

TECHNISCHE UNIVERSITEIT EINDHOVEN

BACHELOR FINAL PROJECT

Bamboozle Structures and Honeycombs

Timo Scholte

t.r.scholte@student.tue.nl

supervised by
prof.dr.ir. J. (Jan) Draisma

June 24, 2016

Abstract

This thesis focuses on creating a better understanding of bamboozle structures. Any information that can be found about the list of bamboozle structures (considering its completeness, underlying structures, or completeness under assumptions) is to be considered as valuable outcome. Since bamboozle structures are a relatively new concept that have so far strictly been viewed as single orbits under space groups, there are still possibilities to improve our understanding of such structures. In this thesis, we relate bamboozle structures to tessellations of three dimensional Euclidean space, namely convex uniform honeycombs. This is done by relating the problem to its two dimensional equivalent, identifying what convex uniform honeycombs hold (near-)bamboozle structures, proving of completeness of the list of used convex uniform honeycombs with the help of computer calculations, and considering what adjustments convex uniform honeycombs can withstand in order to create bamboozle structures. Though the search for new bamboozle structures proved unfruitful, we found that the hexagonal bamboozle structure was in fact not a bamboozle structure, discovered that the square bamboozle structure and the four-coloured rectangular bamboozle structure actually form continuous families, and gained a better understanding of the bamboozle structure and what areas should be considered to find a complete list of possible structures.

Contents

1	Problem Discription	3
1.1	Definitions	3
1.2	List of Known Bamboozle Structures	5
1.2.1	Square Bamboozle Structure	5
1.2.2	Triangular Bamboozle Structure	5
1.2.3	Hexagonal Bamboozle Structure	5
1.2.4	The Rectangular Bamboozle Structures	5
1.3	Goal	8
2	Shortest Distance Graphs in \mathbb{R}^2	9
2.1	The 11 Uniform Convex Tilings	9
2.2	Skew Representations	11
3	Bamboozles in \mathbb{R}^3	14
3.1	Scaling up from \mathbb{R}^2	14
3.2	Results of This Method	15
3.2.1	Cubic Honeycomb	16
3.2.2	Truncated Cubic Honeycomb	17
3.2.3	Alternated Cubic Honeycomb	18
3.2.4	Cantellated Cubic Honeycomb	20
3.2.5	Quarter Cubic Honeycomb	21
3.2.6	Rectified Cubic Honeycomb	22
3.2.7	Runcic Cubic Honeycomb	23
3.2.8	Runcitruncated Cubic Honeycomb	24
3.2.9	Gyroelongated Triangular Prismatic Honeycomb	25
3.2.10	Gyrated Triangular Prismatic Honeycomb	26
3.2.11	Gyroelongated Alternated Cubic Honeycomb	27
3.2.12	Conclusion	29
3.3	Completeness of Honeycombs	29
3.3.1	Results	35
4	Adjustments of Honeycombs	38
4.1	The Rectangular Bamboozle Structures	38
4.2	Applying Adjustments	40
5	Conclusion, Discussion, and Further Research	46

Chapter 1

Problem Discription

1.1 Definitions

Consider \mathbb{R}^3 equipped with the standard inner product and corresponding distance function $d(\cdot)$. We introduce the following concepts, adapted from [5]:

Definition 1. *A group is defined as a set of operations that combine in such a way that they meet the following requirements:*

1. *If two operations a and b are elements of the group, then their product $c=ab$, is also an element of the group.*
2. *The group contains an identity element e , so that for every element a it holds that $ea=ae=a$.*
3. *For every element a , there exists an element a^{-1} , such that $aa^{-1} = a^{-1}a = e$.*
4. *The group is associative: $a(bc) = (ab)c$.*

Definition 2. *A sub-group of the group of orthogonal 3×3 matrices that transforms a finite physical object to configurations that are indistinguishable from the original and leave at least one point unchanged is defined as a point group.*

The elements of point groups are rotations, reflections and improper rotations (the combination of rotations and reflections). Within three-dimensional Euclidean space, there are seven distinct point groups. [5]

Definition 3. *A space lattice (in \mathbb{R}^3) is an infinite set of points of the form $n_1 \cdot u + n_2 \cdot v + n_3 \cdot w$, with u, v, w three fixed linearly independent vectors, and n_1, n_2, n_3 three integers that can vary to form the points within the lattice.*

A space lattice always contains translative symmetry: if $d = n_1 \cdot u + n_2 \cdot v + n_3 \cdot w$ and $d^* = n_1^* \cdot u + n_2^* \cdot v + n_3^* \cdot w$ are both contained within the lattice, then so is $d + d^*$, since $n_i + n_i^*$ is an integer. Furthermore, a space lattice may have more symmetry, i.e., rotational and reflective symmetry. In fact, all lattices have the symmetry of one of seven point groups. To define space lattices in a way that clearly shows their point group, we use unit cells:

Definition 4. *A unit cell is defined as the parallelepiped spanned by three non-coplanar lattice translations.*

Definition 5. A unit cell is primitive if its set of translations has the minimal sum of lengths for such a set, or if it defines a cell with a volume equal to the volume of the shortest set.

Sometimes, it might be more convenient to define a unit cell with lattice translations that are related to the symmetry operations, for instance parallel to rotation axes or lying in a mirror plane. This may lead to a unit cell with a volume that is two, three or four times the volume of the primitive unit cell. This influences where in the unit cell lattice points can be found. This leads to various centering schemes, namely *body-centered*, *single-face-centered*, and *all-face-centered* (or simply *face-centered*). Taking account of symmetry and the various centering schemes, there are fourteen distinct space lattices[5] which are known as the *Bravais lattices*, named after their discoverer, A. Bravais.

The nature of space lattices allows for linear operations O of the form $Ox = Ax + b$ with A a matrix and b a vector of the same dimensions as x . If there exists an n such that $O^n x = x + a$ with a a vector of the same dimensions as x , then the operator O defines a group that is a super group of the translation group. If a is the null vector, then O is a point group. However, if a is a lattice translation, we can define two new symmetry operations:

Definition 6. A screw axis is the combination of a rotation, A , and a translation, b , parallel to the rotation axis and such that, if $A^n = I$, nb is a lattice translation.

Definition 7. A glide plane is the combination of a mirror operation, A , and a translation, b , parallel to the mirror plane and such that $2b$ is a lattice translation.

Combining this set of symmetry operations with the 14 Bravais lattices, provides 230 distinct groups, known as the *space groups*.

Let \mathbb{K} be a subset of \mathbb{R}^3 , such that there exists a $\delta \in \mathbb{R}$ so that $\forall k_0 \in \mathbb{K} \forall k \in \mathbb{K} \setminus k_0$ it holds that $d(k, k_0) > \delta$.

Definition 8. The minimal distance on \mathbb{K} is defined by $d_{min} = \inf_{k_1 \neq k_2 \in \mathbb{K}} \{d(k_1, k_2)\}$.

Definition 9. The (undirected) minimal distance graph of \mathbb{K} is defined as (\mathbb{K}, E) , where $E = \{(k_1, k_2) \mid d(k_1, k_2) = d_{min}\}$.

We embed the minimal distance graph G by means of line segments for edges. This all allows us to define the following new concepts:

Definition 10. The polytope representation of a vertex v within G is created by taking the convex hull of the centers of all edges incident to v .

Definition 11. The polytope structure of G is the combination of the polytope representations of all $v \in V$.

Definition 12 (Bamboozle Structures). Let \mathbb{K} be generated by a single orbit under one of the 230 space groups. The polytope structure of the minimal distance graph of \mathbb{K} is called a bamboozle structure if it meets the following three criteria:

1. The minimal distance graph of \mathbb{K} is connected.
2. All polytopes contained within the structure are polygons, i.e. lie within a 2-d plane.
3. If u and v are neighbours, then their polytope representations do not lie within the same plane.

1.2 List of Known Bamboozle Structures

Currently, five such structures are known. These are the square bamboozle structure, the triangular bamboozle structure, the hexagonal bamboozle structure, and two rectangular bamboozle structures. The square bamboozle structure was already known for some unknown time. The triangular structure was discovered by Koos and Tom Verhoeff, and Tom Verhoeff later discovered the others. In this section we will discuss the known structures, accompanied by imagery created by Tom and Koos Verhoeff.

1.2.1 Square Bamboozle Structure

The square bamboozle structure consist of squares that are placed at angles of 90 degrees. There are three distinct orientations the squares take, represented by the three different colours. See Figure 1.1.

1.2.2 Triangular Bamboozle Structure

The triangular bamboozle structure was the first structure that was recognized by Tom and Koos Verhoeff[6]. The discovery of this structure was what set off the curiosity for similar structures. The name bamboozle structure comes from the name Bamboozle, which was given to this structure. It consists of regular triangles meeting pairwise at angles of $\arccos(\frac{1}{3}) \approx 70.5$ degrees. There are four distinct orientations the triangles take, represented by the four colours. See Figure 1.2.

1.2.3 Hexagonal Bamboozle Structure

The hexagonal bamboozle structure can be seen as a triangular bamboozle structure combined with its mirror image. This is first done around a vertex, where the triangle (and also the structure it is connected to) is reflected to create a regular hexagon. Since the bamboozle structure needs to be point uniform, the other triangles are also changed into regular hexagons. This leaves a structure of regular hexagons meeting pairwise at an angles of $\arccos(\frac{1}{3})$. There are four distinct orientations the triangles take, represented by the four colours. See Figure 1.3.

1.2.4 The Rectangular Bamboozle Structures

There are two rectangular bamboozle structures. Both use rectangles with edge ratios $\sqrt{2} : 1$. Both have their rectangles meeting pairwise at angles of 60 degrees. The difference between is created by the relative orientation of neighbours of a given rectangle. Given a rectangle, and a neighbour at one of the four corners of the rectangle. If we look at the diagonally opposite corner, we have two ways to place a rectangle so that it meets the original rectangle at an angle of 60 degrees. In the four coloured rectangular bamboozle structure, such opposites are placed parallel. Whereas in the six coloured rectangular bamboozle structure, they are not. See Figure 1.4.

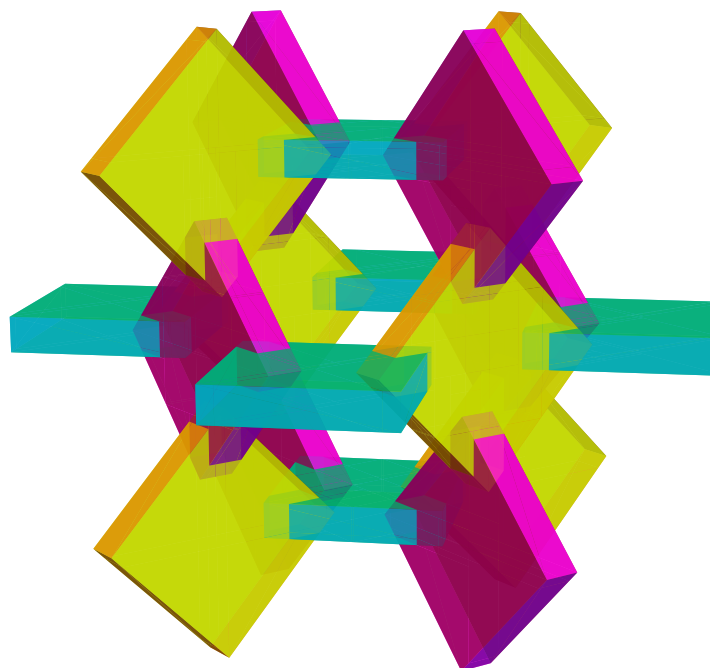


Figure 1.1: Square Bamboozle Structure

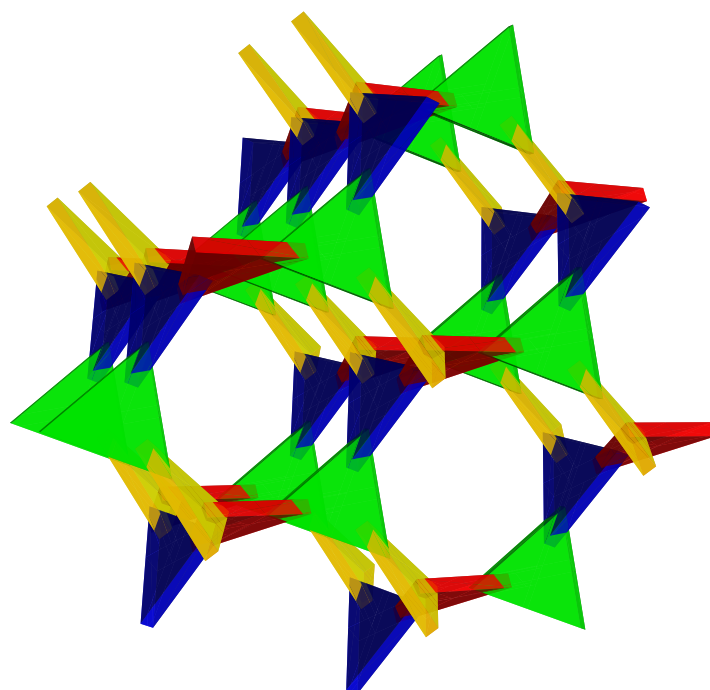


Figure 1.2: Triangular Bamboozle Structure

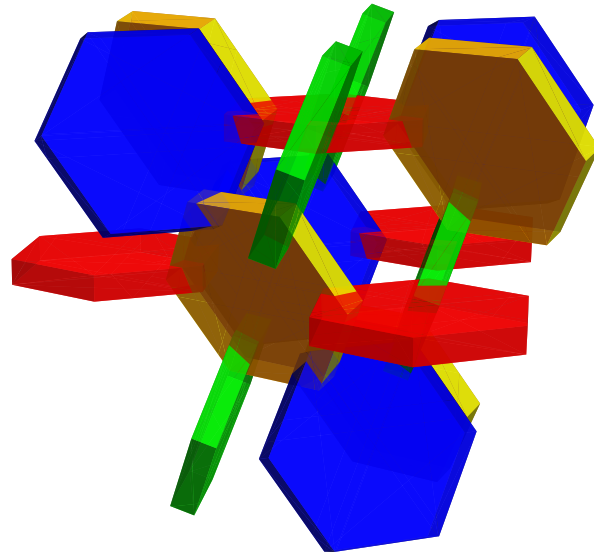


Figure 1.3: Hexagonal Bamboozle Structure

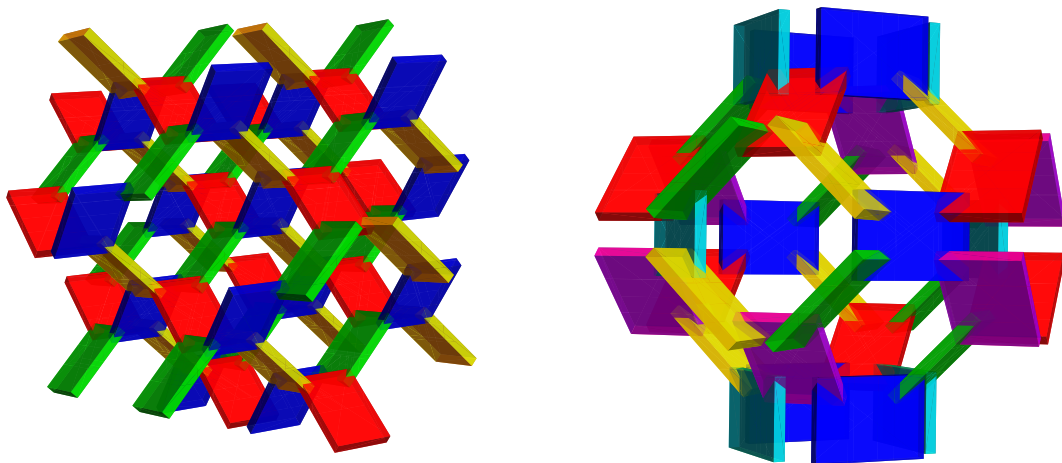


Figure 1.4: Four and Six Coloured Rectangular Bamboozle Structure

1.3 Goal

The goal of this thesis is to try conclude something about the completeness of this given list of bamboozle structures. Though it is conjectured to be complete, there is no prove for this conjecture yet. In order to conclude anything about bamboozle structures, we will also look into underlying structures. This might also lead to finding new requirements bamboozle structures must adhere to, which could therefore narrow the search for new structures, or help in a proof of completeness. Lastly, focussing on underlying structures might also lead to a proof of completeness under certain assumptions, i.e., completeness of bamboozle structures that have a certain quality.

