BACHELOR

Drawing symmetrical graphs using group theory

Tonnaer, L.M.A.

Award date:
2014

Link to publication

Drawing symmetrical graphs using group theory

Bachelor Project

Loek Tonnaer

September 22, 2014

Technische Universiteit Eindhoven

Abstract

Given a graph $G = (V, E)$, it is possible to describe all possible 2- and 3-geometric subgroups of the automorphism group $\mathrm{Aut}(G)$ using group theory. Using these classifications, we have designed algorithms to find all 2- and 3-geometric subgroups, up to conjugacy, or $\mathrm{Aut}(G)$. Once we have found these groups, we define a representation that maps an automorphism of G to an isometry of $\mathbb{R}_2$ or $\mathbb{R}_3$. Based on this representation, we then define a drawing that displays the geometric subgroup.

Contents

1   Introduction

Symmetry is one of the most powerful aesthetic tools to improve the visual appearance of a graph, and to make it appear less complicated. A symmetric drawing of a graph can give greater insight in its structure and properties. It is however not very straightforward to construct a symmetric drawing of a graph. Moreover, different drawings might display different symmetries of the same graph.

The first step to finding symmetric drawings of a graph is to look for automorphisms of the graph that can be displayed as symmetries of a drawing. Once we have found all such automorphisms, we construct drawings that display these automorphisms as symmetries. A drawing in this case can be defined on the (two-dimensional) plane, but more general on the n-dimensional Euclidian space (or $\mathbb{R}^n$). In this paper we will focus on two- and three-dimensional drawings. We define an n-geometric automorphism group as one that can be displayed as symmetries of a drawing in n dimensions. We first gather some knowledge about graph automorphisms and symmetries in $\mathbb{R}^n$, and use group theory to design an algorithm to find all the 2- and 3-geometric automorphism groups of a graph. Once we have found these groups, we design an algorithm to represent every element of an automorphism group as a symmetry of a drawing. Then we design an algorithm to construct such a drawing that displays an entire automorphism group as symmetries.

We have implemented these algorithms in **Sage** [6], which uses **GAP** [5] for most of its group theory functionality.

2 Background

2.1 Automorphisms, drawings and symmetries

This paper aims at the displaying of automorphisms of graphs as symmetries. To clearly understand what this means, we first need some definitions.

Let $G = (V, E)$ be a graph, where V is a (finite) set of vertices and E a set of edges (i.e. 2-tuples of elements from V). In this paper, we will always assume G to be a simple undirected graph, i.e. there is at most one edge between any two different vertices, there are no loops (edges connected at both ends to the same vertex) and edges have no specific direction.

Definition 1. Automorphism: An automorphism of a graph $G = (V, E)$ is a permutation p of V such that:

$$\{u, v\} \in E \Rightarrow \{p(u), p(v)\} \in E.$$

Aut(G): The set of automorphisms of a graph G form a group denoted by Aut(G).

We want to draw the graph G in such a way that several of its automorphisms are realised as symmetries (where possible), at first in the 2-dimensional plane, but more generally in $\mathbb{R}^n$. For this, we need to define what a drawing of a graph is, what a symmetry is and what displaying an automorphism means.

Definition 2. Drawing: A (proper) drawing D of a graph G is an injective function $D : V \to \mathbb{R}^n$ for some $n \in N$. A vertex v is placed at $D(v)$ and an edge $\{u, v\}$ is represented by the line segment between $D(u)$ and $D(v)$.

Strictness: A drawing is strict if for all edges $\{u, v\} \in E$ and all vertices $w \neq u, v$, the point $D(w)$ does not lie on the line segment between $D(u)$ and $D(v)$.

The quality "proper" indicates that the drawing is injective, i.e. no 2 vertices are drawn at the same position. Note that if we want a drawing to be such that the underlying graph can always be found, we must have both properness and strictness. See Figure 1. Without strictness (a) could be isomorphic to the other two. However, if we know that the drawing is strict, we know that (a) must be isomorphic to (c), and both cannot be isomorphic to (b). This way, there is no ambiguity; given a certain strict drawing, the underlying graph (its sets of vertices and edges) can always be recovered. From here on we will always assume properness when speaking of a drawing.
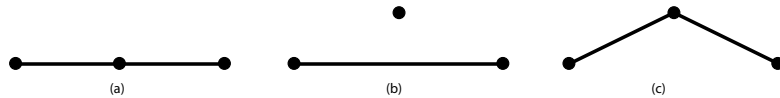
Figure 1: Strictness is an important quality of a drawing

Definition 3. Isometry: An isometry (of $\mathbb{R}^n$) is a mapping of $\mathbb{R}^n$ onto itself that preserves distances.

Symmetry: A symmetry of a drawing D of a graph G is an isometry of $\mathbb{R}^n$ that maps the image of the drawing onto itself.

Definition 4. Displaying an automorphism: A drawing D of a graph G is said to display an automorphism h if there is a symmetry σ of D such that

$$h = D^{-1} \sigma D.$$

Note that $D^{-1}\sigma D$ indeed defines an automorphism.

Example 1. See Figure 2(a).
This is a drawing of a graph $G = (V, E)$ with 4 vertices and 5 edges. The drawing is proper; all 4 vertices are drawn at different positions. It is also strict, no vertex is placed on an edge between other vertices.
Verify that the permutation (12) (in cycle notation) is not an automorphism of G.
The permutations (24), (13) and (13)(24) however are automorphisms of G. To see this, carry out these permutations in the drawing, and verify that the lines representing edges "do not change"; the image appears the same (apart from renumbering). Now note that this drawing displays these three permutations (as symmetries); (13) e.g. is realised by a reflection in the line through vertices 2 and 4. The permutation (13)(24) is realised by a rotation over 180 degrees.
Now consider Figure 2(b). Verify that here, (123)(45) defines an automorphism. This drawing however does not display this automorphism as a symmetry. For instance, edge {2,4} is mapped onto edge {3,4}, which has a different length in this drawing. Hence, distance is not preserved.
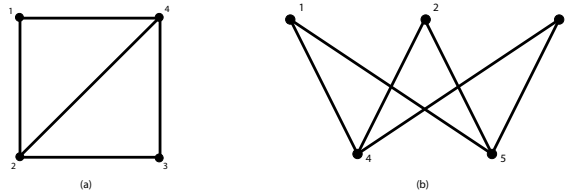
Figure 2: A drawing of a graph

2.2 Geometric automorphisms and automorphism groups

Our goal is now to display automorphisms of a graph, preferably as many as possible. For this we use the concept of a geometric automorphism, as introduced by Eades and Lin [2], generalised to n dimensions, as done by Abelson, Hong and Taylor [1].

Definition 5. n-Geometric automorphism: An automorphism h of a graph G is n-geometric if there is a drawing $D : V \to \mathbb{R}^n$ of G that displays h.
(Strictly) n-geometric subgroup: A subgroup H of $\mathrm{Aut}(G)$ is n-geometric if there is a single drawing $D : V \to \mathbb{R}^n$ that displays every element of H. The subgroup is strictly n-geometric if the drawing D is strict.

Example 2. Consider Figure 3. Here we see two different drawings of the same graph, the complete graph with 4 vertices, K_4. It provides an example that different drawings can display different automorphism groups; (a) displays a 2-geometric automorphism group of size 8, (b) displays a 2-geometric automorphism group of size 6. The symmetries of both groups are represented as rotations and reflections.
Now let's look again at Figure 2(b). As we have seen, the automorphism (123)(45) is not displayed as a symmetry by this drawing. As it turns out, there is no drawing (in $\mathbb{R}^2$) that displays it, hence (123)(45) is not 2-geometric. However, e.g. (13)(45) is 2-geometric, as this drawing displays it.

3

Figure 3: Two drawings of K_4 displaying (a) a 2-geometric automorphism group of size 8 and (b) a 2-geometric automorphism group of size 6.

2.3 Symmetries represented by matrices

Note that since V is a finite set, we can always define a drawing D of a graph $G = (V, E)$ in such a way that the barycentre of its image is at the origin. Now, let $I_n(\mathbb{R})$ be the group of isometries of $\mathbb{R}^n$, and let $O_n(\mathbb{R})$ be the subgroup of $I_n(\mathbb{R})$ that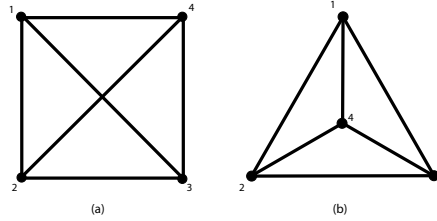 fixes the origin. The elements of $O_n(\mathbb{R})$ are represented by orthogonal $n \times n$-matrices (a matrix A is orthogonal if and only if $AA^T = I$). Let $SO_n(\mathbb{R})$ denote the subgroup of $O_n(\mathbb{R})$ corresponding to the matrices of determinant 1.

Now, if we indeed define a drawing such that the barycentre of its image is at the origin, then all its symmetries form a subgroup of $O_n(\mathbb{R})$. Furthermore, notice that if H is an n-geometric subgroup of $\mathrm{Aut}(G)$, H is isomorphic to a subgroup of $O_n(\mathbb{R})$.

2.4 Characterising n-geometric automorphism groups

First, consider the following lemma from [1].

Lemma 1. *A group $H \subseteq Aut(G)$ is n-geometric with respect to a drawing D if and only if there exists an injective homomorphism $\phi : H \to O_n(\mathbb{R})$ such that for all $v \in V$ and $h \in H$ we have $D(hv) = \phi(h)D(v)$.*

Proof. As mentioned before we can always define a drawing such that the barycentre of $D(V)$ is at the origin, so that isometries representing elements of H belong to $O_n(\mathbb{R})$. Furthermore, we always choose an isometry such that it acts as the identity on the subspace orthogonal to that spanned by $D(V)$. This way we see that each $h \in H$ can be associated with a unique isometry $\phi(h)$. Therefore H is n-geometric if and only if there is a drawing $D : V \to \mathbb{R}^n$ and a homomorphism $\phi : H \to O_n(\mathbb{R})$ such that $\phi(h)D(v) = D(hv)$ for all $h \in H$ and all $v \in V$. From this it follows immediately that ϕ must be injective. Indeed, if $\phi(h) = 1$, then $D(hv) = D(v)$ for all $v \in V$. Since D is injective, we have $hv = v$ for all $v \in V$; i.e. $h = 1$. $\square$

Such a homomorphism $\phi : H \to O_n(\mathbb{R})$ is called a *representation* of H. In matrix terms, every element of H is represented by an orthogonal matrix. A representation is called *faithful* if ϕ is injective.

