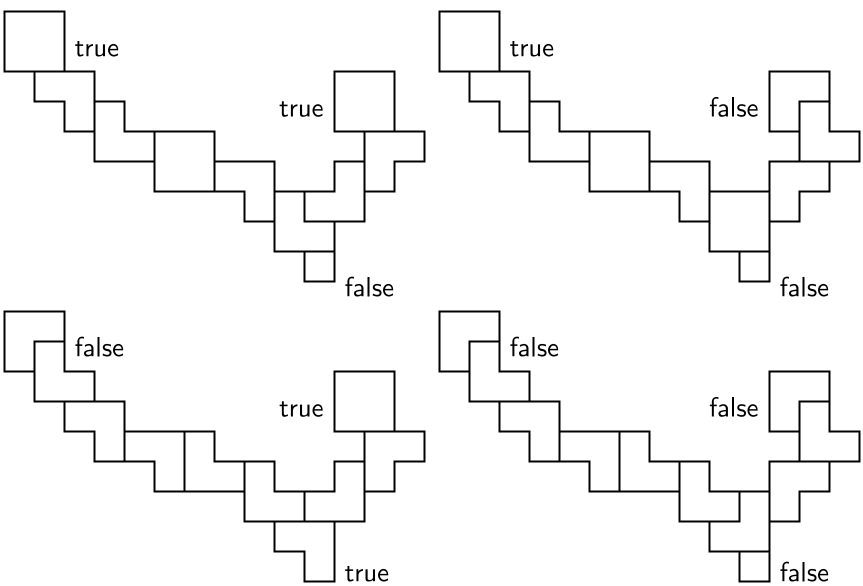
The proof technique involves a reduction to planar 3SAT by engineering "logic gates" out of tilings, but the actual gates are quite complicated to draw in this case. The following result by Moore and Robson uses a larger set of tiles and the same proof technique, which leads to a much cleaner construction.

Theorem 7.4 ([MR01]). Tiling $\Gamma \subset \mathbb{Z}^2$ by $T = \{$, , , , $\}$ is NP-complete. ⌟

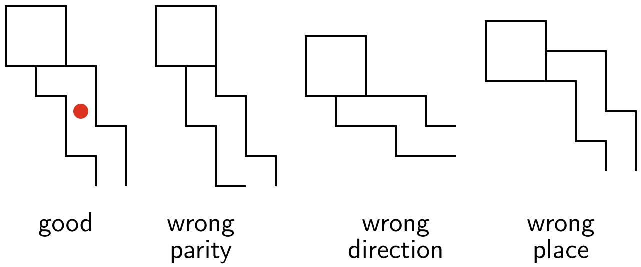
Proof sketch. As noted above, it suffices to translate every planar SAT formula, i.e. a circuit drawn in the plane, into a region Γ that has a tiling if and only if the formula is satisfiable. We will show a simple example, note some of the additional technicalities, and refer readers to the original paper for the full proof. Note that because $a \vee b$ is equivalent to $\neg(\neg a \wedge \neg b)$, it suffices to translate $\neg$ and $\wedge$ gates. Here is an example of Γ constructed from $\neg a \wedge b$. (The interior lines are to be disregarded, they delineate what we consider "gates" vs. "wires" and are irrelevant for tiling.)



A key feature of this region is that after fixing a tiling for the input portions (here a and b), the tiling is locally forced, working from input to output. Here are attempts of tilings constructed from the local uniqueness property. You can see the two tilings of a source, the two tilings of $\neg$ gate, and the four tilings of a $\wedge$ gate.



It may still be mysterious how one should write down steps to construct such a region, but after seeing the example, it should feel reasonable that this process is possible. To say a couple more details, the wires are a priori straight and travel in one of eight directions with knight's move periodicity. Every non-wire element has ports with fixed direction and "parity". The following picture shows the definition of a source.



Note that the truth value carried on a wire can be determined by whether the "middle" square (marked in red above, bounded by parallel lines) is part of the preceding tile (true) or following tile (false), where the order goes from sources to output. Verify this on the previous examples, although it can be tricky to find the middle square when the wire is bending.

Two types of wire bending were shown in our original example, which together allow any direction to change into any other direction. On the left, the wire leaves source a in the $(1, -2)$ direction as required, and then bends to the $(2, -1)$ direction as required for the $\neg$ gate. Between the $\neg$ gate and $\wedge$ gate, the wire does not need to bend. On the other side, the wire leaves b in the $(1, -2)$ direction and makes a $90°$ bend to the $(-2, -1)$ direction, which is the required direction for the second input to an $\wedge$ gate. One checks that after every bend, the truth value is preserved.

There are two more elements of the construction which were not needed in our simple example. Both are simple pictures, and the original paper has details. First, one needs a way to split one wire into two. And secondly, when the graph is not a tree, wires may not fit when they have the wrong "parity", as hinted above. There is an easy way to rotate between all parities, which again is detailed in the original paper. □

To summarize all the results on tiling that we have seen so far, we know that for dominoes, it is easy to determine if a general region Γ is tileable, but for even slightly more complicated tilesets, it is hard. On the other hand, for rectangular regions Γ, there is always an efficient algorithm. In the middle remains simply connected Γ, which we investigated before and used techniques like Conway's tiling group to find efficient algorithms for specific T. One may wonder if simply connected regions can be hard to tile, and indeed they can be, due to a more recent result by Pak and Yang.

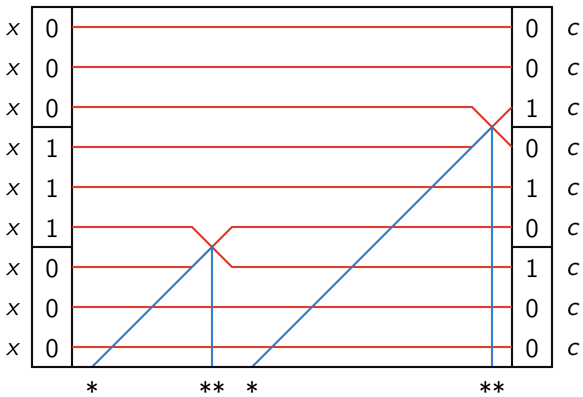
Theorem 7.5 ([PY13b]). There exists a finite set of (Wang) tiles T such that tiling simply connected $\Gamma \subset \mathbb{Z}^2$ by T is NP-complete. ⌟

Proof. The proof is a reduction to cubic monotone 1-in-3SAT, a variant of SAT known to be NP-complete by [Gon85] (where it is stated in slightly different language). 1-in-3SAT means that exactly one literal in every clause is allowed to be true. In 1-in-3SAT, monotone means that there are no negations. (This would trivialize the problem in normal 3SAT by setting all variables to true, but is hard in 1-in-3SAT.) Lastly, cubic means that every variable appears in exactly 3 clauses. By simple counting, this means there are the same number of variables and clauses. One visualizes this as a bipartite graph with variables $x_1, \ldots, x_n$ on one side and clauses $c_1, \ldots, c_n$ on the other, where each vertex has degree 3.

Note that the problem is a little different than our previous use of Wang tiles. Here, T is not part of the input, so our reduction needs to construct a set of Wang tiles T that work no matter how long the SAT formula is. But we note a few similarities:

1. We can still specify arbitrary border colors for our region. Recall how Wang tiling without border colors implies an algorithm for normal tiling, and vice versa. That second algorithm (creating edge patterns like puzzle pieces) actually works for Wang tiling with border colors, as our region no longer needs to be a rectangle and can have edge patterns on the boundary. And obviously, Wang tiling with border colors implies an algorithm for Wang tiling without. So all three problems are equivalent under polynomial time reductions.
2. We can still use the concept of wires, signals, and layering, as long as the number of distinct signals is fixed.

Hence, consider the following example, which represents the CNF formula $(x_1 \vee x_1 \vee x_2) \wedge (x_1 \vee x_2 \vee x_3) \wedge (x_2 \vee x_3 \vee x_3)$. The region Γ is the large rectangle, the border colors are shown on the outside, and a proper tiling is shown on the inside corresponding to $x_1 = 0$, $x_2 = 1$, and $x_3 = 0$.



Along the left, we use border colors to force variable tiles, which emit three 0 or three 1 signals. Along the right, we force clause tiles, which receive exactly one 1 signal and two 0 signals. Variable tiles and clause tiles are forced to come in stacks of 3 using the same technique as in solving normal tiling with Wang tiling: introduce some new colors that are unique to each interior edge.

Using border colors along the bottom, the entire wiring is actually fixed. In particular, we have a special signal ($*$) to start a diagonal signal, a special signal ($**$) to start a vertical signal, and a 2-stack tile that exchanges red signals at the spot where the blue signals intersect. To summarize, there are 6 kinds of wire tiles: red horizontal only, blue vertical with red horizontal, 2 tiles for the exchange, and 2 tiles for the blue diagonal (one blue on the bottom and right, the other on the left and top).