Chapter 2

Shortest Distance Graphs in \mathbb{R}^2

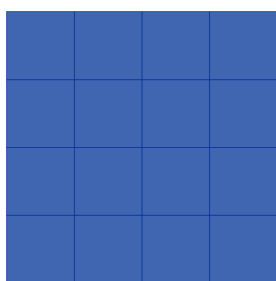
Though this thesis is aimed at understanding bamboozle structures within three-dimensional Euclidean space, it might be valuable to study the requirements of bamboozle structures in two-dimensional Euclidean space. Consider \mathbb{R}^2 equipped with the standard inner product and corresponding distance function $d()$, with \mathbb{K} thus a subset of \mathbb{R}^2 . We consider only the cases for which \mathbb{K} is generated by a single orbit under one of the 17 wallpaper groups[4], which are the \mathbb{R}^2 equivalent of the space groups. The goal in this chapter is to find all subsets $\mathbb{K} \subset \mathbb{R}^2$ generated in this fashion for which the minimal distance graph is connected.

Definition 13. *Within a tiling, we define the vertex type of a vertex around which there are, in a cyclic order, an a -gon, a b -gon, a c -gon, etc. as $a.b.c\dots$ [4]*

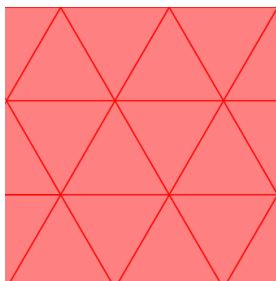
Definition 14. *A uniform convex tiling is a tiling of \mathbb{R}^2 with regular polygons for which all vertices are of the same vertex type.*

2.1 The 11 Uniform Convex Tilings

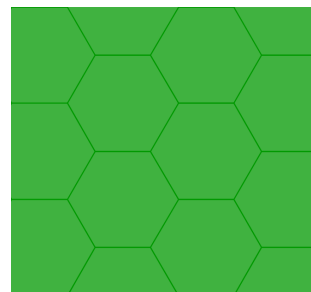
There are exactly 11 uniform convex tilings [4]. Three of these tilings are completely regular, meaning that not only are they vertex and edge transitive, they are also face transitive, and thus contain only one type of regular polygon. They are shown in Figure 2.1. This leaves 8 semi-regular tilings, which are not fully face transitive. They are shown in Figure 2.2.



The Square Tiling

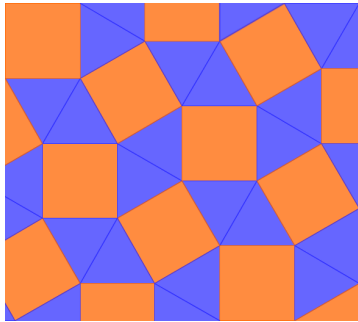


The Triangular Tiling

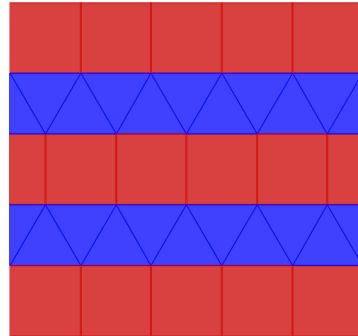


The Hexagonal Tiling

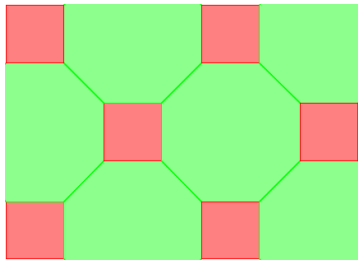
Figure 2.1: The three regular uniform convex tilings.



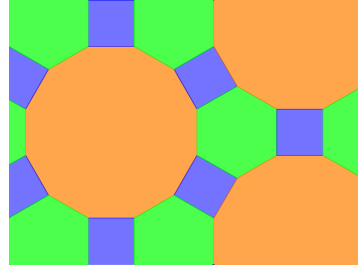
The Snub Square Tiling



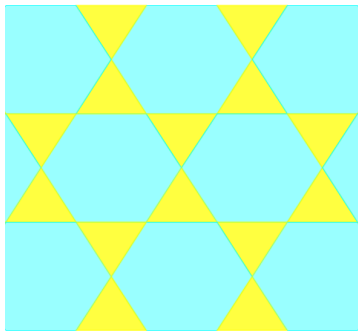
The Elongated Triangular Tiling



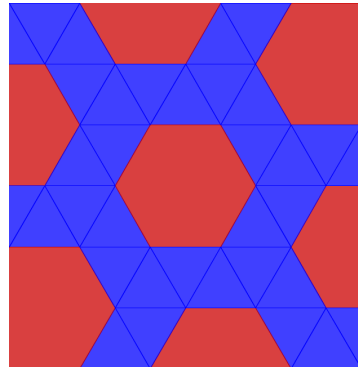
The Truncated Square Tiling



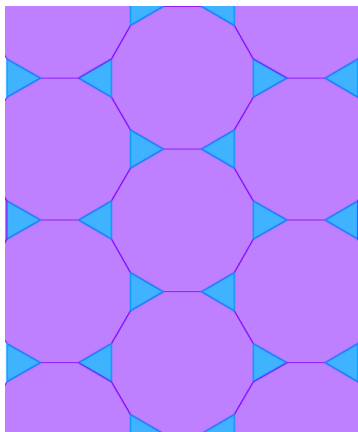
The Truncated Trihexagonal Tiling



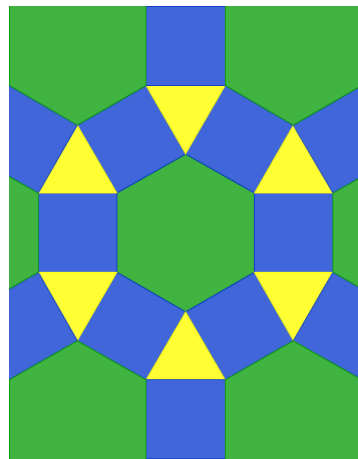
The Trihexagonal Tiling



The Snub Trihexagonal Tiling



The Truncated Hexagonal Tiling



The Rhombitrihexagonal Tiling

Figure 2.2: The eight semi-regular uniform convex tilings.

2.2 Skew Representations

Definition 15. *A polygon is equilateral if all edges have the same length.*

Definition 16. *A polygon is isogonal if given two vertices, there exist a symmetry that will move one vertex to the other. A polygon is m -isogonal if it contains a minimum of m classes of vertices so that all vertices in a single class are isogonal.*

This allows us to introduce the following new concept:

Definition 17. *A skew representation of a uniform convex tiling allows for distortion of n -gons that occur m times at every vertex under the following requirements:*

1. *The regular n -gons can be distorted into m -isogonal equilateral n -gons*
2. *The internal angle α of these distorted n -gons is bounded by $60^\circ < \alpha < 240^\circ$*
3. *The distorted tiling remains point uniform*

The limits for α are based on the fact that an angle less than or equal to 60° creates an isosceles triangle with a third edge length that is either equal to, or less than d_{min} . An internal angle greater than or equal to 240° creates an external angle of less than or equal to 60° , for which the same applies.

Lemma 1. *Skew representations of uniform convex tilings can also be generated by a single orbit under a wallpaper group.*

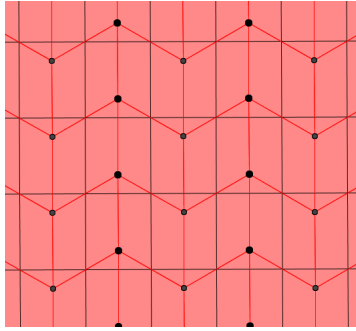
This can be concluded due to the point uniformity of the skew representations, combined with the fact that they span \mathbb{R}^2 . Some examples of skew representations and their corresponding wallpaper groups can be seen in Figure 2.3.

Theorem 1. *The list of all subsets $\mathbb{K} \subset \mathbb{R}^2$ generated by a wallpaper group for which the minimal distance graph is connected is a list of skew representations of the list of all uniform convex tilings of \mathbb{R}^2 .*

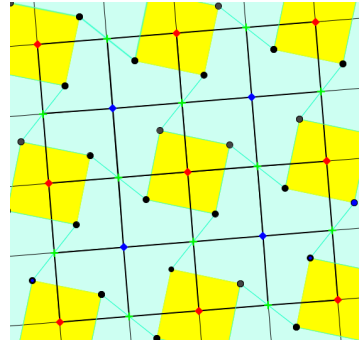
Proof. There are exactly 11 convex uniform tilings. These 11 and their skew representations follow symmetries of the wallpaper groups. Since only equilateral polygons are used, the edges between vertices all have the same length, and there will not be two vertices that could be connected by a shorter edge, since the internal angles of the distorted polygons is bounded. Therefore, taking the vertices of any convex uniform tiling or skew representation creates a subset $\mathbb{K} \subset \mathbb{R}^2$ generated by a wallpaper group for which the minimal distance graph is connected.

Conversely, given a subset $\mathbb{K} \subset \mathbb{R}^2$ generated by a wallpaper group for which the minimal distance graph is connected, we know that its minimal distance graph will form equilateral polygons. If not, every vertex of the graph is of rank two, but then it can never be connected due to the symmetrical properties of the wallpaper groups. This means that we can give this graph a vertex type in the same way as the convex uniform tilings.

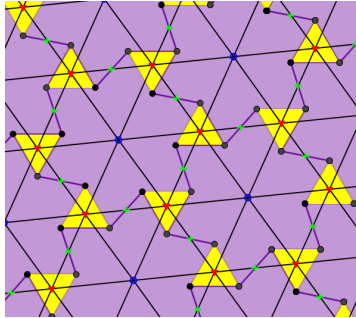
If one of the polygons contained in the graph has m different angles between consecutive edges, we know that all these m angles must be present at every vertex. This is because we only put a single point within a cell, and rotation, translation and (glide-)reflection do not change the relative angles of outgoing edges at a vertex. Furthermore, we know the relative



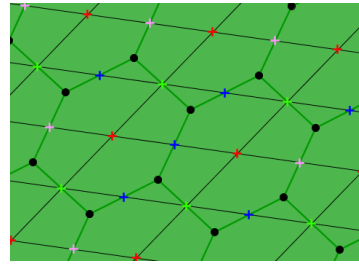
4.4.4.4 graph, located in group pg



4.8.8 graph, located in group p4



3.12.12 graph, located in group p6



6.6.6 graph, located in p2

Figure 2.3: Examples of skew representations of tilings, and their wallpaper group.

occurrence frequency of these angles within polygons is the same as the relative occurrence frequency within the vertices. This is because we know that relative occurrence frequency within the vertices is equal to the relative occurrence frequency taken over all vertices, since all vertices share the same vertex type, and because we know that the relative occurrence frequency within polygons is also equal to the relative occurrence frequency taken over all vertices, since inequality would require vertices with different relative occurrence frequencies. If the polygon has n -vertices, we thus know that $m \mid n$. If σ is the sum over these m different angles, we know that $\sigma \cdot \frac{n}{m} = n \cdot 180^\circ - 360^\circ$, i.e. $\sigma = m \cdot (180^\circ - \frac{360^\circ}{n})$. This means that the total angle used for this polygon per vertex is only dependent on the type of polygon and how often it is present at the vertex, and not on the different angles this polygon contains. Therefore, we know the desired minimal distance graph can only have one of the 21 vertex types defined by Grünbaum [4].

Of course, not every type translates to a convex uniform tiling. Therefore, it might still be the case that the minimal distance graphs allow a unique vertex type. For this to be the case, the vertex type would have to have multiple occurrences of an n -gon for $n > 3$ (Since an equilateral triangle is always a regular polygon). This leaves the types 3.3.6.6, 3.4.4.6, and 5.5.10. For 5.5.10 it holds that the pentagons have to be equiangular, since $2 \nmid 5$. So this leaves two options which are not represented by a uniform convex tiling. However, both these options quickly turn out not to work when drawing them out, because they lose point uniformity.

The symmetry of the wallpaper groups require point uniformity. This trivially leads to the equilateral and m -isogonal properties of the polygons used in the tilings. Therefore the list of all subsets $\mathbb{K} \subset \mathbb{R}^2$ generated by a wallpaper group for which the minimal distance graph is

connected is a list of skew representations of the list of all uniform convex tilings of \mathbb{R}^2 . \square

We can now conclude that every shortest distance graph generated by the orbit of a single point under a wallpaper group can be represented by a uniform convex tiling.