Now consider some basic terms from group theory (see e.g. [3]):

Definition 6. Let H be a group acting on a set U. Let $u \in U$.
Orbit: The subset $Hu = \{x \in U | x = gu \text{ for some } g \in H\}$ of U is the H-orbit of u.
Stabiliser: The subgroup $H_u = \{g \in H | gu = u\}$ of H is the stabiliser of u in H.
Conjugacy: Two elements a and b of H are conjugate if there exists an $h \in H$ such that $b = hah^{-1}$.

The orbits divide the set U in equivalence classes, so each of them can be specified by a representative element. Conjugacy is an equivalence relation, we call the corresponding

equivalence classes of H the conjugacy classes.

Using these terms, we can now characterise n-geometric groups with the following theorem (from [1]).

Theorem 1. *A subgroup $H \subseteq Aut(G)$ is n-geometric if and only if there is an injective homomorphism $\phi : H \to O_n(\mathbb{R})$ such that for representatives $v_1, v_2, \ldots, v_k$ of the orbits of H acting on V there are distinct points $a_1, a_2, \ldots, a_k \in \mathbb{R}^n$ such that $\phi(H_{v_i}) = \phi(H)_{a_i}$ for $i = 1, 2, \ldots, k$.*

Proof. Say H is n-geometric and let $\phi : H \to O_n(\mathbb{R})$ and $D : V \to \mathbb{R}^n$ be the associated homomorphism and drawing as in Lemma 1. Then for $h \in H$ and $v \in V$ we have $D(hv) = \phi(h)D(v)$, and so $h \in H_v$ if and only if $\phi(h) \in \phi(H)_{D(v)}$. Now, if $v_1, v_2, \ldots, v_k$ represent the orbits of H on V, we may take $a_i = D(v_i)$. Clearly, all the a_i are distinct, since D is injective.

To prove the converse, suppose that $\phi : H \to O_n(\mathbb{R})$ is an injective homomorphism and that for representatives $v_1, v_2, \ldots, v_k$ of the orbits of H on V we have distinct points $a_1, a_2, \ldots, a_k$ such that $\phi(H_{v_i}) = \phi(H)_{a_i}$. Note that for any $r \neq 0 \in \mathbb{R}$ and any $a \neq 0 \in \mathbb{R}^n$ we have $\phi(H)_{ra} = \phi(H)_a$. Therefore we may scale the points $a_1, a_2, \ldots, a_k$ such that no two lie on the same distance from the origin.

For $v \in V$ we have $v = hv_i$ for some $h \in H$ and some orbit representative v_i. Then $\phi(h)a_i$ depends only on v and not on the choice of h. To see this, suppose that $v = gv_j$ for some $g \in H$. Then v_i and v_j are in the same orbit of h, and so $i = j$. We have $h^{-1}gv_i = h^{-1}v = v_i$, and so $h^{-1}g \in H_{v_i}$. But $\phi(H_{v_i}) = \phi(H)_{a_i}$ and so $\phi(h)^{-1}\phi(g)a_i = a_i$. Thus indeed $\phi(g)a_i = \phi(h)a_i$. This shows that the drawing $D(v) = \phi(h)a_i$ is well-defined. We now show that D is injective. Let $D(u) = D(v)$, where $u = gv_i$ and $v = hv_j$ for some i and j. Then $\phi(g)a_i = \phi(h)a_j$. Since $\phi(g)$ and $\phi(h)$ are isometries, we have that a_i and a_j are at the same distance from the origin. But since we scaled the points $a_1, a_2, \ldots, a_k$ such that no two lie on the same distance from the origin, we see that $i = j$. But now $\phi(g^{-1}h) \in \phi(H)_{a_i} = \phi(H_{v_i})$ and since ϕ is injective we have $g^{-1}h \in H_{v_i}$. Thus $gv_i = hv_i$ and so $u = v$, and therefore D is indeed injective.

Finally, for $v = hv_i$ and $g \in H$ we have $D(gv) = \phi(gh)a_i = \phi(g)\phi(h)a_i = \phi(g)D(v)$, which proves that H is n-geometric. $\square$

From this proof it follows that the condition that the points $a_1, a_2, \ldots, a_k$ are distinct is equivalent to the requirement that at most one of them is 0.

Our goal is to find all n-geometric automorphism groups of a given graph $G = (V, E)$. It is evident that every subgroup of $Aut(G)$ is n-geometric for $n = |V|$. However, for $n \leq 3$ the conjugacy classes of subgroups of $O_n(\mathbb{R})$ are rather limited, which limits the types of groups that can be 2- or 3-geometric.

Using the characterisation of Theorem 1, we will give classifications of the 2- and 3-geometric groups in the next sections. We have seen that the group should have a faithful representation by orthogonal matrices. Since conjugate representations in $O_n(\mathbb{R})$ produce equivalent drawings (we will later see in Lemma 2 what exactly this means), we only need to describe the finite subgroups of $O_n(\mathbb{R})$ up to conjugacy. For this we need more details about the matrix presentations and the actions of the groups on $\mathbb{R}^n$. We will summarise the results here, mathematical details can for instance be found in [4].

From Theorem 1 we know that a permutation group $H \subseteq Aut(G)$ is n-geometric if there is an isomorphism ϕ between H and a finite subgroup T of $O_n(\mathbb{R})$ such that:

- if H fixes more than one vertex, then T fixes a vertex other than the origin; and

- for every vertex v, $\phi(H_v)$ is the stabiliser in T of a point in $\mathbb{R}^n$

We refer to this group T as the *type* of H.

As it turns out, the list of possible types T and stabilisers is quite restricted. Further-

more, the following theorem (see [3]) will be useful for the characterisation of geometric subgroups in the following sections:

Theorem 2. *(Orbit Stabiliser Theorem)*
Let G be a group acting on a set X, and let $x \in X$. Let $Orb(x)$ be the G-orbit of x and $Stab(x)$ be the stabiliser of x in G. Then:

$$|Orb(x)| = \frac{|G|}{|Stab(x)|}$$

Proof. Define the mapping $\varphi : G \to \mathrm{Orb}(x)$ given by $\varphi(g) = gx$. Clearly φ is surjective, as by definition x is acted on by all elements of G. Now, for any $g, h \in G$ we have $\varphi(g) = \varphi(h) \Leftrightarrow gx = hx \Leftrightarrow g^{-1}(gx) = g^{-1}(hx) \Leftrightarrow x = (g^{-1}h)x \Leftrightarrow g^{-1}h \in \mathrm{Stab}(x)$. This means that $g \equiv h \pmod{\mathrm{Stab}(x)}$. Thus there is a well-defined bijection $G/\mathrm{Stab}(x) \to \mathrm{Orb}(x)$ given by $g\,\mathrm{Stab}(x) \mapsto gx$. So $\mathrm{Orb}(x)$ has the same number of elements as $G/\mathrm{Stab}(x)$. The result immediately follows. $\square$

For many finite subgroups of $O_2(\mathbb{R})$ and $O_3(\mathbb{R})$, the stabilisers are identified up to conjugacy by their orders. In such a case only the orbit lengths are needed to determine whether a group is geometric. The orbits of length less than $|H|$ are called the *short* orbits and those of length $|H|$ the *regular* orbits of H. The orbits of length 1 are said to be *trivial*.

2.5 Conjugate subgroups

The following Lemma (from [1]) shows that we basically only need to find n-geometric subgroups of the automorphism group $\mathrm{Aut}(G)$ of a graph $G = (V, E)$ up to conjugacy, because conjugate n-geometric subgroups essentially have the same drawings. Let $\mathrm{Sym}(V)$ be the group of all permutations of V.

Lemma 2. *Let H be an n-geometric subgroup of Aut(G) with respect to a drawing D and suppose that $H' = g^{-1}Hg \subseteq Aut(G)$, where $g \in Sym(V)$. Then H' is n-geometric with respect to the drawing D' defined by $D'(v) = D(gv)$.*

Proof. Let $\phi : H \to O_n(\mathbb{R})$ be a homomorphism such that $D(hv) = \phi(h)D(v)$ for all $h \in H$ and all vertices $v \in V$ (as in Lemma 1). Define $\phi' : H \to O_n(\mathbb{R})$ by $\phi'(h') = \phi(gh'g^{-1})$. Let $h' \in H'$ and $v \in V$, so there is a $h \in H$ such that $h' = g^{-1}hg$ (namely $h = gh'g^{-1}$). Then we have $D'(h'v) = D(gh'v) = D(hgv)$. But $D(hgv) = \phi(h)D(gv) = \phi(h')D'(v)$, and so $D'(h'v) = \phi(h')D'(v)$, proving that H' is n-geometric with respect to D'. $\square$

Example 3. Let G be a circuit if length 5. In Figure 4 there are 4 different drawings of G. Let $H = \langle (12345) \rangle$, then H clearly is a 2-geometric automorphism group of G, as it is displayed by the drawing in e.g. (a) as a rotation by $\frac{2\pi}{5}$. Now, let $g = (25)(34) \in \mathrm{Sym}(V)$. We easily verify that $H' = g^{-1}Hg = \langle (54321) \rangle \subseteq \mathrm{Aut}(G)$. If we define D' by $D'(v) = D(gv)$ as in Lemma 2 we find the drawing in Figure 4(d), which displays H' as a rotation by $\frac{2\pi}{5}$. Apart from renumbering, (a) and (d) look identical.
We can form a similar argument for (b) and (c).

2.6 Choosing a representation

Given an n-geometric subgroup of the automorphism group $\mathrm{Aut}(G)$ of a graph $G = (V, E)$, it's possible to construct different drawings that display the same geometric automorphism group, depending on the choice of the representation. To see this, let $H \subseteq \mathrm{Aut}(G)$ be a n-geometric group. Consider two different faithful representations ϕ, θ of H with a fixed
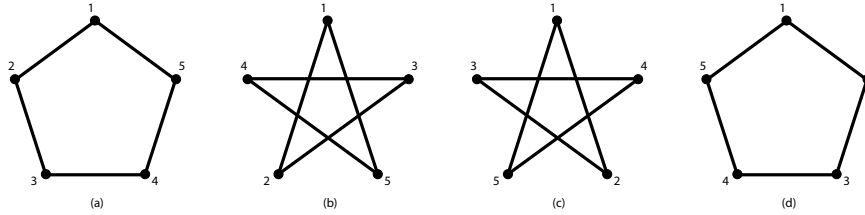
Figure 4: Different representations used to display the same automorphism group

image $T \subseteq O_n(\mathbb{R})$. Then $\phi\theta^{-1}$ is an automorphism of H. Hence, choosing a different representation is equivalent to permuting a particular representation with an automorphism of H.

