José Ezequiel Soto Sánchez

# On Periodic Tilings with Regular Polygons

AUGUST 11, 2020

impa
Instituto de Matemática Pura e Aplicada

Visgraf Vision and Graphics Laboratory

PHD THESIS, IMPA - VISGRAF LAB

ADVISOR: Luiz Henrique de Figueiredo

CO-ADVISOR: Asla Medeiros e Sá

chequesoto.info

*Con amor y gratitud a...*

*Antonio Soto* †

*Paty Sánchez*

*Alma y Marco*

# *Abstract*


## On Periodic Tilings with Regular Polygons

by José Ezequiel Soto Sánchez

Periodic tilings of regular polygons have been present in history for a very long time: squares and triangles tessellate the plane in a known simple way, tiles and mosaics surround us, hexagons appear in honeycombs and graphene structures. The oldest registry of a systematic study of tilings of the plane with regular polygons is Kepler's book *Harmonices Mundi*, published 400 years ago.

In this thesis, we describe a simple integer-based representation for periodic tilings of regular polygons using complex numbers. This representation allowed us to acquire geometrical models from two large collections of images – which constituted the state of the art in the subject –, to synthesize new images of the tilings at any scale with arbitrary precision, and to recognize symmetries and classify each tiling in its wallpaper group as well as in its $n$-uniform $k$-Archimedean class.

In this work, we solve the age old problem of characterizing all triangle and square tilings (Sommerville, 1905), and we set the foundations for the enumeration of all periodic tilings with regular polygons. An algebraic structure for families of triangle-square tilings arises from their representation via equivalence with edge-labeled hexagonal graphs. The set of tilings whose edge-labeled hexagonal dual graph is embedded in the same flat torus is closed by positive-integer linear combinations. We compute Hilbert basis for families of tilings in each topological setting. The bases provide the enumeration of the infinite families of tilings spanned by them. Since tilings of triangles and squares contain all other tilings by regular polygons (with exactly one exception), we set the grounds for the enumeration of all periodic tilings with regular polygons. We use generators in the bases to create a sample set of more than 100 million triangle-square tilings, and we describe their general properties and some asymptotic behaviors. Additionally, we show an interpretation of the algebraic structure of triangle-square tilings as origami foldings.



*Keywords:* tilings, regular polygon, enumeration, symmetry, computer graphics

# *Contents*

# *Preface*

*Symmetry is ubiquitous in the Natural world. It occurs in all scales, from particle physics to chemistry and cosmology. Mathematics describing symmetry in its various forms is therefore essential for our understanding of nature.* [1]

[1] Ranscombe, Peter. *Geometry: a secret weapon in the fight against viruses*. The Lancet Infectious Diseases, 15(7):773, 2015
URL https://doi.org/10.1016/s1473-3099(15)00106-1

Periodic tilings of regular polygons have been studied for a long time: Archimedes explored uniform tilings of the plane and of the sphere, formulating the famous Archimedean solids. The first attempt at a systematic study of tilings of the plane with regular polygons appears in Kepler's book *Harmonices Mundi* published 400 years ago (Figure 1). Squares and triangles tessellate the plane in a known simple way, tiles and mosaics surround us throughout history, and hexagons appear in honeycombs and graphene structures. Although there is much literature on tilings, and knowledge around the subject has evolved in many ways in mathematics, it is still a very active area of research.

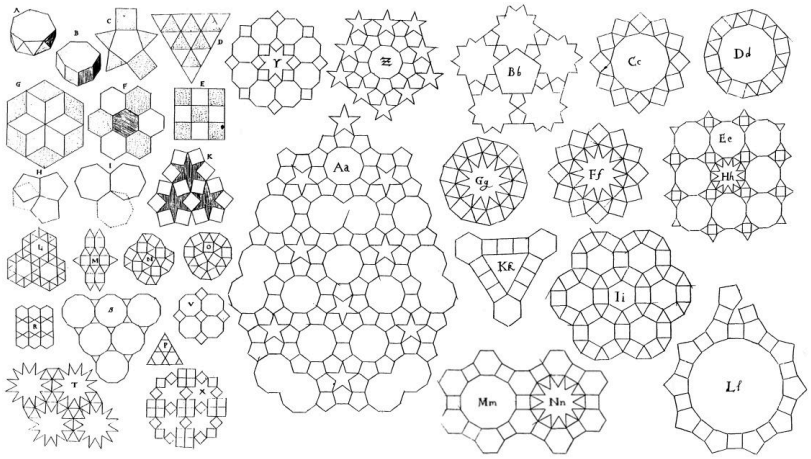



Figure 1: Tilings with regular polygons in *Harmonices Mundi*, Johannes Kepler, 1619 [47].

I have dedicated my research efforts specifically to periodic tilings of the plane with regular polygons and advanced the state of the art in the matter. I describe a novel way to represent periodic tilings of the plane with regular polygons using complex numbers and use our representation for acquiring and modeling existing collections of tilings. I also show the equivalence of a dense version of the tilings with a geometrically constrained graph combinatorial description, which allows the development of an algebraic structure to describe all periodic triangle-square tilings. From this algebraic structure, enumeration and generation strategies are derived.

## *Motivation*

Solid things in nature are frequently built by blocks: atoms gather together to form crystals (Figure 2), cells join tightly to create all sorts of tissues (Figure 3), bees and wasps build tight clusters for their hives (Figure 4) – some of the many natural structures from which humans have learned their ways of building.

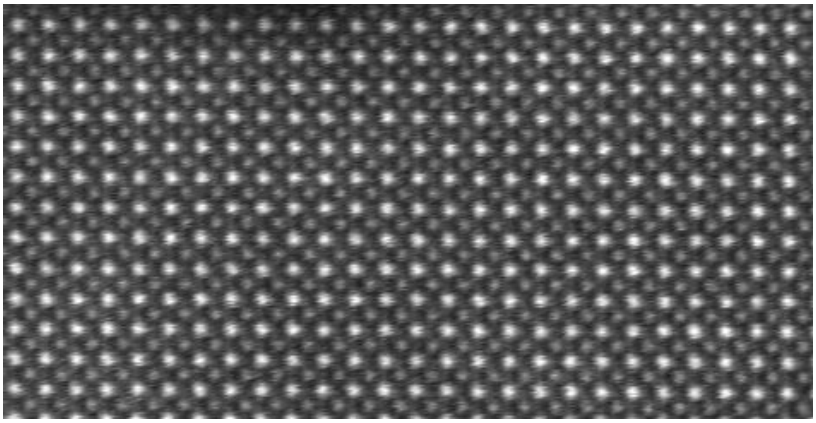


Figure 2: Atomic resolution STEM-HAADF image of $SrTiO_3$. Brighter atoms are Sr and darker ones are Ti. By *Materialist*, CC BY-SA 3.0

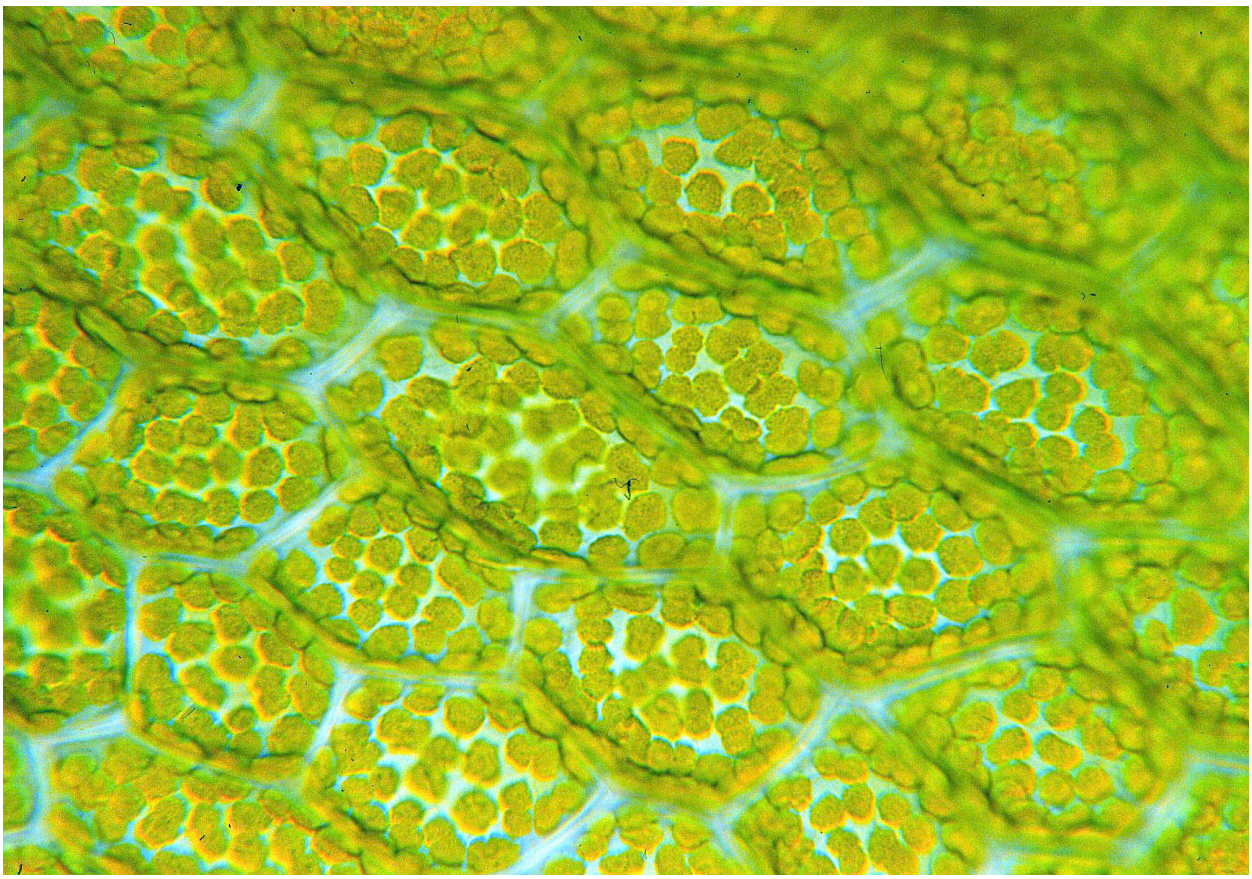


Figure 3: Chloroplasts in leaf cells of the moss *Mnium stellare*. By Thomas Geier, CC BY-SA 3.0

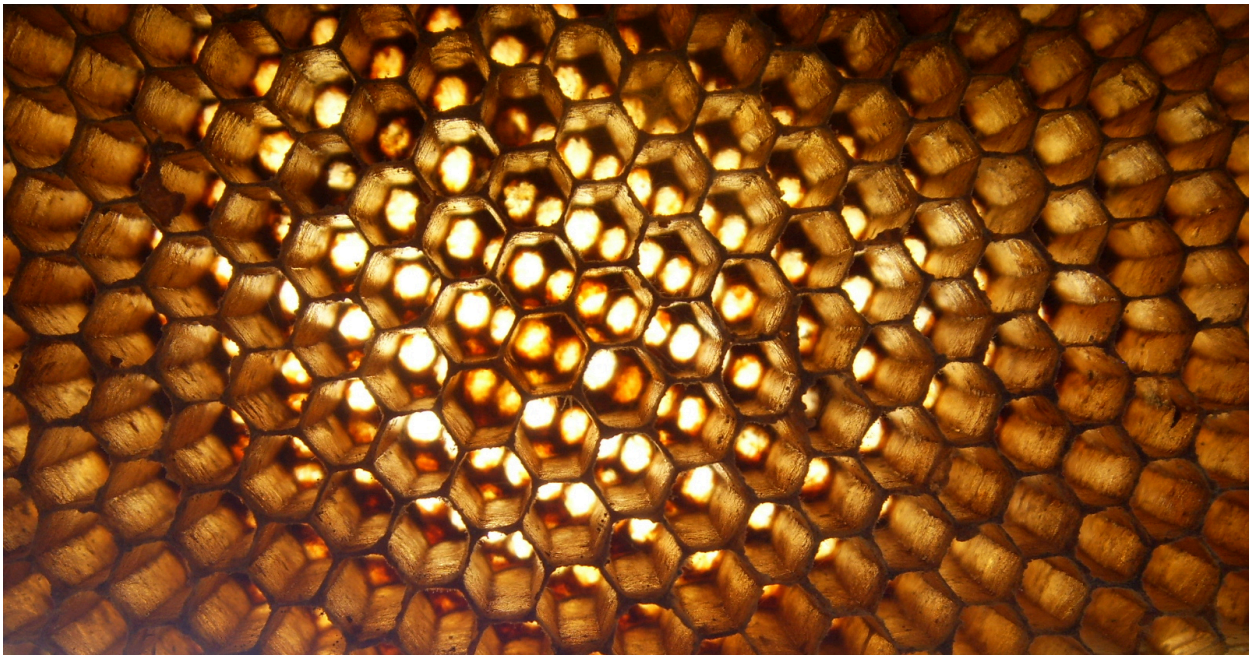


Figure 4: *Apis florea* nest closeup. By Sean Hoyland, Public Domain.

Block building took on monumental proportions in many different human cultures around the world. Tightly arranged stones support and ornament antique pyramids and temples, many still standing today. We still construct and embellish our buildings with clay, concrete, wood, plastic or metal blocks and plates; which are tiled, stacked, and arranged in many different ways.

Ornaments have contained symmetries and patterns since early humans developed them. Weaved baskets and clothing, present at least from the Palaeolithic, are good examples (Figure 5). These cultural developments are considered part of the history of mathematics [90, 91] and, to present day, offer evidence of culturally situated mathematical knowledge from an *ethnomathematics* perspective [1, 12, 30, 61].

Figure 5: Seri basket (*haat hanóohcö*). By Steve Marlett, CC BY-SA 3.0

The block building strategy is used in ornamentation in the form of tilings and tessellations[2]. Antique temples and buildings display colorful and intricate designs, carefully assembled with small pieces by skilled artisans.

One confirms the timeless human interest in tilings with their ubiquity: from the ancient Sumerian mosaics of Uruk IV (Figure 6, around 3000 BC) and Egyptian ornamentation, passing by the Roman style mosaics (Figure 7) and the Mesoamerican Zapotec friezes (Figure 8 and 9), we get to the staggering complexity of the Islamic geometric patterns (Figure 10 and 11), and even to present-day home mosaics, where common square and rectangular shaped tiles are no longer the only options (Figure 12).

[2] **Tessellation** derives from Latin ***tessera*** (plural: ***tesserae***, diminutive ***tessella***): an individual tile, usually formed in the shape of a cube, used in creating a mosaic.

Figure 6: Mosaics from the Stone-Cone Temple in the Eanna District of Uruk IV (around 3000 BC), on display at the Pergamon Museum in Berlin, Germany. By BrokenSphere / Wikimedia Commons.

Figure 7: An ancient roman geometric mosaic from the "Palazzo Massimo alle Terme" National Roman Museum of Rome, Italy. By M. Bellaccini / CC BY-SA 4.0

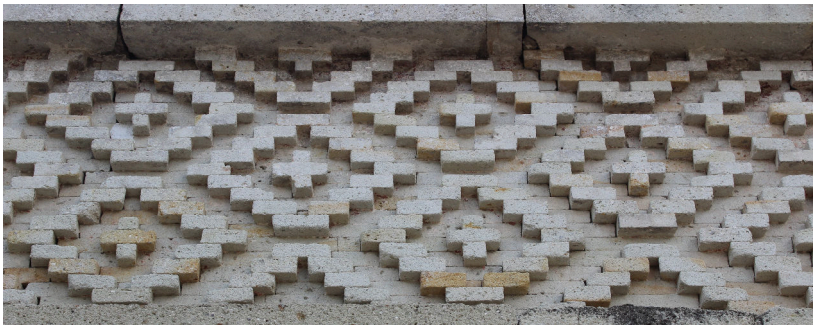


Figure 8: Ornamental tilings in the archaeological site of Mitla, Zapotec culture, Oaxaca, México. By J. Ezequiel Soto S.

Figure 9: Ornamental tilings in the archaeological site of Mitla, Zapotec culture, Oaxaca, México. By J. Ezequiel Soto S.

Figure 10: *Zellige* tessellation work in the Alhambra (Figure 13), with classic motif colours of the 14th century. By Patrick Gruban, CC BY-SA 2.0

Figure 11: *Zellige* work in a fountain in El Hedim Square (1672-1674), Meknès, Morocco. By Fabos, Public Domain.

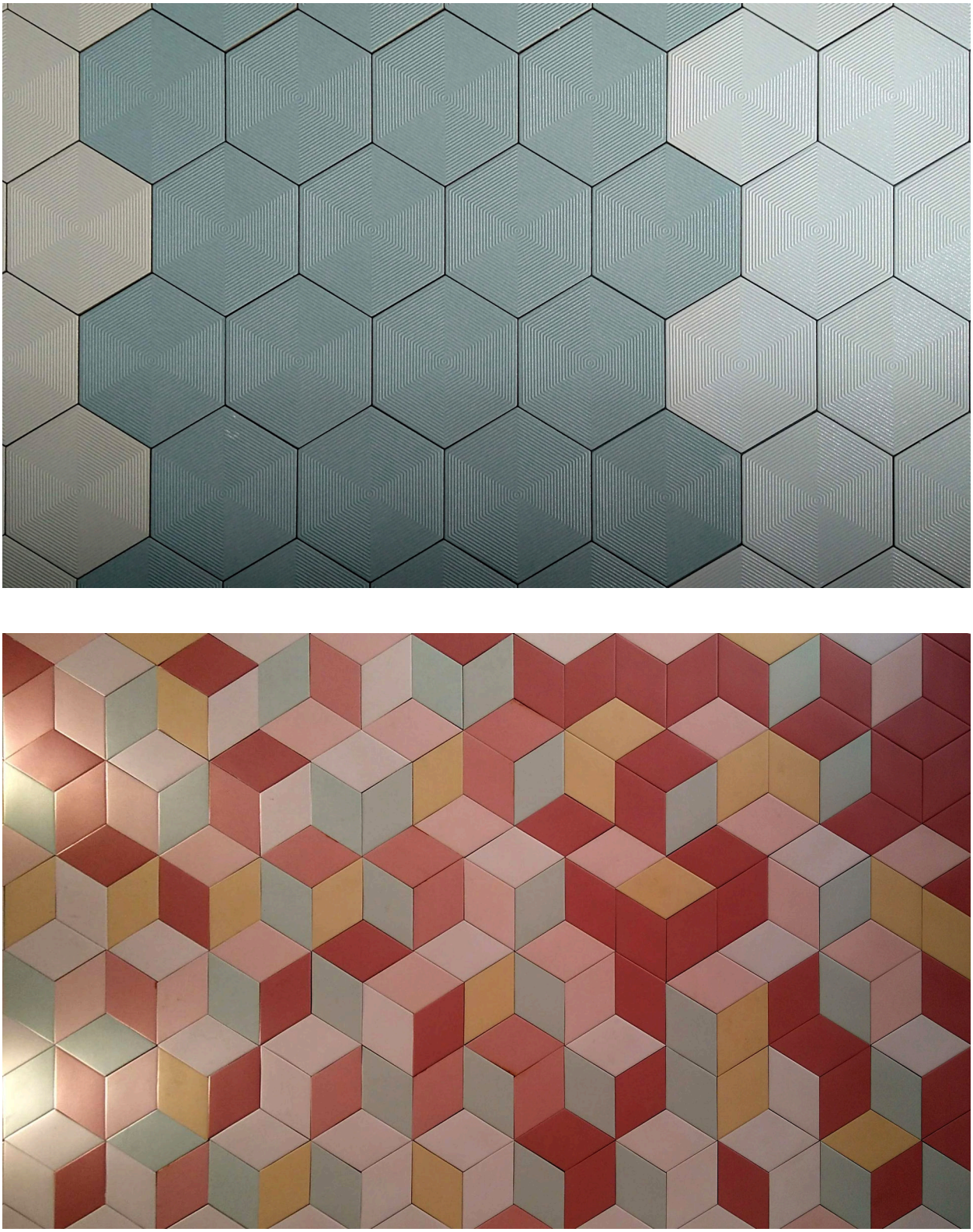


Figure 12: Ceramic tiles in a construction store, Rio de Janeiro, Brazil. By J. Ezequiel Soto S.

Islamic art took the complexity and diversity of geometric patterns further than any other culture. We can speculate that this is due to the religious ban on representing living things, their known mathematical and scientific development, and the time span of almost eight centuries of dominance of the continent and evolving sciences and arts. Nevertheless, their geometric patterns are dazzling, and exemplars of their art spread all the way from Spain and North Africa to the Middle East.

Islamic art has been studied and cataloged by many historians and artists [89], and a lot of work has been done trying to unveil its mathematical principles and properties [12]. The work of M.C. Escher (1898–1972) was profoundly influenced by his visit to the Alhambra [3] in the early 1920s. His notebooks contain copies of some of Alhambra tessellations (Figure 14) and his fascination for symmetry and tilings was declared by himself as a mania:

> I have often wondered at my own mania of making periodic drawings. What can be the reason of my being alone in this field? Why do none of my fellow-artists seem to be fascinated as I am by the interlocking shapes? – M.C.Escher *apud.* [51]

[3] **Alhambra** is one of the main monuments of the Islamic rule over the Iberian Peninsula. The compound of palaces, gardens and fortress, located in Granada, Spain, is a World Heritage Site visited by millions every year.

Figure 13: Alhambra in the evening. By Jebulon, Public Domain.

Escher already had a taste for science and his own research on tessellations (Figure 15) brought him to crystallography, symmetry, and group theory. His legacy is remarkable and highly esteemed by many mathematicians; it includes 137 regular divisions of space among many other fascinating artworks [79].

Tilings and tessellations form a vast and aesthetically appealing area of intersection between art and mathematics, one in which this work was born and to which it contributes.

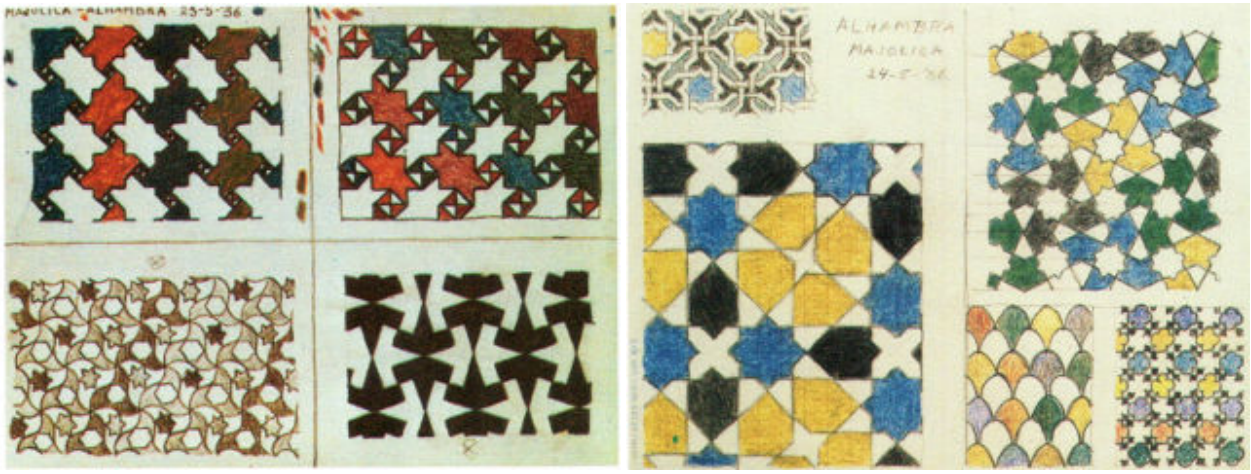



Figure 14: Escher's drawings in his notebook while visiting Alhambra. [79]

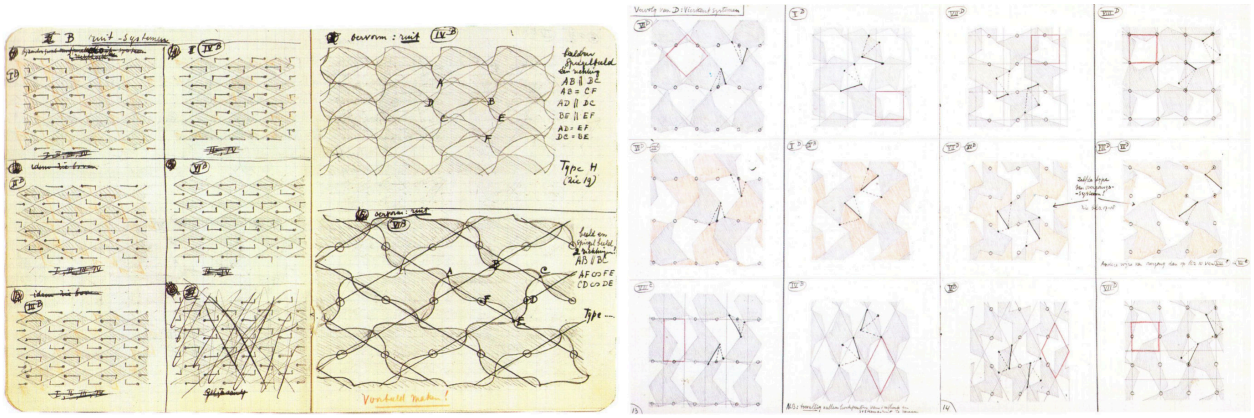


Figure 15: Escher's explorations involving plane symmetry groups. [79]

## *How the journey started*

During the Mathematics Festival, organized by IMPA in April 2017, I was conducting a workshop on tessellations and mandalas, which included an introduction to Escher's work and plane symmetries. I met Asla Medeiros E Sá in the next corridor, where she was taking care of her university stand. The timing was perfect: at the time, she and her father, Ricardo Sá, were finishing their book *Sobre Malhas Arquimedianas* [75], a crafted collection of tilings.

Our collaboration started with some discussions about the content of the book, trying to understand the symmetries in the patterns, and how to present them in the book. This also motivated some research ideas about Archimedean tilings: How many are there? Is there some way to automatically generate and classify them? Can they be characterized by some general rules?

**Archimedean Tilings** are periodic tessellations composed by regular polygons only, like the cover image, reproduced in Figure 16. Also known as $n$-Uniform, Demi-regular or Semi-regular, more on this on Chapter 1.

In January 2018, a seminar was held at IMPA to present the book *Sobre Malhas Arquimedianas* [75] and I was invited to talk about my experience with my workshop at the Festival. Luiz Henrique de Figueiredo was present and he had some ideas for the research Asla and I were trying to start. After some talks, a formal PhD research project started with Luiz Henrique as my advisor and Asla as a co-advisor.

The first results of this research have been presented in conferences and published. We presented the first results in the paper *Synthesizing periodic tilings of regular polygons* [73] at the 31st Conference on Graphics, Patterns and Images (SIBGRAPI 2018). We presented the work: *Acquiring periodic tilings of regular polygons from images* [83] at Computer Graphics International 2019; it was published in *The Visual Computer Journal*.

Tim Weyrich joined the project after Asla visited him in the UK in 2019. He developed a computer search of triangle and square tilings based on our previous work. I spent three weeks visiting him at University College London discussing his work and symmetry classification.

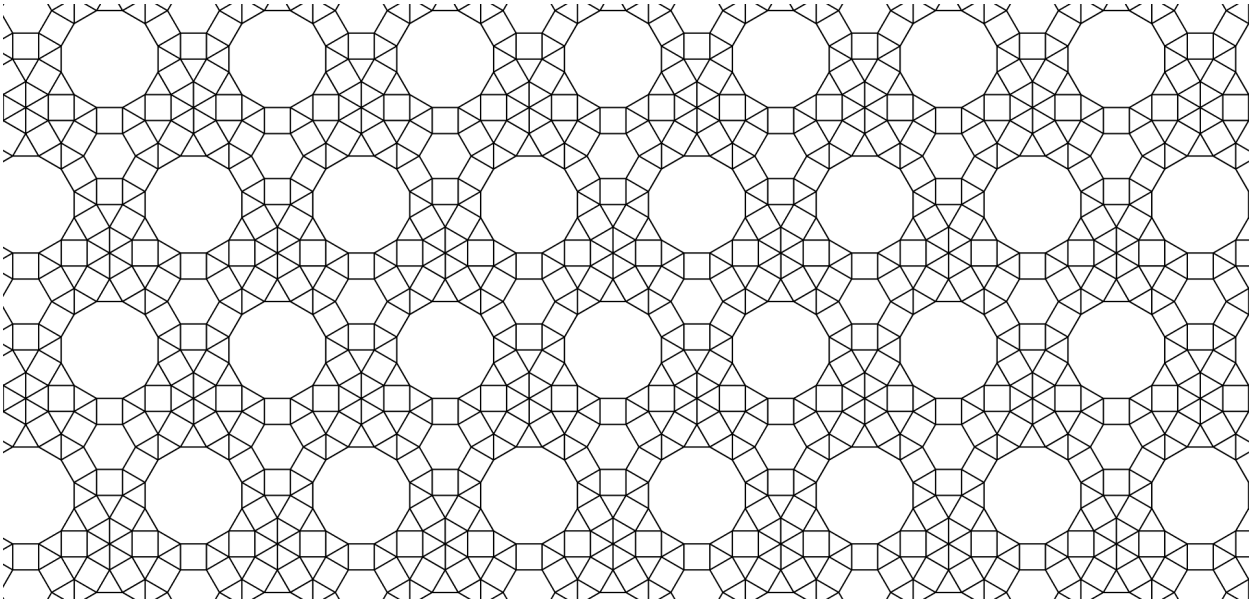


Figure 16: *Archimedean mesh* crafted by Ricardo Sá [75] – Cover image.

## *Contributions of this thesis*

The results of the research project are presented in Chapters 2, 3 and 4 of this thesis. We devised a simple integer-based representation for periodic tilings of regular polygons using complex numbers. This representation allowed us to acquire geometrical models from two large collections of images – which constituted the state of the art in the subject –, to synthesize new images of the tilings at any scale with arbitrary precision, and to recognize symmetries and classify each tiling in its wallpaper group as well as in its $n$-uniform $k$-Archimedean class.

Chapter 5 of this thesis focuses on a generation and enumeration strategy for tilings with regular polygons. The old problem of characterizing all triangle-square tilings is solved [18, 82] via equivalence of a tiling with an edge-labeled hexagonal graph (Figure 17). Defining a restricted dual for triangle-square tilings and its associated edge-labeled hexagonal graph, a geometrically restricted combinatorial formulation is derived: each tiling is equivalent to a valid labeling of its dual hexagonal graph, which is embedded in a flat torus. This representation allows the definition of families of triangle-square tilings by the topology of their dual hexagonal graph. Each family has an algebraic structure: tilings can be combined in positive-integer linear combinations to form new tilings. A Hilbert basis is obtained for each family. Enumeration is derived from this basis. Finally, I use some of the generators to create a sample set of more than 100 million triangle-square tilings and describe some general characteristics. This result sets the foundations for the enumeration of all periodic tilings with regular polygons, since triangle-square tilings contain all the others, given a natural notion of density.

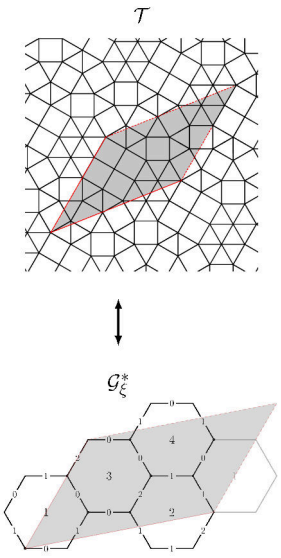



Figure 17: A triangle-square tiling and its edge-labeled dual hexagonal graph.

## *Describing the journey*

Chapter 1 contains the mathematical background on tilings and tessellations. We establish the main definitions and known results in the area and delimit the problem that we are going to tackle: tiling the plane periodically with regular polygons.

Chapter 2 describes in great detail a novel representation for periodic tilings of the plane by regular polygons. This representation was introduced in two papers [73, 83], but not discussed in depth. We show that the tilings can be characterized as projections of cosets of lattices in $\mathbb{Z}^4$ using complex numbers. We will show an abstract representation and a concrete computational framework for this representation. The final section discusses the representation properties using the guidelines proposed by Requicha [71] for representation schemes. We discuss our representation in the light of each of these properties and how it can be achieved by our representation.

The following chapters are applications of our representation to some of the problems related to tilings of regular polygons. Chapter 3 describes the acquisition of computational representations for state-of-the-art collections of tilings, which previously only existed as images.

In Chapter 4 we discuss the $k$-Archimedean $n$-uniform classification of tilings through the automatic identification of its plane symmetry group, one of the 17 wallpaper groups. This chapter is related with Chapter 1, where symmetry groups are presented with greater detail.

In Chapter 5 we define the notion of dense tilings, formed by triangles and squares only, which contain all tilings by regular polygons after refinement. Exploiting the topology of their dual we derive the equivalence of a triangle-square tiling with an edge-labeled hexagonal graph over the torus. This representation allows us to define families of triangle-square tilings and find a set of minimal generators that give the family an algebraic structure. This offers a characterization of all triangle-square tilings. We show some experimental results of using this approach to generate tilings and some geometric insight on their algebraic structure.

In Chapter 6 we point to some immediate applications in art, design and education, some of them being developed by us already. We also state some research questions that remain open after this work.

*Acknowledgments*

My gratitude to Luiz Henrique de Figueiredo for the trust and the openness in the orientation through my research for this thesis. He encouraged me and allowed me to build increasing autonomy as a researcher, always with useful advice.

Many thanks to Asla Medeiros e Sá for opening the door to the collaboration that became my thesis project. I was very lucky to find someone with a shared passion for tilings at the right moment. I have greatly appreciated her advice.

Tim Weyrich generously invited me to University College London to discuss tilings. His collaboration has been very enriching for the project over the last year.

I am grateful to Diego Nehab for being my first teacher at IMPA, for receiving me and orienting me in the first two years on the program. He was very generous and sensitive to help me commit to a research project for which I was enthusiastic. I will always be thankful for his support.

I thank Luiz Velho for the warm welcome to the Visgraf laboratory and for being always supportive with my involvement in creative projects in the lab. Also Djalma Lucio, for the cordiality and always being there to save us with tech issues.

Everyday life in the laboratory was pleasurable thanks to the fellowship and familiarity of Daniel Yukimura, Julia Giannella, Pedro and Caio Souza, Vitor Guerra, Santiago Guisasola, Lenka Ptackova, and José Eduardo Ayres. I will always esteem you.

My love to Zeza Barral, for the affection and the patience with me through lock-down and the thesis writing process.

I am grateful for the constant love and care of my community: Marcos, Pablo, Elsa, Andrés, Mildret, Uriel, Elisa, Agustín, Coco, Angélica, Irving, Jean.

To my dear friend Diego Rodríguez, I thank the constant support in life, and the interest and encouragement on my research. To the large community of migrants that have supported me here, especially to the *mexicariocas*!

I acknowledge and thank some of my teachers throughout the years, for the professional mathematical education that brought me here: José Luis Morales, Marcela González, Javier Alfaro, César Luis García, Guillermo Grabinsky, Rafael Morones†, Rubén Hernández, Eugenio L. Fautsch, Enrique *Cachi* Alonso†, among many others.

My recognition and gratitude to all the IMPA staff that made my student's life so easy, especially to Andrea Nascimento, Josenildo Pedro and Isabel Cherques, for their constant support and attention; to Ana Luisa Machado and Juliana Bressan for welcoming my contribution and ideas in the communication and education events; to Antonio Carlos Oliveira, Florisvaldo de Souza, Miguel Antonio dos Reis, for the everyday smile and cordiality through a very hard first year at the institute.

Thanks to Alexander García Kastellanos for the proofreading.

This work was partially supported by a CNPq doctoral scholarship.

# 1 Tilings and Tessellations

The terms *tiling* and *tessellation* are synonyms. While *tiling* is the common term used in mathematics literature and *tessellation* is more common in art disciplines, they are equivalent and interchangeable. Intuitively, a *tiling*[1] consists of a set of shapes, called *tiles*, that covers a given surface, without gaps or overlaps, a kind of a jigsaw puzzle (Figure 1.1). The classical mathematical questions in the subject are about the number and the shapes of the tiles, as well as the transformation and matching rules for these shapes.

Among many other references, the following three vastly cover the main branches of mathematics in which this thesis is located. Grunbaum & Shephard [35] wrote the seminal book on tilings and patterns. Kaplan [44] wrote a very useful review on the subject in the light of computer graphics, bringing up many specific considerations. Liu et al. [57] take us deeper into symmetry in their survey on computational symmetry for computer graphics. We do not intend to compete with them in presenting the fundamentals of these subjects. In this chapter, we recall the main definitions and state the main results that allow us to build our proposal and refer the reader to the books above for a more in-depth view on these matters.

[1] *Tessellation*, mosaic, paving [34].

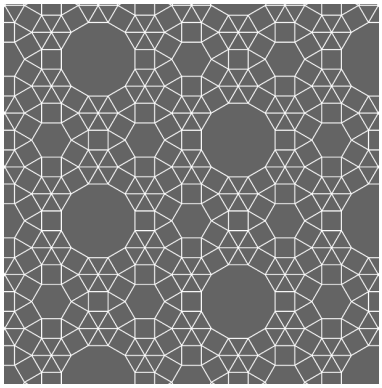


Figure 1.1: Example of a tiling.

## 1.1 Symmetry of plane patterns

Nature and human culture are full of periodic and symmetric patterns, from the atomic structure of crystals to building and city designs [12, 20, 91, 90]. Tilings of the plane are, in a general way, patterns. Understanding the underlying symmetric structure of plane patterns is a fundamental aspect in the study of tilings.

Ornaments have lots of symmetry, as one may observe. Artisans along history have crafted plane periodic patterns in (almost) all possible ways.[2] Symmetry came to the attention of science when crystallographers started modeling crystals by the behavior of atoms joining to form them.[3] Along the 19th century, extensive theoretical background in this area came to light in the work of crystallographers, mathematicians, and physicists.

An *isometry* is an operation that preserves distances. The isometries of the Euclidean plane – also called *rigid motions* – are the basis for defining symmetry. We are interested in the ways the isometries can form *groups* over the plane with certain properties.

[2] There is an active debate about which symmetry groups are actually present in *Alhambra* [10, 11, 33].

[3] See Moeck [59] for a crystallographic history of symmetry.

Let us consider the following basic isometries in the Euclidean plane, illustrated in Figure 1.2: identity, translations, rotations, reflections, and glide reflections (a translation composed with a reflection).

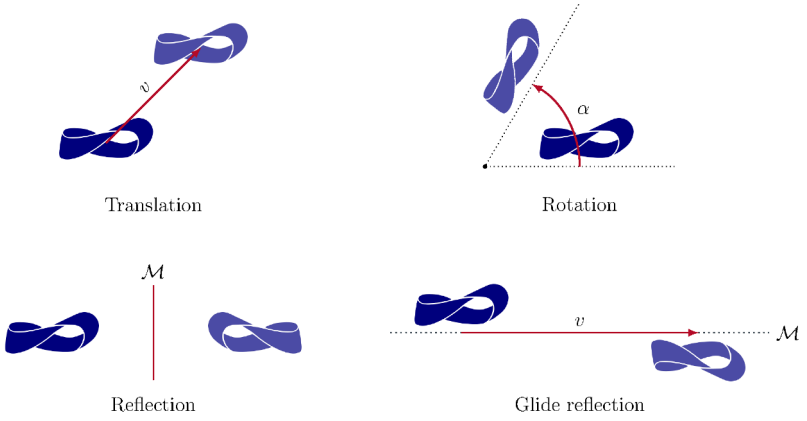



Figure 1.2: The fundamental isometries of the plane.

These operations can be combined by composition, an associative binary operator. Each of these operations has an inverse of the same kind: reflections are their own inverse, a rotation by an angle $\alpha$ is *undone* by a rotation with the same center with angle $-\alpha$ or $2\pi - \alpha$, the inverse of a translation by a vector $v$ is the translation by $-v$, which also applies to glide reflections.

Identity, translation, rotation, reflection, and glide reflection are called *fundamental isometries* because they span the group[4] of all plane isometries. This is known as the Fundamental Theorem of Isometries. The set of all isometries in the plane is infinite. Each real number $\alpha \in [0, 2\pi)$ defines a rotation. Each line in the plane defines a reflection, and so on. Isometries allow us to classify geometric shapes that look the same up to a rigid motion, the *congruence* relation. We say that two geometric objects or subsets of the plane are *congruent* if there is an isometry of the plane sending one object to the other. It is through the isometries that the notion of symmetry is defined.

We say that an object $A$ is *symmetric* with respect to an isometry $\sigma$ when $A = \sigma(A)$, this is, $A$ is *invariant* under the isometry $\sigma$. We also say that $\sigma$ is a *symmetry* of $A$. Note that the identity transformation is an isometry of the plane and so is a symmetry of every object. The symmetries of a given object form a group, its symmetry group. Two symmetry groups $G$ and $H$ are *equivalent* if there is an affine transformation[5] $A$ such that $A\sigma A^{-1} \in H$, for each symmetry $\sigma \in G$, and vice-versa.

We are interested in *discrete* groups of isometries and the kind of structures these groups generate on the plane. A set of points of the plane is discrete when $r > 0$ exists such that $d(p, q) > r$ for every pair of distinct points $p$ and $q$ in the set; equivalently, the set has no limit or accumulation points [44]. The set of images of a point in the plane under an isometry is called an *orbit*. A *discrete symmetry group* is one for which the orbits of all points in the plane are discrete.

[4] A *group* is a pair $(\mathcal{G}, \cdot)$, consisting of a set $\mathcal{G}$ and an associative binary operation $(\cdot)$ defined in $\mathcal{G}$ that has the following properties:

*Closure:* $a \cdot b \in \mathcal{G}$ for any $a, b \in \mathcal{G}$ .

*Associativity:* $(a \cdot b) \cdot c = a \cdot (b \cdot c)$ for all $a, b, c \in \mathcal{G}$.

*Identity:* there is an element $e \in \mathcal{G}$, called the *identity*, such that $a \cdot e = e \cdot a = a$.

*Inverse:* for all $a \in \mathcal{G}$, there is $b = a^{-1}$ in $\mathcal{G}$ such that $a \cdot a^{-1} = a^{-1} \cdot a = e$.

[5] An *affine transformation* is any transformation that preserves co-linearity and ratios of distances.

Discrete groups in the plane are well understood; literature about them is abundant [20, 35, 44]. However, since the results were achieved historically by very different approaches, there are a few distinct notations for representing them. John H. Conway R.I.P., introduced the *orbifold* notation for representing symmetry groups in two-dimensional spaces of constant curvature [20, 21, 22]. *Orbifold signatures* are compact symbols that fully encode the properties of the symmetry group. Besides the *orbifold* notation, we use the classic *crystallographic* notation[6], credited to Hermann and Mauguin, who introduced and refined it in 1928 and 1931, respectively. Its symbols are also very informative and have been widely used in literature. Other notations have been proposed by Coxeter, Schönflies, Polya & Guggenhein, Speiser & Nigle, and Fejes Toth & Caldwell, but we are not going to discuss them.

[6] Also called international notation, adopted by the International Union of Crystallography (IUCr) in the International Tables For Crystallography since its first edition in 1935.

There are three kinds of symmetry groups in the plane according to the number of translations they have:

1. Groups including rotations and reflections, but not translations. Such groups have an integer number of orbits $n$ for each point in the plane – except the center of rotation, which is a singularity. Groups with only rotations are called *n-cyclic groups*, while groups including reflections are called *n-dihedral groups*. Some examples of these kind of groups are shown in Figure 1.3.

   In *orbifold* notation, $n$-cyclic groups are denoted as $n\bullet$, while $n$-dihedral groups are denoted $*n\bullet$, where the $*$ symbol represents a reflection. In crystallographic notation these are $C_n$ and $D_n$, respectively.

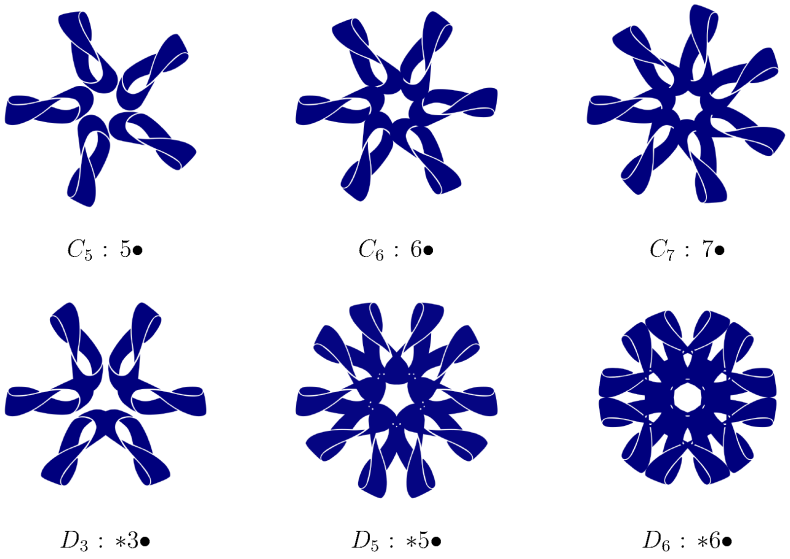



Figure 1.3: Examples of shapes in the plane with cyclic and dihedral symmetry groups.

2. Groups having only one family of collinear translations have orbits that repeat infinitely along that one direction in the form of stripes. These are called *frieze groups*, because the objects over them form *frieze* patterns. There are exactly seven such groups, which are shown in Figure 1.4. The symbols indicate translation $(1 : \infty)$[7], reflection $(m : *)$, glide reflection $(a : \times)$, and 2-fold rotation (2).
3. The remaining groups should have linearly independent sets of translations. The orbits of such groups fill the plane forming symmetric repeating patterns given any shape, and the converse is also true, the symmetries of any repetitive pattern on the plane correspond to one of these groups. These are the 17 *plane symmetry groups*, also called *wallpaper groups*, illustrated in Figure 1.5.[8]

   The existence of exactly 17 wallpaper groups was proved independently by Fedorov in 1891 [27] and Pólya in 1924 [68]. Conway et al. [20, 21, 22] brought a novel view of the groups when they introduced the *orbifold* notation. The result became known as the Magic Theorem. The symbols indicate translation $(1 : \circ)$, reflection $(m : *)$, glide reflection $(g : \times)$, and $n$-fold rotation $(n \in \{2, 3, 4, 6\})$. In the *orbifold* notation, $\circ$ is called a *wonder*, which topologically represents a torus, and $\times$ is called a *miracle*, which represents a topological crosscap.

[7] The $\infty$ in the orbifold notation indicates infinite rotational symmetry around a line in 3-space, which is a subgroup of symmetries with only one independent translation. This is the reason this group is associated with the frieze group.

[8] An important result is that $p1$ is a subgroup of all symmetry groups. We will use this fact in our representations of tilings.

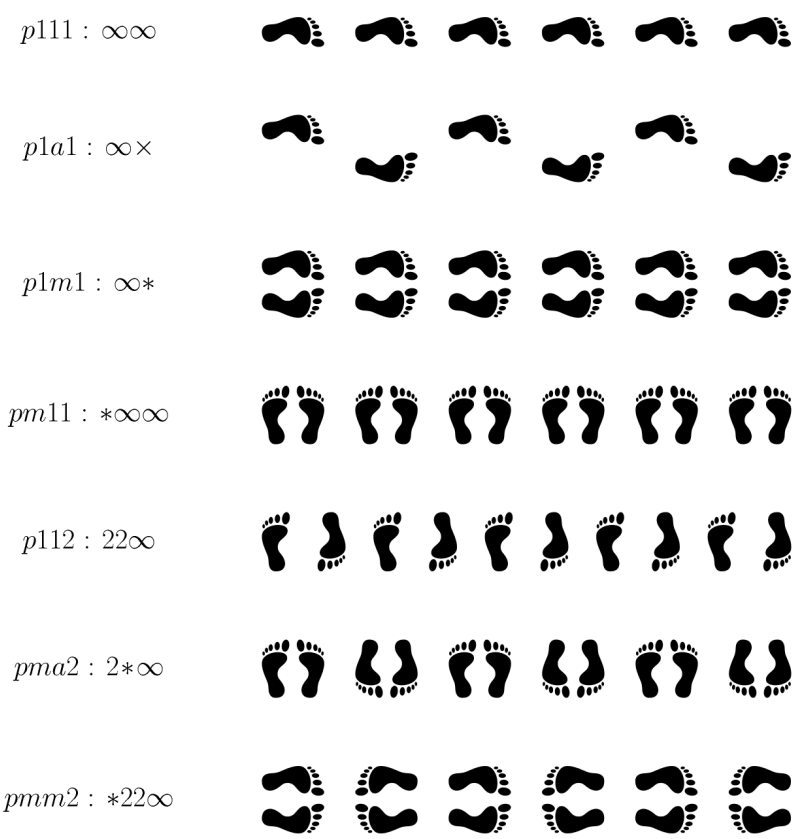



Figure 1.4: Examples for each frieze symmetry group.

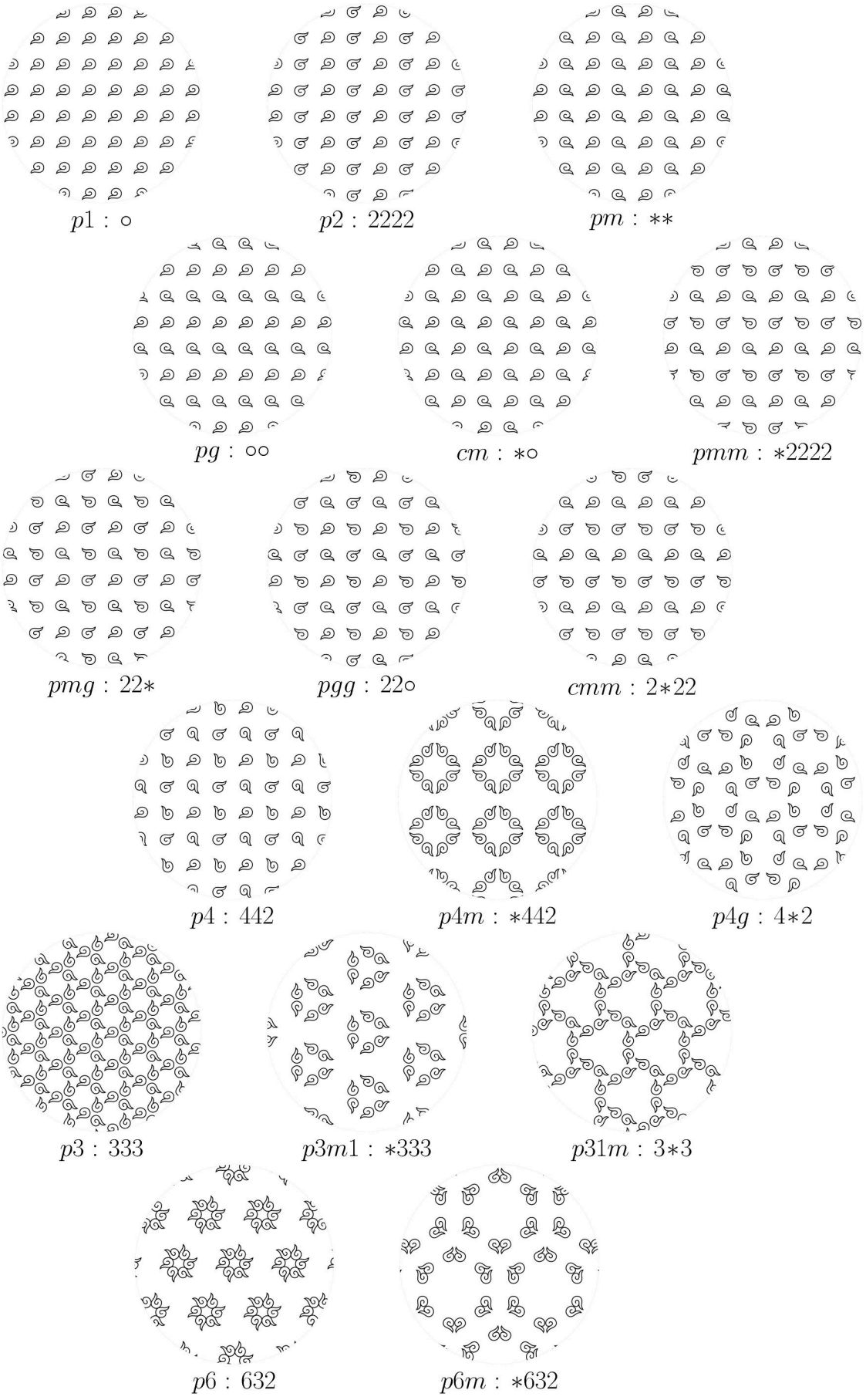



Figure 1.5: Examples of all 17 plane symmetry groups.

## 1.2 *Tilings of the plane*

To start, we want our tiles to be manageable in a geometric sense. Thus, we suppose they are closed and simply connected sets: topologically equivalent to a closed disc. We also assume that they are *uniformly bounded*: there exist $r > 0$ and $R > 0$ such that every tile contains the disc of radius $r$ and is contained in the disc of radius $R$. Since the tiles are closed sets, we allow them to overlap only at their borders, otherwise they don't cover the entire plane.

A *tiling* is a decomposition of the plane by a set of tiles such that:

a) Each tile is a closed set, topologically equivalent to a closed disc.

b) Every point of the plane is contained in a tile.

c) The interiors of the tiles are disjoint.

d) The tiles are uniformly bounded.

By this definition we can imagine many free-form tiles, arranged in many different ways, so let us make some additional assumptions over the tilings we are interested in.

We will consider our tilings to be *polygonal* (all tiles are polygons) and *edge-to-edge*, that is, every edge of a tile is an edge of exactly one other tile, and when two tiles intersect, they do it either on one edge or a vertex, otherwise they are disjoint (Figure 1.6).

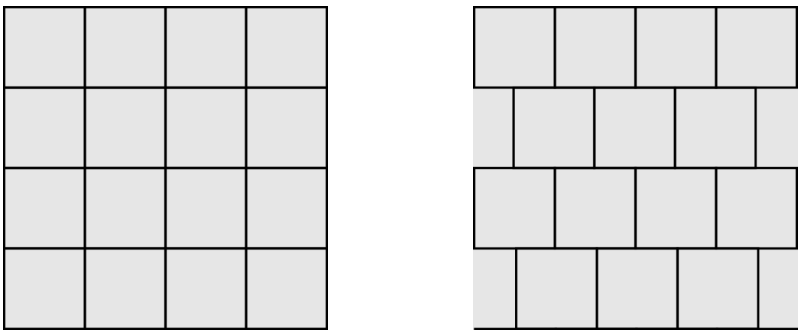


Figure 1.6: Examples of polygonal tilings: in the left an *edge-to-edge* tiling, on the right a non *edge-to-edge* tiling.

Even though one could insert vertices at the intersections of edges with edge endpoints, these would not be proper vertices of the polygons and generate parallel edges, which are not desirable.

In any application of tilings we would like to have a finite set of tiles so that we can represent the tiling in the computer. Congruent tiles are considered to be the same.

A tiling is *k-hedral* if the tile set has exactly $k$ congruence classes of tiles. The *unique* tiles in this set are called *prototiles*. When $k = 1$, the tiling is said to be *monohedral*. When $k = 2$, it is called a *di-hedral* tiling. Given a set $\mathcal{P}$ of prototiles, if there is a tiling formed by congruent copies of its members, we say that this tiling is *admitted* by $\mathcal{P}$.

The cardinality of a set of prototiles must not be confused with the $k$-hedral class of a tiling admitted by it, since there is no condition for the use of all the members of the set. Given a set of shapes, it is not always possible to determine with full generality whether there is a tiling admitted by this set [45].[9]

A tiling is said to be *periodic* when there are two linearly independent translations for which the tiling is invariant. The two translation vectors form a translation *lattice*, a discrete additive group of $\mathbb{R}^2$. In any position, this lattice forms a grid of parallelograms. Each cell in the grid contains all the information of the tiling (Figure 1.7). An immediate consequence of the periodicity of a tiling is that their symmetries must belong to one of the 17 wallpaper groups presented above, since every repetitive pattern in the plane belongs to one of these groups [20].

A tiling that is not periodic is called *aperiodic*. Aperiodic tilings are fascinating [6, 80]. We mention two famous examples: the Penrose tiling (Figure 1.8) and the Ammann–Beenker tiling (Figure 1.9), aperiodic tilings with uncommon 5-fold and 8-fold symmetry, respectively. The history of the Amman-Binker tiling is one of an unusual and inspiring contribution to mathematics. This tiling has a *quasicrystal* structure [6].

[9] This is called *The Tiling Problem* and is a famous example of undecidable problem.

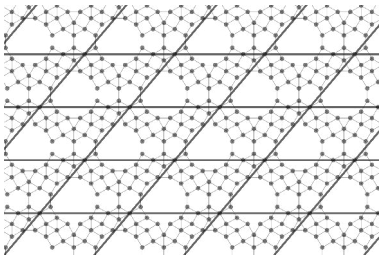


Figure 1.7: Example of a periodic tiling and its translation lattice.

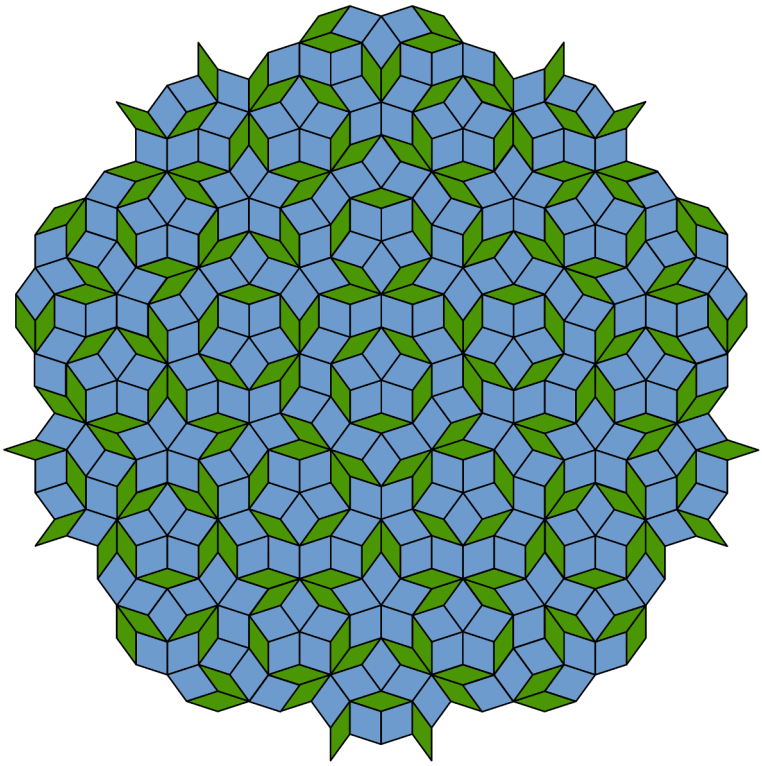


Figure 1.8: A Penrose tiling ($P_3$) using thick and thin rhombi. Public domain.

Since periodicity is a fundamental property for the tilings we study here, from now on, whenever one reads the word *tiling*, it always refers to a periodic one, unless the contrary is explicit.

Tilings remain a very active field of study, with many open problems. In 2017, Rao [70] answered the long-standing question of how many different convex pentagons tile the plane with the aid of the computer. Just a few months ago, the Notices of the AMS published Zong's review of Rao's result and other recent *dramatic progresses*[10] from recent years about the types of convex domains that tile the plane [96].

[10] In his own words.

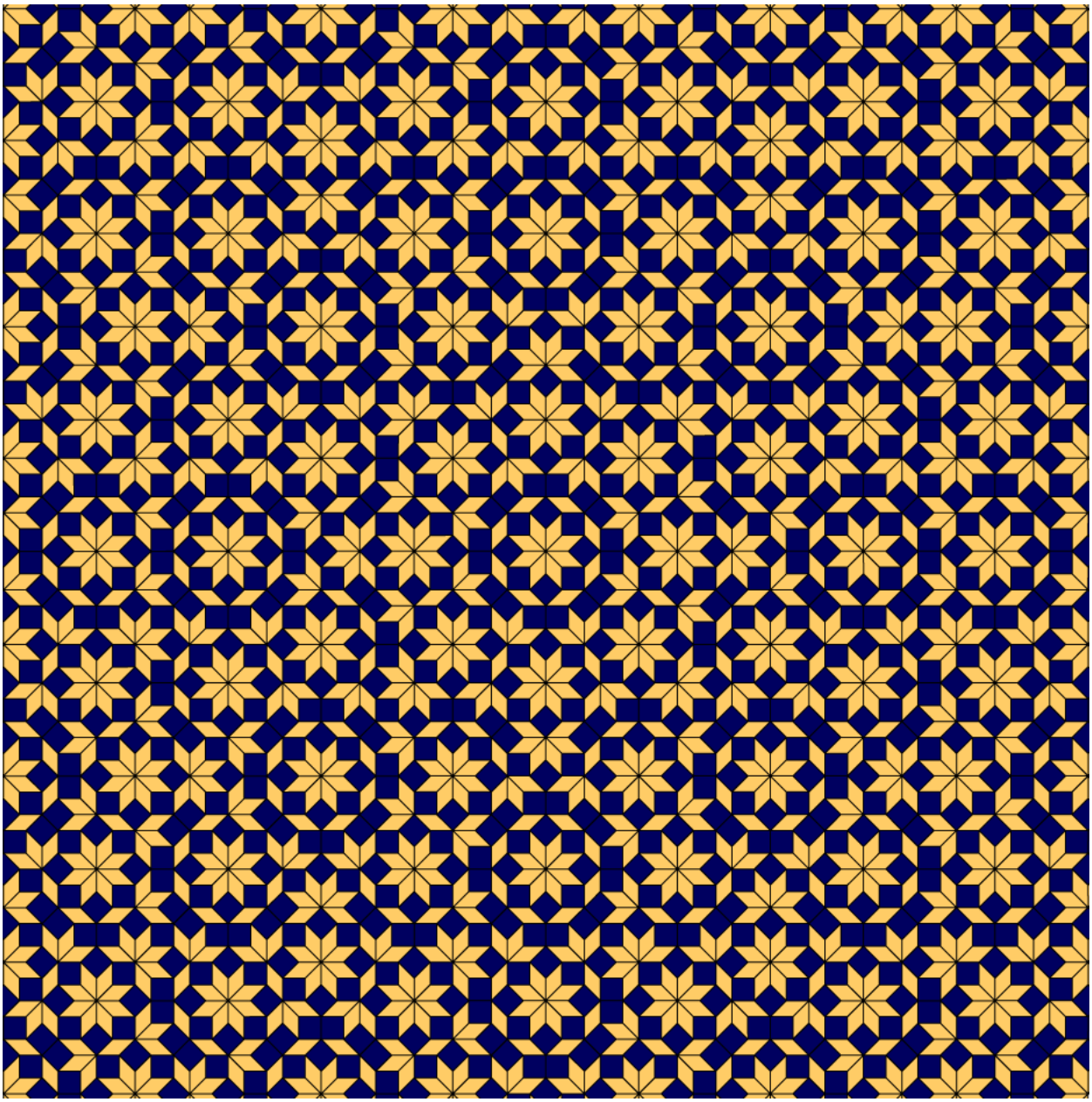


Figure 1.9: Ammann–Beenker tiling, Tilings Encyclopedia [26], CC BY-NC-SA 2.0.