We can find an appropriate order of the wire transpositions by looking at the aforementioned bipartite graph and reading off the edge crossings from left to right, and then we space out the crossings and place the $*$ and $**$ signals accordingly. The resulting Γ may be very wide, but definitely still polynomial, so the reduction is done. □

Pak and Yang actually proved more. In the same paper, they extend this result by showing that one may take T to be a set of rectangles, by essentially finding a way to simulate the Wang tiles with rectangles. This is in contrast to previous work by Kenyon–Kenyon and Rémila mentioned in previous sections, which showed that when T contains just 2 rectangles, the problem admits a polynomial time algorithm. This original paper used 10^6 rectangles, which has since been reduced many times, and the current record is 117 [Yan13]. It remains open to close the gap between 117 and 2.

8 #P-complete tilings

We turn our attention to counting tilings. Here is one basic question: how many tilings of Γ by dominoes exist? Like most questions about dominoes, it turns out that counting the number of domino tilings admits a polynomial time algorithm. This follows from general techniques about perfect matchings, which will be covered in Chapter 11. But how about other problems? It turns out that most are quite difficult.

Definition 8.1. Suppose we have a decision problem L with two inputs x and y. The counting problem "given x, how many y exist such that $(x, y) \in L$?" belongs to the class $\#\mathsf{P}$ (sharp P) if $L \in \mathsf{P}$. A problem in $\#\mathsf{P}$ is $\#\mathsf{P}$-complete if every other problem in $\#\mathsf{P}$ can be reduced to it. ⌟

When $L \in \mathsf{NP}$, we denote $\#L \in \#\mathsf{P}$ the problem of counting the number of proofs of x, and we call it the counting version of L. (Note that there can be many ways to prove x, but typically there will be one "obvious" way, which is what we will mean.) For example, #SAT counts the number of accepting assignments to a CNF formula. The counting analog is always harder then the original decision problem: given a solution to the counting problem, simply output whether or not the count is at least 1 to solve the decision version.

The problem #SAT is #P-complete. This is a consequence of the proof the Cook–Levin theorem, which was what is called a parsimonious reduction. A reduction solving L_1 using L_2 is called parsimonious if it relies on a bijection between L_1 and L_2 in which inputs identified by the bijection have an equal number of proofs. For example, the reduction we gave solving SAT using 3SAT is *not* parsimonious. Recall that we replaced every long clause $(\ell_1, \ldots, \ell_n)$ with

$$(\ell_1 \vee \ell_2 \vee z_3) \wedge (\neg z_3 \vee \ell_3 \vee z_4) \wedge \cdots \wedge (\neg z_{n-2} \vee \ell_{n-2} \vee z_{n-1}) \wedge (\neg z_{n-1} \vee \ell_{n-1} \vee \ell_n).$$

Note that by introducing new variables, we increased the number of satisfying assignments. (For example, if $\ell_1 \vee \ell_2$ and ℓ_3 are both true, then z_3 can be assigned anything.) However, it is an elementary exercise to modify this to become a parsimonious reduction, where there is exactly one satisfying assignment to the new variables for every satisfying assignment to the original variables, and this would show that #3SAT is #P-complete. Planar #SAT is also #P-complete. In fact, even #2SAT, the problem of satisfying CNF formulas with 2 literals per clause, is #P-complete, despite the fact that the decision version admits a polynomial time algorithm.

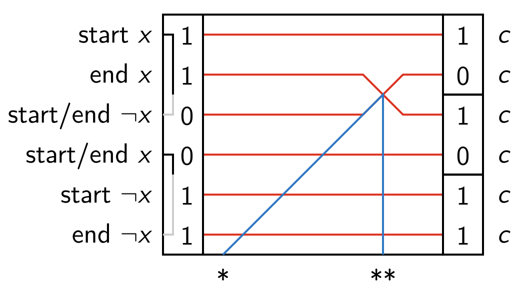
We note that the reduction used in the Moore–Robson result on the NP-completeness of tiling by $\{$, , , , $\}$ was indeed parsimonious: the tiling was unique once the inputs were fixed, so the number of tilings is the same as the number of satisfying assignments. Because planar #SAT is #P-complete, this tiling problem is also #P-complete.

How about the Pak–Yang result on simply connected regions? The reduction is indeed parsimonious: recall that the wiring is fixed, so the number of tilings is the number of satisfying assignments. However, it is actually not known whether or not counting cubic monotone 1-in-3SAT is #P-complete. Instead, to prove #P-completeness, the authors use a similar parsimonious reduction to 2SAT.

Theorem 8.2 ([PY13b]). There exists a finite set of (Wang) tiles T such that tiling simply connected Γ by T is #P-complete. ⌟

Proof. Given a 2SAT formula, the main challenge is that the number of times each variable is used is no longer fixed, and that some of these must be negated. However, we are allowed to engineer an appropriate left border for every formula. Because we know for every x_i, how many times it appears positively and negatively in the formula, we introduce the following 4 border signals: "start x", "end x", "start $\neg x$", and "end $\neg x$", as well as signals for edge cases where the start is the end, x does not appear negatively, etc. This creates wires from the start of each variable to the end of each variable, ensuring that they are the same value, and the wire tile adjacent to "start $\neg x$" is required to negate the value of the wire. Lastly, we change the clause tiles accept 10, 01, and 11 (because this is 2SAT), and the rest is the same as before. We note this is parsimonious, so the theorem is proved.

Below is the example $(x_1 \vee \neg x_1) \wedge (x_1 \vee x_2) \wedge (x_2 \vee \neg x_2)$ with satisfying assignment $x_1 = 1$, $x_2 = 0$. Note that we set the border signals by counting that x_1 appears twice positively and once negatively, likewise for x_2.



□

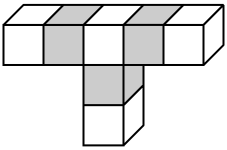
Of course, the question about #P-completeness is most interesting when in fact, the decision version of the problem is in P, as is the case for #2SAT. Are there tiling problems of this nature? In our usual setting of the plane $\mathbb{Z}^2$, nothing is known: it remains open, for example, whether tiling by bars $1 \times k$ and $\ell \times 1$ is #P-complete, even for general (non-simply connected) regions. If we allow ourselves $\mathbb{Z}^3$, the following is a theorem of Valiant.

Theorem 8.3 ([Val79]). Counting the number of tilings of $\Gamma \subset \mathbb{Z}^3$ by dominoes is #P-complete. ⌟

Note that by running any perfect matching algorithm as before, tileability by dominoes, even in 3 dimensions, admits a polynomial time algorithm, so this is interesting.

Proof. The following is known: it is #P-complete to count the number of perfect matchings (subset of edges containing every vertex exactly once) in cubic (3-regular) bipartite graphs. So it suffices to solve this problem using domino tiling in a parsimonious way.

Every graph can be embedded in 3 dimensions without crossing edges. Moreover, we can draw the edges as integer paths, i.e. a sequence of points in $\mathbb{Z}^3$ where adjacent points differ by exactly 1 in exactly 1 coordinate. We can further stipulate that the paths are sufficiently spaced apart so that no two paths touch, considered as sequences of cubes. Lastly, recall that our graph is bipartite. If we color $\mathbb{Z}^3$ in a checkerboard pattern, we can make sure that one part has vertices on white cubes, and the other has vertices on black cubes. A vertex is drawn below.



Note that all paths between vertices involve an even number of cubes. Hence, given a perfect matching, we can construct a domino tiling as follows: for every edge in the matching, (uniquely) tile the path between the two vertices (including the endpoints). Because the matching is perfect, the remaining paths in the graph have both endpoints already tiled, so they are still even length, and there exists a unique tiling of them. It is clear that this is a bijection. □

In [PY13a], Pak and Yang improve upon this result by showing that even for simply connected $\Gamma \subset \mathbb{Z}^3$, tileability by dominoes is #P-complete.

9 Sequences of tiling counts

Until now, we have considered the question of whether Γ is tileable by T. We have also considered the question of counting how many ways Γ can be tiled by T. In this section, we will go one step further and ask: given a sequence of regions $\Gamma_1, \Gamma_2, \ldots$, what can we say about the sequence $a(1), a(2), \ldots$, where $a(n)$ is the number of tilings of Γ_n by T?

One elementary example found in undergraduate textbooks involves tiling the sequence of regions , , , ... by $T = \{$, $\}$. A simple induction argument shows that $a(n)$ is the nth Fibonacci number.