Example 4. Consider again Figure 4 of a circuit G of length 5. Let $p = (12345)$ and $H = \langle p \rangle$. Verify that all 4 drawings display H. Each uses a representation that maps a generator of H (p, p^2, p^3 or p^4 respectively) to a rotation by $\frac{2\pi}{5}$. Clearly, (a) and (b) are different, showing that a different representation can result in a different drawing. As we have already seen in Example 3, (a) and (d), as well as (b) and (c) are the same up to relabeling, since p and p^4, as well as p^2 and p^3 are conjugate in $\mathrm{Aut}(G)$ by $(25)(34)$. Furthermore, notice that p and p^2 are conjugate by $(2345) \in S_5 \backslash \mathrm{Aut}(G)$, therefore the drawings that display them use the same points for vertices, but with different edges.

7

3 Drawing 2-dimensional symmetric graphs

3.1 Classification of 2-geometric subgroups

We first look at the finite subgroups of $SO_2(\mathbb{R})$, represented by orthogonal matrices of determinant 1. Then we expand this search to subgroups of $O_2(\mathbb{R})$.

The only finite subgroups of $SO_2(\mathbb{R})$ are the cyclic subgroups $\mathcal{C}_k$ of order k, generated by a single element. $\mathcal{C}_k$ fixes the origin, and all its other orbits are regular. The action on $\mathbb{R}^2$ of a generator of C_k can be represented by the *rotation matrix*

$$\begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}, \quad \text{where } \theta = \frac{2\pi m}{k} \text{ for some } m \text{ coprime to } k.$$

Note that geometrically this defines a rotation around the origin over an angle θ.

The finite subgroups of $O_2(\mathbb{R})$ that are not contained in $SO_2(\mathbb{R})$ are the groups $\mathcal{D}_k$ of order $2k$. For $k > 1$ these are the *dihedral* groups. A group $\mathcal{D}_k$ is the group of symmetries of a regular polygon with k sides, including both rotations and reflections. The group $\mathcal{D}_1$ is cyclic or order 2 and its orbits have lengths 1 and 2. Thus, the 2-geometric cyclic permutation groups are $\mathcal{D}_1$ and the groups $\mathcal{C}_k$. We therefore find the following result for cyclic groups:

Result 1. A cyclic permutation group is 2-geometric if and only if its order is 2 or it has at most one fixed vertex and its non-trivial orbits are regular.

If we draw a vertex somewhere in the plane, but not in the origin, then the action of $\mathcal{C}_k$ will rotate this point around the origin over $\frac{2\pi m}{k}$, with m coprime to k. Therefore, after exactly k such rotations, it will be back where it started, but not sooner. If we draw the vertex in the origin instead, it will not move under the action of $\mathcal{C}_k$. See e.g. Figure 5; here the solid lines represent the image of the x-axis under the action of $\mathcal{C}_{10}$.
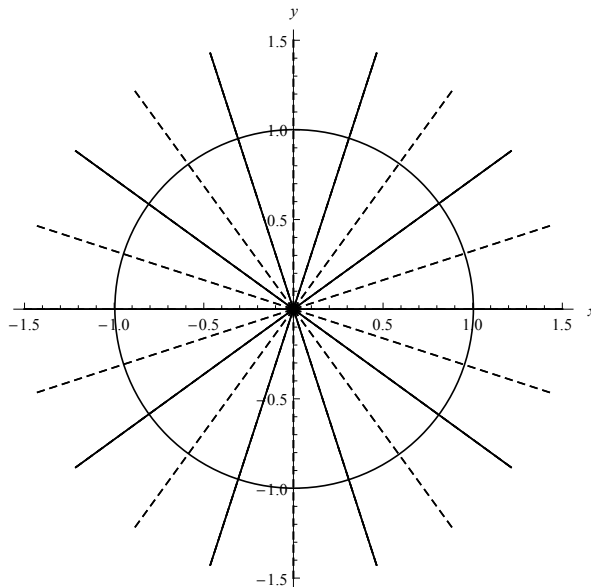

Figure 5: The image of the x-axis under the action of $\mathcal{C}_{10}$ or $\mathcal{D}_{10}$

The intersection of $\mathcal{D}_k$ with $SO_2(\mathbb{R})$ is the cyclic group $\mathcal{C}_k$, and $\mathcal{D}_k$ can be generated by the rotation matrix given above and the *reflection matrix*

$$\begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}.$$

Note that this matrix geometrically represents a reflection in the x-axis. Every element of $\mathcal{D}_k$ not in $\mathcal{C}_k$ has order 2 and fixes two points on the unit circle. Thus, $\mathcal{D}_k$ has two short orbits of length k on the unit circle. Hence, we find the following result for dihedral groups:

Result 2. A dihedral group is 2-geometric if and only if it has a 2-geometric cyclic subgroup of index 2. If $k > 2$ this subgroup of index 2 is unique.

Look again at Figure 5. Now we let the lines represent the image of the x-axis under the action of $\mathcal{D}_{10}$. If we place a vertex in the origin, we see that it again remains fixed under the action of $\mathcal{D}_{10}$. A vertex placed on the x-axis or on any of the other lines in Figure 5 will have an orbit of size 10 under the action of $\mathcal{D}_{10}$. However, if we place a vertex not in the origin, and not such that the line through that vertex and the origin makes an angle of $\frac{i\pi}{10}$ with the x-axis, for $i = 1, 2, \ldots, 20$ (these are the solid and dashed lines in Figure 5, it will have an orbit of size 20. This is because it is not in the fixed point space of any non-trivial element of $\mathcal{D}_{10}$. Notice that the points on the dashed lines are in fact part of the fixed point space for some non-trivial element of $\mathcal{D}_{10}$; a rotation combined with a reflection.

3.2 Finding 2-geometric subgroups

3.2.1 Algorithms

We now describe algorithms to find all the 2-geometric automorphism groups of a graph $G = (V, E)$, first the cyclic groups, then the dihedral groups. We have implemented these algorithms in **Sage**. These implementations can be found in the Appendix.
Using Result 1 we specify the following algorithm for finding the 2-geometric cyclic groups.

Algorithm 1. *Finding the 2-geometric cyclic groups*
Input: A graph $G = (V, E)$, defined by its vertices V and edges E.
Output: All 2-geometric cyclic subgroups of the automorphism group $\mathrm{Aut}(G)$ of graph G, up to conjugacy.
 1: $\mathrm{Aut}(G) \leftarrow$ Automorphism group of the graph G
 2: $ub \leftarrow$ Maximum of the orbit lengths of $\mathrm{Aut}(G)$ on V (to be used as an upper bound for the order of a 2-geometric element)
 3: $repG \leftarrow$ List of representatives for the conjugacy classes of $\mathrm{Aut}(G)$
 4: $Cks \leftarrow$ Empty list (used to store all cyclic 2-geometric cyclic subgroups of $\mathrm{Aut}(G)$)
 5: **for** all $h \in repG$ **do**
 6: **if** $(\mathrm{order}(h) = 2)$ **or** $(\mathrm{order}(h) > 2$ **and** $\mathrm{order}(h) \leq ub$ **and** $|\mathrm{fix}(h)| \leq 1$ **and** all non-singleton cycles of h are of length $\mathrm{order}(h))$ **then**
 7:   Append $\langle h \rangle$ to Cks
 8: **end if**
 9: **end for**
10: **return** Cks

Note that no specific method is mentioned of how to compute $\mathrm{Aut}(G)$, its orbit lengths, and representatives of its conjugacy classes. Computing the automorphism group $\mathrm{Aut}(G)$ is a hard problem, however several quite efficient algorithms are known for it, see for instance [7], [8] or [9]. **Sage** has such an algorithm for finding the automorphism group of a graph implemented, as well as for finding orbit lengths and representatives for the conjugacy classes of $\mathrm{Aut}(G)$.
It is clear that after running the algorithm, Cks contains exactly those (2-geometric) cyclic permutation groups that correspond to Result 1, up to conjugacy.

Definition 7. Normaliser: The *normaliser* $N_A(H)$ of a subset H in the group A is defined as $N_A(H) = \{a \in A | a^{-1}Ha = H\}$.

We make use of the idea of the previous algorithm and of Result 2 for the following algorithm to find the 2-geometric dihedral groups.

Algorithm 2. *Finding the 2-geometric dihedral groups*
Input: A graph $G = (V, E)$, defined by its vertices V and edges E.
Output: All 2-geometric dihedral subgroups of the automorphism group $\mathrm{Aut}(G)$ of graph G, up to conjugacy.
 1: $\mathrm{Aut}(G) \leftarrow$ Automorphism group of the graph G
 2: $ub \leftarrow$ Maximum of the orbit lengths of $\mathrm{Aut}(G)$ (to be used as an upper bound for the order of a 2-geometric element)
 3: $repG \leftarrow$ List of representatives for the conjugacy classes of $\mathrm{Aut}(G)$
 4: $Cks2 \leftarrow$ Empty list (used to store all cyclic 2-geometric subgroups of $\mathrm{Aut}(G)$ with at most 1 fixed point)
 5: **for** all $h \in repG$ **do**
 6:    **if** $(\mathrm{order}(h) \geq 2$ **and** $\mathrm{order}(h) \leq ub$ **and** $|\mathrm{fix}(h)| \leq 1$ **and** all non-singleton cycles of h are of length $\mathrm{order}(h))$ **then**
 7:        Append h to $Cks2$
 8:    **end if**
 9: **end for**
10: $Dks \leftarrow$ Empty list (used to store all 2-geometric dihedral subgroups of $\mathrm{Aut}(G)$)
11: **for** all $g \in Cks2$ **do**
12:    $N \leftarrow$ Normaliser of $\langle g \rangle$ in $\mathrm{Aut}(G)$
13:    $Ncr \leftarrow$ List of representatives for the conjugacy classes of N
14:    $Ncr2 \leftarrow$ All elements of Ncr of order 2
15:    **for** all $a \in Ncr2$ **do**
16:        **if** $(ga)^2 = e$ **and** $g \neq a$ **then**
17:            Append $\langle g, a \rangle$ to Dks
18:        **end if**
19:    **end for**
20: **end for**
21: **return** Dks

Notice that the first part of the algorithm is almost identical to Algorithm 1, it selects the same elements, apart from those with order 2 and more than one fixed point. The only other difference is that Cks stores groups generated by a single permutation, whereas $Cks2$ only stores single permutations. We use this similarity in our implementation by combining both algorithms. Again, functions such as computing the normaliser are readily usable in **Sage**, so have not been described in more detail.