Chapter 3

Bamboozles in \mathbb{R}^3

3.1 Scaling up from \mathbb{R}^2

In \mathbb{R}^2 we saw that shortest distance graphs generated by the orbit of a single point under a wallpaper group were limited to tilings that could be represented by uniform convex tilings. In \mathbb{R}^3 , we want to look at shortest distance graphs generated by the trajectory of a single point under a space group. It could thus be very useful to look at the three dimensional uniform convex tessellations, i.e. the convex uniform honeycombs. For this we introduce the following concepts:

Definition 18. *A polyhedron is a three dimensional solid, which has straight edges and polygonal faces.*

Definition 19. *A polyhedron is vertex-transitive if and only if for any given vertices u and v , the polyhedron has some automorphism $f()$ so that $f(u) = v$.*

Definition 20. *A polyhedron is called uniform if and only if its faces are regular polygons, and the polyhedron is vertex-transitive.*

Definition 21. *A tessellation is (point-)uniform if and only if for any given vertices a and b , the tessellation has some automorphism $f()$ so that $f(a) = b$.*

This leads to the following definition of a convex uniform honeycomb:

Definition 22. *A Convex Uniform Honeycomb is a (point-)uniform tessellation of three dimensional Euclidean space consisting of non-overlapping convex uniform polyhedra*

There are 28 of these Convex Uniform Honeycombs known [3]. These 28 could possibly be used to find bamboozle structures. If we look at the vertices within these honeycombs, we see that the minimal distance graph is connected and the affine span of its edges is \mathbb{R}^3 , and that the vertices form an orbit of a single vertex within a space group. However, the neighbours of a given vertex within the minimal distance graph do not lie in a plane. Therefore, we must think of a method to find bamboozles contained within honeycombs.

For instance, we might look at vertex v within a honeycomb, and find a subset U of the neighbours of v , so that the vertices $u \in U$ and v lie in a plane. Since all vertices share the same vertex type, we can select a symmetrical subset of neighbours for all $u \in U$ (that includes v). We can repeat this process until we have a sub-graph of the minimal distance

graph that does have the property that all neighbours of each vertex lie within a plane. It could then be the case that the vertices we excluded from the subsets of the neighbours no longer occur in the sub-graph, so that the minimal distance graph of this sub-graph is the sub-graph itself. If this is the case, the new sub-graph meets all requirements for a bamboozle structure. If this is not the case, it technically does not meet the requirements.

There are some issues with this method. The first one has to do with selecting the particular subset of neighbours. If a certain vertex-type has more than three neighbours contained within a plane, then it might be possible to create a bamboozle structure without including all these neighbours, but instead using only a subset of available neighbours. Since we want to limit the choices that need to be made whilst constructing these bamboozle structures, we adopt the following rule: Given a vertex, a neighbour that has to be included, and an orientation for the plane, there should only be one possible polygon. It is quite possible that adding other rules for the placement of polygons could result in more bamboozle structures, consisting of polygons we do not consider. However, this limitation makes the existence of a bamboozle in a given honeycomb almost completely dependent of the vertex-type of that particular honeycomb.

Another important issue is to do with parallel neighbouring polygons. Initially, allowing neighbouring polygons to be parallel goes against the criteria for bamboozle structures, but it could also be useful, as can be seen in some of the examples. As long as not all the neighbours of a particular polygon are parallel to that polygon, a structure similar to a bamboozle structure could still be achieved.

The structures that fail to meet all requirements since some neighbouring polygons are parallel, or the fact that their sub-graph is not a minimal distance graph, shall be referred to as near-bamboozle structures, in contrast to the true bamboozle structures that fulfil all requirements.

3.2 Results of This Method

We will now discuss the eleven honeycombs for which this method gives a non trivial result. Ten of the eleven prismatic honeycombs are left out of consideration, because they all require square polygons to reach between layers of two dimensional tilings, which would therefore all create trivially similar results. Furthermore, there are seven honeycombs that do not allow a polygon fitted in a vertex type. These honeycombs are thus also not listed. That leaves the following eleven honeycombs:



Figure 3.1: Vertex type of the Cubic Honeycomb, and a fitting polygon

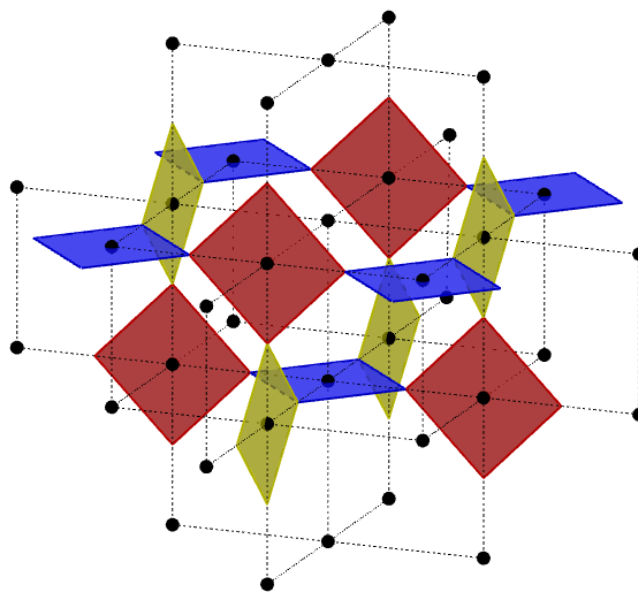


Figure 3.2: Cubic Bamboozle, within the Cubic Honeycomb

3.2.1 Cubic Honeycomb

The cubic honeycomb has eight cubes put together at every vertex. Its vertex type can be seen in Figure 3.1. It is clear that every vertex allows three planes in which sub-groups of neighbours lie. These three planes are perpendicular to the edges from v to one of the pair of neighbours that are not contained in the subset. Within these planes there lie four neighbours. This gives us the choice to leave one of these neighbours out of the subset.

If we choose to use all four neighbours, and choose non parallel planes to choose the subset of the neighbours' neighbours, we end up with the cubic bamboozle that was already discovered by Koos Verhoeff, depicted in Figure 3.2. Since the vertices that were left out of the subsets, do not occur in later subsets, this structure fits all requirements to be a bamboozle structure.



Figure 3.3: Vertex type of the Truncated Cubic Honeycomb, and a fitting polygon

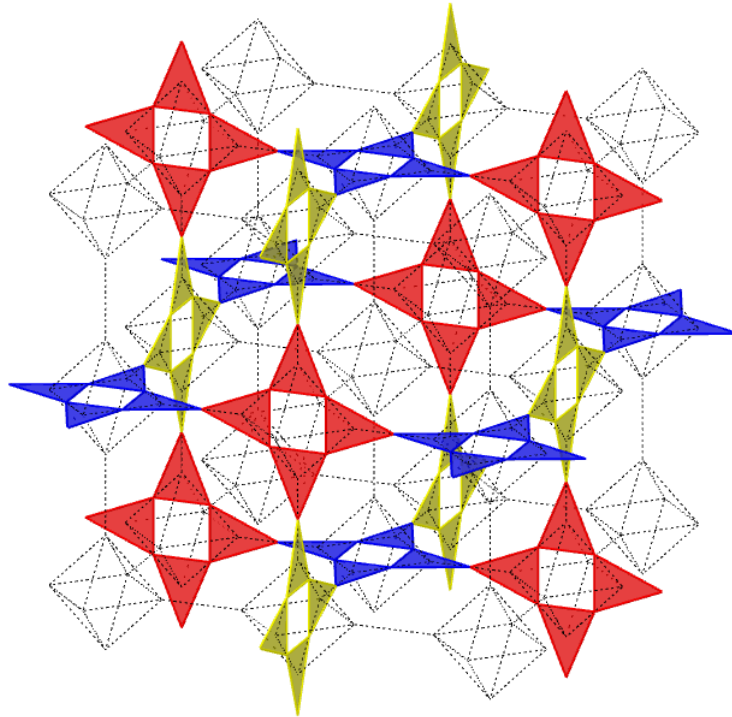


Figure 3.4: Triangles within the Truncated Cubic Honeycomb

3.2.2 Truncated Cubic Honeycomb

The truncated cubic honeycomb consists of truncated cubes, and octahedra filling up the truncated space. Every vertex contains one octahedron, and four truncated cubes. If we take the vertex type described in figure 3.3, we can expand the polygons across the rest of honeycomb. However, in two of the three directions contained within the triangles, we are forced to place a polygon that is parallel to the original polygon. It is also noteworthy that the complete bamboozle structure, depicted in Figure 3.4, looks quite similar to the cubic bamboozle structure, depicted in Figure 3.2. This is caused by the parallelism of neighbouring polygons combined with the fact that the honeycombs are quite similar.

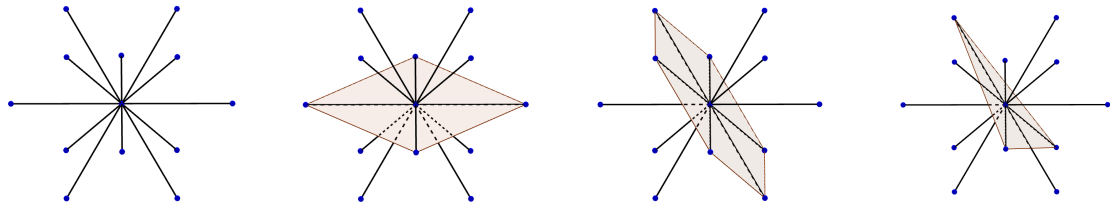


Figure 3.5: Vertex type of the Alternated Cubic Honeycomb, and fitting polygons

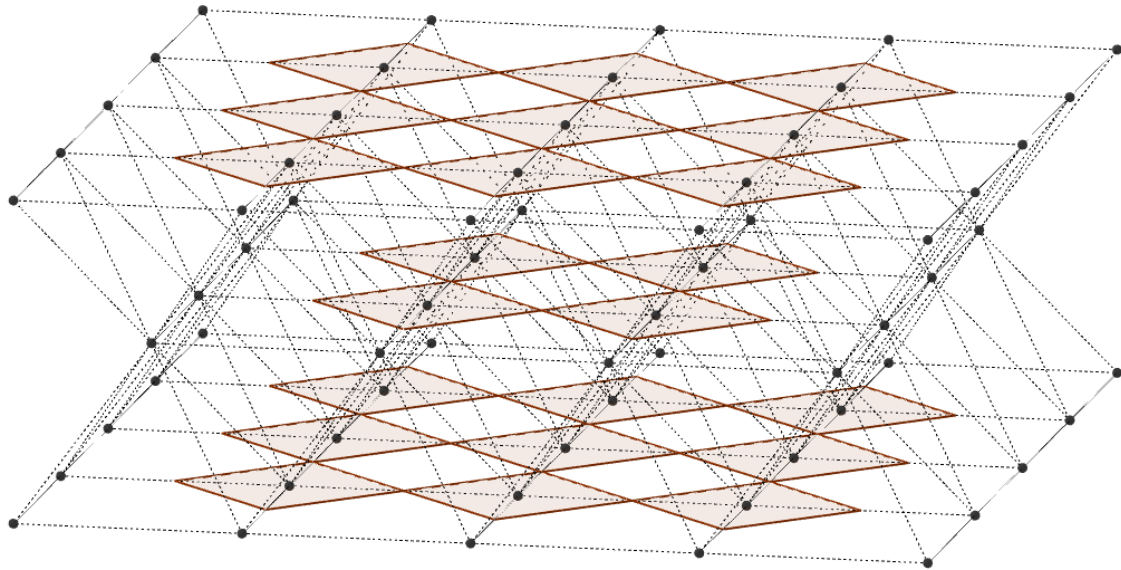


Figure 3.6: Squares within Alternated Cubic Honeycomb

3.2.3 Alternated Cubic Honeycomb

The alternated cubic honeycomb has eight tetrahedra and 6 octahedra put together at every vertex. In Figure 3.5 we can see this vertex type, and the polygons that fit inside. However, not all three polygons can be used to form bamboozle structures. If we choose the square, all the polygons need to be parallel as can be seen in Figure 3.6, and this parallelism violates the connectivity requirement. In Figure 3.7, we can see the placement of hexagons within the alternated cubic honeycomb. Note that it does not fulfil all the requirements to be a bamboozle structure, since the sub-graph contains all vertices, and is thus not a minimal distance graph. In Figure 3.8, we can see the placement of triangles within the alternated cubic honeycomb, which does fulfil all the requirements, and can be recognized as the Bamboozle, that hangs near the elevators of the Metaforum building.

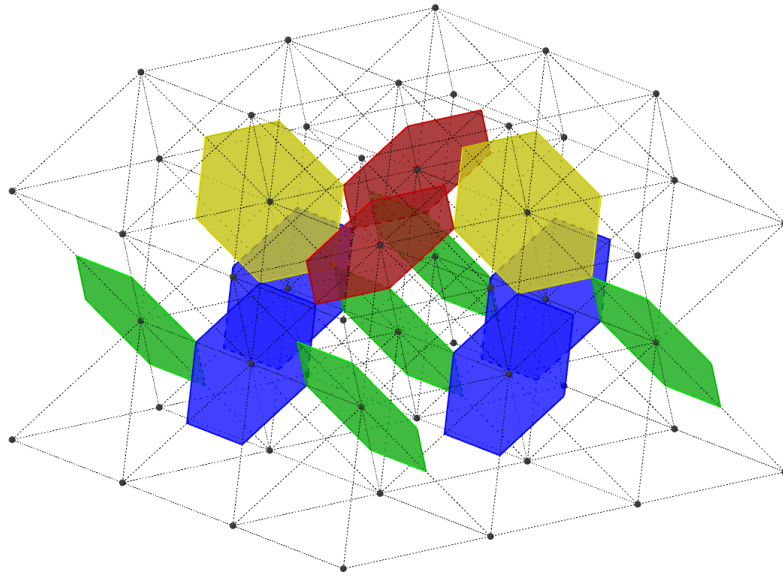


Figure 3.7: Hexagons within Alternated Cubic Honeycomb

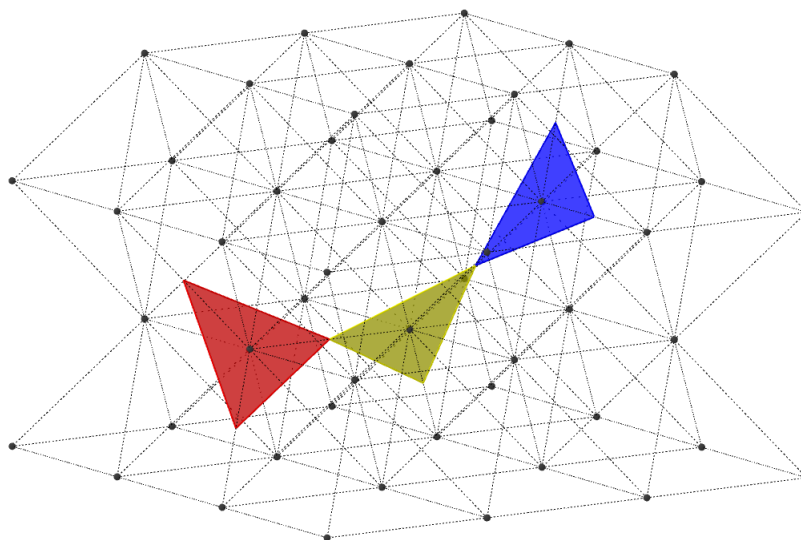


Figure 3.8: Triangles within the Alternated Cubic Honeycomb



Figure 3.9: Vertex type of the Cantellated Cubic Honeycomb, and a fitting polygon