## 1.3 *Periodic tilings with regular polygons*

Pythagoras formalized the arrangement of regular polygons into regular solids and Archimedes generalized these to some quasi-regular solids, today known as the Archimedean solids. Kepler continued the work on the solids and, 400 years ago, was the first to try to systematize the regular tilings (tessellations) of the plane (Figure 1.10).

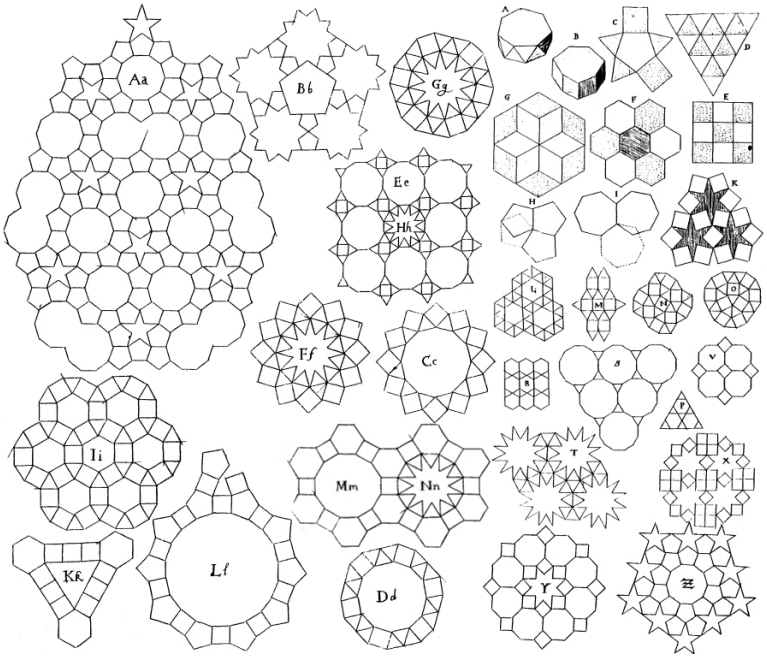



Figure 1.10: Tilings of regular polygons in *Harmonices Mundi*, Johannes Kepler, 1619.

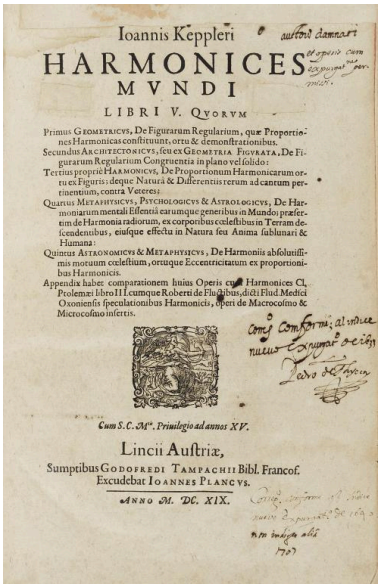

Ioannis Keppleri
HARMONICES
MVNDI
LIBRI V. QVORVM
Lincii Auſtriæ,
ANNO M. DC. XIX.

Figure 1.11: *Harmonices Mundi* [The Harmony of the World], Johannes Kepler, 1619.

Archimedean tilings are edge-to-edge tilings with regular polygons. The possible tilings are governed by an equation about the internal angles of the $n$ polygons that share that vertex, which must cover all the space around that vertex. If the polygons sharing a vertex have $k_i$ sides each ($i = 1, \ldots, n$), then, by the sum of the internal angles of a regular polygon, we have

$$\sum_{i=1}^{n} \frac{(k_i - 2)}{k_i} \pi = 2\pi.$$

This equation has 17 integer solutions (Table 1.1), which turn into 21 after considering their circular orderings. Order matters when we interpret the solution in a geometric sense [34, 82], because they correspond to the order the polygons are arranged around the vertex. However, only 15 configurations correspond to vertices that can form tilings, the ones showed in Figure 1.12. The remaining solutions work locally but create unfeasible arrangements in neighboring vertices.

Only 11 types of vertices tile uniformly, forming the known Archimedean tilings, shown in Figure 1.13. The other 4 types cannot form tilings on their own, only when combined with other types of vertices. The 11 Archimedean tilings were already documented by Kepler in 1619 and later by Sommerville in 1906 [82], according to Ouyang et al. [64].

| Polygons | Number of sides ($k_i$) | | | | | | Geometric solutions |
|---|---|---|---|---|---|---|---|
| $n = 3$ | 3 | 7 | 42 | | | | – |
| | 3 | 8 | 24 | | | | – |
| | 3 | 9 | 18 | | | | – |
| | 3 | 10 | 15 | | | | – |
| | 3 | 12 | 12 | | | | $G$ |
| | 4 | 5 | 20 | | | | – |
| | 4 | 6 | 12 | | | | $H$ |
| | 4 | 8 | 8 | | | | $J$ |
| | 5 | 5 | 10 | | | | – |
| | 6 | 6 | 6 | | | | $K$ |
| $n = 4$ | 3 | 3 | 4 | 12 | | | $L, M$ |
| | 3 | 3 | 6 | 6 | | | $Q, R$ |
| | 3 | 4 | 4 | 6 | | | $N, P$ |
| | 4 | 4 | 4 | 4 | | | $S$ |
| $n = 5$ | 3 | 3 | 3 | 4 | 4 | | $T, U$ |
| | 3 | 3 | 3 | 3 | 6 | | $V$ |
| | 3 | 3 | 3 | 3 | 3 | 3 | $W$ |

Table 1.1: Solutions to the Archimedean vertex equation. Geometric solutions are illustrated in Figure 1.12.

Some literature considers the tilings $K$, $S$ and $W$ as the only regular tilings, because they are the only ones with just one type of polygon. The other 8 tilings, consisting of one type of vertex and more than one type of polygon, have been called demi-regular or semi-regular tilings [5, 24, 31, 34, 93]. However, there is no consensus about these definitions, as discussed by Aslaksen [5].

We are not going to try to settle this dispute of terms, but rather choose the one more often used in the literature and consistent with our own view of tilings with regular polygons: the one based on vertex types. A periodic tiling with regular polygons is *n-uniform* if it contains $n$ classes of transitively equivalent vertices, and *k-Archimedean* if these vertices belong to $k$ Archimedean classes, the ones listed in Table 1.1 and Figure 1.12.

In 1905, Sommerville [82] analyzed combinations of Archimedean vertices. In 1968, Krötenheerdt found the twenty 2-uniform tilings, which are the ones that have two classes of vertices [35, 86]. More comprehensive approaches to the study of periodic tilings of the plane with regular polygons were made by Chavey in three stages [17, 16, 18]. In his second work, he discusses classifications by vertex and edge classes of equivalence of some high-order tilings [16].

Galebach [28] investigated high-order $n$-uniform, $k$-Archimedean tilings in 2002. The result is a collection of images in his website: `probabilitysports.com/tilings.html`. The number of tilings found by him for classes up to 6-uniform 6-Archimedean tilings, plus the 7-uniform 7-Archimedean class of tilings, are cited in The On-Line Encyclopedia of Integer Sequences (OEIS) [81] (Table 1.2). However, he has not yet published any methodological or theoretical discussion of his findings.

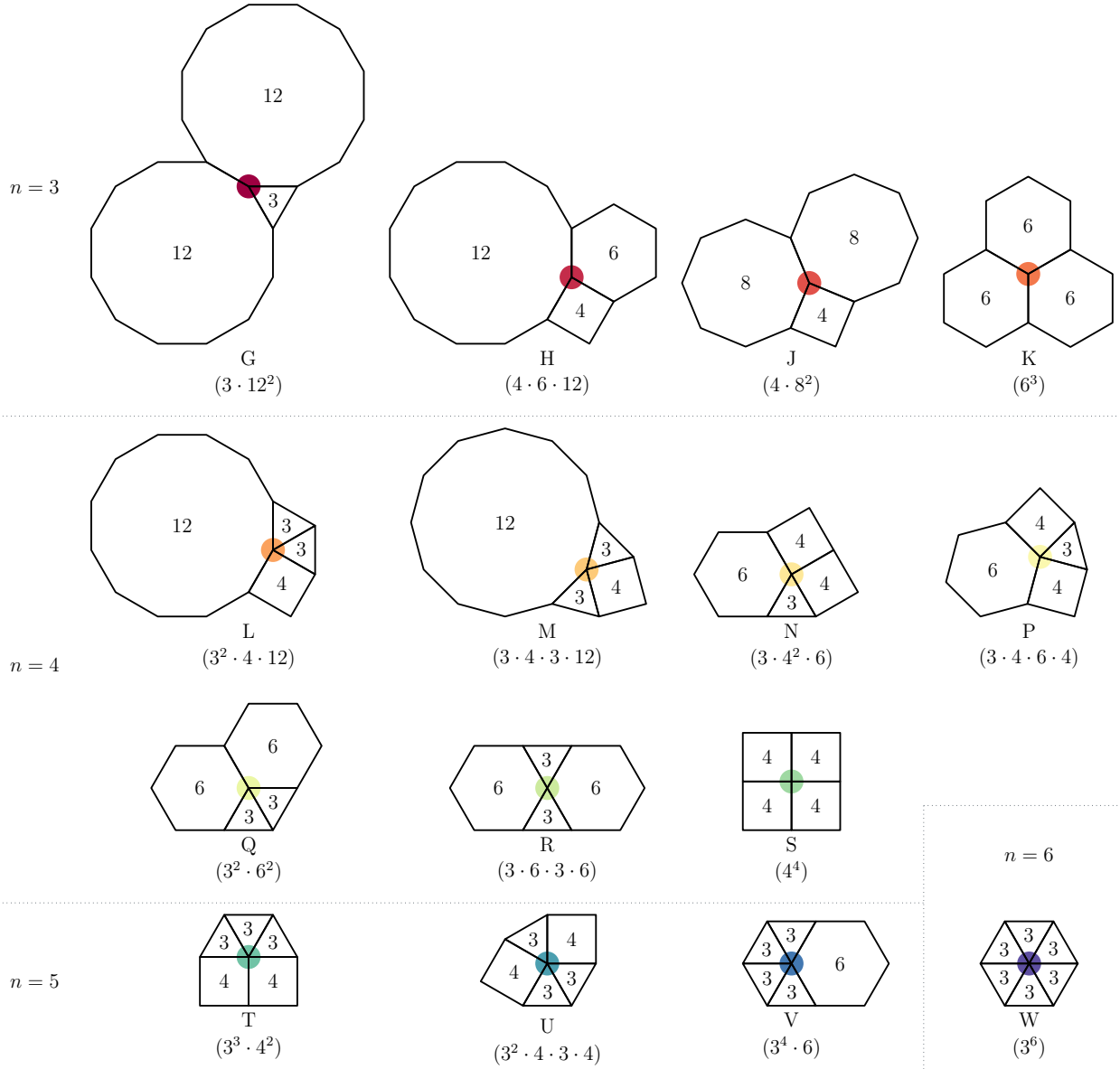



Figure 1.12: Archimedean vertices.

Lengrenn [53] made the most recent survey on periodic tilings with regular polygons in 2009, gathering the existing references and trying to order the information available at the time. Recent work on tilings with regular polygons include Chavey's dodecagon dense tilings [18], Akiyama's corona limits [4], Connelly's circle packings [19], Peng et al. hybrid meshes for surfaces [66], and Kormos et al. synthesis of molecular arrangements [49].

The subject has gotten constant attention in mathematics education and communication through time [29, 31], since it is easy to enunciate the most simple problems, such as the vertices solutions, and the results are visually appealing.

In art and design, Critchlow [24] included the demi-regular tilings in his design source book. Recently, Hofmann [41] made his own approach to the construction of Archimedean tilings using the dual. His approach disregards periodicity and other formal considerations, hence, the results are very general.

Sá & Sá [75] published a crafted collection of tilings with regular polygons. Their work is linked to the start of this research project [73]. We acquired their collection for this work using an integer-based compact representation [83]; see Chapter 3 for more details.

Our work focuses on this specific subject: periodic tilings with regular polygons. The representation we present in Chapter 2 allows us to model the tilings, including the full Galebach collection [28], as we show in Chapter 3. In Chapter 4 we describe a method to automatically find the symmetries of a tiling and classify it in its $n$-uniform $k$-Archimedean class.

Every question mark in Table 1.2 is an open problem, and we are building our path towards solving some of them. In Chapter 5 we set the foundations for the enumeration of all periodic tilings with regular polygons. We characterize all periodic tilings with triangles and squares by showing the algebraic structure of well-defined families of tilings and their basis.

| | | $k$-Archimedean | | | | | | | | | | | | | | Total |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | |
| $n$-uniform | 1 | 11 | – | – | – | – | – | – | – | – | – | – | – | – | – | 11 |
| | 2 | 0 | 20 | – | – | – | – | – | – | – | – | – | – | – | – | 20 |
| | 3 | 0 | 22 | 39 | – | – | – | – | – | – | – | – | – | – | – | 61 |
| | 4 | 0 | 33 | 85 | 33 | – | – | – | – | – | – | – | – | – | – | 151 |
| | 5 | 0 | 74 | 149 | 94 | 15 | – | – | – | – | – | – | – | – | – | 332 |
| | 6 | 0 | 100 | 284 | 187 | 92 | 10 | – | – | – | – | – | – | – | – | 673 |
| | 7 | 0 | ? | ? | ? | ? | ? | 7 | – | – | – | – | – | – | – | ? |
| | 8 | 0 | ? | ? | ? | ? | ? | 20 | 0 | – | – | – | – | – | – | ? |
| | 9 | 0 | ? | ? | ? | ? | ? | ? | 8 | 0 | – | – | – | – | – | ? |
| | 10 | 0 | ? | ? | ? | ? | ? | ? | 27 | 0 | 0 | – | – | – | – | ? |
| | 11 | 0 | ? | ? | ? | ? | ? | ? | ? | 1 | 0 | 0 | – | – | – | ? |
| | 12 | 0 | ? | ? | ? | ? | ? | ? | ? | ? | ? | ? | 0 | – | – | ? |
| | 13 | 0 | ? | ? | ? | ? | ? | ? | ? | ? | ? | ? | ? | 0 | – | ? |
| | 14 | 0 | ? | ? | ? | ? | ? | ? | ? | ? | ? | ? | ? | ? | 0 | ? |
| | $> 14$ | 0 | ? | ? | ? | ? | ? | ? | ? | ? | ? | ? | ? | ? | ? | ? |

Table 1.2: $n$-uniform $k$-Archimedean tilings [28, 81].

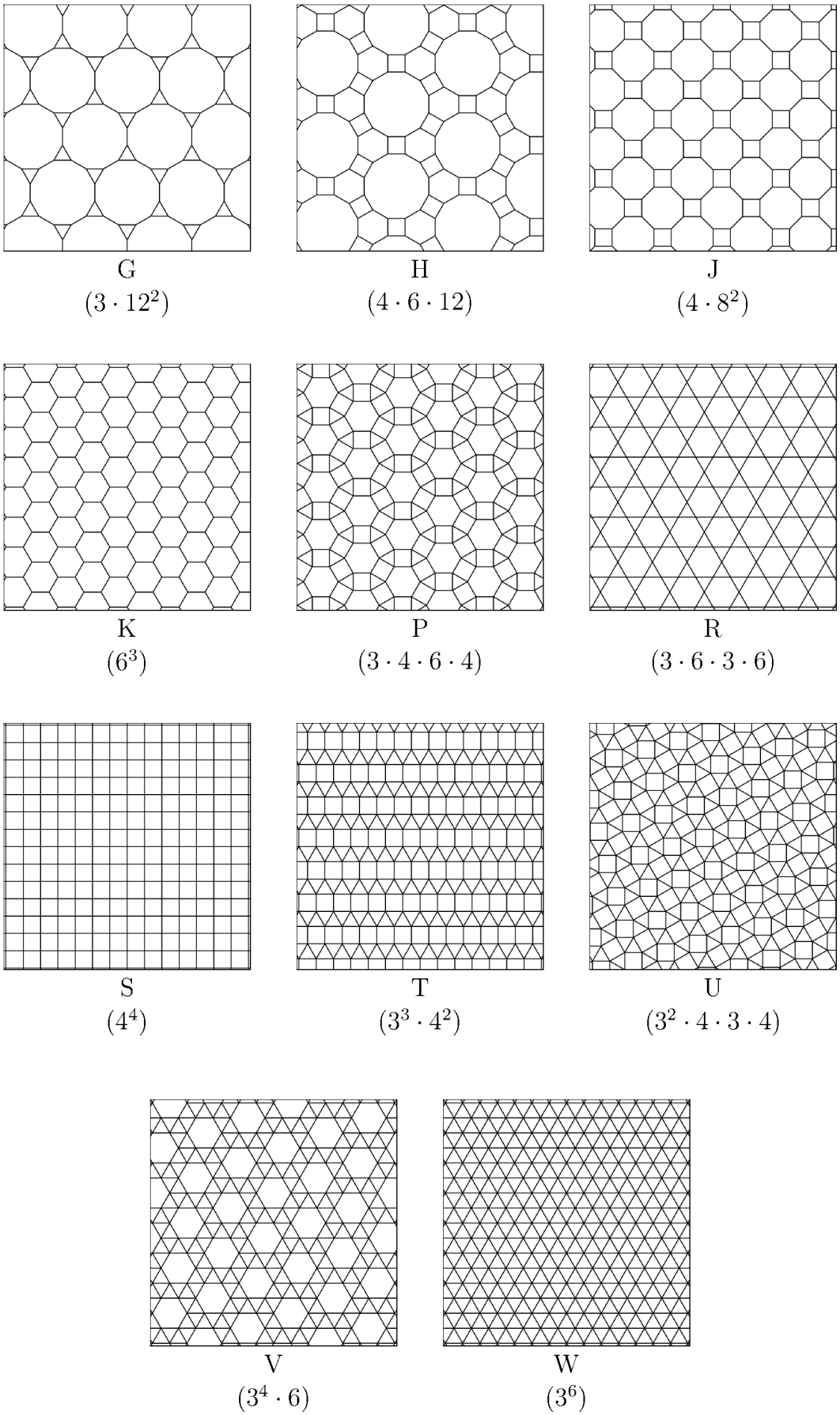



Figure 1.13: The eleven 1-uniform Archimedean tilings.

# 2 Integer Lattice Representation

We conceptualized the periodic tilings of the plane with regular polygons at the end of Chapter 1. Now it is time to describe a concrete representation for these tilings. To make the representation concrete we need to give coordinates to the vertices of the tilings. In principle, we could simply use Cartesian coordinates. However, the Cartesian coordinates of the vertices in a tiling are rarely exact rational numbers.[1] We seek a numerically exact representation. We shall see presently how to give integer coordinates to all vertices, how to build a symbol encoding all the information necessary to recreate a tiling, and discuss the applications and properties of our representation.

[1] The only exception is the square tiling shown as $4^4$ in Figure 1.13.

## 2.1 Translation cells

As discussed in Chapter 1, every tiling is a plane periodic pattern, and so, regardless of its symmetry group, it is invariant under the translation group ($\circ : p1$). Therefore, there are two linearly independent translation vectors $t_1$ and $t_2$ such that the tiling is invariant under every integer linear combination of $t_1$ and $t_2$.[2] This creates a parallelogram grid of identical regions of the plane, a *translation grid*.

[2] These vectors are not unique, as we discuss in Section 2.6. For now, we don't even require that they are minimal.

The *basic translation cell* is defined as the translation cell of the lattice referent to the origin, this is, the region of the plane that contains the origin, delimited by the parallelogram formed by the convex combinations of the translation vectors. Then, the basic translation cell for the translation grid is

$$T_{0,0} = \{\lambda_1 t_1 + \lambda_2 t_2 \mid \lambda_1, \lambda_2 \in [0,1)\}.$$

Every point $q$ in the plane is equivalent under translations to a point $p$ in the basic translation cell, this is $q = p + n_1 t_1 + n_2 t_2$ with $n_1, n_2 \in \mathbb{Z}$. In algebraic terms $p$ and $q$ are in the same coset of the translation lattice $\mathbb{Z}t_1 + \mathbb{Z}t_2$, a discrete additive group of $\mathbb{R}^2$. Note that the fundamental domain is *half open*, avoiding duplication of points in half the border.

Since the basic cell is bounded, we can represent the tiling by a finite set of points, the vertices inside it. If the translation vectors are minimal, the parallelogram is a fundamental domain of the translation lattice. No two points in the fundamental domain are equivalent, so there is exactly one representative of each class of equivalence of vertices by the translation group.

The vertices inside the basic cell are called *seeds*. The seeds allow us to reproduce the whole tiling in any region of the plane by simple translation operations, which is very convenient for computational applications.

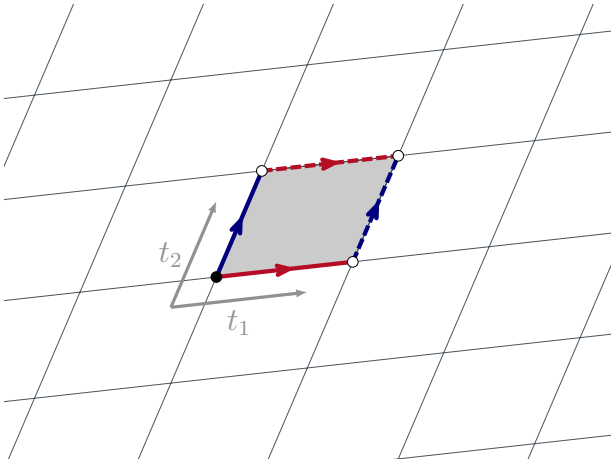



Figure 2.1: Fundamental domain over translation lattice. Its equivalence with a flat torus is shown by the topology annotation.

The Euclidean plane modulo the equivalence of the translation lattice is topologically equivalent to a *flat torus*. Thus, a fundamental domain can be modeled as a flat torus, with opposite sides identified and preserving the metric of the plane, as it is illustrated in Figure 2.1.

## 2.2 *Basic directions*

Our approach to representing periodic tilings of the plane with regular polygons is to think of them as regular systems of points, in the sense of Hilbert and Cohn-Vossen [40]. Since all the faces are regular polygons with unit side length (without loss of generality), neighboring vertices are at distance 1 from each other, and edges and faces can be all deduced from the vertices, as we will explain.

The first important fact is that only 14 of the 15 Archimedean vertices can be combined together in higher-order tilings. The $J$ Archimedean vertex type $(4^2 \cdot 8)$, which is the only one that includes octagons, cannot be combined with any other type. For this reason we disregard this vertex type.

A tiling combining the other 14 vertices will have 3, 4, 6, or 12 sided regular polygons only. This way, all the edges are aligned with one of twelve distinct directions with respect to each other, those of the regular dodecagon of unit side. Without loss of generality, assuming one of the edges is horizontal, we can then represent these directions using the 12th complex root of the unit.

Let $\omega$ be the principal 12th root of the unit: $\omega = e^{\frac{2\pi i}{12}} \in \mathbb{C}$. Then, the relative position of a vertex with respect to any other vertex can be represented by

$$\omega^k = e^{\frac{2\pi i}{12}k}, \quad \text{for some} \quad k \in \{0, 1, \ldots, 11\} \text{ (Figure 2.2).}$$

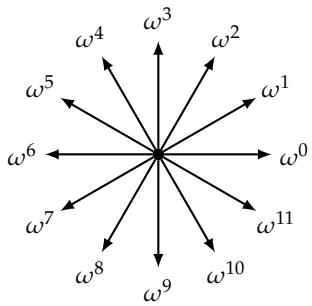



Figure 2.2: Basic directions represented as powers of the principal 12-th complex root of the unity $\omega$.

These directions represent every edge of the polygons in the tiling, as it is illustrated in Figure 2.3. Let us focus on a vertex $v$ of the

polygon forming an edge in the direction $\omega^k$ with another vertex. Then, the adjacent edge of the polygon in $v$ is $\omega^{k\pm j}$, being $j = 6 - {}^{12}/_{m}$ for a polygon with $m$ sides. This is $j = 2$ when the polygon is an equilateral triangle, $j = 3$ for a square, $j = 4$ for a regular hexagon, and $j = 5$ for a regular dodecagon. Note that $\omega^{k\pm j} = \omega^{k\pm j \bmod 12}$. This fact will be useful when computing polygonal faces.

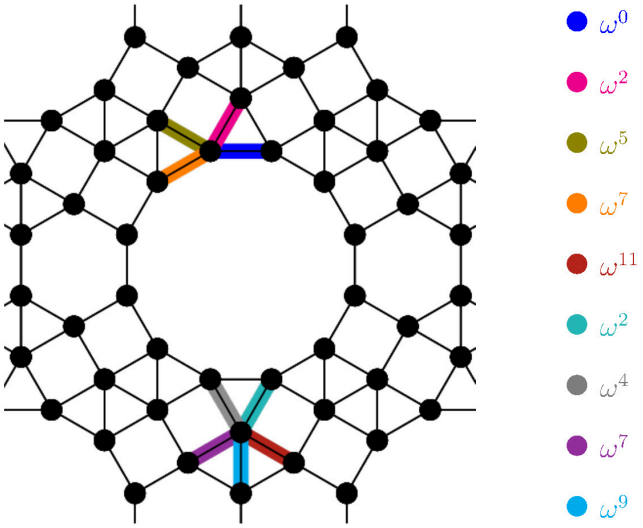



Figure 2.3: Basic directions identified in an example tiling.

While $\omega^k$ represents an edge and the position of an adjacent vertex with respect to another given vertex, concatenation of these steps can span positions of farther vertices in the tiling. Since we can always find a path through the edges of the tiling from one vertex to each other, we can represent the position of a vertex with respect to another one with a polynomial in $\omega$, as illustrated in Figure 2.4, where:

$$\begin{aligned} V - O &= \omega + \omega^{10} + \omega^{11} + \omega^0 + \omega + \omega^2 + \omega^3 \\ &= \omega^{11} + \omega^{10} + \omega^3 + \omega^2 + 2\omega + 1. \end{aligned}$$

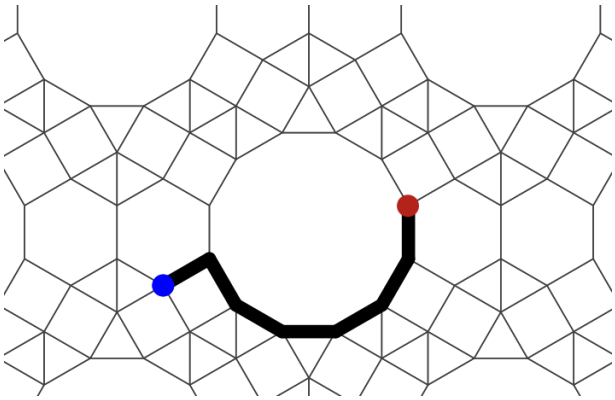


Figure 2.4: Position of vertex $V$ with respect to $O$ as a polynomial in $\omega$.

When the path connects a vertex to one of its nearest translation equivalent vertices, the path represents a translation vector of the tiling. Then, translation vectors can also be represented as polynomi-

als in $\omega$, as illustrated in Figure 2.5, where:

$$\begin{aligned} T &= \omega + \omega^3 + \omega^2 + \omega^3 + \omega^2 + \omega + \omega^{11} \\ &= \omega^{11} + 2\omega^3 + 2\omega^2 + 2\omega. \end{aligned}$$

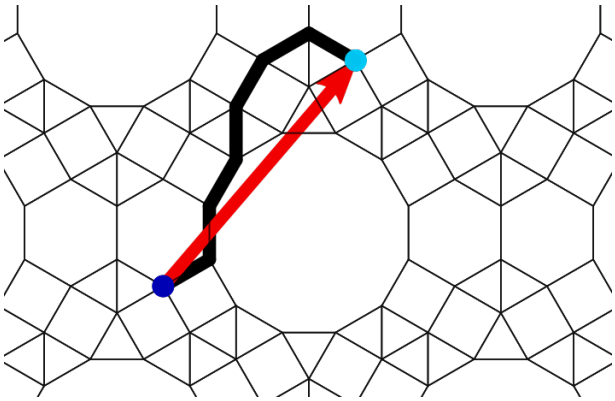


Figure 2.5: Translation vector $T$ as a polynomial in $\omega$.

## 2.3 *Integer quadruplets*

Vertices and translation vectors can both be represented as integer polynomials of $\omega$, denoted $\mathbb{Z}[\omega]$. We may observe, from the illustrated examples, that the paths chosen to reach the vertices are not unique: several polynomials represent the same vertex. However, polynomials in $\omega$ can be reduced mod $(\omega^4 - \omega^2 + 1)$, which is the minimal polynomial of $\omega$.[3]

If two polynomials $p(\omega)$ and $q(\omega)$ represent the same point, then $p(\omega) - q(\omega)$ would be a path that takes the origin to itself. Then $p(\omega) - q(\omega) = 0$, which means that the minimal polynomial in $\omega$ divides $p(\omega) - q(\omega)$. Therefore $p(\omega)$ and $q(\omega)$ reduce to the same cubic expression.

Thus, the set of all polynomials in $\omega$ with integer coefficients can be reduced to the set of polynomials in $\omega$ with integer coefficients of degree up to 3. This reduction provides unique polynomial labels for each vertex, at most cubic, in the form $a_0\omega^0 + a_1\omega^1 + a_2\omega^2 + a_3\omega^3$. The coefficients of this reduced polynomial are used as coordinates, which gives us a unique 4-integer representation for every vertex in the tiling. In particular,

$$\begin{aligned}
\omega^0 &= 1 && = [1,0,0,0]\,, & \omega^6 &= -1 && = [-1,0,0,0]\,, \\
\omega^1 &= \omega && = [0,1,0,0]\,, & \omega^7 &= -\omega && = [0,-1,0,0]\,, \\
\omega^2 &= \omega^2 && = [0,0,1,0]\,, & \omega^8 &= -\omega^2 && = [0,0,-1,0]\,, \\
\omega^3 &= \omega^3 && = [0,0,0,1]\,, & \omega^9 &= -\omega^3 && = [0,0,0,-1]\,, \\
\omega^4 &= -1+\omega^2 && = [-1,0,1,0]\,, & \omega^{10} &= 1-\omega^2 && = [1,0,-1,0]\,, \\
\omega^5 &= -\omega+\omega^3 && = [0,-1,0,1]\,, & \omega^{11} &= \omega-\omega^3 && = [0,1,0,-1].
\end{aligned}$$

This means we have a basis to represent every vertex in the tiling and both translation vectors with integer coefficients. The lattice

[3] $z^4 - z^2 + 1$ is the 12-th *cyclotomic polynomial*: the unique irreducible polynomial with integer coefficients that divides $z^{12} - 1$ but not $z^k - 1$ for $k < 12$. It is also the monic polynomial with integer coefficients of least degree having $\omega$ as a root.

generated by the polynomials in $\omega$ is denoted

$$\mathbb{Z}[\omega] = \mathbb{Z}1 + \mathbb{Z}\omega + \mathbb{Z}\omega^2 + \mathbb{Z}\omega^3.$$

Thus, the vertices of the tilings are plane projections of elements of a 4-dimensional integer lattice[4], and from now on we will refer to vertices and vectors by their unique representation in $\mathbb{Z}[\omega]$:

$$[a_0, a_1, a_2, a_3], \quad a_i \in \mathbb{Z},$$

which corresponds to $a_0 1 + a_1\omega + a_2\omega^2 + a_3\omega^3$ in the tiling. We call this 4-tuple the *lattice coordinates* of a point.

Taking the paths showed as examples in the last section and reducing their polynomials, we obtain the lattice coordinates for the vertex in Figure 2.4:

$$V - O = \omega^{11} + \omega^{10} + \omega^3 + \omega^2 + 2\omega + 1 = 2 + 3\omega = [2, 3, 0, 0],$$

and also for the translation in Figure 2.5:

$$T = \omega^{11} + 2\omega^3 + 2\omega^2 + 2\omega = 3\omega + 2\omega^2 + \omega^3 = [0, 3, 2, 1].$$

The mapping from $\mathbb{Z}[\omega]$ to the plane is defined in the natural way, interpreting each $\omega^k \in \mathbb{C}$ as an Euclidean point whose $(x, y)$ coordinates are the real and imaginary part, respectively, thus:

$$\omega^k \mapsto \left(\mathrm{Re}(\omega^k), \mathrm{Im}(\omega^k)\right) = \left(\cos\left(\frac{2\pi k}{12}\right), \sin\left(\frac{2\pi k}{12}\right)\right) \in \mathbb{R}^2.$$

Then, the mapping $\mathbb{Z}[\omega] \to \mathbb{R}^2$ is given by:

$$\begin{pmatrix} 1 \\ \omega \\ \omega^2 \\ \omega^3 \end{pmatrix} \to \begin{pmatrix} 1 & 0 \\ \sqrt{3}/2 & 1/2 \\ 1/2 & \sqrt{3}/2 \\ 0 & 1 \end{pmatrix} = W.$$

This way, to obtain the plane coordinates of an element in $\mathbb{Z}[\omega]$, we right-multiply its lattice coordinates $a = [a_0, a_1, a_2, a_3] \in \mathbb{Z}^4$ by matrix $W$, obtaining the corresponding point $aW \in \mathbb{R}^2$.

The key idea for representing tilings is that all we need to fully characterize and reproduce a tiling are the pair of translation vectors and the vertices inside the basic translation cell. Thus, a tiling is represented as a matrix of $(2 + n) \times 4$ integers in the $\omega$-basis representation, where $n$ is the number of vertices in the basic translation cell, the seeds.

It is important to note that not every element in $\mathbb{Z}[\omega]$ is a vertex of a tiling, because the vertices form a discrete subset of the plane, whereas $\mathbb{Z}[\omega]$ is dense in $\mathbb{C} \cong \mathbb{R}^2$. The vertices of a tiling form a discrete set of cosets of the translation lattice. The seeds are the representatives of these cosets.

Following, some examples of tilings with regular polygons and their integer lattice representation as a $(2 + n) \times 4$ matrix. The first two rows of the matrix represent the translation vectors that define the grid. In Figure 2.6 and Figure 2.7, we show a pair of tilings and their representation.

[4] In classic lattice theory, the notation for this is $\Lambda(1, \omega, \omega^2, \omega^3)$.

$$\left[\begin{array}{rrrr} 3 & 1 & -3 & -2 \\ 0 & 1 & 3 & 1 \\ \hline 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & -1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 2 & 0 & -1 & -1 \\ 1 & 0 & 1 & 0 \\ 2 & 1 & -2 & -2 \\ 1 & 1 & 0 & -1 \\ 0 & 1 & 2 & 0 \\ 2 & 1 & -1 & -2 \\ 2 & 1 & -1 & -1 \\ 1 & 1 & 1 & -1 \\ 1 & 1 & 1 & 0 \\ 0 & 1 & 3 & 0 \\ 1 & 1 & 2 & 0 \\ 2 & 1 & 1 & 0 \\ 3 & 1 & -1 & -1 \\ 2 & 1 & 1 & -1 \\ 3 & 1 & 0 & -1 \end{array}\right]$$

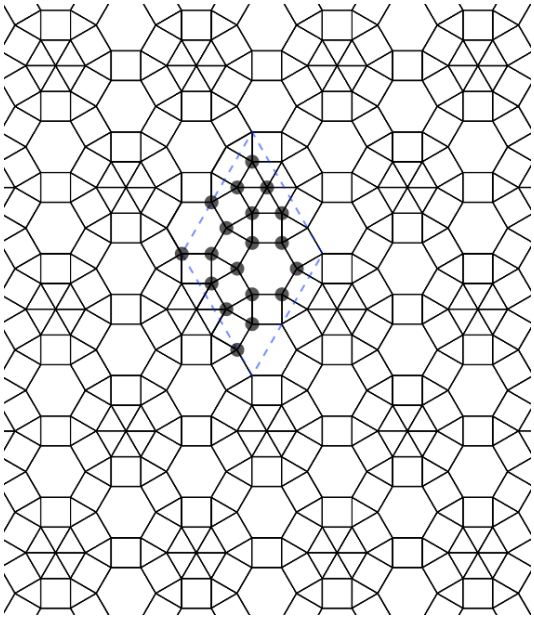
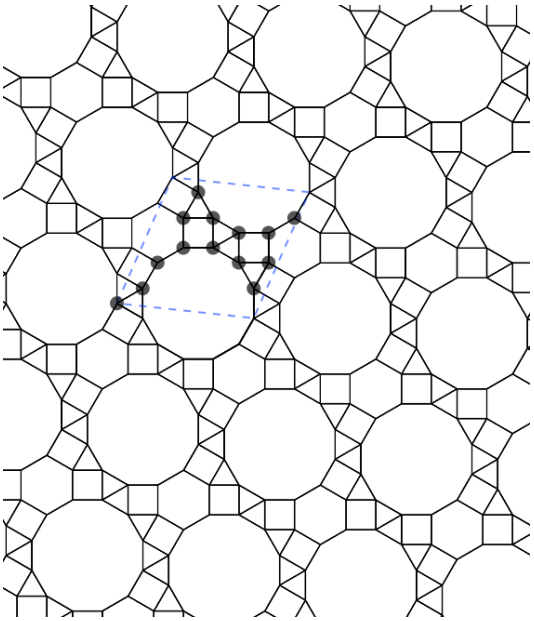


Figure 2.6: Example of a tiling and its representation.



$$\left[\begin{array}{rrrr} 0 & 1 & 2 & 2 \\ 2 & 3 & 0 & -2 \\ \hline 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 2 & 1 & 0 \\ 0 & 2 & 1 & 1 \\ 0 & 2 & 2 & 1 \\ 1 & 2 & 1 & 1 \\ 1 & 2 & 1 & 0 \\ 1 & 3 & 1 & 0 \\ 1 & 3 & 1 & -1 \\ 2 & 3 & 0 & -1 \\ 2 & 3 & 1 & -1 \\ 2 & 3 & 1 & 0 \\ 2 & 4 & 1 & 0 \end{array}\right]$$

Figure 2.7: Example of a tiling and its representation.

## 2.4 *Isometries in lattice coordinates*

Since $\mathbb{Z}[\omega]$ is isomorphic to the additive group $\mathbb{Z}^4$, we can operate algebraically on lattice coordinates with addition and multiplication by integer scalar in the usual way of a linear space:

$$[a_0, a_1, a_2, a_3] + [b_0, b_1, b_2, b_3] = [a_0 + b_0, a_1 + b_1, a_2 + b_2, a_3 + b_3],$$
$$k[a_0, a_1, a_2, a_3] = [ka_0, ka_1, ka_2, ka_3].$$

Isometries of the Euclidean plane play an important role in understanding tilings. They are useful in detecting symmetry and in computing equivalences, among other operations. We shall now give the explicit form they take in our representation.

*Translation.* Translations are obtained by simple addition in lattice coordinates. More precisely, given a translation vector $t = [t_0, t_1, t_2, t_3]$, the translation of a vertex $v = [v_0, v_1, v_2, v_3]$ is given by

$$v + t = [v_0 + t_0, v_1 + t_1, v_2 + t_2, v_3 + t_3]$$

*Rotation.* Rotations are given by multiplication in $\mathbb{C}$. Since our edges are always aligned with the basic directions, we are only interested in rotations of multiples of $\pi/6$ (30°). This means we only consider multiplications by powers of $\omega$, which are, of course, iterated multiplications by $\omega$. This way, rotation by $\omega$ in lattice coordinates is:

$$\begin{aligned}\omega\left(a_0 + a_1\omega + a_2\omega^2 + a_3\omega^3\right) &= \left(a_0\omega + a_1\omega^2 + a_2\omega^3 + a_3\omega^4\right)\\ &= \left(-a_3 + a_0\omega + (a_1 + a_3)\,\omega^2 + a_2\omega^3\right),\end{aligned}$$

where we have replaced $\omega^4$ by $-1 + \omega^2$. Thus, in lattice coordinates $R_\omega(a) = [-a_3, a_0, a_1 + a_3, a_2]$, or in matrix form:

$$R_\omega(a) = a\mathcal{R} = a\begin{pmatrix} 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\\ -1 & 0 & 1 & 0\end{pmatrix}.$$

Rotations by $\omega^k$ are given by the powers of matrix $R$ directly, $R_{\omega^k}(a) = a\mathcal{R}^k$ or, in the opposite direction, by its inverse: $a\mathcal{R}^{-k}$. The rotation of a point $p \in \mathbb{Z}[\omega]$ by $\omega^k$ around vertex $v$ is given by a conjugation with the translation of $v$ to the origin: $(p - v)\mathcal{R}^k + v$.

*Reflection.* The two basic reflections are $\mathcal{M}_0$ across the horizontal line, and $\mathcal{M}_{1/2}$ across the bisector of 1 and $\omega$, the line with 15 degree slope through the origin. All other reflections in $\mathbb{Z}[\omega]$ can be reduced to one of these two by rotations.

The reflection $\mathcal{M}_0$ across the horizontal line has the following effect on the basis:

$$\begin{pmatrix}1\\ \omega\\ \omega^2\\ \omega^3\end{pmatrix} \xrightarrow{\mathcal{M}_0} \begin{pmatrix}1\\ \omega^{11}\\ \omega^{10}\\ \omega^9\end{pmatrix} = \begin{pmatrix}1\\ \omega - \omega^3\\ 1 - \omega^2\\ -\omega^3\end{pmatrix}.$$

Thus, in lattice coordinates we have

$$\mathcal{M}_0\left([a_0, a_1, a_2, a_3]\right) = [a_0 + a_2, a_1, -a_2, a_1 - a_3]\,,$$

or in matrix form:

$$\mathcal{M}_0(a) = a\mathcal{M}_0 = a\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}.$$

The reflection $\mathcal{M}_{1/2}$ across the 15-degree line has the following effect on the basis:

$$\begin{pmatrix} 1 \\ \omega \\ \omega^2 \\ \omega^3 \end{pmatrix} \xrightarrow{\mathcal{M}_{1/2}} \begin{pmatrix} \omega \\ 1 \\ \omega^{11} \\ \omega^{10} \end{pmatrix} = \begin{pmatrix} \omega \\ 1 \\ \omega - \omega^3 \\ 1 - \omega^2 \end{pmatrix}.$$

Thus, in lattice coordinates we have

$$\mathcal{M}_{1/2}\left([a_0, a_1, a_2, a_3]\right) = [a_1 + a_3, a_0 + a_2, -a_3, -a_2]\,,$$

or in matrix form:

$$\mathcal{M}_{1/2}(a) = a\mathcal{M}_{1/2} = a\begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 \\ 1 & 0 & -1 & 0 \end{pmatrix} = a\mathcal{M}_0\mathcal{R}.$$

A reflection of a point $p \in \mathbb{Z}[\omega]$ across the $\omega^k$ direction through a vertex $v$ reduces to the horizontal reflection through the origin by conjugation:

$$(p - v)\mathcal{M}_k + v = (p - v)\mathcal{R}^{-k}\mathcal{M}_0\mathcal{R}^k + v.$$

A reflection of a point $p \in \mathbb{Z}[\omega]$ across the bisector of $\omega^k$ and $\omega^{k+1}$ through a vertex $v$ reduces to the $\mathcal{M}_{1/2}$ reflection by conjugation:

$$(p - v)\mathcal{M}_{k+1/2} + v = (p - v)\mathcal{R}^{-k}\mathcal{M}_0\mathcal{R}^{k+1} + v.$$

## 2.5 *Geometry in lattice coordinates*

Besides the isometries, the matrix multiplication mapping from $\mathbb{Z}[\omega]$ to $\mathbb{R}^2$ allows us to define some Euclidean metric operations that can be applied in a compact way over the lattice coordinates.

Let $(x_a, y_a) = aW$ be the Euclidean coordinates of an element in $\mathbb{Z}[\omega]$ with lattice coordinates $a$. We maintain the row-vector result from the multiplication by matrix $W$ to avoid overloading the notation with transposition. With this in mind, let us analyze the main metric operations in the plane:

*Dot product.* The dot product between two row vectors is simply the multiplication of the first by the second transposed, this way, the dot product in lattice coordinates is given by:

$$(x_a, y_a) \cdot (x_b, y_b) = aW \cdot (bW)^T = aWW^T b^T = aQb^T$$

$$Q = \begin{pmatrix} 1 & \sqrt{3}/2 & 1/2 & 0 \\ \sqrt{3}/2 & 1 & \sqrt{3}/2 & 1/2 \\ 1/2 & \sqrt{3}/2 & 1 & \sqrt{3}/2 \\ 0 & 1/2 & \sqrt{3}/2 & 1 \end{pmatrix}.$$

Since matrix $W$ has rank 2, matrix $Q$ is singular. $Q$'s null-space is generated by $\{(\sqrt{3}, -2, 0, 1), (1, -\sqrt{3}, 1, 0)\}$. This has no effect on the direct use of the dot product directly in $\mathbb{Z}[\omega]$ points, since all the vectors are in the integers, but impedes the use of some geometric dot-product applications, such as projection.

*Norm.* The expression of the norm in lattice coordinates is derived directly from the dot product:

$$\left\|(x_a, y_a)\right\|^2 = aQa^T.$$

*Area.* The area of a parallelogram given by two vectors in $\mathbb{R}^2$ is the absolute value of the determinant of the matrix containing the coordinates of the two vectors. When the vectors are in $\mathbb{Z}[\omega]$, the area can be computed directly from their lattice coordinates. Let $v_i = a_i W$, for $i = 1, 2$, and $\mathcal{O}$ a rotation by $-\pi/2$ in the Euclidean plane, which generates an orthogonal vector, then:

$$\begin{aligned} \det \begin{pmatrix} v_1 \\ v_2 \end{pmatrix} &= \begin{vmatrix} v_{1x} & v_{1y} \\ v_{2x} & v_{2y} \end{vmatrix} \\ &= v_{1x} v_{2y} - v_{1y} v_{2x} \\ &= (v_{1x}, v_{1y})(v_{2y}, -v_{2x})^T \\ &= (v_{1x}, v_{1y}) \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} (v_{2x}, v_{2y})^T \\ &= a_1 W \mathcal{O} W^T a_2^T \\ &= a_1 \begin{pmatrix} 0 & 1/2 & \sqrt{3}/2 & 1 \\ -1/2 & 0 & 1/2 & \sqrt{3}/2 \\ -\sqrt{3}/2 & -1/2 & 0 & 1/2 \\ -1 & -\sqrt{3}/2 & -1/2 & 0 \end{pmatrix} a_2^T \\ &= a_1 Q^{\perp} a_2^T. \end{aligned}$$

Anti-symmetric matrix $Q^{\perp}$ has the same null space as $Q$, and the area of the translation cell given by vectors $a_1$ and $a_2$ is given by the absolute value of the determinant[5]

$$A(a_1, a_2) = |a_1 Q^{\perp} a_2^T|.$$

These are the more general geometric operations in $\mathbb{Z}[\omega]$. Their representation in lattice coordinates, and, whenever possible, as matrix multiplications, translates into ease of implementation in computer code.

[5] The resulting area is a number of the form $|a + b\sqrt{3}/2|$, a unique representation up to sign because $\sqrt{3}$ is irrational. This fact will be useful in the future.

## 2.6 *Translation lattices*

Lattices have several representations, for example, in Figure 2.8 we see the underlying lattice for a tiling and several other valid translation lattices for the same tiling. Blue- and orange-colored lattices are equivalent; however, they are not minimal, while red, purple and green lattices are equivalent to the main lattice, the gray grid. Any of these could be used as translations of the tiling, just by updating the coordinates of the seeds, generating new symbols for the same tiling.

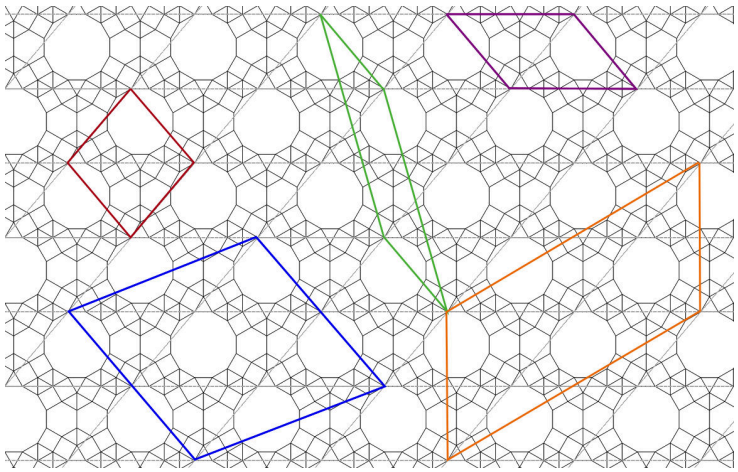


Figure 2.8: Equivalent lattices over a tiling.

Let $t_1, t_2 \in \mathbb{R}^2$ be a pair of linearly independent vectors for a translational lattice over $\mathbb{Z}[\omega]$, then $t_i = [a_{i1}, a_{i2}, a_{i3}, a_{i4}]\, W$, and the translation vectors can be written in a $2 \times 2$ matrix $T = AW$:

$$\begin{pmatrix} t_1 \\ t_2 \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \end{pmatrix} W$$

Two pairs of translation vectors $T$ and $T'$ determine the same lattice iff there is a $2 \times 2$ *unimodular*[6] matrix $U$ such that $T' = UT$ [13]. Write $T = AW$ and $T' = A'W$ as above, then, $T' = UT$ iff $A' = UA$, because the basic directions in $W$ are linearly independent over $\mathbb{Z}$. Multiplication by matrix $U$ does not affect the determinant and hence neither the area of any of the different cells determined by distinct pairs of translations.

[6] A *unimodular* matrix is an integer matrix with determinant $\pm 1$, or equivalently, an integer matrix with integer inverse.

There is a particular matrix that will serve as reference to a translation lattice, the *Hermite normal form*.[7] If $A$ is an $m \times n$ integer matrix, then there is a unimodular $m \times m$ matrix $U$ such that $H = UA$ is an $m \times n$ integer matrix in row-reduced form, called the Hermite normal form of $A$.

[7] The Hermite normal form is the integer equivalent to a reduced echelon form and can be found with an integer Gaussian elimination process.

This way, in order to be able to identify if two translations define the same lattice, we would first verify if they both have the same area, and after that, we would compare their Hermite normal form.

Given a translation lattice $T = AW$, we can identify if a vertex $a \in \mathbb{Z}[\omega]$ is inside the basic translation cell simply by verifying the solution of $(\lambda_1, \lambda 2) = (aW)T^{-1}$ is such that $\lambda_i \in [0, 1)$, $i = 1, 2$. In fact, $(\lambda_1, \lambda_2)$ indicate the vertex is in the cell

$$T_{\lfloor \lambda_1 \rfloor, \lfloor \lambda_2 \rfloor} = T_{0,0} + \lfloor \lambda_1 \rfloor t_1 + \lfloor \lambda_2 \rfloor t_2.$$

## 2.7 *Representation of periodic tilings with regular polygons*

The following theorem estates in a constructive way the conditions that must hold for a symbol with elements in $\mathbb{Z}[\omega]$ to represent a tiling of the plane with regular polygons.

*Theorem.* Let

$$\begin{bmatrix} T \\ S \end{bmatrix} = \begin{bmatrix} t_1 \\ t_2 \\ s_1 \\ \vdots \\ s_n \end{bmatrix}$$

be a $(2+n)\times 4$ matrix. $T$ is a $2\times 4$ integer matrix representing the translation lattice in $\mathbb{Z}[\omega]$. $S$ is an $n\times 4$ integer matrix with the lattice coordinates of $n$ seeds: $\{s_i \mid i = 1,\ldots,n\}$, representatives of translation equivalence classes of vertices in $\mathbb{Z}[\omega]$.

Let $V = \{s_i + k_1 t_1 + k_2 t_2 \mid i = 1,\ldots,n,\ k_1,k_2 \in \mathbb{Z}\}$ be the set of all the vertices of the tiling.[8] Then, $(T,S)$ is a valid *tiling symbol*, that is, corresponds to a periodic tiling of the plane with regular polygons iff all the following conditions hold:

a) Matrix $T$ represents a translation lattice: its rows are linearly independent.

b) Every element in $V$ is unique.

c) The distance between two distinct vertices in $V$ is at least 1.

d) Every two vertices at distance exactly 1 form an edge aligned with a direction in $\Omega = \{\omega^k \mid k = 0,\ldots,11\}$.

e) Let the edges $\overline{pq}$ in direction $\omega^k$ and $\overline{pr}$ in direction $\omega^{k+j}$, with $j = \pm 2, \pm 3, \pm 4, \pm 5$, be in the same face. Then, this relation is transitive, meaning that for an edge $\overline{qs}$ in direction $\omega^h$, the immediate vertex-neighbouring edge $\overline{qp} = -\omega^k$ is at $\omega^{h+j}$ (Figure 2.9).[9]

f) Every vertex in $V$ is a vertex of a polygonal face, and all the vertices of each polygonal face are vertices in $V$.

g) Every point of the fundamental region belongs to a polygonal face.

*Proof* ($\Rightarrow$) Let $\mathcal{T}$ be a periodic tiling of the plane with regular polygons with unit polygon sides and one edge aligned with some element in $\Omega$. We have shown in the previous sections that:

- All vertices and both translation vectors of $\mathcal{T}$ are elements of $\mathbb{Z}[\omega]$.
- A translation lattice exists, it is origin independent, and defines a half-open fundamental region which contains a finite set of representatives of vertices of $\mathcal{T}$, the seeds.
- The polygonal faces are formed by sequences of directions in $\Omega$.

[8] For computer implementations, this set is typically a finite *point cloud* easily generated with $k_1, k_2 \in \{-K,\ldots,K\}$ for some *small* $K \in \mathbb{N}$.

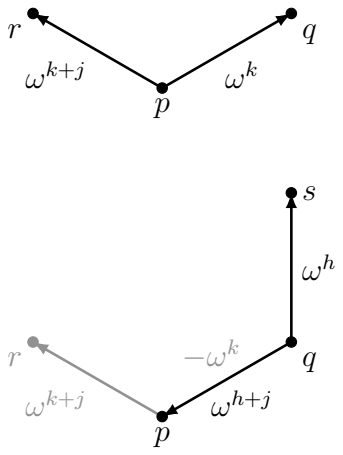



Figure 2.9: Transitive property of neighboring edge positions (e), the difference in the $\omega$ exponent $j$ between consecutive pairs of them is constant.

[9] As observed before, the value of $j$ determines both edges belong to a regular polygonal face of $3, 4, 6, 12$ sides, respectively.

Then, (a), (b) and (g) derive from the fact that the translation vectors define a lattice. (f) is derived directly from $\mathcal{T}$ being a polygonal tiling. (c), (d) and (e) are guaranteed, since the faces of $\mathcal{T}$ are regular polygons with unit length sides that are aligned with directions in $\Omega$. Therefore, conditions (a) to (g) hold.
($\Leftarrow$) Now suppose we have a symbol $(T, S)$ as described in the theorem and show it is a periodic tiling of the plane with regular polygons.

To start, if there was a point $P$ in the plane which does not belong to a face of the tiling, there would be a translation $t = (k_1 t_1 + k_2 t_2)W$, $k_i \in \mathbb{Z}$ such that $P + t$ ends up in the fundamental region, and would contradict condition (g), hence, the faces cover the plane.

The periodicity of the symbol is given by the translation lattice and the construction of the set $V$. Let us show that the polygonal faces intersect either in an edge or in a vertex.

If there was a vertex of one face inside any edge, this point would be at a distance less than 1 of the vertices of such an edge. The same is the case if any two edges intersect, the vertices of one such edge would be closer than 1 to the vertices of the other. Loose single vertices or single edges inside a polygon would contradict conditions (e), (f) and (g). And, overlapping vertices would contradict condition (b) and (c).

A loose corner of a smaller polygon inside another would either: have a vertex closer than 1 to a vertex of the surrounding polygon, be a loose corner of the interior polygon contradicting condition (e), or be at exact distance 1 of a vertex of the surrounding polygon, which would contradict the existence of this polygon, since the edges forming it are not immediate neighbors, contradicting condition (e).

We have shown that, in fact, the object described by the symbol covers the plane with edge-to-edge regular polygons in a periodic way. □

Condition (e) is a fundamental geometric rigidity characteristic of Archimedean vertices: edges around a vertex form a *star*-shaped structure, a footprint of the vertex type (Figure 1.12).

We will translate this theorem into practical algorithms and methods for acquiring, analyzing and building periodic tilings of the plane with regular polygons.

## 2.8 *Tiling the plane using a symbol*

The tiling is an infinite object, but, since all the information is contained in a finite region represented by the symbol, we will explain in detail how it is that the formulated conditions can be verified in an efficient way to build a tiling.

The $(2 + n) \times 4$ integer matrix is our *concrete representation* of a tiling. Each data item (translation vector or seed) is given by its lattice coordinates in $\mathbb{Z}[\omega]$, and so is represented uniquely by four integers. We also call it the external representation, because it can be saved to files, published, and shared.

We will formulate constructive algorithms for creating the tiling from the elements in the symbol. These algorithms serve at the same time to check the validity of the symbol.

### 2.8.1 *Covering a region of interest with a system of points*

The first challenge to cover an arbitrary region $R \subset \mathbb{R}^2$ of the plane with the tiling is to determine the coefficients for the translation lattice over which we should compute *replicas* of the fundamental region that cover the desired region (Figure 2.10). This is a known problem in computer graphics related to *rendering* textures.

The lattice coordinates of the translation vectors can be arranged in a $2 \times 4$ matrix $T$, and mapped into $\mathbb{R}^2$ to form a $2 \times 2$ invertible matrix $T_e = TW$. The region $R$ is transformed into the translation lattice space through $T_e^{-1}$. Here, coefficients that correspond to the cover of the region by translation cells (index sets $I$, $J$) are computed to guarantee that

$$R \subseteq \bigcup_{I \times J} T_{i,j},$$

where $T_{i,j}$ is the translated lattice cell:

$$\begin{aligned} T_{i,j} &= \{(i+\lambda_1)t_1 + (j+\lambda_2)t_2 : \lambda_1, \lambda_2 \in [0,1)\} \\ &= T_{0,0} + it_1 + jt_2 = T_{0,0} + (i,j)T. \end{aligned}$$

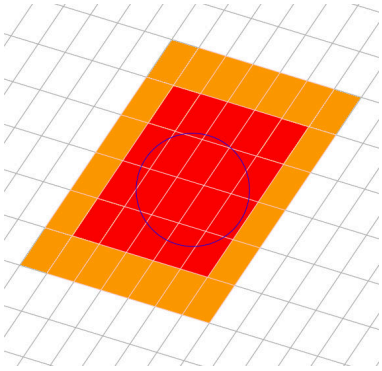


Figure 2.10: Covering a circular region with a tiling. The red area is the coverage. The orange region helps reconstructing the topology over the red region.

For completely reconstructing the tiling topology, a ring around the cover is needed, as illustrated in Figure 2.10. This will be explained in detail in the next subsection. These indices are included in sets $I$ and $J$ for convenience.

That said, the set of vertices we described as part of the representation is defined in a finite form, covering region $R$:

$$V_R = \{s_i + k_1 a_1 + k_2 a_2 \,|\, i = 1, \ldots, n,\ k_1 \in I,\ k_2 \in J\}$$

### 2.8.2 *Hashing*

Since each vertex in the set has an integer representation in $\mathbb{Z}[\omega]$ we use *hashing*, a technique that lets us locate a vertex by its coordinates in a data structure indexed by a unique code generated from the integer representation. The hash key of an element $[1,2,3,4] \in \mathbb{Z}[\omega]$ may be the string `"1,2,3,4"` or an integer in the form `0x0001000200030004`. This allows the algorithms that rely on comparisons with the whole point cloud to take linear time.

The *internal representation* is enough to deduce the tiling topology entirely and condenses the view of the tilings as systems of points introduced at the beginning of the chapter. For applications where $R$ is not defined, we use index sets $I = J = \{-1, 0, 1\}$, just to get the translation cells adjacent to the fundamental region, which allows us a proper topology reconstruction.

### 2.8.3 *Tiling topology reconstruction*

*Edges.* An edge exists between any two vertices at distance one in the point cloud. Computing the distance between every pair of vertices would take $O(N^2)$ operations, given $N = |V_R|$. Instead of this we take each vertex in $v \in S$ and ask whether $v + \omega^k \in V$ for each $k = 0, 1, \ldots, 11$. This can be done directly by testing whether an entry for $v + \omega^k$ exists in the hash.

The *star* for each vertex is a list that stores the relative position of the neighboring vertices by the exponents of their directions in $\Omega$ (Algorithm 1). Then, the *star* of a vertex $v \in V_R$ is simply an ordered list of the form:

$$\text{star}(v) = \{k : v + \omega^k \in V\} \subset \{0, 1, \ldots, 11\}$$

Note that this is done only for the $n$ *seeds* (vertices in $S$), looking for neighbors in $V_R$, which costs only $12n$ hash queries. Once the stars are obtained for the seeds, they can be propagated using the translation cells over $I \times J$.

The stars replace the edges in a classic mesh topology representation, since they allow the location of all the neighbors of a vertex via the hash. Rendering the edges of the tiling can be done in parallel to the star computation or after, without the need to explicitly store the edges. For each $k \in \text{star}(v)$, we know $w = v + \omega^k \in V_R$, and, to avoid duplicating the edge drawing, the edge $\overline{vw}$ is drawn only when $\text{hash}(v) < \text{hash}(w)$ (Algorithm 2).

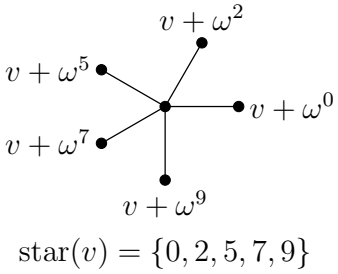



Figure 2.11: Example of a star around a vertex in a tiling.