3.2.2   Extensive example: K_4

Example 5. We will demonstrate the algorithms in the previous section for the complete graph K_4 (see also Figure 3). It has 4 vertices and 6 edges; one between every pair of vertices. Its automorphism group is exactly the symmetry group with 4 elements: $\mathrm{Aut}(K_4) = S_4$, where $S_n = \mathrm{Sym}(n)$ is the group of all permutations of n elements. We now simply execute Algorithm 1 step by step to find all cyclic subgroups of K_4.
Clearly, the maximum orbit length is 4, so set an upper bound $ub \leftarrow 4$ for the order of a 2-geometric element.
We now want to find representatives of the conjugacy classes of S_4. It has a total of five conjugacy classes:

- $(1)_4$: No change (1 element: $\{e\}$, where e is the unit permutation).
- (2): Interchanging two (6 elements: $\{(23), (13), (12), (03), (13), (01)\}$).
- (3): A cyclic permutation of three (8 elements: $\{(123), (132), (032), (023), (013), (031), (012), (021)\}$).
- (4): A cyclic permutation of all four (6 elements: $\{(0123), (0321), (0231), (0132), (0312), (0213)\}$).
- $(2)(2)$: Interchanging two, and also the other two (3 elements: $\{(01)(23), (02)(13), (03)(12)\}$).

Notice that we have numbered the vertices 0 to 3, rather than 1 to 4. This is simply because it is the convention **Sage** uses. As representatives, we use the following elements (arbitrarily chosen): $e, (23), (123), (0123), (01)(23)$.

We now check these representatives one by one, to see if they have order 2, or order greater than 2 and at most one fixed vertex. If so, we add the group generated by it to our list Cks. We find:

- order$(e) = 1 < 2$, so we do not select it. Although it technically can be seen as a geometric element, e generates the trivial group which is of no real interest to us; it will always be displayed in any drawing.
- order$((23))=2$, so we will add $\langle (23) \rangle$ to Cks without having to count its fixed points.
- order$((123))=3$, it has only one fixed point, namely 0, so we add $\langle (123) \rangle$ to Cks.
- order$((0123))=4$, it has no fixed points, so we add $\langle (0123) \rangle$ to Cks.
- order$((01)(23))=2$, so we add $\langle (01)(23) \rangle$ to Cks.

So now our list Cks contains four 2-geometric cyclic subgroups of $\text{Aut}(K_4)$.

Now let's use Algorithm 2 to find all 2-geometric dihedral subgroups of $\text{Aut}(K_4)$. The first part is similar to Algorithm 1; additionally we just need to check the fixed points of the order 2 elements (23) and $(01)(23)$. We find that fix$((23)) = \{0,1\}$, so $|\text{fix}((23))| = 2 > 1$, so in this case we do not select (23). Furthermore, we find fix$((01)(23)) = \emptyset$ so $|\text{fix}((01)(23))| = 0 \leq 1$, so we do select $(01)(23)$. We then find $Cks2$ containing 3 permutations:
$Cks2 = \{(123), (0123), (01)(23)\}$.

For the normalisers of the groups they generate, we find:

- $N_{\text{Aut}(G)}(\langle (123) \rangle) = \langle (23), (123) \rangle$.
- $N_{\text{Aut}(G)}(\langle (0123) \rangle) = \langle (13), (0123), (02)(13) \rangle$.
- $N_{\text{Aut}(G)}(\langle (01)(23) \rangle) = \langle (23), (01)(23), (0213) \rangle$.

Selecting only those elements of order 2, we find:

- $N2_{(123)} = \{(23), (12), (13)\}$.
- $N2_{(0123)} = \{(13), (01)(23), (02), (02)(13), (03)(12)\}$.
- $N2_{(01)(23)} = \{(23), (01), (01)(23), (02)(13), (03)(12)\}$.

Finally, as representatives of their conjugacy classes we can find the following (ignoring the unit permutation e):

- $N2cr_{(123)} = \{(23), (123)\}$.
- $N2cr_{(0123)} = \{(13), (01)(23), (0123), (02)(13)\}$.
- $N2cr_{(01)(23)} = \{(23), (01)(23), (02)(13), (0213)\}$.

Now, for every $g \in Cks2$ and $a \in N2cr_g$ we find out if it satisfies $(ga)^2 = e$. If so, we add $\langle g, a \rangle$ to Dks, as it is a 2-geometric dihedral group. We ignore those a for which $a = g$, since we would actually find a cyclic group in that case. We find:

- $((123)(23))^2 = e$. So we add $\langle(123),(23)\rangle$ to Dks.
- $((0123)(13))^2 = e, ((0123)(01)(23))^2 = e, ((0123)(02)(13))^2 = (02)(13)$. So we add $\langle(0123),(13)\rangle$ and $\langle(0123),(01)(23)\rangle$ to Dks.
- $((01)(23)(23))^2 = e, ((01)(23)(02)(13))^2 = e, ((01)(23)(0213))^2 = (01)(23)$. So we add $\langle(01)(23),(23)\rangle$ and $\langle(01)(23),(02)(13)\rangle$ to Dks.

So we have now found our list Dks containing five 2-geometric dihedral subgroups of $\text{Aut}(G)$.

In a later example, we will use these groups to choose a representation and a drawing to display them.

3.3 Displaying a 2-geometric automorphism group

3.3.1 Algorithms

Now that we are able to find all the 2-geometric subgroups of the automorphism group $\text{Aut}(G)$ of a graph $G = (V, E)$, we can design an algorithm to define a drawing of the graph in the 2-dimensional plane. We have seen that the only 2-geometric automorphism groups are the cyclic groups and the dihedral groups, and that they can be represented as rotations and reflections. The general idea is that we will draw the orbits on circles of different radii, and if there is a fixed point we will draw it at the origin.

However, let us first consider a special case where drawing orbits as circles is not possible: notice that although most 2-geometric groups have at most one fixed vertex, it is possible for a cyclic group to be 2-geometric and have more than one fixed vertex, if and only if its order is 2. This is a special case, since we cannot simply draw its fixed point at the origin as there are more than one. The idea is that we draw all its fixed points on a single line, for instance the x-axis. Now notice that all other vertices have orbit length 2. We will draw two vertices in the same orbit as two points at equal distances above and below the reflection line. If we take the x-axis as reflection line, this means that we exactly use the representation $\phi : H \to O_2(\mathbb{R})$ given in matrix form by

$$\phi(h) = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix},$$

where $H = \langle h \rangle$. A question that remains is where exactly to place these points, to make a clear picture. One option to do this will be described in Algorithm 4. In Section 3.3.2 we will look into the question whether such a drawing can be strict.

For all other cases, we know there is at most one fixed point. If there is one, we will draw it at the origin. All the other orbits are of length $|H|$ or $\frac{1}{2}|H|$, so according to Theorem 2 their representatives have a stabiliser of order 1 or 2. We first define a representation, recall from Section 3.1 that this is possible with a rotation matrix and (for dihedral groups) a reflection matrix. Also recall that the choice of the rotation matrix is not unique because of the choice for m, and that this can influence the appearance of the drawing.

With these insights, we will firstly design an algorithm to assign a representation to any 2-geometric subgroup of the automorphism group $\text{Aut}(G)$ of a graph $G = (V, E)$.

Algorithm 3. *Finding a representation for a 2-geometric subgroup*
Input: A 2-geometric subgroup H of the automorphism group $\text{Aut}(G)$ of a graph $G = (V, E)$.
Output: A representation $\phi : H \to O_2(\mathbb{R})$ of H.

1: **if** $|\text{fix}(H)| > 1$ **then**
2:    $h \leftarrow$ The generator of H (we know by construction that H must have a single generator of order 2 in this case)
3:    Define $\phi(h) \leftarrow \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$
4: **else**
5:    $Hgens \leftarrow$ The generators of H (we know by construction that there must be 1 or 2)
6:    **if** $|Hgens| = 1$ **then**
7:       $h \leftarrow$ The generator of H
8:       $k \leftarrow \text{Order}(h)$
9:       $m \leftarrow 1$ (can be any other $m < k$ coprime to k, yielding a different representation)

10:       $\theta \leftarrow \frac{2\pi m}{k}$
11:       Define $\phi(h) \leftarrow \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}$
12:       Define $\phi(h^i) = \phi(h)^i$ for $i = 2, \ldots, k$
13:    **else**
14:       $h_1 \leftarrow$ The generator of H or order 2 (we know there are two generators, at least one of them has order 2)
15:       $h_2 \leftarrow$ The other generator of H (we know it has order ≥ 2)
16:       Define $\phi(h_1) \leftarrow \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$
17:       $k \leftarrow \text{Order}(h_2)$
18:       $m \leftarrow 1$ (can be any other $m < k$ coprime to k, yielding a different representation)

19:       $\theta \leftarrow \frac{2\pi m}{k}$
20:       Define $\phi(h) \leftarrow \begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}$
21:       Define $\phi(h_1^i h_2^j) = \phi(h_1)^i \phi(h_2)^j$ for $i = 1, 2$, $j = 1, \ldots, k$
22:    **end if**
23: **end if**

As is implied by the comments in the algorithm, there is some freedom of choice which can lead to different representations. For groups with more than 1 fixed point, the angle of the reflection axis can be altered, eventually leading to a rotated drawing. For rotation matrices, different values for m coprime to k can be chosen. Since $m = 1$ is always coprime to any k, this is used as a standard value. For the effect of different choices for m, recall Section 2.6.

Once we have chosen a representation, the idea is that we draw each orbit on a circle with a different radius. For every orbit, we pick a representative v, and find its stabiliser H_v.
If $H_v = H$, then v is the fixed point so we draw it at the origin.
If the dimension of $\phi(H_v)$ is one, H_v must represent a reflection, so we want to draw v on its reflection line, so that it is indeed fixed by this reflection. The other vertices in its orbit will be drawn on the same circle around the origin, at equal distances from each other.
If H_v is the trivial group (that consist of only the identity element), then its orbit is regular and is represented by a rotation. We draw the vertices of this orbit on a circle around the origin, with equal distances from each other.

With these ideas we define the following algorithm for defining a drawing D that dis-

plays a 2-geometric automorphism group:

Algorithm 4. *Defining a drawing that displays a 2-geometric automorphism group*
Input: A 2-geometric subgroup H of the automorphism group $\mathrm{Aut}(G)$ of a graph $G = (V, E)$, a representation $\phi : H \to O_2(\mathbb{R})$.
Output: A drawing D that displays H as symmetries.