Here is another example about domino tilings. Let Γ_n be a $2n \times 2n$ grid. Then we can bound the asymptotic growth of $a(n)$: $2^{n^2} < a(n) < 4^{2n^2}$. The lower bound comes from choosing one of two tilings of a 2×2 square n^2 times. The upper bound comes from using a checkerboard pattern and picking one of four white squares adjacent to each of the $2n^2$ black squares. Kasteleyn calculated the precise asymptotic to be $e^{4Gn^2/\pi} \approx 3.210^{n^2}$, where $G = 1 - \frac{1}{4} + \frac{1}{9} - \frac{1}{25} + \cdots \approx 0.916$ is Catalan's constant [Kas61].

Asymptotics are interesting, but the complexity of an infinite sequence can often be better captured through studying its generating function. For a more detailed description of the below classes, see Chapters 4 and 6 of Richard Stanley's *Enumerative Combinatorics* [Sta11, SF99]. We recap some of the main ideas as they pertain to our discussion here.

Definition 9.1. The (ordinary) generating function associated with the sequence $a(1), a(2), \ldots$ is the formal power series $A(t) = \sum_{n=1}^{\infty} a(n) t^n$.

- If $A(t) = \frac{P(t)}{Q(t)}$ for some polynomials P and Q with rational coefficients, we say that $A(t)$ is rational. Equivalently, $a(n)$ satisfies a linear recurrence with constant coefficients.
- If $P_0(t) + P_1(t)A(t) + \cdots + P_n(t)A(t)^n = 0$ for some $P_0, \ldots, P_n$ with rational coefficients, we say that $A(t)$ is algebraic of degree n.
- If $a(n) = f(n, \ldots, n)$ for some f for which $F(\bar{x}) = \sum_{\bar{n}} f(\bar{n}) \bar{x}^{\bar{n}}$ is rational, we say $A(t)$ is the diagonal of a rational.
- If $P_0(t)A(t) + P_1(t)A'(t) + \cdots + P_n(t)A^{(n)}(t)$ for some $P_0, \ldots, P_n$ with rational coefficients, we say $A(t)$ is differentiably finite. Equivalently, $a(n)$ satisfies a linear recurrence with coefficients polynomial in n. ⌟

These classes are known to contain each other in the listed order, but are different. For some examples, the Fibonacci numbers satisfy a linear recurrence with constant coefficients, and is hence rational. The central binomial coefficients $\binom{2n}{n}$ have generating function is $A(t) = \frac{1}{\sqrt{1-4t^2}}$ (the proof is not immediate but well-known, see Wikipedia). This is clearly not rational, but is algebraic as it satisfies $(1-4t^2)A(t)^2 - 1 = 0$.

The generating function of the closely related sequence $a(n) = \binom{2n}{n}^2$ is the diagonal of $F(w,x,y,z) = \frac{1}{(1-w-x)(1-y-z)}$. To see this, note that $\frac{1}{1-f} = \sum_n f^n$, and the coefficient of $w^n x^n y^n z^n$ in $(\sum_n (w+x)^n)(\sum_n (y+z)^n)$ is clearly $\binom{2n}{n}^2$. In [Fur67], Furstenberg noted that this generating function is not algebraic.

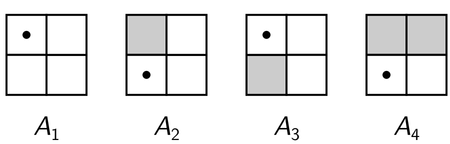
The number of permutations of n, that is, $a(n) = n!$, is differentiably finite, as it satisfies the recurrence $a(n) = na(n-1)$. This is not a diagonal, as diagonal functions have non-zero radius of convergence, whereas $A(t) = \sum_n n!t^n$ converges only at $t = 0$. Lastly, one example that lies outside of this hierarchy entirely is the number of alternating permutations: those that satisfy $\sigma(1) < \sigma(2) > \sigma(3) < \sigma(4) > \cdots$ [FGS05].

Now we can ask: where do tilings fall in this picture? Here is one answer, regarding the case where Γ_n is a rectangular strip with fixed height. The following is due to Merlini, Sprugnoli, and Verri.

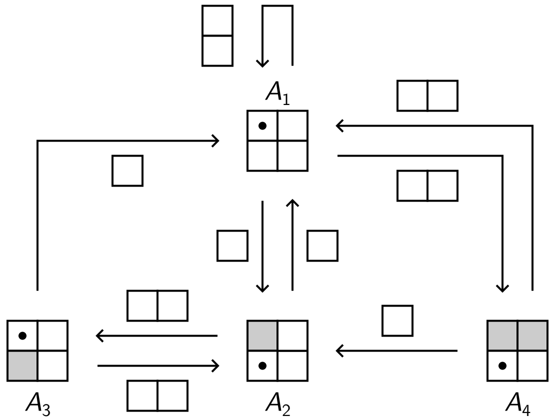
Theorem 9.2 ([MSV00]). Fix any set of T and let Γ_n be a $k \times n$ rectangle. Then if $a(n)$ denotes the number of tilings of Γ_n by T, $A(t)$ is rational. ⌟

Proof. Let m be the maximum width of any tile in T. Consider tiling Γ_n from left to right. That is, we always identify the leftmost column with an unoccupied square, then place a tile that covers the topmost unoccupied square in this column. Note that under this process, we only have to focus on a $k \times m$ window at all times. We call this window that notes which squares are currently occupied the tiling state.

Below, we draw all the tiling states for $T = \{\square, \square\!\square, \boxminus\}$ with $k = 2$. The leftmost, topmost square (where the next tile will go) is marked with a dot. Note that the configurations that do not appear in our tiling process are not shown.



We can draw a directed graph with states as vertices and edges labeled by tiles to show how we can transition between states by placing a tile on the dot. (In computer science, this is known as a deterministic finite automaton.) We note that every tiling of $2 \times n$ corresponds to a closed walk in this graph from the starting state A_1 to itself.



Next, let $A_1(t), \ldots, A_s(t)$ be generating functions, where the corresponding sequence $a_i(n)$ is the number of tilings of a strip that ends in state A_i and has length n. Note that $A_1(t) = A(t)$. We write a system of equations. In particular, if one transitions from state A_i to A_j using a tile that occupies c new columns, we include the term $t^c A_i(t)$ in the expression for $A_j(t)$. We also include the term 1 for our initial state A_1, because $a_1(0) = 1$. Our running example produces the following.

$$\begin{aligned}
A_1(t) &= 1 + tA_1(t) + A_2(t) + A_3(t) + A_4(t)\\
A_2(t) &= tA_1(t) + tA_3(t) + A_4(t)\\
A_3(t) &= tA_2(t)\\
A_4(t) &= t^2 A_1(t)
\end{aligned}$$

This is a linear system of equations. So we solve using some simple linear algebra and conclude that $A_1(t)$ is a rational function in t. In particular, for this example, we get:

$$A(t) = A_1(t) = \frac{t-1}{-t^3 + t^2 + 3t - 1} = 1 + 2t + 7t^2 + 22t^3 + 71t^4 + \cdots . \quad \square$$

If you prefer to think in linear recurrences, one can easily translate the above system of equations. For instance, the last equation above translates to $a_4(n) = a_1(n-2)$. By finding the recurrence for a_1, this proof essentially shows that the simple induction argument mentioned in the beginning for domino tilings is actually sufficient to solve all strip tiling problems.

As noted by Garrabrant and Pak in [GP14a], tilings actually belong to a class smaller than rationals: $\mathbb{N}$-rational functions. This is the smallest set of rational functions containing polynomials and closed under sum, product, and the operation $f^* = \frac{1}{1-f} = \sum_n f^n$ whenever this sum exists (it does if and only if $f(0) = 0$). These are often studied in relation to automata and regular languages in theoretical computer science.

To get more complicated sequences, we can do one of two things: either allow the region to increase in both dimensions, or consider non-integer tiles. Let us discuss the second option first. We note that tiles with rational side lengths can be effectively made integer by scaling, so we would need to use irrational side lengths.

For example, consider tiling a $1\times n$ rectangle by $1\times(\frac{1}{2}+\epsilon)$ and $1\times(\frac{1}{2}-\epsilon)$ tiles, where $0<\epsilon<\frac{1}{2}$ is any irrational number. By elementary counting, the number of possible tilings is the central binomial coefficient $\binom{2n}{n}$. We noted before that this is algebraic and not rational.

If we take T to be rectangles $1\times(\frac{1}{2}\pm\epsilon_1)$ and $1\times(\frac{1}{2}\pm\epsilon_2)$ for irrational, algebraically independent $0<\epsilon_1,\epsilon_2<\frac{1}{2}$, then with a bit of work we can count that there are $\binom{2n}{n}^2$ ways to tile $1\times n$. We noted before that this is diagonal and not algebraic.

In general, we have the following result, due to Garrabrant and Pak. Note the additional restriction that T consists of tiles of height 1, i.e. touching both the top and bottom boundaries of the region, although the authors conjecture that the theorem would also hold without the restriction.