**Algorithm 1**

**procedure** *star*$(v)$
  $s \leftarrow [\,]$
  $j \leftarrow 0$
  **for** $k = 0$ **to** 11 **do**
    **if** $v + \omega^k \in V$ **then**
      $j \leftarrow j + 1$
      $s[j] \leftarrow k$
    **end**
  **end**
  **return** $s$
**end**

**Algorithm 2**

**procedure** *edges*$(v)$
  **for** $k = 0$ **to** 11 **do**
    $w \leftarrow v + \omega^k$
    **if** $w \in V$ **and** $\text{hash}(v) < \text{hash}(w)$ **then**
      *drawedge*$(v, w)$
    **end**
  **end**
**end**

*Faces.* Our representation offers a very simple way to reconstruct the faces around each vertex. As we have stated, each vertex is the vertex of a regular polygon, each corner of a star is the corner of a regular polygon, and the difference between the exponents of the directions in $\Omega$ stored in the vertex star reveals the type of polygon.

Moreover, regular polygons are sequences of constant turns, then, multiplications by some $\omega^h$ suffice, and there is a face in a vertex $v$ for each element in its star (Figure 2.12). So, let $\sigma(v) = \{k_i \mid i = 1, \ldots, e_v\}$, let $j_i = k_{i+1} - k_i$ for $i < e_v$ and $j_{e_v} = (k_1 - k_{e_v}) \bmod 12$, and $h_i = 6 - j_i$. Then, the $i$-th face adjacent to $v$ has $s_i = {}^{12}/_{h_i}$ sides and is given by

the sequence of points (Algorithm 3):

$$\begin{aligned} p_1 &= v \\ p_2 &= p_1 + \omega^{k_i} \\ p_3 &= p_2 + \omega^{k_i+h_i} \\ &\vdots \\ p_{s_i} &= p_{s_i-1} + \omega^{k_i+(s_i-2)h_i} \end{aligned}$$

Remember that if $j_i = 2,3,4,5$, then the polygon has $s_i = 3,4,6,12$ number of sides respectively ($s_i = {}^{12}/_{(6-j_i)}$). Since $\omega$ is a rotation by 30 degrees, internal angles are obtained by $\alpha_i = 30j_i$, that is, $\alpha_i = 60,90,120,150$, respectively. This is the reason we use $h_i = 6 - j_i$ as angle for the turns of the polygon, because it corresponds to $180 - \alpha_i$ degrees external turns.

The sequence $\{j_i\}$ of differences between the values of the star of $v$ also reveals the type of Archimedean vertex of $v$ when translated to the sequence $\{s_i\}$. Sequence $\{s_i\}$ corresponds to the number of sides of the polygons around $v$, which is, in turn, the symbol for the Archimedean vertices (Figure 1.12).

We should not keep all the faces generated by this procedure, however, because this would create duplicate faces. The face is only going to be computed when $v$ is the leftmost lowest vertex of the face, this is, when $0 \leq k_i, k_{i+1} < 6$, since $k_{i+1} \geq 6$ makes $v + \omega^{k_{i+1}}$ leftmost and at least as low as $v$ in the same face. If $v$ is such that $0 \leq k_i, k_{i+1} < 6$, we call $v$ the *anchor* of the face (Algorithm 4). Every face has exactly one anchor but not every vertex is an anchor of a face. Once edges and faces have been computed for the fundamental region, they can be replicated using the translation cells over $I \times J$.

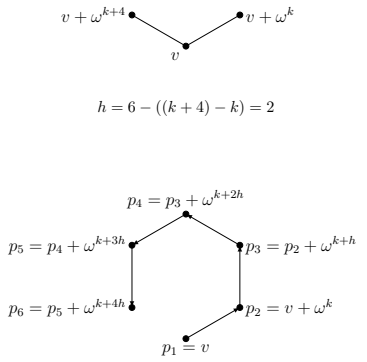



Figure 2.12: Example of a face reconstruction on an anchor vertex using the star difference $h$.

**Algorithm 3**

**procedure** *face*$(v, k, m)$
  $f \leftarrow [\,]$
  $f[1] \leftarrow v$
  $f[2] \leftarrow v + \omega^k$
  **for** $i = 3$ **to** $m$ **do**
    $k \leftarrow (k + {}^{12}/_m) \bmod 12$
    $f[i] \leftarrow f[i-1] + \omega^k$
  **end**
  **return** $f$
**end**

**Algorithm 4**

**procedure** *faces*$(v)$
  $s \leftarrow [\,]$
  $j \leftarrow 0$
  **for** $k = 0$ **to** $5$ **do**
    **if** $v + \omega^k \in V$ **then**
      $j \leftarrow j + 1$
      $s[j] \leftarrow k$
    **end**
  **end**
  **if** $\#s > 1$ **then**
    **for** $i = 1$ **to** $(\#s - 1)$ **do**
      $h \leftarrow 6 - (s[i+1] - s[i])$
      $m \leftarrow {}^{12}/_h$
      *drawface*$(face(v, s[i], m))$
    **end**
  **end**
**end**

### 2.8.4 *Rendering*

Everything that happens in the basic cell is repeated exactly in all the cells of the translation grid. This is the reason why one only reconstructs the tiling around the seeds, that is, take the basic cell as the window. The set of all edges emanating from the seeds is called the *skeleton* of the tiling. The set of all faces anchored at the seeds form a *patch* of the tiling (Figure 2.13). These fundamental pieces are *translation units*: they can be repeated by translations to reconstruct the whole tiling. When this local reconstruction is used for rendering only, it does not require storing all vertices in the point cloud, just the ones in the basic cell and in its outer cells, nine cells in all.

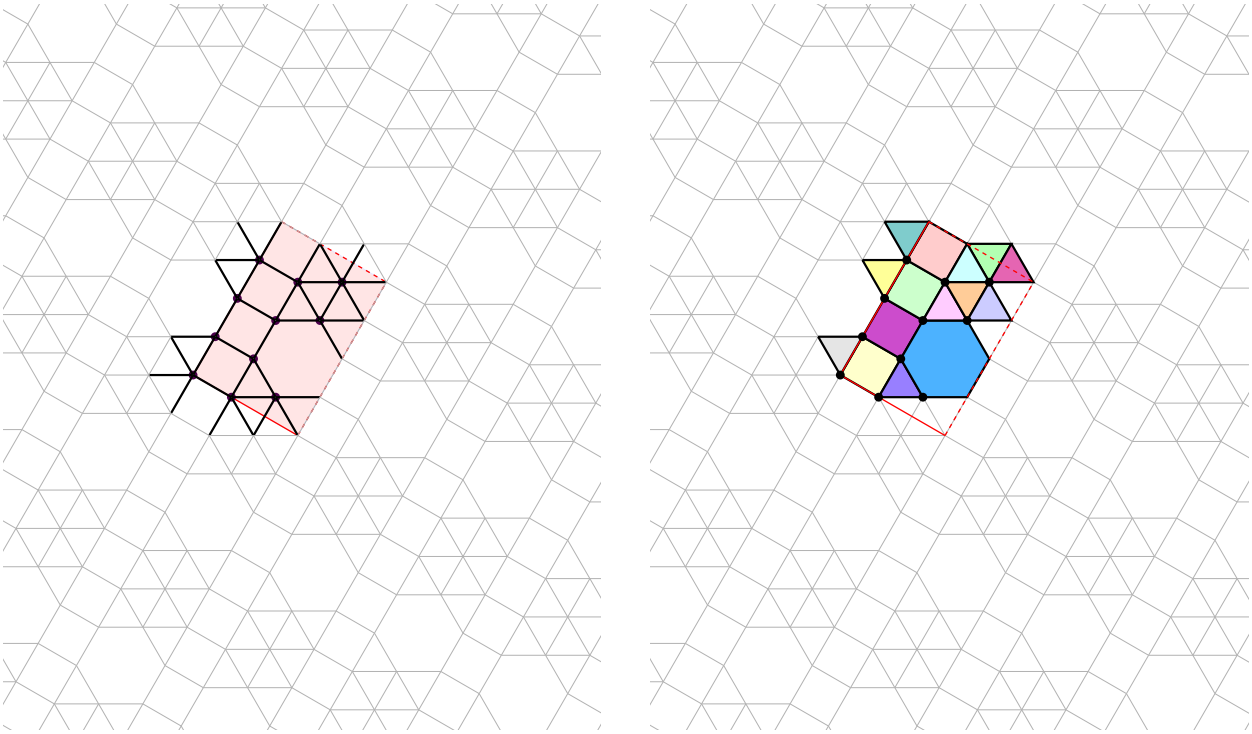


Figure 2.13: Example of the skeleton and the patch of a tiling.

In some computer graphics applications we address the following problem: given a point $p = (p_x, p_y)$, to obtain its lattice coordinates and reduce it to an equivalent point in the fundamental region. To solve it, we solve the system:

$$(\lambda_1, \lambda_2)\, TW = (p_x, p_y)\,.$$

The integer parts of $\lambda_1$ and $\lambda_2$ are the translation lattice coordinates, and the fractional parts are the convex coordinates inside the fundamental region. Once there, we can find the closest seed $s$ to the point and locate the position of $p$ relative to the star of $s$, which gives us the face in which the point is contained.

### 2.8.5 *Dual*

The dual vertices are the centers of the faces of the primal tiling. Two vertices of the dual are connected by an edge iff the two corresponding faces are adjacent in the primal tiling (Figure 2.14). The dual of a tiling with regular polygons is not Archimedean necessarily; in fact, that is the case only for the three regular tilings of triangles, squares and hexagons. The dual of the 11 Archimedean tilings are known as Laves or Catalan tilings [35].

The center of a face can be computed as the barycenter of its vertices or as the midpoint of a pair of opposite vertices in the face, when the face is not a triangle. The vertices of the dual tiling can be written explicitly for each of the polygons in (fractional) lattice coordinates.

The vertex of the dual tiling corresponding to the face anchored at vertex $v$ with first start value $k$ (edge $\omega^k$) is, in each case:

*Triangle:* $v + \frac{1}{3}\omega^{k+1}$

*Square:* $v + \frac{1}{2}\omega^k(1 + \omega^3)$

*Hexagon:* $v + \omega^{k+2}$

*Dodecagon:* $v + \omega^k(\omega^2 + \omega^3)$

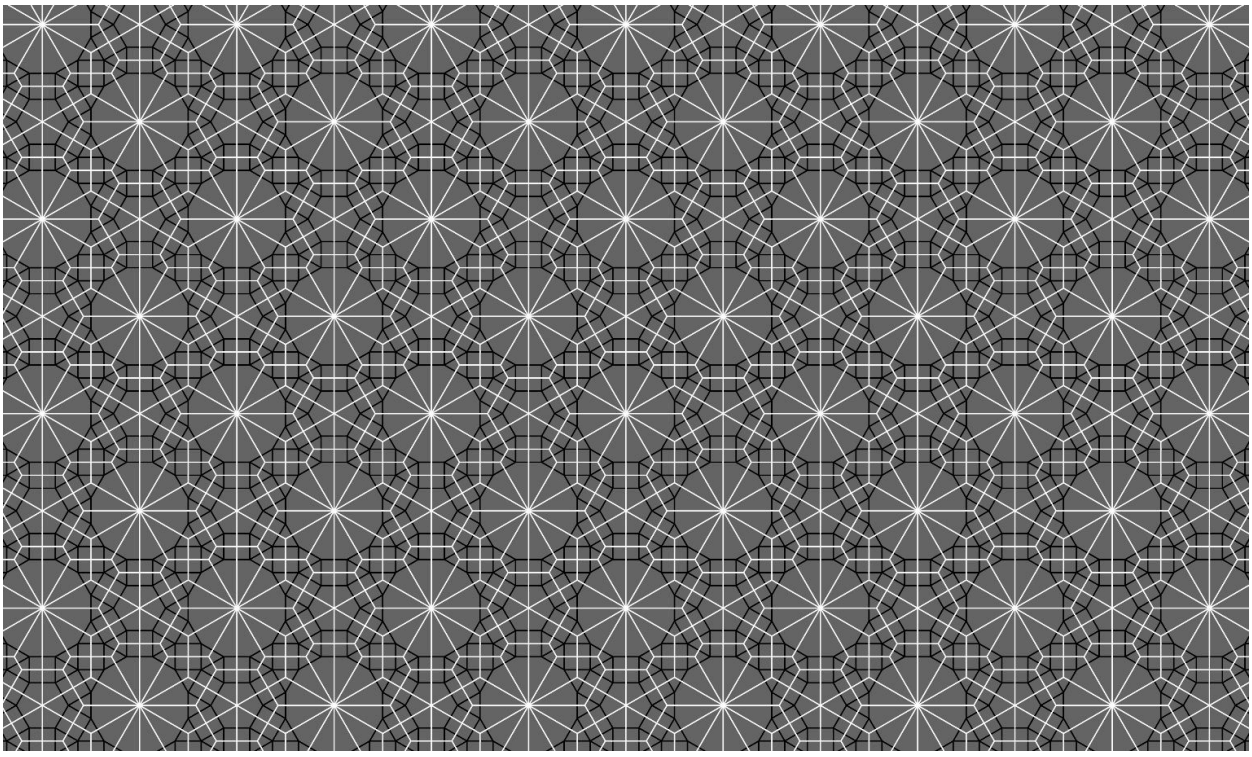


Figure 2.14: Example of the dual (white) of a tiling (black).

## 2.9 *Symbol equivalence*

One of the important questions we can answer with our representation is whether a given pair of tiling symbols are equivalent, that is, represent the same tiling.

Let $(T, S)$ and $(T', S')$ be two tiling symbols (Section 2.7) and let us assume that both $T$ and $T'$ are minimal translation lattices for the tiling they represent.[10] Let $V = \{s_i + k_1 t_1 + k_2 t_2 \mid i = 1, \ldots, n;\ k_1, k_2 \in \mathbb{Z}\}$. If the symbols represent the same tiling then all the following conditions hold:

- There is a $2 \times 2$ unimodular matrix $U$, such that $T = UT'$. Equivalently, the Hermite normal form of $T$ is the same of that of $T'$.
- $\#S = \#S'$, and there is some $o \in S'$ such that the image of $S'$ under $U$ with origin $o$ is contained in $V$.

The matrix $U$ accounts for any rotation or reflection to which the common lattice would be subject to in the second representation. There would be ambiguity by a reflection only in case the fundamental region itself is symmetric by a reflection across $t_1 + t_2$.

The point $o$ in the second condition might be any point of $S'$, since the translation lattice is origin independent.

The second condition could be formulated with reference to both fundamental regions only, but this would require remapping both sets $S$ and $S'$ to a new common translation lattice given by the Hermite normal form, and also a remapping after each translation by a given origin.

[10] We address the translation lattice minimization in Section 3.3.

## 2.10 *Properties of the representation*

We shall now discuss some properties of the representation of tilings with regular polygons explained here. As a guideline, we use the framework proposed by Requicha [71] for discussing properties of representation schemes. A *representation scheme* is a relation from a model space $M$ of mathematical objects to a representation space $R$ of syntactically correct representations. In other words, a representation scheme is a set of pairs $(m, r) \in M \times R$. In our case, the model space $M$ is the space of all periodic tilings of the plane by regular polygons, up to Euclidean similarity. The representation space $R$ is the set of all $(2 + n) \times 4$ integer matrices, with $n \in \mathbb{N}, n \geq 1$. Such matrices are interpreted according to the representation defined in Section 2.7. Each row contains lattice coordinates of a point in $\mathbb{Z}[\omega]$; the first two rows represent the translation vectors; the remaining $n$ rows represent the seeds.

*Domain.* The domain of a representation scheme is the set of all objects in $M$ that can be represented in $R$. The domain characterizes the descriptive power of the scheme. As we have argued in Section 2.7, all tilings can be represented in our scheme after applying an Euclidean similarity for normalization. Therefore, the domain of our

representation is the full modeling space of all periodic tilings of the plane by regular polygons, up to Euclidean similarity. In this sense, our representation is *comprehensive*.

*Validity.* The set of valid representations is the range of the representation scheme, that is, the set of all representations in $R$ that represent an object in $M$. Each representation is a $(2+n) \times 4$ integer matrix. This matrix corresponds to a finite non-empty set of cosets of an additive subgroup of $\mathbb{Z}[\omega]$ of rank at most 2: the first two rows generate a subgroup of $\mathbb{Z}[\omega]$ and the remaining $n$ rows are coset representatives. A valid representation is a matrix that represents a tiling. Not every representation is valid, not every matrix corresponds to a tiling, only the ones that satisfy the conditions listed in the representation theorem. These conditions are verifiable with the algorithms given in the other sections in this chapter.

*Completeness.* A representation $r$ in $R$ is *unambiguous* when there is exactly one object $m$ in $M$ that is represented by $r$. A representation scheme is *complete* or *unambiguous* when all valid representations are unambiguous. Our representation for tilings is complete in this sense because a tiling can be unambiguously reconstructed from a matrix that represents it, if the matrix does represent a tiling.

*Uniqueness.* A representation scheme is *unique* when all valid representations are unique: every object in $M$ has exactly one representation in $R$. As remarked by Requicha [71], most representation schemes for geometric objects are not unique for conceptually trivial reasons: duplicates and permutations of elements listed in the representation (when lists are used as proxies for sets) and position of geometric elements. Our representation for tilings is subject to this trivial non-uniqueness: the order of the translation vectors and of the seeds in the matrix is conceptually immaterial but give different matrices. (Our abstract representation does not suffer from this non-uniqueness because it uses sets, but it cannot be used as a representation space.)

Representing a tiling has various sources of non-uniqueness: the vertex chosen to be the origin and the orientation of the tiling with respect to the basic directions give different matrices. This is an inevitable consequence of defining $M$ as the space of tilings up to Euclidean similarity.

Our representation suffers from non-uniqueness also for nontrivial algebraic reasons: the same translation lattice can be represented by different pairs of vectors. We have addressed how to establish the equivalence between two symbols in Section 2.9, using a unitary matrix $U$ that must exist if both translation lattices are equivalent. What we have described allows to solve the equivalence of the translation lattice.

Finally, there is the issue of *minimality* of the translation vectors described in Section 2.6. A valid tiling symbol might have translation vectors that are not minimal, which creates non-trivial redundancy of symbols. We describe how to deal with this in Section 3.3. The basic idea is to produce a vertex cloud and evaluate the translations from the origin to vertices with the same star as itself. If the seeds find themselves contained in the cloud for smaller translation vectors than the original, these new ones are saved. Fundamental region and seeds are corrected with respect to this lattice.

*Equivalence.* Closely related to uniqueness is the issue of deciding whether two representations define the same tiling. This is a crucial task for identifying and removing duplicates when acquiring or generating large collections of tilings. The procedure described in Section 2.9 can be applied, or one can generate two large vertex clouds, one for each representation, and test whether one contains the seeds of the other tiling completely, both ways, after applying a translation or a rotation. More precisely, for each seed in one representation, translate the translation grid so that the seed becomes the origin. Then, for each basic direction $\omega^k$, rotate the vertex cloud about the origin by multiplying by $\omega^k$, and check whether the rotated vertex cloud contains the seeds of the other representation.

*Conciseness.* A representation should contain just enough information to represent an object. Some redundancy is allowed, if it simplifies some operations. Our representation for tilings is very concise. By definition, minimal representations contain no redundancy. The entries in the matrices of minimal representations are typically small integers.

*Ease of creation.* Our representation for tilings is suitable for both manual and automatic creation from images. We describe with detail in Chapter 3 how to extract a representation for a tiling automatically from an image of it.

To extract a representation for a tiling manually from an image or even a sketch of it, one first performs the normalization process that satisfies the hypothesis of our representation: fix a vertex to be the origin and rotate the drawing around the origin so that some edge becomes horizontal. Then, locate two vertices nearest to the origin that are equivalent to the origin by two linearly independent translations. This could be done by printing two semi-transparent drawings of the tilings and sliding one over the other until they match, or just by drawing a line joining the vertex to its equivalent pair. This is the hard step for a human, especially for large repetitive tilings. We explain an automatic procedure to identify the translation lattice over point clouds in Chapter 3.

Once the translation vectors have been found, the seeds are the vertices inside the parallelogram defined by them. The lattice coordinates are found by following paths to the vertices, as described in Section 2.7. This process automatically gives a minimal representation because the vertices defining the translation vectors are the feasible ones closest to the origin.

To show the ease of creation of tilings within our representation we have created a web tiling prototype creator[11]. The application starts with the origin seed and assists the user for the insertion of new seeds over feasible $\mathbb{Z}[\omega]$ points, after each insertion the edges are created automatically. The user defines two pairs of equivalent vertices and tests the tiling result in real time. The lattice coordinates of each element are given in text form so they can be easily saved. The process is illustrated on Figure 2.15.

[11] chequesoto.info/tiledraw/

*Efficacy in the context of applications.* Our representation is simple to understand, as explained in Section 2.7. It is simple to reconstruct a tiling from a representation, as explained in Section 2.8. Complementing the compact external representation (an integer matrix) with an explicit internal representation of its vertices (the cloud) simplifies many algorithms.

No complex data structures and no nearest-neighbors searches are needed. Moreover, most tasks have robust solutions based on error-free integer arithmetic. There are no geometric tests and no arbitrary tolerances that often plague geometric algorithms.

## 2.11 *Related work*

The general approach given by Kaplan [44] is well suited for representing all the varieties of isohedral tilings, given a parametrization of the tile borders and its own adjacency. However, that approach becomes too complex when we try to generalize it for periodic tilings having more than one tile. In a similar effort, Cosineau [23] approaches the problem from the symmetry group choice operating over a defined tile. This approach requires that we have a defined tile, and over it the geometric mappings are applied.

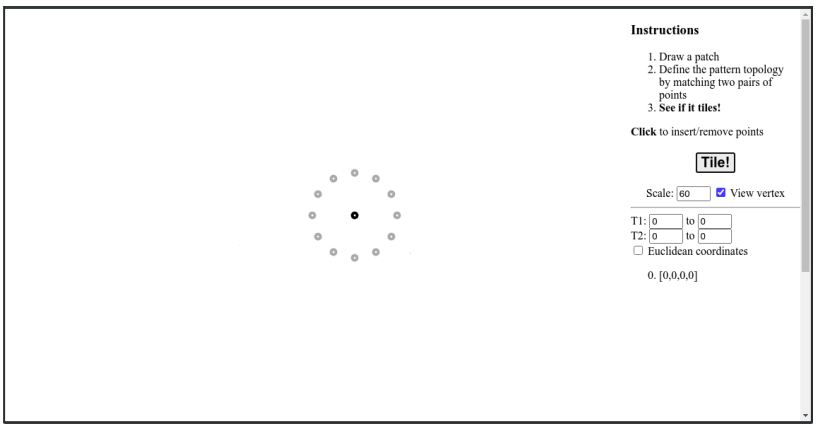
Delgado-Friedrichs [25] describes a general data structure for representing periodic tilings accurately with a graph symbol and a pair of adjacency functions between *chambers*. He uses a triangulation of the original tiles such that each triangle includes exactly one edge of the tiling. Again, the graph symbol and the chamber labeling gives all the symmetries of the tiling. However, the rendering of such symbols in large scale is costly, since the classification of each point relies only in the neighboring chambers. Our translation-only approach gives a fast method for classifying an arbitrary point and for drawing the tiling inside arbitrarily large regions of the plane. Equivalence between tilings in their representation becomes a general graph isomorphism problem, which is very complex to solve in general.

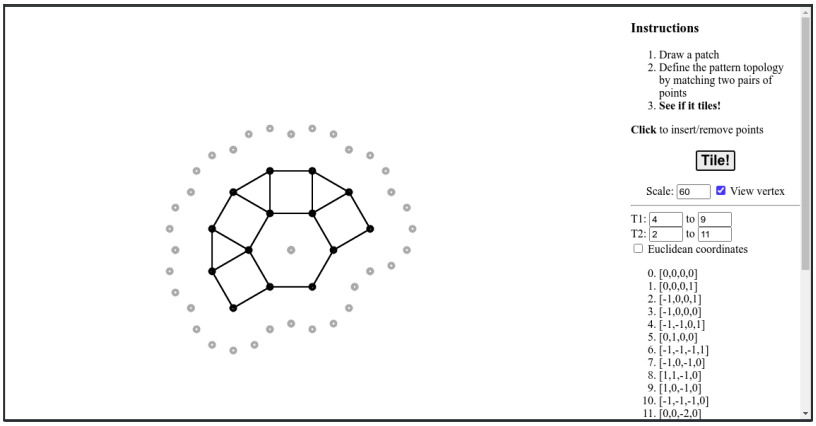
Our approach is closer in spirit to that of Ostromoukhov [63]. He explains how to produce a representation of a plane ornamental pattern, such as an Islamic pattern, by manually analyzing its structure starting from a sketch or an image of the pattern. He also shows how to use the analytical representation to synthesize a drawing. A key point in Ostromoukhov's approach is strand analysis, which represents how the symmetries of the pattern behave in a fundamental region. Our approach with seeds is similar, but much simpler, since it only requires translational symmetry.

The choice to represent the tilings using complex numbers or 4-dimensional integer vectors has been derived independently by other authors in different contexts. Peng et al. [66] make a different choice of 4-tuples to represent the coordinates. Conelly et al. [19] appeal to the same complex basis, however, they do not discuss in detail how the basis would be used to create the tilings. As we have pointed out, the sets in $\mathbb{Z}[\omega]$ that characterize tilings with regular polygons are a very restricted subset of $\Lambda(1, \omega, \omega^2, \omega^3)$.

Our representation was used for independent implementations, like the one by Claudio Esperança, that can be found in the URL `observablehq.com/@esperanc/synthesizing-periodic-tilings-of-the-plane`. Also, Weyrich implemented a triangle and square tiling generation strategy,[12] based on an intensive computer vertex insertion in $\mathbb{Z}[\omega]$, guided by some optimizations that exploit the properties of our representation. The vertex insertion performed by his algorithm resembles the user activity in our GUI tiling creator shown in Figure 2.15.

[12] Unpublished.



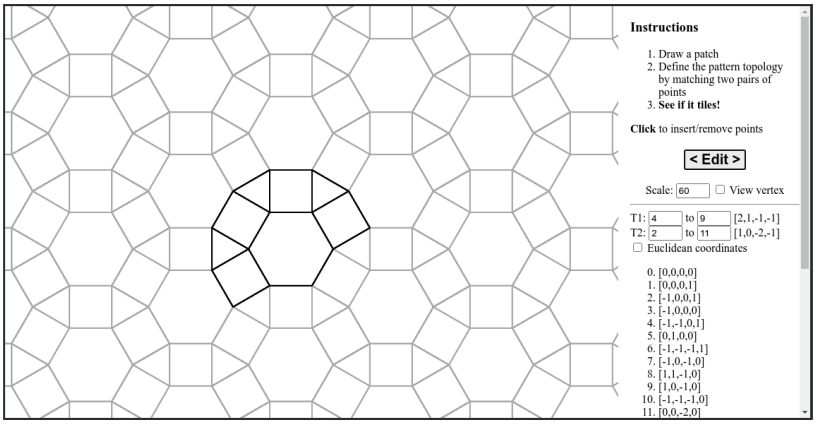




Figure 2.15: Process of manual tiling creation using the web tiling prototype creator. The output in lattice coordinates is printed in real time on the sidebar.

# 3 *Acquisition*

When we started this research, the state of the art in high-order Archimedean tilings were image collections, not computational representations. We will describe the collections we used and how we processed the images to convert them to our integer lattice representation. Once represented as vertex clouds in $\mathbb{Z}[\omega]$, we automatically detect minimum translation grids that determine the fundamental region and select the seeds to build a full symbol, as described in Chapter 2.

This acquisition resulted in a data set that allowed some insights about the $\mathbb{Z}[\omega]$ representation of periodic tilings with regular polygons. It also provided a test set for the symmetry detection algorithm described in Chapter 4.

## 3.1 *Image sets*

*Sa&Sa.* The first image set to which we had access to is the product of many years of computer-aided arts craft. After his first book on the subject in 1982 [74], Ricardo Sá continued working on Archimedean tilings for many years.

The research project that includes this thesis started in 2017 when Asla and Ricardo Sá were preparing the publication of their collection of tilings [75]. After the book was published, they kindly shared 213 images of vertex constellations as colored dots, like the one in Figure 3.1, which is a $189 \times 183$ PNG image. The dots have a diameter of about 8 pixels.

*Galebach.* Brian Galebach investigated high-order $n$-uniform tilings in 2002, apparently using a computer search. The result is a collection of images in his website[1]. Even though he has published no computer program and no technical paper about his findings, not even a description of the methodology he used, the numbers of tilings found by him are cited in The On-Line Encyclopedia of Integer Sequences (OEIS) [81], and remain the state of the art.

[1] probabilitysports.com/tilings.html

The collection by Galebach [28] contains 1351 images with low-resolution line drawings of tilings, like the one in Figure 3.2, which is a $768 \times 576$ PNG image. The lines are 1-pixel wide and have no antialiasing.

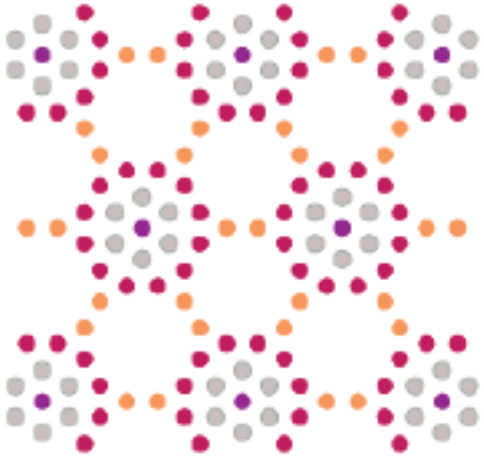


Figure 3.1: Sample input image: tiling GMUW, Sa&Sa collection.

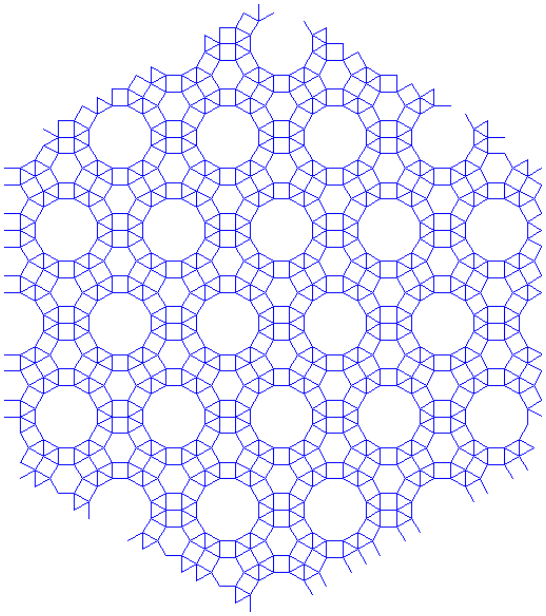


Figure 3.2: Sample input image: 3 uniform tiling 35, Galebach collection.

The challenge that we posed ourselves was to acquire representations of the tilings in both collections so that we could have a computational database that could be shared and used to analyze the variety of tilings.

The full pipeline of the acquisition is divided in two main stages:

1. Transform the image into a system of points in $\mathbb{Z}[\omega]$.
   - Acquire a set of vertices from the image.
   - Correct the vertices, identifying its lattice coordinates.
2. Identify a translation lattice for the tiling, select the seeds, and build a tiling symbol.
   - Find two translations that leave the tiling invariant.
   - Find the seeds inside the fundamental region.
   - Minimize the translation vectors and correct the symbol.

## 3.2 *From pixel space to $\mathbb{Z}[\omega]$*

*Acquiring vertices.* The first step is to find approximate coordinates for the vertices in the image. This is the only step that depends on the nature of the input. We use standard image processing to identify and extract the vertices from the input image.

For the vertex constellations from Sa&Sa collection, we approximated the vertices by finding the centroids of the connected components in the image, using *regionprops* in MATLAB. We had to rotate a few images to ensure that every image had its edges aligned with the basic directions in $\Omega$.

For the line drawings from Galebach collection, we convolved the input image with a $3 \times 3$ kernel of ones. In this new image, we selected the pixels having value of at least four. The corresponding pixels in the original image contain a vertex where at least three edges meet. Recall that lines are 1-pixel wide and have no antialiasing. We then approximated the vertices by finding the centroids of the connected components in the new image, as before.

Finally, we ensured that all implied edges have approximately unit length by scaling the extracted points about their centroid with a scale of $1/d$, where $d$ is the distance from the origin to its nearest neighbor. The origin is the point nearest to the centroid of the point cloud.

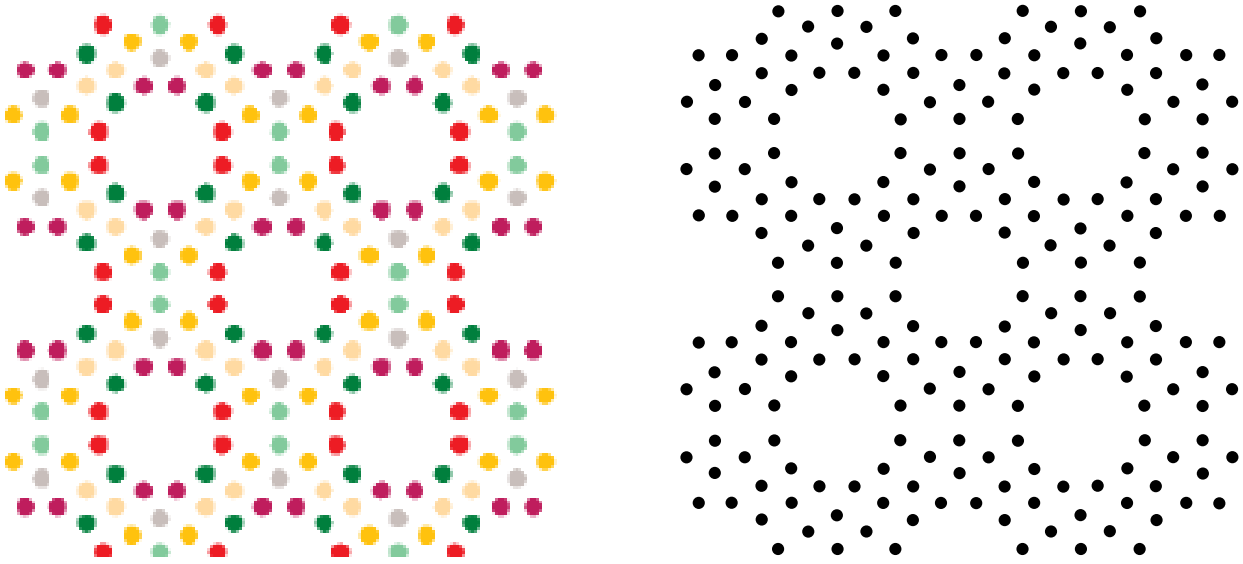


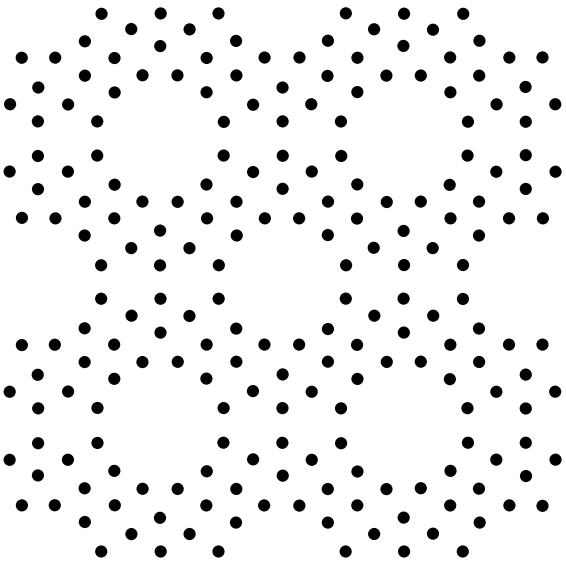
Figure 3.3: Input image and the approximate vertex coordinates: HLMNQTU tiling, Sa&Sa collection.

*Correcting vertices.* The approximate $(x, y)$ coordinates for the vertices found in the first step are corrected in the second step by enforcing the restriction that the edges have unit length and are aligned with the basic directions in $\Omega$. The output of this step are exact coordinates in $\mathbb{Z}[\omega]$ for the vertices in the tiling.

We build a spanning tree $T$ for the vertices using front propagation as follows. The tree $T$ and the front $F$ start with the origin, which is the vertex closest to the centroid of the vertices. While $F$ is not empty, we remove a vertex $v \in F$ and find all vertices $w$ not in $T$ such that $1 - \delta < d(v, w) < 1 + \delta$. The edge $vw$ is added to the tree $T$ and the vertex $w$ is added to the front $F$. Ideally, we seek those $w$ such that $d(v, w) = 1$, but we need to use a tolerance of $\delta$ because the vertex coordinates in pixel space obtained from the first step are an approximation. The value $\delta = 0.35$ worked for all 1564 vertex clouds. This large tolerance attests to how robust this step is.

Each time an edge $vw$ is found, we store the (corrected) lattice coordinates in $w' \in \mathbb{Z}[\omega]$ for $w$, which are given by $v' + \omega^k$, where $\omega^k$ is the direction closest to $w - v$. We avoid any error propagation throughout the insertion of new vertices by using the original approximate Euclidean coordinates for computing the neighboring candidates.

The spanning tree $T$ is just a tool for systematically visiting all vertices and finding their coordinates in $\mathbb{Z}[\omega]$. These coordinates for a vertex $v'$ correspond to the unique path in $T$ from the origin to $v'$. The tree $T$ is not needed in the following steps, so it is discarded.



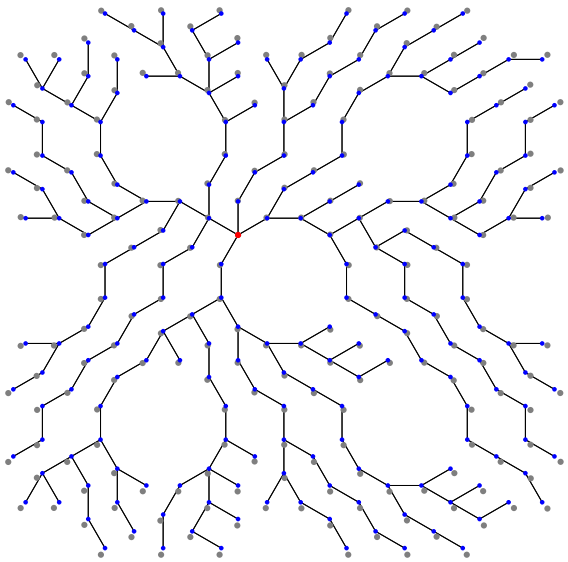


Figure 3.4: Approximate vertex coordinates and corrected ones with spanning tree: HLMNQTU tiling, Sa&Sa collection.

## 3.3 *Minimal translation lattices*

Since we now have a set of vertices in $\mathbb{Z}[\omega]$, we apply the edge reconstruction algorithm described in Algorithm 1. The stars contain fundamental information to discriminate possible translations, since two vertices can only be equivalent if they have the exact same star.

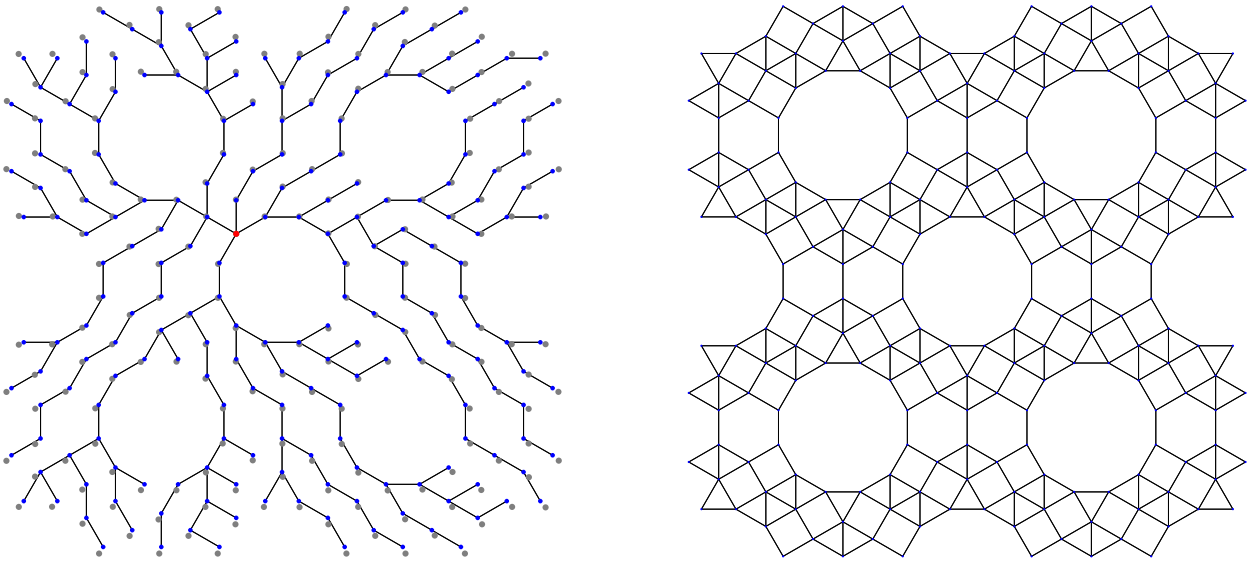


Figure 3.5: Edge reconstruction over the corrected coordinates in $\mathbb{Z}[\omega]$: HLMN-QTU tiling, Sa&Sa collection.

*Detecting translations.* We look for translation vectors $t$ such that the tiling is invariant under a translation by $t$. This could be seen as graph automorphism; however, this is true only for the infinite vertex cloud or under the reduction to the flat torus, which at this point is unknown.

Since the vertex cloud is finite, we must settle for a partial automorphism: some vertices and edges will be sent outside the cloud, and some vertices and edges in the cloud will not be the image of any vertex or edge. Even efficient algorithms for graph automorphisms [46, 58] cannot find translations, only rotations and reflections. Note that the problem of finding translation vectors is not the standard cloud registration problem in computer vision, which deals with two clouds of points [36]. In that context, the problem is solved by finding the best rigid transformation that maps one cloud to the other, in the sense of least squares.

However, we are looking for exact solutions, not approximate ones. Exact solutions are possible because the vertices have exact integer coordinates in $\mathbb{Z}[\omega]$, and so do translation vectors. Our solution is exact in this sense; it does not use the $(x, y)$ coordinates of the vertices, which are necessarily approximate, since they involve $\sqrt{3}$.

To find the translation vectors, we need to analyze the translational symmetries of the vertex cloud. We test all possible translations in $\mathbb{Z}[\omega]$, in a clever way. A translation vector must send the origin to another vertex of the same type, and with the same orientation in the tiling. Thus, a necessary condition for two vertices to have the same

type is that they have the same star, hence the computation in the previous step.

For each vertex $v$ having the same star as the origin, we make the translation $t = v$ and assign a score it:

$$\text{score}(t) = \#((V \pm t) \cap V)$$

Where $V$ is the vertex cloud and $V \pm t = \{w \pm t \mid w \in V\}$ is the result of translating $V$ by $t$ in both directions. A vertex $w$ is counted in this score iff $w + t$ or $w - t$ is some vertex $w_t \in V$ and $w$ and $w_t$ have the same star. This score measures how much of the tiling is preserved under translation by $t$. We use coordinates in $\mathbb{Z}[\omega]$ to test whether $w \pm t \in V$ via the hash table, so there is no need to compare with every other vertex.

We order the pairs $(\text{score}(t), \|tW\|)$ lexicographically in order of decreasing score and increasing length, and select the top two translation vectors that are linearly independent. Including lengths in this selection helps to find a small translation cell.

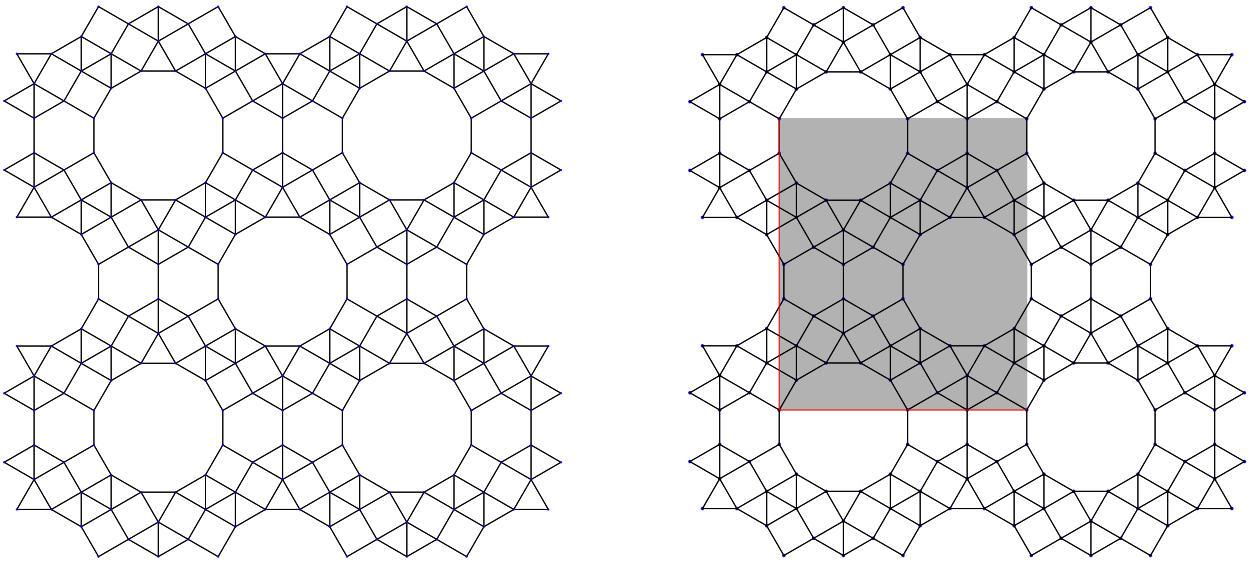


Figure 3.6: Translations detected by score function: HLMNQTU tiling, Sa&Sa collection.

*Finding seeds.* The seeds are the vertices inside the translation cell determined by the translation vectors $t_1$ and $t_2$ found in the previous step. The basic cell is $\{\lambda_1 t_1 + \lambda_2 t_2 \,:\, \lambda_1, \lambda_2 \in [0, 1)\}$. Note that this cell is half-open, that is, open at the sides not containing the origin.

For each vertex $v$ in the tiling, we find its coordinates $(\lambda_1, \lambda_2)$ in the basis $(t_1, t_2)$ by solving a standard $2 \times 2$ linear system. This step uses the $(x, y)$ coordinates of $v, t_1, t_2$. Then $v$ is a seed iff $-\varepsilon \leq \lambda_1, \lambda_2 \leq 1 - \varepsilon$. This tests ensures that we find all seeds, including the ones on the closed sides of the translation cell, but not the ones on its open sides. The value $\varepsilon = 10^{-6}$ worked well in all cases.

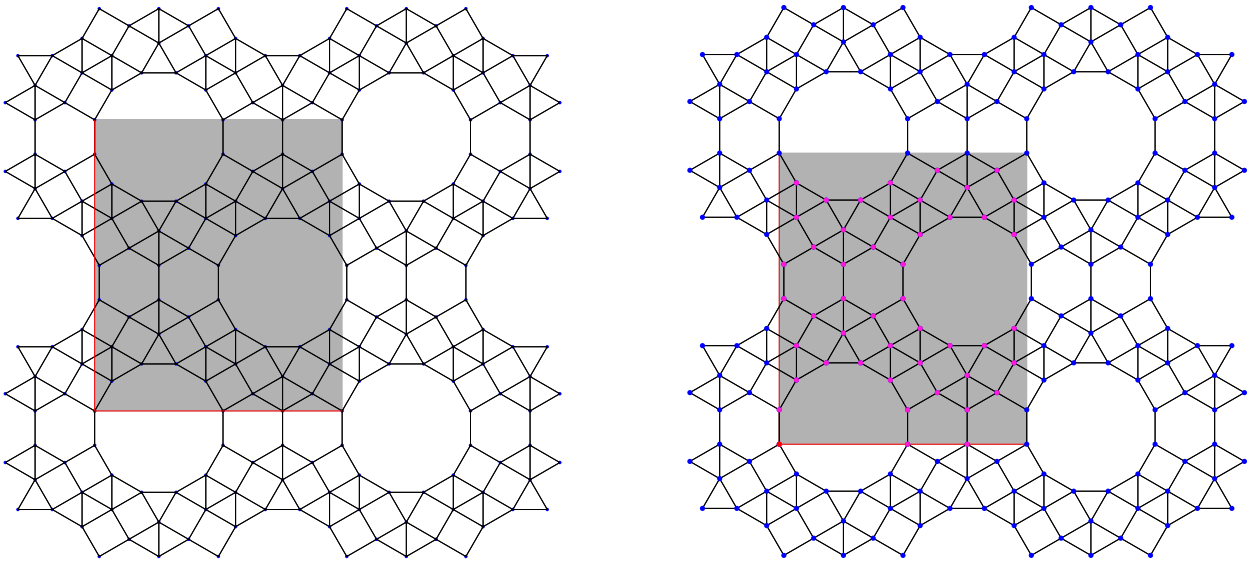


Figure 3.7: Seeds for the initial translation cell: HLMNQTU tiling, Sa&Sa collection.

*Minimizing translations.* The translation cell, and the set of seeds found in the previous steps may not be minimal. This is due to the size and shape of the vertex cloud, which might introduce a geometric bias in translation vector scores.

Nevertheless, if we have succeeded in reproducing almost all vertices in the cloud by translating the seeds inside the translation cell along the translation vectors, then we have acquired the tiling correctly. In particular, we can generate arbitrarily large vertex clouds for the tiling.

We find minimal translation vectors and seed sets by detecting translations and finding seeds, as described above, on a new synthetic vertex cloud generated by replicating the available translation cell in a $5 \times 5$ grid, and insisting that the candidate translations reproduce all vertices in the $3 \times 3$ middle sub-grid.

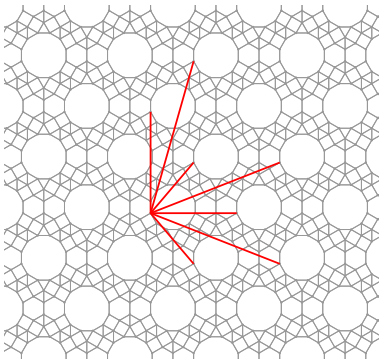


Figure 3.8: Finding minimal translations.

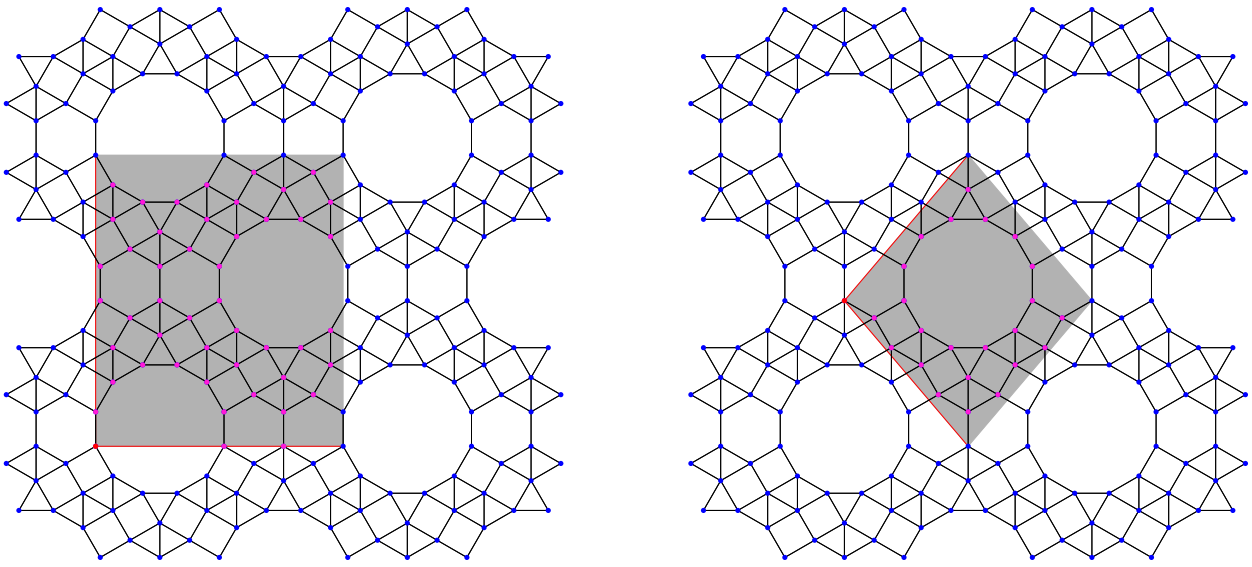


Figure 3.9: Finding minimal translations, fundamental region, and minimal seed set: HLMNQTU tiling, Sa&Sa collection.

## 3.4 *Results*

The pipeline described above was applied to the Sa&Sa [75] and Galebach [28] tilings collections. For each tiling, we verified how well the tiling reconstructed from our acquired representation matched the original input data. More precisely, we computed how many corrected vertices appeared in the vertex cloud obtained by translating the seeds along the translation vectors to cover the input cloud.

The results are shown in Table 3.1, which should be read as follows: In the Sa&Sa collection of 213 tilings, 195 tilings were perfectly reconstructed, 9 tilings had at least 99% success but less than 100%, 7 tilings had at least 98% success but less than 99%, 1 tiling had at least 96% success but less than 97%, 1 tiling had at least 49% success but less than 50%. In the Galebach collection of 1351 tilings, the results are even better, probably because their clouds are larger: 1346 tilings were perfectly reconstructed and the remaining 5 tilings had at least 99% success.

| Set | Input reconstruction | | | | | | Acquired |
|---|---|---|---|---|---|---|---|
| Sa&Sa | % success | 100 | 99 | 98 | 96 | 49 | 98.6 |
| | # tilings | 195 | 94 | 7 | 1 | 1 | 210/213 |
| Galebach | % success | 100 | 99 | | | | 100.0 |
| | # tilings | 1346 | 5 | | | | 1351/1351 |

Table 3.1: Verification results for the two collections of tilings.

In summary, over 98% of the tilings in Sa&Sa collection were correctly acquired: all but three tilings were perfectly reconstructed. For the tilings by Galebach, 100% of the tilings were perfectly reconstructed with no exceptions. In both collections, the exceptional vertices that were not reconstructed are typically at the fringe of the original cloud and have been acquired erroneously because they have very few neighbors and so the wrong star.

The few cases where our algorithm found an incorrect translation cell were the tiles labeled NPRTUW, NQTUVW, and PTU6, from Sa&Sa (Figure 3.10). This failure is due to insufficient input data. Despite appearances, there are some features in the input images for which there is incomplete information to deduce the translation pattern. In Figure 3.10, we have colored some elements to show deceptive apparent patterns in small samples. One can perceive that the algorithm fails only with proper undecidable cases, where translations are not clear even after careful human inspection. It seems clear that any pattern with a sufficiently large point cloud sample will be captured properly by the algorithm, as it is the case with Galebach's collection.

This pipeline is quite robust and can be easily adapted to handle other kinds of images, by using suitable image processing to extract vertices. Previously [73], we had the user select the vertices directly on the image, which could even be photographs with mild perspective distortions (Figure 3.11). The pipeline can also be used to handle vector graphics, such as SVG and PostScript, as long as it is possible to extract vertex coordinates, even in different coordinate spaces.

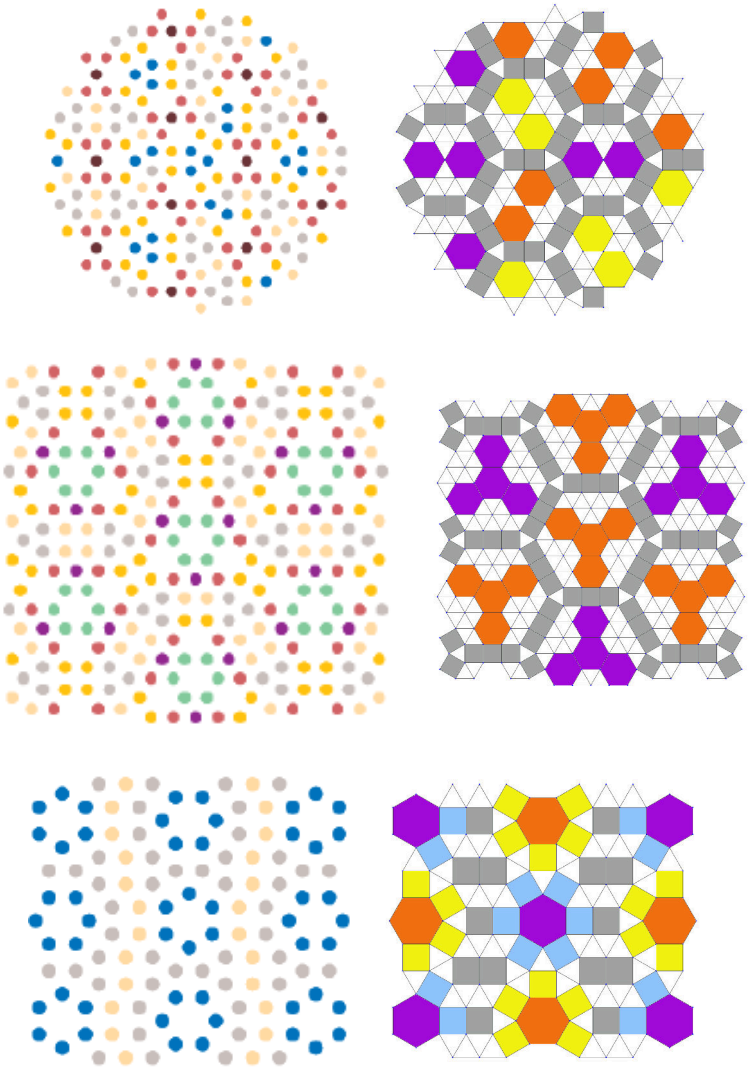


Figure 3.10: The input images do not contain sufficient information for reconstructing the tilings. The colored features, which are potential equivalences, have ambiguous (incomplete) translation information on the sample.

The data acquired from the two collections of tilings is available at the project web page[2] for further research on the topic. The data are JSON files containing the name of the tiling and the coordinates in $\mathbb{Z}[\omega]$ of its translation vectors and seeds. These coordinates are very small integers (less than 15 in absolute value). The file for the whole Sa&Sa collection [75] has about 64K bytes; the file for the Galebach collection [28] has about 343K bytes. We include some examples of the tilings in the collection in Section 3.7.

[2] chequesoto.info/thesis.html

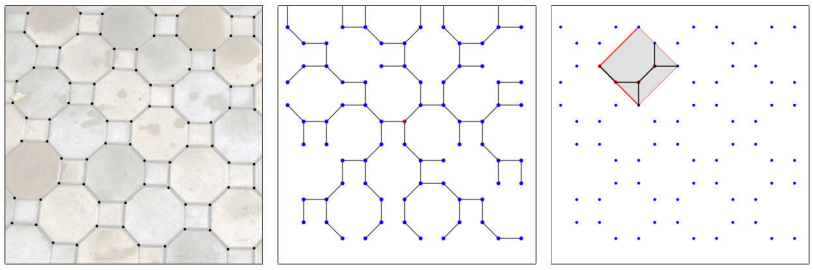


Figure 3.11: User-assisted acquisition of tilings from photographs with mild perspective distortion [73]. Our acquisition algorithm is robust enough to work on a setup like this. For large distortions, one would apply an homography first.

## 3.5 *Coincidences*

We used the equality decision algorithm described in Section 2.9 to detect coincidences in the two collections. As explained, to avoid testing all pairs of models, we only tested pairs with the same number of seeds and fundamental regions with the same area. These are necessary but not sufficient conditions for two representations to define the same tiling. We tested 364 potential coincidences, having the same number of seeds, area, and Hermite normal form. We found 143 coincidences, some of which are not obvious because they involve rotations. Figure 3.12 shows an example. After a rotation, the tiling GLMT2 in the collection by Sa&Sa [75] is the same as the 4-uniform tiling 37 and the 4-Archimedean, 4-uniform tiling 16 in Galebach [28].

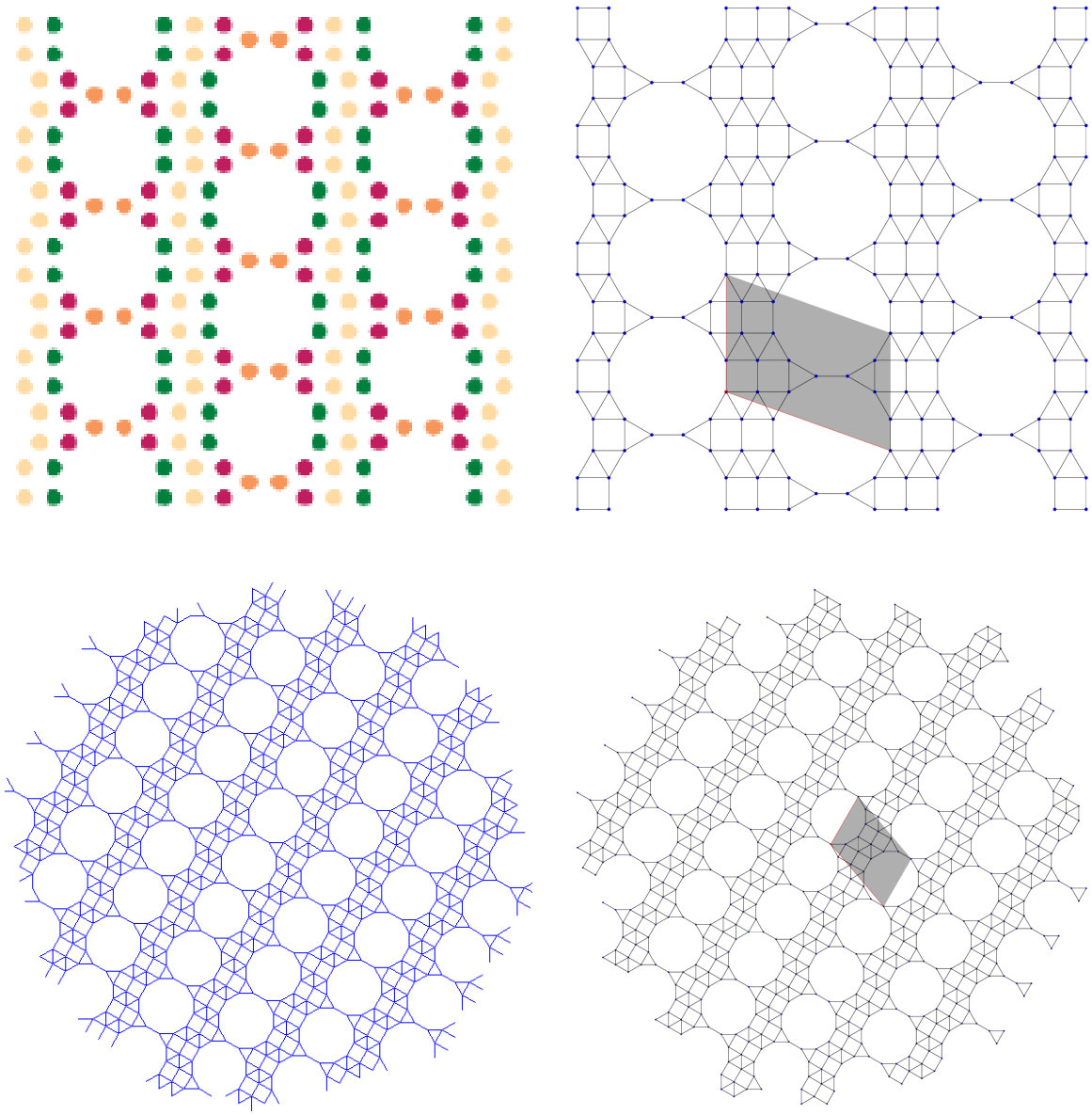


Figure 3.12: An example of a coincidence between the two sets: the tiling in the top row is the same as the tiling in the bottom row, after a rotation. Input images on the left, reconstructed tilings on the right.