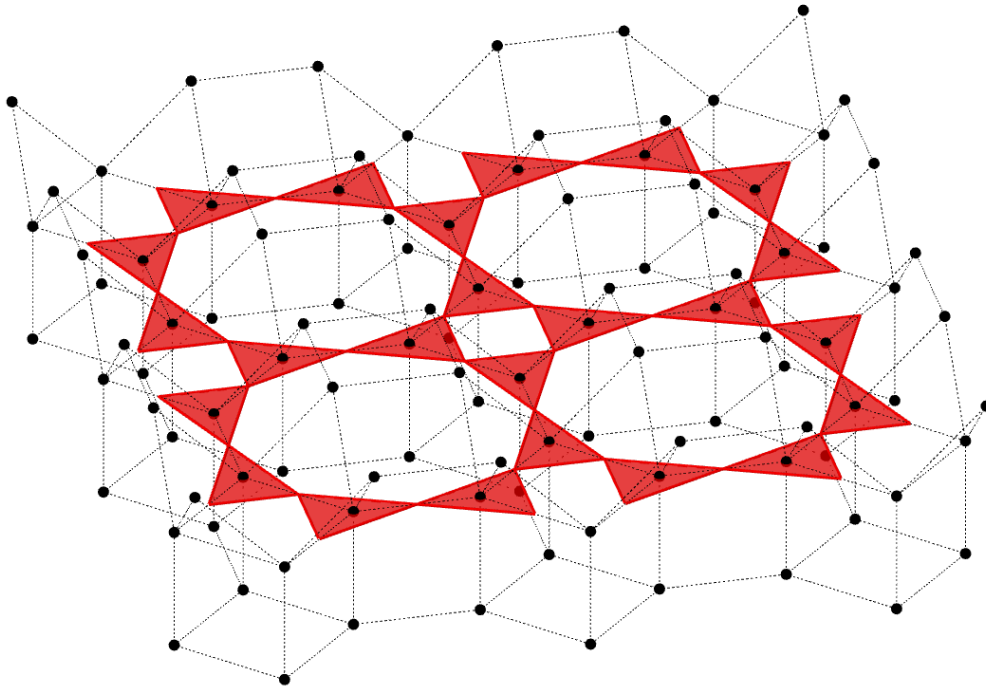


Figure 3.10: Triangles within the Cantellated Cubic Honeycomb

3.2.4 Cantellated Cubic Honeycomb

The cantellated cubic honeycomb has two cubes, one cuboctahedron, and two rhombicuboctahedra to a vertex. The vertex type can be seen in Figure 3.9. Although the vertex type allows multiple orientations of the fitting triangle, when we expand the bamboozle structure we find that all the polygons need to be parallel. This means that all connected polygons lie within a plane, and therefore a structure that spans across \mathbb{R}^3 cannot be connected. The structure is shown in Figure 3.10



Figure 3.11: Vertex type of the Quarter Cubic Honeycomb, and a fitting polygon

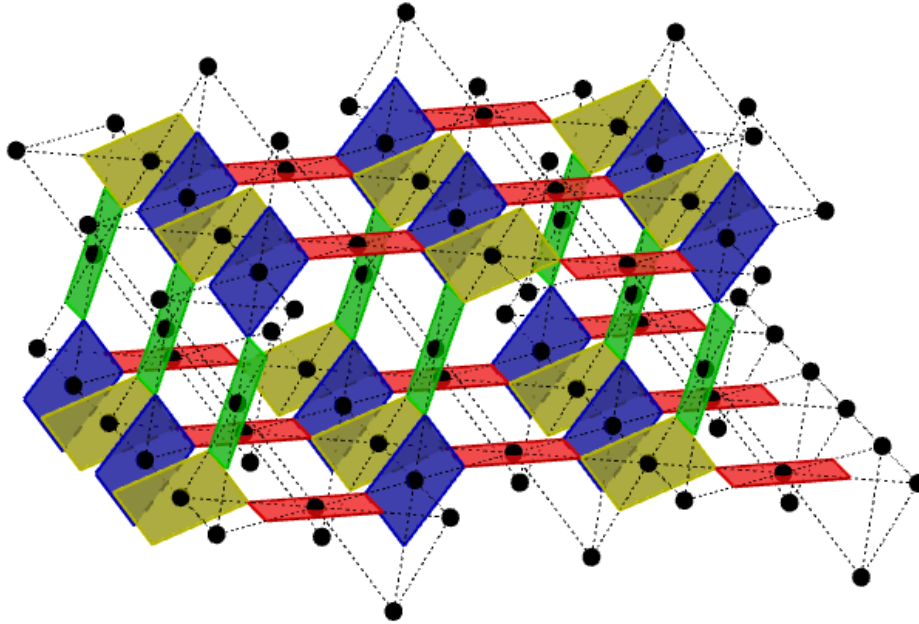


Figure 3.12: Rectangles within the Quarter Cubic Honeycomb

3.2.5 Quarter Cubic Honeycomb

The quarter cubic honeycomb has six truncated tetrahedra and 2 tetrahedra to a vertex. The vertex type is shown in Figure 3.11. In Figure 3.12, we can see the bamboozle structure within this honeycomb. Neighbours that were not included in the sub-graph do eventually become part of the sub-graph. This can be concluded from the honeycomb edges that connect two polygons, but that are not contained within these polygons. Thus the sub-graph is not a minimal-distance graph, so this bamboozle structure does not fit all requirements. However, it is important to note that this bamboozle structure is quite similar to the 4-rectangle bamboozle structure by Tom Verhoeff. The main difference is the ratios of the rectangles. In Verhoeff's structure, rectangles have the ratio $\sqrt{2} : 1$, whereas these rectangles have the ratio $\sqrt{3} : 1$. Furthermore, this bamboozle structure looks slightly crooked compared to the bamboozle structure of Tom and Koos Verhoeff. There is likely to be a link between the two differences, and the fact that Verhoeff's bamboozle structure does meet all the requirements. This shall be discussed in more detail in Chapter 4.



Figure 3.13: Vertex type of the Rectified Cubic Honeycomb, and a fitting polygon

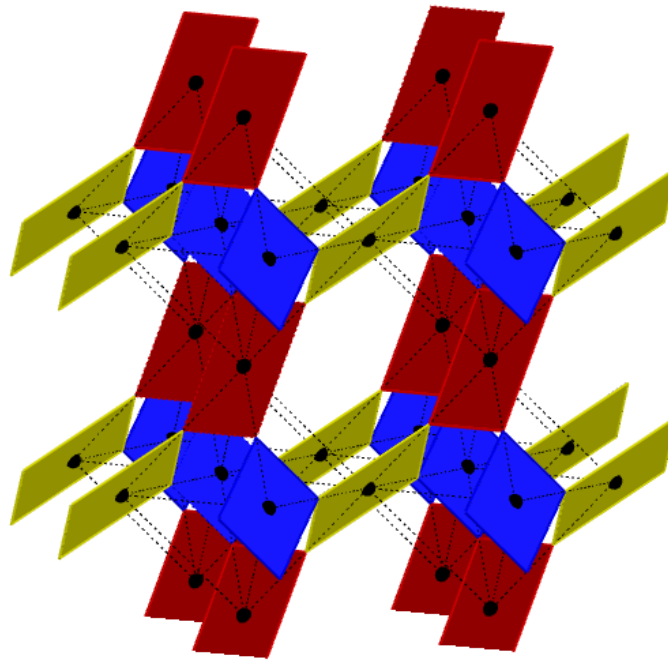


Figure 3.14: Rectangles within the Rectified Cubic Honeycomb

3.2.6 Rectified Cubic Honeycomb

The rectified cubic honeycomb has four cuboctahedra and one octahedron to a vertex. The vertex type can be seen in Figure 3.13. In Figure 3.14, we can see the bamboozle structure within this honeycomb. The sub-graph of the bamboozle structure is not a minimal distance graph, meaning the bamboozle structure does not fit all the requirements. Like the bamboozle structure contained within the quarter cubic honeycomb, the structure appears to be crooked, although it is unclear what structure it is based on. The only candidate for this would be the square bamboozle structure, and in Chapter 4 we discuss this possibility.



Figure 3.15: Vertex type of the Runcic Cubic Honeycomb, and a fitting polygon

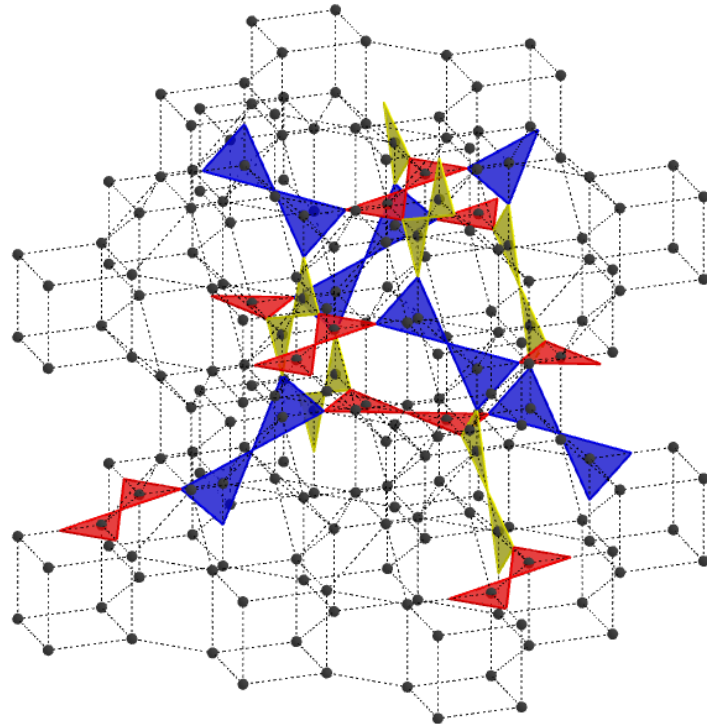


Figure 3.16: Triangles within the Runcic Cubic Honeycomb

3.2.7 Runcic Cubic Honeycomb

The runcic cubic honeycomb has four rhombicuboctahedra, one cube, and one tetrahedron to a vertex. Its vertex type is shown in Figure 3.15. In Figure 3.16, we can see the bamboozle structure within this honeycomb. The subgraph is a minimal distance graph, but some neighbouring polygons are forced to be parallel. Therefore, it does not fulfil all requirements. However, it is a near-bamboozle structure.



Figure 3.17: Vertex type of the Runcitruncated Cubic Honeycomb, and a fitting polygon

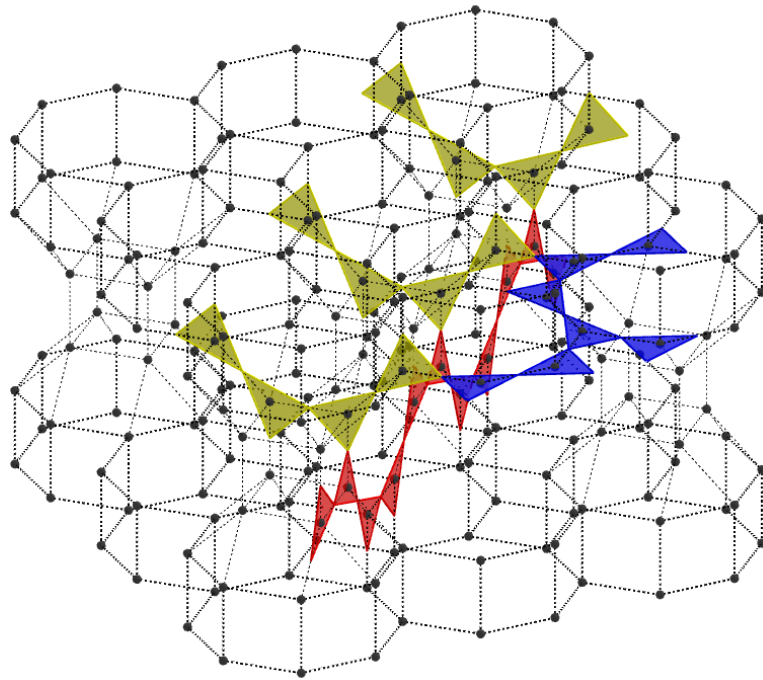


Figure 3.18: Triangles within the Runcitruncated Cubic Honeycomb. Whilst developing this structure, one of the blue triangles was forced to have three parallel neighbours.

3.2.8 Runcitruncated Cubic Honeycomb

The runcitruncated cubic honeycomb has two octogonal prisms, one rhombioctahedron, one cube, and one truncated cube to a vertex. The vertex type is shown in 3.17. While developing this vertex type into a bamboozle structure, we came across a polygon that needed three neighbours that were parallel. This meant the the structure was no longer point-uniform, and could therefore not be a bamboozle structure.

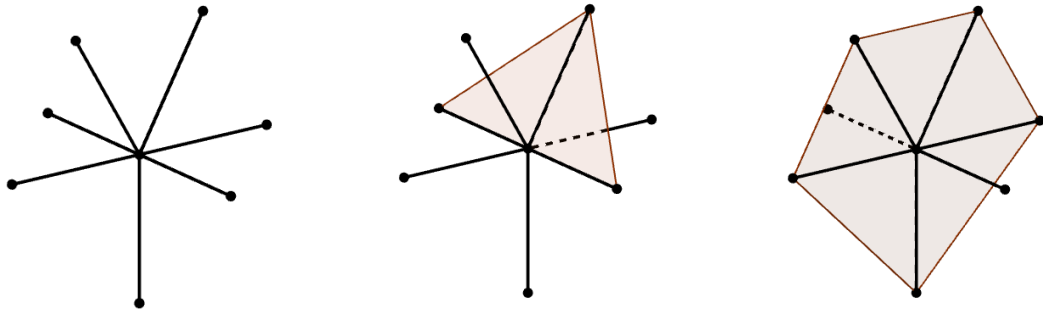


Figure 3.19: Vertex type of the Gyroelongated Triangular Prismatic Honeycomb, and a fitting polygon

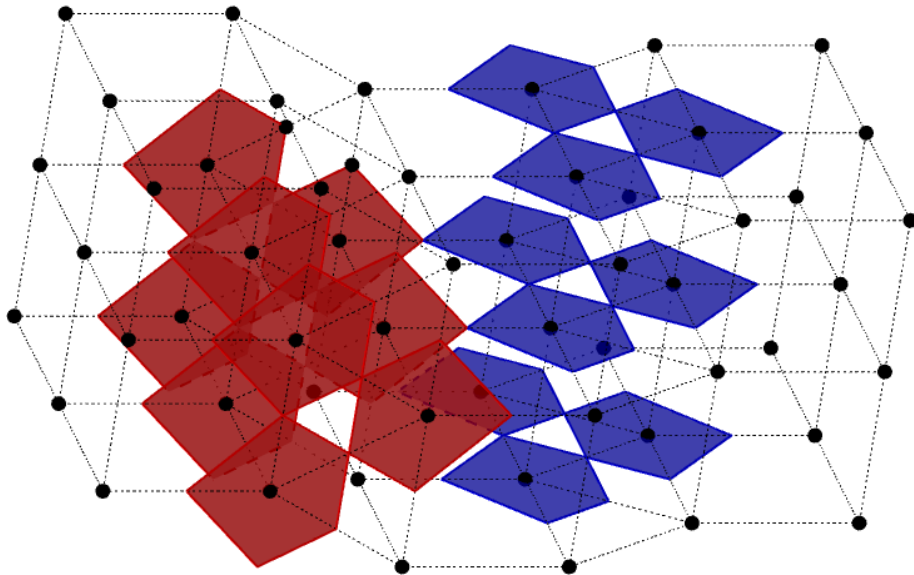


Figure 3.20: Pentagons within the Gyroelongated Triangular Prismatic Honeycomb

3.2.9 Gyroelongated Triangular Prismatic Honeycomb

The gyroelongated triangular prismatic honeycomb has six triangular prisms, and four cubes to a vertex. In Figure 3.19 the vertex type is shown. In Figure 3.20, the bamboozle structure within this honeycomb is shown. The subgraph is not a minimal distance graph, and only one out of five neighbours is not parallel to a given polygon. Therefore the bamboozle structure does not fulfil all requirements, but could be referred to as a near-bamboozle structure.