1: **if** $|\mathrm{fix}(H)| > 1$ **then**
2:   Let $v_1, v_2, \ldots, v_f$ be the fixed points of H acting on the vertex set V of the graph, and $u_1, u_2, \ldots, u_m$ representatives for the orbits of H of length 2. Here $f = |\mathrm{fix}(H)|$ and $m = \frac{|V| - f}{2}$
3:   $h \leftarrow$ The generator of H (by construction we know H has exactly one generator, of order 2)
4:   $d \leftarrow$ A vector $d \in \mathbb{R}^2$ such that $|d| = 1$ and $\phi(h)d = d$ (i.e. d is an eigenvector of $\phi(h)$ of length 1 for eigenvalue 1)
5:   **for** $i \leftarrow 1, 2, \ldots, f$ **do**
6:     $D(v_i) \leftarrow (i - \frac{1}{2}(f+1))d$ (this places the fixed points of H on the reflection line, with in-between distances of 1, centred around the origin)
7:   **end for**
8:   **for** $j = 1, 2, \ldots, m$ **do**
9:     $D(u_j) \leftarrow \frac{1}{2}(f-1)\begin{pmatrix} \cos\theta_j & -\sin\theta_j \\ \sin\theta_j & \cos\theta_j \end{pmatrix} d$, where $\theta_j = \frac{\pi}{m+1}j$ (this places the orbit representatives on a circle around the origin through v_1 and v_f, on one side of the reflection line, with equal in between distances)
10:     $D(hu_j) \leftarrow \phi(h)D(u_j)$ (this places the other element of the orbit of u_j at the reflection of $D(u_j)$ in the reflection axis)
11:   **end for**
12: **else if** $|\mathrm{fix}(H)| \leq 1$ **then**
13:   Let $v_1, v_2, \ldots, v_r$ be representatives for the orbits of H acting on the vertex set V, where r is the number of H-orbits of V.
14:   $Hgens \leftarrow$ The generators of H (we know by construction there must be 1 or 2)
15:   $rotation \leftarrow \max_{h \in Hgens}\{\mathrm{ord}(h)\}$ (if there are 2 generators, one has order 2 and the order of the other equals the number of rotations that will be displayed; if there is only 1 generator, its order also equals the number of rotations that will be displayed)
16:   $originfound \leftarrow$ FALSE (boolean that keeps track of whether a point has already been placed at the origin)
17:   **for** $i \leftarrow 1, 2, \ldots, r$ **do**
18:     **if** $originfound$ **then**
19:       $i_2 = i - 1$ (if a point has been placed at the origin, i_2 indicates the correct radius for the circle where orbit i should be drawn)
20:     **else**
21:       $i_2 = i$
22:     **end if**
23:     $H_{v_i} \leftarrow$ The stabiliser of v_i
24:     **if** $H_{v_i} = H$ **then**
25:       $D(v_i) \leftarrow (0,0)$ (if v_i is a fixed point, place it at the origin. By construction there will be at most 1 such fixed point)
26:       $originfound \leftarrow$ TRUE (ensures that upcoming orbits are drawn on circles with correct radius)
27:     **else if** H_{v_i} is the trivial group **then**
28:       $D(v_i) = i_2 \cdot (\cos\frac{\pi}{2 \cdot rotation}, \sin\frac{\pi}{2 \cdot rotation})$ (this places v_i such that it is not in the fixed point space of any non-trivial element of $\phi(H)$, see explanation below)

14

29:        **else if** H_{v_i} can be generated by one element **then**
30:          $h \leftarrow$ A generator of H
31:          $D(v_i) \leftarrow$ An eigenvector of $\phi(h)$ of length i_2, for eigenvalue 1 (so that $D(v_i)$ is fixed by $\phi(h)$)
32:        **end if**
33:        **for** every u in the H-orbit of v_i **do**
34:          Find $h \in H$ such that $u = hv_i$
35:          $D(u) \leftarrow \phi(h)D(v_i)$
36:        **end for**
37:    **end for**
38: **end if**

If $|\text{fix}(H)| > 1$, H has one generator h and $\phi(h)$ is a reflection matrix. If we use $\phi(h) = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$ as in Algorithm 3, then for d in Algorithm 4 we can simply use the vector (1,0) (however (-1,0) would also work). The fixed points are then placed on the reflection axis (which in this case would be the x-axis), with distances of 1 in between, centered around the origin. The other vertices are placed on a circle around the origin, with a radius equal to the distance of the farthest placed fixed points to the origin. They are placed at equal distance from each other, and such that they are reflected in the reflection axis. This is not to display a rotational symmetry, but to prevent edges from passing through vertices (rendering the drawing not strict).

If $|\text{fix}(H)| \leq 1$, then orbits are represented as circles around the origin, with radii $1, 2, \ldots, r$ where r is the number of orbits that contain more than 1 vertex. The vertices of each orbit are then placed on the same circle, with equal in-between distances, and such that they also display reflections (if necessary). Note that instead of $1, 2, \ldots, r$ we could use any other strictly increasing sequence as circle radii, this is purely an aesthetic choice.

If there is a fixed point, its orbit (which is only the point itself) will be placed at the origin. If in the algorithm this fixed point has index i^* (so vertex v_{i^*} is the fixed point), there would be no orbit drawn at a circle of radius i^* if we used the for-statement index i to decide the circle radius. This is where we use the boolean *originfound* to determine i_2 as either i itself, or $i - 1$. This prevents the drawing algorithm to "skip" a circle radius, which results in a better and more regular looking drawing.
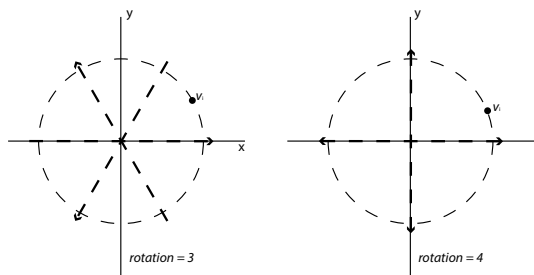
Figure 6: A reflection line rotated an odd (3) and an even (4) number of times

If the stabiliser H_{v_i} of an orbit representative v_i is the trivial group, we want to define $D(v_i)$ such that it is not in the fixed point space of any non-trivial element of $\phi(H)$. Otherwise, there would be a non-trivial element of $\phi(H)$ that fixes $D(v_i)$, and as such D would not properly display H. Now note that if H has only one generator, that it must be a cyclic group, and the fixed point space of $\phi(H)$ would either be the origin, or the empty set, depending on whether H has a fixed point. In this case we can place $D(v_i)$ anywhere

on the circle with radius i_2. So let's assume H has two generators, and is therefore a dihedral group. If we have a representation as in Algorithm 3, we see that there is a permutation $h \in H$ such that $\phi(h)$ represents a reflection in the x-axis. Furthermore, as mentioned in the algorithm, we can find $rotation$; the number of rotations that will be displayed by D. From this, we see that if we rotate the x-axis by a multiple of $\frac{\pi}{rotation}$ we find another reflection axis for another element of $\phi(H)$. Hence, we don't want to place $D(v_i)$ on any of those axes, since it would then be in the fixed point space of a non-trivial element of $\phi(H)$. Moreover, note that if $rotation$ is odd, the x-axis rotated over $\frac{2\pi}{2 \cdot rotation}$ is also a reflection axis for some element of $\phi(H)$, as it is in fact the same line as the x-axis rotated over $\lfloor \frac{rotation}{2} \rfloor \cdot \frac{2\pi}{rotation}$ (see Figure 6). By construction, for the vertices u that have a non trivial stabiliser H_u, $D(u)$ will be drawn on one of these reflection lines. For vertices u that have as stabiliser the trivial group, we want to place them such that they are not in the fixed point space of any non-trivial representation $\phi(h)$. As we have seen, for odd values of $rotation$ this also includes the x-axis rotated over $\frac{2\pi}{2 \cdot rotation}$. However, this also holds for even values of $rotation$, as these lines contain the fixed points of a rotation combined with a reflection. For aesthetic reasons, we want to place $D(v_i)$ in the middle of those fixed-point-space lines. Therefore, in the algorithm, we place D_{v_i} on the circle with radius i_2 at an angle of $\frac{\pi}{2 \cdot rotation}$. See again Figure 6 for examples where $rotation = 3$ and $rotation = 4$ respectively.

As discussed in Section 2.6, drawings can differ depending on the conjugacy class of the representation ϕ. However, with a fixed representation there are still more possible drawings for this algorithm. For the first part of the algorithm (where there are multiple fixed points), the order of the fixed points and the orbit representatives is important. Also, the choice for d influences the drawing. There are however only 2 options for d, and switching between them is equivalent to rotating the image over 180 degrees.

For the second part (where there is at most one fixed point), the order of the orbit representatives is also important. Moreover, if the stabiliser H_{v_i} of an orbit representative v_i can be generated by one element h for instance, two different eigenvectors of $\phi(h)$ of length i_2 for eigenvalue 1 can be found.

3.3.2 On strictness

An important question is whether a drawing defined by Algorithm 4 is strict, and whether it is possible at all to find such a drawing. We point out some situations where it is not possible to find a strict drawing at all.

Let H be a 2-geometric subgroup of the automorphism group $\text{Aut}(G)$ of a graph G, such that $|\text{fix}(H)| > 1$. Let G_f be the subgraph of G containing only the fixed points of H as vertices. If there is a vertex of G_f of degree greater than 2, then G_f and therefore G cannot have a strict drawing. See for instance Figure 7, of a graph with 4 vertices and edges $\{1, 2\}, \{1, 3\}, \{2, 3\}$ and $\{3, 4\}$. Note that this graph cannot be drawn strictly on one line, as should be the case for G_f. A reason for this is that vertex 3 has degree 3, which is greater than 2. If all vertices are placed on a line, it can only "receive" a maximum of 2 edges for the drawing to remain strict. Also note that if the two outermost vertices on a line have a degree greater than 1, the drawing can also not be strict. See for instance vertex 1 in Figure 7. So, to sum up, the subgraph G_f (and therefore the full graph G) cannot be strict if there are less than two points of degree 1 or smaller, or if there is a vertex of degree 3 or higher. Note that if this isn't the case, the subgraph G_f can always be drawn strictly, by correctly ordering its vertices. This does however not yet fully guarantee strictness for a drawing of the full graph.

Now let H be another 2-geometric subgroup of $\text{Aut}(G)$, such that $|\text{fix}(H) \leq 1$. To avoid the problem described above, we can try not to draw vertices on the same line. If H
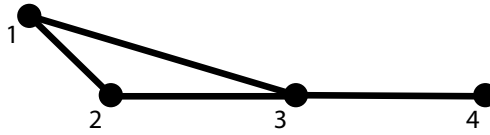
Figure 7: Not every drawing displaying an automorphism group can be strict.

is a cyclic group, so that its representations are solely rotations, we only need to draw each orbit on the same circle around the origin, where the vertices in the orbit have equal in-between distances on that circle. Thus, we can draw them in such a way that there is no line in the plane that contains more than 2 vertices, simply by (slightly) rotating the drawings of the orbit circles independently. However, if H is a dihedral group, its representations also include reflections. In this case, depending on the number of orbits, it is possible that multiple vertices from different orbits have the same stabiliser, and therefore must be drawn on the same line. In this case, a situation as in Figure 7 can occur, and it might be possible that no strict drawing can be found.