## 3.6 Related work

The collection of tilings by Galebach [28] is the state of the art in the classification of $k$-uniform tilings [81]. Only low-resolution images with line drawings of tilings are available at that site, but unfortunately not vertex coordinates or code. Efforts in producing catalogs on tilings of regular polygons are recurrent in the literature.

In 1989, Chavey [16] collected results and drawings of regular tilings, arguing that classifications and theoretical results on the topic are scattered across several papers. Chavey provides drawings of 165 tilings of regular polygons, arranged by vertex orbits and labeled according to vertex types.

In 2019, Wikipedia [94] included a catalog of 564 tilings that cite Chavey [16] and Galebach [28] as sources. The SVG images in Wikipedia contain coordinates that could be extracted and used in our pipeline. However, the tilings in Wikipedia are a subset of those by Galebach [28] and have fewer vertices.

Liu et al. [57] carefully reviewed the field of computational symmetry, including the long history of symmetry detection algorithms, dating back to 1932. They mention that detection of reflection symmetry used to dominate the field of symmetry detection in computer vision. Their review classifies methods by the types of symmetries detected, as well as the type of method used for detection. We stress that we are interested only in translations for now, even though periodic tilings of regular polygons have several types of symmetries.[3] Thus, our problem falls into the class of lattice detection algorithms. Liu et al. [55] classified a few papers as lattice detection. All of them perform their search for translation direction in frequency space. Our approach is based on voting schemes, which are more frequently used to detect rotations and reflections. We search for global translation symmetry although we have only local information in hand.

[3] Symmetry detection will be addressed using our representation in Chapter 4.

Liu et al. [56] investigate the problem of automatically inferring the lattice structure of near-regular textures in real-world images. The first step toward automatic texel discovery is the detection of repeated interest points present in the image. The key trade-off is to extract enough interest points to reveal some repeated structure reliably without overwhelming the subsequent lattice finder with false positives. Liu et al. [56] report good performance for their algorithm by testing it on the data used in the Symmetry Detection from Real-World Images Competition 2013. The problem that we solve here is different for two main reasons: (1) we seek translations in $\mathbb{Z}[\omega]$ and search in a few directions; (2) our data is composed of several interlocked grids over the same lattice. This would most probably be a challenging dataset for previous algorithms.

## 3.7 *Samples of the acquired collections*

$$\left[\begin{array}{cc|cccccccccccccccccccccccc} 2 & -2 & 0 & -1 & 0 & -1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & -1 & -1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 3 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 3 & 3 & 3 & 3 \\ 2 & 4 & 0 & 2 & 0 & 2 & 1 & 1 & 2 & 4 & 1 & 2 & 2 & 4 & 4 & 3 & 3 & 2 & 2 & 4 & 4 & 5 & 2 & 4 & 5 & 5 \\ 0 & 3 & 0 & 2 & 0 & 1 & 0 & 1 & 1 & 3 & 0 & 0 & 1 & 2 & 3 & 1 & 2 & 0 & 1 & 2 & 3 & 3 & 0 & 2 & 2 & 3 \end{array}\right]^T$$

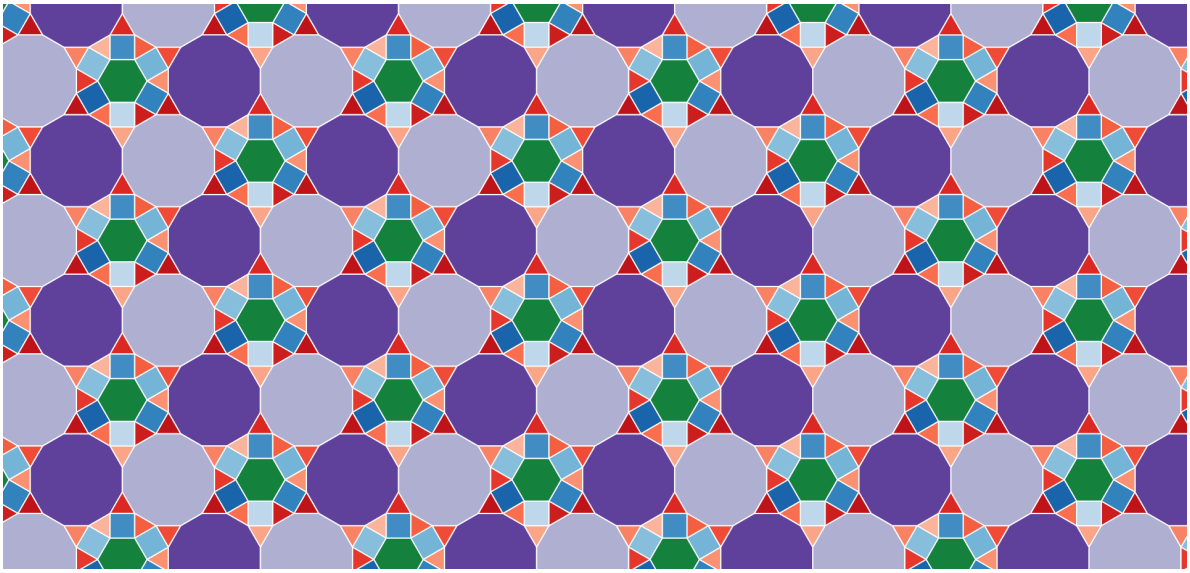


Figure 3.13: *GMP* tiling from Sa&Sa collection [75].

$$\left[\begin{array}{cc|cccccccccccccccccccccc} -2 & 2 & -1 & -1 & 0 & 0 & 1 & 0 & 0 & -1 & -1 & 1 & -1 & 0 & 0 & 1 & 1 & 0 & 0 & -1 & -1 & 0 & 0 & 1 & \cdots \\ 0 & 6 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 2 & 2 & 2 & 2 & 2 & \cdots \\ 4 & 2 & 2 & 2 & 0 & 0 & 0 & 2 & 2 & 4 & 4 & 0 & 4 & 2 & 2 & 1 & 1 & 3 & 3 & 4 & 4 & 2 & 2 & 1 & \cdots \\ 6 & 0 & 3 & 4 & 0 & 1 & 1 & 3 & 4 & 6 & 6 & 1 & 5 & 2 & 4 & 1 & 2 & 4 & 5 & 5 & 6 & 2 & 3 & 0 & \cdots \end{array}\right]^T$$

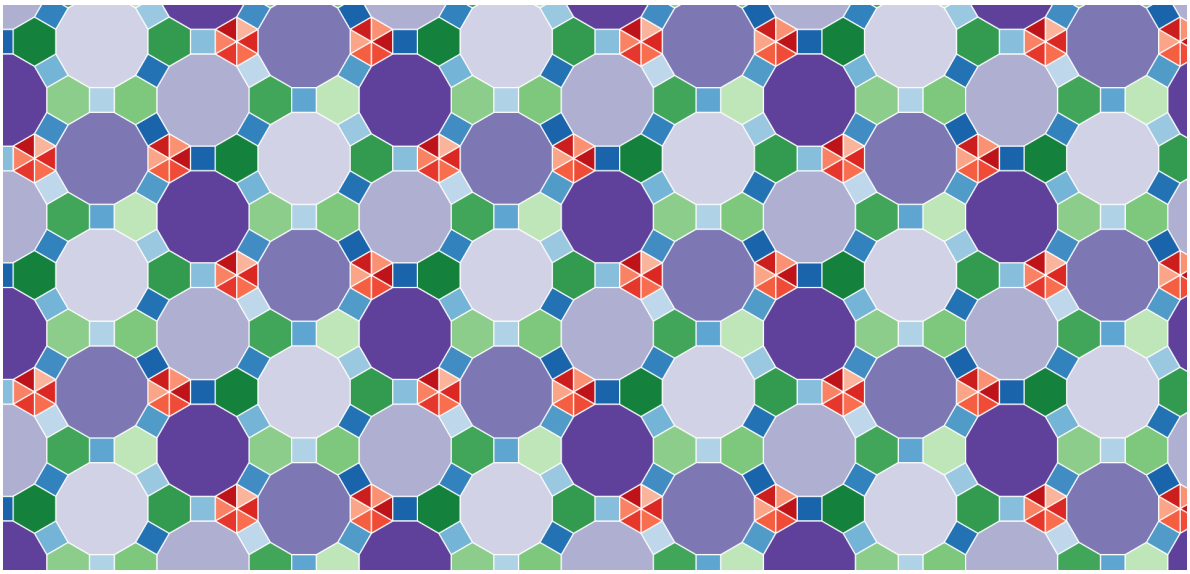


Figure 3.14: Tiling number 32 from Galebach collection of 5-uniform tilings [28].

$$\left[\begin{array}{cc|cccccccccccccccccc} 0 & 1 & 0 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\ 2 & 2 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 2 & 2 & 2 & 2 & 2 & 2 & 3 & 3 \\ 1 & -1 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & -4 & 0 & -1 & 0 & 1 & -2 & 1 & -2 & -1 & 0 & 1 & -3 & 1 & -3 & -2 & 0 & 1 & -2 & -1 \end{array}\right]^T$$

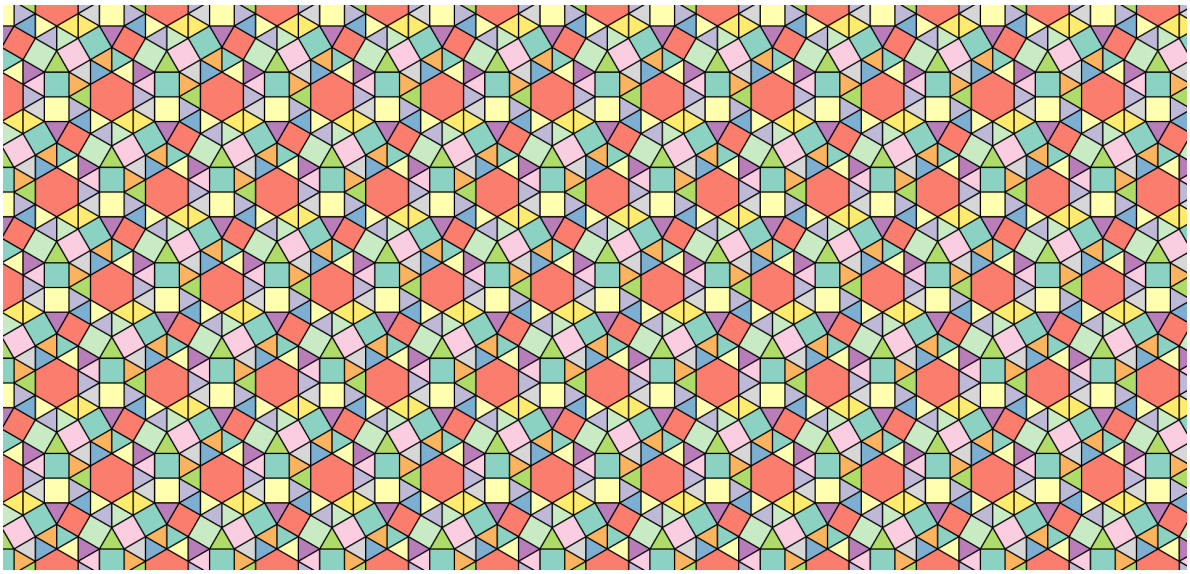


Figure 3.15: *TUV* tiling from Sa&Sa collection [75].

$$\left[\begin{array}{cc|cccccccccccccccccc} -1 & 4 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 2 & 1 & 1 & 1 & 2 & 3 & 1 & 2 & 3 & 4 & 2 & 3 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 4 & -1 & 0 & 1 & 0 & 0 & 2 & 1 & 0 & 2 & -1 & 1 & 3 & 2 & 1 & -1 & 3 & 2 & 0 & -1 & 3 & 2 \\ 1 & -1 & 0 & 1 & 1 & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}\right]^T$$

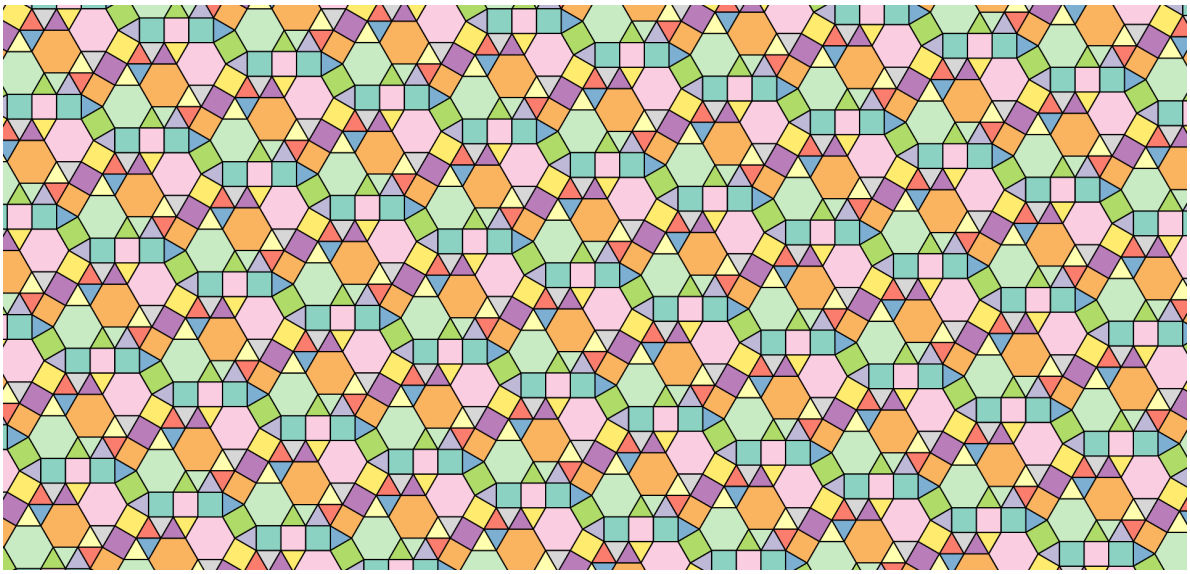


Figure 3.16: Tiling number 7 from Galebach collection of 6-uniform 6-Archimedean tilings [28].

$$\left[\begin{array}{cc|cccccccccccccccccc} 0 & 2 & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 2 & 1 & 0 & 2 & 2 & 0 & 2 & 1 & 2 & 2 & \cdots \\ 3 & 3 & 0 & 1 & 1 & 2 & 2 & 2 & 2 & 2 & 2 & 3 & 3 & 3 & 3 & 3 & 3 & 3 & 4 & 4 & \cdots \\ 2 & -2 & 0 & 0 & -1 & 0 & 0 & -1 & -1 & 1 & -2 & -1 & 1 & -2 & -2 & 2 & -1 & 1 & -2 & -1 & \cdots \\ 1 & -4 & 0 & -1 & -1 & -1 & 0 & -2 & -1 & 0 & -2 & -2 & 0 & -3 & -2 & 0 & -2 & 0 & -3 & -3 & \cdots \end{array}\right]^T$$

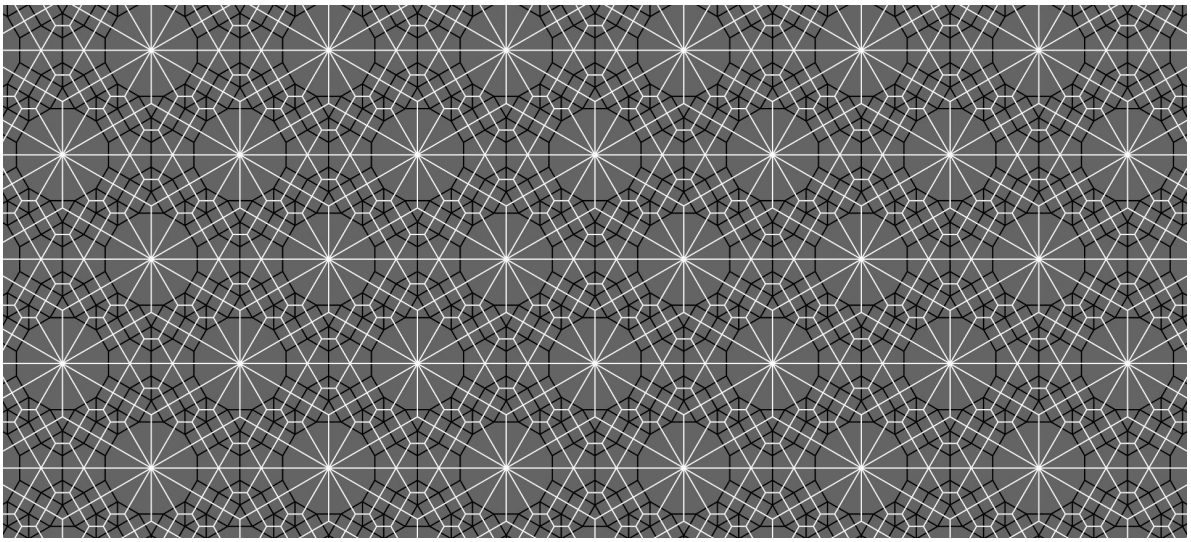


Figure 3.17: *HLMNQTU* tiling, and its dual, from Sa&Sa collection [75].

$$\left[\begin{array}{cc|cccccccccccccccccc} 3 & 6 & 0 & 1 & 2 & 1 & 2 & 1 & 2 & 3 & 3 & 2 & 3 & 3 & 3 & 3 & 3 & 4 & 4 & 3 & \cdots \\ 3 & 3 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 2 & 1 & 2 & 1 & 1 & 2 & 2 & 2 & \cdots \\ 3 & -3 & 0 & 0 & -1 & 0 & -1 & 1 & 0 & -2 & -1 & 1 & -2 & 0 & -1 & 1 & 1 & -2 & -2 & 0 & \cdots \\ 0 & -3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & -1 & 0 & 1 & -2 & -1 & -1 & \cdots \end{array}\right]^T$$

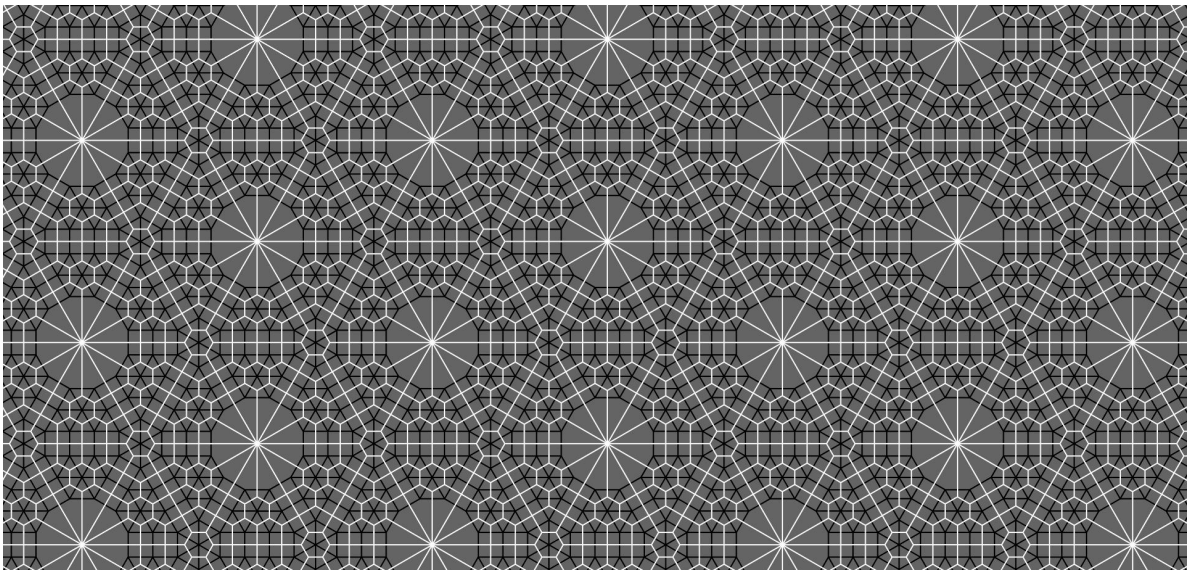


Figure 3.18: Tiling number 610, and its dual, from Galebach collection of 6-uniform tilings [28].

$$\left[\begin{array}{rr|cccccccccccccccccc} -2 & 4 & 0 & -1 & 0 & 0 & -2 & -2 & -2 & 0 & -1 & 0 & 0 & 0 & -2 & -2 & 0 & -1 & 0 & 0 & \cdots \\ 0 & 4 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 2 & 2 & 2 & 2 & 2 & 2 & \cdots \\ 4 & -2 & 0 & 2 & 1 & 1 & 3 & 3 & 3 & 0 & 2 & 1 & 1 & 1 & 3 & 3 & 0 & 2 & 1 & 1 & \cdots \\ 4 & -4 & 0 & 1 & 0 & 1 & 1 & 2 & 3 & -1 & 1 & -1 & 0 & 1 & 1 & 2 & -2 & 1 & -2 & -1 & \cdots \end{array}\right]^T$$

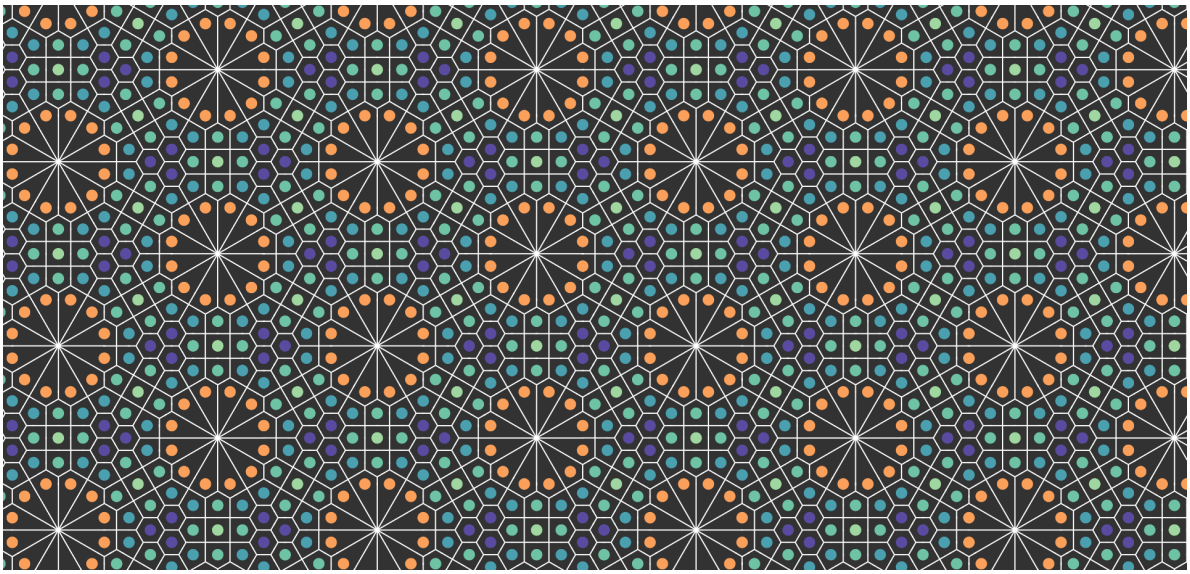


Figure 3.19: *LSTUW* tiling vertices colored by type, and its dual, from Sa&Sa collection [75].

$$\left[\begin{array}{rr|ccccccccccccccccc} -2 & 2 & -1 & 0 & 0 & -1 & -1 & 0 & -1 & -1 & -1 & 1 & -1 & 0 & -1 & 0 & 0 & -1 & 1 & -1 & \cdots \\ 0 & 3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & \cdots \\ 4 & 2 & 2 & 0 & 0 & 2 & 2 & 1 & 3 & 2 & 2 & 0 & 4 & 1 & 3 & 2 & 2 & 4 & 0 & 4 & \cdots \\ 3 & 0 & 2 & 0 & 1 & 1 & 3 & 1 & 3 & 1 & 2 & 1 & 3 & 1 & 2 & 1 & 2 & 3 & 1 & 2 & \cdots \end{array}\right]^T$$

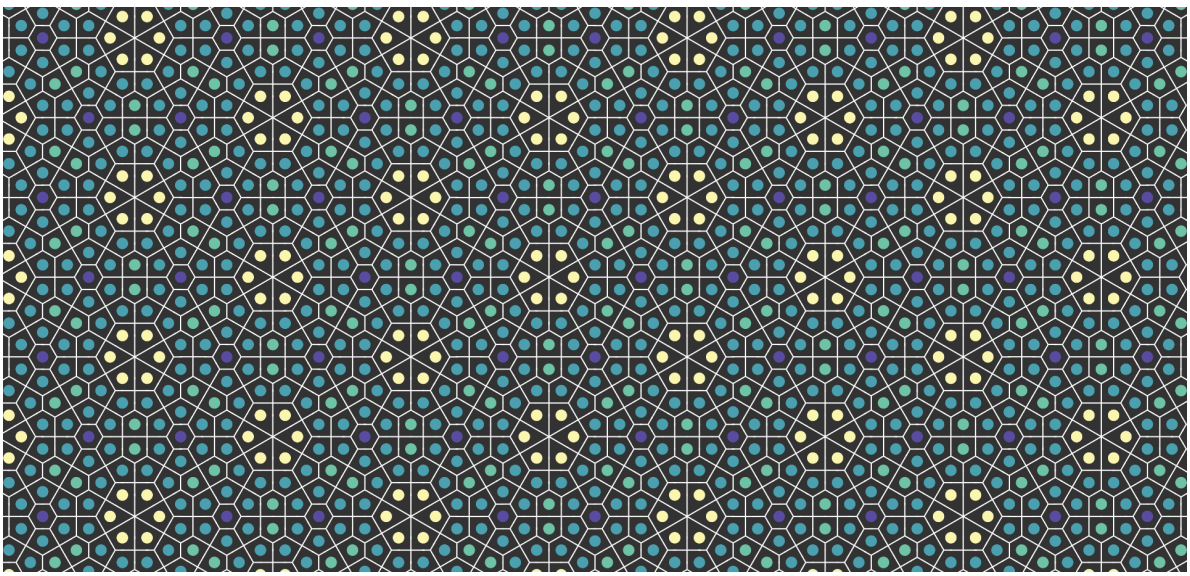


Figure 3.20: Vertices from tiling number 215 colored by type, and its dual, from Galebach collection of 5-uniform tilings [28].

# 4 Symmetry Classification

We proved that our representation is a proper symbol for periodic tilings of the plane with regular polygons, we showed how to reconstruct the topology from a symbol in Chapter 2, and how to acquire a symbol from an image in Chapter 3.

Our interest now is to classify a tiling as $n$-uniform, $k$-Archimedean. The Archimedean class is easily deduced from the stars of the seeds, which represent all vertices in the tiling. The uniformity class is harder because the fundamental region for the translation lattice can contain orbits of other symmetries. Hence, we must reduce the seeds by every possible symmetry equivalence. This requires testing the tiling for invariance under all plane symmetries, that is, finding the plane symmetry (wallpaper) group to which the tiling belongs. Given its symmetry group, we identify its generating region, which contains exactly one vertex for each transitive equivalence class of vertices. The number of such classes gives the uniformity class.

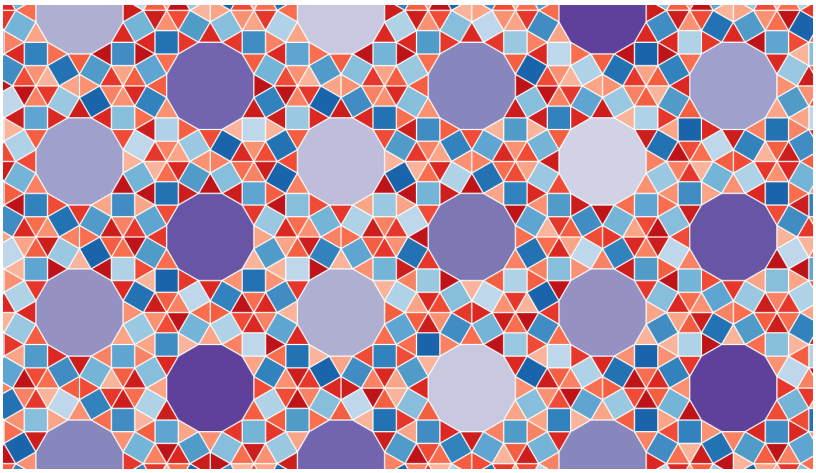


Figure 4.1: Example of a 5-uniform, 3-Archimedean tiling. Tiling number 288 from Galebach collection of 5-uniform tilings [28].

Much work has been done from the perspective of computer vision in the search for symmetries in images [2, 3, 59]. Liu et al.'s survey of computational symmetry [57] gives a historical view on this perspective. However, our methods for symmetry detection work directly over the integer lattice coordinates with arbitrary precision, using the operations defined in Section 2.4. On the other hand, they are of course limited to tilings by regular polygons.

## 4.1 *Symmetry groups in practice*

Plane symmetry groups were presented in Chapter 1, along with their features and notation. One of the classic problems in symmetry is: given a periodic pattern on the plane, determine its symmetry group. Washburn and Crowe [91] proposed a systematic procedure to recognize the symmetry group of any pattern based on its symmetries: 2-, 3-, 4-, or 6-fold rotations, reflections and glide reflections. Their flowchart, shown in Figure 4.2, is widely used in the literature [12].

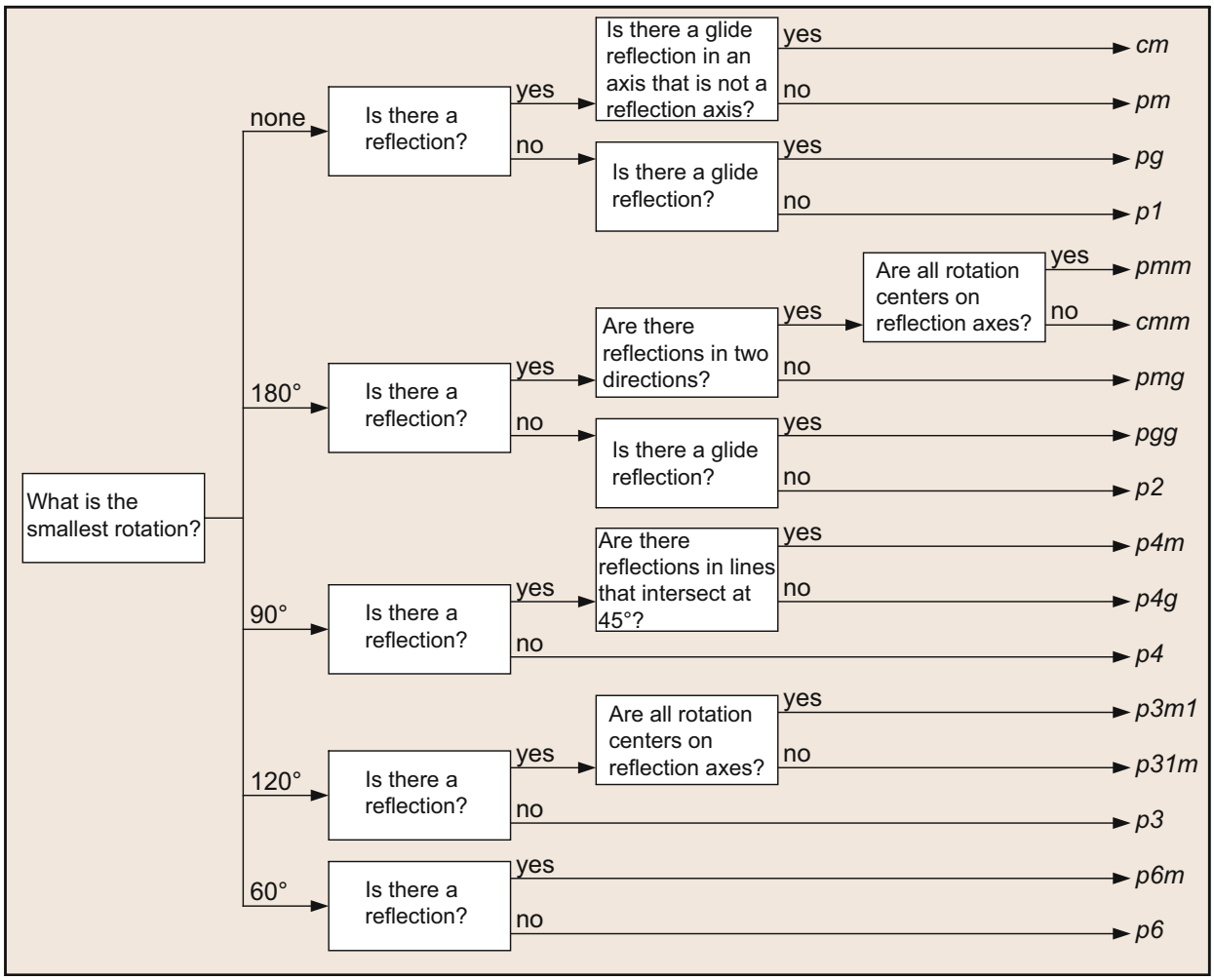



Figure 4.2: Washburn and Crowe flowchart for the classification of symmetry groups of plane periodic patterns by their basic symmetries [12].

Once the plane symmetry group has been identified, the *generating region* of the symmetry group can be delimited. This is the minimal region of the plane that contains one representative point for every transitive equivalence class under the symmetry group. In other words, the generating region contains all the information required to generate the whole pattern by the repeated application of its symmetries. However, this kind of reconstruction is quite more complex than the one by the translation cell.

The generating region is smaller than the fundamental region of the translation lattice. In most cases the generating region is not be origin-independent, since rotation centers and mirror lines constitute singularities of the plane under the symmetry. While translation vectors can always be deduced from the symmetry group's generating region, the opposite is not true.

Generating regions for each symmetry group have been thoroughly described by Schattschneider [78] and are the common reference when dealing with symmetries in computer graphics [57]. Figure 4.3 shows lattice cells with annotated symmetries for each symmetry group.

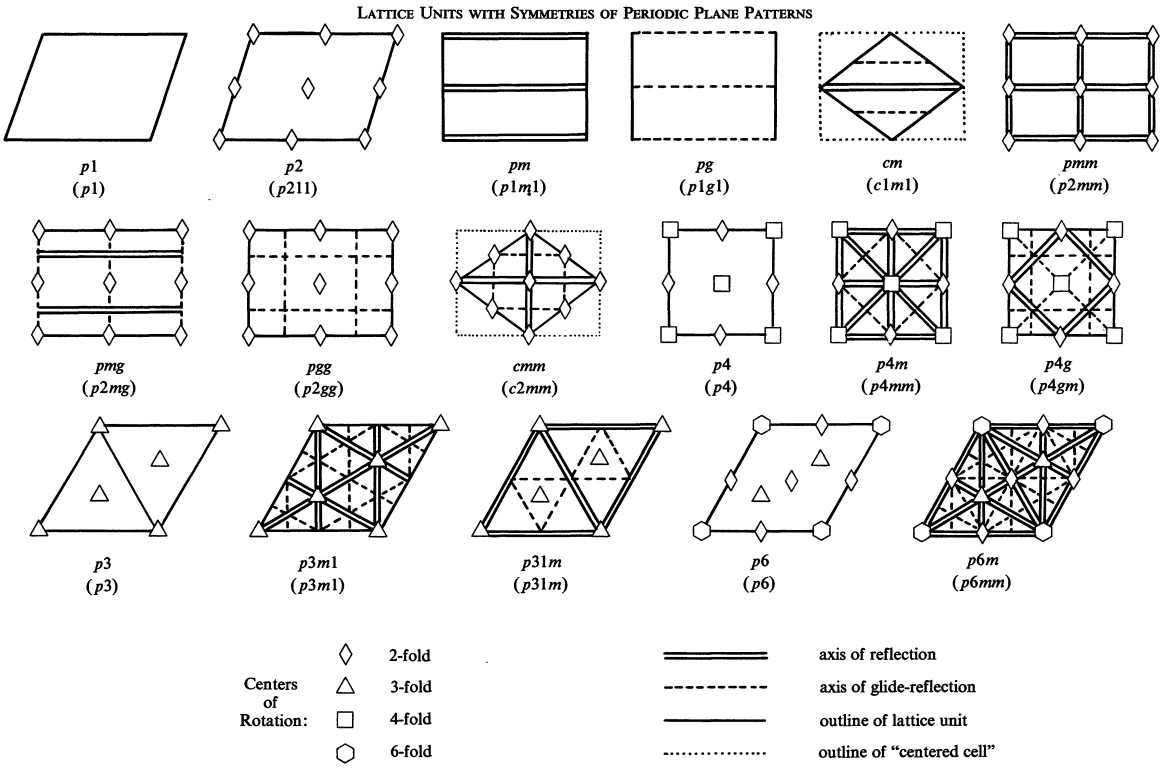



Figure 4.3: Lattice units with annotated symmetries by symmetry group [78].

## 4.2 *Symmetry search*

Isometric transformations can be computed directly in $\mathbb{Z}[\omega]$, as we showed in Section 2.4. This allows us to test for symmetries using integer operations in lattice coordinates. A tiling is symmetric iff it is invariant under an isometry, by the definition of symmetry given in Section 1.1. Thus, to find the symmetries of a tiling, we apply isometric operations to the tiling and test for equivalence between the original tiling and its image under the isometry.

It is impossible to test every possible position of rotation centers and mirrors in the plane, since they are infinite. However, regular polygons and Archimedean vertices have a finite set of candidates of singularities for each symmetry: points and lines where the symmetry holds locally, singularities of the symmetry in a local sense.

The analysis can be done on different levels, but the more specific the triage, the higher the algorithmic cost. For example, from the edge point of view, every midpoint between two vertices is a candidate of a 2-fold rotation center, but this is true only if both of the faces adjacent at that edge are the same polygon type. Determining this in an automatic fashion requires more processing effort.

We will describe the local criteria that qualify a point or a line to be a candidate for a symmetry singularity: rotation centers and mirror lines. We shall develop a different strategy for glide reflections because they do not express themselves in a local manner, since they involve a translation. The local criteria serve to build sets of candidates for every symmetry, and, then, each candidate is used to test the tiling invariance under the corresponding symmetry.

To exploit the efficiency of hashing for determining the existence of a point, we create a vertex cloud $V$ around the fundamental domain that is guaranteed to contain the image of the seeds under each possible symmetry. A ball centered at the origin with radius equal to 3 times the maximum diagonal of the fundamental region guarantees this, since it contains the Minkowski sum of the fundamental region with its possible rotations.

A set of candidates $C_\sigma$ is created for each symmetry $\sigma$ accordingly: a point set for rotations, a line set for reflections and pairs point-vector for glide reflections. After that, we systematically apply each transformation to the set $S$ of seeds, and check the equivalence to the original tiling through the correspondence of the image with $V$. This is, $\sigma$ is a symmetry of the tiling iff $\sigma(S) \subseteq V$. This is equivalent to reducing $\sigma(S)$ to the original plane-torus topology and compare it to itself, but simpler and very efficient.

*Rotations.* Centers of rotation can only occur at: vertices, face centroids, or edge midpoints (2-fold only).

- Vertices
  - R: 2-fold.
  - K: 3-fold.
  - S: 2 and 4-fold.
  - W: 3 and 6-fold.
- Faces
  - Triangle: 3-fold.
  - Square: 2 and 4-fold.
  - Hexagon: 2, 3 and 6-fold.
  - Dodecagon: 2, 3, 4 and 6-fold.
- Edges (midpoint) might have a 2-fold symmetry only in $m \cdot m$ edges, such as the edges present in vertex types: G, K, N, Q, S, T, U, V, W.

*Reflections.* Mirror lines can only occur across edges, across angle bisectors, or across edge bisectors.

- Vertices belong to candidate mirror lines that either bisect the internal angles of faces or are aligned with an edge, only when the star is symmetric by this mirror line:
  - G (1): bisecting the 3 face, aligned with the opposite edge.
  - K (3): at each bisector (edge).
  - M (1): bisecting the 3 and 12 faces.
  - P (1): bisecting the 4 and 6 faces.
  - Q (1): aligned with $3 \cdot 3$ and $6 \cdot 6$ edges.
  - R (2): bisecting either 3 or 6 faces.
  - S (4): aligned with the edges or bisecting 4 faces.
  - T (1): aligned with the $4 \cdot 4$ edge, bisecting the opposite 3 face.
  - U (1): aligned with the $3 \cdot 3$ edge, bisecting the opposite 3 face.
  - V (1): bisecting the 6 face, aligned with a $3 \cdot 3$ edge.
  - W (6): aligned with edges or bisecting faces.
- Edges might have mirror line candidates across their bisectors. Mirror lines across edges have all been considered in the first item.

*Glide reflections.* Candidates for glide reflections are somewhat more complex than the other symmetries. They are even harder to detect visually. Since they involve a translation, glide symmetries do not express locally, but a couple of facts are useful to detect them.

Let $\gamma : \mathbb{Z}[\omega] \longrightarrow \mathbb{Z}[\omega]$ be a glide reflection that leaves the tiling invariant. Since mirrors are their own inverse, we know that $\gamma^2$ must be a valid translation of the tiling, since the mirror part cancels itself out. Then, the translation part $t_g$ of the glide $g$ must be of the form:

$$t_g = \frac{1}{2}\left(k_1 t_1 + k_2 t_2\right), \quad k_1, k_2 \in \{-1, 0, 1\},$$

which is very useful, since the mirror is the line defined by vector $t_g$. Since the image of the transformed points must be in $\mathbb{Z}[\omega]$, points through which glide reflections lines pass must be either vertices or edge midpoints. This stated, we can now build a list of candidates for glide reflections.

Given a tiling, a set of candidates $C_\sigma$ is built for each symmetry $\sigma$ based on the criteria explained above. For each candidate, the tiling is transformed by $\sigma$, the moment a transformed seed is found to be outside $V$, the candidate can be dismissed (Algorithm 5). If the symmetry $\sigma$ holds for that candidate, it is stored in the set of symmetries of the tiling $D_\sigma$ (Algorithm 6). The result is a list of features (points or lines) on the tiling for each type of symmetry, which guide the classification of the tiling into its corresponding symmetry group, as it is illustrated in Figure 4.2.

**Algorithm 5**

**procedure** *issymetry*$(c, \sigma)$
  **for** $j = 1$ **to** $\#S$ **do**
    **if** $\sigma_c(s_j) \notin V$ **then**
      **return** false
    **end**
  **end**
  **return** true
**end**

**Algorithm 6**

**procedure** *symfeatures*$(C_\sigma)$
  $D_\sigma \leftarrow [\,]$
  **for** $j = 1$ **to** $\#C_\sigma$ **do**
    **if** *issymm*$(c_j, \sigma)$ **then**
      $D_\sigma \leftarrow c_j$
    **end**
  **end**
  **return** $D_\sigma$
**end**

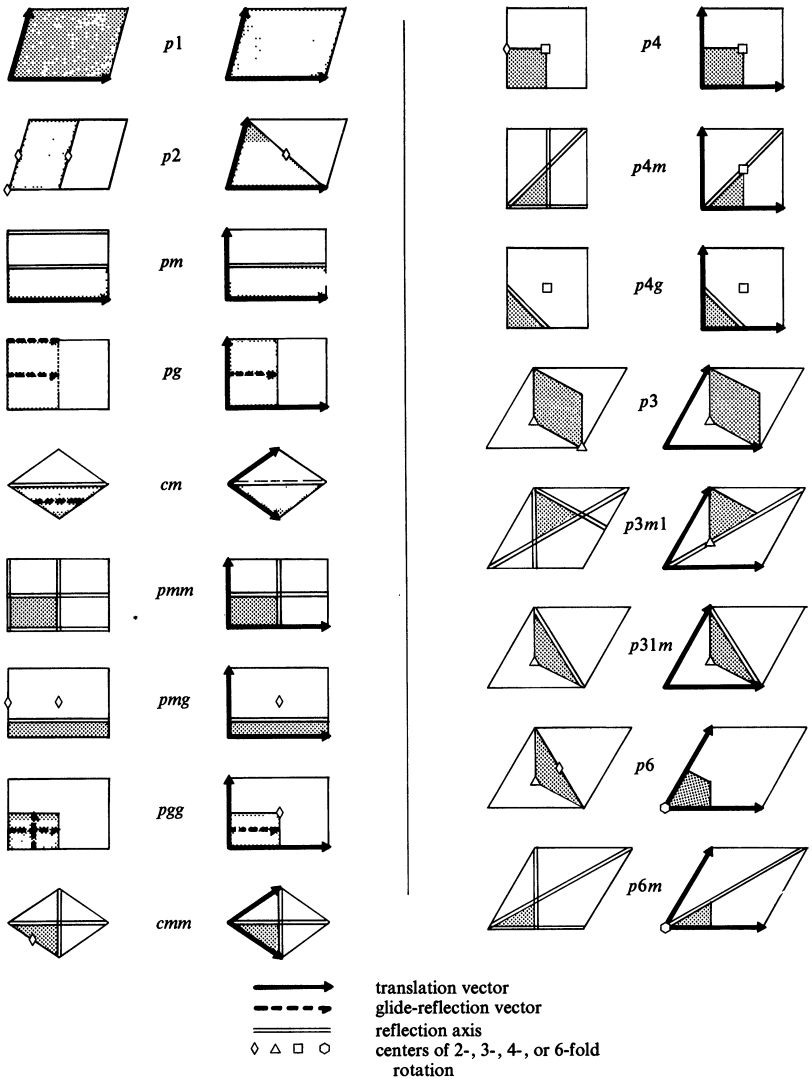



Figure 4.4: Generating regions by symmetry group [78].

## 4.3 *n-uniform classification*

For each symmetry group, the symmetry features or singularities guide the construction of a generating region. This process was illustrated by Schattschneider (Figure 4.4). Each symmetry reduces the translation fundamental region by a natural factor: $r$-fold rotation symmetry divides the area of the generating region by a factor of $r$, reflections and glide reflections reduce the area by a half. The factors from the distinct symmetries present in the group multiply. For example, the generating region of a tiling in $p6m$ symmetry group has $1/12$ of the area of the translation fundamental region.

The most delicate part of delimiting the generating regions is the half closure, which ensures the correct classification of the points inside it without repetition. Finding the uniformity class of the tiling is a point classification problem, some of them are very likely to stand on the borders of the generating region, differentiating open and closed segments of the polygonal region is important. Mirror lines are closed, while sides that delimit the rotation folds are alternatively open and closed. 2-fold rotations generate aligned open and closed segments, for example.

Since many symmetry features have fractional values in $\mathbb{Z}[\omega]$ (centroids of faces and mid-points of edges), we scale all coordinates by a factor of 6 to keep our representation over the integers. In this scaled integer representation, we save some symmetry features (while other remain implicit) together with the seeds inside the generating region according to the symmetry group. Each symmetry group has its own recipe for developing the tiling again. The reduced symbol generated by the symmetry group, even though it is smaller in data, depends on the specific symmetry group recipe for reconstruction, which makes a rendering program based on it more complex than one that uses the method described in Chapter 2. In practice, we build a translation region using the symmetries and then proceed as explained in Chapter 2.

### *Chemical notation*

We propose a **chemical notation** for tilings, where a list of the types of vertices is indexed by its multiplicity in the generating region. The multiplicity remains implicit when it is 1, and the symmetry group is included as a prefix. This way, a chemical style symbol of an $n$-uniform, $k$-Archimedean tiling would have the form:

$$\mathcal{G}_\sigma \,:\, \Lambda^1_{m_1}\Lambda^2_{m_2}\cdots\Lambda^k_{m_k}, \quad \sum_{i=1}^{k} m_i = n,$$

where $\mathcal{G}_\sigma \in \{p1, p2, pm, pg, \ldots, p6m\}$ is the symmetry group, and $\Lambda^i \in \{G, H, J, K, \ldots, W\}$ are the Archimedean vertex types present in the tiling, in alphabetic order, with multiplicity $m_i \in \mathbb{N}$, $i = 1, \ldots, k$. This way, symmetry group and both $n$ and $k$ classes become evident in the symbol.

We show the generating regions for each of the symmetry groups with its open and closed segments, with annotated symmetries in Figure 4.6 to Figure 4.21. We omit the algorithmic details, but the annotated symmetries provide a clear guide for delimiting such regions.

## 4.4 *Results*

After applying the symmetry classification algorithm to our experimental collections, we were able to verify Galebach's collection classifications as $n$-uniform, $k$-Archimedean given by the tiling numbers showed at the end of Chapter 1 up to the 7-th row. Also, the algorithm classified the Sa&Sa collection tilings by their $n$-uniform, $k$-Archimedean class.

Some statistics for the symmetry group classification of the tilings acquired from Sa&Sa and Galebach collections are shown in Table 4.1.

| **Symmetry group** | | **Collection** | | Example |
|---|---|---|---|---|
| | | Galebach | Sa&Sa | figure |
| $p1$ | ○ | 0 | 0 | |
| $p2$ | 2222 | 59 | 6 | Figure 4.6 |
| $pm$ | ** | 1 | 0 | Figure 4.7 |
| $pg$ | ○○ | 1 | 0 | Figure 4.8 |
| $cm$ | *○ | 3 | 0 | Figure 4.9 |
| $pmm$ | *2222 | 289 | 43 | Figure 4.10 |
| $pmg$ | 22* | 138 | 4 | Figure 4.11 |
| $pgg$ | 22○ | 28 | 2 | Figure 4.12 |
| $cmm$ | 2*22 | 482 | 46 | Figure 4.13 |
| $p4$ | 442 | 4 | 2 | Figure 4.14 |
| $p4m$ | *442 | 5 | 3 | Figure 4.15 |
| $p4g$ | 4*2 | 15 | 10 | Figure 4.16 |
| $p3$ | 333 | 7 | 1 | Figure 4.17 |
| $p3m1$ | *333 | 13 | 10 | Figure 4.18 |
| $p31m$ | 3*3 | 13 | 1 | Figure 4.19 |
| $p6$ | 632 | 40 | 6 | Figure 4.20 |
| $p6m$ | *632 | 253 | 76 | Figure 4.21 |
| **Total** | | **1351** | **210** | |

Table 4.1: Symmetry group classification results for the two acquired image collections.

Exemplars of each symmetry group have been selected and displayed over the next pages, with the symmetry features annotated according to the code on Figure 4.5. Each caption gives the chemical symbol, from which can be deduced its $n$-uniform $k$-Archimedean classifications, as well as a reference to the collection it has been taken from, with its original name. The fundamental region is outlined in red. The symmetry group generating region is filled in light blueish gray, outlined in dark gray, with closed boundaries in continuous lines and open ones in dashed lines. The symmetry group generating region is filled with light blueish-gray and the vertices inside it have been marked.

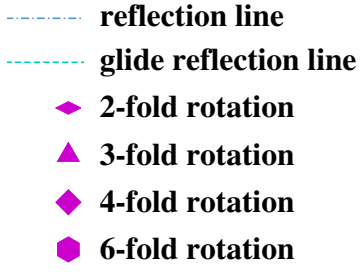



Figure 4.5: Symmetry annotation code.

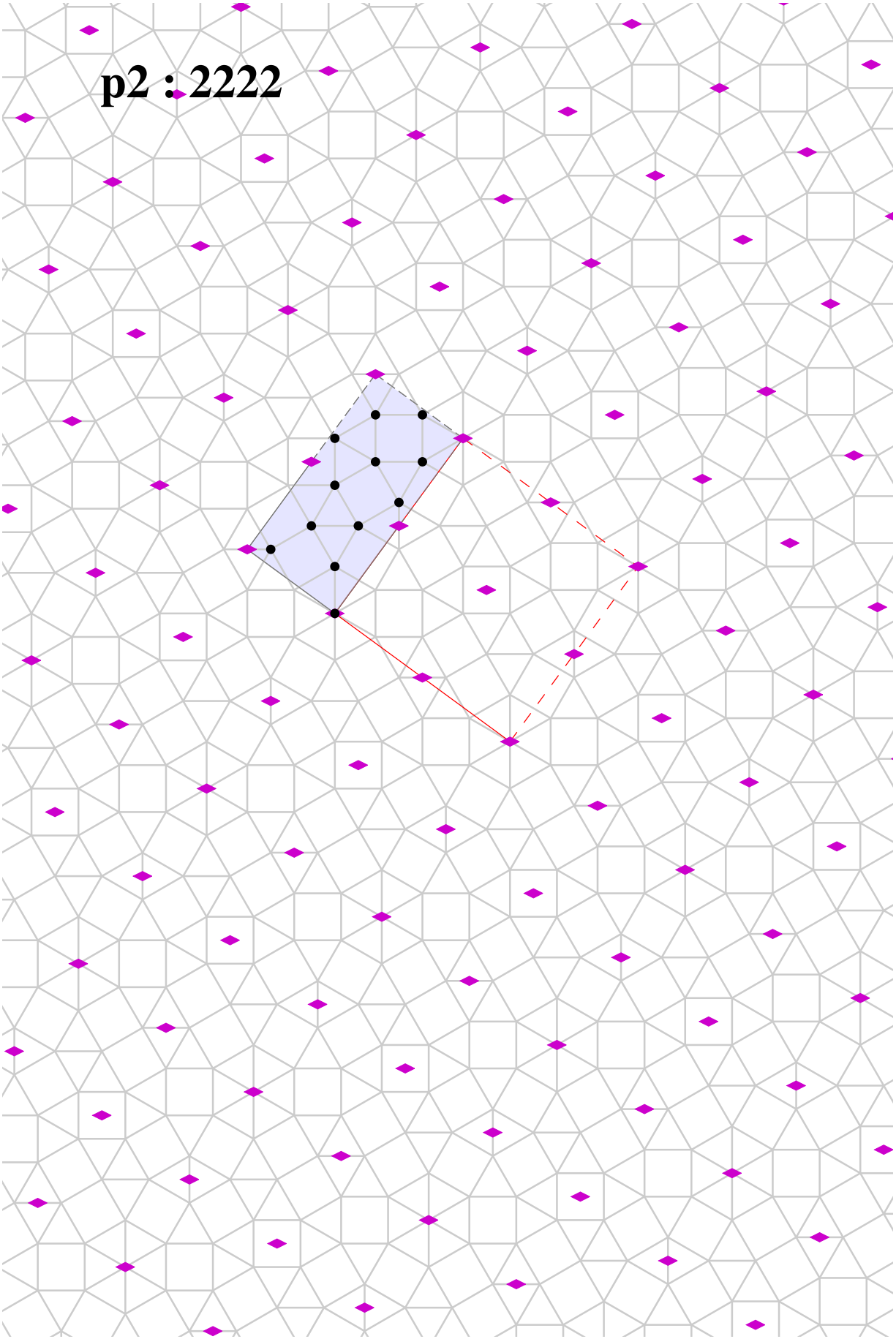



Figure 4.6: $p2(2222) : T_2U_9W$ tiling. 11-th $TUW$ tiling from Sa&Sa collection.

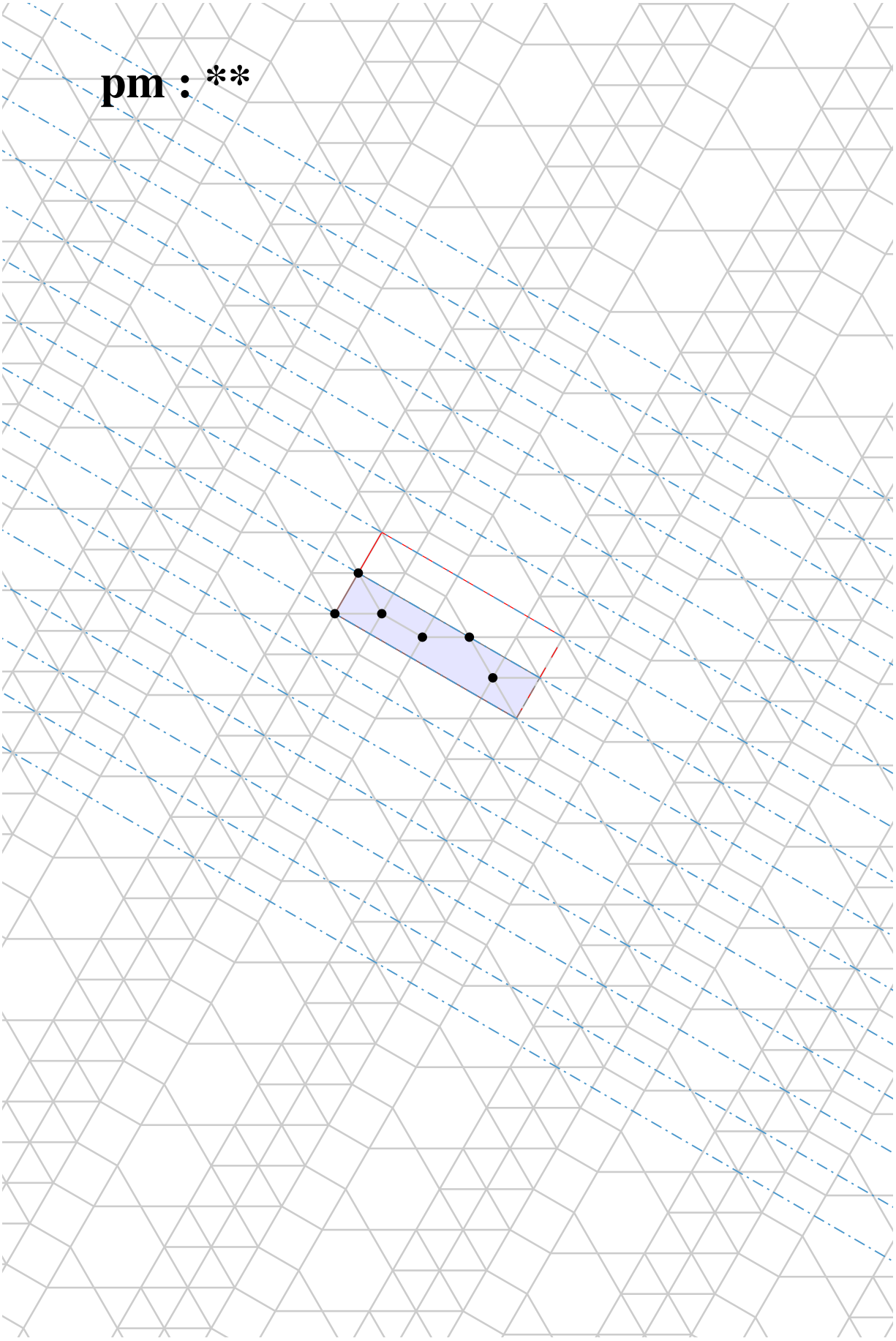



Figure 4.7: $pm(**)$ : $NRTVW_2$ tiling. Number 453 from 6-uniform Galebach collection.

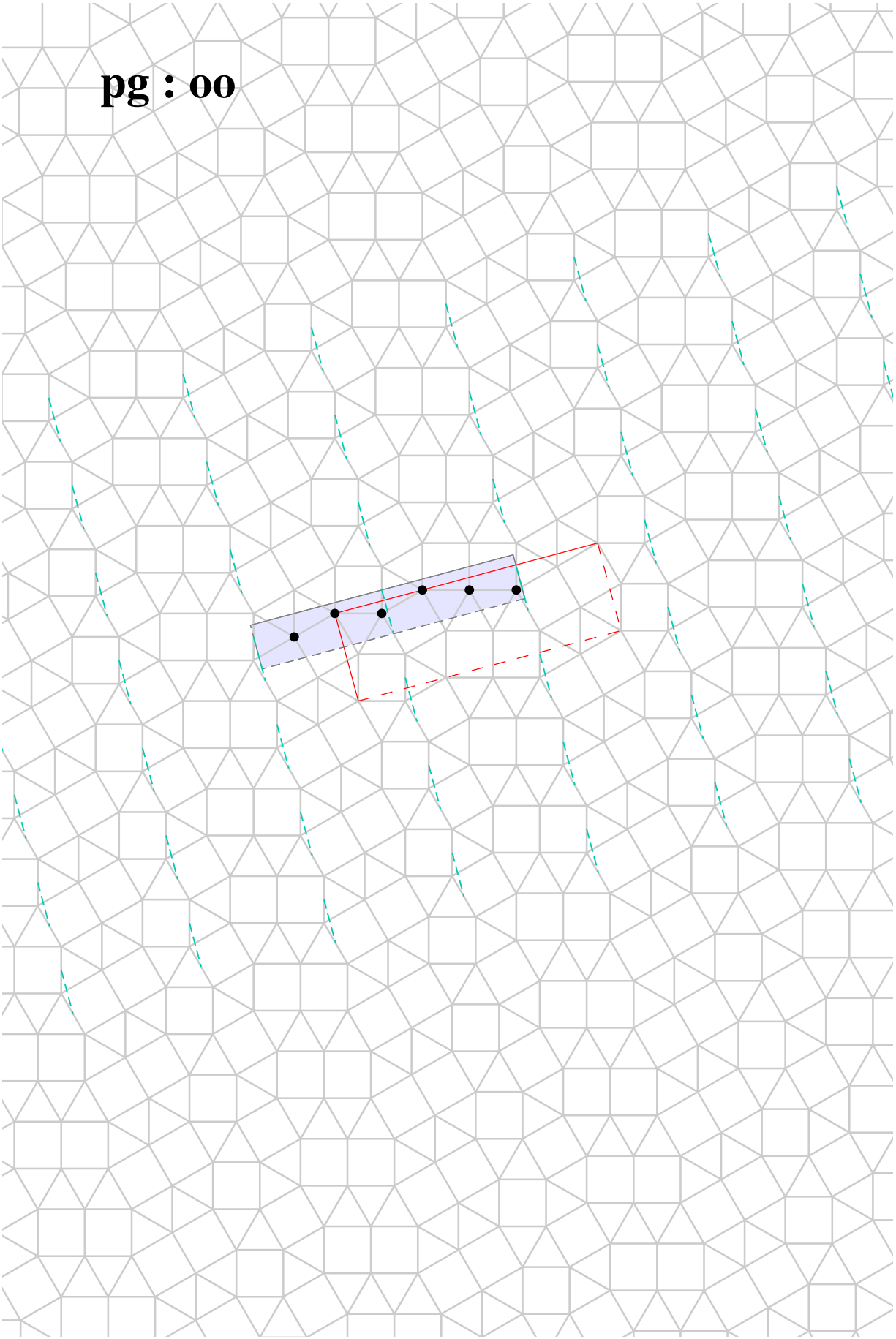



Figure 4.8: $pg(\circ\circ) : T_2U_4$ tiling. Number 661 from 6-uniform Galebach collection.