THEOREM 9.3 ([GP14a]). Given $a(n)$, the following are equivalent

1. There exists a set of (real) height 1 tiles T such that $1\times n$ is tileable by T in $a(n)$ ways.
2. $a(n)$ is the sum of products of binomial coefficients with linear arguments (i.e. something like $\sum_{k,\ell}\binom{2n+k+2}{k}\binom{2k}{\ell}^2$). ⌟

In particular, one can show that $a(n)$ satisfying (2) has diagonal generating function, and is hence also differentiably finite.

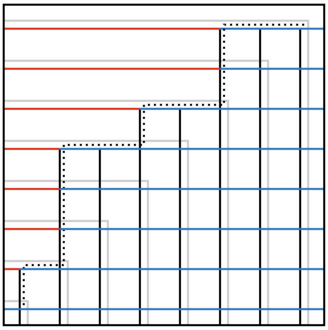
Next, recall that Wang tiles and normal (integer or rational) tiles are computationally effectively the same. This will make our arguments a bit simpler as we discuss what are the possible sequences that can be expressed in square Γ_n, extending in both dimensions. We do have an obvious asymptotic bound: $0\le a(n)\le |T|^{n^2}$, by choosing one of $|T|$ tiles in each spot, so $a(n)$ cannot grow too quickly (say, doubly exponential). Resolving this in full is still an open question, but at the very least, we know of many positive examples of $a(n)$ that can be counted, and they span our entire hierarchy of classes. Here are a few, again due to Garrabrant and Pak.

PROPOSITION 9.4 ([GP14b]). There exists a set of Wang tiles T such that with Γ_n being the $n\times n$ square, $a(n)$ counts

1. the Catalan numbers.
2. the number of permutations $n!$.
3. the number of alternating permutations.
4. the number of connected graphs on n vertices. ⌟

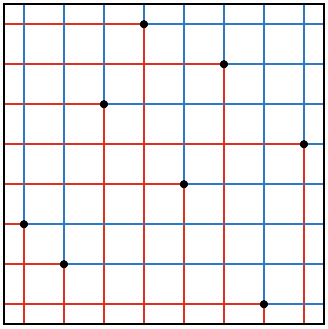
Proof. To be able to translate these results into normal tilings of squares, we will not allow border colors in the sense we used when studying NP-completeness. However, we can still make a set of border tiles: the pattern in which they are arranged just needs to be fixed, as all of ours will be.

1. The idea is to encode Dyck paths. We first encode the diagonal by using boundary tiles to force the gray wires to be present in all tilings. Next, we introduce color-changing wires that mark off the Dyck path (shown as a dotted line). These start off red, then change to blue (except the bottom wire, which is always blue). Blue wires always generate downwards black wires. There are no tiles in which red wires touch a black wire from above or enter a diagonal tile, so they are forced to stay in a region outlined by a Dyck path.



Notably, Catalan numbers form an algebraic rational generating function that is not rational. Computing this is actually quite simple: one recalls the recurrence $C_{n+1} = \sum_{i=0}^{n} C_i C_{n-i}$ and then computes that generating function satisfies $F(t)^2 = \frac{F(t)-1}{t}$.

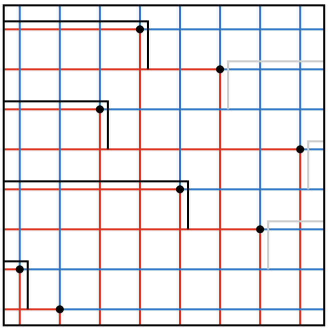
2. We simply draw permutation matrices. Our wires start from the left and bottom, and the ones of the permutation matrix are the places where wires can change from red to blue. Otherwise, colors must pass through. Because every wire must change from red to blue exactly once, this counts permutations. (Coloring changing points are marked for visibility, they are not part of the wiring.)



Recall that $n!$ gives a differentiably finite generating function that is not a diagonal of a rational (or algebraic).

3. We modify the previous example by introducing some new wires. If we think about $\sigma(i)$ as the dot in the ith row, we enforce $\sigma(1) < \sigma(2)$ with a black wire that starts on the left, goes down one square when hitting the dot, and must hit a red wire. Similarly, a light gray wire enforcing $\sigma(2) > \sigma(3)$ starts on the right and must hit a blue wire when going down.

   We note that some special care needs to be taken for the boundary tiles in this case. The left and right boundary tiles need to come in stacks of 2 to alternate the existence of wires. Also, the bottom-left and bottom-right corners need to have 1-stack and 2-stack options, in order to make both odd and even n work.



   Recall that the number of alternating permutations is not even differentiably finite.

4. Our last example is surprising because unlike the previous ones, there is no natural way to draw a connected graphs on n vertices in a $n \times n$ grid. (One could represent a graph by its adjacency matrix, but there is no obvious way to check that it is connected.) Instead, the technique is to recall the following recurrence for $c(n)$, the number of connected graphs on n vertices:

$$c(n) = \sum_{k=1}^{n} \binom{n-1}{k-1} (2^k - 1) c(k-1) c(n-k).$$

   The idea is to forget about graphs and focus on representing the equation directly. The full details are left to the reader. □

Part III

Related problems

10 Partitions and rim hooks

In this final part, we will see some miscellaneous topics that can be read at any point after the first few sections of Part I. Here, we turn our attention to labeled tilings called rim hook tableaux. These place numbers on top of tilings of regions that look like Young diagrams of partitions. Let us review some basic notions about partitions:

DEFINITION 10.1. A partition λ of n into k parts is a tuple $(\lambda_1, \ldots, \lambda_k)$ such that $\lambda_1 \geq \cdots \geq \lambda_k$ and $\lambda_1 + \cdots + \lambda_k = n$. The Young diagram of λ (in English notation) is constructed by stacking rows of length λ_i in left alignment. A standard Young tableau of λ is an assignment of $\{1, \ldots, n\}$ to the squares of the Young diagram such that the numbers increase along rows and columns. ⌟

For example, this is a standard Young tableau of $\lambda = (3, 3, 2, 1)$.

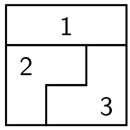
1	2	4	6
3	5	9	10
7	8		
11			

DEFINITION 10.2. A rim hook of size k is a tile with k squares and at most 1 square in each northwest-southeast diagonal (henceforth diagonal). The set of rim hooks of size k is denoted T_k. ⌟

Note that this notion of diagonal coincides with the diagonals we saw when coloring. For example, $T_2 = \{$, $\}$ is the set of dominoes and $T_3 = \{$, , , $\}$ is the set of tiles used in Conway–Lagarias' tiling of triangular regions. Notably, ,  $\notin T_3$. Also, because rim hooks are equivalently constructed by starting with a square and only adding squares to the top and right, $|T_k| = 2^{k-1}$.

DEFINITION 10.3. A k-rim hook diagram (k-RHD) is a partition λ that admits a tiling by T_k. Note that a k-RHDs must be a partition of kn for some n. A k-rim hook tableau (k-RHT) is a particular tiling of a k-RHD λ by T_k with labels $\{1, \dots, n\}$ on the rim hooks, increasing along every row and column. ⌟

For example, this is a 3-RHT of $\lambda = (3, 3, 3)$.



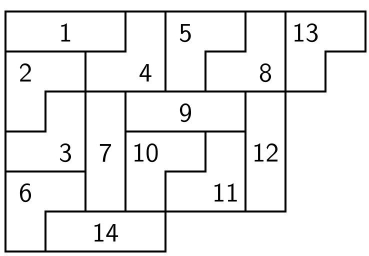
The primary goal of this section will be investigate the structure of k-RHDs and k-RHTs. Recall that Young diagrams are related by a poset structure, in which $\lambda < \mu$ if λ is visually a subset of μ. The poset forms a lattice, which is known as Young's lattice. The following structural result is due to Fomin and Stanton.

Theorem 10.4 ([FS97]). Let Y denote Young's lattice, and denote Y_k the subposet of Y consisting of k-RHDs. Then

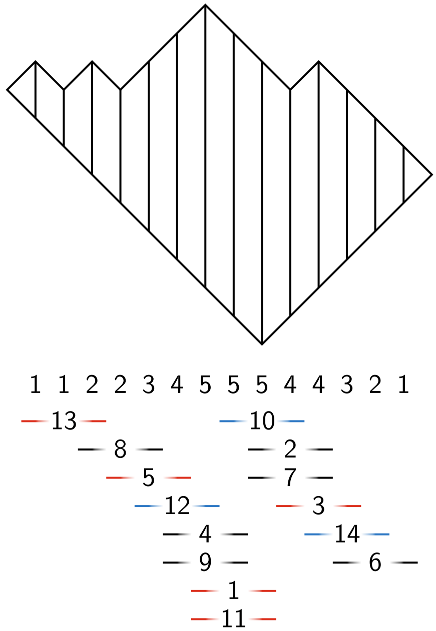
$$Y_k \cong Y^k.$$

Moreover, the respective labeled structures (k-RHTs and k-tuples of standard Young tableaux) are also in bijection. ⌟

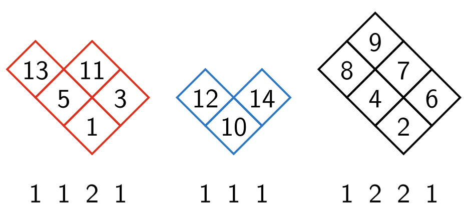
Before a formal proof, we will follow an example from [Pak00]. Consider the partition $\lambda = (9, 8, 7, 7, 7, 4)$. One possible 3-RHT of λ is shown below.