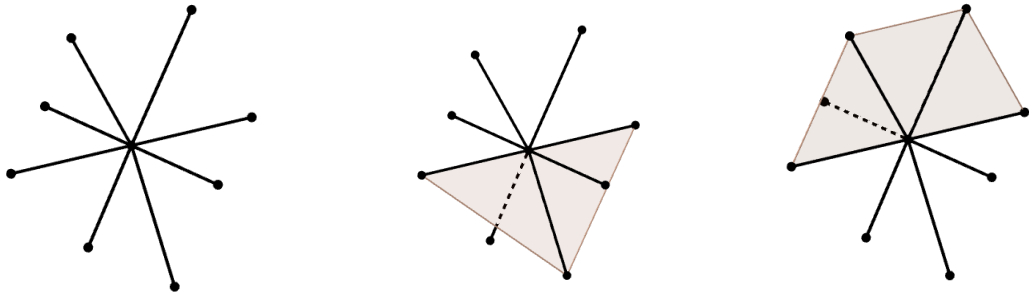


Figure 3.21: Vertex type of the Gyrate Triangular Prismatic Honeycomb, and fitting polygons

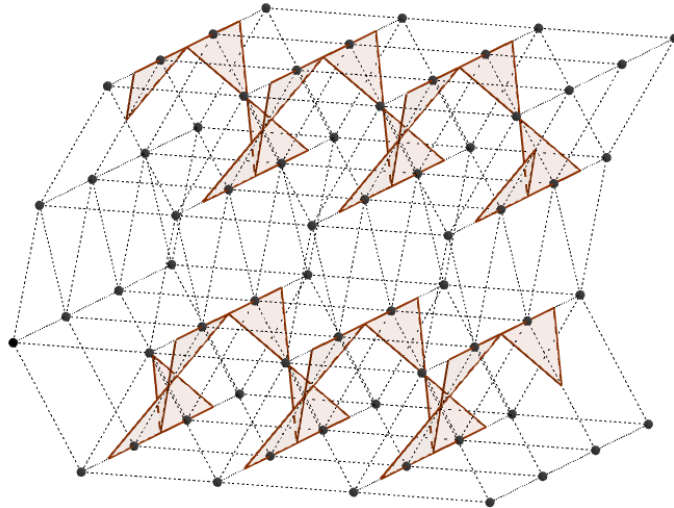


Figure 3.22: Triangles within the Gyrate Triangular Prismatic Honeycomb

3.2.10 Gyrate Triangular Prismatic Honeycomb

The gyrate triangular prismatic honeycomb has twelve triangular prisms to a vertex. The vertex type is shown in 3.21. In Figures 3.22 and 3.23 we can see the structures within the honeycomb. However, neither of the two fulfil the connectivity requirement, and can thus not be bamboozle structures. Furthermore, both structures use a subgraph that is not a minimal distance graph, and the structure in Figure 3.23 has polygons that are all parallel.

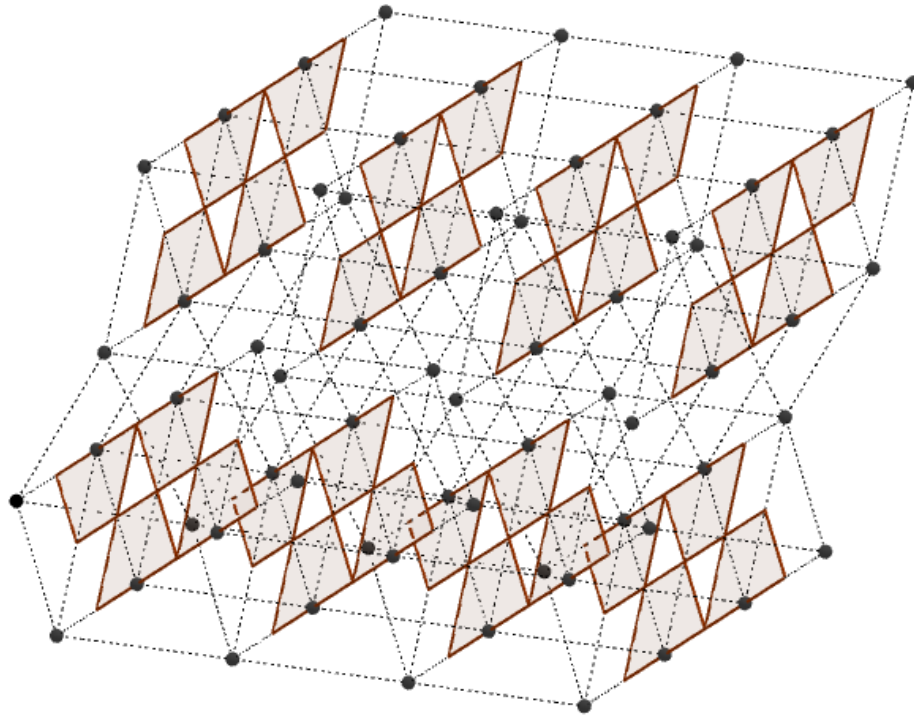


Figure 3.23: Trapezoids within the Gyroelongated Triangular Prismatic Honeycomb

3.2.11 Gyroelongated Alternated Cubic Honeycomb

The gyroelongated alternated cubic honeycomb has six triangular prisms, four tetrahedra, and three octahedra to a vertex. Its vertex type is shown in Figure 3.24. However, the vertex type does not allow expansion along the entire honeycomb; If the edge that is shared by the triangular prisms is used for a polygon, there is no way to determine the orientation of the neighbouring polygon in that direction. See Figure 3.25. Since the triangular prisms lie in plane formation, at least one of these edges is required to fulfil the connectivity requirement. Therefore, there could not exist a bamboozle structure within this honeycomb, unless an alternate system is used to determine the orientation of these polygons, which is out of our consideration.



Figure 3.24: Vertex type of the Gyroelongated Alternated Cubic Honeycomb

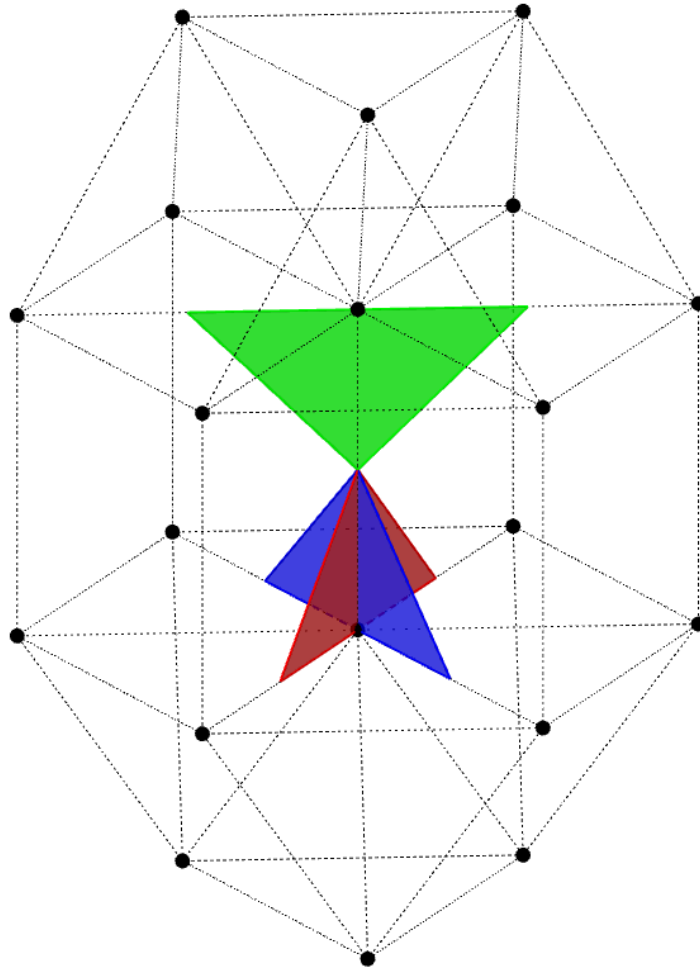


Figure 3.25: Triangles within the Gyroelongated Alternated Cubic Honeycomb. This figure depicts two ways to link the triangles within the honeycomb, and that it is not possible to determine which of these to use.

3.2.12 Conclusion

The method has proven to be fruitful. Though it has found only two bamboozle structures which fit all the requirements, both of which were already known, it has produced lots of results that fulfil nearly all requirements. Furthermore, we have seen that the hexagonal bamboozle structure created by Tom and Koos Verhoeff does not fulfil all requirements, namely, the graph between used vertices is not a minimal distance graph. Therefore, all results that share the same qualities as the hexagonal bamboozle structure should be considered interesting enough for further investigation.

The structures that failed to fulfil the requirement that neighbouring polygons should not be parallel require more consideration. Though they might not receive true bamboozle structure status, they should too be deemed sufficiently interesting. One quality that is worth investigating is their relationship to known bamboozle structures. Since most honeycombs are built as variation on the cubic honeycomb, it is interesting to see what effect these variations have on the square bamboozle structure contained within this honeycomb, as can be seen with the truncated cubic honeycomb. Furthermore, the bamboozle structure contained within the gyroelongated triangular prismatic honeycomb can also be linked to a different method of structuring squares within the cubic honeycomb.

Lastly, it is important to note that this method did not produce a complete list of bamboozle structures. This is clear since the 6 coloured rectangular bamboozle structure found by Tom Verhoeff did not appear using this method. It might therefore be interesting to see if an underlining structure can be found for this bamboozle structure, and find out why it did not appear in the list of honeycombs. An important note on this is that there are point uniform tessellations of \mathbb{R}^3 that consist of solids that are not all point uniform, and which therefore do not appear in the list of convex uniform honeycombs.

3.3 Completeness of Honeycombs

In order to validate a proof of completeness for bamboozle structures based on honeycombs, a complete list of honeycombs is required. The list of honeycombs we use (based on [3]) consists of 28 unique honeycomb structures. However, as of 2006 the completeness of this list was yet to be proven, though it is conjectured to be complete.

Theorem 2. *The list of 28 Convex Uniform Honeycombs is complete.*

We want to prove this theorem. In order to do so, we first need a list of multisets of convex uniform polyhedra that fit together at a vertex, and see which of these could tessellate the full Euclidean space. The convex uniform polyhedra can be classified in to four different groups: *platonic solids*, *archimedean solids*, *prisms*, and *anti-prisms*.

Definition 23. *A uniform polyhedron is a platonic solid if and only if all its faces are congruent regular polygons.*

Definition 24. *A uniform polyhedron is a prism if and only if every vertex is of vertex type 4.4.n, with $n > 2$.¹*

Definition 25. *A uniform polyhedron is an anti-prism if and only if every vertex is of vertex type 3.3.3.n, with $n > 2$.²*

¹A cube is both a platonic solid and a prism.

²A dodecahedron is both a platonic solid and an anti-prism.

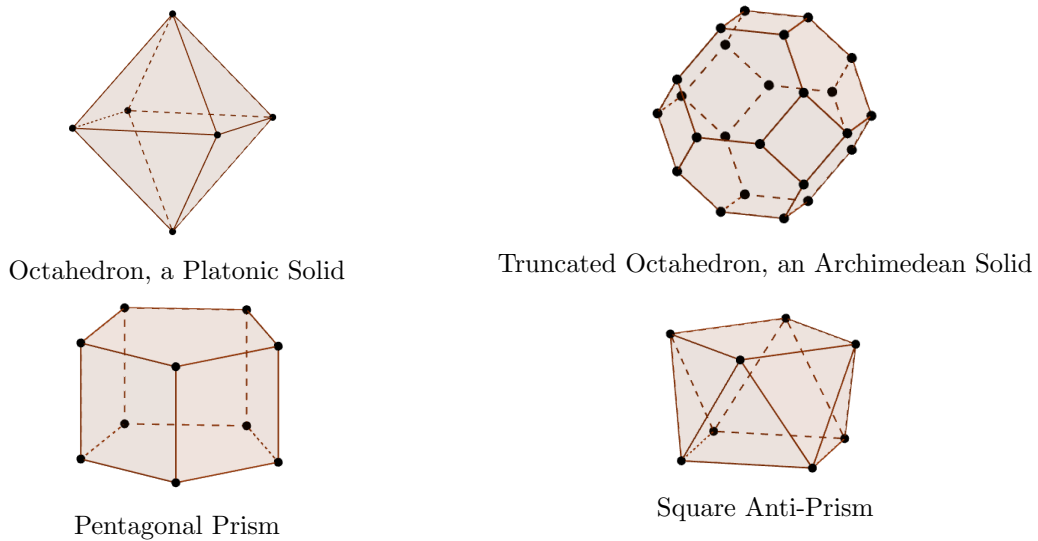


Figure 3.26: Examples of the different types of Convex Uniform Polyhedra

Definition 26. *A uniform polyhedron is an archimedean solid if and only if it does not fit the descriptions of platonic solids, prisms, or anti-prisms.*

It is important to note that the definitions for prisms and anti-prisms allow any $n > 2$, which means that there is an infinite number of prisms and anti-prisms. This makes it impossible to run an analysis over all the convex uniform polyhedra. Therefore, we need to make a restriction for which prisms and anti-prisms we want to consider in our analysis, and justify this restriction by proving that the remaining polyhedra could never be used in a honeycomb.

If we look at honeycombs that consist solely of prisms and anti-prisms, we see that these are in fact just tessellations of two dimensional euclidean space stacked on top of each other. We know the number of two dimensional tessellations is 11. We also know that prisms and anti-prisms have the same n -gon on both sides. Therefore, we can conclude that one layer of the stack contains a particular tessellation of two dimensional euclidean space, the layer on top of it will have the same tessellation, and in general every layer will.