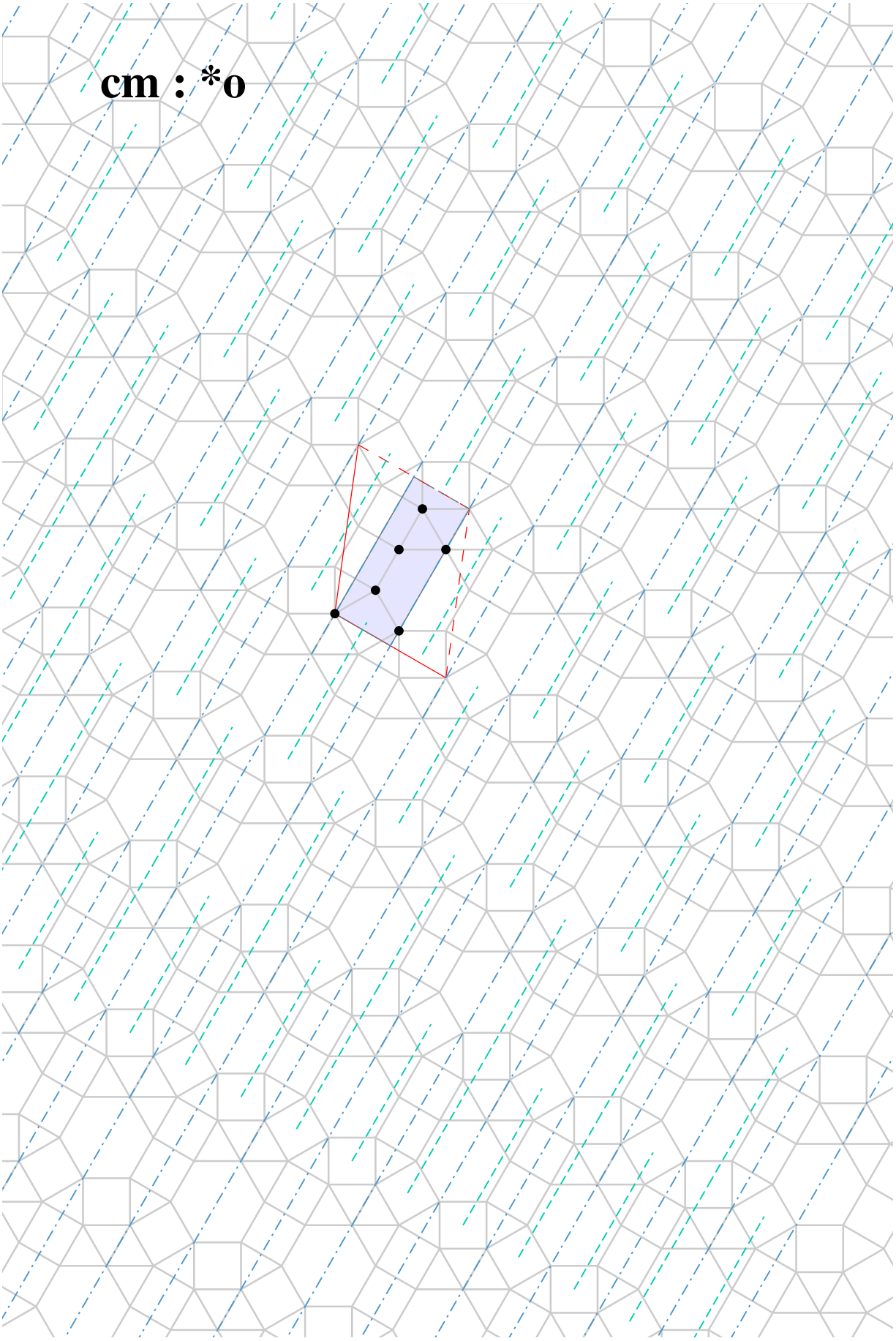



Figure 4.9: $cm(*\circ)$ : $NP_2U_2V$ tiling. Number 478 from 6-uniform Galebach collection.

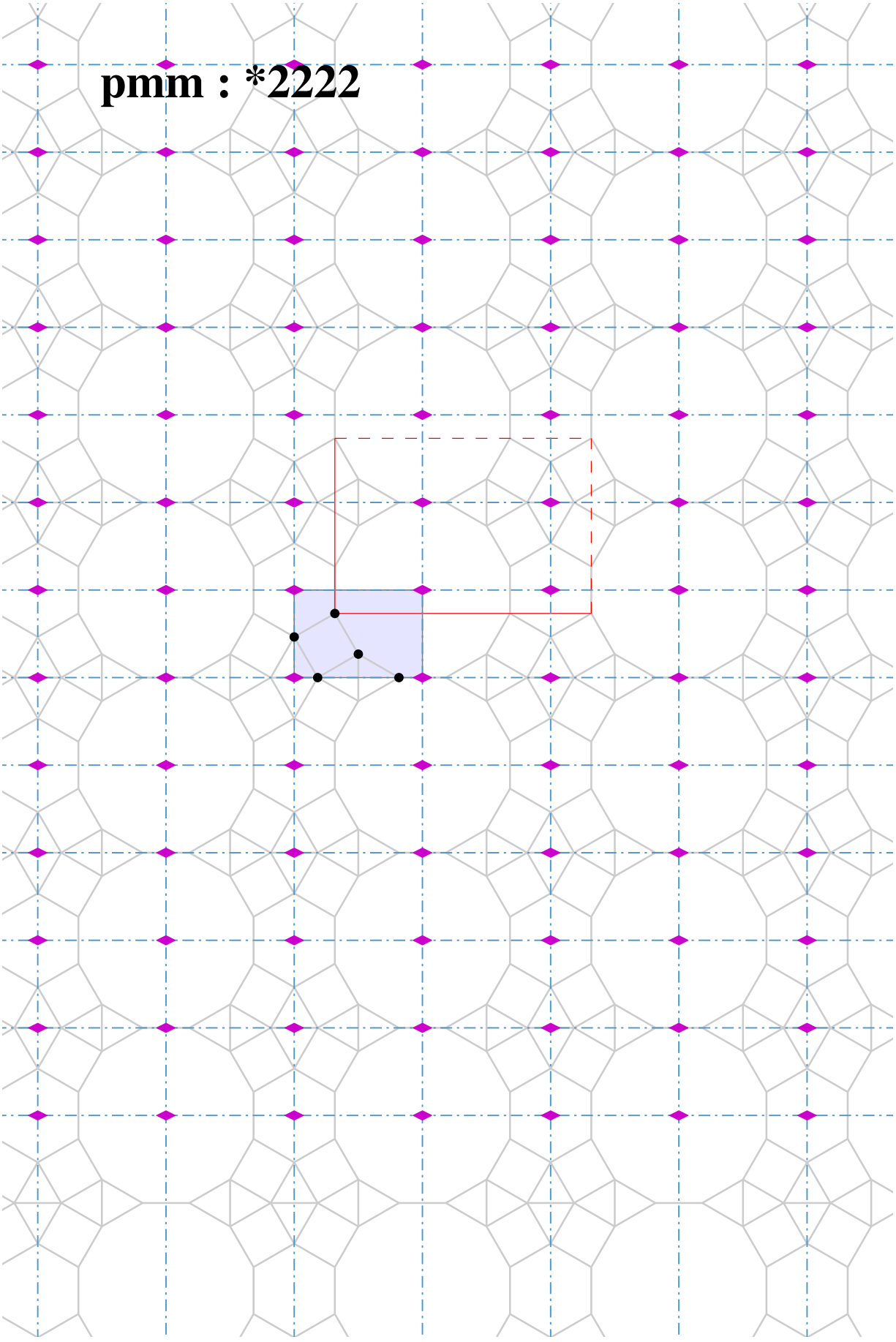



Figure 4.10: *pmm*(∗2222) : *GHLPU* tiling. *GHLPU* tiling from Sa&Sa collection.

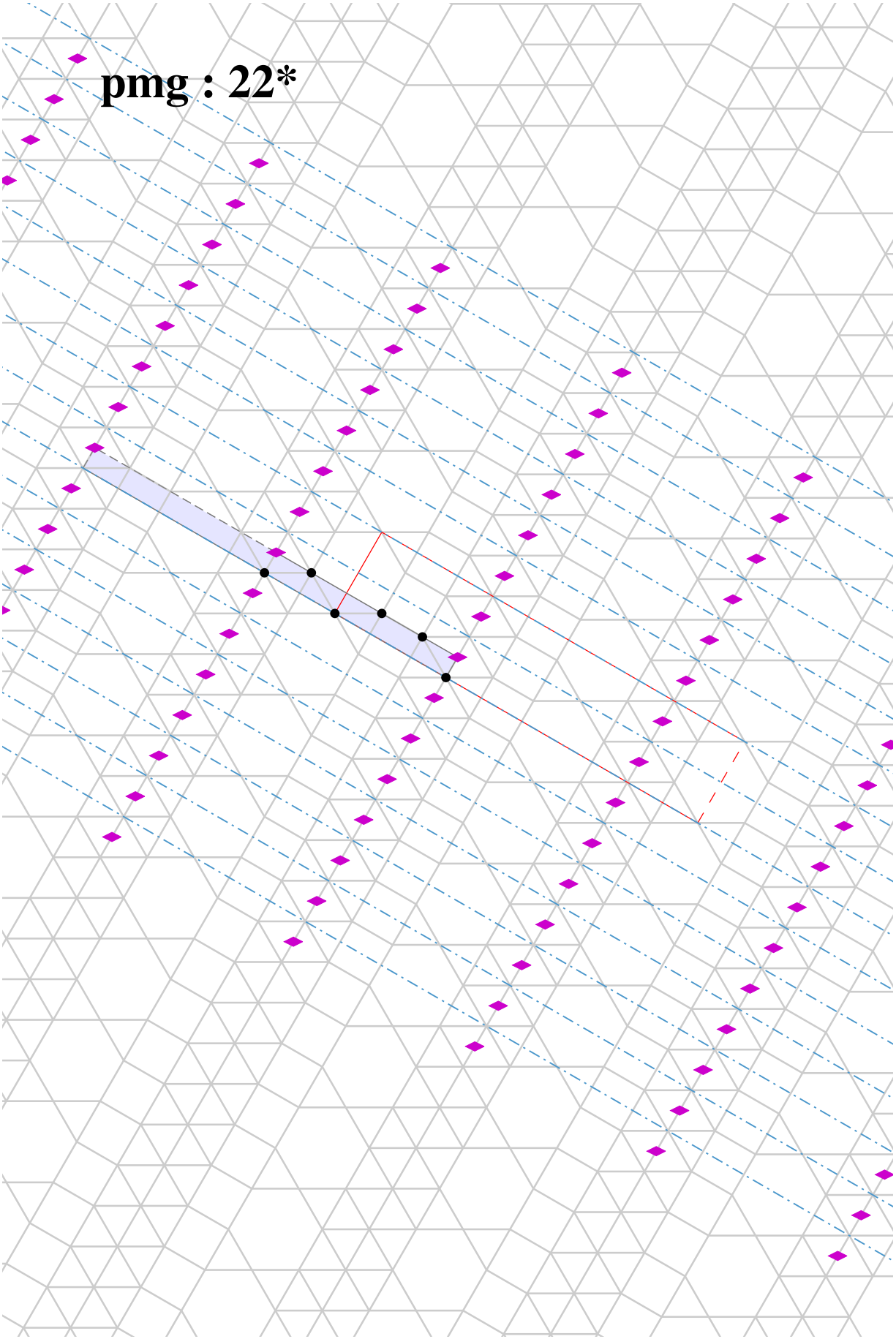



Figure 4.11: $pmg(22*) : NRTVW_2$ tiling. Number 451 from 6-uniform Galebach collection.

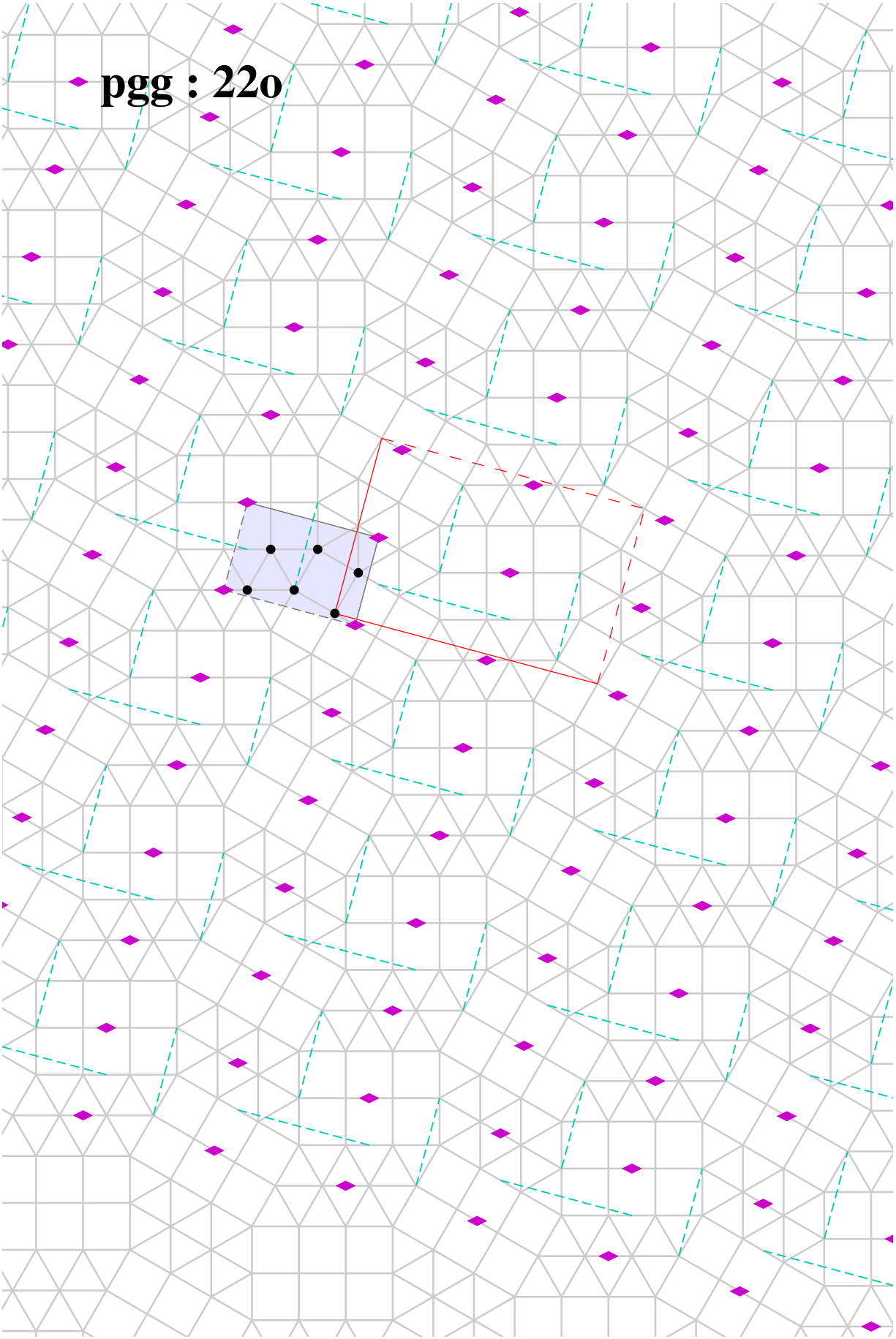



Figure 4.12: *pgg*(22∘) : $ST_3UW$ tiling. Number 364 from 6-uniform Galebach collection.

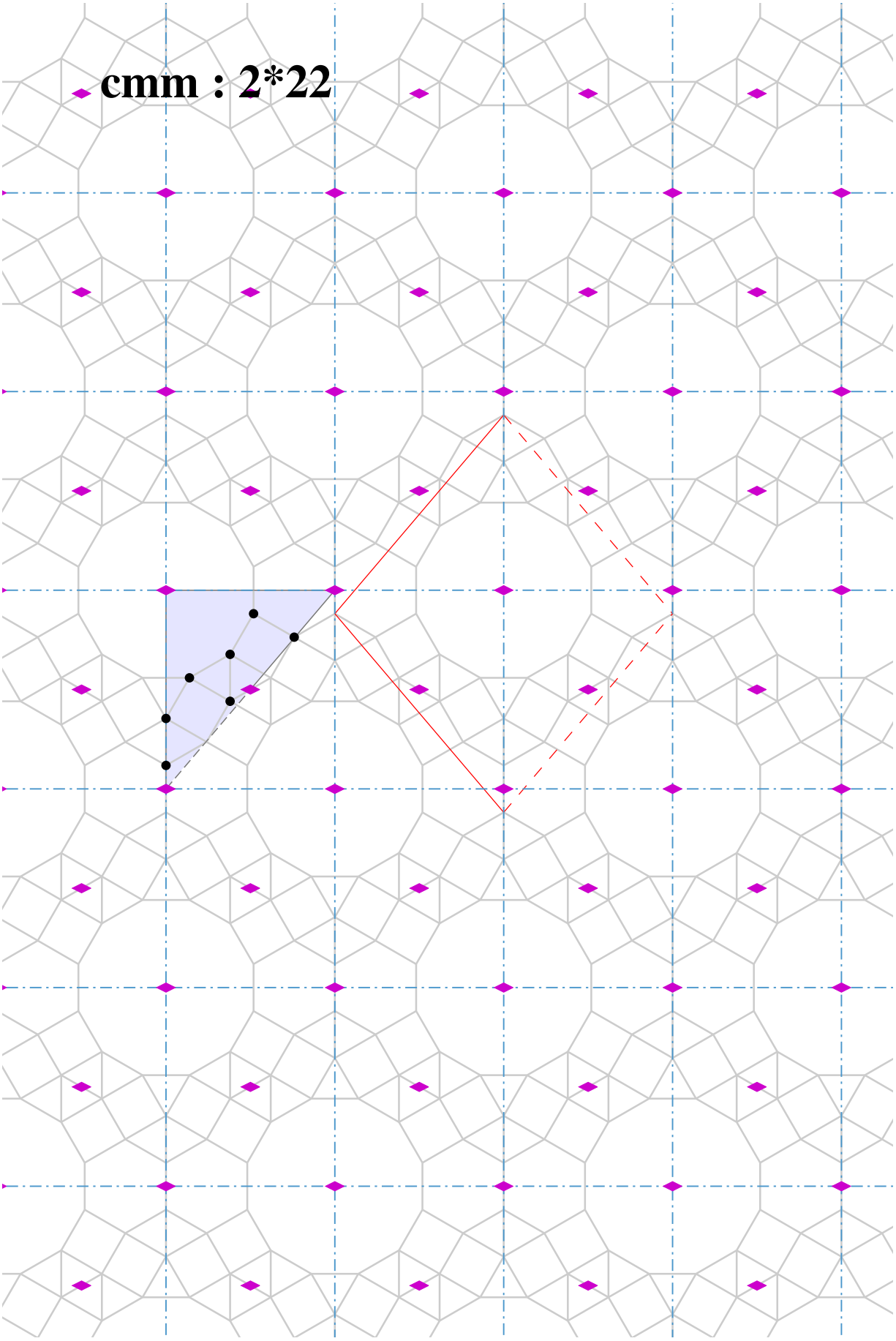



Figure 4.13: *cmm*(2∗22) : *HLMNQTU* tiling. *HLMNQTU* tiling from Sa&Sa collection, number 3 from 7-uniform 7-Archimedean Galebach collection.

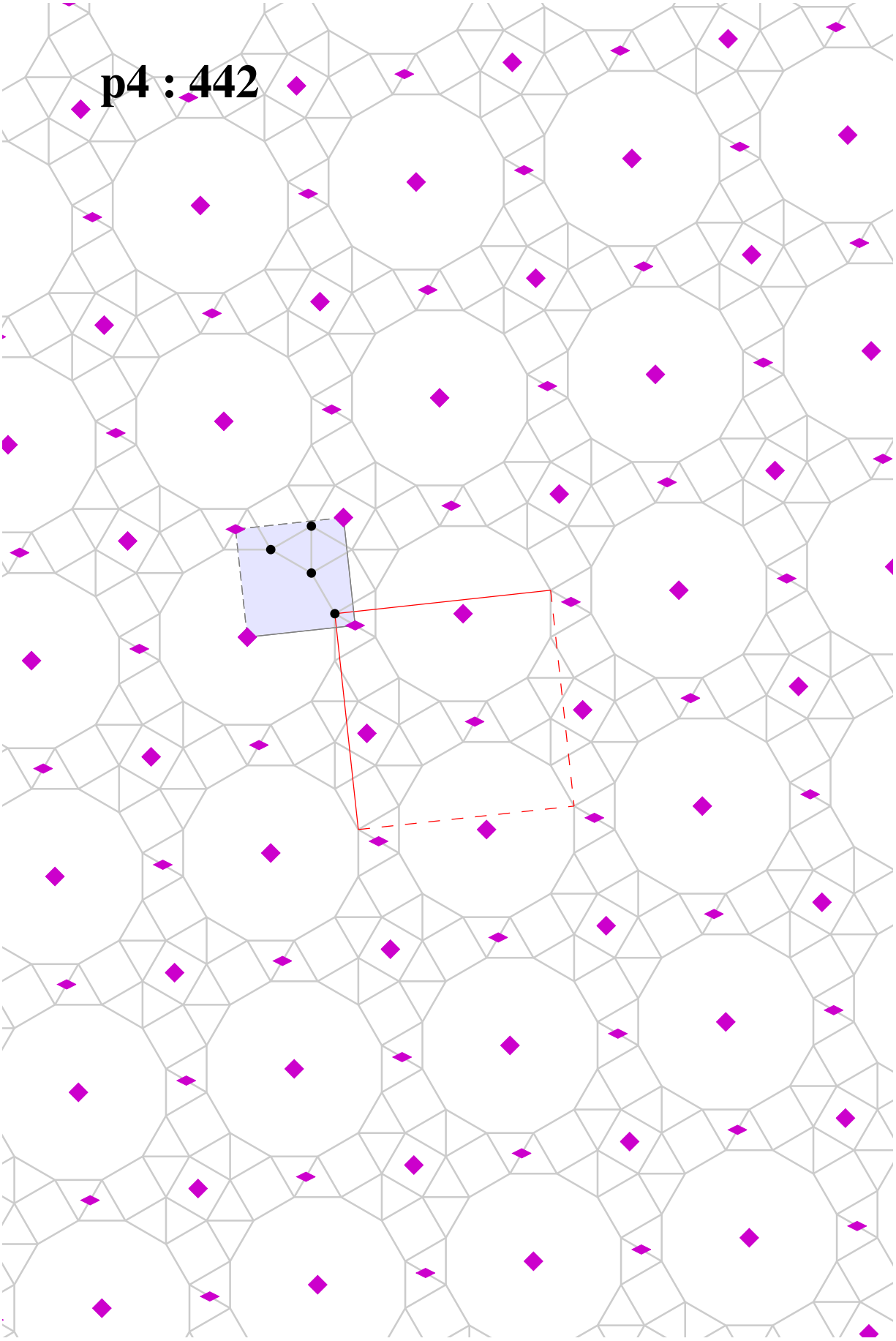



Figure 4.14: $p4(442)$ : $L_2MU$ tiling. $LMU$ tiling from Sa&Sa collection, number 132 from 4-uniform Galebach collection.

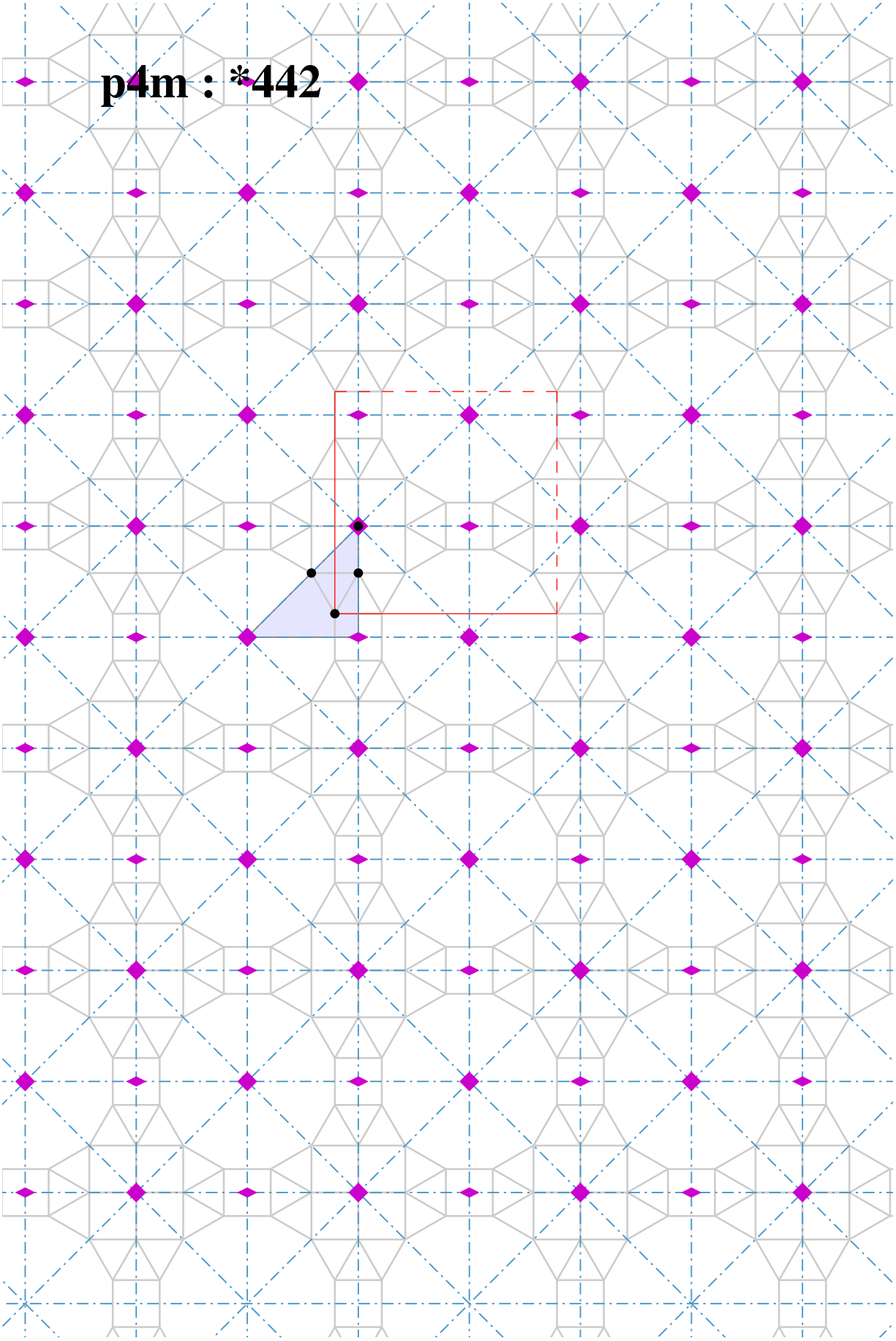



Figure 4.15: $p4m(*442)$ : *LMST* tiling. *LMST* tiling from Sa&Sa collection, number 81 from 4-uniform Galebach collection.

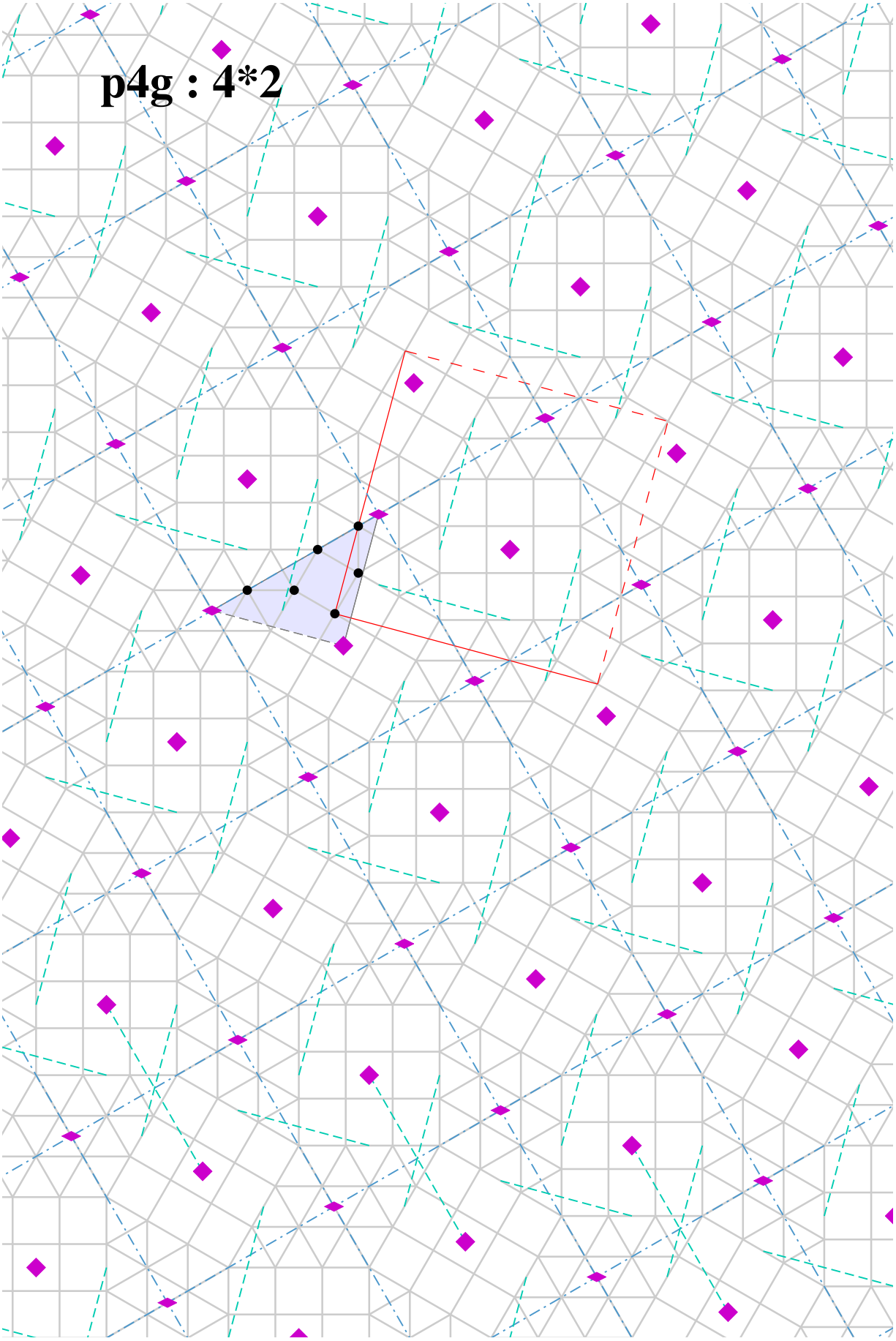



Figure 4.16: $p4g(4*2)$ : $ST_2UW_2$ tiling. Number 215 from 6-uniform Galebach collection.

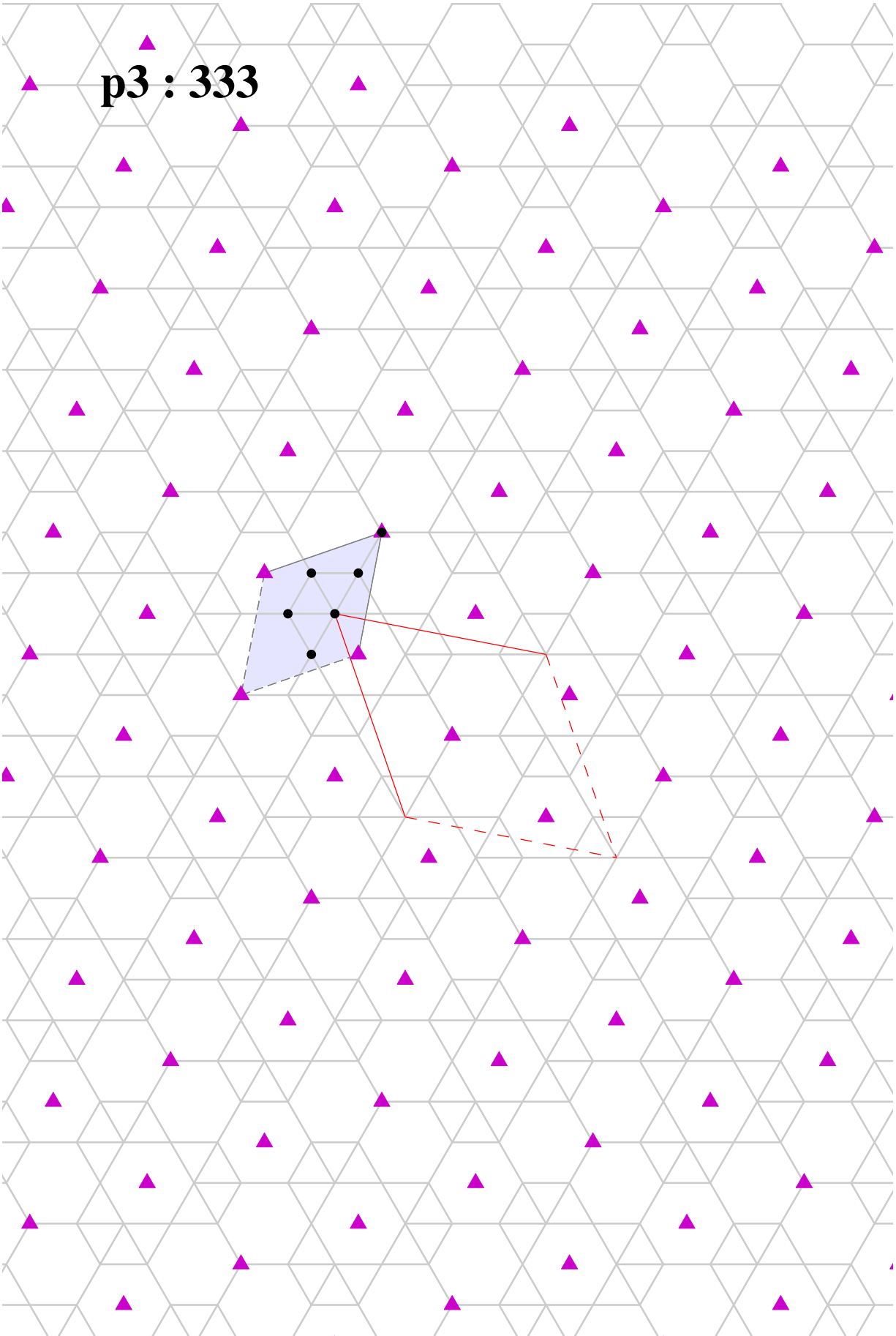



Figure 4.17: $p3(333)$ : $KQ_3RV$ tiling. Number 72 from 6-uniform Galebach collection.

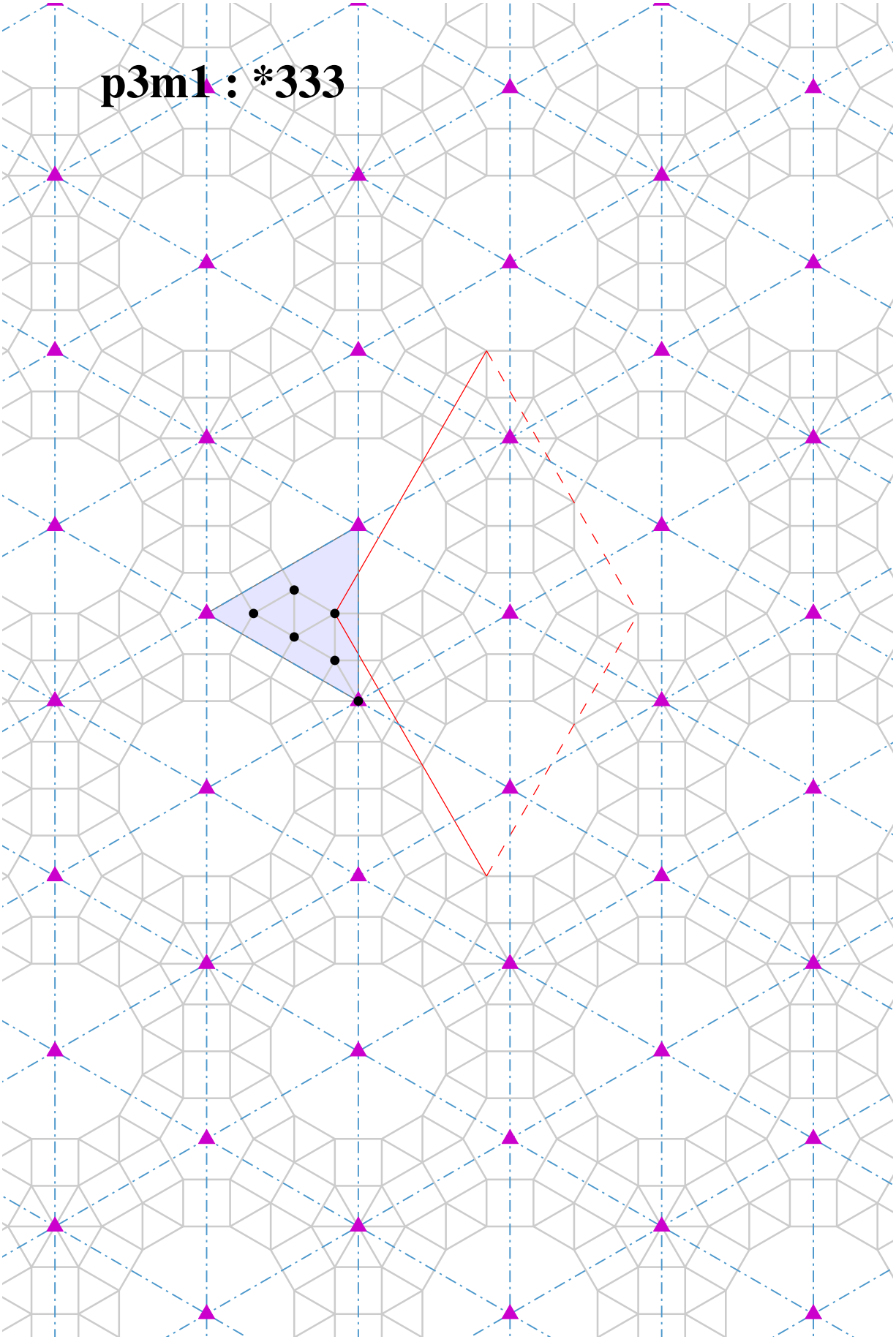



Figure 4.18: $p3m1(*333)$ : $L_2PTUW$ tiling. *LPTUW* tiling from Sa&Sa collection, number 480 from 6-uniform Galebach collection.

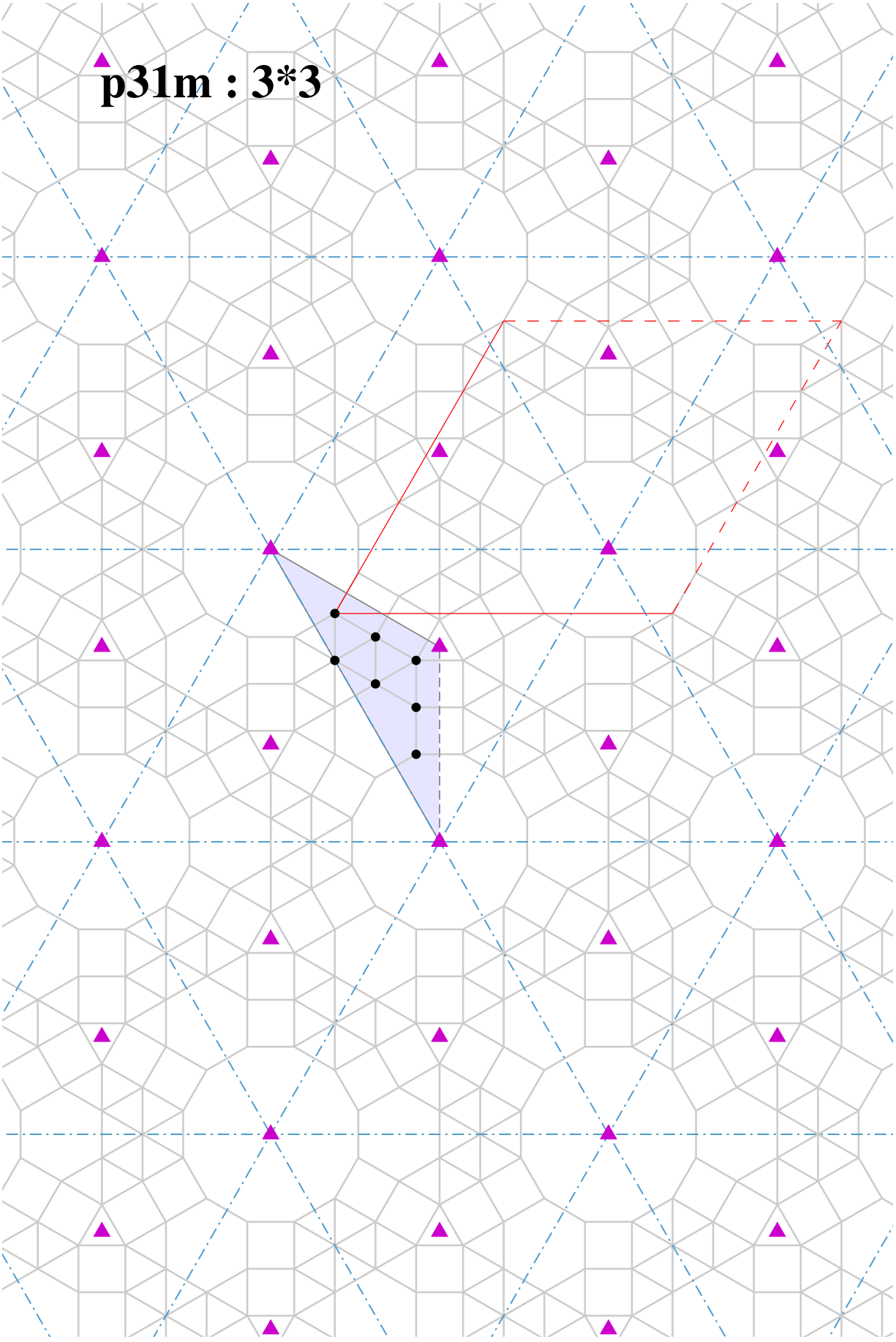



Figure 4.19: $p31m(3*3)$ : $HLNTUVW$ tiling. Number 2 from 7-uniform 7-Archimedean Galebach collection.

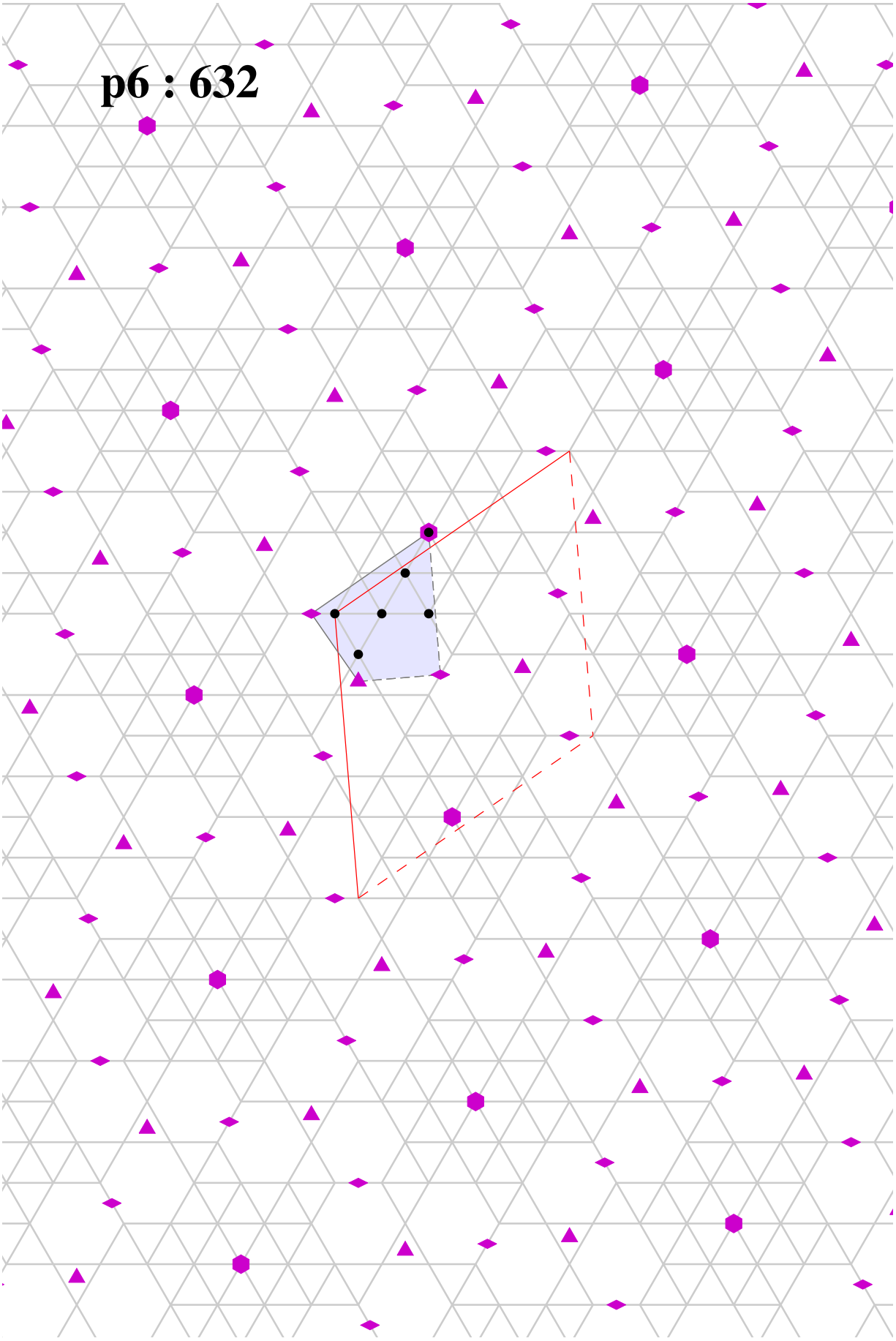



Figure 4.20: $p6(632)$ : $QRV_2W_2$ tiling. Number 540 from 6-uniform Galebach collection.

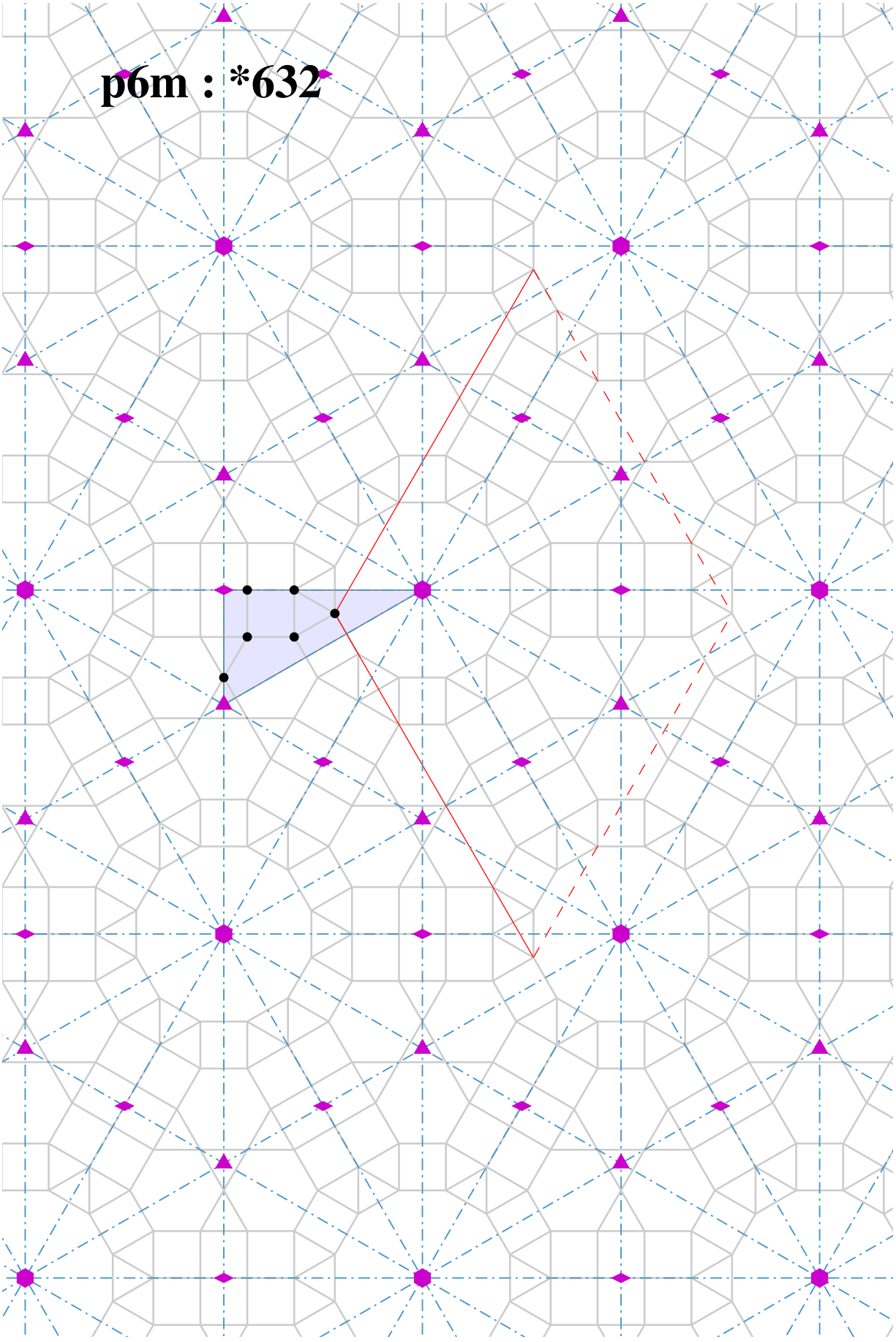



Figure 4.21: $p6m(*632)$ : $LNPRST$ tiling. Number 4 from 6-uniform 6-Archimedean Galebach collection.

## 4.5 *Related work*

Valiente et al. [87] designed a motif symmetry classification by transforming digitized images to Bézier curves and then analyzing their symmetries by explicitly applying symmetry group rules. The main difference with the computer vision approach is that they use graphical entities to compute explicit symmetries for them.

Agustí-Melchor et al. [3] present a novel computational framework for automatic classification method by symmetries, for periodic images applied to content-based image retrieval. They apply an automatic parameter estimation based on the contents of an image and classify based on a distance computation to a set of prototypes.

Liu et al. [57] carefully reviewed the field of computational symmetry, including the long history of symmetry detection algorithms, dating back to 1932. Liu et al. [56] investigate the problem of automatically inferring the lattice structure of near-regular textures in real-world images. The first step toward automatic texel discovery is the detection of repeated interest points present in the image. The key trade-off is to extract enough interest points to reveal some repeated structure reliably without overwhelming the subsequent lattice finder with false positives. Liu et al. [56] report good performance for their algorithm by testing it on the data used in the Symmetry Detection from Real-World Images Competition 2013.

Plachinda [67] applies crystallographic image processing (CIP) to raster images, an automated image processing method used in crystallography to obtain scores for each symmetry group based on Fourier transform of the image.

Moeck [59] discusses several methods applied to crystallographic symmetry detection in noisy images. The methods described rely strongly on statistical measures, given that they work with actual crystallographic imaging, where a lot of pseudo-symmetric structures appear.

Adanova & Tari [2] present a fully automated method to detect the symmetry group and extract fundamental domains of ornaments belonging to 13 symmetry groups. They apply their method to ornaments where the underlying tiling is not evident. They achieve good results obtaining unit cells under the symmetry group, which we have called generating regions.

The problem solved here has a different nature than the methods reported so far in the literature. Our model for periodic tilings with regular polygons has a precise integer representation where symmetries can be computed with absolute precision. We could try to apply some image processing methods over raster images of our tilings, but we would be losing on the properties of the representation, and risking to have an incomplete identification of the symmetry groups, since none of the reported methods can classify correctly all 17 wallpaper groups.

# 5 *Enumeration and Generation*

It is natural to ask whether and how one can generate arbitrarily complex tilings. More precisely, can we find, or at least count, all the tilings satisfying some constraints? Constraints like the area of the fundamental region, the number of vertices or faces, etc. Is there a set of equations or expressions that determine such tilings?

We shall focus on *triangle-square tilings*, that is, tilings formed solely by triangles and squares, because these tilings contain all tilings by regular polygons after refinement. Through this chapter, we explain a representation of triangle-square tilings via equivalence with edge-labeled hexagonal graphs, which characterizes these tilings completely. An algebraic structure for families of tilings whose dual graph is embedded in congruent plane tori arises from this representation. This algebraic structure allows the computation of a basis for the tilings in each topological setting and hence an enumeration of the infinite family of tilings spanned by each basis. Moreover, the basis' elements of each family correspond to fundamental *unfoldings* of the triangle tiling. This implies every triangle-square tiling is an unfolding of the triangle tiling, and so, it admits two distinct *foldings* into triangle tilings.

A triangle-square tiling $\mathcal{T}$ is a combinatorial arrangement of triangles and squares on the plane, which is completely determined by one of two complementary sets of triangles $\mathcal{E}$ and $\mathcal{L}$, present in the tiling. Observe the tiling in Figure 5.1 and its colored version, with $\mathcal{E}$ in red and set $\mathcal{L}$ in gray. A tiling containing triangles satisfies that either one of these sets is not empty and both have pair cardinality.

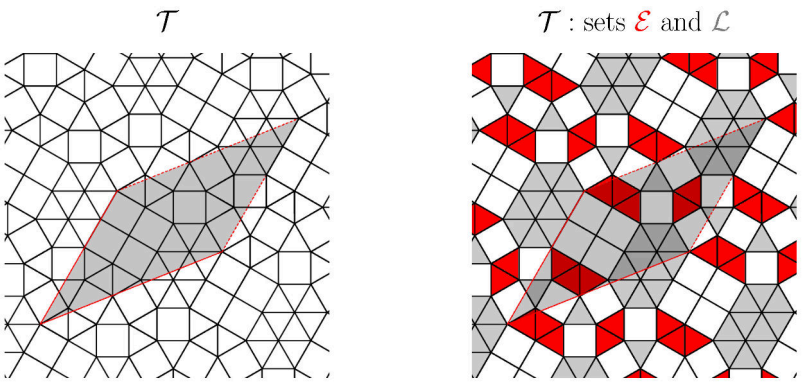



Figure 5.1: Tiling $\mathcal{T}$, with sets of triangles $\mathcal{E}$ and $\mathcal{L}$ colored.

We define the *restricted dual tilings* ($\mathcal{T}_\mathcal{E}^*$ and $\mathcal{T}_\mathcal{L}^*$) as the dual tilings on each of these sets, and analyze its combinatorial structure. In our example, illustrated in Figure 5.2, one can see the restricted duals and how they correspond to two connected components of $\mathcal{T}^*$, the dual tiling of $\mathcal{T}$.

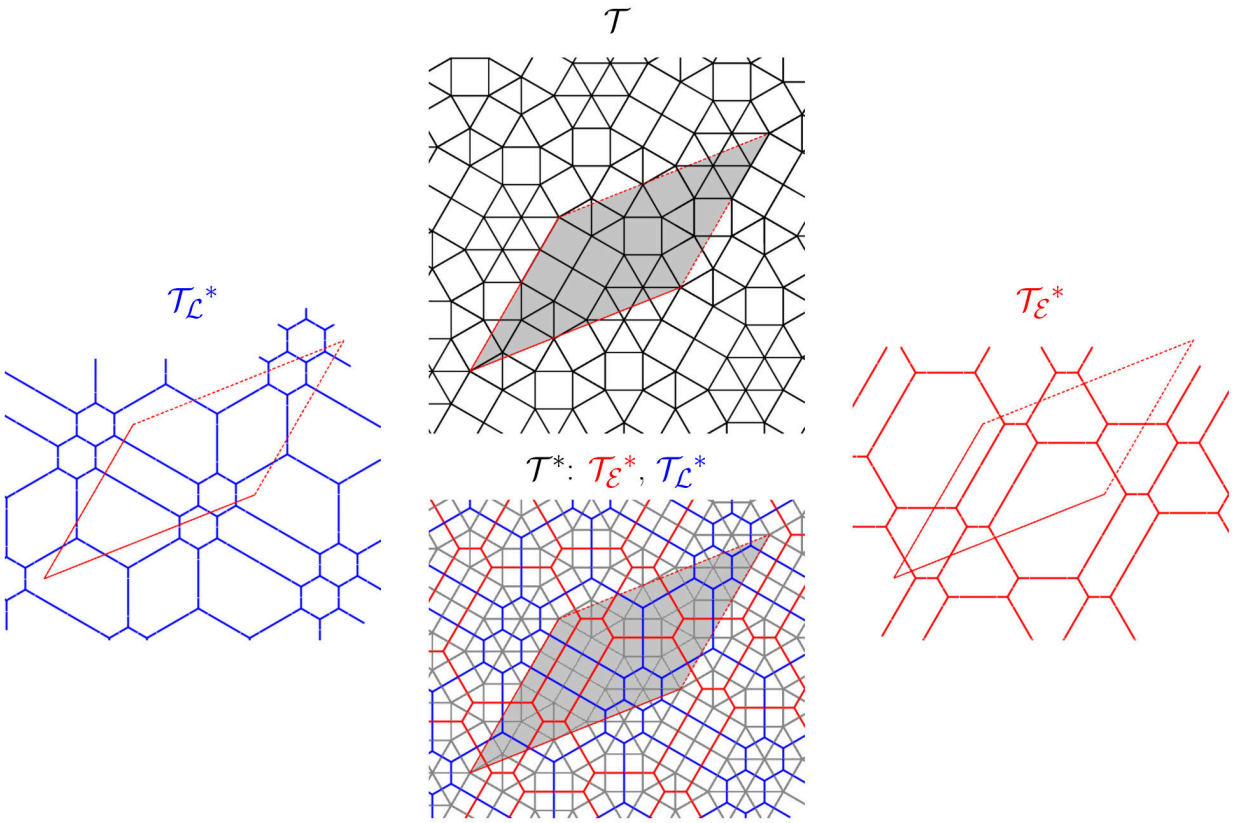



Figure 5.2: Tiling $\mathcal{T}$. Below: $\mathcal{T}^*$, its dual, colored by its two separated components: $\mathcal{T}_\mathcal{E}^*$ and $\mathcal{T}_\mathcal{L}^*$. The restricted dual tilings are shown isolated on the sides.

Each restricted dual tiling is isomorphic to a hexagonal graph, as one can observe. We focus on $\mathcal{T}_\mathcal{E}^*$, but the analysis for $\mathcal{T}_\mathcal{L}^*$ is analogous, they differ only by a rotation by $\omega$. The number of squares that each edge of the restricted dual crosses can be encoded in the graph, as edge labels. This way, we define $\mathcal{G}_\xi^*$, the *edge-labeled dual hexagonal graph* of $\mathcal{T}$, with labels $\xi$. Figure 5.2 illustrates $\mathcal{G}_\xi^*$, the edge-labeled dual hexagonal graph of $\mathcal{T}$, corresponding to $\mathcal{T}_\mathcal{E}^*$.

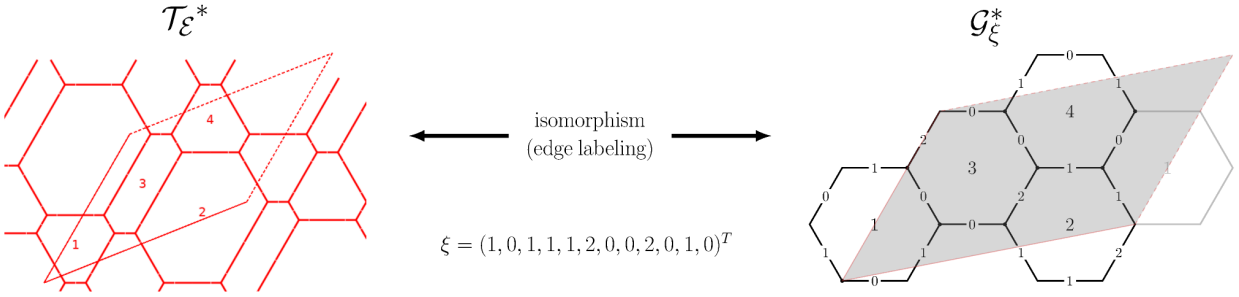



Figure 5.3: Restricted dual tiling $\mathcal{T}_\mathcal{E}^*$, and its corresponding edge-labeled dual hexagonal graph $\mathcal{G}_\xi^*$.

We draw the graph $\mathcal{G}_{\tilde{\xi}}^*$ as a hexagonal tiling. The dual of a hexagonal tiling is a triangular one. Thus, the primal $\mathcal{G}_{\tilde{\xi}}$ of the edge-labeled dual hexagonal graph is a triangle tiling with labels on its edges. Figure 5.4 illustrates both $\mathcal{G}_{\tilde{\xi}}^*$ and $\mathcal{G}_{\tilde{\xi}}$ with their corresponding labels.