The bijection works as follows. First, rotate the figure 135° counter-clockwise, which is called Russian notation for partitions. Measure the lengths of all of the diagonals (now vertical lines). Then, because every 3-rim hook has 3 squares across 3 contiguous diagonals, we can draw each rim hook as a flattened line beneath the 3 diagonals that it contributes to. (If multiple rim hooks start on the same diagonal, for example 4 and 9, it doesn't matter which comes first.) They maintain their labels and get assigned 1 of 3 colors, depending on their starting diagonal mod 3.



Lastly, we consider each color separately, and collect the rim hooks that start on the same diagonal. Taking black rim hooks for example, there is at first 1 (8), then 2 (4, 9), then 2 more (2, 7), then 1 last one (6). Turning each rim hook into a square and drawing them in this order, with labels increasing from bottom to top, the diagonal length sequence we get is 1221, and the partition is as shown below. Note that there are actually 2 partitions corresponding to a diagonal length sequence of 1221, the one shown below and its mirror image. We always choose to set the main diagonal of the colored partitions along the squares whose corresponding rim hooks pass through the main diagonal in the original, here 2 and 7.



The bijection identifies the original k-RHT we showed with the three standard Young tableaux above, with labels appropriately changed while maintaining the order. It also identifies the k-RHD $\lambda = (9, 8, 7, 7, 7, 4)$ with the 3-tuple of partitions $((3, 2), (2, 1), (3, 3))$, if you forget about the labels.

Proof of Theorem 10.4. The proof is really just a list of definitions, and seeing that these definitions exactly characterize the structures we encountered in the example above.

Definition 10.5. Call $f : \mathbb{Z} \to \mathbb{Z}$ a fairy (short and magical) sequence if:

1. f is weakly increasing on $(-\infty, 0]$ and weakly decreasing on $[0, \infty)$.
2. $|f(i) - f(i+1)| \le 1$ for all $i \in \mathbb{Z}$.
3. $f(i) = 0$ for all i sufficiently large (in absolute value). ⌟

Fairy sequences are in natural bijection with diagonal length sequences of partitions, where $f(0)$ records the height of the main diagonal. We saw how to biject between these in both directions in the above example, so the proof details are left to the reader to verify.

Definition 10.6. Call $f : \mathbb{Z} \to \mathbb{Z}$ a k-fairy sequence if f is fairy and for all $0 \le a, b \le k-1$, we have $\sum_{i \equiv a \pmod k} f(i) = \sum_{i \equiv b \pmod k} f(i)$. ⌟

There is a natural bijection between k-fairy sequences and k-RHDs. Every k-RHD can be split into lines of length k as we did in the example, and every line contributes 1 to every sum mod k in the above definition, so the sums are the same. Conversely, given a k-fairy sequence, one constructs a T_k-tiling τ of λ inductively.

Next, given $f_0, \ldots, f_{k-1}$ fairy sequences, define

$$g(i) = \sum_{r=0}^{k-1} f_r \left(\left\lfloor \frac{i+r}{k} \right\rfloor \right).$$

This captures what we mean by setting the main diagonal in the colored partitions. We can check easily that g is a k-fairy sequence, and in fact that such g are in bijection with $f_0, \ldots, f_{k-1}$. □

Corollary 10.7. For any partition λ of n, the number of k-RHTs of λ can be computed in time polynomial in n. In particular, if λ is in bijection with partitions $\mu_1, \ldots, \mu_k$ of $m_1, \ldots, m_k$ respectively, then there are

$$\binom{n}{m_1, \ldots, m_k} \#\,\mathrm{SYT}(\mu_1) \cdots \#\,\mathrm{SYT}(\mu_k)$$

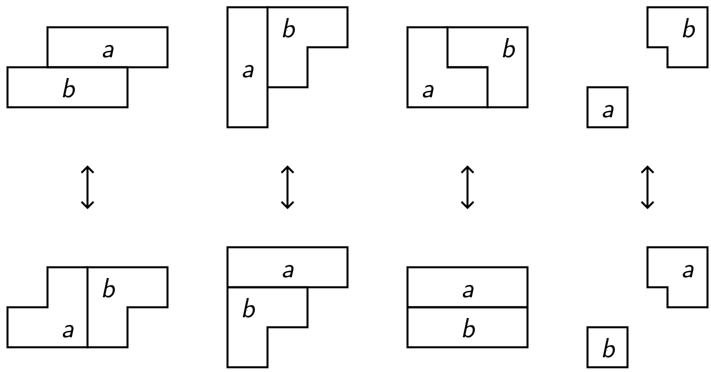
k-RHTs of λ, where $\binom{n}{m_1,\ldots,m_k} = \frac{n!}{m_1! \ldots m_k!}$ is a multinomial coefficient and $\#\,\mathrm{SYT}(\lambda)$ denotes the number of standard Young tableaux of λ. ⌟

The number of standard Young tableaux of λ can be computed cleanly and efficiently by the hook length formula (see Wikipedia), from which the corollary follows.

This bijection between rim hook tableaux and tuples of Young tableaux also has consequences for flip connectivity. We can prove the following:

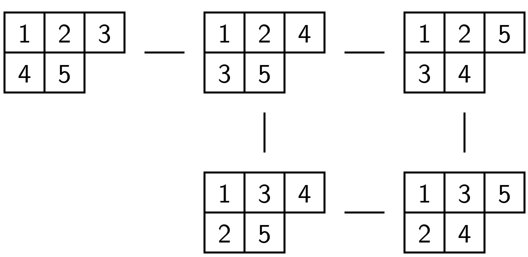
THEOREM 10.8. [Pak00] All k-RHTs (and consequently k-RHDs) are connected by flips involving 2 tiles (2-flips). ⌟

With $k = 2$, this is a special case of our previous result that domino tilings of simply connected regions are connected by flipping two horizontal tiles for two vertical ones. But even with $k = 3$, this is new and interesting. For 3-RHTs, the following (and their rotations) are all the possible types of flips (with $a < b$). (The last type shows labels changing on two tiles, without the tile shapes changing.)



Proof of Theorem 10.8. Instead of k-RHTs, we can equivalently talk about a tuple of standard Young tableaux $(\mu_1, \dots, \mu_k)$ by the bijection. Define a partial order P on the squares of $\mu_1 \cup \cdots \cup \mu_k$, increasing along rows and columns (squares of different partitions are incomparable).

Recall that for a poset $P = (X, <)$ with $|X| = n$, we say that $f : X \to [n]$ is a linear extension if f is a bijection and order-preserving. Define a graph $G(P) = (V, E)$ where V consists of all the linear extensions of P, and two linear extensions f and g are connected by edge if and only if $f(x) = g(x)$ for all but two $x \in X$. For example, if the shape is just the one partition $(3, 2)$, then $G(P)$ is the following graph.



It remains to show that $G(P)$ is connected, since the bijection would imply that the k-RHTs are connected by 2-flips. To do this, let f be the lexicographically smallest linear extension of P, and g' any other linear extension of P. Let g be the lexicographically smallest linear extension connected to g', and suppose for contradiction m is the smallest number for which f and g differ. But by the ordering, in g, it must be that m is to the left and below $m-1$. Flipping them produces a lexicographically smaller linear extension connected to g', a contradiction. □

Going back to the example where we showed all possible flips of 3-RHTs, one immediate corollary of this theorem is that difference in the number of the two triangular trominoes is constant for any fixed k-RHD. This gives a completely different proof of the same lemma that we used tiling groups for in an earlier section. But unlike tiling groups, this technique allows us to prove similar relations for higher k, giving a set of equations not unlike the Dehn-Somerville equations for simplicial polytopes. (As mentioend before, tiling groups become much smaller as the number of tiles increases, which reduces their usefulness.)