4 Drawing 3-dimensional symmetric graphs

4.1 Presentation of a group

Before we start with the classification of 3-geometric subgroups of $O_3(\mathbb{R})$, we first discuss how to define a group by a *presentation*. A group H can be presented in the form $H = \langle \mathcal{X} | \mathcal{R} \rangle$, where $\mathcal{X}$ is a list of generators for H and $\mathcal{R}$ is a list of relations between the generators. For example, the cyclic group $\mathcal{C}_k$ can be presented as $\langle g | g^n = e \rangle$, where e is the unit permutation. The dihedral group $\mathcal{D}_k$ can be presented as $\langle r, f | r^k = f^2 = (rf)^2 = e \rangle$. Here r represents a rotation and f a reflection.

4.2 Classification of 3-geometric subgroups

4.2.1 The subgroups of $SO_3(\mathbb{R})$

The finite subgroups of $SO_3(\mathbb{R})$ are the groups $\mathcal{C}_k, \mathcal{D}_k$ (for $k > 1$), $\mathcal{T}, \mathcal{O}$ and $\mathcal{I}$. Here $\mathcal{C}_k$ are the cyclic groups and $\mathcal{D}_k$ the dihedral groups, as seen in the previous section.
$\mathcal{T}$ is the tetrahedral group of order 12, which represents the rotations of a regular tetrahedron. It is isomorphic to the permutation group $\mathrm{Alt}(4)$, where $\mathrm{Alt}(n)$ is the subgroup of $\mathrm{Sym}(n)$ consisting of all even permutations of n objects.
$\mathcal{O}$ is the octahedral group of order 24, which represents the rotations of a regular octahedron. It also represents the rotations of a cube, since a cube is the dual polyhedron of an octahedron. It is isomorphic to $\mathrm{Sym}(4)$.
$\mathcal{I}$ is the icosahedral group of order 60, which represents the rotations of a regular icosahedron. It is isomorphic to $\mathrm{Alt}(5)$.
The fixed points of a cyclic rotation group form a line, its axis. Therefore, we find the following result:

Result 3. A cyclic permutation group is 3-geometric if and only if all of its orbits are trivial or regular.

In particular, we see that every cyclic group of prime order is 3-geometric. To see this, let $H = \langle g \rangle$ be a group of order p acting on a finite set X, where p is prime. Suppose H does have a non-trivial short orbit, say of size $q < p$. Then, for $x \in X$ in that orbit, we find that $g^q x = x$, and so $g^{\lceil \frac{p}{q} \rceil q - p} x = x$. However, $\lceil \frac{p}{q} \rceil q - p < q$, which contradicts the assumption that the orbit size is q. Therefore, H has no non-trivial short orbits, so every orbit is either trivial or regular, hence H is 3-geometric.
Now, let T be a rotation group that is not cyclic. As it turns out (see [4]), T has exactly three short orbits on the unit sphere. Let k_1, k_2 and k_3 respectively be the orders of the stabiliser of a point in these three orbits. Then we find the following famous formula of Jordan:

$$\frac{1}{k_1} + \frac{1}{k_2} + \frac{1}{k_3} = 1 + \frac{2}{|T|}$$

We now assume, without loss of generality, that $k_1 \leq k_2 \leq k_3$. Then for $T = \mathcal{D}_k$ we find

$$\frac{1}{k_1} + \frac{1}{k_2} + \frac{1}{k_3} = 1 + \frac{1}{k},$$

so that the only possible triple (k_1, k_2, k_3) is $(2, 2, k)$. For $T = \mathcal{T}$ we find

$$\frac{1}{k_1} + \frac{1}{k_2} + \frac{1}{k_3} = 1 + \frac{1}{6},$$

so that the only possible triple (k_1, k_2, k_3) is $(2, 3, 3)$. For $T = \mathcal{O}$ we find

$$\frac{1}{k_1} + \frac{1}{k_2} + \frac{1}{k_3} = 1 + \frac{1}{12},$$

so that the only possible triple (k_1, k_2, k_3) is $(2, 3, 4)$. For $T = \mathcal{I}$ we find

$$\frac{1}{k_1} + \frac{1}{k_2} + \frac{1}{k_3} = 1 + \frac{1}{30},$$

so that the only possible triple (k_1, k_2, k_3) is $(2, 3, 5)$. Furthermore, note that we can include the cyclic group $\mathcal{C}_k$ by associating it with the triplet $(1, k, k)$. Now, these groups have presentations (see Section 4.1)

$$\langle x, y | x^{k_1} = y^{k_2} = (xy)^{k_3} = e \rangle.$$

For type $\mathcal{D}_k$ we always assume that $k > 1$, since $\mathcal{D}_1 = \mathcal{C}_2$ (so we will consider it as a cyclic group).

From these descriptions, we find the following results.

Result 4. A dihedral group of order 4 that fixes at most one vertex can always be represented as a group of rotations.

A dihedral group of order greater than 4 that fixes at most one vertex can be represented as a group of rotations if and only if this is true for its cyclic subgroup of index 2.

Note that the tetrahedral, octahedral and icosahedral groups are based on the rotational symmetries of the Platonic Solids. Although there are five of these solids (see Figure 8), there are only three groups representing them, since the cube is the polyhedral dual of the octahedron, and the dodecahedron is the polyhedral dual of the icosahedron. The tetrahedron is its own dual. We will use these solids to try to geometrically understand the following results.

Figure 8: The Platonic Solids

Result 5. A permutation group isomorphic to $\mathcal{T}$ can be represented as a group of rotations if and only if

- it has at most one fixed vertex, and
- the lengths of its non-trivial short orbits are 4 or 6.

Note that this result excludes groups with orbits of length 3. To understand this result from a geometrical point of view, note that a regular tetrahedron has 4 vertices, 4 faces and 6 edges. Hence, a short orbit of length 4 can be associated to the rotational symmetries of a regular tetrahedron acting on one of its vertices or faces. Similarly, a short orbit of length 6 can be associated to the rotational symmetries of a regular tetrahedron acting on one of its edges.

For the next result, we first need the following definition:

Definition 8. Centraliser: The *centraliser* $C_A(H)$ of a subset H in the group A is defined as $C_A(H) = \{a \in A | ah = ha \text{ for all } h \in H\}$.

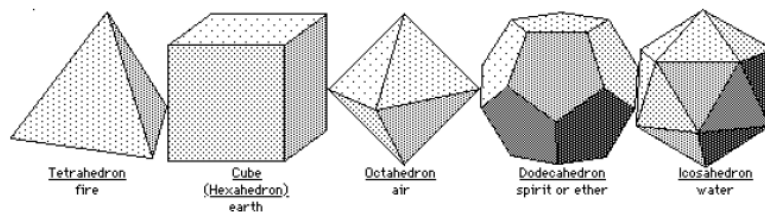**Result 6.** A permutation group isomorphic to $\mathcal{O}$ can be represented as a group of rotations if and only if

- it has at most one fixed vertex,
- the lengths of its non-trivial short orbits are 6, 8 or 12,
- the vertex stabilisers are all cyclic, and
- if an orbit of length 12 occurs, then the stabilizer of a vertex in the orbit is not contained in a normal subgroup of order 4 (equivalently, its centralizer has order 4).

Note that this excludes groups with orbits of length 3 or 4. From a geometrical point of view, we could consider the rotations of a regular octahedron. However, since $\mathcal{O}$ also represents the rotations of a cube, the dual polyhedron of a regular octahedron, we instead look at the rotational symmetries of a cube. Note that a cube has 6 faces, 8 vertices and 12 edges. Therefore, a short orbit of size 6, 8 or 12, can be associated to the rotational symmetries of a cube acting on its faces, vertices or edges respectively.

Now consider a point x, either a vertex or the centre of an edge or face of a cube, and consider all rotational symmetries of the cube that fix x (i.e. the stabiliser of x in $\mathcal{O}$). Geometrically it's not hard to see that all these rotations must be around one axis, which is the line through x and the origin. Hence, this stabiliser must be cyclic.

Result 7. A permutation group isomorphic to $\mathcal{I}$ can be represented as a group of rotations if and only if

- it has at most one fixed vertex, and
- the lengths of its non-trivial short orbits are 12, 20 or 30.

Note that this excludes groups with orbits of length 5, 6, 10 or 15. Analogue to the previous groups, we can associated the non-trivial short orders to the rotational symmetries of a regular icosahedron acting on its vertices, faces or edges respectively, as it has 12 vertices, 20 faces and 30 edges.

Lastly, note that for a permutation group to be 3-geometric, it is not necessary that all possible orbit lengths occur.

4.2.2   The subgroups of $O_3(\mathbb{R})$ not contained in $SO_3(\mathbb{R})$

Let S be a group of rotations as in the previous subsection. From such a group, we get a larger group by taking the direct product $S^* = S \times \langle z \rangle$ of S with the central inversion $z = -I$. We therefore find another result:

Result 8. Let S be a group of rotations, and $S^* = S \times \langle z \rangle$ with $z = -I$. A permutation group isomorphic to S^* is 3-geometric if and only if it can be written as the direct product of a group of type S that can be represented by rotations with a cyclic group of order 2 whose generator has at most one fixed point.

Now all we need to find are those finite groups K of $O_3(\mathbb{R})$ not contained in $SO_3(\mathbb{R})$ that do not contain the central inversion. Let K be such a group, let S be the intersection of $K \times \langle z \rangle$ with $SO_3(\mathbb{R})$ and let T be the intersection of K with $SO_3(\mathbb{R})$. Then $K \times \langle z \rangle = S \times \langle z \rangle$, and so K is isomorphic to S. What's more, T is a subgroup of index 2 in both K and S. Conversely, if S is a subgroup of $SO_3(\mathbb{R})$ that has a subgroup T of index 2, then the set $K = T \cup (S \backslash T) z$ is a group. Thus, K can be described by the symbol $(S|T)$. There are only four possibilities for the type of K: $(\mathcal{C}_{2k}|\mathcal{C}_k)$, $(\mathcal{D}_k|\mathcal{C}_k)$, $(\mathcal{D}_{2k}|\mathcal{D}_k)$ and $(\mathcal{O}|\mathcal{T})$. Note that there are no possibilities with $S = \mathcal{T}$ or $S = \mathcal{I}$. These groups are isomorphic to Alt(4) and Alt(5) respectively, which have no subgroup of index 2. Since $\mathcal{O}$ is isomorphic to Sym(4), we see immediately that $\mathcal{T}$ is a subgroup of index 2 in $\mathcal{O}$, since Alt(4) is a subgroup of index 2 in Sym(4).