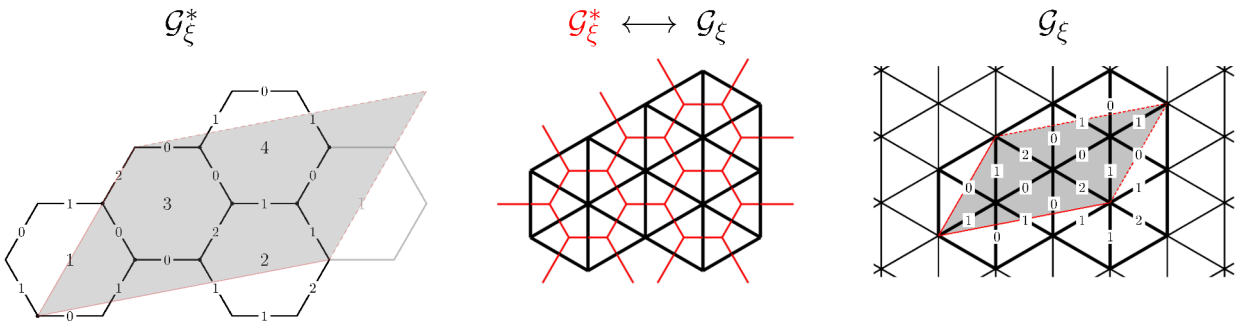



Figure 5.4: Restricted dual tiling $\mathcal{T}_{\mathcal{E}}{}^*$, and its corresponding edge-labeled dual hexagonal graph $\mathcal{G}_{\tilde{\xi}}^*$.

$\mathcal{G}_{\tilde{\xi}}$ is a *folded* version of $\mathcal{T}$, obtained by folding the triangle faces in set $\mathcal{E}$ over the rest of the tiling using $\mathcal{T}_{\mathcal{L}}{}^*$ as the valleys of a *crease pattern*. In origami mathematics, a crease pattern is a diagram containing the lines through which the paper is folded. Creases through which the paper is folded downwards are called *mountains* (concave folds), and *valleys* (convex folds) when the fold is upwards [42]. Figure 5.5 illustrates $\mathcal{G}_{\tilde{\xi}}$ obtained by folding $\mathcal{T}$ over the crease pattern with valleys on $\mathcal{T}_{\mathcal{L}}{}^*$. The labels indicate the number of folds under each edge.

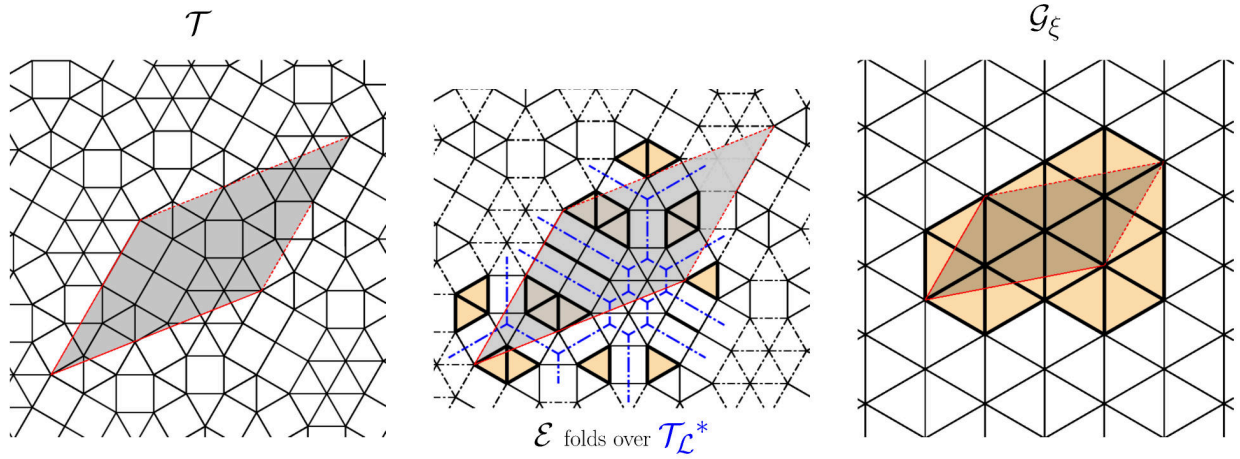



Figure 5.5: Restricted dual tiling $\mathcal{T}_{\mathcal{L}}{}^*$ used as an origami crease-pattern to obtain the folded tiling $\mathcal{G}_{\tilde{\xi}}$.

All the results for set $\mathcal{E}$ have their analog on the complementary set of triangles $\mathcal{L}$. We show how the two representations relate through their corresponding labelings. $\mathcal{T}_{\mathcal{L}}{}^*$ defines the valleys to obtain the folding $\mathcal{G}_{\tilde{\xi}}$, with triangles on $\mathcal{E}$, and vice-versa, $\mathcal{T}_{\mathcal{E}}{}^*$ defines the valleys to obtain the folding $\mathcal{G}_{\lambda}$, with triangles on $\mathcal{L}$.

Figure 5.6 and Figure 5.7 show the relations between the objects defined through this chapter.

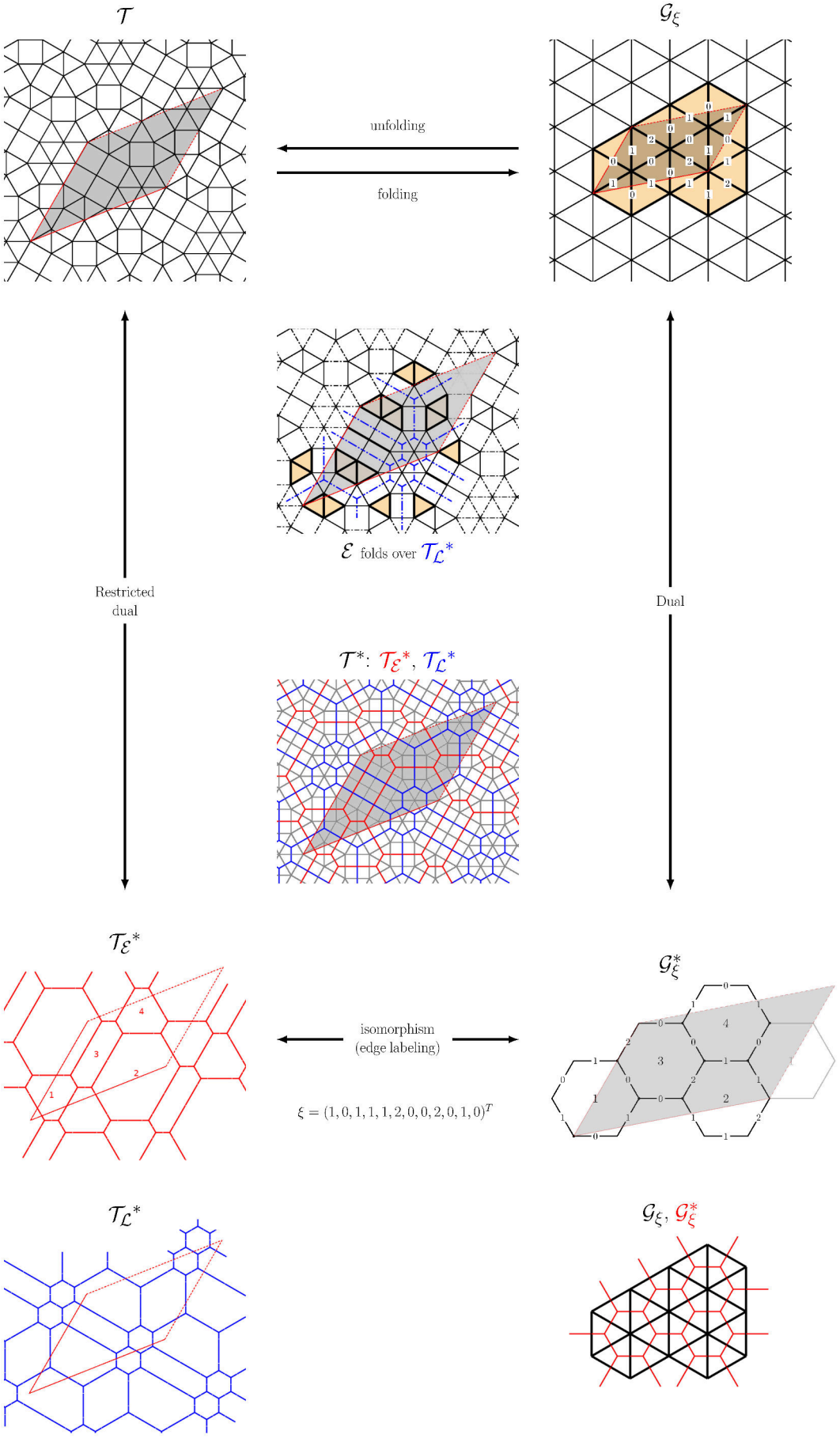



Figure 5.6: Relations between a triangle-square tiling $\mathcal{T}$, its restricted duals $\mathcal{T}_\mathcal{E}{}^*$ and $\mathcal{T}_\mathcal{L}{}^*$, the edge-labeled dual hexagonal graph $\mathcal{G}_\xi^*$ and the folded triangle tiling $\mathcal{G}_\xi$.

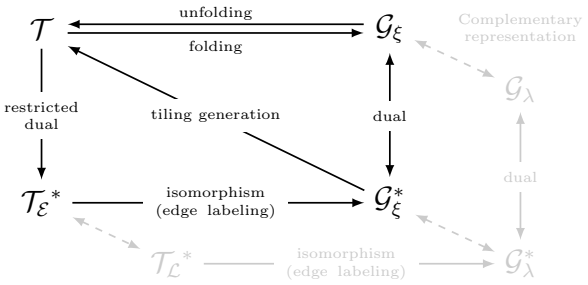



Figure 5.7: Relations between a triangle-square tiling $\mathcal{T}$, its restricted dual $\mathcal{T}_\mathcal{E}{}^*$, the edge-labeled dual hexagonal graph $\mathcal{G}_\xi^*$ and the folded triangle tiling $\mathcal{G}_\xi$.

The enumeration and generation of triangle-square tilings is given by the equivalence of $\mathcal{T}$ with the edge-labeled dual hexagonal graph $\mathcal{G}_\xi^*$. This is done by enumerating all possible plane tori with hexagonal tilings with $h$ faces, which corresponds to triangle-square tilings with $2h$ triangles in set $\mathcal{E}$. Over each topology indexed by a pair $(i,h)$, we deduce the geometric constraints that define a valid labeling. A valid labeling ensures that the dual hexagonal graph corresponds to a triangle-square tiling. For each family $\mathbb{T}_{(i,h)}$ of tilings in the $(i,h)$ topology, a Hilbert basis for the set of valid labelings is computed. Every valid labeling is obtained as a positive integer linear combination of elements in the basis, which correspond to fundamental minimal *unfoldings* of the triangle tiling.

Figure 5.8 illustrates the algebraic structure of tilings derived from this representation. In it, the tiling from the examples above is obtained from the linear combination of two basic labelings from family $\mathbb{T}_{(3,4)}$, to which the tiling in our examples belongs. The operations between labelings are, in fact, operations between tilings in the same topology, as Figure 5.8 illustrates.

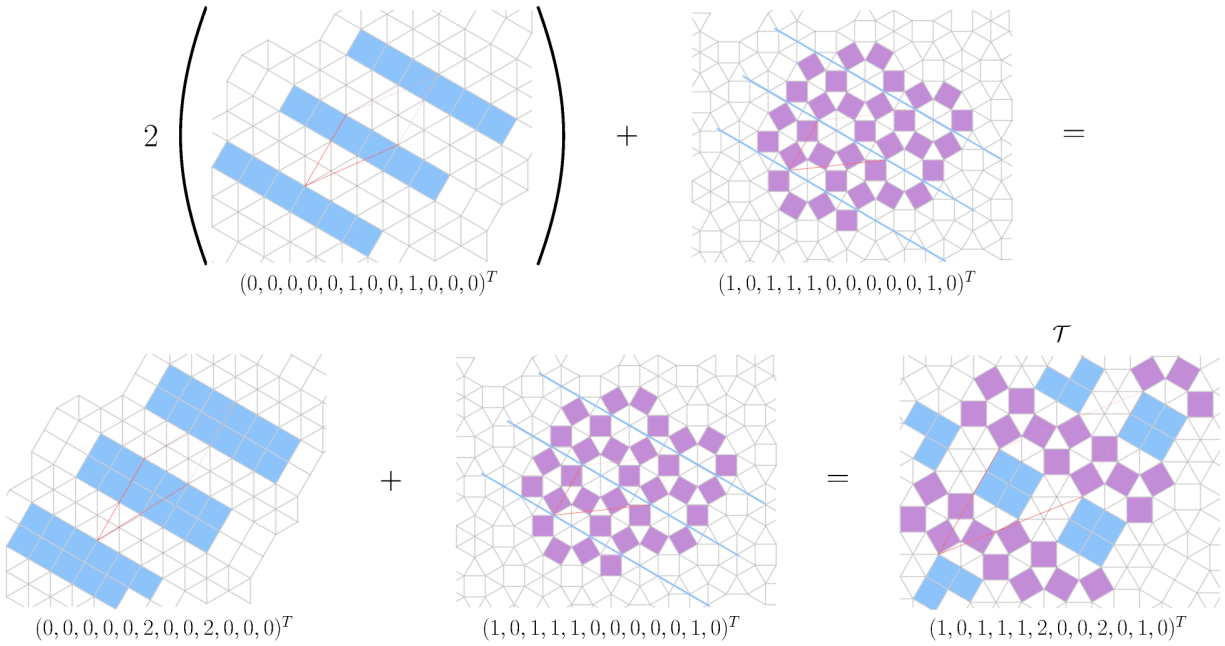



Figure 5.8: Example of the positive linear combination of two basic tilings.

The Hilbert bases are used for the enumeration of the infinite families of triangle-square tilings. Later, subsets of these basis are used for the generation of a very large sample set of tilings. This chapter covers the theoretic and algorithmic elements for all that has been described, organized in the following way.

In Section 5.1, we define triangle-square tilings and relate them to the notion of *dense tilings*. We also show how they contain all the other tilings with regular polygons.

In Section 5.2, we discuss the complementary sets of triangles present in a tiling, and define the restricted dual tiling and the edge-labeled dual hexagonal graph. The edge-labeled dual hexagonal graph is the main combinatorial structure we are going to use for the analysis and generation of triangle-square tilings.

Section 5.3 is dedicated to analyze the possible translation lattices on which hexagonal tilings can be embedded. We derive an enumeration and generation scheme for the plane tori tiled with $h$ hexagons.

In Section 5.4, we analyze the geometric constraints that define a valid labeling for a dual hexagonal graph $\mathcal{G}^*_\xi$. A valid labeling $\xi$ is one such that $\mathcal{G}^*_\xi$ is isomorphic to some restricted dual tiling $\mathcal{T}_\mathcal{E}{}^*$. We derive the equations to find valid labelings. From these equations, we derive the existence of a Hilbert basis for each tiling family in a topological setting and compute its elements.

Section 5.5 is dedicated to explain in detail the generation algorithm that allows us to compute a triangle-square tiling $\mathcal{T}$ directly from a valid labeling on $\mathcal{G}^*_\xi$.

In Section 5.6, we derive formulas for computing the area of the generated triangle-square tiling directly from a valid labeling on $\mathcal{G}^*_\xi$. We analyze how the area formulas define the relationship between two complementary labelings on Section 5.7. We also propose a *complexity measure* for triangle-square tilings based on the complementary properties of the edge-labeled dual graphs.

Section 5.8 is devoted to the generation of a sample set of tilings with the proposed methodology. For each $h \in \{1, 2, \ldots, 20\}$, for each torus with a hexagonal tiling with $h$ faces, we compute all the binary combinations of a set of generators (a subset of the Hilbert basis), obtaining a set of more than 100 million tilings. We compute and analyze some general statistics for the sample.

In Section 5.9, we analyze the tiling basis and the algebraic structure of the tiling from a geometric point of view. We show that the representation proposed, and the operations on the tilings derived from it, can be interpreted as *unfoldings* of a triangle tiling, the dual of $\mathcal{G}^*_\xi$.

Finally, in Section 5.10 we make some remarks on the work in progress and the research paths that are opened with the proposed representation of triangle-square tilings. There are plenty of details that await to be fully understood and some immediate applications in which we foresee future work. In Section 5.11 we collect the main related works in the literature.

## 5.1 *Triangle-square tilings*

We represent periodic tilings of the plane with regular polygons as regular systems of points, in the sense of Hilbert & Cohn-Vossen [40], which satisfy the representation theorem (Section 2.7). The vertices of a periodic tiling with regular polygons form cosets of the additive group generated by the translation vectors.

*Triangle-square* tilings, formed by triangles and squares only, have vertices that form *dense systems of points* in $\mathbb{Z}[\omega]$. Given a translation lattice and a set of vertices, the set is *dense* if no more vertices can be inserted in the fundamental region such that the set continues to satisfy the representation theorem. A dense system of points is the maximal point set containing the given tiling's vertices, and satisfying the representation theorem with the same translation lattice.

In a dense system of points, the star of a vertex $v$ does not have consecutive edges $\omega^k$ and $\omega^{k+4}$, since the vertex at $\omega^{k+2}$ makes the system denser and satisfies the representation theorem, as shown in Figure 5.9. The case for consecutive edges $\omega^k$ and $\omega^{k+5}$ is similar; however, a point can be either at $\omega^{k+2}$ or $\omega^{k+3}$, see Figure 5.9. In either case, stars with these properties span only triangles and squares, eliminating the possibility of regular hexagon and dodecagon faces.

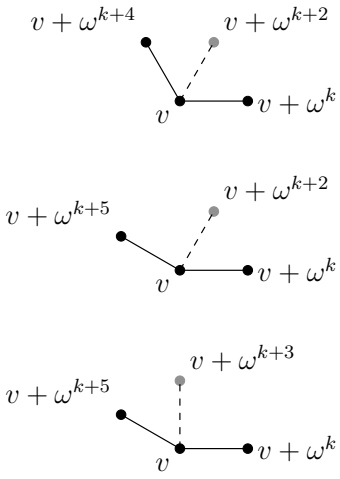



Figure 5.9: Density criteria on vertex stars.

The argument analyzing the density at the vertex level shows a natural process of *vertex insertion* to make a tiling dense. When doing this, one has to ensure that the inserted points satisfy the transitivity of the internal edge angles, condition (e) in the representation theorem, so that the insertion of points between $\omega^k$ and $\omega^{k+5}$ edges is consistent. The result of such insertion is shown in Figure 5.10. There is only one possibility for the hexagon but two for the dodecagon.

Thus, dense tilings are obtained by inserting one vertex at the centroid of each hexagon and 7 vertices inside each dodecagon in one of the two possible ways. The described process has an inverse *vertex elimination* procedure that enables the construction of regular hexagon and dodecagon faces from dense tilings by removing some vertices of type $W$ or $U_6W$ stars, respectively (Figure 5.10).

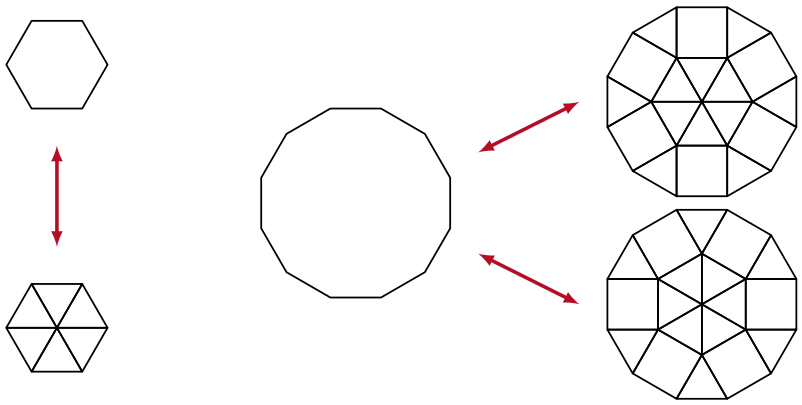


Figure 5.10: Vertex insertion and removal processes for regular hexagon and dodecagon faces.

Every tiling with regular hexagon and dodecagon faces is a subset of a dense tiling. Thus, instead of inserting vertices into existing tilings, we restrict our combinatorial representation effort to triangle-square periodic tilings. Triangle-square tilings have been considered as a family on their own previously in the literature of different areas [18, 65, 66]. We shall consider vertex removal later as a separate process over a generated triangle-square tiling.

Some remarks before dedicating ourselves to the combinatorial representation of triangle-square tilings:

- The only Archimedean vertex types present in triangle-square tilings are: $S, T, U, W$ (Figure 5.11).

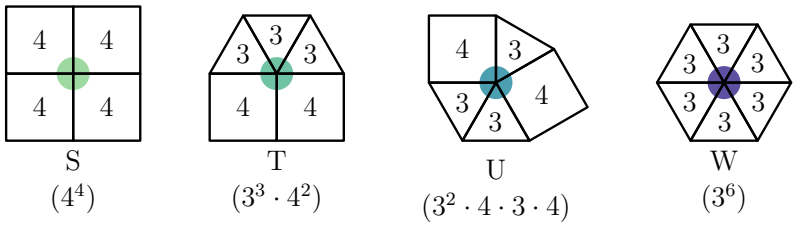



Figure 5.11: Archimedean vertex types in triangle-square tilings.

- The area of the fundamental region of a triangle-square tiling tells us exactly the number and the nature of the faces. In Section 2.4, we showed that the area of any fundamental region has the form: $a + b(\sqrt{3}/2)$. This corresponds to $a$ squares and $2b$ triangles, for triangle-square tilings.

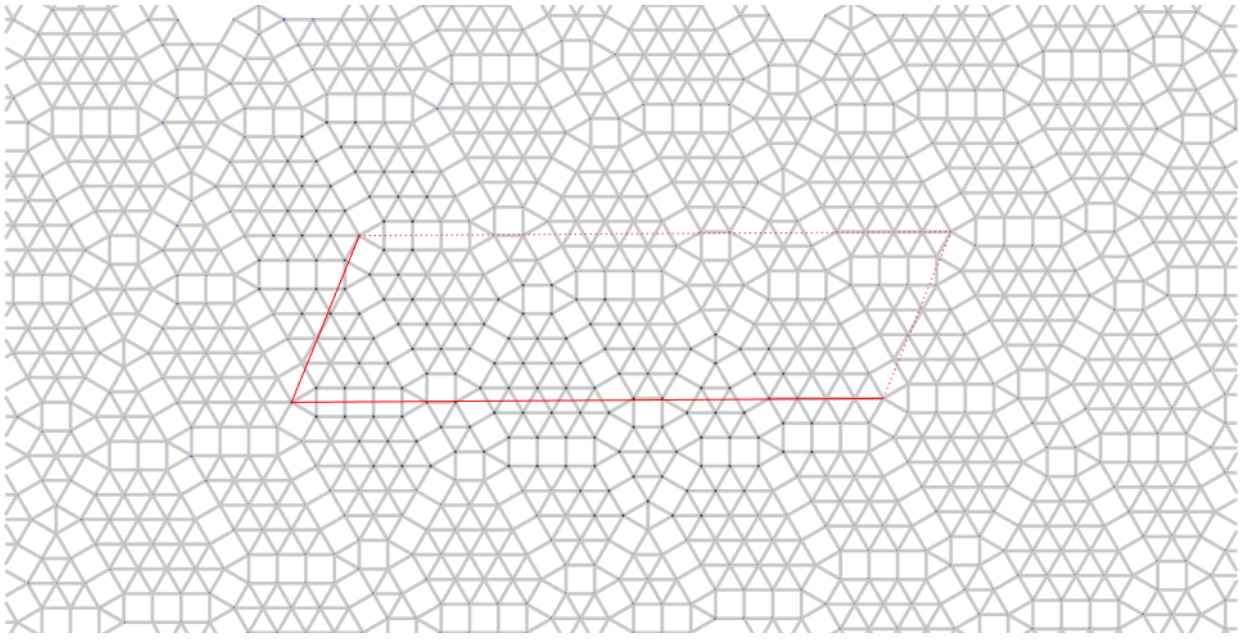


Figure 5.12: Example of a large complex triangle-square tiling.

## 5.2 *Restricted dual tilings and graphs*

Let $\mathcal{T}$ be a *triangle-square tiling*, as described in Section 5.1. We start by observing that each triangle in $\mathcal{T}$ can appear in exactly 4 positions, which can only be paired in specific ways, as shown in Figure 5.13. Thus, a triangle in $\mathcal{T}$ can only be adjacent to either a triangle in a pairing position — that is, with sides in the same orientation — or a square. Therefore, there are two disjoint sets of triangles in the tiling, as illustrated in Figure 5.14. The first set, called $\mathcal{E}$, has edges with orientations in $\Lambda(\omega, \omega^3)$; the second set, called $\mathcal{L}$, has edges in $\Lambda(1, \omega^2)$. These two sets are interchangeable by a rotation by $\omega$ and are present in every tiling aligned with the basic directions.

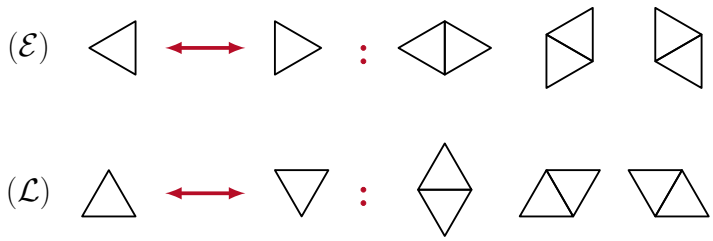



Figure 5.13: Edge-adjacent pairs of triangles form disjoint sets, the first in $\Lambda(1, \omega^2)$ and the second by $\Lambda(\omega, \omega^3)$.

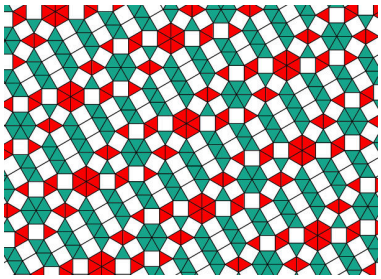


Figure 5.14: Disjoint sets of triangles in a triangle-square tiling: $\mathcal{E}$ colored in red, and $\mathcal{L}$ colored in green.

We define the *restricted dual tilings* of $\mathcal{T}$, denoted ${\mathcal{T}_\mathcal{E}}^*$ and ${\mathcal{T}_\mathcal{L}}^*$, as the pair of tilings generated by the dual vertices corresponding to the triangles in sets $\mathcal{E}$ and $\mathcal{L}$, respectively. Restricted dual tilings are illustrated in Figure 5.15. The dual vertices corresponding to square faces are disregarded. The edges of the restricted dual pass through the squares without changing direction, connecting two vertices corresponding to triangles in the same set only. The edges of ${\mathcal{T}_\mathcal{E}}^*$ are aligned with directions generated by $\{1, \omega^2\}$, and the edges of ${\mathcal{T}_\mathcal{L}}^*$ with directions spanned by $\{\omega, \omega^3\}$.

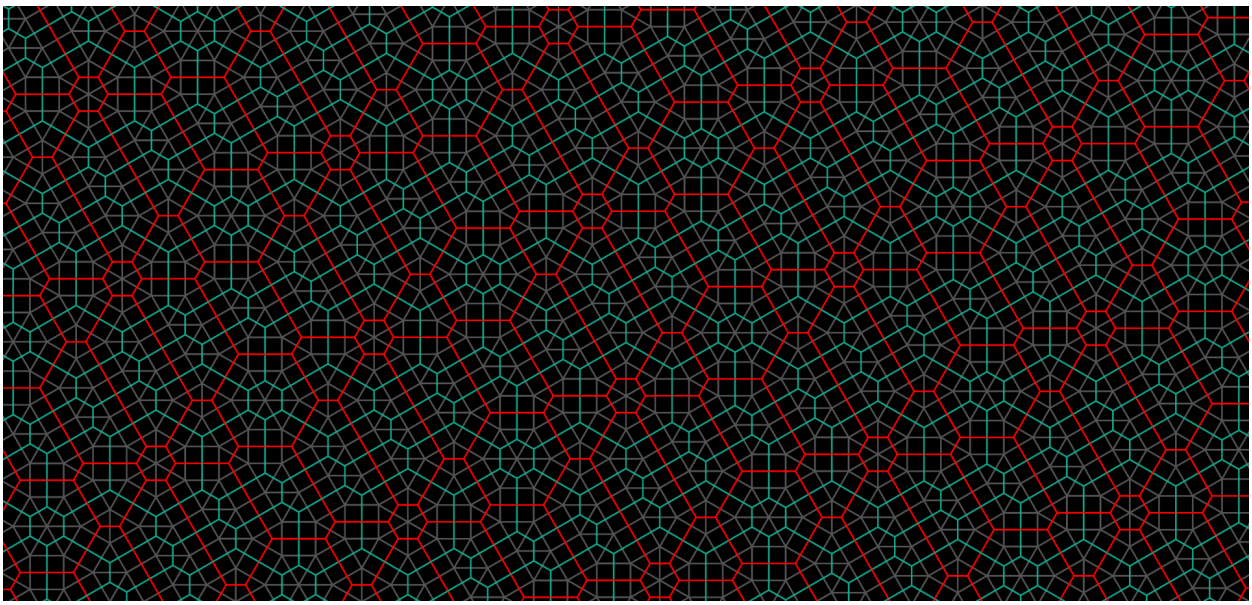


Figure 5.15: Triangle-square tiling with both restricted dual tilings: ${\mathcal{T}_\mathcal{E}}^*$ colored in red, and ${\mathcal{T}_\mathcal{L}}^*$ colored in green.

Another way to look at the restricted dual tilings is as if they were edge-colorings of $\mathcal{T}^*$, the dual of $\mathcal{T}$. We attribute a color to each edge of $\mathcal{T}^*$ based on its orientation with respect to the basic directions. Red is attributed to edges aligned with $\Lambda(1, \omega^2)$ and blue to edges aligned with $\Lambda(\omega, \omega^3)$. The edges of $\mathcal{T}^*$ on each face of $\mathcal{T}$ are colored as shown in Figure 5.16.

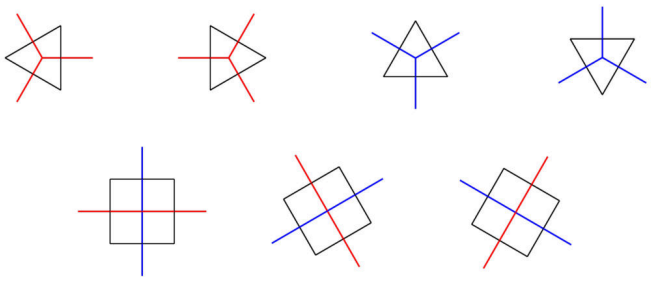


Figure 5.16: Faces of $\mathcal{T}$ and their corresponding edges in $\mathcal{T}^*$, colored.

The edge-coloring defines two separable connected components of $\mathcal{T}^*$, which are exactly the two restricted dual tilings. Triangles in the same set correspond to vertices in the same colored component of $\mathcal{T}^*$, while vertices corresponding to the squares are disregarded for being contained between parallel edges of the same color. The red component corresponds to ${\mathcal{T}_\mathcal{E}}^*$ and the blue one to ${\mathcal{T}_\mathcal{L}}^*$. Then, the restricted dual tilings are complementary: together, they form the proper dual of $\mathcal{T}$.[1]

[1] One could write that: $\mathcal{T}^* = {\mathcal{T}_\mathcal{E}}^* \bowtie {\mathcal{T}_\mathcal{L}}^*$, to represent this, $\bowtie$ meaning the overlay of the two tilings with vertex insertion on the edges intersections.

There are tilings where both orientations of triangles are not present, that is, $\mathcal{E} = \emptyset$ or $\mathcal{L} = \emptyset$. The squares tiling ($S$ Archimedean tiling) is such that $\mathcal{E} = \mathcal{L} = \emptyset$; its restricted dual is composed of closed geodesics over the plane torus, with no vertices. We define the restricted dual $\mathcal{T}^*_\emptyset$ as a set of parallel geodesics through the torus for consistency only. It is illustrated in Figure 5.17.

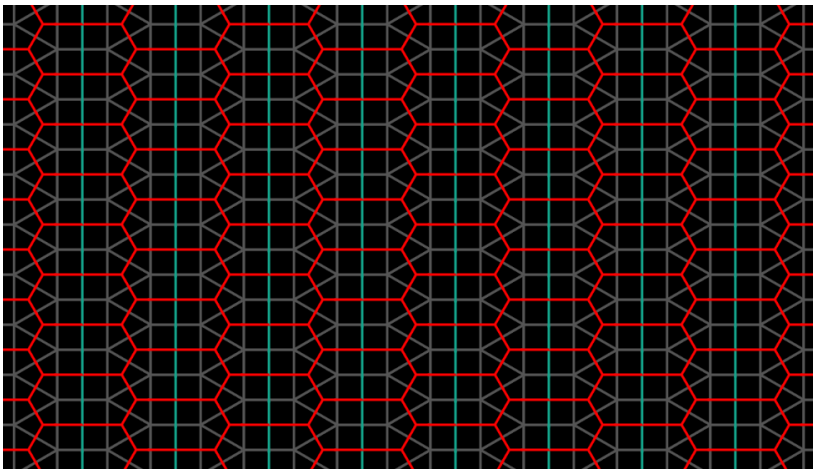


Figure 5.17: Triangle-square tiling where $\mathcal{L} = \emptyset$. The restricted dual ${\mathcal{T}_\mathcal{L}}^*$ is a set of parallel geodesics, with no vertices.

Each *restricted dual* tiling of $\mathcal{T}$ has the same translation lattice as $\mathcal{T}$ with vertex sets $\mathcal{E}$ and $\mathcal{L}$ respectively. If the area of the fundamental region of $\mathcal{T}$ is $a + b(\sqrt{3}/2)$, then $|\mathcal{E}| + |\mathcal{L}| = 2b$.

By the lattice containing the set of triangles, it can be deduced that the *restricted dual* tiling $\mathcal{T}_{\mathcal{E}}{}^*$ ($\mathcal{E} \neq \emptyset$) is formed by closed *equiangular* — but not equilateral in general — hexagons, with vertices in $\mathcal{E}$. The edges of $\mathcal{T}_{\mathcal{E}}{}^*$ make orthogonal intersections with the edges or the geodesics of $\mathcal{T}_{\mathcal{L}}{}^*$ inside the squares of $\mathcal{T}$, as illustrated in Figure 5.15 and 5.17. Also, $\mathcal{T}_{\mathcal{E}}{}^*$ is connected ($\mathcal{E} \neq \emptyset$), and the length of an edge is equal to $s + 1/\sqrt{3}$, where $s$ is the number of squares it crosses.

Everything we have stated so far is proved by construction. To prove the existence of the restricted dual tilings we look at a triangular face on the tiling, which is a vertex of the restricted dual, say $w \in \mathcal{E}$. Three dual edges are adjacent to it. Take one of them and start passing across the adjacent square faces of the tiling. This sequence must be finite, reaching a triangle again after a finite number of squares, otherwise it contradicts the existence of the translation lattice of the tiling $\mathcal{T}$. This is true for each of the edges, for each element in $\mathcal{E}$. When $\mathcal{E} = \emptyset$, there is no vertex to start with, neither where to arrive at, the stripe of square faces comes back to itself in the plane torus topology and closes; otherwise it contradicts the existence of the translation lattice of $\mathcal{T}$. The argument for $\mathcal{L}$ is analogous.

We define the $\mathcal{G}^*_{\xi}$, the *edge-labeled dual graph* of $\mathcal{T}$, as the abstract graph isomorphic to $\mathcal{T}_{\mathcal{E}}{}^*$, where $\xi$ is a vector of edge labels. Each label in $\xi$ is the number of squares crossed by the corresponding edge in $\mathcal{T}^*_{\mathcal{E}}$, as illustrated in Figure 5.18. The dual graph $\mathcal{G}^*$ is a 3-regular toroidal graph, which we draw as a regular hexagonal tiling over the plane torus, with vertices in $\Lambda(1, \omega^2)$.[2] The translation lattice for this tiling is different from the translation lattice for $\mathcal{T}$.

[2] We use the fact that the 2-torus with Euclidean metric $T^2 = \mathbb{R}^2/\Gamma$, where $\Gamma \cong \mathbb{Z}^2$ is a lattice in $\mathbb{R}^2$.

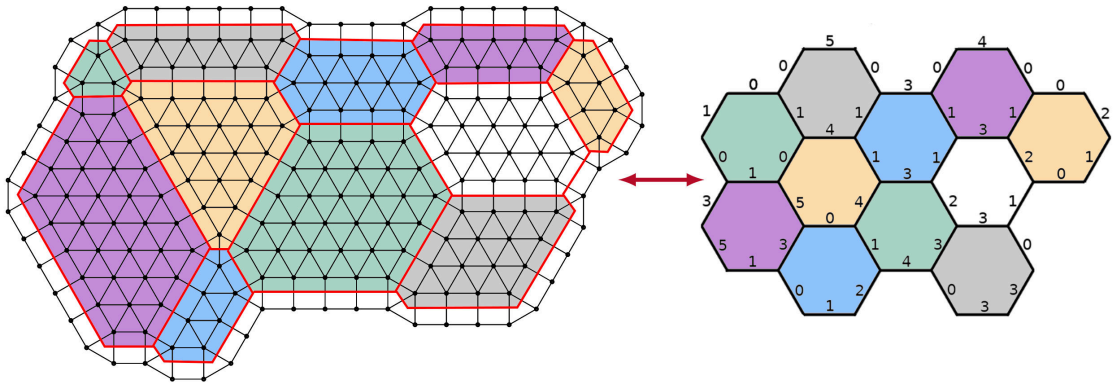


Figure 5.18: Part of a restricted dual over a region of a triangle-square tiling with its associated dual graph with labels in the edges. Labels on the edges of $\mathcal{G}^*_{\xi}$ correspond to the number of squares crossed by the edge on the restricted dual.

We use the equivalence of a triangle-square tiling $\mathcal{T}$ with its edge-labeled dual graph $\mathcal{G}^*_{\xi}$ for tiling generation. Finding a valid labeling $\xi$ for a hexagonal toroidal graph $\mathcal{G}^*$ is equivalent to finding a triangle-square tiling.

The relations between the restricted dual tilings, the edge-labeled graphs associated with them, and tiling generation is illustrated in Figure 5.6 and Figure 5.7. In the following sections we analyze the generation of hexagonal tilings on the plane torus, as well as the properties of a labeling that characterize a triangle-square tiling.

## 5.3 *Honeycombs on the plane torus*

To be able to create a triangle-square tiling from an edge-labeled dual graph, we need to understand the regular hexagonal tilings of the plane torus. We start by observing that its vertices belong to the lattice $\Lambda(1, \omega^2)$. Since we are dealing with a 2-dimensional lattice, the matrix with the translation lattice vectors has the following structure, after row reduction:

$$T = \begin{pmatrix} \alpha & \gamma \\ 0 & \beta \end{pmatrix} \begin{pmatrix} 1 \\ \omega^2 \end{pmatrix}$$

where $\alpha, \beta, \gamma \in \mathbb{Z}$, and, without loss of generality, $0 \leq \gamma < \beta$ and $\alpha, \beta > 0$. Considering the matrix in its reduced form ensures that we generate every possible translation lattice without repetition. If $\gamma = 0$ we require $\alpha \geq \beta$ to avoid the equivalent lattice by reflection across $\omega$.

Hexagonal tilings have two types of vertices: those with star $\{\omega^0, \omega^4, \omega^8\}$ and those with star $\{\omega^2, \omega^6, \omega^{10}\}$, colored red and blue in Figure 5.19, respectively. Then, any valid translation vector should take a vertex to another with the same star, which are located at relative positions: $3k_1\omega^0$, $3k_2\omega^2$, or $k_3(1 + \omega^2)$, with $k_i \in \mathbb{Z}_+$ for $i = 1, 2, 3$ (Figure 5.19). Thus, the translation matrix is given by:

$$T = \begin{pmatrix} 3k_1 + k_3 & k_3 \\ 0 & 3k_2 \end{pmatrix} \begin{pmatrix} 1 \\ \omega^2 \end{pmatrix}.$$

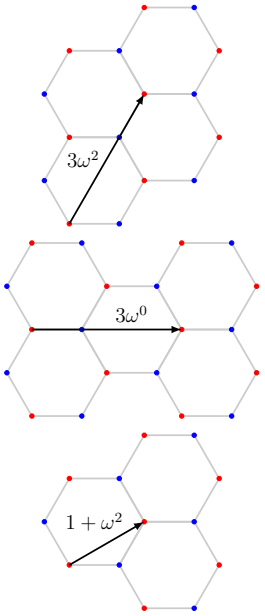


Figure 5.19: Possible translations in the hexagonal tiling.

This matrix form gives us a general way to compute the translation lattices (plane tori) that correspond to regular hexagonal tilings. The determinant of the matrix in lattice coordinates is $9k_1k_2 + 3k_2k_3$, the number of pairs of triangles contained in the fundamental region, since the determinant of the basis matrix is $\sqrt{3}/2$. Thus, the number of hexagonal faces is given by:

$$h = (3k_1 + k_3)k_2.$$

Using Euler's formula for a graph over the torus and the fact that the graph is 3-regular, we conclude that the tiling has exactly $2h$ vertices and $3h$ edges.

Without loss of generality, we assume that a vertex with star $\{\omega^0, \omega^4, \omega^8\}$ is at the origin, the centroid of the face indexed 1 is in $[0, 1]$ — in $\Lambda(1, \omega^2)$ coordinates — and the faces are indexed in lexicographical order by their lattice coordinates (see the examples in Figure 5.23).

To populate the fundamental region with face centroids and vertices, we check lattice coordinates $[\, i, j \,]$ for $0 \leq j \leq \beta + \gamma$ and $0 \leq i \leq \alpha + \gamma$ inside the fundamental region:

$i \equiv (j - 1) \bmod 3$ corresponds to a face centroid.

$i \equiv j \bmod 3$ corresponds to a vertex with star $\{\omega^0, \omega^4, \omega^8\}$.

$i \equiv (j + 1) \bmod 3$ corresponds to a vertex with star $\{\omega^2, \omega^6, \omega^{10}\}$.

Once we have the lattice coordinates of vertices and faces in the fundamental region, we can build the graph topology. The neighboring faces are located in the relative coordinates shown in Figure 5.20. The topology of the plane torus generates complex neighboring around each face, some faces are neighbors along more than one edge. This means that pairs of indices are not enough to define an edge, the neighboring position is important. To avoid repetition, the lower half edges of each face are computed and saved, the others can be deduced from the neighboring faces (Figure 5.21). This gives us the total of the $3h$ edges computed and stored only once. A face adjacency $h \times 6$ matrix $F$ is built, where each row holds the face's neighbors indices: $(n_1, \ldots, n_6)$ in the order of the edge enumeration indicated in Figure 5.21. This information, together with the centroids' lattice coordinates, is enough to reconstruct the full graph topology.

Given $h \in \mathbb{N}$, we are then able to compute all the unique translation lattices with $h$ hexagonal faces, the faces' lattice coordinates inside the fundamental region, and the graph topology over the plane torus. The number of different translation lattices is related to the number of factorizations of $h$ in the form $(3k_1 + k_3)k_2$. For example, for $h$ prime there are always two solutions:

$$T_{(1,h)} = \begin{pmatrix} 1 & 1 \\ 0 & 3h \end{pmatrix}, \text{ and } \quad T_{(2,h)} = \begin{pmatrix} h & h \bmod 3 \\ 0 & 3 \end{pmatrix}.$$

The number of translation lattices $\varphi(h)$ for $h \in \{1, \ldots, 500\}$ is shown in the Figure 5.22. The first 20 cases of the numbers in the plot are shown in Table 5.1.

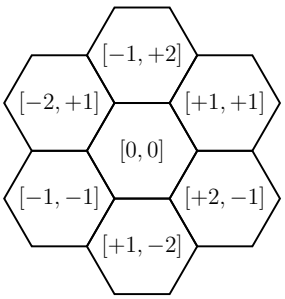



Figure 5.20: Neighboring rule for the hexagonal regular graph on reduced lattice coordinates.

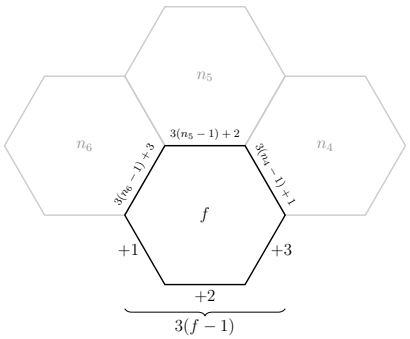



Figure 5.21: Edge indexing for a given face, all the edges can be deduced from the face $f$ and its neighbors' indices.

Each lattice corresponds to a different topology of the hexagonal graph, which, as we will see, is meaningful to the existence of valid edge labelings for the graph. Each configuration defines specific restrictions over the labeling.

| $h$ | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $\varphi(h)$ | 1 | 2 | 2 | 3 | 2 | 4 | 2 | 5 | 3 | 5 | 2 | 8 | 2 | 5 | 5 | 7 | 2 | 8 | 2 | 9 |

Table 5.1: Number of translation lattices $\varphi(h)$ for tilings with $h \in \{1, \ldots, 20\}$ hexagonal faces on the plane torus.

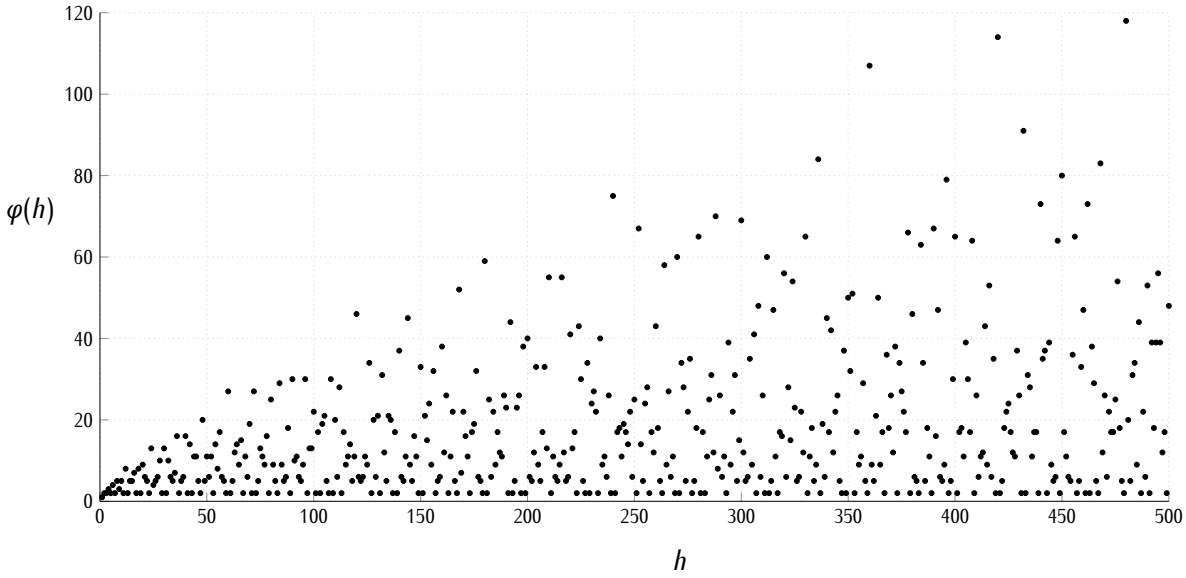



Figure 5.22: Number of translation lattices $\varphi(h)$ for tilings with $h \in \{1,\ldots,500\}$ hexagonal faces on the plane torus.

*Examples.* Let $h = 8$, $\varphi(8) = 5$. We show two of the five possible hexagonal tilings in Figure 5.23, the ones corresponding to $T_{(3,8)}$ and $T_{(4,8)}$, respectively. The translation lattices are:

$$T_{(1,8)} = \begin{pmatrix} 1 & 1 \\ 0 & 24 \end{pmatrix} \qquad T_{(2,8)} = \begin{pmatrix} 2 & 2 \\ 0 & 12 \end{pmatrix} \qquad T_{(3,8)} = \begin{pmatrix} 4 & 1 \\ 0 & 6 \end{pmatrix}$$

$$T_{(4,8)} = \begin{pmatrix} 4 & 4 \\ 0 & 6 \end{pmatrix} \qquad T_{(5,8)} = \begin{pmatrix} 8 & 2 \\ 0 & 3 \end{pmatrix}$$

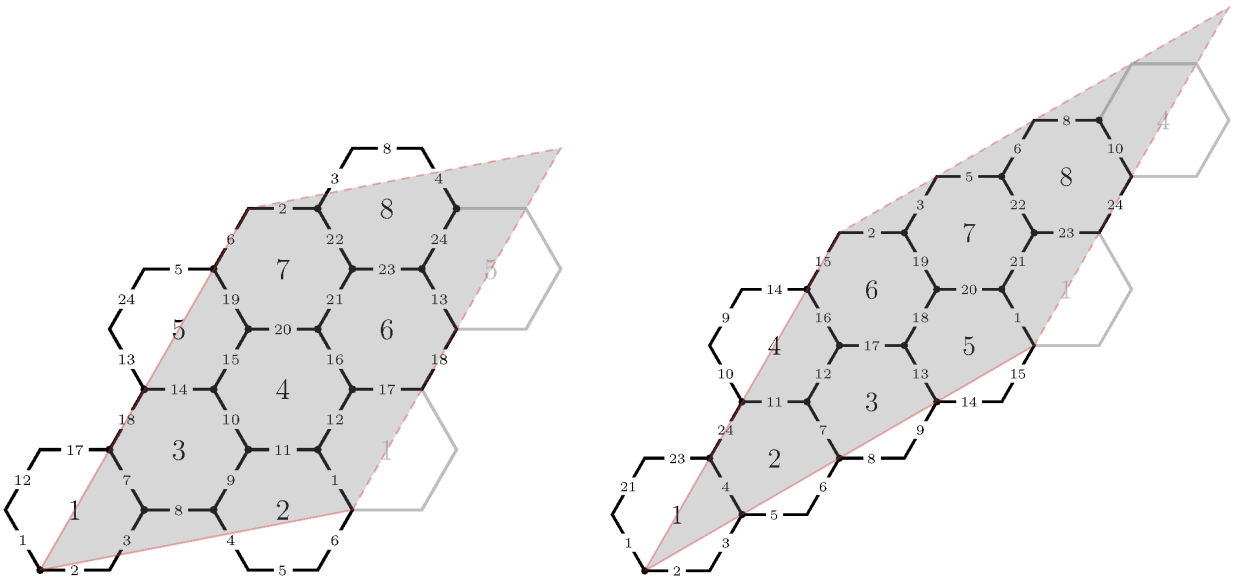


Figure 5.23: Hexagonal graphs with translation lattices $T_{(3,8)}$ and $T_{(4,8)}$, for $h = 8$. Face and edge indexes are shown.

## 5.4 *Edge-labeled graphs: basis for triangle-square tilings*

We are interested in labeling the edges of a hexagonal tiling on the plane torus in a way that the labeling corresponds to a triangle-square tiling dual graph $\mathcal{G}^*_\xi$. The values of the labels are not at all arbitrary; geometric constraints over them guarantee that the corresponding edge lengths form a proper restricted dual tiling ${\mathcal{T}_\mathcal{E}}^*$.

Let the labels of a hexagonal face in $\mathcal{G}^*_\xi$ be $a, b, c, d, e, f \geq 0$ as shown in Figure 5.24. In order to produce a closed hexagon in ${\mathcal{T}_\mathcal{E}}^*$, we require that the cycle closes after going around the hexagon, that is, using the basic directions:

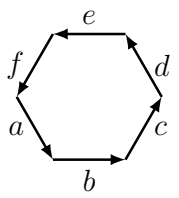



Figure 5.24: Labels on the edges of a hexagonal face must create a closed loop when translated to edge lengths.

$$a\omega^{10} + b\omega^0 + c\omega^2 + d\omega^4 + e\omega^6 + f\omega^8 = 0, \text{ this is}$$

$$\begin{pmatrix} 1 & 1 & 0 & -1 & -1 & 0 \\ -1 & 0 & 1 & 1 & 0 & -1 \end{pmatrix} \begin{pmatrix} a \\ b \\ c \\ d \\ e \\ f \end{pmatrix} = 0.$$

This homogeneous system of rank 2 is the geometric constraint for one hexagon.[3] Applied to the hexagonal graph topology, the columns of the matrix spread to the columns corresponding to the indexes of the edges around a face.

[3] Ball [8] analyzed all equiangular polygons and reached very similar conclusions for the hexagons. The basic directions offer a compact representation.

Let $\xi \in \mathbb{Z}^{3h}$ be the vector of edge labels for the hexagonal toroidal graph $\mathcal{G}^*_\xi$. Then, the geometric constraint that guarantees that this labeling corresponds to a restricted dual tiling ${\mathcal{T}_\mathcal{E}}^*$ is expressed as an integer sparse linear system of the form $G\xi = 0$, where, for the $k$-th face with edge indexes $E_k = \{e_1, e_2, \ldots, e_6\}$, the sub-matrix of $G$ is given by:[4]

$$G_{[\{2k-1,2k\}\times E_k]} = \begin{pmatrix} 1 & 1 & 0 & -1 & -1 & 0 \\ 0 & 1 & 1 & 0 & -1 & -1 \end{pmatrix}.$$

[4] We have expressed the second row as the sum of the rows of the original matrix, which gives a better numerical structure.

Then, $G$ is a sparse $2h \times 3h$ integer matrix, with only $8h$ non-zero entries. $G$ is rank-deficient, the dimension of its null-space is $(h+2)$.[5]

[5] Null-space dimension is an experimental result. The geometric intuition is that determining all the hexagons but one to fit the tiling is enough, hence the need for $2(h-1)$ equations to fully determine the system. This argument is equivalent to the circle packing conjecture by Fejes Tóth: *The triangle packing in the plane, minus one packing disc, is solid* [19].

Let $p = h + 2$ be the dimension of the null-space of $G$, and let $\{N_1, N_2, \ldots, N_p\}$ be a base for it: a set of $p$ linearly independent vectors in $\mathbb{Z}^{3h}$, such that $GN_i = 0$, for $i = 1, \ldots, p$. Then, to have a valid labeling $\xi$, we need $\eta = (\eta_1, \ldots, \eta_p)^T \in \mathbb{Z}^p$, such that:

$$\xi = N\eta = \sum_{i=1}^{p} \eta_i N_i \geq 0,$$

since edge labels must be at least 0 to have a meaning. The geometric constraint is satisfied, given that:

$$G\xi = G\left(\sum_{i=1}^{p} \eta_i N_i\right) = \sum_{i=1}^{p} \eta_i G N_i = 0,$$

where $N = (N_i)_{i=1}^{p}$ is a $3h \times p$ integer matrix.

Null-space $N$ can be computed applying a stable sparse matrix decomposition algorithm over $G$. Methods based on LU decomposition have proved efficient for this task over rectangular sparse matrices [32]. Using Kowal's LUQ decomposition [50], we obtain a $3h \times p$ matrix $N$ whose columns are a basis for $\text{null}(G)$.[6]

[6] The method is not designed for integer matrices, it introduces some floating point fractional entries and rounding errors. The result obtained is *cleaned* by rounding the floating point residuals close to zero, and by multiplying the columns where we identify rational values. An adapted implementation of a method of this nature for integer matrices is desirable in the near future to extend the results.

Experimentally, matrix $N$ has the following structure:

$$N = P^{-1} \begin{pmatrix} C \\ D \end{pmatrix},$$

where $P$ is a permutation matrix, $C$ is a dense block, and $D$ is a non-singular, sparse block with positive entries of size $p$. Since $\xi \in \text{null}(G)$, we know that $\xi = N\eta$, then:

$$\xi = P^{-1} \begin{pmatrix} C\eta \\ D\eta \end{pmatrix}.$$

Thus, since $D$ is very sparse and has positive entries, it imposes the constraint $\eta \geq 0$.

Let $\Xi = \{\xi \in \mathbb{Z}^{3h} \mid G\xi = 0,\ \xi \geq 0\}$ be the set of all valid labelings over a graph $\mathcal{G}^*_\xi$. Notice that every conical combination of elements $\xi^j \in \Xi$ is also a valid labeling, since

$$G\left(\sum_j a_j \xi^j\right) = \sum_j a_j G\xi^j = 0, \text{ and}$$
$$\sum_j a_j \xi^j \geq 0,$$

for arbitrary $a_j \in \mathbb{Z}_+$.

Summarizing, a valid labeling $\xi$ belongs to the cone generated by the columns of $N$,

$$C(N) = \{N\eta \mid \eta \in \mathbb{R}^p_+\},$$

intersected with the lattice $\mathbb{Z}^{3h}$, such that $\xi \geq 0$. Since $\xi = 0$ is a valid labeling itself, $C(N)$ is a *pointed* cone, and the set of all valid labelings

$$\Xi = C(N) \cap \{\xi \in \mathbb{Z}^{3h} \mid \xi \geq 0\}$$

is an *affine monoid*, by Gordon's lemma [15, 38].

By Caratheodory's theorem and van der Corput's theorem, there is a finite set of generators $X$ such that $\Xi = \text{cone}(X)$. $X$ is called a *Hilbert base* [15, 38], and, since the null-space of $G$ has dimension $p$, we know that $|X| = m \geq p$.

Counting Hilbert basis of a homogeneous system of Diophantine equations is #**P**-hard and belongs to class #**NP** in general [38, 39]. However, algorithms for finding Hilbert basis of affine monoids exist, like the *project-and-lift* algorithm [37] and *Normaliz* [14, 15], the one that we use.

*Bases for triangle-square tilings*

The existence of a Hilbert basis for the cone of valid labelings for each family of tilings provides a natural enumeration, since the basis can be lexicographically fully ordered. This result provides the full characterization of all triangle-square tilings.

We showed how to generate the dual hexagonal topology $F_{(i,h)}$ over the translation lattice given by matrix $T_{(i,h)}$, and how to find generators $X_{(i,h)}$ (the Hilbert basis) of the set of valid labelings, for each $i = 1, \ldots, \varphi(h)$, for each $h \in \mathbb{N}$. These elements generate an infinite family of tilings $\mathbb{T}_{(i,h)}$, and they need to be computed only once and stored. For each vector of non-negative integers of $m$ elements, $a \in \mathbb{Z}_+^m$, we obtain a valid labeling $\xi = X_{(i,h)}a$ to generate a triangle-square tiling. This means that the family of tilings in a flat torus given by $T_{(i,h)}$, with graph topology given by $F_{(i,h)}$, is

$$\mathbb{T}_{(i,h)} = \text{span}\left(X_{(i,h)}\right),$$

a tiling space isomorphic to $\mathbb{Z}_+^m$, an affine monoid of tilings, with $m$ generators, for each $i = 1, \ldots, \varphi(h)$, for each $h \in \mathbb{N}$. In the following section, we explain in detail how to generate a tiling with these elements.

Using *Normaliz* [14] we computed the Hilbert basis for each family of tilings $\mathbb{T}_{(i,h)}$, with topology $i = 1, \ldots, \varphi(h)$, for each $h = 1, \ldots, 17$. Table 5.2 shows the values of $m$: the number of generators in the Hilbert basis for each family, which corresponds to the dimension of the space generated by them. This means $\mathbb{T}_{(i,h)}$ is isomorphic to $\mathbb{Z}_+^m$ by the values in the table.

Number $m$ of elements in the Hilbert basis of $\mathbb{T}_{(i,h)}$

| $h$ | $i \in \{1, \ldots, \varphi(h)\}$ | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 1 | 3 | | | | | | | |
| 2 | 4 | 4 | | | | | | |
| 3 | 5 | 6 | | | | | | |
| 4 | 6 | 6 | 8 | | | | | |
| 5 | 7 | 13 | | | | | | |
| 6 | 8 | 12 | 19 | 19 | | | | |
| 7 | 9 | 31 | | | | | | |
| 8 | 10 | 12 | 28 | 12 | 48 | | | |
| 9 | 11 | 12 | 78 | | | | | |
| 10 | 12 | 58 | 64 | 64 | 124 | | | |
| 11 | 13 | 201 | | | | | | |
| 12 | 14 | 34 | 171 | 238 | 238 | 39 | 171 | 323 |
| 13 | 15 | 523 | | | | | | |
| 14 | 16 | 318 | 648 | 648 | 844 | | | |
| 15 | 17 | 1225 | 940 | 2291 | 1366 | | | |
| 16 | 18 | 184 | 1588 | 56 | 288 | 1072 | 2208 | |
| 17 | 19 | 3573 | | | | | | |

Table 5.2: Number $m$ of elements in the Hilbert basis for each tiling family $\mathbb{T}_{(i,h)}$.

*Examples.* Let $h = 8$, we consider the first topology shown in Figure 5.23, corresponding to $T_{(3,8)}$. The translation lattice is given by:

$$T_{(3,8)} = \begin{pmatrix} 4 & 1 \\ 0 & 6 \end{pmatrix}.$$

The Hilbert basis $X_{(3,8)}$ has 28 elements. By a heuristic method applied to find some generators in the Hilbert basis[7], we obtained $\widehat{X}_{(3,8)}$, a subset of 11 generators. This is a $24 \times 11$ matrix of minimal valid labelings

[7] This method was developed before the use of *Normaliz* [14]. Every element found with it was a proper element of the Hilbert basis, as we verified before, using the specialized software.

$$\widehat{X}_{(3,8)} = \begin{pmatrix}
1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\
0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\
0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\
1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 \\
0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0
\end{pmatrix}$$

Then, we take the positive linear combination:

$$\begin{aligned} \xi &= \widehat{X}_{(3,8)}(2,2,2,2,2,1,1,0,2,0,0)^T \\ &= (4,3,3,2,4,2,3,2,4,3,2,2,4,2,3,4,4,4,2,1,4,0,4,1)^T \end{aligned}$$

which generates the valid labeling in Figure 5.25, each value of the labeling is shown on its corresponding edge, the indexes can be consulted in Figure 5.23. The resulting triangle-square tiling is shown in Figure 5.39.

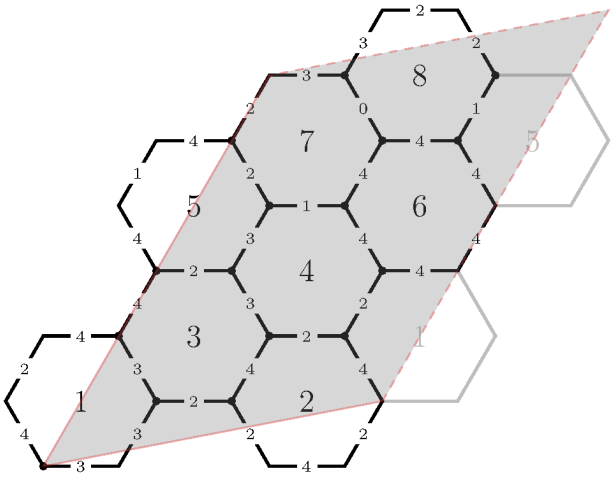



Figure 5.25: Translation lattice $T_{(3,8)}$. A valid edge labeling is shown.