The largest polygon that can be used in a two dimensional uniform convex tessellation is the dodecagon, so we know that prisms and anti-prisms used in these honeycombs cannot contain polygons larger than the dodecagon. Furthermore, since the platonic and euclidean solids do not have faces larger than the decagon, we know we do not have to consider more than the first 10 prisms and anti-prisms (up to the dodecagonal prism and anti-prism). This is because the polygon faces of the other prisms and anti-prisms cannot connect to platonic or archimedean solids, or be used to form a two dimensional tessellation.

It is important to note that there is third way to combine the different types of polyhedra: One could make stacks of prisms or anti-prisms so that their n -gon sides touch, and then fill up the rest of the space with platonic solids, archimedean solids, prisms and anti-prisms in such a way that it is not a stack of a two dimensional tiling. This way there is no obvious limit on n . Since the existing lists of honeycombs does not include such honeycombs, such a honeycomb is hard to find, but non-existence would be difficult to prove, we assume that such honeycombs do not exist. However, it is important to remember this possibility.

Now that we have a list of polyhedra that could be used to form honeycombs, we need a way to see which of these polyhedra could fit together in a vertex. To do this we introduce the following concept:

Definition 27. *The solid angle of an object given an origin, expressed in steradians, is the surface of the projection of the object on a unit sphere centred at the origin*

This quickly leads to the following lemma:

Lemma 2. *The total amount of steradians at a single point is equal to the area of the surface of a unit sphere, i.e. 4π*

If we measure the solid angles of our polyhedra with the origin at a vertex, we can determine which of these polyhedra fit together by computing the combinations of solid angles that add up to 4π .

Luckily, there exist simple formulas to determine such solid angles, based on what polygons come together in a point. These formulas use the planar angles α_i . Since we use regular n -gons, these planar angles are equal to the internal angles of these n -gons: $\alpha_i = \frac{(n-2)\cdot\pi}{n}$. For three polygons to a point, the solid angle θ can be determined by: [1]

$$\cos\left(\frac{\theta}{2}\right) = \left[2 - \sum_i^3 \sin^2\left(\frac{\alpha_i}{2}\right)\right] / 2 \prod_i^3 \cos\left(\frac{\alpha_i}{2}\right)$$

For four polygons to a point, the formula becomes:

$$\cos\left(\frac{\theta}{2}\right) = \left[2 - \sum_i^4 \sin^2\left(\frac{\alpha_i}{2}\right) - 2 \prod_i^4 \sin\left(\frac{\alpha_i}{2}\right)\right] / 2 \prod_i^4 \cos\left(\frac{\alpha_i}{2}\right)$$

For five polygons to a point, the polyhedron can be split in such a way that the five polygons are split into two groups, which form a polyhedron with four polygons to that point, and a polyhedron with three polygons to a point. In both cases the extra polygon comes from the facet that was created by the split. Now, we can use the previous two formulas to determine the total solid angle, given that we can compute the planar angle for the new facet. There are no polyhedra with more than five polygons to a point contained within the group of convex uniform polyhedra.

This gave us Table 3.1, which is a list of all polyhedra we considered and their corresponding solid angle expressed in a numerical value. The polyhedra are named by the polygons that meet at vertices. This means that a tetrahedron, where three regular triangles meet in every vertex, is denoted by 3^3 , and a rhombicuboctahedron, where three squares and a regular triangle meet at every vertex, is denoted by 3.4^3 .

Now that we know the solid angles, we can determine which combinations of these add up to 4π . To do this we created an algorithm to determine the sums of combinations of solid angles. Of course, backtracking every possible combination would be timely, so we implemented some helpful pruning. The key to this pruning is to first order the solid angles from small to large. This allows you to know that if the n^{th} solid angle doesn't fit anymore, neither will any of the other solid angles that could otherwise be added later in the process. The algorithm then works as follows:

Polyhedra	Solid Angle
3^3	0.55129
3.4^2	1.04720
3^4	1.35935
4^3	1.57080
$3^3.4$	1.79377
$4^2.5$	1.88496
3.6^2	1.91063
$3^3.5$	2.05956
$4^2.6$	2.09440
$3^3.6$	2.23826
$4^2.7$	2.24399
$4^2.8$	2.35619
$3^3.7$	2.36648
$4^2.9$	2.44346
$3^2.4^2$	2.46192
$3^3.8$	2.46290
$4^2.10$	2.51327
$3^3.9$	2.53803
$4^2.11$	2.57039
$3^3.10$	2.59821
$4^2.12$	2.61799
3^5	2.63455
$3^3.11$	2.64749
$3^3.12$	2.68858
3.8^2	2.80176
5^3	2.96174
4.6^2	3.14159
3.4^3	3.48143
$3^4.4$	3.58963
$3^2.5^2$	3.67375
3.10^2	3.87132
$3.6.8$	3.92699
5.6^2	4.24874
$3.4^2.5$	4.44631
$3^4.5$	4.50969
$4.6.10$	4.71239

Table 3.1: Solid Angles

Algorithm *Polyhedra_Combinations*(Solid(1..n))

```

1.  $i \leftarrow 1$ 
2.  $j \leftarrow 1$ 
3.  $Combinations \leftarrow \{\}$ 
4.  $Choice \leftarrow$  constant zero array of length  $n$ 
5.  $Sum \leftarrow 0$ 
6.  $Working \leftarrow True$ 
7. while  $Working$ 
8.     do if  $|Sum - 4\pi| \leq Solid[i]$ 
9.         then  $Choice[i] \leftarrow Choice[i] + 1$ 
10.             $Sum \leftarrow Sum + Solid[i]$ 
11.             $j \leftarrow i$ 
12.     else if  $Sum = 4\pi$ 
13.         then Append  $Choice$  to  $Combinations$ 
14.         if  $j = n$ 
15.             then  $j \leftarrow j - 1$ 
16.         while  $Choice[j] = 0$  and  $j > 1$ 
17.             do  $j \leftarrow j - 1$ 
18.         if  $Choice[j] > 0$ 
19.             then  $Choice[j] \leftarrow Choice[j] - 1$ 
20.                 $Sum \leftarrow Sum - Solid[j]$ 
21.                 $i \leftarrow j + 1$ 
22.             for  $k \leftarrow j + 1$  to  $n$ 
23.                 do  $sum \leftarrow sum - Choice[k] \cdot Solid[k]$ 
24.                  $choice[k] \leftarrow 0$ 
25.         else  $Working \leftarrow False$ 
26. return  $Combinations$ 

```

It is important to note that all the equality checks in this algorithm are done on numerical representations instead of exact values. Though we use values that are more accurate than the values in Table 3.1, there is still a rounding error. This could lead to mistakes if the rounding error is too large, but this did not seem to be the case.

This algorithm generated a rather large set of allowed combinations. However, for these to be possible honeycombs, all the touching faces have to be the same polygons. This allowed for a very simple reduction: Every combination which contained an odd number of a particular polygon could never be a honeycomb. Though this was a rather simple deduction, it still eliminated roughly four fifths of the possible combinations. This left 113 possible combinations. After this, we ran an analysis to find the polyhedra that occurred in none of the combinations, so that we did not have to do further calculations on polyhedra that are not used.

The list of combinations only showed which combinations of polyhedra had solid angles that add up to 4π . However, for these combinations to be possible honeycombs, the polyhedra should be able to fit together in a non-overlapping way. To see which combinations of polyhedra could fit this constraint, we tried to fit the polyhedra together in a non-overlapping way. To do this, we first introduced the following concepts:

Definition 28. *The central vertex is the vertex around which we try to fit the polyhedra.*

Definition 29. *The circumcenter of a set of vertices is a point with equal distance to every vertex in the set.*

Definition 30. *The central vector of a polyhedron is a vector used for the construction of the polyhedron which points from the central vertex towards the circumcenter of the neighbours of the central vertex within that polyhedron.³*

Lemma 3. *Given two polyhedra that share a face, we can check if the polyhedra overlap by checking if the central vectors of the polyhedra are on the same side of the face (which can be done with linear equations).*

We then work in the following manner: Given a certain combination of polyhedra, we choose one of these polyhedra, and then try all the ways it could add the rest of the polyhedra (so that they touched one of the faces of the polyhedra that were already placed, and the central vectors are not on the same side of this face). If all the polyhedra from the combination are placed, we check if all the faces of the new polyhedra are also the face of a polyhedron on the other side. This is done by storing the faces that are connected to the central vertex for every new polygon. We then check if all faces occur twice in the overall storage. From this we can conclude that the faces of every polyhedron are also faces of a polyhedron on the other side. If this is the case, this fact combined with the fact that the solid angles add up to 4π means that the polyhedra do not overlap, meaning the combination is possibly the point configuration of a honeycomb. If the model runs through all orders to add polyhedra from the combination, without ever reaching an accepted order, this means that there is no non-overlapping way to fit the polyhedra from the combination together in a point.

It is important to note that a combination does not necessarily uniquely determine its relative placement; there might be multiple ways this combination of polyhedra fits together. Therefore, we cannot implement pruning that stops trying to fit some combination as soon as a solution is found. However, the lack of this pruning also means that some solutions might occur multiple times, so we must filter through our solution set to find the unique placements.

For this filtering it is important to note the form in which the polyhedra are stored. For the calculations we restricted the polyhedra combinations to verfs:

Definition 31. *Given a vertex v , and a number of polyhedra that contain this vertex, the *verf* is defined as the convex hull of v and the vertices that neighbour v in the polyhedra.*

Lemma 4. *Given two convex polyhedra A and B that meet at vertex v : The verfs of these polyhedra with vertex v do not overlap if and only if the polyhedra do not overlap.*

Proof. If the two verfs do not overlap, then for every face of a verf that contains vertex v , we can make a plane that separates the verfs of A and B . If there exists a vertex u in polyhedra A that is on the other side of the plane, then either u was also a neighbour of v , or A is not convex. Both of these options neglect the conditions of the lemma. The other direction is trivial, since the verfs are subsets of the polyhedra. \square

This lemma justifies limiting calculations to verfs, i.e., it shows that our proves of non-overlapping verfs suffice. Furthermore, given a combination of polyhedra, and two solutions,

³Though in general a circumcenter of three or more points does not need to exist, it does exist for neighbours of a vertex in a convex uniform polyhedron.

Row	3^3	3.4^2	3^4	4^3	5.4^2	3.6^2	6.4^2	8.4^2	$3^2.4^2$	10.4^2	12.4^2	3.8^2	4.6^2	3.4^3	$4.6.8$
1	8	0	6	0	0	0	0	0	0	0	0	0	0	0	0
2	8	0	6	0	0	0	0	0	0	0	0	0	0	0	0
3	4	6	3	0	0	0	0	0	0	0	0	0	0	0	0
4	2	0	0	0	0	6	0	0	0	0	0	0	0	0	0
5	2	0	0	0	0	6	0	0	0	0	0	0	0	0	0
6	1	2	0	0	0	3	2	0	0	0	0	0	0	0	0
7	1	0	0	1	0	0	0	0	0	0	0	0	0	3	0
8	0	12	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	12	0	0	0	0	0	0	0	0	0	0	0	0	0
10	0	8	0	0	0	0	2	0	0	0	0	0	0	0	0
11	0	8	0	0	0	0	2	0	0	0	0	0	0	0	0
12	0	6	0	4	0	0	0	0	0	0	0	0	0	0	0
13	0	6	0	4	0	0	0	0	0	0	0	0	0	0	0
14	0	4	0	2	0	0	0	0	0	0	2	0	0	0	0
15	0	4	0	2	0	0	0	0	0	0	2	0	0	0	0
16	0	4	0	0	0	0	4	0	0	0	0	0	0	0	0
17	0	4	0	0	0	0	4	0	0	0	0	0	0	0	0
18	0	4	0	0	0	0	4	0	0	0	0	0	0	0	0
19	0	2	0	4	0	0	2	0	0	0	0	0	0	0	0
20	0	2	0	0	0	0	0	0	0	0	4	0	0	0	0
21	0	0	2	0	0	0	0	0	4	0	0	0	0	0	0
22	0	0	1	0	0	0	0	0	0	0	0	4	0	0	0
23	0	0	0	8	0	0	0	0	0	0	0	0	0	0	0
24	0	0	0	2	0	0	2	0	0	0	2	0	0	0	0
25	0	0	0	2	0	0	0	4	0	0	0	0	0	0	0
26	0	0	0	2	0	0	0	0	1	0	0	0	0	2	0
27	0	0	0	1	0	0	0	2	0	0	0	1	0	1	0
28	0	0	0	1	0	0	0	0	0	0	0	0	1	0	2
29	0	0	0	0	4	0	0	0	0	2	0	0	0	0	0
30	0	0	0	0	0	2	0	0	1	0	0	0	2	0	0
31	0	0	0	0	0	1	0	0	0	0	0	1	0	0	2
32	0	0	0	0	0	0	6	0	0	0	0	0	0	0	0
33	0	0	0	0	0	0	0	2	0	0	0	0	0	0	2
34	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0

Table 3.2: Unique Placements of Combinations; columns indicate what polyhedra are used, rows indicate the combinations. Some rows are identical because some combinations can be placed in multiple distinct ways.

the verfs can be used to see if the relative placement is different. Since different relative placement means different verfs. This filtering is kept manual on purpose, since making a script that tries to rotate verfs to see if they can be aligned with a different verf would take a long time and is very susceptible to numerical errors. The small number of instances for which this check is required justifies the simplicity of the manual check, which is sufficiently reliable.

3.3.1 Results

The discribed method generated a set of 67 correct combinations of polyhedra. After filtering out the doubles, and removing polyhedra that do not occur in any combination, we were left with the combinations in Table 3.2.

This list includes the 28 known honeycombs. These require no further investigation since it is already known that they form honeycombs. This allows us to remove 26 vertex types

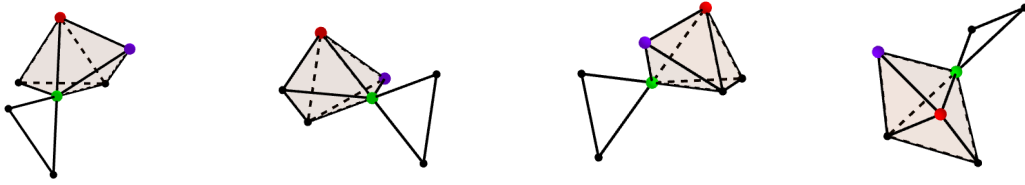


Figure 3.27: Vertex Configuration in row 4

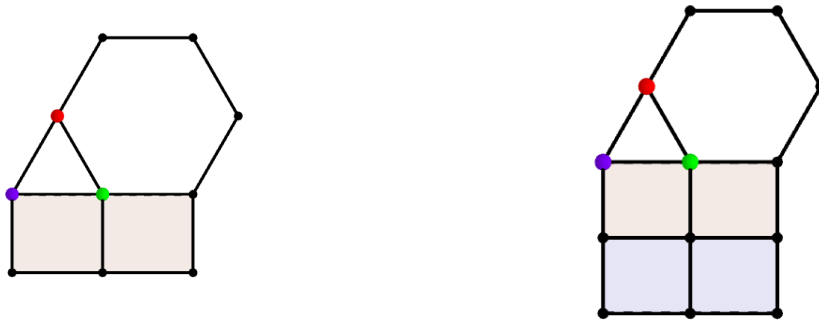


Figure 3.28: Cross-sections of the vertex types in rows 10 and 16

from consideration. This is because the gyroelongated triangular prismatic honeycomb and the elongated triangular prismatic honeycomb share the same vertex type, as do the gyroelongated alternated cubic honeycomb and the elongated alternated cubic honeycomb. This leaves us with 8 vertex types, namely those in rows 4, 6, 10, 14, 15, 16, 18, and 29. The types in rows 14, 15, 18, and 29 are prismatic stacks. Though their polygon faces do fit together, they cannot be used to create a tessellation of two-dimensional space, so these polyhedra combinations cannot be honeycombs. This leaves four combinations that require further investigation.