Theorem 10.9 ([Pak00]). Let T_k be the set of tiles and $N = |T_k|$. Let v_i, v_i' be the number of tiles t_i in tilings τ, τ' of a k-RHD, respectively. Then we have the following group isomorphism:

$$\mathbb{Z}^N / \langle (v_1, \ldots, v_N) - (v_1', \ldots, v_N') \rangle_{\tau,\tau'} \cong \begin{cases} \mathbb{Z}^{m+1} & \text{if } k = 2m+1 \\ \mathbb{Z}^m \times \mathbb{Z}_2 & \text{if } k = 2m. \end{cases}$$

In other words, when $k = 2m+1$, there are $m+1$ linearly independent equations relating the number of each tile, and when $k = 2m$, there are m linearly independent equations over $\mathbb{Z}$ and one more mod 2. ⌟

Note that one equation that is always satisfied is the area invariant: that the sum of the number of tiles is constant. When $k = 2$, for example, we get that the number of horizontal tiles is constant mod 2, and when $k = 3$, we additionally get the equation we just mentioned above.

Of course, all of these results are for k-RHDs, whereas the theorems they purportedly extend are true for all simply connected Γ. In later papers, both this result on linear relations between tile counts [MP02] and flip-connectivity of tilings [She02] were extended to all simply connected Γ.

11 Matchings

Recall that the problem of domino tilings in $\mathbb{Z}^2$ is a specific case of the problem of perfect matchings in general graphs. In this section, we will study matchings in general. Recall that an $O(m\sqrt{n})$ algorithm to decide if a general graph has a perfect matching is given by Micali and Vazirani in [MV80], but it is rather complicated. Instead, we will present some general techniques to work for a wider range of problems, being satisfied with any polynomial time algorithm or even randomized polynomial time algorithm. We give a definition to make clear what we mean by this.

DEFINITION 11.1. The complexity class BPP (bounded probabilistic polynomial) consists of all decision problems L for which there exists a polynomial time algorithm A satisfying:

1. If $x \in L$, then $\Pr_r[A(x,r) = 1] \geq \frac{2}{3}$.
2. If $x \notin L$, then $\Pr_r[A(x,r) = 0] \geq \frac{2}{3}$. ⌟

By repeating a BPP algorithm a polynomial number of times with independent random strings r and outputting the majority result, one can boost the success probability to 99.99% or even $1 - 2^{-n}$. Hence for all practical purposes, BPP is just as good as P, and the constant of $\frac{2}{3}$ in the definition is arbitrary (any constant strictly greater than $\frac{1}{2}$ will do). Some people actually believe that $\mathsf{BPP} = \mathsf{P}$.

PROPOSITION 11.2. Let $G = (V_1 \sqcup V_2, E)$ be a simple bipartite graph. Then there is a BPP algorithm to decide if G has a perfect matching. ⌟

Note that this will be a proof to illustrate basic ideas only. For bipartite graphs, determining the existence of a perfect matching has a fast, clean, and deterministic solution given by Hopcroft—Karp algorithm (see Wikipedia), which is not the solution we will present here.

Proof. First, only the case $|V_1| = |V_2|$ is interesting, there are no perfect matchings otherwise. So let $n = 2k$, where $k = |V_1| = |V_2|$. Consider the matrix $A_G = (a_{ij})$ over all $i \in V_1$ and $j \in V_2$ defined by

$$a_{ij} = \begin{cases} x_{ij} & \text{if } (i,j) \in E \\ 0 & \text{otherwise,} \end{cases}$$

introducing a variable x_{ij} for every edge. (When $x_{ij} = 1$ for all edges, this is called the biadjacency matrix.) A perfect matching is a generalized diagonal in A_G of non-zero entries. Recalling that

$$\det(A_G) = \sum_{\sigma \in S_k} \operatorname{sign}(\sigma) a_{1\sigma(1)} \cdots a_{k\sigma(k)},$$

it is clear that G has no perfect matching if and only if $\det(A_G) = 0$ (as an equality of polynomials over any field).

Although we can easily compute determinants of scalar-valued matrices in polynomial time (by Gaussian elimination or other methods), $\det(A_G)$ as a polynomial may have exponentially many terms. The key observation is that polynomials tend to have very few zeros, so if you randomly pick a point to evaluate $\det(A_G)$ on, chances are that if you only find zeros, then the polynomial is really identically zero. This strategy is formalized by the following extremely fundamental lemma.

LEMMA 11.3 (Schwartz–Zippel lemma). Let $Q(x_1, \dots, x_m) \in \mathbb{F}_q[x_1, \dots, x_m]$ be a non-zero polynomial and $\deg(Q) = d$. Pick $a_1, \dots, a_m \in \mathbb{F}_q$ uniformly at random. Then

$$\Pr[Q(a_1, \dots, a_m) = 0] \le \frac{d}{q}.$$ ⌟

Proof. By induction on m. When $m = 1$, the fact that $\Pr[Q(a_1) = 0] \le \frac{d}{q}$ is just the fundamental theorem of algebra. Now suppose it is true for $m-1$ and write

$$Q(x_1, \dots, x_m) = \sum_{i=1}^{d} Q_i(x_1, \dots, x_{m-1}) x_m^i.$$

Because Q is non-zero, we can consider the largest i for which Q_i is non-zero. Because Q_i is multiplied by x_m^i, it has degree at most $d - i$, so by induction

$$\Pr[Q_i(a_1, \dots, a_{m-1}) = 0] \le \frac{d-i}{q}.$$

In the event that $Q_i(a_1, \dots, a_{m-1}) \ne 0$, we note that because we chose the largest i, we get that $Q(a_1, \dots, a_{m-1}, x_m)$ is a degree i polynomial in x_m. By the base case, we get that

$$\Pr[Q(a_1, \dots, a_m) = 0 \mid Q_i(a_1, \dots, a_{m-1}) \ne 0] \le \frac{i}{q}.$$

Adding these together, the claim is proved. □

To finish the algorithm to decide if G has a perfect matching, note that every term in $\det(A_G)$ has degree k. Hence, find a prime p between $3k$ and $6k$ (one exists by Bertrand's postulate, and we can check the whole range for primes using any method like the Sieve of Eratosthenes) and work in the field $\mathbb{F}_p$. Pick $a_{ij} \in \mathbb{F}_p$ randomly for all edges (i,j) and compute $\det(A_G)$ on these inputs (for example, by Gaussian elimination). Output that a perfect matching does not exist if the determinant is zero, and that one exists otherwise.

In the case that there is no perfect matching, the determinant is always going to be zero, and we will always output correctly. If there is a perfect matching, by the Schwarz–Zippel lemma, we fail to detect it with probability at most $\frac{k}{3k} = \frac{1}{3}$, so the chance of success is $\frac{2}{3}$ and we are good. $\square$

The key ideas above were the matrix A_G and the Schwartz–Zippel lemma. Now, we will use extremely similar ideas to tackle the case of general graphs.

THEOREM 11.4. There exists a BPP algorithm to determine if a simple graph $G = (V, E)$ has a perfect matching. ⌟

Proof. We will still use the Schwartz–Zippel lemma, but the matrix A_G for bipartite graphs no longer makes sense for general graphs. Instead, consider the matrix $B_G = (b_{ij})$ over all $i, j \in V$, defined by

$$b_{ij} = \begin{cases} x_{ij} & \text{if } \{i,j\} \in E \text{ and } i < j \\ -x_{ji} & \text{if } \{i,j\} \in E \text{ and } i > j \\ 0 & \text{otherwise,} \end{cases}$$

introducing a variable x_{ij} for every edge, written with $i < j$. For example,

$$B_{K_4} = \begin{bmatrix} 0 & x_{12} & x_{13} & x_{14} \\ -x_{12} & 0 & x_{23} & x_{24} \\ -x_{13} & -x_{23} & 0 & x_{34} \\ -x_{14} & -x_{24} & -x_{34} & 0 \end{bmatrix}.$$

Note that B_G is a skew-symmetric matrix, that is, $B_G = -B_G^T$. With a skew-symmetric matrix, one can consider the Pfaffian of the matrix, which has nice properties. Before the definition, let us set some common notation for permutations: for the permutation $\left(\begin{smallmatrix} 1 & 2 & 3 \\ 2 & 1 & 3 \end{smallmatrix}\right)$, we may write it as 213 in one-line notation (without parentheses), or $(12)(3) = (12)$ in cycle notation.

DEFINITION 11.5. Let $B = (b_{ij}) \in \mathbb{F}^{n \times n}$ be a skew-symmetric matrix, and consider perfect matchings M of $\{1, \dots, n\}$, written as $\{(i_1, j_1), \dots, (i_k, j_k)\}$ satisfying $i_r < j_r$ for all r.

Let $\operatorname{sign}(M) = \operatorname{sign}(i_1 j_1 i_2 j_2 \dots j_k)$. (To emphasize, this is a permutation in one-line notation.) This is well-defined because although the edges can be written in any order, it takes two transpositions in the permutation (one for i and one for j) to compensate for any swap in the edge order, and sign does not change for an even number of transpositions.