## 5.5 *Dual-graph tiling generation*

Let $h \in \mathbb{N}$ be the number of faces on a hexagonal toroidal graph $\mathcal{G}^*_\xi$ with a valid labeling $\xi$ on its edges. To build a triangle-square tiling we make a graph traversal through $\mathcal{G}^*_\xi$, inserting the vertices corresponding to the squares on $\mathcal{T}_\mathcal{E}{}^*$ edges. By keeping track of the visited vertices in lattice coordinates for $\mathcal{G}^*_\xi$, we compute the translation lattice for $\mathcal{T}_\mathcal{E}{}^*$ and $\mathcal{T}$. At the end, we insert the vertices of the triangulated hexagonal faces of $\mathcal{T}_\mathcal{E}{}^*$. We shall explain the details of this *dual-graph tiling generation* algorithm.

Let $v \in \mathbb{Z}[\omega]$ be the writing head that inserts vertices on the generated tiling $\mathcal{T}$, and $u \in \Lambda(1, \omega^2)$ be a pointer to the location on the graph traverse over $\mathcal{G}^*_\xi$. We build a queue of $(j, v, u)$ data packages with the following information:

$j$ is the edge index, signed. The sign indicates the direction the edge is traversed: $j > 0$ indicates going from left to right, and $j < 0$ the opposite.

$v$ is the position of the writing head in lattice coordinates in $\mathbb{Z}[\omega]$.

$u$ is the pointer location on the graph in lattice coordinates $\Lambda(1, \omega^2)$.

The data structure representing the hexagonal dual graph $\mathcal{G}^*_\xi$ has the following information:

$F$ is the face adjacency $h \times 6$ matrix described in Section 5.3. The canonical labeling of our edges makes it simple to deduce the neighboring edges and faces by their indices.

$\xi$ is a valid labeling for the edges of $\mathcal{G}^*_\xi$, $0 \leq \xi \in \mathbb{Z}^{3h}$.

Let $V \subset \mathbb{Z}[\omega]$ be the set of vertices of the generated tiling. Let $J_{\text{out}} = \{1, \ldots, 3h\}$ be the set of non-visited edges. The convention for the position of the writing head is shown in Figure 5.26, vertices in positions $v + c\omega^k$ and $v + c\omega^k + \omega^{k-3}$ are inserted in $V$ for each $c = 0, 1, \ldots, \xi_j$ while traversing the $j$-th edge of $\mathcal{G}^*_\xi$. Once this is done,

index $j$ is removed from $J_{\text{out}}$, and the two neighboring edges are inserted in the queue with the following updated values:

*Left:* $(j_L, v + \xi_j \omega^k, u + \overline{\omega}^k)$

*Right:* $(j_R, v + \xi_j \omega^k + \omega^{k-1}, u + \overline{\omega}^k)$

Notice that in the left turn the writing head does not change position, while in the right turn we add $\omega^{k-1}$ to get the correct position.

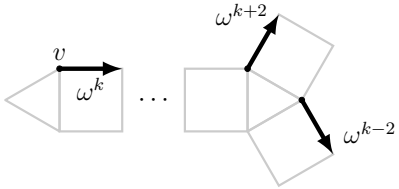



Figure 5.26: Graph traversal writing head convention.

The value of $k$ in $\omega^k$ can be deduced from the sign and the remainder modulo 3 of the edge index $j$, since it indicates its position with reference to the face, as indicated in Figure 5.21 and Figure 5.28,

$$k = \begin{cases} 10 & \text{if } j > 0 \;\&\; j \bmod 3 = 1, \\ 4 & \text{if } j < 0 \;\&\; j \bmod 3 = 1, \\ 0 & \text{if } j > 0 \;\&\; j \bmod 3 = 2, \\ 6 & \text{if } j < 0 \;\&\; j \bmod 3 = 2, \\ 2 & \text{if } j > 0 \;\&\; j \bmod 3 = 0, \\ 8 & \text{if } j < 0 \;\&\; j \bmod 3 = 0. \end{cases}$$

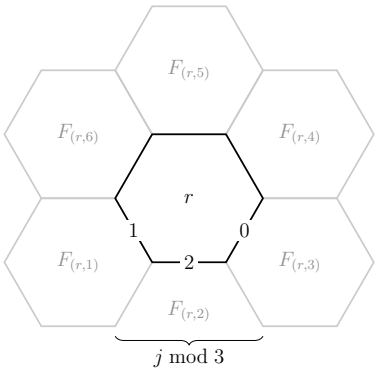



Figure 5.27: Positions of neighboring faces given by matrix $F$ and orientations of edges given by $j$ mod 3.

The neighboring edges $j_L$ and $j_R$ can also be deduced from $j$. The face to which edge $j$ corresponds is given by $r = \lfloor (j-1)/3 \rfloor + 1$. The indices of the neighboring edges are given by the face adjacency matrix $F$ in the following way (Figure 5.27 and Figure 5.28):

$$j_L = \begin{cases} 3(r-1)+2 & \text{if } j > 0 \;\&\; j \bmod 3 = 1, \\ -(3(F_{(r,6)}-1)+2)4 & \text{if } j < 0 \;\&\; j \bmod 3 = 1, \\ 3(r-1)+3 & \text{if } j > 0 \;\&\; j \bmod 3 = 2, \\ -(3(F_{(r,1)}-1)+3) & \text{if } j < 0 \;\&\; j \bmod 3 = 2, \\ -(3(F_{(r,4)}-1)+1) & \text{if } j > 0 \;\&\; j \bmod 3 = 0, \\ 3(F_{(r,3)}-1)+1 & \text{if } j < 0 \;\&\; j \bmod 3 = 0 \end{cases}$$

$$j_R = \begin{cases} -(3(F_{(r,1)}-1)+3) & \text{if } j > 0 \;\&\; j \bmod 3 = 1, \\ 3(F_{(r,6)}-1)+3 & \text{if } j < 0 \;\&\; j \bmod 3 = 1, \\ 3(F_{(r,3)}-1)+1 & \text{if } j > 0 \;\&\; j \bmod 3 = 2, \\ -(3(r-1)+1) & \text{if } j < 0 \;\&\; j \bmod 3 = 2, \\ 3(F_{(r,4)}-1)+2 & \text{if } j > 0 \;\&\; j \bmod 3 = 0, \\ -(3(r-1)+2) & \text{if } j < 0 \;\&\; j \bmod 3 = 0 \end{cases}$$

$3(F(r,i)-1)+e$ is the $e$-th edge of the $i$-th neighbor face of $r$. With $e \in \{1,2,3\}$, as shown in Figure 5.21, and $i \in \{1,\ldots,6\}$, as shown in Figure 5.27 and Figure 5.28.

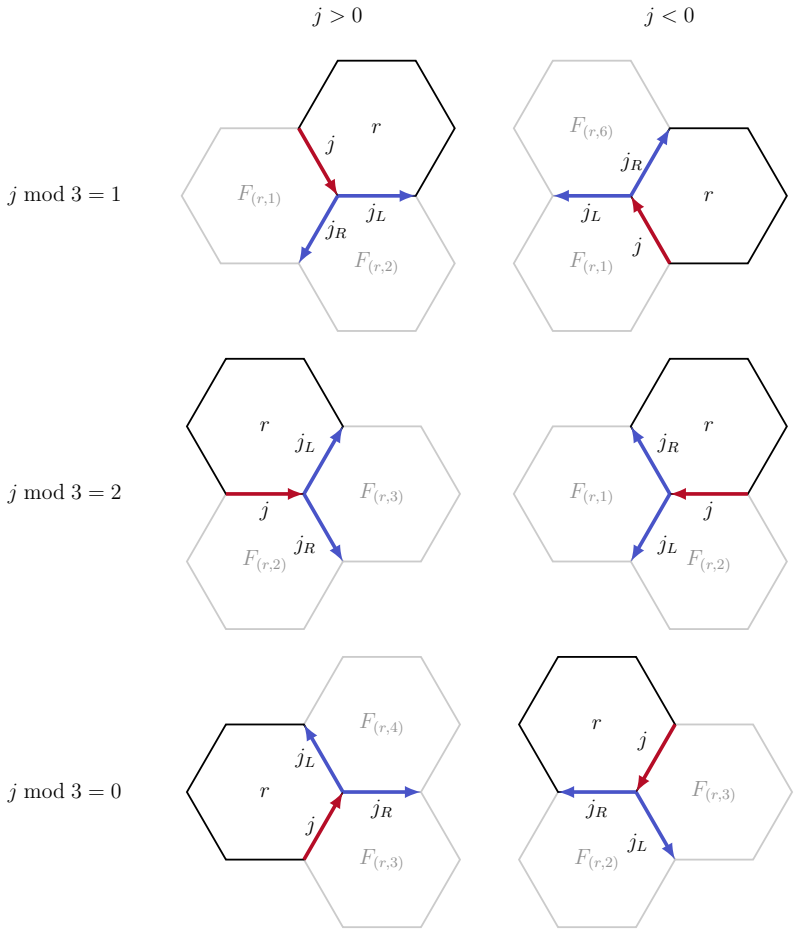



Figure 5.28: Positions of neighboring faces given by matrix $F$, indices of edges $j_L$, $j_R$, and orientations of edges given by $j$ mod 3 and its sign.

Proper graph traversal can be implemented in many different ways depending on the topology representation chosen for the dual graph $\mathcal{G}^*_\xi$. What matters is to traverse all the edges in the graph. To ensure that the graph traversal visits the edges only once, one hashes the visited edges in lattice coordinates. Finally, to avoid it growing in all directions, we only admit the region $u \geq 0$ in $\Lambda(1, \omega^2)$ coordinates, which contains the translation lattice of $\mathcal{G}^*_\xi$.

The process has two stopping criteria, the first one is that all the edges of the graph have been traversed. This guarantees that all the vertices corresponding to squares and triangles in set $\mathcal{E}$ have been inserted in $V$. The second criterion is that both translation vectors have been found. Let $T$ be the translation lattice matrix for $\mathcal{G}^*_\xi$. Each time $u = T_1$ or $u = T_2$ in lattice coordinates, the corresponding position of the writing head $v$ is stored as the corresponding translation vector in $T_V$.

The insertion of the vertices corresponding to the triangles in set $\mathcal{L}$ is achieved by a second pass on the tiling, by analyzing the stars of the vertices that have been already inserted. We create a hashed cloud $C_V$ with a $3 \times 3$ copy of $V$ on the translation lattice. Again, we form a queue, this time with the vertices in $V$. We take each vertex in the queue and compute its star over $C_V$. If it has two edges which

power of $\omega$ have a difference greater than 3, we insert vertices in $V$ every $\omega^2$ until the gap is filled, we update $C_V$. The star of each vertex is computed on the fly to guarantee it takes into account all inserted vertices, even the new ones. When the queue is empty, $V$ is a set of vertices corresponding to a triangle-square tiling.

Then, the full *dual-graph tiling generation* algorithm consists of two parts:

1. The dual graph traversal, that inserts the vertices corresponding to squares and triangles in $\mathcal{E}$: Algorithm 7. The details for the computation of $k$, $j_L$ and $j_R$ are omitted on behalf of clarity.

2. The insertion of vertices corresponding to the triangles in set $\mathcal{L}$, the ones inside the hexagonal faces of $\mathcal{T}_{\mathcal{E}}{}^*$: Algorithm 8.

The tiling formed by the set of vertices $V$ and the translation lattice $T_V$ might have vertices outside the fundamental region and even some duplicates. We use the normalization process described in Chapter 2 to obtain a proper tiling symbol.

**Algorithm 7**

**procedure** *Dual_traversal*$(\mathcal{G}^*_\xi)$
  $V \leftarrow \{[0,0,0,0]\}$
  $T_V \leftarrow \{\}$
  $J_{\text{out}} \leftarrow \{1,\ldots,3h\}$
  $Q \leftarrow \{(2,[0,1,0,0],[0,0])\}$
  **while** $J_{\text{out}} \neq \emptyset$ **and** $|T_V| < 2$ **do**
    $(j,v,u) \leftarrow Q.\text{pop}()$
    **if** $j \in J_{\text{out}}$ **and** $u \geq 0$ **then**
      **for** $m = 0$ **to** $\xi_j$ **do**
        $v \leftarrow v + \omega^k$
        $V \leftarrow v$
        $V \leftarrow v + \omega^{k-3}$
      **end**
      $J_{\text{out}} \leftarrow J_{\text{out}} \setminus \{j\}$
    **end**
    $Q \leftarrow (j_L, v, u + \overline{\omega}^k)$
    $Q \leftarrow (j_R, v + \omega^{k-1}, u + \overline{\omega}^k)$
    **if** $u = T_i$ **then**
      $T_V \leftarrow v$
    **end**
  **end**
  **return** $V, T_V$
**end**

**Algorithm 8**

**procedure** *Triangle_dense*$(V, T_V)$
  $C_V \leftarrow$ *Hashed_cloud*$(V, T_V)$
  $Q \leftarrow V$
  **while** $Q \neq \emptyset$ **do**
    $v \leftarrow Q.\text{pop}()$
    **for** $k = 0$ **to** 11 **do**
      **if** $v + \omega^k \in C_V$ **then**
        **if** $v + \omega^{k+3} \notin C_V$ **and**
          $v + \omega^{k+2} \notin C_V$ **then**
          $V \leftarrow v + \omega^{k+2}$
          $C_V \leftarrow v + \omega^{k+2}$
          $Q \leftarrow v + \omega^{k+2}$
        **end**
      **end**
    **end**
  **end**
  **return** $V, T_V$
**end**

*Examples*

We present some triangle-square tilings generated from their hexagonal edge labeled dual graphs on the torus. These concrete cases of the representation, with their corresponding images, allow us to fully grasp the generative potential of the strategy we have shown.

1. Let $h = 1$, the only realizable translation lattice is:

$$T_{(1,1)} = \begin{pmatrix} 1 & 1 \\ 0 & 3 \end{pmatrix}$$

Then $\mathcal{G}^*_\xi$ has one face, 2 vertices and 3 edges, as shown in Figure 5.29.

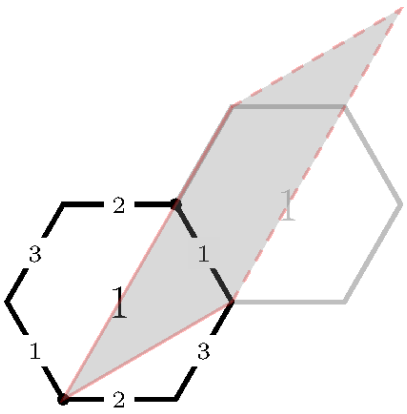



Figure 5.29: $\mathcal{G}$ regular graph for $h = 1$, edges showing their index.

Let $\xi \in \mathbb{Z}^3$ be the edge labels. In this particular case, because of the self adjacency of the only face, matrix $G$ is equal to zero in every entry, meaning $N$, its null-space, is the $3 \times 3$ identity. Thus, $\xi$ is any combination of non-negative integers.

$\xi = (0,0,0)^T$ generates the $W$ Archimedean tiling.

$\xi = (0,1,0)^T$ spans the $T$ Archimedean tiling.

$\xi = (1,1,0)^T$ results in the $U$ Archimedean tiling.

$\xi = (1,1,1)^T$ is the $P$ Archimedean tiling, removing – actually, not inserting – the vertices in the interior of the hexagonal faces.

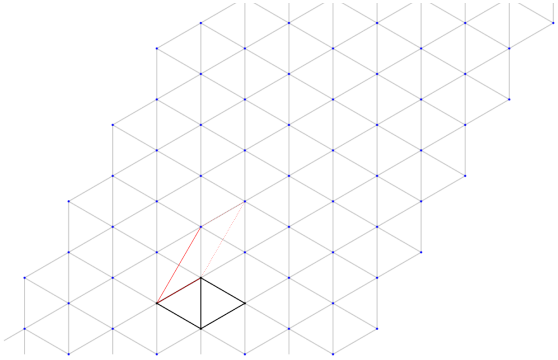


Figure 5.30: Triangle-square tiling generated by $\mathcal{G}^*_\xi$, with $\xi = (0,0,0)^T$.

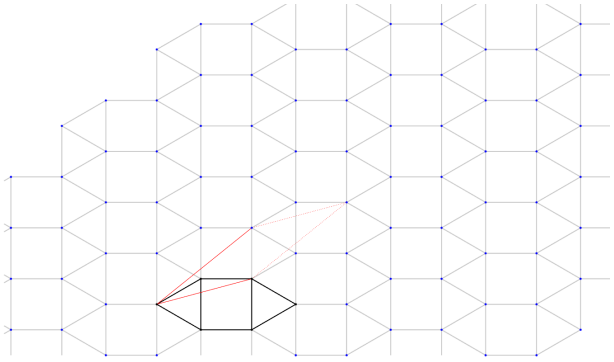


Figure 5.31: Triangle-square tiling generated by $\mathcal{G}^*_\xi$, with $\xi = (0,1,0)^T$.

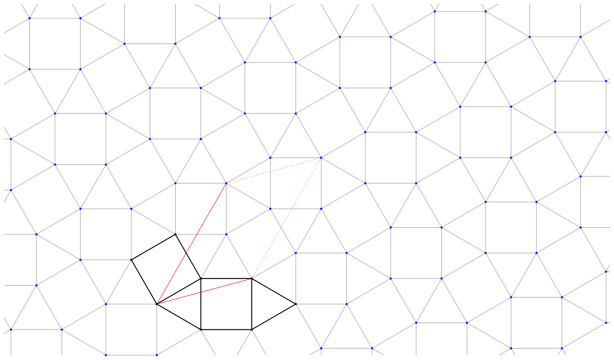


Figure 5.32: Triangle-square tiling generated by $\mathcal{G}^*_\xi$, with $\xi = (1,1,0)^T$.

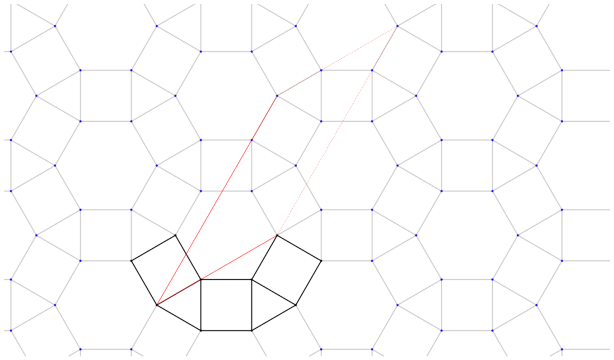


Figure 5.33: Triangle-square tiling generated by $\mathcal{G}^*_\xi$, with $\xi = (1,1,1)^T$.

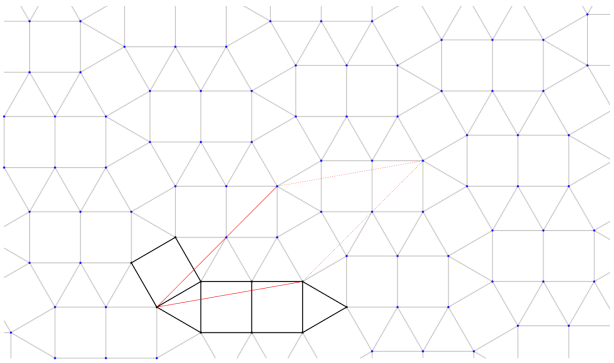


Figure 5.34: Triangle-square tiling generated by $\mathcal{G}^*_\xi$, with $\xi = (1,2,0)^T$.

The simplest topology of the dual graph ($h = 1$) and small values of the labels spans the Archimedean triangle-square tilings. Tilings get more *interesting* as $h$ and the variability of the edge labels increase.

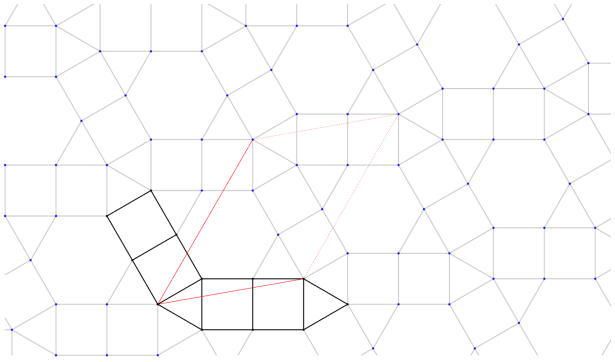


Figure 5.35: Triangle-square tiling generated by $\mathcal{G}^*_\xi$, with $\xi = (2,2,0)^T$.

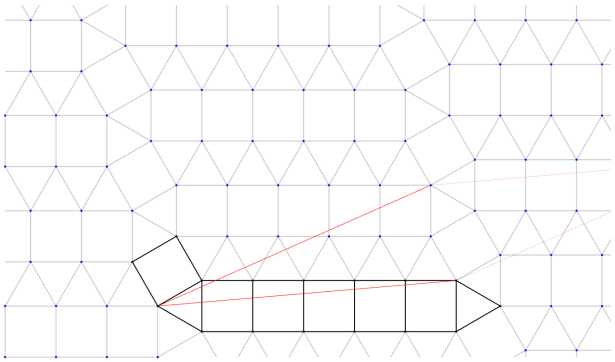


Figure 5.36: Triangle-square tiling generated by $\mathcal{G}^*_\xi$, with $\xi = (1,5,0)^T$.

One of the things we can observe from these first examples is that symmetry from the tilings shows itself in the $\mathcal{G}^*_\xi$ edge labeled dual graph. For example, for $h = 1$, tilings generated with labels $\xi_1 = (1,0,0)^T$, $\xi_2 = (0,1,0)^T$, and $\xi_3 = (0,0,1)^T$ are rotations of the same tiling, and this is obvious as the labels can be permuted over an equivalent topology. Moreover, the rotation symmetry present in tiling generated by $\xi_1 = (1,1,1)^T$ can be detected directly in the labeled dual graph. In this particular case, analyzing symmetry in the dual graph does not make much of a difference, but as the tilings get more complex, the computation cost of making a symmetry search over the dual graph is significantly smaller.

2. Consider the constant labeling $\xi = (0,0,\ldots,0)^T$, no matter what the value of $h$ is, we obtain the $W$ Archimedean tiling repeated $h$ times. This is useful for applying vertex removal, every lattice generates different tilings containing only triangles and hexagons.

The tiling on Figure 5.37 is obtained taking $\xi = (0,0,\ldots,0)^T$ over the following translation lattice for $h = 8$:

$$T_{(3,8)} = \begin{pmatrix} 4 & 1 \\ 0 & 6 \end{pmatrix}$$

The tiling on Figure 5.38 is obtained taking $\xi = (1,1,\ldots,1)^T$ over the same translation lattice, $T_{(3,8)}$.

These examples indicate us that some results loose complexity when the labels show redundant patterns over the dual graph topology. This means, that automatic generated tilings using the edge

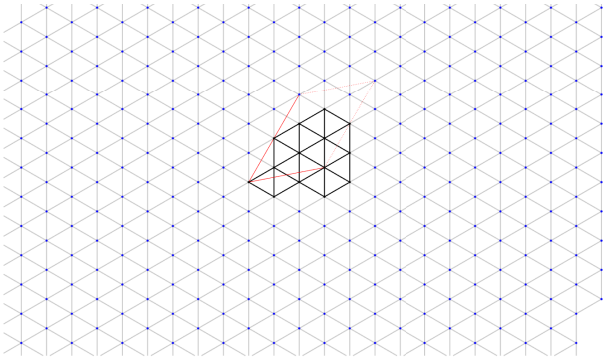


Figure 5.37: Triangle-square tiling generated by $\mathcal{G}^*_\xi$, with $\xi = (0, 0, \ldots, 0)^T$.

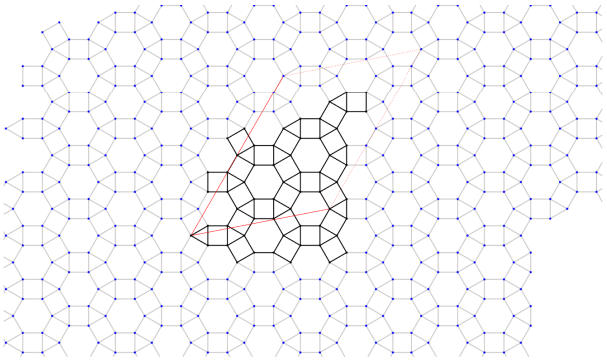


Figure 5.38: Triangle-square tiling generated by $\mathcal{G}^*_\xi$, with $\xi = (1, 1, \ldots, 1)^T$.

labeling of the regular graphs must be tested for minimal translation lattices in a similar way to acquired tilings in Chapter 3.

3. Finally, we follow up the labeling generated in the example of the last section, with $h = 8$ and translation lattice $T_{(3,8)}$. The tiling in Figure 5.39 is generated by the labeling shown in Figure 5.25.

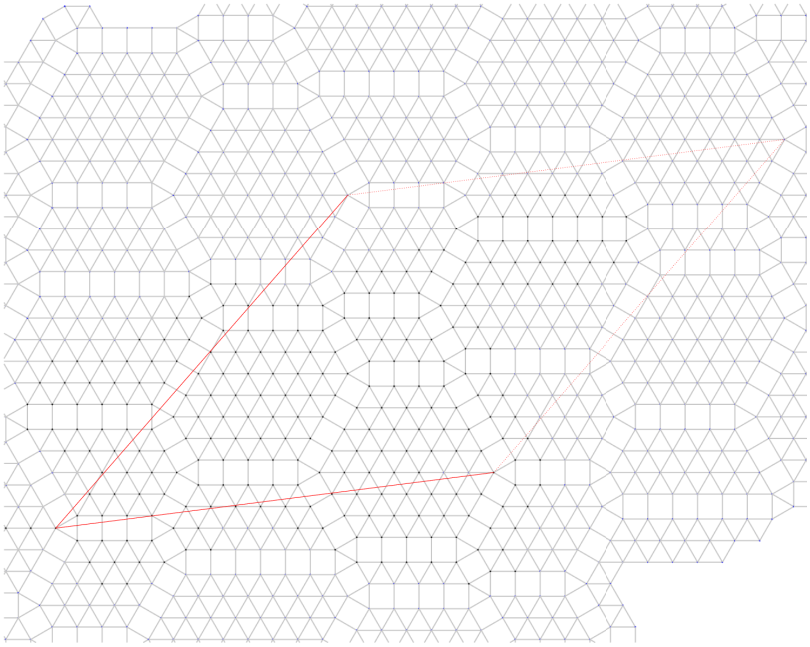


Figure 5.39: Triangle-square tiling generated by $\mathcal{G}^*_\xi$, with $X_{(3,8)}$ and hexagonal graph shown in Figure 5.25.

It is interesting to see the basic labelings that generated the tiling above, the subset of the Hilbert basis $\widehat{X}_{(3,8)}$, for $h = 8$. The 11 elements considered in the subset are shown in Figure 5.40. We discuss the geometric interpretation of this basic elements in Section 5.9.

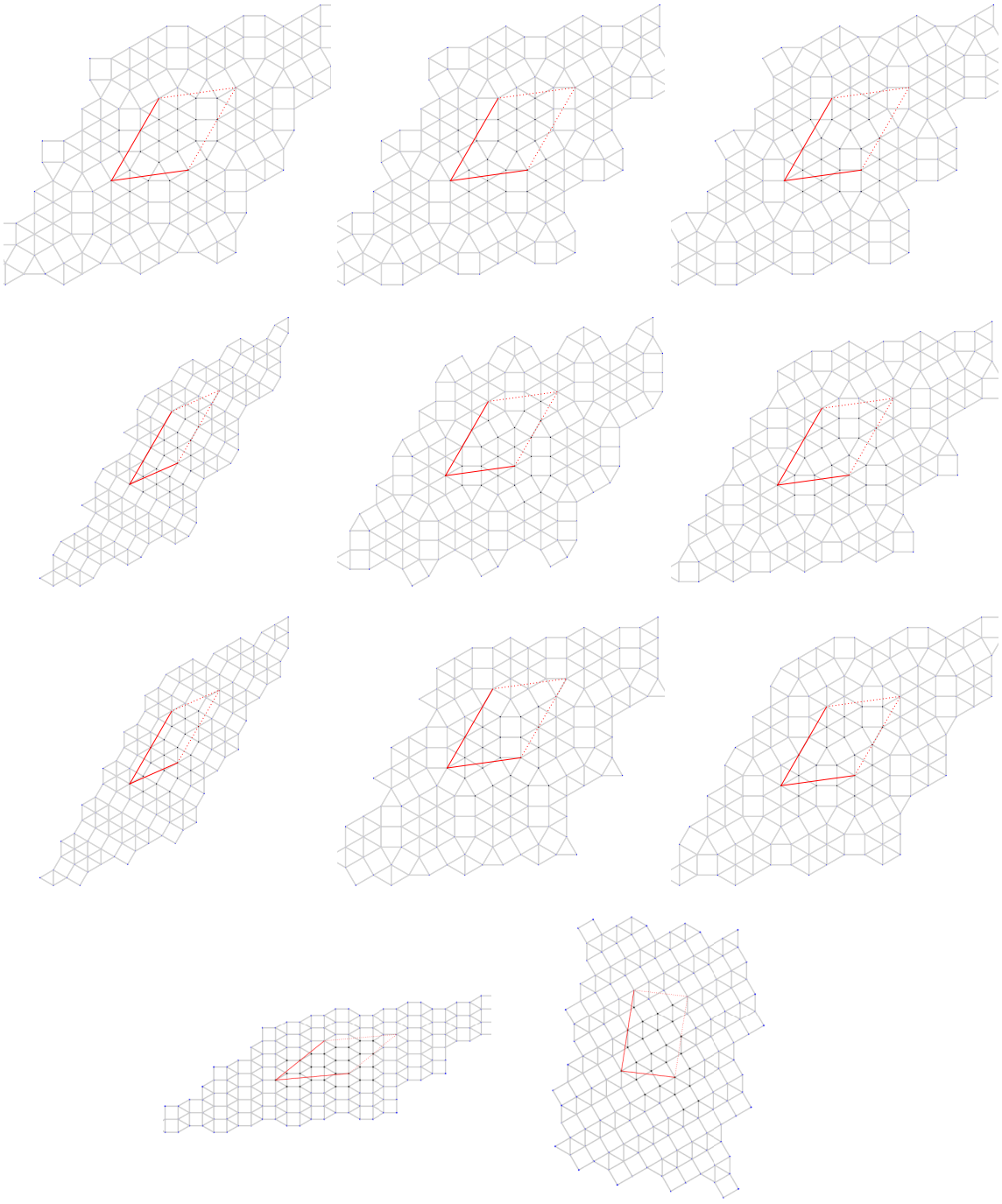


Figure 5.40: Triangle-square tilings for $h = 8$, generated by $\mathcal{G}^*_\xi$, with the basic labelings $\xi \in \widehat{X}_{(3,8)}$.

## 5.6 *Area of labeling generated tilings*

Let $\mathcal{T}$ be a triangle-square tiling and $\mathcal{G}^*_\xi$ its edge labeled dual graph with labels $\xi \in \mathbb{Z}^{3h}$. It is possible to obtain algebraic expressions relating the labeling on the graph $\mathcal{G}^*_\xi$ with the number of squares ($a$) and triangles ($2b$) in the fundamental region of $\mathcal{T}$. Understanding the relation of the labeling with the area reveals some implications of the complementary property between the restricted dual tilings and the labeled dual graphs.

Take a hexagonal face of $\mathcal{G}^*_\xi$ with edge labels $a, b, c, d, e, f$. Remember that edge lengths in ${\mathcal{T}_\mathcal{E}}^*$ are given by the labels, as shown in Figure 5.41. In the tiling $\mathcal{T}$, this corresponds to a triangulated region inside a hexagonal face of side lengths given by the labels, as one can see in Figure 5.18. Since the area of an equilateral triangle of side $k$ is equal to $(\sqrt{3}/4)k^2$, we can compute the area of the hexagon by subtracting the triangles of sides $a$, $c$ and $e$ from the triangle of side $(a+b+c)$. Obtaining the following quadratic expression for the area:

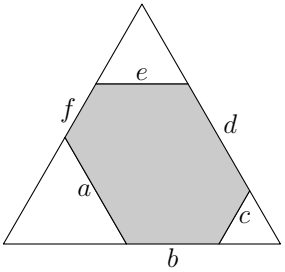



Figure 5.41: The area of an equiangular, not equilateral hexagon can be computed using the triangle containing it.

$$
\begin{aligned}
\text{Area}_H &= \sqrt{3}/4 \left( (a+b+c)^2 - a^2 - c^2 - e^2 \right) \\
&= \sqrt{3}/4 \left( b^2 + 2(ab+ac+bc) - e^2 \right) \\
&= \sqrt{3}/4 \begin{pmatrix} a & b & c & d & e & f \end{pmatrix} \begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} a \\ b \\ c \\ d \\ e \\ f \end{pmatrix} \\
&= \sqrt{3}/4 \left( x^T D_1 x \right)
\end{aligned}
$$

Where $x = (a, b, c, d, e, f)^T$, and $D_1$ is the $6 \times 6$ matrix.[8]

When we sum the areas of all the faces, the quadratic terms cancel out, since the edge labeled as $e$ in the upper neighbor corresponds to a $b$ edge in the current face area equation. In our choice of enumeration of the edges, explained in Section 5.3, we enumerated the lower half edges of each face with sequential indexes (Figure 5.21). This way, the matrix for the area of all the triangulated hexagonal faces of the tiling generated by the labeling has diagonal $3 \times 3$ blocks of the form:

$$
\Delta_k = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix}
$$

Which is a circulant matrix with $\det(\Delta_k) = 2$.

Then, being $\xi \in \mathbb{Z}^{3h}$ the labels for the hexagonal toroidal graph, the interior area of the triangulated faces of the corresponding triangle-square tiling $\mathcal{T}$ generated by $\xi$ over $\mathcal{G}^*_\xi$ is given by the integral quadratic form $(\sqrt{3}/4)\xi^T \Delta \xi$. Where $\Delta$ is an $3h \times 3h$ integer matrix

[8] The same derivation can be made to the 6 possible orientations of the labeled hexagon, obtaining matrices $D_i$, for $i = 2, \ldots, 6$ that are circular shifts by $i-1$ of $D_1$. Since they are all equivalent to measure the area, we keep our canonical orientation.

with a $3 \times 3$ diagonal block for each face, this is:

$$\Delta = \begin{pmatrix} \Delta_1 & 0 & \cdots & 0 \\ 0 & \Delta_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots \\ 0 & 0 & \cdots & \Delta_h \end{pmatrix} = (\Delta_1 \oplus \Delta_2 \oplus \cdots \oplus \Delta_h)$$

Where $(\Delta_1 \oplus \Delta_2 \oplus \cdots \oplus \Delta_h)$ is a decomposition of the quadratic form given by $\Delta$, a *regular* quadratic form [9], since $\det(\Delta) = 2^h$.

This development corresponds to the triangulated area inside the equiangular hexagonal faces of $\mathcal{T}$ generated by the labeling. Additionally, each vertex in the hexagonal graph corresponds to a triangle on the tiling, which adds $2h(\sqrt{3}/4)$ to the area. At last, we consider the number of squares, given by the length of the edges, which corresponds to an area of $\xi_1 + \xi_2 + \ldots + \xi_{3h} = \mathbb{1}^T\xi$, where $\mathbb{1}^T = (1, 1, \ldots, 1)$.

Thus, the total area of the fundamental region of the triangle-square tiling $\mathcal{T}$ generated by labeling $\xi$ over graph $\mathcal{G}^*_\xi$ is:

$$\begin{aligned} \text{Area} &= \sqrt{3}/4 \left(\xi^T\Delta\xi + 2h\right) + \sum_{j=1}^{3h} \xi_j \\ &= \sqrt{3}/4 \left(\xi^T\Delta\xi + 2h\right) + \mathbb{1}^T\xi \end{aligned}$$

We already established in Section 2.4 that the area of the fundamental region of a tiling has the form $a + b\sqrt{3}/2$, and pointed out in Section 5.1 that for triangle-square tilings, this correspond exactly to the number of squares and triangles. Then, if $\xi$ is an edge labeling of $\mathcal{G}^*_\xi$ dual graph of $\mathcal{T}$, which fundamental region area is $a + b\sqrt{3}/2$, we have that:

$$\begin{aligned} \xi^T\Delta\xi + 2h &= 2b \\ \mathbb{1}^T\xi &= a \end{aligned}$$

With this formulas in hand, one could state the problem of finding a tiling with area $a + b(\sqrt{3}/2)$ in the form of an optimization program:

$$\begin{aligned} \min \quad & \|\xi^T\Delta\xi + 2h - 2b\| + \|\mathbb{1}^T\xi - a\| \\ \text{s.t.} \quad & G\xi = 0, \\ & \xi \geq 0 \end{aligned}$$

However, this has many solutions and it is not a practical approach. We are not going to analyze this further.

*Example.* The labeling shown in Figure 5.25 generates the triangle-square tiling in Figure 5.39. The area of the fundamental region is $\xi^T\Delta\xi + 2h = 374 + 16 = 390$ triangles and $\mathbb{1}^T\xi = 67$ squares. The fundamental region of the tiling generated by this labeling has a total of $67 + 390(\sqrt{3}/4) \approx 235.9u^2$. Observe that most of the area comes from the triangulated regions of triangles in set $\mathcal{L}$, which grows rapidly as the label values increment, while the area of the triangles in set $\mathcal{E}$ is fixed in $2h$, independent from the value of the labeling.

## 5.7 *Complementary edge labeled dual graphs*

When we started this analysis we chose one of two possible dual graph orientations, the one for the set $\mathcal{E}$ of triangles, however, all the analysis made so far is analog for $\mathcal{L}$.

Let $\mathcal{T}$ be a triangle-square tiling with $\mathcal{E}$ and $\mathcal{L}$ disjoint sets of triangle faces in the two orientations. These are also the sets of vertices of the restricted dual tilings ${\mathcal{T}_{\mathcal{E}}}^*$ and ${\mathcal{T}_{\mathcal{L}}}^*$, respectively. Let $\mathcal{G}^*_\xi$ and $\mathcal{G}^*_\lambda$ be the labeled dual hexagonal graphs, with labels $\xi$ and $\lambda$. Let the area of the fundamental region of $\mathcal{T}$ be $a + b\sqrt{3}/2$.

Since sets $\mathcal{E}$ and $\mathcal{L}$ are disjoint and contain all the triangles in $\mathcal{T}$, we have that $|\mathcal{E}| + |\mathcal{L}| = 2b$. Also, the set of vertices in $\mathcal{G}^*_\xi$ is the same as in ${\mathcal{T}_{\mathcal{E}}}^*$, then $|\mathcal{E}| = 2h$. Respectively, $|\mathcal{L}| = 2\ell$, the number of vertices in $\mathcal{G}^*_\lambda$. Clearly, $h + \ell = b$, and putting together the geometric constraints derived in Section 5.4 and the area equations derived in Section 5.6, we have the following complementary formulation for the edge labeled dual graphs:

$$\begin{aligned} \xi^T \Delta \xi &= 2(b-h) & \qquad \lambda^T \Delta' \lambda &= 2(b-\ell) \\ \mathbb{1}^T \xi &= a & \qquad \mathbb{1}^T \lambda &= a \\ G\xi &= 0 & \qquad G'\lambda &= 0 \\ \xi \geq 0, \quad \xi &\in \mathbb{Z}^{3h} & \qquad \lambda \geq 0, \quad \lambda &\in \mathbb{Z}^{3\ell} \end{aligned}$$

In the case that $\mathcal{L} = \emptyset$, we have that $\ell = 0$ and $h = b$, and the complementary problem is empty, since $\mathcal{G}^*_\lambda$ is the empty graph. Moreover, this imposes $\xi^T \Delta \xi = 0$, meaning that the labeling must be *flat*, in the sense that no hexagonal triangulated regions exist. This is consistent with the definition given in Section 5.2 for this case: the tiling contains infinite parallel stripes of triangles and squares. The same is true when $\mathcal{E} = \emptyset$ in a symmetric way.

Another case for $\ell = 0$ happens when the labeling $\xi = 0$. Observe that this implies $h = b$, but also $a = 0$. This labeling means that the edges of ${\mathcal{T}_{\mathcal{E}}}^*$ cross no squares, it has equilateral hexagons only. Then, tiling $\mathcal{T}$ is the $W$ Archimedean tiling repeated $h$ times over the flat torus, as illustrated in Figure 5.37. This variety of generated tilings are useful when applying a vertex removal strategy, which generates hexagons in distinct positions of the translation lattice.

Each labeling determines both the tiling and the complementary labeling, since the tiling can be reconstructed from each of them, and it defines both labelings. Each branch of the complementary representations in Figure 5.7 completely characterizes a tiling, from the construction of the restricted dual and the labeled hexagonal graph back to its generation from the labeling. There is a strong symmetry of this analysis under a rotation by $\omega$.

Suppose $h, \ell > 0$, then

$$h = \frac{1}{2}\lambda^T \Delta' \lambda, \quad \text{and} \quad \ell = \frac{1}{2}\xi^T \Delta \xi.$$

Meaning that the number of vertices in the hexagonal dual graph is a quadratic function of the complementary labeling. In a geometric sense, increments in the labeling potentially grow the area of triangulated hexagonal regions, in which each triangle is a vertex of the complementary dual graph. The exception to this simultaneous growth happens when the labels increment keeping the area constant. When $\ell = 0$ it means the resulting tiling has no triangulated regions, meaning $\mathcal{L} = \emptyset$, as it has been shown before.

### *Complexity of a triangle-square tiling*

With these complementary properties in mind, we propose the **triangle-square tiling complexity measure**, given by:

$$\begin{aligned} \varsigma(\mathcal{T}) &= \frac{4h\ell}{(h+\ell)^2} \\ &= \frac{2h}{b^2}\xi^T \Delta \xi \end{aligned}$$

Which is a symmetric function on $h$ and $\ell = b - h$, capturing the symmetry of the representation of the tiling by both labelings. This is a real-valued function in $[0, 1]$. $\varsigma(\mathcal{T}) = 0$ when the tiling is formed either by infinite stripes of triangles and squares or by triangles only. $\varsigma(\mathcal{T}) = 1$ when $h = \ell$, meaning that the tiling has the same number of triangles in each of the orientations.

The second expression for $\varsigma(\mathcal{T})$ shows how the complexity relates to the labeling. If the label values becomes too large, the complexity of the tiling drops significantly, since the tiling gets overpopulated with triangles in set $\mathcal{L}$, while $h$ remains constant. The number of squares grows linearly on the values of the labeling, while the number of triangles in $\mathcal{L}$ increases in a quadratic way. This explains the asymptotic drop in the complexity measure as the labeling grows arbitrarily.

The more interesting tilings, the ones with the higher complexity measure, have a certain *equilibrium* between the faces in them. These tilings are those where $h \simeq b/2$, with relatively small numbers on the labeling, which keeps the ratio of squares and triangles in a controlled threshold.

*Example.* The tiling from Figure 5.39, has a complexity measure of 0.041. The small complexity measure is due to the large triangulated areas visible in the figure. The tilings in the basis, shown in Figure 5.40 have either complexity measure 0.32 or 0, when they are formed by infinite stripes.

## 5.8 Generating triangle-square tilings

We have shown that, for each $h \in \mathbb{N}$, for each $i = 1, \ldots, \varphi(h)$, we are able to generate a translation lattice $T_{(i,h)}$, a hexagonal tiling topology $F_{(i,h)}$, and a Hilbert basis $X_{(i,h)}$ for the set of valid labelings, such that

$$\mathbb{T}_{(i,h)} = \text{span}\left(X_{(i,h)}\right)$$

is a triangle-square tiling space isomorphic to $\mathbb{Z}_+^m$. In fact, the set of valid labelings is closed under positive integer linear combinations. The linear combinations of valid labelings can be interpreted as operations on the tilings themselves as long as they belong to the same generated family $\mathbb{T}_{(i,h)}$.

The examples we have shown before are sufficient to conclude that there are duplicates in $\mathbb{T}_{(i,h)}$ up to isometry, and also across the different spaces up to lattice reduction. However, even when the basic tilings of each space show redundancy, we can only filter after the generation. To filter duplicates in collections of tilings generated in each family, we can apply the following processes:

1. Find the minimal fundamental region and seeds, as explained in Section 3.3.
2. Find the symmetry group and classify the tiling in its uniformity and Archimedean classes, as explained in Section 4.3.

For now, we use the tilings in the form they are generated to better understand the meaning of the Hilbert base and the behavior of these tiling spaces. In fact, we do not even need to actually generate the tilings, since all the information is collected from the labeling without the need to produce the geometric object.

We generate samples for each space with binary coefficients for the subset of the Hilbert basis $\widehat{X}_{(i,h)}$ found with a heuristic approach.[9] This is, we generate all the labelings of the form:

$$\xi = \sum_{j=1}^{m'} a_j \widehat{X}^j_{(i,h)}$$

with $a_j \in \{0, 1\}$, except $\xi = 0$. For each $\mathbb{T}_{(i,h)}$, with $i \in \{1, ..., \varphi(h)\}$, this generates $2^{m'} - 1$ tilings. Then, for each value of $h$, the total generated tilings is at least $\varphi(h)2^{h+2} - 1$. We ran a test set for $h = 1, \ldots, 20$, which corresponds to $\varphi(1) + \ldots + \varphi(20) = 79$ different families, a total of over 100 million tilings.[10] The total number of tilings generated for each family $\mathbb{T}_{(i,h)}$ using binary combinations of the generators is shown in Table 5.3. If ones desires to include the trivial labeling $\xi = 0$, it adds one more tiling in each table entry.

For each valid labeling, which is a generated tiling, we compute and collect the following data:

- Area coefficients $a$ and $b$.
- Complexity measure.

[9] This method was developed before the use of *Normaliz* [14]. Every element found with it was a proper element of the Hilbert basis, as we verified before, using the specialized software. With it, $m'$ elements of the Hilbert basis were computed: $\widehat{X}_{(i,h)}$, with $m' = h + 2$ or $m' = h + 3$. The method tested binary combinations of the columns of matrix $N$ described in Section 5.4 and stored the minimal ones that were linearly independent from the others.

[10] Using the generators, this was accomplished in a few hours of computation.

- The vector with the coefficients of the labeling on the generators, which allows the generation of the tiling on demand.

Area and complexity measure are obtained from the labeling, with no need to generate the geometric objects. We identify each of these cases with the corresponding indices of the tiling family $(i,h)$, with $i \in \{1,...,\varphi(h)\}$. We also compute the index $\ell$ of the complementary labeling.

| $h$ | $i \in \{1,\ldots,\varphi(h)\}$ | | | | | | | | | Total |
|---|---|---|---|---|---|---|---|---|---|---|
| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | |
| 1 | 7 | | | | | | | | | 7 |
| 2 | 15 | 15 | | | | | | | | 30 |
| 3 | 31 | 63 | | | | | | | | 94 |
| 4 | 63 | 63 | 127 | | | | | | | 253 |
| 5 | 127 | 255 | | | | | | | | 382 |
| 6 | 255 | 511 | 511 | 511 | | | | | | 1788 |
| 7 | 511 | 1023 | | | | | | | | 1534 |
| 8 | 1023 | 2047 | 2047 | 2047 | 2047 | | | | | 9211 |
| 9 | 2047 | 4095 | 4095 | | | | | | | 10237 |
| 10 | 4095 | 8191 | 8191 | 8191 | 8191 | | | | | 36859 |
| 11 | 8191 | 16383 | | | | | | | | 24574 |
| 12 | 16383 | 32767 | 32767 | 32767 | 32767 | 32767 | 32767 | 32767 | | 245752 |
| 13 | 32767 | 65535 | | | | | | | | 98302 |
| 14 | 65535 | 131071 | 131071 | 131071 | 131071 | | | | | 589819 |
| 15 | 131071 | 262143 | 262143 | 262143 | 262143 | | | | | 1179643 |
| 16 | 262143 | 524287 | 524287 | 524287 | 524287 | 524287 | 524287 | | | 3407865 |
| 17 | 524287 | 1048575 | | | | | | | | 1572862 |
| 18 | 1048575 | 2097151 | 2097151 | 2097151 | 2097151 | 2097151 | 2097151 | 2097151 | | 15728632 |
| 19 | 2097151 | 4194303 | | | | | | | | 6291454 |
| 20 | 4194303 | 8388607 | 8388607 | 8388607 | 8388607 | 8388607 | 8388607 | 8388607 | 8388607 | 71303159 |
| | | | | | | | | | Total | 100502457 |

Table 5.3: Total number of tilings generated for each family $\mathbb{T}_{(i,h)}$ as binary combinations of the generators.

Tilings were stored in the form of a binary string of size $3g$, representing the labeling $\xi$ for the dual hexagonal graph $\mathcal{G}^*_\xi$. This is enough to recreate and draw the tiling, since we have the translation matrix $T_{(i,h)}$, the face adjacency matrix $F_{(i,h)}$, and the generators set $X_{(i,h)}$ stored for each pair $(i,h)$. All the generated tilings can be stored in this compressed format and the geometric object is only created when it is required, using the algorithms described in Section 5.5. The data set is publicly available on the thesis website.[11]

[11] chequesoto.info/thesis.html

It is important to remember that the tilings in the set can generate many new exemplars. Any positive linear combination of the coefficients — or the labelings — generates a valid labeling, and thus another tiling. The challenge at this point is developing a system that creates the tilings in real time given a vector of non-negative integer coefficients. This way, one could explore the families on their coefficient space $\mathbb{Z}^m_+$, at least for the tiling families with lower $h$.

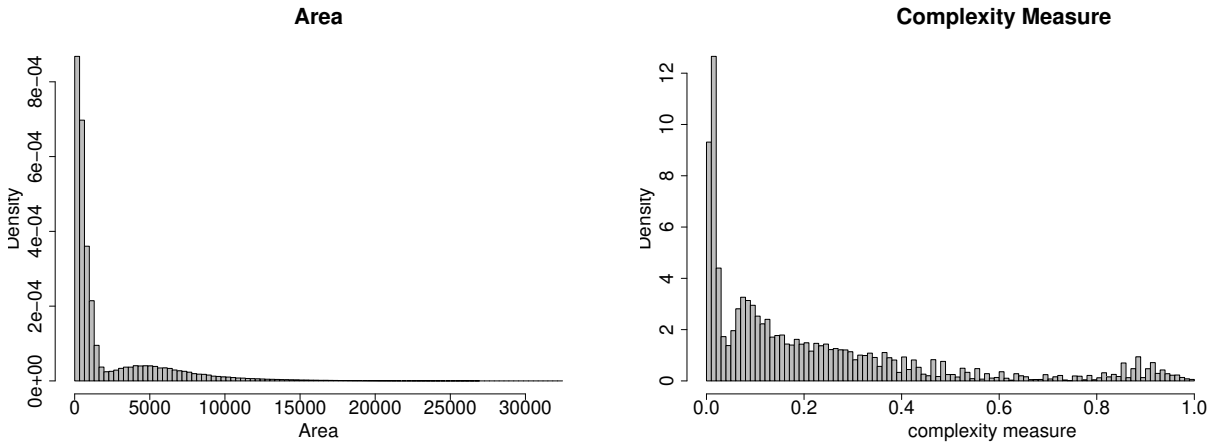



Figure 5.42: Distributions of area and complexity measure over the sample set of tilings.

The set of tilings produces tilings with very large areas despite the coefficients being binary, as one can observe in the histogram in Figure 5.42. In the figure we also show the distribution of the complexity measure on the sample set, where we can see a kind of three-modal distribution. One would probably find the group closer to 1 more interesting visually. We shall see some examples later.

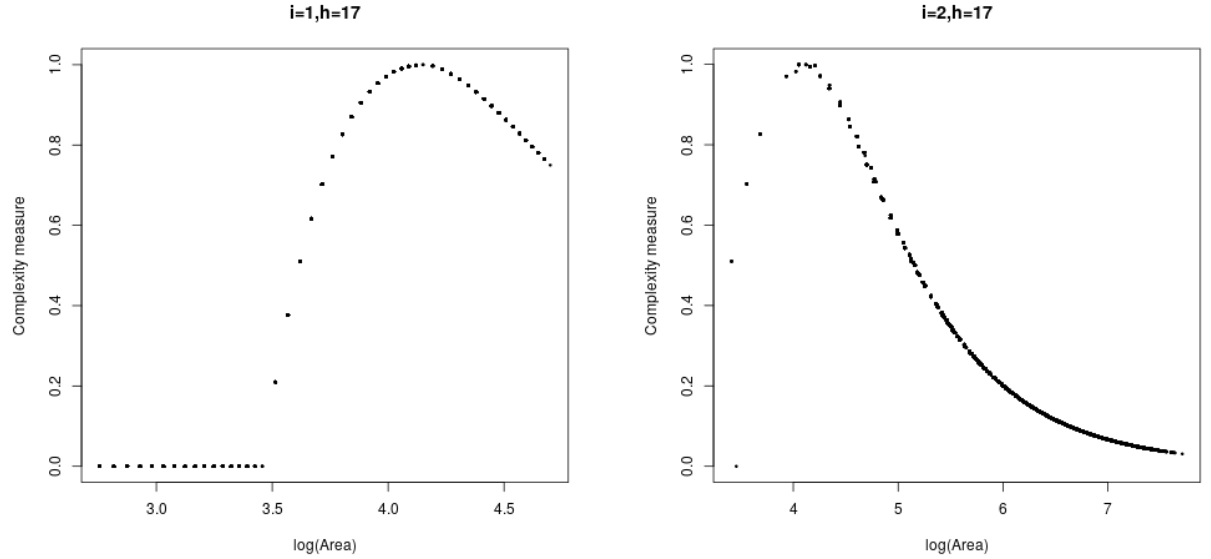



Figure 5.43: Complexity measure vs. logarithm of the area scatter plot, for sample tilings in families $\mathbb{T}_{(1,17)}$ and $\mathbb{T}_{(2,17)}$. Each point is a tiling.

We can observe some of the behaviors described in Section 5.6 for the complexity measure in relation with the area in Figure 5.43. The topology of $\mathbb{T}_{(1,17)}$ favors tilings with infinite stripes, where $\ell$ and the complexity measure are equal to zero. These tilings form the group + arranged in a horizontal line in the left of the first plot of Figure 5.43, since their complexity is zero. As the labeling increments its values, both area and complexity measure grow at first, and rapidly. After $h = \ell$, where the complexity becomes 1, the complexity decays asymptotically to zero as the labeling values and the area increase, as one can observe in the plot on the right of Figure 5.43.

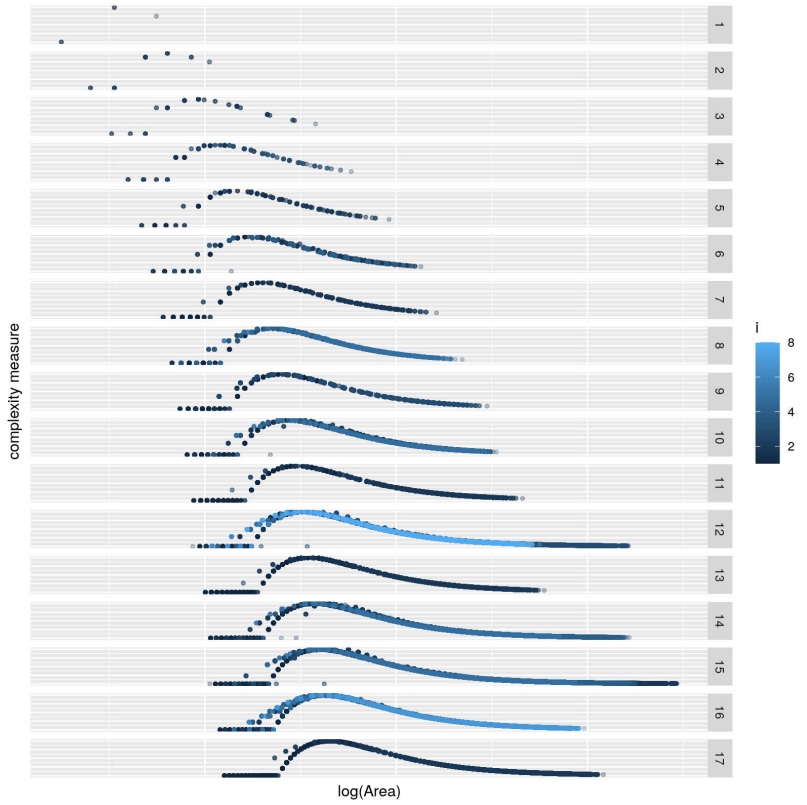



Figure 5.44: Complexity measure vs. logarithm of the area vs. $h$ scatter plot, for sample tilings in families with $h \leq 17$. Colors by $i \in \{1, \ldots, \varphi(h)\}$. Each point is a tiling.

The described behavior is consistent throughout different values of $h$, as we can observe in Figure 5.44. One can see the horizontal groups in the bottom left, that correspond to the infinite stripes tilings, whose complexity measure is zero. Also, we can observe the rapid growth in complexity as $\ell$ approaches $h$, and the posterior asymptotic decay in the complexity as the area grows.

There are many duplicates in the generated set. However, before eliminating them, a combinatorial vertex removal must be applied. For example: the labeling of all labels with value 1 could be reduced to the $h = 1$ torus, as illustrated in Figure 5.38; but each one of its faces can be transformed into a dodecagon. The same is the case for labeling $\xi = 0$ in each topology (although they are not counted in the results), hexagons must be created in each possible position in a sequential fashion before deciding which tilings can be removed as duplicates. Thus, we cannot eliminate directly on this collection, only after we have considered the combinatorial vertex elimination to form the full possibilities of tilings with regular polygons. This collection will be used to complete Table 1.2 in the future.

Finally, some examples of the tilings in the set are shown in the following page. Figure 5.45 has a complexity measure close to 1, we can see the equilibrium between the types of faces. Figure 5.46 for a larger value of $h$, grows in area but has a lower complexity measure, in contrast to Figure 5.47, where a tiling with $h = 8$ occupies a lot of area with a big loss of complexity, as one can perceive.

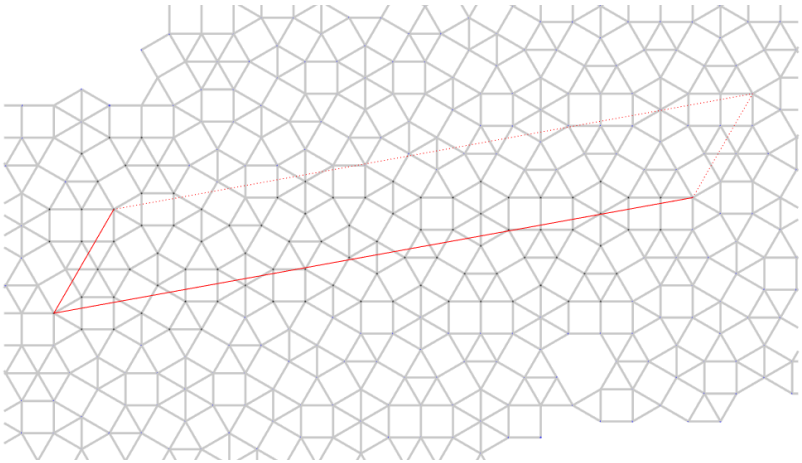


Figure 5.45: Example of a tiling generated in the sample set, $h = 17$, $i = 2$, complexity measure $\sim 0.99$, area of the fundamental region $\sim 57.6$.

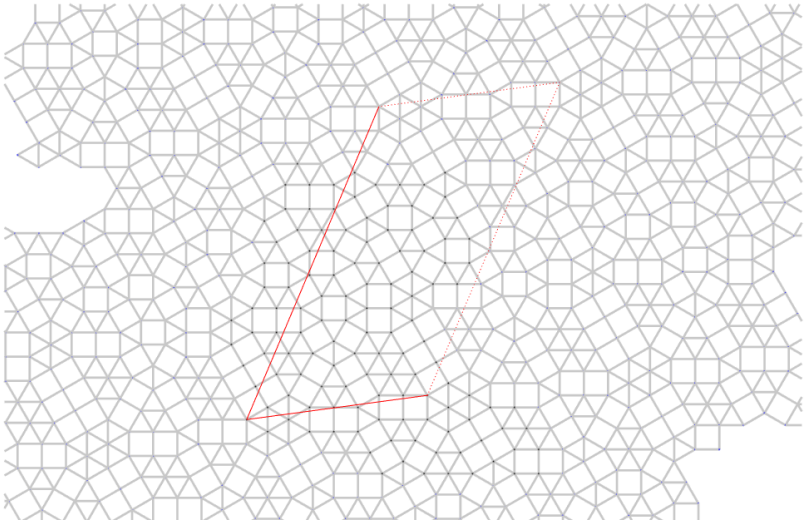


Figure 5.46: Example of a tiling generated in the sample set, $h = 20$, $i = 5$, complexity measure $\sim 0.95$, area of the fundamental region $\sim 91.0$.

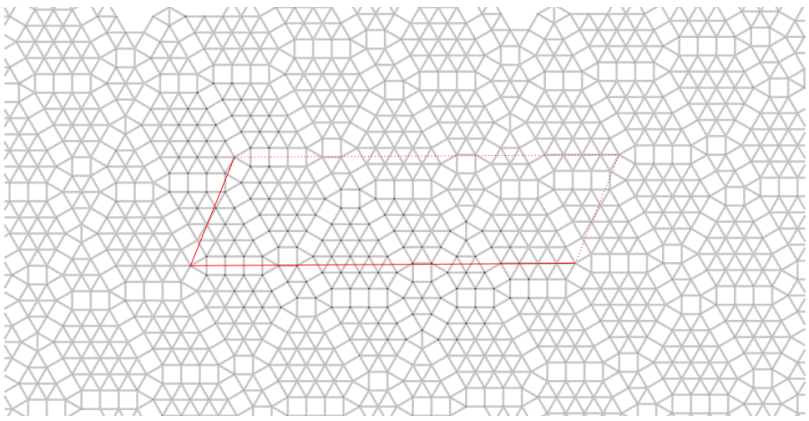


Figure 5.47: Example of a tiling generated in the sample set, $h = 8$, $i = 5$, complexity measure $\sim 0.33$, area of the fundamental region $\sim 120.3$.

## 5.9 *Unfolding the triangle tiling*

We have used the edge-labeled hexagonal dual graph $\mathcal{G}^*_\xi$ to generate a triangle-square tiling $\mathcal{T}$. However, we have not yet looked at its dual. The dual of a hexagonal tiling is a triangular tiling. Then, what we obtain as the dual of $\mathcal{G}^*_\xi$ is a triangular tiling on the same plane torus with the same edge-labeling, since the edges of the dual have a direct correspondence with the edges of the primal. We define the tiling $\mathcal{G}_\xi = (\mathcal{G}^*_\xi)^*$ as the triangle tiling with labeled edges that can be interpreted as a *folding* of tiling $\mathcal{T}$, as we will explain.