The combination of polyhedra in row 4 is identical to that of the quarter cubic honeycomb. The difference in the configuration comes from the fact that the two tetrahedra have to share a face. In Figure 3.27 this vertex type is shown around the green vertex, with only the nearest neighbours showing. If we consider the purple vertex, we can see that the tetrahedron it is connected to needs another tetrahedron attached to one of its faces in order to have the same vertex type as the green vertex. However, if this happens, then the red and/or the green vertex will be attached to three tetrahedra, and will therefore be of a different vertex type. Therefore this vertex type cannot be used for a honeycomb.

In the left half of Figure 3.28 we can see a cross-section of the vertex type in row 10. We used a cross-section due to the prismatic nature of the polyhedra in use. The rectangles at the bottom are in fact triangular prisms that are oriented horizontally. The other polygons are faces of prisms, both above and below the cross-section. The vertex type is created around the green vertex. In order for the purple vertex to be of the same type as the green vertex, we need to add hexagonal prisms to the edge between the red and the purple vertex. However, this means that the red vertex can no longer be of the same vertex type. Therefore, this

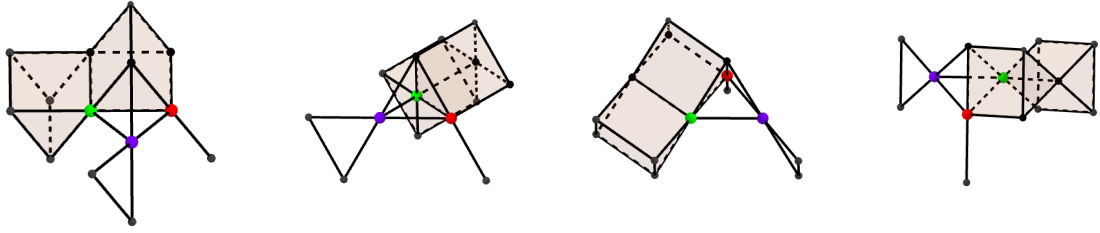


Figure 3.29: Vertex type in row 6

vertex type cannot be used for a honeycomb.

In the right half of Figure 3.28 we can see the cross-section of the vertex type in row 16. It is similar to row 10, with the difference being that the horizontally oriented prisms now include a hexagonal prism, making them indistinguishable from the other prisms. The argument used for row 10 can also be used for row 16: if the purple and green vertex need to share a vertex type, the red vertex cannot also share this vertex type.

In Figure 3.29 we see various views of the vertex type described in row 6, centered around the green vertex. We show the neighbours of the green vertex, but also the neighbours of the purples and red vertices. There is also an emphasis on the triangular prisms around the green vertex. If we want to use this vertex type for a point-uniform honeycomb, both the red and the purple vertex need to have the same vertex type as the green vertex. For the purple vertex this would mean extending the surrounding triangles outward into triangular prisms. However, this would mean the the red vertex would be surrounded by two triangular prisms which have different orientations. Therefore, the red vertex would not have the same vertex type, and thus, this vertex type cannot be used for a convex uniform honeycomb.

To conclude, we can say that we have proven Theorem 2. This means that our list of bamboozle structures contained within (unadjusted) convex uniform honeycombs is also complete. From here on, we can see what adjustments could be made to honeycombs that contain structures that are almost bamboozle structures.

Chapter 4

Adjustments of Honeycombs

In this chapter, we will focus on broadening our list of existing bamboozle structures by adjusting the honeycombs they are contained in. It could be seen as the three-dimensional version of the skew representation discussed in Chapter 2. With this skew representation, we acknowledged that minimal distance graphs are not necessarily part of the finite list of uniform convex tilings, but must be variations of uniform convex tilings. In this chapter, we try to find out what adjustments on convex uniform honeycombs could create bamboozle structures. To achieve this, we will restrict ourselves to structures discussed in Section 3.2. In that section, we excluded some honeycombs because their vertex type did not contain three edges within one plane. The adjustments discussed in this chapter could potentially change this. However, it is not directly clear what steps should be taken so that the adjusted honeycomb contains a bamboozle structure. Therefore, they will be excluded from consideration, though it might be fruitful to study these cases in more depth.

4.1 The Rectangular Bamboozle Structures

A good place to start our search for allowed adjustments is the rectangular bamboozle structures. We found a rectangular structure within the quarter cubic honeycomb that appeared to be a crooked version of the four coloured rectangular bamboozle structure. This might indicate that the four coloured rectangular bamboozle structure is contained within a valid variation of the quarter cubic honeycomb. Furthermore, it is noteworthy that none of the found structures appeared to be a different version of the six coloured rectangular structure, which justifies further attention for the six coloured rectangular structure.

We start by observing the four coloured bamboozle structure as described by Tom and Koos Verhoeff. If we create the minimal distance graph, we see that it does not contain polyhedra. However, if we also allow edges of another length (in this particular case $d_{min} \cdot \frac{2}{\sqrt{3}}$) we can see a variation of the quarter cubic honeycomb. This particular honeycomb uses adjusted tetrahedra (known as a tetragonal disphenoids [2]) and adjusted truncated tetrahedra (truncated tetragonal disphenoids). These polyhedra are depicted in Figure 4.1. They are put together so that two neighbouring longer edges are always parallel.

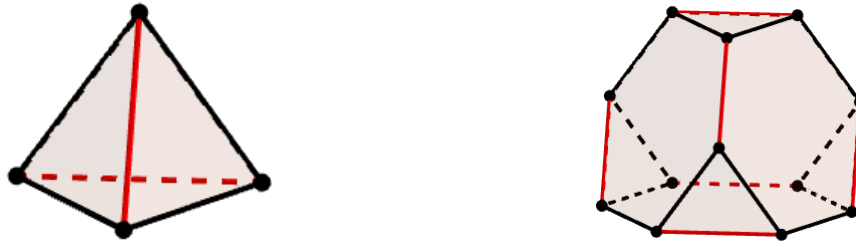


Figure 4.1: The Tetragonal Disphenoid and the Truncated Tetragonal Disphenoid. The red edges are longer than the black edges with a ratio of $\frac{2}{\sqrt{3}} : 1$.

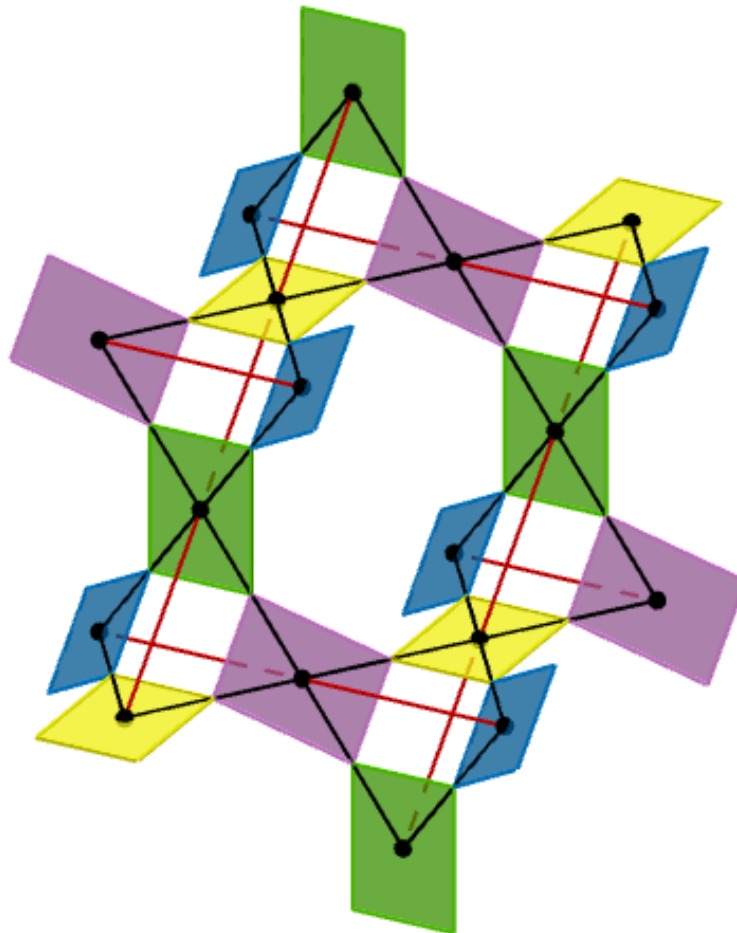


Figure 4.2: Four Coloured Rectangular Bamboozle Structure within a variation of the Quarter Cubic Honeycomb

These disphenoids can be defined by two edge lengths. If we look at the tetragonal disphenoid, we see that one edge length occurs on four edges, and the other edge length occurs on two edges. We require that the second edge length is larger than the first edge length. Furthermore, if the second edge length is larger than the first length edge by a ratio of $\sqrt{2} : 1$, the tetragonal disphenoid degenerates to a square, with the larger edges being the diagonals. For any ratio larger than this, the disphenoids ceases to exist. This means that the adjusted honeycomb exists for ratio $\alpha : 1$ if and only if $1 < \alpha < \sqrt{2}$. However, some of these ratios do not form bamboozle structures, because of new edges with length smaller than 1. Therefore, we know that the four-coloured bamboozle structure exists as a continuous family for $1 < \alpha < \frac{\sqrt{15}}{2}$. In Figure 4.2, such a bamboozle structure is shown for ratio $\frac{2}{\sqrt{3}} : 1$. This is the structure found by Tom and Koos Verhoeff.

When we look at the six-coloured rectangular bamboozle structure in a similar fashion, we can see that it too lies within a honeycomb that consists of tetragonal disphenoids. However, the placement of these disphenoids is different. The four coloured rectangular bamboozle structure has its disphenoids placed so that two neighbouring longer edges are always parallel, whereas the six coloured rectangular bamboozle places its disphenoids so that two neighbouring long edges are always perpendicular, while leaving short edges parallel. This change in placement of the tetragonal disphenoids results in a different filling of the rest of the space. Depicted in Figure 4.3, we can see that the other cells are no longer polyhedra, because not all faces are polygons. There are multiple ways to counter this, all of which requires accepting a third edge length. However, one method loses rotational symmetry of cell type, one method require a third cell type, and one method requires vertices that do not occur in the bamboozle structure. In Figure 4.3 the third option is depicted, for this seemed the most logical option, though there is not much reasoning for this choice.

This cell explains why the six-coloured honeycomb did not appear in any form on the list in Chapter 3.2. For starters, in its current form it is not convex, and no change in edge lengths could change this, since there are vertex types that contain six triangles. In order to fit these together in a non-flat way, a non-convex arrangement is required. Furthermore, neither the cell nor the honeycomb it builds is vertex transitive. A different way to turn the cells into polyhedra could make the honeycomb itself vertex-transitive, but as long as the cell is not vertex transitive, it would still not be a uniform honeycomb. Lastly, we could try to find out what equilateral polyhedron the cell is based on. To do this, we tried to recreate the cell as if the surrounding disphenoids were regular tetrahedra. However, as depicted in Figure 4.4, this forced the red square to become a rectangle instead, which therefore did not lead to an equilateral polyhedron. Since we cannot place the underlying honeycomb cells as modified regular polyhedra, it might be easier to see the honeycomb as a further variation on the quarter cubic honeycomb where the tetrahedra are twisted.

4.2 Applying Adjustments

After seeing some allowed adjustments on honeycombs used to create the rectangular bamboozle structures, namely elongating certain unused edges, and rotating around vertices, we can try to apply these adjustments to create true bamboozle structures out of the remaining structures found in chapter 3.2. In some cases, we will refrain from exact proofs of non-existence, due to the complexity of such proofs. In these cases, will we try to persuade with

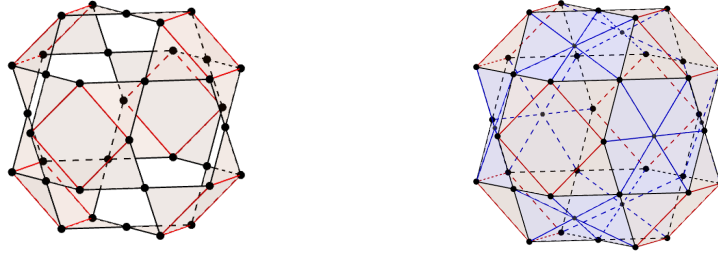


Figure 4.3: Cell type used to create the six coloured rectangular bamboozle structure

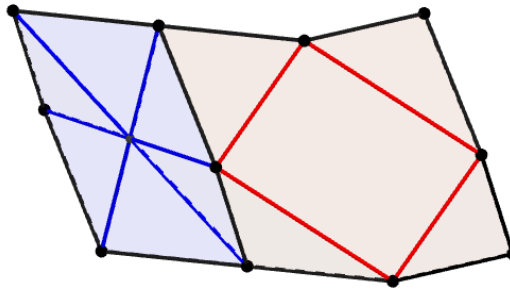


Figure 4.4: Local effect of basing the cell around a regular tetrahedron

speculations, but always consider the lack of complete certainty. Furthermore, the reader is strongly encouraged to participate in these speculations, by either drawing or trying to picture certain possibilities mentioned, for they are hard to put into a flat image.

The first case we investigate is the near-bamboozle structure found within the rectified cubic honeycomb. Since the structure fails to use a minimal distance graph as its sub-graph, we will try to elongate unused edges. This leads us to the shortened triangular anti-prism, depicted in Figure 4.6. Any edge elongation with a ratio of $\alpha : 1$ with $1 < \alpha < \sqrt{3}$ creates an adjusted honeycomb. However, only if $1 < \alpha < 2\sqrt{\frac{2}{3}}$ does said honeycomb contain a bamboozle structure. This again generates a continuous family of bamboozle structures. If we use a ratio of $\sqrt{2} : 1$, we get the square bamboozle structure, depicted in Figure 4.7.