Since, as we have seen before, $\mathcal{D}_1 = \mathcal{C}_2$, we find that $(\mathcal{D}_1|\mathcal{C}_1) = (\mathcal{C}_2|\mathcal{C}_1)$ and $(\mathcal{D}_2|\mathcal{D}_1) =$

$(\mathcal{D}_2|\mathcal{C}_2)$, so we can refrain from using the notation $\mathcal{D}_1$, and will again always assume $k > 1$ when considering $\mathcal{D}_k$.

Note that although the groups $\mathcal{C}_2$ and $\mathcal{C}_1^*$ are isomorphic, their representations in $O_3(\mathbb{R})$ are not conjugate; the first one is contained in $SO_3(\mathbb{R})$, whereas the second one is not. Likewise, $\mathcal{D}_2, \mathcal{C}_2^*$ and $(\mathcal{D}_2|\mathcal{C}_2)$ are isomorphic, but their representations in $O_3(\mathbb{R})$ are not conjugate; D_2 is contained in $SO_3(\mathbb{R})$ and $\mathcal{C}_2^*$ contains the central inversion.

Now, for these four types of finite subgroups K of $O_3(\mathbb{R})$ not contained in $SO_3(\mathbb{R})$ and not containing the central inversion, we find the following results:

Result 9. A cyclic group of order $2k$ is 3-geometric of type $(\mathcal{C}_{2k}|\mathcal{C}_k)$ if and only if it is generated by an element g such that

- g has at most one fixed vertex, and
- all of the non-trivial short orbits of g have lengths 2 or k if k is odd, or length 2 if k is even.

A group of type $(\mathcal{D}_k|\mathcal{C}_k)$ has a fixed axis and acts on the 2-dimensional space orthogonal to that axis. We therefore find the following result:

Result 10. A dihedral group of order $2k$ is 3-geometric of type $(\mathcal{D}_k|\mathcal{C}_k)$ if and only if all of its non-trivial short orbits have length k.

Result 11. A dihedral group of order $4k$ is 3-geometric of type $(\mathcal{D}_{2k}|\mathcal{D}_k)$ if and only if

- it has at most one fixed vertex, and
- it has a 3-geometric cyclic subgroup of index 2.

Result 12. A permutation group H isomorphic to $\mathcal{O}$ is 3-geometric of type $(\mathcal{O}|\mathcal{T})$ if and only if

- it has at most one fixed vertex, and
- the stabilisers of the vertices in the non-trivial short orbits are not contained in the (unique) normal subgroup of order 4 and are of type $\mathcal{C}_2, \mathcal{D}_2$ or $\mathcal{D}_3$.

Thus we see that the possible lengths of the non-trivial short orbits for a 3-geometric permutation group H isomorphic to $\mathcal{O}$ of type $(\mathcal{O}|\mathcal{T})$ are 4, 6 and 12. The group $(\mathcal{O}|\mathcal{T})$ is the group of all rotations and reflections of the tetrahedron.

To sum up, the finite subgroups of $O_3(\mathbb{R})$, up to conjugacy, are:

$$\mathcal{C}_k, \mathcal{D}_k, \mathcal{T}, \mathcal{O}, \mathcal{I}, \mathcal{C}_k^*, \mathcal{D}_k^*, \mathcal{T}^*, \mathcal{O}^*, \mathcal{I}^*, (\mathcal{C}_{2k}|\mathcal{C}_k), (\mathcal{D}_k|\mathcal{C}_k), (\mathcal{D}_{2k}|\mathcal{D}_k) \text{ and } (\mathcal{O}|\mathcal{T}).$$

4.3   Finding 3-geometric subgroups

As we have seen in the previous section, there are five types of 3-geometric groups that can be represented by rotations in $\mathbb{R}^3$: cyclic groups $\mathcal{C}_k$, dihedral groups $\mathcal{D}_k$, the tetrahedral group $\mathcal{T}$, the octahedral group $\mathcal{O}$ and the icosahedral group $\mathcal{I}$. We saw that $\mathcal{D}_k$ has presentation $\langle x, y | x^2 = y^2 = (xy)^k = e \rangle$ and the groups $\mathcal{T}, \mathcal{O}$ and $\mathcal{I}$ have presentations $\langle x, y | x^2 = y^3 = (xy)^j = e \rangle$ for j equals 3, 4 or 5 respectively. We will only use the type $\mathcal{D}_k$ for $k \geq 2$, since $\mathcal{D}_1 = \mathcal{C}_2$.

From each rotation group S, we can get a larger group S^* by taking the direct product of S with the central inversion $-I$. If H is a geometric subgroup of type S, we find candidates for the groups of type S^* by looking inside the centraliser of H for elements of order 2 with at most one fixed vertex. Such an element will be represented by the central inversion.

Furthermore, there are four other types that do not consist entirely of rotations. They can be described by the symbol $(S|T)$, where S and T are finite groups of rotations. This

entails that the group itself is isomorphic to a group of type S, and contains a subgroup of rotations of index 2 of type T. The four possible types are $(\mathcal{C}_{2k}|\mathcal{C}_k), (\mathcal{D}_k|\mathcal{C}_k), (\mathcal{D}_{2k}|\mathcal{D}_k)$ and $(\mathcal{O}|\mathcal{T})$.

Note that it is possible that a permutation group H can be represented as a 3-geometric group in multiple ways. For example, a cyclic group of order $4m$ that fixes at most one vertex and with all other orbits regular can be presented as the types $\mathcal{C}_{4m}$ and $(\mathcal{C}_{4m}|\mathcal{C}_{2m})$.

We start with defining an algorithm to find all 3-geometric groups (up to conjugacy) of types $\mathcal{C}_k, \mathcal{C}_k^*$ and $(\mathcal{C}_{2k}|\mathcal{C}_k)$, using Results 3, 8 and 9.

Algorithm 5. *Finding the 3-geometric groups of types $\mathcal{C}_k, \mathcal{C}_k^*$ and $(\mathcal{C}_{2k}|\mathcal{C}_k)$*
Input: A graph $G = (V, E)$, defined by its vertices V and edges E.
Output: All 3-geometric subgroups of the automorphism group $\mathrm{Aut}(G)$ of graph G of types $\mathcal{C}_k, \mathcal{C}_k^*$ and $(\mathcal{C}_{2k}|\mathcal{C}_k)$, up to conjugacy.

1: $\mathrm{Aut}(G) \leftarrow$ Automorphism group of the graph G
2: $repG \leftarrow$ List of representatives for the conjugacy classes of $\mathrm{Aut}(G)$
3: $Ck, Ckr, C2k \leftarrow$ Empty lists (used to store all 3-geometric subgroups of $\mathrm{Aut}(G)$ of types $\mathcal{C}_k, \mathcal{C}_k^*$ and $(\mathcal{C}_{2k}|\mathcal{C}_k)$ respectively)
4: **for** All g in $repG$ **do**
5:     $k \leftarrow \mathrm{order}(g)$
6:     $geometric \leftarrow$ True (initiate value)
7:     **for** All cycles c of g **do**
8:         **if** $\mathrm{length}(c) \neq 1$ **and** $\mathrm{length}(c) \neq k$ **then**
9:             $geometric \leftarrow$ False (if there is a cycle of length other than 1 or k, $\langle g \rangle$ cannot be 3-geometric)
10:         **end if**
11:     **end for**
12:     **if** $geometric$ **then**
13:         Append $\langle g \rangle$ to Ck.
14:         $C \leftarrow$ The centraliser of $\langle g \rangle$ in $\mathrm{Aut}(G)$
15:         **for** All a in C **do**
16:             **if** $\mathrm{order}(a)=2$ **and** $|\mathrm{fix}(a)| = e$ **then**
17:                 Append $\langle g, a \rangle$ to Ckr
18:             **end if**
19:         **end for**
20:         **if** k is even **then**
21:             **for** All a in C **do**
22:                 **if** $|\mathrm{fix}(a)| \leq 1$ **and** $a^2 = g$ **then**
23:                     Append $\langle a \rangle$ to $C2k$ (note that a will have cycles of length 1, 2 and/or $2k$)
24:                 **end if**
25:             **end for**
26:         **end if**
27:     **end if**
28: **end for**
29: **return** $Ck, Ckr, C2k$

For a graph $G = (V, E)$, this algorithm returns all 3-geometric subgroups (up to conjugacy) of the automorphism group $\mathrm{Aut}(G)$ of types $\mathcal{C}_k, \mathcal{C}_k^*$ and $(\mathcal{C}_{2k}|\mathcal{C}_k)$ respectively in the three lists Ck, Ckr and $C2k$.

As was the case for the algorithms for finding 2-geometric groups, no specific methods are mentioned for functions such as computing the automorphism group $\mathrm{Aut}(G)$ of a graph

G, or the centraliser $C_A(h)$ of a permutation h in the group A. These functions are implemented in **Sage**, so we can readily use them in our implementation.

We continue with describing an algorithm for finding all 3-geometric groups (up to conjugacy) of types $\mathcal{D}_k, \mathcal{D}_k^*, (\mathcal{D}_k|\mathcal{C}_k)$ and $(\mathcal{D}_{2k}|\mathcal{D}_k)$. For this, we use Results 4, 8, 10 and 11. We start by finding representatives for the conjugacy classes of 3-geometric elements, as done in the previous algorithm. For clarity, we copy the pseudocode that does this from the previous algorithm. In our implementation we simply use the results of the previous algorithm, so we do not have execute the same computations twice.

Algorithm 6. *Finding the 3-geometric groups of types $\mathcal{D}_k, \mathcal{D}_k^*, (\mathcal{D}_k|\mathcal{C}_k)$ and $(\mathcal{D}_{2k}|\mathcal{D}_k)$*
Input: A graph $G = (V, E)$, defined by its vertices V and edges E.
Output: All 3-geometric subgroups of the automorphism group $\mathrm{Aut}(G)$ of graph G of types $\mathcal{D}_k, \mathcal{D}_k^*, (\mathcal{D}_k|\mathcal{C}_k)$ and $(\mathcal{D}_{2k}|\mathcal{D}_k)$, up to conjugacy.