If we take one face of $\mathcal{G}^*_\xi$ with labels equal to zero in each edge, we obtain a $W$ type vertex on $\mathcal{T}$, center of Figure 5.49. There are exactly five minimal local labelings for the edges of a face: three middle axis *unfoldings*, and two alternate. These are shown around the zero labeling on Figure 5.49.

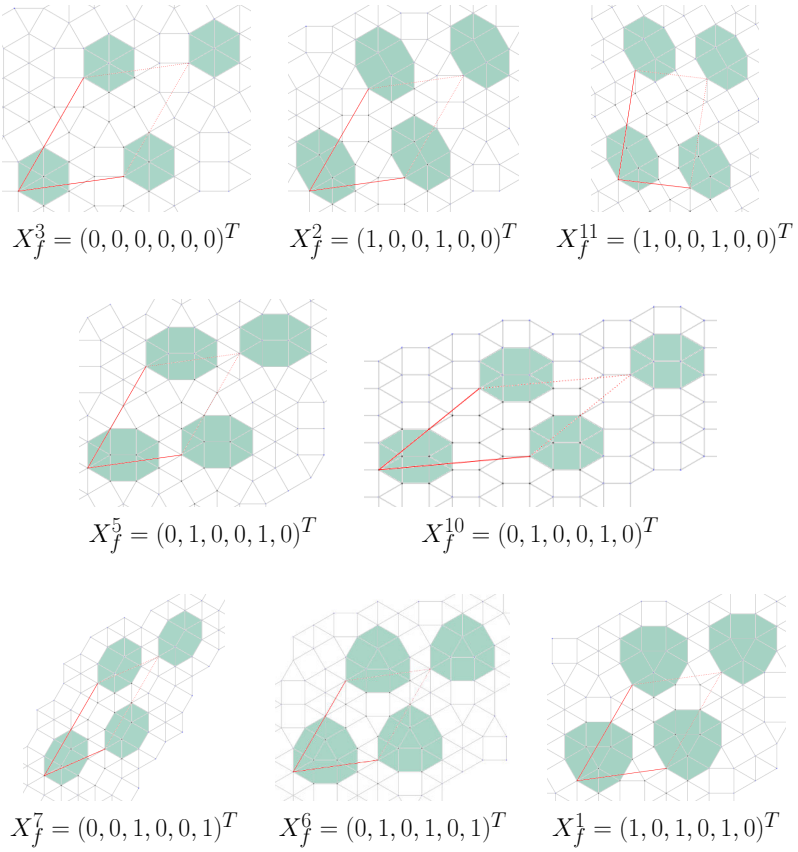



Figure 5.48: Some basic labelings $X^j$, from family $\mathbb{T}^8_3$, with face 1 of $\mathcal{G}_\xi$ highlighted in green.

Observe the face with index 1 of the hexagonal tiling under the the basic labelings from family $\mathbb{T}_{(3,8)}$, this is $i = 3$ and $h = 8$, shown in Figure 5.40. We have highlighted the face of interest in green in Figure 5.48 to show how it presents the five basic labelings or unfoldings. The basic labelings from Figure 5.49 are present in generators $X^3$, $X^2$ and $X^{11}$, $X^5$ and $X^{10}$, $X^6$, and $X^7$. The paired cases are the same locally, but, as one can observe in Figure 5.48, they correspond to different minimal labelings in the general setting.

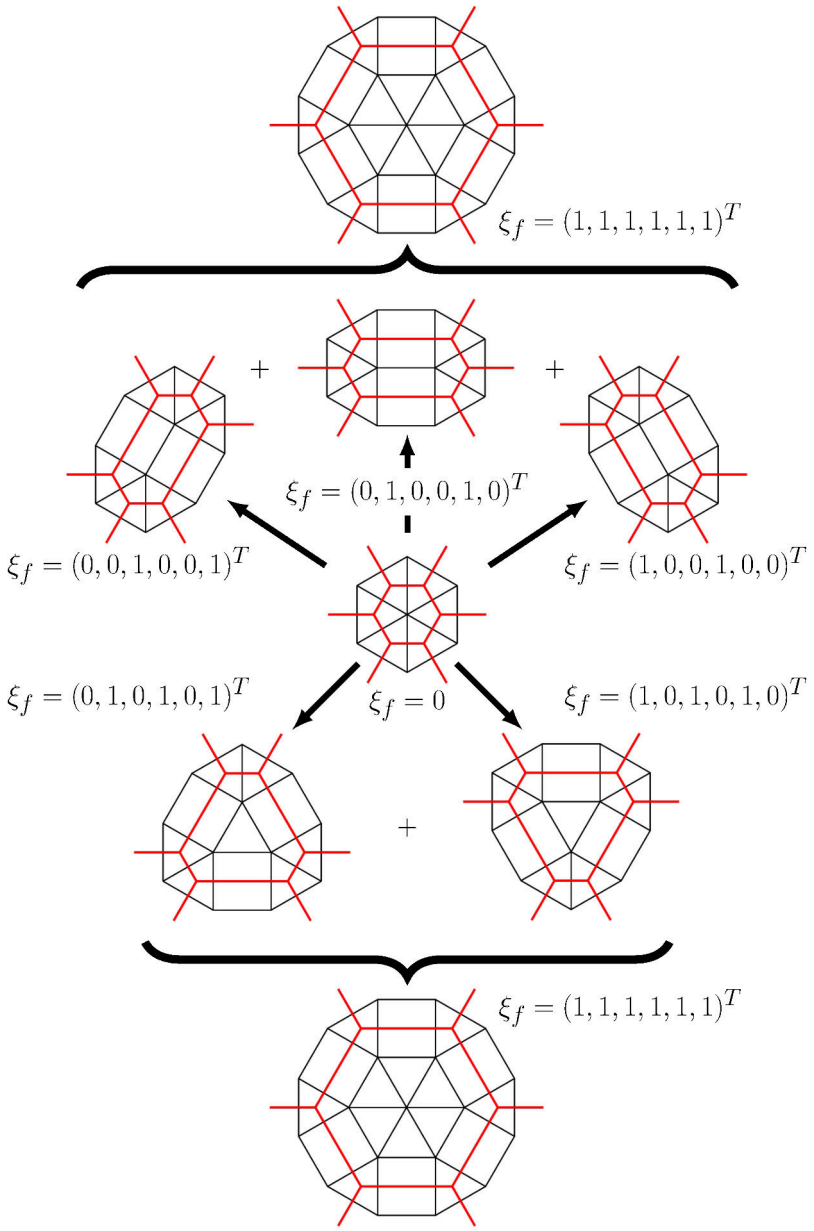



Figure 5.49: Each basic labeling $X^j$ represents an edge length in $\mathcal{T}_{\mathcal{E}}{}^*$ and an *unfolding* instruction on $\mathcal{T}_{\bar{\xi}}$. The image shows examples of the basic elements as individual unfoldings over the triangle tiling and the resulting ones when they are added.

Each basic labeling $X^j$ extends the local middle axis unfolding through the corresponding geodesic in $\mathcal{T}_{\bar{\xi}}$ in a way that it closes properly around the torus. This is illustrated in Figure 5.50 with $X^4$ and $X^{10}$, basic labelings of $\mathbb{T}^8_3$. What appears to be parallel stripes are actually the same geodesics over the plane tori, going around multiple times. Alternate unfoldings, generate triangulated hexagonal faces of the complementary labeling, as it is illustrated in Figure 5.50 with generators $X^1$ and $X^5$. This kind of unfolding requires a triangle from set $\mathcal{L}$, which we see unfolding together with the rest of the structure. As we can see in the set of generators, both the geodesics and the three-way unfoldings appear on their proper multiplicity, with no repetitions (Figure 5.40).

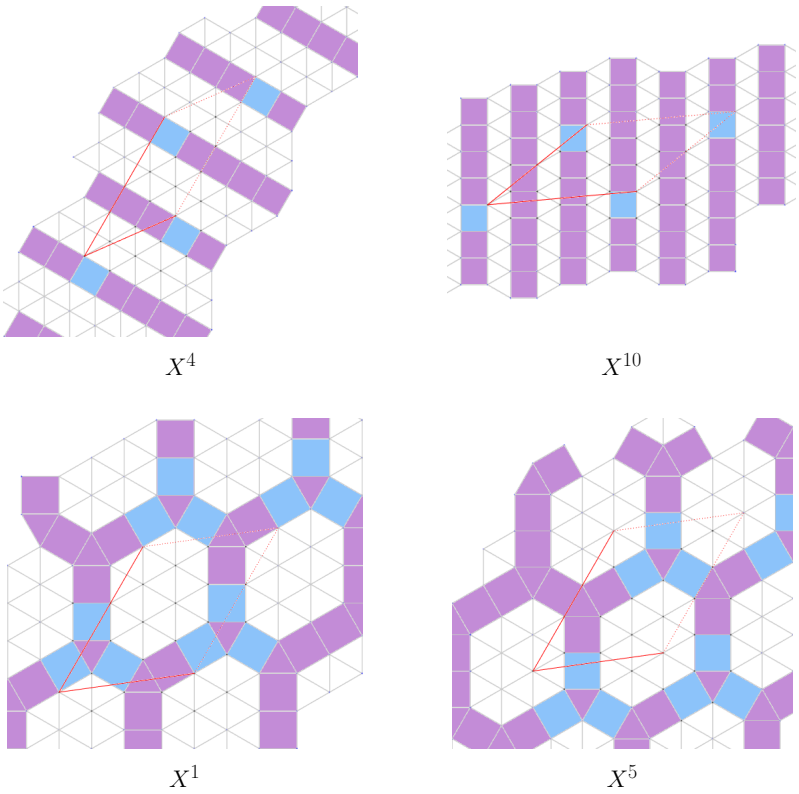



Figure 5.50: The unfolding of basic labeling $X^j$ goes around the torus following a geodesic, or through a hexagonal structure corresponding to a complementary labeling.

As we combine the basic labelings to form new tilings, more complex structures appear. We can think about the operations on the basic labelings as operations on the tilings themselves. As we have proved, any positive linear combination of valid labelings generates a valid labeling. The effect of the addition of a geodesic unfolding over another tiling is illustrated locally in Figure 5.51. The operation does not affect the topological elements of $\mathcal{G}^*_\xi$, the triangular faces in set $\mathcal{E}$, but only the spacing around them. As we have proved, the labeling increments reflect in the area corresponding to squares and to triangles in the complementary set $\mathcal{L}$, while $h$ remains constant, and the vertices in $\mathcal{E}$ are the reference positions that allow the unfolding operations.

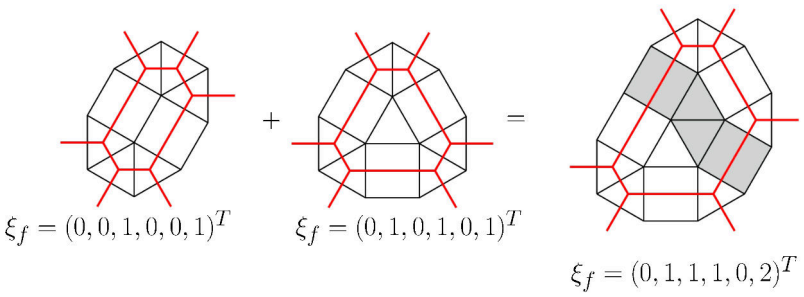



Figure 5.51: Example of the unfolding obtained when two basic elements in the base are added. The effect is one of unfolding over a *fracture line*.

Let us track a sequence of operations between tilings corresponding to basic labelings in $\mathbb{T}_{(4,6)}$ in Figure 5.52. It is not easy to read how the tilings combine in the addition operation over the labeling, sometimes. Even though the structure of one of the unfoldings is easy to perceive, the result has a complex structure. We have annotated some of the unfoldings in Figure 5.53 to aid the reader to perceive the way that the tilings unfoldings weave together. This geometric unfolding interpretation of the labelings' algebraic structure has an intrinsic beauty and has potential uses in physical fabrication of unfolding structures, even though this is not necessary to apply the algebraic operations on the tilings for their generation.

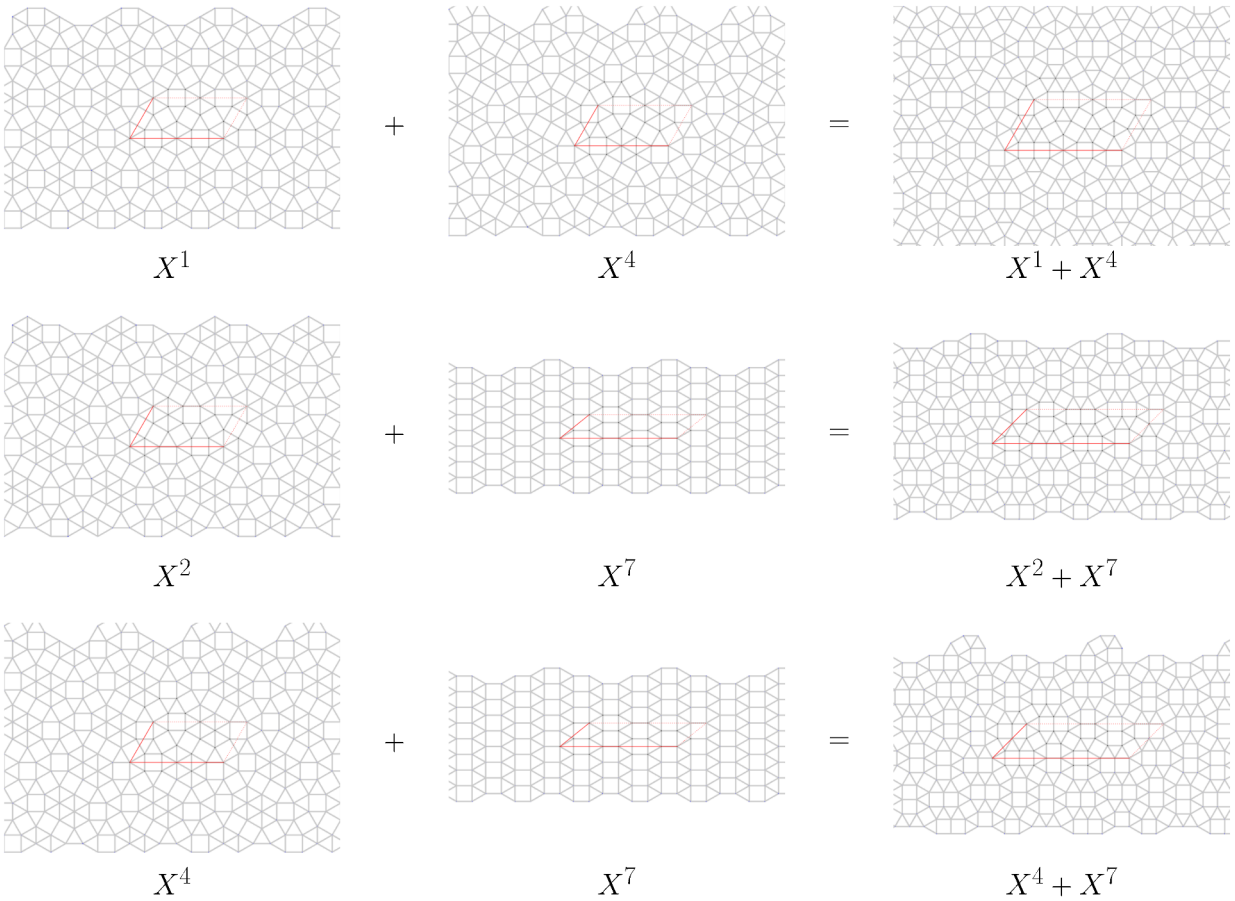



Figure 5.52: Examples of operations between tilings corresponding to basic labelings in $\mathbb{T}_{(4,6)}$.

The unfolding to which we are referring is properly origami paper folding. This is illustrated in Figure 5.6 and with more detail in Figure 5.54. Examples of some paper models are included in Figure 5.55. The creases are in the edges of the tiling in $\Lambda(\omega, \omega^3)$, which correspond to the triangles in set $\mathcal{E}$, and the valleys are indicated by the complementary restricted dual $\mathcal{T}_{\mathcal{L}}{}^*$. The result of folding over $\mathcal{T}_{\mathcal{L}}{}^*$ is the triangle labeled tiling $\mathcal{G}_{\tilde{\xi}}$, which labels indicate the numbers of foldings under the corresponding edge. The result is symmetric for the complementary dual, $\mathcal{G}_{\lambda}$ is obtained by folding over $\mathcal{T}_{\mathcal{E}}{}^*$.

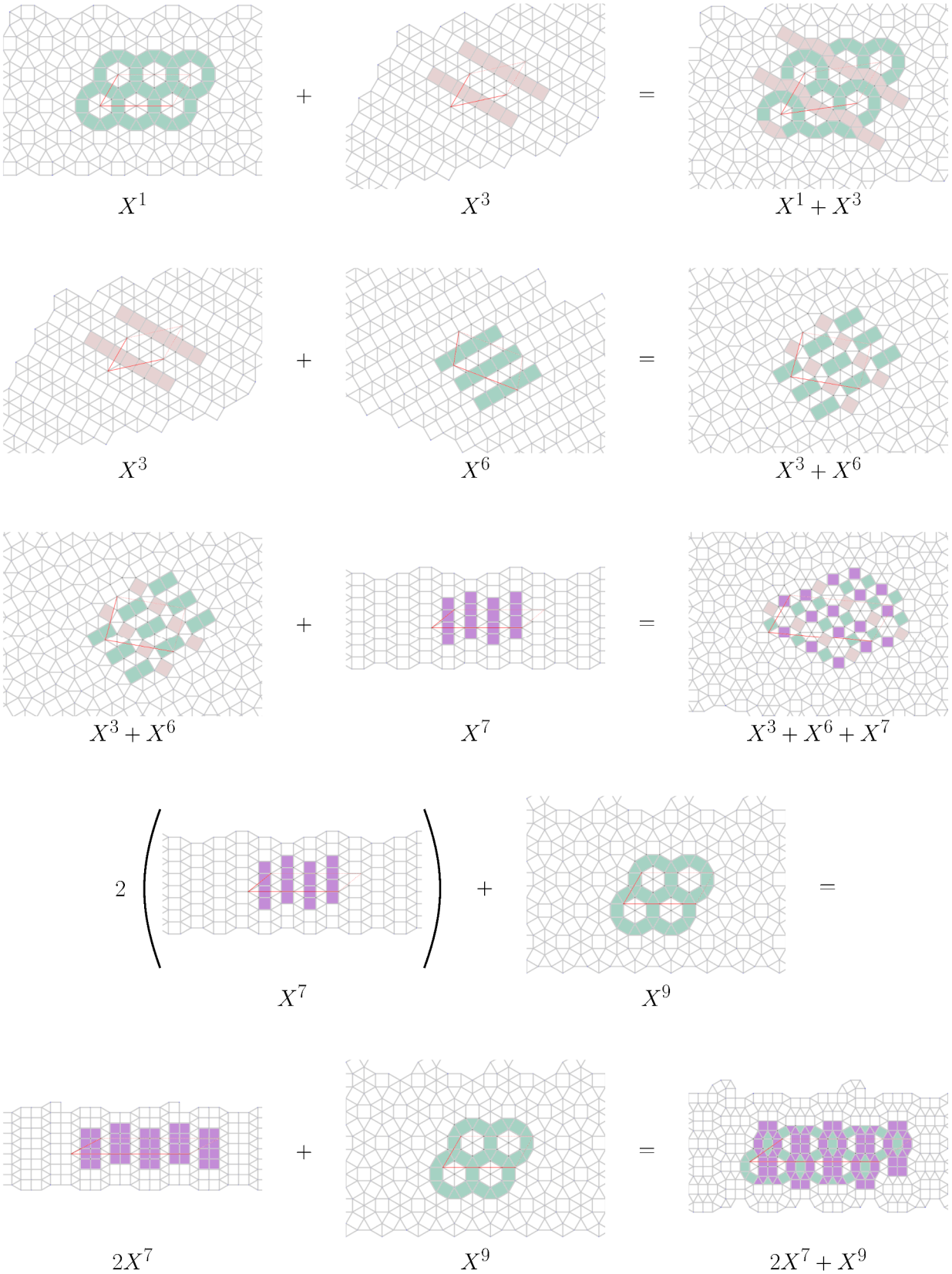



Figure 5.53: Examples of operations between tilings corresponding to basic labelings in $\mathbb{T}_{(4,6)}$, we highlight the unfoldings at each step.

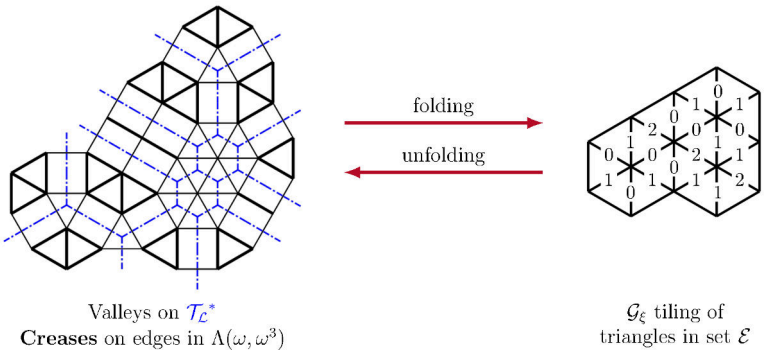



Figure 5.54: An origami crease pattern diagram to obtain $\mathcal{G}_\xi$ by folding the tiling over $\mathcal{T}_{\mathcal{L}}{}^*$.

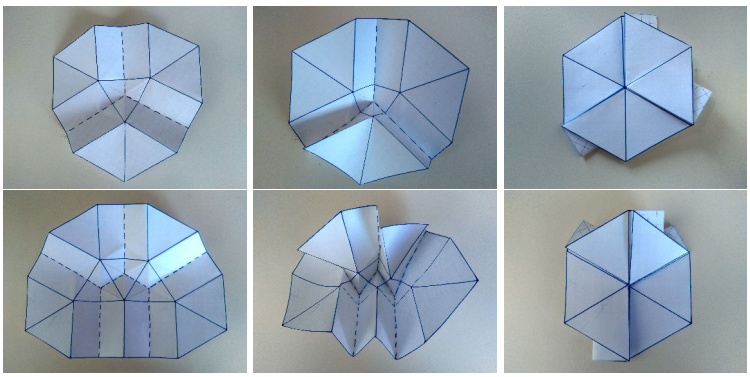


Figure 5.55: Origami folding local examples using the complementary restricted dual.

The algebraic structure offered by the basic labelings and their interpretation as unfoldings allow us to complete the diagram of relations of a triangle-square tiling $\mathcal{T}$ with its restricted dual $\mathcal{T}_{\mathcal{E}}{}^*$, its edge-labeled hexagonal graph $\mathcal{G}_{\tilde{\xi}}^*$, and finally, the folded tiling $\mathcal{G}_{\tilde{\xi}}$. The folded tiling is the proper dual of $\mathcal{G}_{\tilde{\xi}}^*$, an edge-labeled triangle tiling, whose labels encode unfolding operations between triangles in set $\mathcal{E}$ that generate tiling $\mathcal{T}$. As it is the case with the restricted dual and the hexagonal dual graph, $\mathcal{G}_{\tilde{\xi}}$ has a complementary folding $\mathcal{G}_\lambda$, a triangle tiling with the faces in the set $\mathcal{L}$.

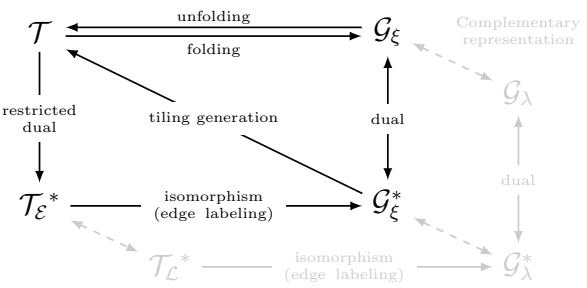



Figure 5.56: Relations between a triangle-square tiling $\mathcal{T}$, its restricted dual $\mathcal{T}_{\mathcal{E}}{}^*$, the edge-labeled dual hexagonal graph $\mathcal{G}_{\tilde{\xi}}^*$ and the folded triangle tiling $\mathcal{G}_{\tilde{\xi}}$.

## 5.10 Remarks & future work

The full characterization of triangle-square tilings and its algebraic structure presented here allow us to represent, generate and operate triangle-square tilings. The challenge to fully understand the tiling spaces that arise from the edge-labeled dual graph takes us to new branches of mathematics and will require further study. We enunciate some direct conclusions from what we have presented, but have not yet been fully formalized or organized. For each subject presented we also point out the work that needs no be done in the short and medium term.

### *Further understanding the algebraic structure*

*Complementary tilings.* Each tiling in a family $\mathbb{T}_{(i,h)}$ has a fixed value $\ell$, by complementary property of the restricted dual we know that this tiling belongs to some family $\mathbb{T}_{(\iota,\ell)}$. We have yet to develop a direct method to identify $\iota \in \{1,\dots,\varphi(\ell)$ from one of the available representations: $\mathcal{T}$, $\mathcal{T}_{\bar{\xi}}$ or $\mathcal{G}^*_{\bar{\xi}}$.

For example, the basic element $X^1$ of $\mathbb{T}_{(3,8)}$ is a tiling in family $\mathbb{T}_{(1,1)}$ with $\lambda = (1,2,2)^T$ after a rotation by $\omega$, as it is clear in the colored version in Figure 5.50. There is no ambiguity here, because there is only one topology for $\ell = 1$. However, as $\ell$ grows we need to understand how to identify the corresponding shape of its torus given by $\iota \in \{1,\dots,\varphi(\ell)$. This is related with the direct computation of $\mathcal{G}^*_\lambda$ from a tiling $\mathcal{T}$. We have presented how this relation is understood but no direct methods for this computation.

The algorithm to obtain $\mathcal{G}^*_\lambda$ from a given triangle-square tiling $\mathcal{T}$ would be similar to the one we use in Section 5.5. A graph traversal over restricted dual ${\mathcal{T}_\mathcal{L}}^*$ would register each vertex found, while a writing head $u \in \Lambda(1,\omega^2)$ builds the labeled graph. When the translations are reached on the traversal in ${\mathcal{T}_\mathcal{L}}^*$ the topology for $\mathcal{G}^*_\lambda$ is found. Obtaining the complementary labeling for the tilings in our test set requires the geometric construction of the tilings and their processing. At that point we will make the reduction and symmetry processing as well.

Deducing the corresponding topology of the complementary representation would allow us to operate the tiling with elements of this other family, generating new tilings and possibly obtaining further reductions of the generator sets.[12] This shall strengthen our derivation of the Hilbert basis for the tiling families in Section 5.4, and, possibly, establish the relations between families $\mathbb{T}_{(i,h)}$ and $\mathbb{T}_{(\iota,\ell)}$.

[12] The compass points towards **syzygies** and some other concepts from algebraic geometry.

*Sub-families.* A tiling from a family $\mathbb{T}_{(i,h)}$ is contained in families $\mathbb{T}_{(j,kh)}$ for $k \in \mathbb{N}$ when the translation lattice $T_{(i,h)}$ is a sub-lattice of $T_{(j,kh)}$. For example, in Figure 5.57 the lattice in the left is a sub-lattice of the one in the right, a tiling generated over it could be generated in the second setting and then reduced.

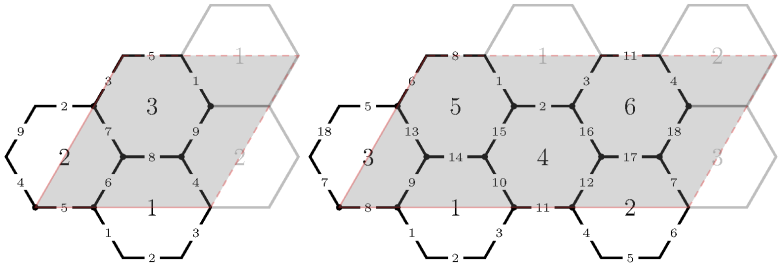


Figure 5.57: Translations lattices for hexagonal tilings with $h = 3$ and $h = 6$, respectively. The first is a sub-lattice of the second, a tiling generated over it could be generated in the second setting and then reduced.

Clarifying the criteria to identify the tori containing many tiling families would reduce enumeration efforts in the future. We would like to clearly identify the relation between some pairs of natural numbers $(i, h)$ and $(I, H)$ such that $\mathbb{T}_{(i,h)} \subset \mathbb{T}_{(I,H)}$. This would translate to some nesting structure of the tiling families, which would reduce the number of generators and the redundancy in tiling enumeration strategies.

Another way around this redundancy problem is to avoid periodicity in the edge labeling of $\mathcal{G}^*_\xi$, which is part of the symmetry detection strategies over the dual hexagonal graph we explain below.

### *Generating tilings with regular polygons*

As we described briefly before, one can directly obtain tilings with regular polygons from the triangle-square tilings with a *vertex removal* strategy. In addition, we have shown that we can compute many characteristics of a tiling $\mathcal{T}$ from the labels in $\mathcal{G}^*_\xi$ without actually producing the geometry of the tiling. This is also the case for detecting potential dodecagons and hexagons in the tiling.

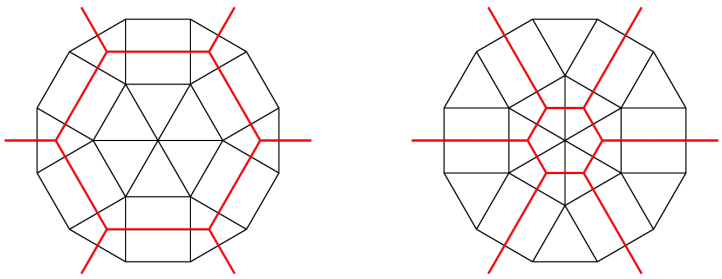


Figure 5.58: Dodecagon with its restricted dual. It corresponds to a face with all 6 labels equal to 1 in one orientation, and to a face with all zero labels surrounded by labels of value at least 1 in the other.

Hexagons appear either as faces with all their labels equal to zero or inside the triangulated faces of the hexagons in the restricted dual. While dodecagon faces correspond either to a hexagonal face labeled with 1 in all its edges, or to a face in which all labels are zero surrounded by labels with a value of at least 1, as one can see in Figure 5.58.

This means that one can even impose this kind of restrictions when generating a labeling to generate tilings containing hexagons or dodecagons. Moreover, every vertex type can have its restricted dual structure revealed and detected on the labeling.

*Symmetry detection in edge-labeled dual graphs*

There is another feature that can be detected in the edge-labeled hexagonal dual graph: symmetry. $\mathcal{G}^*_\xi$ contains all the information to recreate the geometry of $\mathcal{T}$, then any symmetry present in the original tiling must show itself in the edge-labeled hexagonal graph.

However, there are some exceptions: the $p4$ symmetry groups. When one rotates the hexagonal tiling by $\pi/2$, the result is on the complementary representation. The same is true for the 45° mirror. To verify if these symmetries exists one must compare the labeled dual graph with the complementary one. We show a triangle-square tiling for which this is the case in Figure 5.59. This is yet another reason to develop the equations and algorithms that allow us to derive $\mathcal{G}^*_\lambda$ from $\mathcal{G}^*_\xi$.

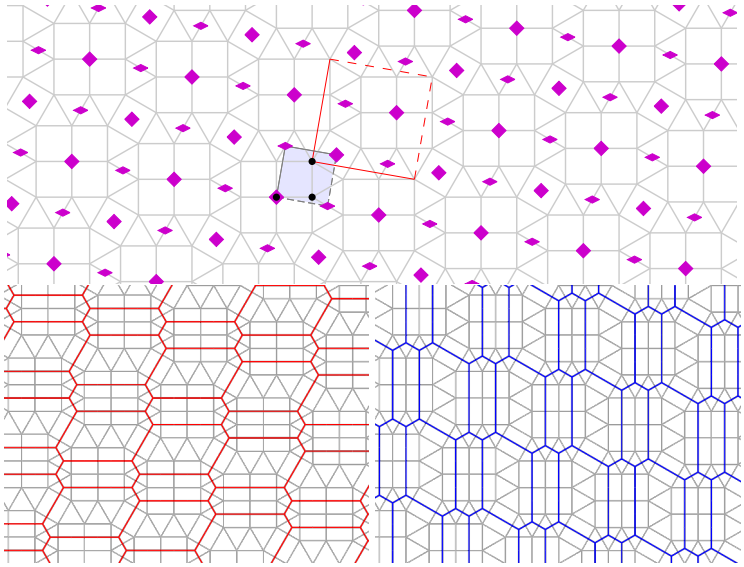


Figure 5.59: Example of a triangle-square tiling with 4-fold symmetry, both restricted duals are the same under the $\pi/2$ rotation. Tiling number 23 from the 3-uniform Galebach collection.

*Applications*

We are interested in modeling the foldings of our tilings, and there are already some references on the matter of mathematical origami [42, 52]. We belive that our folding tilings might be useful in the design and fabrication of unfolding physical structures.

Connelly et al. [19] describe a *lattice flexing* to create triangle and rhombus tilings from a tiling-square one and use it to create uniform circle packings. The process is easy to apply on our generated tilings, hence, this is a natural application of the results presented here.

Chavey [18] has shown the creation of dodecagon dense tilings from triangle-square tessellations, this is another natural and direct application of the work presented here.

Kormos [49] developed self-assembly molecules into $k$-uniform tilings. His approach considers only triangle-square tilings and the collection generated might be of interest for this area. Crasto de Lima et al. [54] study their electronic properties and topological phases, characterizing quantum spin Hall on them.

## 5.11 Related work

The hexagonal tiling[13], and its geometry, is an active research interest around which there is a large body of literature for varied geometric problems [48, 92], even for creating conformal tilings for visualization [84], and for applications in chemistry and materials science [7, 60]. Szucs [85] makes a profound discussion on affine tilings and regular maps on the flat torus. Here we contribute with the understanding of all the realizable hexagonal tilings on the graph torus from an algorithmic point of view. Our use of an edge labeling to represent triangle-square tilings is a novel application and one of the main contributions of this thesis, since it offers an algebraic structure for tilings.

[13] Polyhexes, fullerenes.

Weyrich [14] proposes a different approach to the generation of triangle-square tilings: a computer exhaustive search by vertex insertion in $\mathbb{Z}[\omega]$. He has also produced large collections of tilings.

[14] Unpublished work, to be included in a work co-authored by Medeiros e Sá, de Fiqueiredo and myself.

Roth [72] explored tilings of regular squares and triangles where the number of vertex types has been limited to a subset of all possible vertex types, mapped them to binary necklaces and classified them. His work is restricted to certain types of tilings appearing in crystalline structures.

Connelly et al. [19] show the relevance of triangle-square tilings for the computation of circle packing in the torus. They make a brief description of the tilings that resembles our representation, but they do not exploit it further.

Hull [42], Lang [52] and Verrill [88] analyze and model origami foldings, and combine Archimedean tilings and their duals to create origami tesselations. However, their foldings are aimed at creating aesthetically appealing foldings, not the compacting ones we are trying to model here. Their approach is useful to model the foldings we are proposing.

Medeiros e Sá et al. [76, 77] used dual inflation strategies over 3D tilings to create low density structures and void structures for digital fabrication. Sá et al. [75] also observed the existence of the fracture lines in the Archimedean meshes, as they call tilings with regular polygons.

The unfolding proposal resembles Verrill et al. hinged tilings from origami [88] and Nguyen et al. explorations of convex uniform tilings in origami [62], that use the dual of Archimedean tilings to create new tiling patterns. Also, the unfolding interpretation shown in this chapter offers a formal representation of the *fracture lines* observed on Archimedean meshes [75], and resembles *fault lines* from other tilings literature and crystallography [35, 49].

While this thesis was being written, Zeller et al. [95] published a program for displaying and exploring two-dimensional periodic tilings. Their enumeration is based on the concept of Delaney-Dress symbols and on the notion of Dress complexity.

# 6 *Concluding Remarks*

Tilings are a thrilling and very aesthetically pleasing subject. However, it is easy to get lost in their infinitude. We have covered a very long path for the representation of periodic tilings with regular polygons: from the integer-based representation of the tilings for their acquisition and analysis, we got to the algebraic structure that completely characterizes the triangle-square tilings in topological families. There is no doubt that new problems will appear in connection to computer science, branches of mathematics we foresee, and some others we do not even imagine yet.

We have solved a long-standing problem of characterizing all the tilings formed by triangles and squares [18, 82]. One of the clear continuations to this work is the actual completion of the table of $n$-uniform $k$-Archimedean tilings (Table 1.2), by searching further in the tilings families using higher coefficients for the generators and by applying the vertex removal strategy in a combinatorial way, followed by the application of lattice reduction and symmetry classification algorithms. Even though the process is outlined and has a strong foundation in this work, many important details need to be worked out.

Another open challenge is to develop a set of algorithms that compute the tiling on the fly and present it on the screen using only the generators' coefficients and the topological information. The algorithm described in Section 5.5 needs some improvements to accomplish that. But this would be the approach that needs the least amount of data to produce any tiling on screen for a user, since it does not depend on the proper geometric object. It remains an open question whether such an algorithm could take advantage of the use of the unfolding view of the tilings' generators and their combining operations.

Some of the supporting results used for deriving the algebraic structure of triangle-square tilings still require deeper understanding. Number theory is required to explain the behavior of $\varphi(h)$, the number of plane tori of certain size. The proper computation of Hilbert bases for the tiling families will require an algebraic geometry approach. The proper integer decomposition of the matrix with the geometric constraints of the labeling remains a challenge for developing a specific integer linear algebra solution. These are among the many other open problems we have mentioned in Section 5.10.

Besides the direct derivations of this work, there are various applications. We have mentioned some across the thesis, such as circle packing applications, including the *flexing* of the lattice, as described by Connelly [19]; the creation of dense dodecagon tilings from the triangle-square tilings as proposed by Chavey [18]; and, as reference to works like Kormos' [49], the development of self-assembly molecules into $k$-uniform tilings.

A big question that we will certainly investigate is whether the approaches to representing the tilings described here have some analogous representations on curved surfaces or even higher dimensions.

## *Art & design*

One of the first prototypes for our tiling web viewer had an interactive mode, in which one could change the positions of the vertices, and the change would be applied in the full tiling, examples of drawings produced by this prototype are shown in Figure 6.1. This idea, together with the possibility of designing curve paths on the edges constitutes a very interesting application for designers or common users who might want to design their own tilings from morphing tilings with regular polygons.

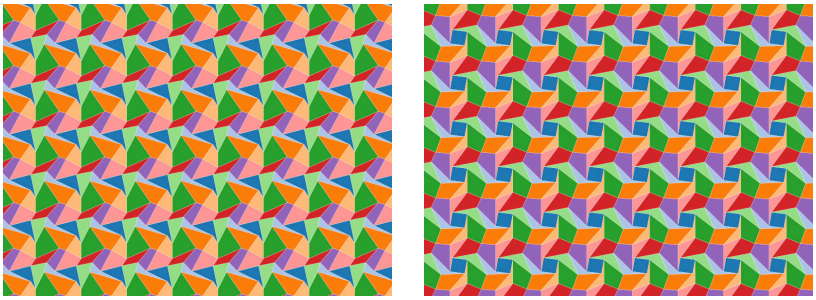


Figure 6.1: Examples of morphed tilings, produced by changing the position of some vertices of tilings with regular polygons.

Kaplan [43] developed a parametric algorithm to create islamic style designs from tilings with regular polygons, as illustrated in Figure 6.2. A straightforward application of our collections of tilings is to implement Hankin's method as described by Kaplan in his work.

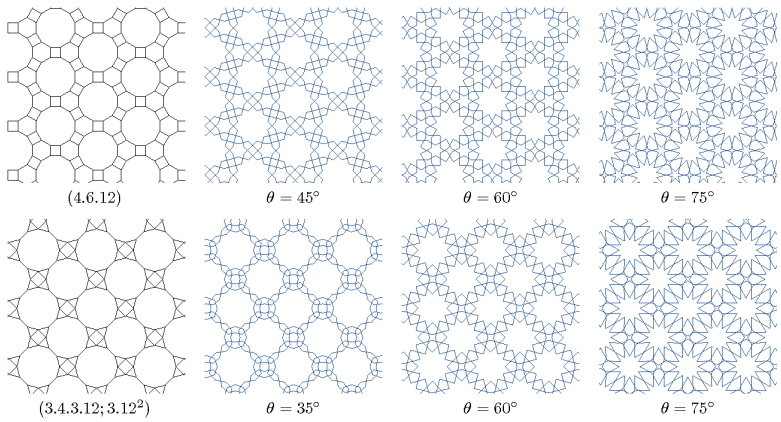



Figure 6.2: Examples of star patterns constructed using Hankin's method. Figure from Kaplan's work [43].

Origami folding of tilings represents an interesting approach for the fabrication of physical folding devices. Many applications are possible if we combine our tilings foldings with ideas for digital fabrication in the literature.

## *Education*

*The infinite tiling puzzle* is the name of the prototype for an interactive exhibit that has been derived from this work: a large set of laser cut pieces of the regular polygons that form tilings to create an interactive environment where the public learns about tilings. Some photos of the prototype are shown in Figure 6.3. The main idea is to invite the public to physically manipulate the polygons and discover which combinations form an Archimedean vertex, verifying Table 1.1 in their own experience, making sense of the equation that offers those solutions. Also, a very large catalog of tilings would be available for the participants to try to reproduce them in a playful challenging fashion.

Figure 6.3: *The infinite puzzle* prototype.

The origami folding would have applications in education and communication as well: from prints of the tilings with folding instructions to discover the properties of the triangle-square tilings to interactive content showing how the tilings can be added and multiplied by positive numbers.

# *Bibliography*


[1] Abas, Syed Jan. *Islamic geometrical patterns for the teaching of mathematics of symmetry*. Symmetry: Culture and Science, 12(1-2):53–65, 2001

[2] Adanova, Venera; Tari, Sibel. *Analysis of planar ornament patterns via motif asymmetry assumption and local connections*. Journal of Mathematical Imaging and Vision, 61(3):269–291, 2019
URL https://doi.org/10.1007/s10851-018-0835-8

[3] Agustí-Melchor, Manuel; Rodas-Jordá, Angel; Valiente-González, Jose-Miguel. *Classification of Repetitive Patterns Using Symmetry Group Prototypes*. SpringerLink, pp. 84–91, 2011
URL http://dx.doi.org/10.1007/978-3-642-21257-4_11

[4] Akiyama, Shigeki; Caalim, Jonathan; Imai, Katsunobu; Kaneko, Hajime. *Corona limits of tilings: Periodic case*. Discrete & Computational Geometry, 61(3):626–652, 2019
URL https://doi.org/10.1007/s00454-018-0033-x

[5] Aslaksen, Helmer; et al. *In Search of Demiregular Tilings*. Sarhangi, R. y Sharp, J. Bridges London: Mathematical Connections in Art, Music, and Science. Londres, Tarquin Books, pp. 533–536, 2006

[6] Baake, Michael. *A Guide to Mathematical Quasicrystals*. In Jens-Boie Suck; Michael Schreiber; Peter Häussler, eds., *Quasicrystals: An Introduction to Structure, Physical Properties and Applications*, pp. 17–48. Springer Berlin Heidelberg, Berlin, Heidelberg, 2002. ISBN 978-3-662-05028-6
URL http://dx.doi.org/10.1007/978-3-662-05028-6_2

[7] Baca, Martin; Shabbir, Ayesha. *Total labelings of toroidal polyhexes*. Science International, 24(3), 2012

[8] Ball, Derek. *Equiangular polygons*. Mathematical Gazette, 86(507):396–407, 2002. ISSN 0025-5572
URL http://dx.doi.org/10.2307/3621131

[9] Barot, Michael; González, Jesús Arturo Jiménez; de la Peña, José-Antonio. *Quadratic Forms - Combinatorics and Numerical Results*. Springer International Publishing, 2019. ISBN 978-3-030-05626-1
URL http://dx.doi.org/10.1007/978-3-030-05627-8

[10] Blanco, Maria Francisca; Harris, ALN. *Symmetry groups in the Alhambra*. In *VisMath*, volume 13, pp. 1–42. 2011

[11] Bodner, B Lynn; et al. *The planar crystallographic groups represented at the Alhambra*. Proceedings of Bridges 2013: Mathematics, Music, Art, Architecture, Culture, pp. 225–232, 2013

[12] Bonner, Jay. *Islamic Geometric Patterns - Their Historical Development and Traditional Methods of Construction*. Springer-Verlag New York, 2017. ISBN 978-1-4419-0216-0
URL http://dx.doi.org/10.1007/978-1-4419-0217-7

[13] Bremner, Murray R. *Lattice basis reduction: an introduction to the LLL algorithm and its applications*. CRC Press, 2011

[14] Bruns, W.; Ichim, B.; Römer, T.; Sieg, R.; Söger, C. *Normaliz. Algorithms for rational cones and affine monoids*. Available at https://www.normaliz.uni-osnabrueck.de
URL https://www.normaliz.uni-osnabrueck.de

[15] Bruns, Winfried; Ichim, Bogdan. *Normaliz: Algorithms for affine monoids and rational cones*. Journal of Algebra, 324(5):1098–1113, 2010. ISSN 0021-8693
URL http://dx.doi.org/10.1016/j.jalgebra.2010.01.031

[16] Chavey, D. *Tilings by regular polygons II: A catalog of tilings*. Computers & Mathematics with Applications, 17(1):147–165, 1989. ISSN 0898-1221
URL http://dx.doi.org/10.1016/0898-1221(89)90156-9

[17] Chavey, Darrah. *Periodic tilings and tilings by regular polygons I: Bounds on the number of orbits of vertices, edges and tiles*. Mitt. math. Semin. Giessen, 164(2):37–50, 1984

[18] Chavey, Darrah. *Tilings by Regular Polygons III: Dodecagon-Dense Tilings*. Symmetry-Culture and Science, 25(3):193–210, 2014

[19] Connelly, Robert; Dickinson, William. *Periodic planar disc packings*. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences, 372(2008), 2014. ISSN 1471-2962
URL https://doi.org/10.1098/rsta.2012.0039

[20] Conway, John H.; Burgiel, Heidi; Goodman-Strauss, Chaim. *The Symmetries of Things*. CRC Press, 2016. ISBN 978-143986489-0

[21] Conway, John H; Friedrichs, Olaf Delgado; Huson, Daniel H; Thurston, William P. *On three-dimensional space groups*. Contributions to Algebra and Geometry, 42(2):475–507, 2001
URL https://arxiv.org/abs/math/9911185v1

[22] Conway, John H.; Huson, Daniel H. *The Orbifold Notation for Two-Dimensional Groups*. Structural Chemistry, 13(3):247–257, 2002. ISSN 1572-9001
URL https://doi.org/10.1023/A:1015851621002

[23] Cousineau, Guy. *Tilings as a programming exercise*. Theoretical computer science, 281(1):207–217, 2002

[24] Critchlow, Keith. *Order in Space: A Design Source Book*. Thames & Hudson, 2000. ISBN 978-050034033-2

[25] Delgado-Friedrichs, Olaf. *Data structures and algorithms for tilings I*. Theoretical Computer Science, 303(2-3):431–445, 2003

[26] Dirk Frettlöh, Franz Gähler, Edmund Harriss. *Tilings Encyclopedia*, 2020. [Online; accessed 27. Mar. 2020]
URL http://tilings.math.uni-bielefeld.de

[27] Fedorov, Evgraf S. *The symmetry of regular systems of figures*. Zap. Mineralog. Obsc.(2), 28(1-146):28, 1891

[28] Galebach, Brian L. *n-Uniform Tilings*, 2002. [Online; accessed 24. Mar. 2020]
URL http://probabilitysports.com/tilings.html

[29] Gandhi, Dr. Haneet. *Method in Mathness, CATALOGUING 1-UNIFORM TILINGS*. Azim Premji University, 2017
URL http://publications.azimpremjifoundation.org/1343

[30] Gerdes, Paulus; et al. *African Basketry: Interweaving Art and Mathematics in Mozambique*. In *Bridges Coimbra Mathematics, Music, Art, Architecture, Culture Conference Proceedings*, pp. 9–16. 2011

[31] Ghyka, Matila. *The Geometry of Art and Life*. Dover Publications, 1977. ISBN 978-048623542-4

[32] Gotsman, Craig; Toledo, Sivan. *On the Computation of Null Spaces of Sparse Rectangular Matrices*. SIAM Journal on Matrix Analysis and Applications, 2008
URL https://epubs.siam.org/doi/abs/10.1137/050638369

[33] Grünbaum, Branko. *What symmetry groups are present in the Alhambra*. Notices of the AMS, 53(6):670–673, 2006

[34] Grunbaum, Branko; Shephard, Geoffrey C. *Tilings by Regular Polygons*. Mathematics Magazine, 50(5):227–247, 1977. ISSN 0025-570X
URL http://dx.doi.org/10.2307/2689529

[35] Grunbaum, Branko; Shephard, Geoffrey C. *Tilings and Patterns*. W.H.Freeman & Co Ltd, 1986. ISBN 978-071671194-0

[36] Hartley, Richard; Zisserman, Andrew. *Multiple view geometry in computer vision*. Cambridge university press, 2003

[37] Hemmecke, Raymond. *ON THE COMPUTATION OF HILBERT BASES OF CONES | Mathematical Software*. WORLD SCIENTIFIC, pp. 307–317, 2002
URL http://dx.doi.org/10.1142/9789812777171_0032

[38] Henk, Martin; Weismantel, Robert. *On Hilbert bases of polyhedral cones*, 1996. [Online; accessed 17. Jun. 2020]
URL https://opus4.kobv.de/opus4-zib/frontdoor/index/index/docId/223

[39] Hermann, Miki; Juban, Laurent; Kolaitis, Phokion G. *On the Complexity of Counting the Hilbert Basis of a Linear Diophantine System*. SpringerLink, pp. 13–32, 1999
URL http://dx.doi.org/10.1007/3-540-48242-3_2

[40] Hilbert, David; Cohn-Vossen, Stephan. *Geometry and the Imagination*. 87. American Mathematical Society, 1999

[41] Hofmann, Paul. *The Big List System of Tilings of Regular Polygons*, 2020. [Online; accessed 24. Mar. 2020]
URL https://www.biglist-tilings.com

[42] Hull, Thomas C. *Counting mountain-valley assignments for flat folds*. arXiv preprint arXiv:1410.5022, 2014
URL https://arxiv.org/abs/1410.5022v1

[43] Kaplan, Craig S. *Islamic Star Patterns from Polygons in Contact*. In *Proceedings of Graphics Interface 2005*, GI '05, pp. 177–185. Canadian Human-Computer Communications Society, Waterloo, CAN, 2005. ISBN 1568812655

[44] Kaplan, Craig S. *Introductory Tiling Theory for Computer Graphics*. Morgan & Claypool Publishers, 2009. ISBN 978-160845017-6

[45] Kari, Jarkko. *On the Undecidability of the Tiling Problem*. SpringerLink, pp. 74–82, 2008
URL http://dx.doi.org/10.1007/978-3-540-77566-9_7

[46] Kawarabayashi, Ken-ichi; Mohar, Bojan. *Graph and map isomorphism and all polyhedral embeddings in linear time*. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pp. 471–480. 2008

[47] Kepler, Johannes. *The harmony of the world*, volume 209. American Philosophical Society, 1997

[48] Kołodziejczyk, Krzysztof. *Realizable Quadruples for Hex-polygons*. Graphs and Combinatorics, 23(1):61–72, 2007. ISSN 1435-5914
URL http://dx.doi.org/10.1007/s00373-006-0688-6

[49] Kormoš, Lukáš; Procházka, Pavel; Makoveev, Anton O.; Čechal, Jan. *Complex k-uniform tilings by a simple bitopic precursor self-assembled on Ag(001) surface*. Nature Communications, 11(1856):1–6, 2020. ISSN 2041-1723
URL http://dx.doi.org/10.1038/s41467-020-15727-6

[50] Kowal, Pawel. *LREM SOLVE: Matlab Solver for Linear Rational Expectation Models*, 2006
URL https://EconPapers.repec.org/RePEc:dge:qmrbcd:157

[51] Krašek, Matjuška Teja. *Sharing some Common Interests of M.C. Escher*. M.C. Escher's Legacy: a Centennial Celebration, pp. 199–206, 2003
URL http://dx.doi.org/10.1007/3-540-28849-X_20

[52] Lang, Robert J. *Twists, tilings, and tessellations: mathematical methods for geometric origami*. CRC Press, 2017

[53] Lenngren, Nils. *k-Uniform Tilings by Regular Polygons*. Technical report, 2009
URL http://www.diva-portal.org/smash/get/diva2:444746/FULLTEXT01.pdf

[54] Crasto de Lima, F.; Ferreira, Gerson J.; Miwa, R. H. *Topological flat band, Dirac fermions and quantum spin Hall phase in 2D Archimedean lattices*. 21(40):22344–22350, 2019. ISSN 1463-9076
URL http://dx.doi.org/10.1039/C9CP04760C

[55] Liu, Shihuan; Leng, Ming; Ouyang, Peichang. *The Visualization of Spherical Patterns with Symmetries of the Wallpaper Group*. Complexity, 2018, 2018. ISSN 1076-2787
URL http://dx.doi.org/10.1155/2018/7315695

[56] Liu, Siying; Ng, Tian-Tsong; Sunkavalli, Kalyan; Do, Minh N; Shechtman, Eli; Carr, Nathan. *PatchMatch-based automatic lattice detection for near-regular textures*. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 181–189. 2015

[57] Liu, Yanxi; Hel-Or, Hagit; Kaplan, Craig S.; Gool, Luc J. Van. *Computational Symmetry in Computer Vision and Computer Graphics*. Foundations and Trends in Computer Graphics and Vision, 5(1-2):1–195, 2010
URL http://dx.doi.org/10.1561/0600000008

[58] McKay, Brendan D; Piperno, Adolfo. *Practical graph isomorphism, II*. Journal of Symbolic Computation, 60:94–112, 2014

[59] Moeck, Peter. *On classification approaches for crystallographic symmetries of noisy 2D periodic patterns*. IEEE Transactions on Nanotechnology, 18:1166–1173, 2019
URL https://arxiv.org/abs/1902.04155v3

[60] Moura, Lucia; Stojmenovic, Ivan. *Backtracking and isomorph-free generation of polyhexes*. Handbook of Applied Algorithms: Solving Scientific, Engineering, and Practical Problems, pp. 39–83, 2007

[61] Mukhopadhyay, Swapna. *The decorative impulse: ethnomathematics and Tlingit basketry*. ZDM, 41(1):117–130, 2009. ISSN 1863-9704
URL http://dx.doi.org/10.1007/s11858-008-0151-7

[62] Nguyen, Uyen; Fritzson, Ben; et al. *Origami Explorations of Convex Uniform Tilings Through the Lens of Ron Resch's Linear Flower*. In *Bridges 2018 Conference Proceedings*, pp. 515–518. Tessellations Publishing, 2018

[63] Ostromoukhov, Victor. *Mathematical tools for computer-generated ornamental patterns*. In *International Conference on Raster Imaging and Digital Typography*, pp. 193–223. Springer, 1998

[64] Ouyang, P.; Zhao, W.; Huang, X. *Beautiful Math, Part 5: Colorful Archimedean Tilings from Dynamical Systems*. IEEE Computer Graphics and Applications, 35(6):90–96, 2015
URL https://doi.org/10.1109/mcg.2015.135

[65] Oxborrow, Mark; Henley, Christopher L. *Random square-triangle tilings: A model for twelvefold-symmetric quasicrystals*. Physical Review B, 48(10):6966–6998, 1993. ISSN 2469-9969
URL http://dx.doi.org/10.1103/PhysRevB.48.6966

[66] Peng, Chi-Han; Pottmann, Helmut; Wonka, Peter. *Designing Patterns Using Triangle-Quad Hybrid Meshes*. ACM Trans. Graph., 37(4), 2018. ISSN 0730-0301
URL http://dx.doi.org/10.1145/3197517.3201306

[67] Plachinda, Paul. *Detection of symmetry using a crystallographic image processing algorithm*. arXiv preprint arXiv:1908.07932, 2019
URL https://arxiv.org/abs/1908.07932v1

[68] Polya, Georg. *XII. Über die Analogie der Kristallsymmetrie in der Ebene*. Zeitschrift für Kristallographie-Crystalline Materials, 60(1-6):278–282, 1924

[69] Ranscombe, Peter. *Geometry: a secret weapon in the fight against viruses*. The Lancet Infectious Diseases, 15(7):773, 2015
URL https://doi.org/10.1016/s1473-3099(15)00106-1

[70] Rao, Michael. *Exhaustive search of convex pentagons which tile the plane*. ArXiv e-prints, 2017
URL https://arxiv.org/abs/1708.00274

[71] Requicha, Aristides G. *Representations for Rigid Solids: Theory, Methods, and Systems*. University of Wisconsin–Madison. ACM Computing Surveys, 1980
URL https://dl.acm.org/doi/10.1145/356827.356833

[72] Roth, Johannes. *Restricted square-triangle tilings*. Zeitschrift für Kristallographie-Crystalline Materials, 223(11-12):761–764, 2008
URL https://doi.org/10.1524/zkri.2008.1049

[73] e Sá, Asla Medeiros; de Figueiredo, Luiz Henrique; Soto Sánchez, José Ezequiel. *Synthesizing periodic tilings of regular polygons*. In *2018 31st SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*, pp. 17–24. IEEE, Computer Press, 2018

[74] Sá, Ricardo. *Edros*. Projeto Editores Associados Ltda., 1982

[75] Sá, Ricardo; e Sá, Asla Medeiros. *Sobre Malhas Arquimedianas*. Editora Olhares, São Paulo, 2017

[76] Medeiros e Sá, Asla; Mello, Vinícius Moreira; Rodriguez Echavarria, Karina; Covill, Derek. *Adaptive voids*. Visual Computer, 31(6):799–808, 2015. ISSN 1432-2315
URL http://dx.doi.org/10.1007/s00371-015-1109-8

[77] Medeiros e Sá, Asla; Rodriguez Echavarria, Karina; Arnold, David. *Dual joints for 3D-structures*. Visual Computer, 30(12):1321–1331, 2014. ISSN 1432-2315
URL http://dx.doi.org/10.1007/s00371-013-0883-4

[78] Schattschneider, Doris. *The Plane Symmetry Groups: Their Recognition and Notation*. American Mathematical Monthly, 85(6):439–450, 1978. ISSN 0002-9890
URL http://dx.doi.org/10.2307/2320063

[79] Schattschneider, Doris. *Visions of Symmetry: Notebooks, Periodic Drawings, and Related Work of M.C. Escher*. W. H. Freeman, 1992. ISBN 978-071672352-3

[80] Shephard, G.C.; Ammann, R.; Grünbaum, B. *Aperiodic Tiles*. Discrete & computational geometry, 8(1):1–26, 1992
URL http://eudml.org/doc/131207

[81] Sloane, N.J.A. *The On-Line Encyclopedia of Integer Sequences (OEIS)*, 2002. [Online; accessed 24. Mar. 2020]
URL https://oeis.org/A068600

[82] Sommerville, Duncan M. Y. *XXIX.–Semi-regular Networks of the Plane in Absolute Geometry*. Transactions of the Royal Society of Edinburgh, 41(3):725–747, 1906
URL http://dx.doi.org/10.1017/S0080456800035560

[83] Soto Sánchez, José Ezequiel.; e Sá, Asla Medeiros.; de Figueiredo, Luiz Henrique. *Acquiring periodic tilings of regular polygons from images*. Visual Computer, 35(6):899–907, 2019. ISSN 1432-2315
URL http://dx.doi.org/10.1007/s00371-019-01665-y

[84] Sullivan, John M; et al. *Conformal tiling on a torus*. Proc. Bridges, Coimbra, pp. 593–596, 2011

[85] Szucs, J. M.; Klein, D. J. *Regular affine tilings and regular maps on a flat torus*. Discrete Applied Mathematics, 105(1):225–237, 2000. ISSN 0166-218X
URL http://dx.doi.org/10.1016/S0166-218X(00)00174-8

[86] Toth, Csaba D; O'Rourke, Joseph; Goodman, Jacob E. *Handbook of discrete and computational geometry*. Chapman and Hall/CRC, 2017

[87] Valiente, José M.; Albert, Francisco; Gomis, José María. *A Computational Model for Pattern and Tile Designs Classification Using Plane Symmetry Groups*. SpringerLink, pp. 849–860, 2005
URL http://dx.doi.org/10.1007/11578079_88

[88] Verrill, Helena; et al. *Origami tessellations*. In *Conference Proceedings of Bridges: Mathematical Connections in Art, Music, and Science*, pp. 55–68. 1998

[89] Wade, David. *Pattern in Islamic Art*, 2020. [Online; accessed 27. Mar. 2020]
URL https://patterninislamicart.com

[90] Washburn, Dorothy K.; Crowe, Donald W. *Symmetry Comes of Age: The Role of Pattern in Culture*. University of Washington Press, 2004. ISBN 978-029598366-0

[91] Washburn, Dorothy Koster; Crowe, Donald Warren. *Symmetries of Culture: Theory and Practice of Plane Pattern Analysis*. University of Washington Press, 1988. ISBN 978-029597084-4

[92] Wei, Xianglin; Wang, Jianjun; Gao, Feixing. *A note on area of lattice polygons in an Archimedean tiling*. Journal of Applied Mathematics and Computing, 48(1):573–584, 2015. ISSN 1865-2085
URL http://dx.doi.org/10.1007/s12190-014-0819-9

[93] Weisstein, Eric W. *Demiregular Tessellation*, 2020. [Online; accessed 24. Mar. 2020]
URL https://mathworld.wolfram.com/DemiregularTessellation.html

[94] Wikipedia. *Euclidean tilings by convex regular polygons*, 2020. [Online; accessed 08. Jun. 2020]
URL https://en.wikipedia.org/wiki/Euclidean_tilings_by_convex_regular_polygons

[95] Zeller, Rüdiger; Friedrichs, Olaf Delgado; Huson, Daniel H. *Tegula – exploring a galaxy of two-dimensional periodic tilings*. 2020
URL https://arxiv.org/abs/2007.10625v1

[96] Zong, Chuanming. *Can You Pave the Plane Nicely with Identical Tiles?* Notices of the AMS, p. 635, 2020
URL http://dx.doi.org/https://doi.org/10.1090/noti2075