Another interesting case is that of the hexagonal bamboozle structure. It turns out there is no way to adjust the alternated cubic honeycomb in such a way that the subgraph used by the hexagonal honeycomb is a minimal distance graph. This is due to the fact that polygons used for bamboozle structures need to have a centroid located at equal distance x from all the vertices, and the fact that hexagons divide 360° up amongst 6 vertices. When this is done equally, all the sides have length x too, because regular triangles are formed, and when it is done unequally, some angles become less than 60° , making those sides have lengths less than x . Therefore, if hexagons are used in a bamboozle structure, two adjacent neighbours of a given hexagon will have to be neighbours amongst themselves, which can only create a two dimensional tessellation.

The following case concerns the near-bamboozle structure within the truncated cubic honeycomb. In this structure, neighbouring polygons might be parallel. It is quite clear that

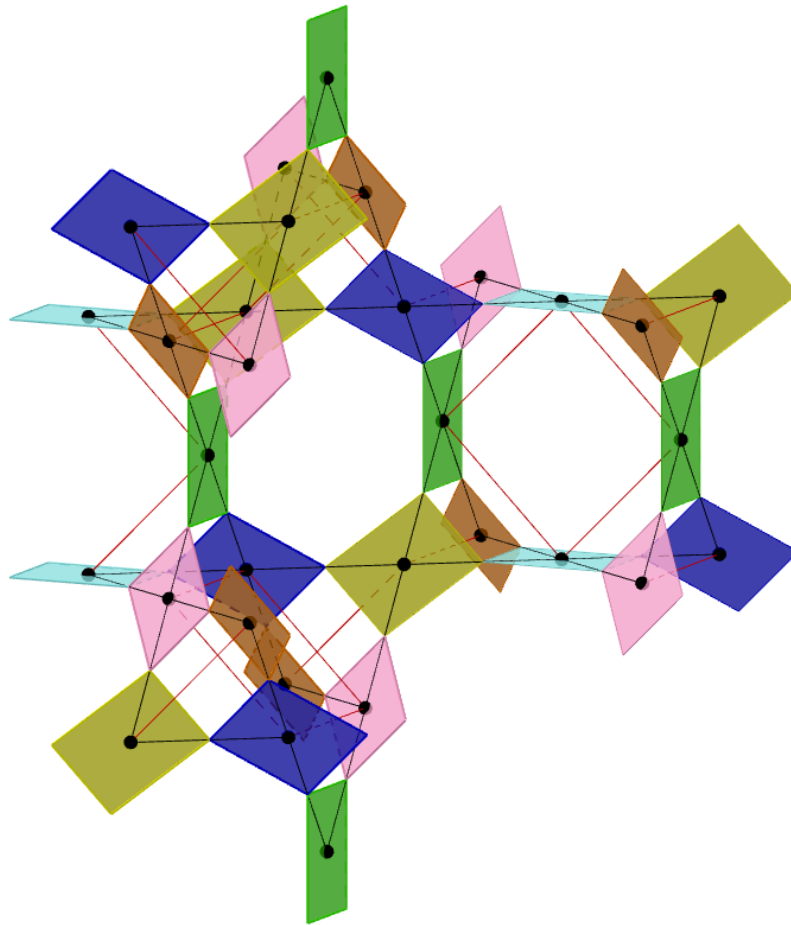


Figure 4.5: Six-Coloured Rectangular Bamboozle Structure within a variation of the Quarter Cubic Honeycomb

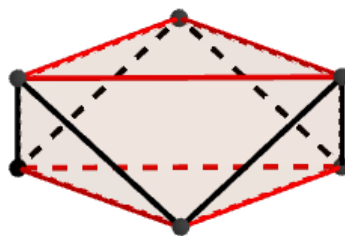


Figure 4.6: Shortened Triangular Anti-Prism, with edge ratio $\sqrt{2} : 1$ (red:black)

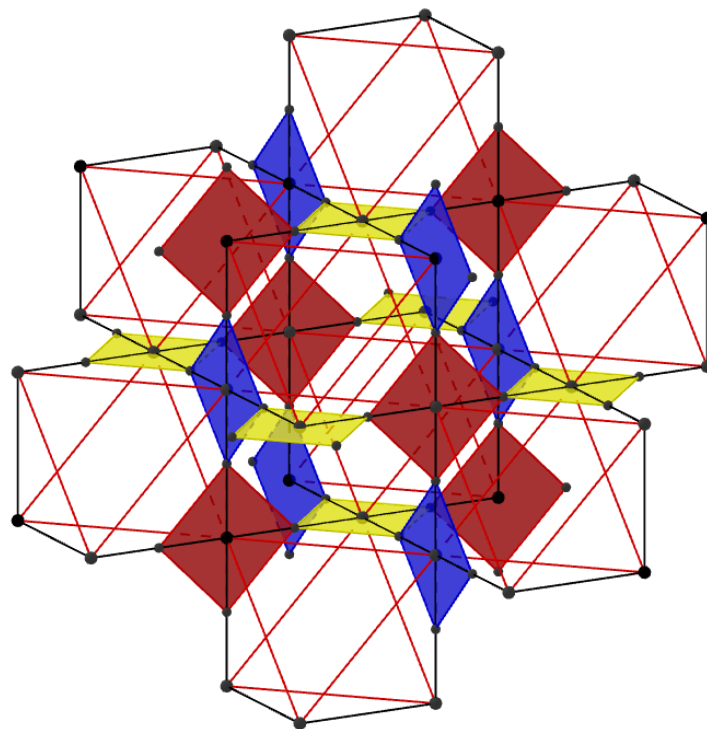


Figure 4.7: Square Bamboozle Structure within a variation of the Rectified Cubic Honeycomb

the structure is strongly related to the square bamboozle structure, but losing this parallelism could change this relation. In order to achieve this we need an adjusted octahedron that contains a ring of four vertices, which are connected by four equilateral edges and do not lie within a plane. This can be achieved by taking a regular octahedron, defining what ring of four vertices is used, and extending and shortening the other edges in a manner depicted in Figure 4.8. The ratio between the various edge lengths is enough to define this octahedron. When using this adjusted octahedron to construct a honeycomb, this also allows a rotation along edges that connect two octahedra. If we assume that, like the rectangular bamboozle structure, existence occurs for various ratios of edge lengths, we can limit our search by only varying two variables. However, after numerous attempts, all variations seem to be fruitless.

Another case concerns the near-bamboozle structure within the gyroelongated triangular prismatic honeycomb. This structure has neighbouring parallel polygons, as well as a subgraph that is not a minimal distance graph. The first problem cannot be solved. The way the polygons are connected creates a strip of regular triangles. Combined with the fact that the neighbours of any vertex must lie within a plane, this means that this strip of regular triangles must lie within a plane. Therefore, some neighbouring polygons will be parallel. An important observation is the fact that, like the near-bamboozle structure inside the truncated cubic honeycomb, this structure can be collapsed into a square-based structure within the cubic honeycomb, as depicted in Figure 4.9. However, neighbouring polygons can still be parallel. To fix this, we would need to alternate connections of polygons. To do this we would need to shift the different layers of square tilings so that they allow such alternation. However, this leads us to the alternated cubic honeycomb, which has already been investigated.

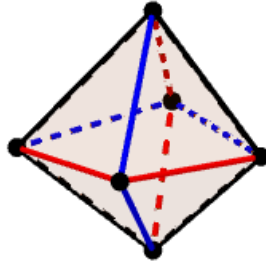


Figure 4.8: Adjusted Octahedron, with a black regular ring, red shortened edges, and blue extended edges.

Since neighbours will always remain parallel within this near-bamboozle structure, it is not important to see if the other problem could be dealt with. However, if we were interested, it probably could be done. Though the edges that would need to be elongated lie parallel to edges that cannot be elongated, it might be possible to zig-zag the elongated edges so that they become longer, but do not span further in the original direction.

That leaves the near-bamboozle structure within the Runcic Cubic Honeycomb. The approach to turn it into a true bamboozle structure would be to twist it along an edge that connects two parallel polygons. However, when attempting this, some edges are elongated, and symmetry then causes other edges to also elongate. Some of these newly elongated edges are used for other polygons. Therefore twisting the edges destroys the structure.

In conclusion, we can say that the list of bamboozle structures lying within adjusted versions of convex uniform honeycombs based on our allowed adjustments coincides with the list of bamboozle structures given by Tom and Koos Verhoeff, minus the hexagonal bamboozle structure and allowing the discussed variations.

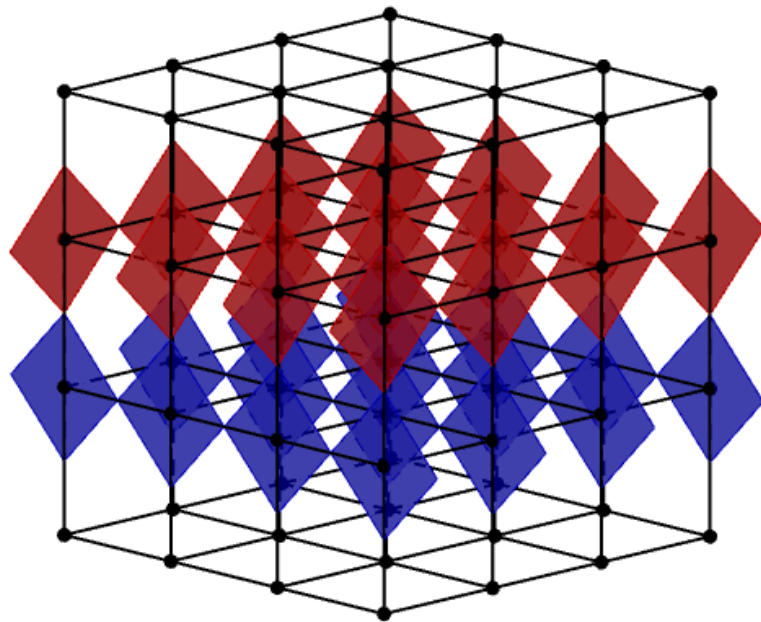


Figure 4.9: Collapsed version of the structure within the Gyroelongated Triangular Prismatic Honeycomb

Chapter 5

Conclusion, Discussion, and Further Research

The goal of this thesis was to explore what bamboozle structures exist. We started out with simplifying the problem to a two dimensional space, where we created a proof of completeness by relating all single orbits under wallpaper groups to uniform convex tilings. As we scaled back up to three-dimensional space, where we looked at the three-dimensional equivalent of the uniform convex tiling, the convex uniform honeycomb, suggesting that since the convex uniform honeycombs can be described as single trajectories under space groups they too might hold bamboozle structures. Since the list of honeycombs was not proven to be complete, we proved this completeness, with the exception of a particular case of possible honeycombs consisting of a stack of prisms or anti-prism with the rest of the space being filled up with non-parallel prism or anti-prism, or platonic or Archimedean solids. We found that three out of five known bamboozle structures do lie within convex uniform honeycombs, and that the other convex uniform honeycombs do not hold other bamboozle structures, only some near-bamboozle structures. We found that the two other known bamboozle structures lie within adjusted versions of the same convex uniform honeycomb. We speculated on adjustments of the convex uniform honeycombs that hold the near-bamboozle structures, and found that two of them were adjusted versions of known bamboozle structures, whereas the others did not appear to be adjustable into true bamboozle structures. The most notable entry of this is the hexagonal bamboozle structure, which turned out to not fulfil all qualities of a bamboozle structure. Lastly, we found that both the square bamboozle structure, and the four-coloured rectangular bamboozle structure are actually part of continuous families.

However, we have not proven the completeness of the list of bamboozle structures, and in my opinion the list is not necessarily complete. Though we have proven completeness of the bamboozle structures restricted to the convex uniform honeycombs, under the assumption that a particular type of honeycomb does not exist, the rectangular bamboozle structures prove that not all bamboozle structures are restricted to convex uniform honeycombs. Another noteworthy point is that in section 3.2, we only considered bamboozle structures that were choice free as soon as one polygon was placed, though this was never justified. It could also be the case that some bamboozle structures require choice algorithms when placed. Moreover, though we have attempted to adjust honeycombs that contain near-bamboozle structures, this section is mostly speculation instead of proofs of impossibility. Furthermore, the restriction of only

considering honeycombs that contain near-bamboozle structures could eliminate honeycombs that could be adjusted to hold bamboozle structures. However, taking all convex uniform honeycombs into account here would require a difficult search strategy.

Another thing to consider is the fact we know for certain that single trajectories under space groups are not restricted to convex uniform honeycombs. With convex uniform honeycombs, only vertex-transitive polyhedra are used. This is an unnecessary requirement, because we need only a vertex-transitive honeycomb. There exist so-called scaliform honeycombs, that are created with non-vertex-transitive cells, but are still vertex transitive themselves. In the two-dimensional case, this was taken into account with skew representation, but no three-dimensional equivalent was used. The reason scaliform honeycombs were not discussed, is because they allow so-called Johnson solids, of which there are 92. This would have made the proof of completeness of the honeycombs a lot harder, especially since the method of looking at a single vertex would no longer be sufficient, since we also need to make sure the structure is in fact point uniform. Furthermore, the strict convexity of polyhedra is also not required, which is used in the two-dimensional case, but not in the three-dimensional case.

Furthermore, it is important to understand that for a vertex transitive bamboozle structure, the underlying honeycomb does not need to be vertex transitive. As long as the vertices used for the bamboozle structure are in the same vertex transitivity class, the honeycomb could have any number of vertex transitivity classes. Allowing convex k -uniform honeycombs could lead to more bamboozle structures.

All of these topics could be studied more thoroughly to widen our understanding of bamboozle structures. However, the possibilities for interesting research related to bamboozle structures are not restricted to these topics. For instance, one could instead focus on k -uniform bamboozle structures, where the minimal distance graph of the trajectory of k points under a space group is considered. Another interesting field of research would be bamboozle structures in higher dimensions, where the neighbouring points of any point lie within a hyperplane. Yet another interesting research could focus on bamboozle structures within spherical or hyperbolic space. All in all, there is still a lot of research possible.

Bibliography

- [1] Alexandru T. Balaban. Symmetry in chemical structures and reactions. *Computers & Mathematics with Applications*, 12B(3/4):999–1020, 1986.
- [2] H.S.M. Coxeter. *Regular Polytopes*. Dover books on advanced mathematics. Dover Publications, 3rd edition, 1973.
- [3] Branko Grünbaum. Uniform tilings of 3-space. *Geombinatorics*, 4(2):49–56, 1994.
- [4] Branko Grünbaum and Geoffrey Shepard. Tilings by regular polygons. *Mathematics Magazine*, 50(5):227–247, 1977.
- [5] Edward Prince. *Mathematical Techniques in Crystallography and Materials*. Springer-Verlag Berlin Heidelberg, 2nd edition, 1994.
- [6] Tom Verhoeff and Koos Verhoeff. Folded Strips of Rhombuses, and a Plea for the 2 : 1 Rhombus. *Proceedings of Bridges 2013: Mathematics, Music, Art, Architecture, Culture.*, pages 71–78, 2013.