Let $b_M = b_{i_1 j_1} \cdots b_{i_k j_k}$. The Pfaffian of B is then defined as the quantity

$$\mathrm{pf}(B) = \sum_M \mathrm{sign}(M) b_M. \qquad \lrcorner$$

As an example, let's compute the Pfaffian of B_{K_4}. There are 3 perfect matchings of $\{1,2,3,4\}$: $\{(1,2),(3,4)\}$, $\{(1,3),(2,4)\}$, and $\{(1,4),(2,3)\}$. The permutations 1234, 1324, and 1423 have signs 1, -1, and 1 respectively. Hence $\mathrm{pf}(B_{K_4}) = x_{12}x_{34} - x_{13}x_{24} + x_{14}x_{23}$.

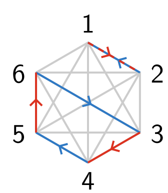
The Pfaffian's most useful property is the following. The proof technique is an important idea that we will use again.

Proposition 11.6. Let B be skew-symmetric. Then $\det(B) = \mathrm{pf}(B)^2$. $\lrcorner$

Proof. We start with a lemma.

Lemma 11.7. The set $\Omega_n = \{\sigma \in S_n : \sigma \text{ has only even cycles}\}$ is in bijection with ordered pairs of perfect matchings of $\{1,\ldots,n\}$. $\lrcorner$

Proof. Clearly both sets are empty when n is odd, so assume that n is even. Consider $\{1,\ldots,n\}$ as the vertices of K_n. Given two perfect matchings M, M' of K_n, consider $M \cup M'$ (as the union of subgraphs). Every vertex is incident to one edge from M and one edge from M', so the graph $M \cup M'$ consists of disjoint even cycles, each of whose edges alternate between M and M'. Orient every cycle so that the edge leaving the smallest vertex in each cycle is from M. An example is shown below (M is red, M' is blue), producing (12)(3456).



Conversely, given a permutation σ, break it into cycles and alternate the edges between M and M', starting with M for the edge leaving the smallest element of the cycle. This exactly recovers the two matchings above. $\square$

Denote $\det(B) = \sum_{\sigma \in S_n} b_\sigma$ where $b_\sigma = \mathrm{sign}(\sigma) b_{1\sigma(1)} \cdots b_{n\sigma(n)}$. We can simplify this sum with the following two observations.

1. Suppose σ has a fixed point $\sigma(i) = i$. Because B is skew-symmetric, its diagonal is zero, so $b_\sigma = 0$.
2. Suppose σ has an odd cycle (but no fixed points). Fix an order on all odd cycles in S_n such that every cycle is adjacent to its inverse, and let σ^* be σ with the smallest odd cycle reversed. Recall that reversing a cycle does not change the sign of the permutation. However,

reversing an odd cycle flips the indices of an odd number of $b_{i\sigma(i)}$ into $b_{\sigma(i)i} = b_{\sigma(i)\sigma^*(\sigma(i))}$. Because B is skew-symmetric, this picks up an odd number of negative signs, so $b_\sigma = -b_{\sigma^*}$. Because $\sigma^{**} = \sigma$, we have paired up all such σ to cancel each other out.

We are left summing over σ made of even cycles only. From the lemma, denote ϕ the bijection from pairs of perfect matchings of $\{1, \ldots, n\}$ to permutations of only even cycles. We get that

$$\begin{aligned}\det(B) &= \sum_{M,M'} \operatorname{sign}(\phi(M, M'))b_{1\phi(M,M')(1)} \cdots b_{n\phi(M,M')(n)} \\ &= \sum_{M,M'} \operatorname{sign}(M)\operatorname{sign}(M')b_M b_{M'} \\ &= \operatorname{pf}(B)^2.\end{aligned}$$

The fact that $b_{1\phi(M,M')(1)} \cdots b_{n\phi(M,M')(n)} = b_M b_{M'}$ is just a reordering of terms, by a quick definition check. To show that $\operatorname{sign}(\phi(M, M')) = \operatorname{sign}(M)\operatorname{sign}(M')$, just note that both are equal to $(-1)^{n-c} = (-1)^c$, where c is the number of cycles in $\phi(M, M')$. □

To finish the algorithm to determine if a graph has a perfect matching, we use the Schwarz–Zippel lemma again to test if $\det(B_G) = 0$. Because $\det(B_G) = \operatorname{pf}(B_G)^2$, this is also a test if $\operatorname{pf}(B_G) = 0$. But looking at the definition of the Pfaffian, we pick up a term exactly when a perfect matching M of K_n has all of its edges in E, i.e. G has a perfect matching, so we are done. □

As one more application of these ideas, we will apply them to the related problem of exact matchings. Given $G = (V, E)$, a subset of edges $R \subset E$, and an integer $0 \le r \le n$, the problem is to decide if there exists a perfect matching $M \subset E$ such that $|M \cap R| = r$.

Theorem 11.8. There is a BPP algorithm to decide if an exact matching exists. ⌟

Proof. Define the matrix $B_{G,R} = (b_{ij})$, where

$$b_{ij} = \begin{cases} x_{ij} & \text{if } \{i,j\} \in E \setminus R \text{ and } i < j \\ x_{ij}z & \text{if } \{i,j\} \in R \text{ and } i < j \\ -x_{ji} & \text{if } \{i,j\} \in E \setminus R \text{ and } i > j \\ -x_{ji}z & \text{if } \{i,j\} \in R \text{ and } i > j \\ 0 & \text{otherwise.} \end{cases}$$

Note that $B_{G,R}$ is skew-symmetric. Recalling the definition of Pfaffian, every term of the Pfaffian corresponds to a perfect matching of G, and the exponent of z in each term corresponds to the number of edges in R. Hence if we write $\operatorname{pf}(B_{G,R}) = \sum_{i=0}^{k} p_i(x) z^i$, we are interested to know if $p_r(x) = 0$. Our goal is to do polynomial identity testing via the Schwartz–Zippel lemma on $p_r(x)$, so it remains to show how to compute $p_r(x)$ efficiently given a random input of x_{ij}'s.

Fix inputs x_{ij}. We can write $\det(B_{G,R}) = \sum_{i=0}^{n} q_i(x) z^i$. As a polynomial in z, it has degree at most n, and hence by testing $n+1$ values for z along with the fixed x_{ij}'s, we can use Lagrange interpolation to compute the coefficients of $\det(B_{G,R})$ as a polynomial in z. Taking the square root of this polynomial, we get the coefficients of $\operatorname{pf}(B_{G,R})$ as a polynomial in z, one of which is just $p_r(x)$. □

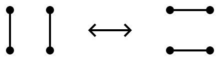
Notably, this is one of the few problems for which, as of writing, a BPP algorithm is known, but no deterministic algorithm is known. Another one is polynomial identity testing: whether a polynomial $p(x)$ is identically zero. A major problem that used to belong to this category was testing if an integer is prime, but a deterministic algorithm for that was found relatively recently: the AKS primality test.

Next, just as we counted tilings, we now tackle the problem of counting matchings. Before tackling the problem in generality, we will see the solution for domino tilings. This was originally given by Kasteleyn, who studied the problem for its relevance in crystal physics, where domino tilings are referred to as the dimer model.

Theorem 11.9 ([Kas61]). *Let $G \subset \mathbb{Z}^2$ be a simply connected subgraph. There is a polynomial time algorithm to compute the number of perfect matchings in G.* ⌟

Proof. Give horizontal edges weight $x_e = 1$, give vertical edges weight $x_e = i = \sqrt{-1}$, and let B be the (weighted, skew-symmetric) adjacency matrix of G. We claim that $|\operatorname{pf}(B)|$ is the number of perfect matchings of G. Because the Pfaffian is related to the determinant, we get a polynomial time algorithm for counting the number of perfect matchings.

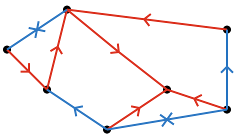
Looking at the definition of Pfaffian, it suffices to show that under this choice of weights, $\operatorname{sign}(M) b_M$ is constant for all M. Because we know that domino tilings for simply connected regions are connected by flips, it suffices to show that $\operatorname{sign}(M) b_M$ is preserved by a flip.



Note that after a flip, $\operatorname{sign}(M)$ is negated, because the permutation $i_1 j_1 \cdots i_k j_k$ undergoes a single transposition. At the same time, b_M is also negated, because two horizontal edges (product 1) are exchanged for two vertical edges (product -1). □

The key idea in this was that the product of edge weights around a square is -1. With a bit more work, one can make the above technique work for non-simply connected regions. In fact, we can generalize even further to all planar graphs G. The result is known as the FKT (Fisher–Kasteleyn–Temperley) algorithm, given in [Kas67].

To briefly outline the proof, note that a choice of weights $x_{ij} \in \{\pm 1\}$ for the edges of a graph G can be thought of as orienting every edge. For planar G, call such a choice a Pfaffian orientation if all faces with an even number of edges, an odd number of edges are pointed clockwise. We first notice that every planar graph has a Pfaffian orientation. Fix a spanning tree T of G and an arbitrary orientation (below in red). Then, it can be shown that the remaining edges can be efficiently oriented (below in blue, the edges marked with $\times$ can be either direction).



Lastly, it can be shown that such a choice of weights makes $\operatorname{sign}(M)b_M$ have the same sign for all matchings M. A polynomial time algorithm for computing $|\operatorname{pf}(B)|$, which is exactly the number of perfect matchings, follows again from determinant algorithms.

How about non-planar graphs? The problem turns out to be hard to compute exactly, but we have some approximation results due to Chein.

Theorem 11.10 ([Chi04]). Let $G = (V, E)$ be a simple graph and assign edge weights $x_{ij} \in \{\pm 1\}$ uniformly at random. Then the number of perfect matchings of G is $\mathbb{E}[\det B]$, where B is as usual. ⌟

Proof. We'll use a casework technique that we seen before. Write $\det(B) = \sum_{\sigma \in S_n} b_\sigma$ where $b_\sigma = \operatorname{sign}(\sigma) b_{1\sigma(1)} \cdots b_{n\sigma(n)}$. By linearity of expectation, we can consider the contributions of each b_σ. Notice the following.

1. Suppose σ has a fixed point $\sigma(i) = i$. Because B is skew-symmetric, $b_\sigma = 0$.
2. Suppose $\sigma^2 \neq 1$. Then the product b_σ contains some b_{ij} but not b_{ji}. The choice of $b_{ij} \in \{\pm 1\}$ was independent of all other edges, so we have $\mathbb{E}[b_\sigma] = \mathbb{E}[b_{ij} \cdots] = \mathbb{E}[b_{ij}]\mathbb{E}[\cdots] = 0$.

We are left with σ such that $\sigma^2 = 1$ and there are no fixed points. These are exactly the σ that represent perfect matchings on $\{1, \ldots, n\}$, each cycle in σ being a pair. Additionally, $b_{1\sigma(1)} \cdots b_{n\sigma(n)} = (-1)^{n/2} = \operatorname{sign}(\sigma)$ (by counting the number of transpositions), so $b_\sigma = 1$ in this case. □

By itself, this simple result allows weak randomized approximation using Markov's inequality. The original paper proves some more bounds on the variance for a slightly tighter result.

Bibliography

[AT08] Arseniy V Akopyan and Aleksey S Tarasov. A constructive proof of kirszbraun's theorem. *Mathematical Notes*, 84(5):725–728, 2008.

[Ber66] Robert Berger. *The undecidability of the domino problem*. Number 66. American Mathematical Soc., 1966.

[BNRR95] Daniele Beauquier, Maurice Nivat, Eric Remila, and Mike Robson. Tiling figures of the plane with two bars. *Computational Geometry*, 5(1):1–25, 1995.

[Chi04] Steve Chien. A determinant-based algorithm for counting matchings in a general graph. In *Proceedings of the fifteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 728–735. Citeseer, 2004.

[CL90] J.H Conway and J.C Lagarias. Tiling with polyominoes and combinatorial group theory. *Journal of Combinatorial Theory, Series A*, 53(2):183–208, 1990.

[dBK75] NG de Bruijn and DA Klarner. A finite basis theorem for packing boxes with bricks. *Philips Res. Rep*, 30:337–343, 1975.

[FGS05] Philippe Flajolet, Stefan Gerhold, and Bruno Salvy. On the non-holonomic character of logarithms, powers, and the n-th prime function. *The Electronic Journal of Combinatorics [electronic only]*, 12, 2005.

[FS97] Sergei Vladimirovich Fomin and Dennis W Stanton. Rim hook lattices. *Algebra i Analiz*, 9(5):140–150, 1997.

[Fur67] Harry Furstenberg. Algebraic functions over finite fields. *Journal of Algebra*, 7(2):271–277, 1967.

[Gon85] Teofilo F Gonzalez. Clustering to minimize the maximum intercluster distance. *Theoretical computer science*, 38:293–306, 1985.

[GP14a] Scott Garrabrant and Igor Pak. Counting with irrational tiles. *arXiv preprint arXiv:1407.8222*, 2014.

[GP14b] Scott Garrabrant and Igor Pak. Counting with wang tiles, 2014.

[JR15] Emmanuel Jeandel and Michael Rao. An aperiodic set of 11 wang tiles. *arXiv preprint arXiv:1506.06492*, 2015.

[Kas61] Pieter W Kasteleyn. The statistics of dimers on a lattice: I. the number of dimer arrangements on a quadratic lattice. *Physica*, 27(12):1209–1225, 1961.

[Kas67] Pieter Kasteleyn. Graph theory and crystal physics. *Graph theory and theoretical physics*, pages 43–110, 1967.

[Kir34] Mojzesz Kirszbraun. Über die zusammenziehende und lipschitzsche transformationen. *Fundamenta Mathematicae*, 22(1):77–108, 1934.

[KK92] C. Kenyon and R. Kenyon. Tiling a polygon with rectangles. In *Proceedings of the 33rd Annual Symposium on Foundations of Computer Science*, SFCS '92, page 610–619, USA, 1992. IEEE Computer Society.

[Kor04] Michael Robert Korn. *Geometric and algebraic properties of polyomino tilings*. PhD thesis, Massachusetts Institute of Technology, 2004.

[KP04] Michael Korn and Igor Pak. Tilings of rectangles with t-tetrominoes. *Theoretical computer science*, 319(1-3):3–27, 2004.

[LMP05] T Lam, E Miller, and I Pak. Tiling rectangles with rectangles. unpublished manuscript, 2005.

[MP99] Roman Muchnik and Igor Pak. On tilings by ribbon tetrominoes. *J. Comb. Theory, Ser. A*, 88:188–193, 1999.

[MP02] Cristopher Moore and Igor Pak. Ribbon tile invariants from the signed area. *Journal of Combinatorial Theory, Series A*, 98(1):1–16, 2002.

[MR01] Cristopher Moore and John Michael Robson. Hard tiling problems with simple tiles. *Discrete & Computational Geometry*, 26(4):573–590, 2001.

[MSV00] Donatella Merlini, Renzo Sprugnoli, and M Cecilia Verri. Strip tiling and regular grammars. *Theoretical Computer Science*, 242(1-2):109–124, 2000.

[MV80] Silvio Micali and Vijay V Vazirani. An o(v—v— c —e—) algoithm for finding maximum matching in general graphs. In *21st Annual Symposium on Foundations of Computer Science (sfcs 1980)*, pages 17–27. IEEE, 1980.

[Pak00] Igor Pak. Ribbon tile invariants. *Transactions of the American Mathematical Society*, 352(12):5525–5561, 2000.

[PST16] Igor Pak, Adam Sheffer, and Martin Tassy. Fast domino tileability. *Discrete & Computational Geometry*, 56:377–394, 2016.

[PY13a] Igor Pak and Jed Yang. The complexity of generalized domino tilings. *arXiv preprint arXiv:1305.2154*, 2013.

[PY13b] Igor Pak and Jed Yang. Tiling simply connected regions with rectangles. *Journal of Combinatorial Theory, Series A*, 120(7):1804–1816, 2013.

[PZ17] Hugo Parlier and Samuel Zappa. Distances in domino flip graphs. *The American Mathematical Monthly*, 124:710 – 722, 2017.

[Rei05] Michael Reid. Klarner systems and tiling boxes with polyominoes. *Journal of Combinatorial Theory, Series A*, 111(1):89–105, 2005.

[Rém05] Eric Rémila. Tiling a polygon with two kinds of rectangles. *Discrete & Computational Geometry*, 34(2):313–330, 2005.

[SF99] Richard P. Stanley and Sergey Fomin. *Enumerative Combinatorics*, volume 2 of *Cambridge Studies in Advanced Mathematics*. Cambridge University Press, 1999.

[She02] Scott Sheffield. Ribbon tilings and multidimensional height functions. *Transactions of the American Mathematical Society*, 354(12):4789–4813, 2002.

[Sta11] Richard P. Stanley. *Enumerative Combinatorics*, volume 1 of *Cambridge Studies in Advanced Mathematics*. Cambridge University Press, 2 edition, 2011.

[Tas14] Martin Tassy. *Tiling by bars*. PhD thesis, Ph. D. thesis, Brown University, 2014.

[Thu90] William P Thurston. Conway's tiling groups. *The American Mathematical Monthly*, 97(8):757–773, 1990.

[Val79] Leslie G Valiant. Completeness classes in algebra. In *Proceedings of the eleventh annual ACM symposium on Theory of computing*, pages 249–261, 1979.

[Wal65] DW Walkup. Covering a rectangle with t-tetrominoes. *The American Mathematical Monthly*, 72(9):986–988, 1965.

[Yan13] Jed Chang-Chun Yang. *Computational complexity and decidability of tileability*. University of California, Los Angeles, 2013.