1: $\mathrm{Aut}(G) \leftarrow$ Automorphism group of the graph G
2: $repG \leftarrow$ List of representatives for the conjugacy classes of $\mathrm{Aut}(G)$
3: $Dk, Dkr, DkCk, D2k \leftarrow$ Empty lists (used to store all 3-geometric subgroups of $\mathrm{Aut}(G)$ of types $\mathcal{D}_k, \mathcal{D}_k^*, (\mathcal{D}_k|\mathcal{C}_k)$ and $(\mathcal{D}_{2k}|\mathcal{D}_k)$ respectively)
4: **for** All g in $repG$ **do**
5:     $k \leftarrow \mathrm{order}(g)$
6:     $geometric \leftarrow$ True (initiate value)
7:     **for** All cycles c of g **do**
8:         **if** $\mathrm{length}(c) \neq 1$ **and** $\mathrm{length}(c) \neq k$ **then**
9:             $geometric \leftarrow$ False (if there is a cycle of length other than 1 or k, $\langle g \rangle$ cannot be 3-geometric)
10:         **end if**
11:     **end for**
12:     **if** $geometric$ **then**
13:         $N \leftarrow$ The normaliser of $\langle g \rangle$ in $\mathrm{Aut}(G)$
14:         $Ncr \leftarrow$ List of representatives for the conjugacy classes of N
15:         $Ncr2 \leftarrow$ All elements of Ncr of order 2
16:         **for** all $a \in Ncr2$ **do**
17:             **if** $(ga)^2 = e$ **and** $g \neq a$ **then**
18:                 Append $\langle g, a \rangle$ to Dk
19:                 $C \leftarrow$ The centraliser of $\langle g, a \rangle$ in $\mathrm{Aut}(G)$
20:                 **for** All c in C **do**
21:                     **if** $\mathrm{order}(c)=2$ **and** $|\mathrm{fix}(c)| = e$ **then**
22:                         Append $\langle g, a, c \rangle$ to Dkr
23:                     **end if**
24:                 **end for**
25:                 **if** $|\mathrm{fix}(\langle g, a \rangle)| > 1$ **then**
26:                     $geometric \leftarrow$ True (initiate value)
27:                     **for** All orbits o of $\langle g, a \rangle$ **do**
28:                         **if** $|o| \neq 1$ **and** $|o| \neq k$ **and** $|o| \neq 2k$ **then**
29:                             $geometric \leftarrow$ False (if $\langle g, a \rangle$ has an orbit of length other than 1, k or $2k$, it cannot be 3-geometric of type $(\mathcal{D}_k|\mathcal{C}_k)$)
30:                       **end if**
31:                   **end for**
32:                 **if** $geometric$ **then**
33:                   Append $\langle g, a \rangle$ to $DkCk$
34:                 **end if**

35:         **end if**
36:         **if** $|\text{fix}(\langle g, a \rangle)| \leq 1$ **then**
37:           **if** k is even **then**
38:             $m \leftarrow \frac{k}{2}$
39:             Append $\langle g, a \rangle$ to $D2k$ ($\langle g, a \rangle$ is of type $(\mathcal{D}_{2m}|\mathcal{D}_m)$)
40:           **end if**
41:           Append $\langle g, a \rangle$ to $DkCk$ (if $\langle g, a \rangle$ fixes at most one vertex, it is of type $(\mathcal{D}_k|\mathcal{C}_k)$)
42:         **end if**
43:       **end if**
44:     **end for**
45:   **end if**
46: **end for**
47: **return** $Dk, Dkr, DkCk, D2k$

This algorithm returns four lists of permutation groups, $Dk, Dkr, DkCk$ and $D2k$, containing all 3-geometric subgroups (up to conjugacy) of the automorphism group $\text{Aut}(G)$ of a graph $G = (V, E)$ of types $\mathcal{D}_k, \mathcal{D}_k^*, (\mathcal{D}_k|\mathcal{C}_k)$ and $(\mathcal{D}_{2k}|\mathcal{D}_k)$ respectively.

Finally, we define an algorithm to find all the other 3-geometric groups, those of type $\mathcal{T}, \mathcal{O}, \mathcal{I}, \mathcal{T}^*, \mathcal{O}^*, \mathcal{I}^*$ and $(\mathcal{O}|\mathcal{T})$. For this, we make use of Results 5, 6, 7, 8 and 12.

Algorithm 7. *Finding the 3-geometric groups of types* $\mathcal{T}, \mathcal{O}, \mathcal{I}, \mathcal{T}^*, \mathcal{O}^*, \mathcal{I}^*$ *and* $(\mathcal{O}|\mathcal{T})$
Input: A graph $G = (V, E)$, defined by its vertices V and edges E.
Output: All 3-geometric subgroups of the automorphism group $\text{Aut}(G)$ of graph G of types $\mathcal{T}, \mathcal{O}, \mathcal{I}, \mathcal{T}^*, \mathcal{O}^*, \mathcal{I}^*$ and $(\mathcal{O}|\mathcal{T})$, up to conjugacy.

1: $\text{Aut}(G) \leftarrow$ Automorphism group of the graph G
2: $T, O, I, Tr, Or, Ir, OT \leftarrow$ Empty lists (used to store all 3-geometric subgroups of $\text{Aut}(G)$ of types $\mathcal{T}, \mathcal{O}, \mathcal{I}, \mathcal{T}^*, \mathcal{O}^*, \mathcal{I}^*$ and $(\mathcal{O}|\mathcal{T})$ respectively)
3: $repG \leftarrow$ List of representatives for the conjugacy classes of $\text{Aut}(G)$
4: $repG2 \leftarrow$ All elements of $repG$ of order 2
5: $AutG3 \leftarrow$ All elements of $\text{Aut}(G)$ of order 3
6: **for** All g in $repG2$ **do**
7:   $C \leftarrow$ The centraliser of g in $\text{Aut}(G)$
8:   $repG3 \leftarrow AutG3$ (initiate the list $repG3$)
9:   **while** There exist elements $h_1, h_2 \in repG3$ such that $a^{-1}h_1 a = h_2$ for some $a \in C$ **do**
10:     Delete h_2 from $repG3$ (upon termination of the while statement, $repG$ will contain representatives for the conjugacy classes of the elements of order 3 in $\text{Aut}(G)$ under the action of the centraliser C)
11:   **end while**
12:   **for** All h in $repG3$ **do**
13:     **if** order(gh)=3 **and** $|\text{fix}(\langle g, h \rangle)| \leq 1$ **then**
14:       $geometric \leftarrow$ True ($\langle g, h \rangle$ is a candidate of type $\mathcal{T}$)
15:       **for** All orbits o of $\langle g, h \rangle$ **do**
16:         **if** $|o| = 3$ **then**
17:           $geometric \leftarrow$ False ($\langle g, h \rangle$ has a non-trivial short orbit of size other than 4 or 6, so it cannot be 3-geometric)
18:         **end if**
19:       **end for**
20:       **if** $geometric$ **then**
21:         Append $\langle g, h \rangle$ to T

22:          $C \leftarrow$ The centraliser of $\langle g, h \rangle$ in $\mathrm{Aut}(G)$
23:          **for** All a in C **do**
24:             **if** order(a)=2 **and** $|\mathrm{fix}(a)| = e$ **then**
25:                Append $\langle g, h, a \rangle$ to Tr
26:             **end if**
27:          **end for**
28:       **end if**
29:    **else if** order(gh)=4 **and** $|\mathrm{fix}(\langle g, h \rangle)| \leq 1$ **then**
30:       $geometric \leftarrow$ True ($\langle g, h \rangle$ is a candidate of type $\mathcal{O}$)
31:       **for** All orbit representatives v of $\langle g, h \rangle$ **do**
32:          $S \leftarrow$ The stabiliser of v in $\langle g, h \rangle$
33:          **if** S is not cyclic **or** (order$(S) \neq 1$ **and** order$(S) \neq 2$ **and** order$(S) \neq 3$ **and** order$(S) \neq 4$ **and** order$(S) \neq 24$) **or** (order$(S) = 2$ **and** centraliser $C_{\langle g,h \rangle}(S)$ has order 4) **then**
34:             $geometric \leftarrow$ False
35:          **end if**
36:       **end for**
37:       **if** $geometric$ **then**
38:          Append $\langle g, h \rangle$ to O
39:          $C \leftarrow$ The centraliser of $\langle g, h \rangle$ in $\mathrm{Aut}(G)$
40:          **for** All a in C **do**
41:             **if** order(a)=2 **and** $|\mathrm{fix}(a)| = e$ **then**
42:                Append $\langle g, h, a \rangle$ to Or
43:             **end if**
44:          **end for**
45:       **end if**
46:       $geometric \leftarrow$ True ($\langle g, h \rangle$ is also a candidate of type $(\mathcal{O}|\mathcal{T})$)
47:       **for** All orbit representatives v of $\langle g, h \rangle$ **do**
48:          $S \leftarrow$ The stabiliser of v in $\langle g, h \rangle$
49:          **if** order$(S) \neq 1$ **and** order$(S) \neq 2$ **and** order$(S) \neq 4$ **and** order$(S) \neq 6$ **and** order$(S) \neq 24$ **then**
50:             $geometric \leftarrow$ False
51:          **else if** (order$(S) = 6$ **and** S is not of type $\mathcal{C}_2$) **or** (order$(S) = 4$ **and** S is not of type $\mathcal{C}_2$) **or** (order$(S) = 2$ **and** S is not of type $\mathcal{C}_2$) **then**
52:             $geometric \leftarrow$ False
53:          **else if** centraliser $C_{\langle g,h \rangle}(S)$ has order 4 **and** order$(S) \neq 1$ **and** order$(S) \neq 24$ **then**
54:             $geometric \leftarrow$ False
55:          **end if**
56:       **end for**
57:       **if** $geometric$ **then**
58:          Append $\langle g, h \rangle$ to OT
59:       **end if**
60:    **else if** order(gh)=5 **and** $|\mathrm{fix}(\langle g, h \rangle)| \leq 1$ **then**
61:       $geometric \leftarrow$ True ($\langle g, h \rangle$ is a candidate of type $\mathcal{I}$)
62:       **for** All orbits o of $\langle g, h \rangle$ **do**
63:          **if** $|o| = 5$ **or** $|o| = 6$ **or** $|o| = 10$ **or** $|o| = 15$ **then**
64:             $geometric \leftarrow$ False ($\langle g, h \rangle$ has a non-trivial short orbit of size other than 12, 20 or 30, so it cannot be 3-geometric)
65:          **end if**
66:       **end for**
67:       **if** $geometric$ **then**
68:          Append $\langle g, h \rangle$ to I

```
69:                C ← The centraliser of ⟨g, h⟩ in Aut(G)
70:                for All a in C do
71:                    if order(a)=2 and |fix(a)| = e then
72:                        Append ⟨g, h, a⟩ to Ir
73:                    end if
74:                end for
75:            end if
76:        end if
77:    end for
78: end for
79: find groups with reflection
80: return  T, O, I, Tr, Or, Ir, OT
```

This algorithm returns seven list of permutation groups, T, O, I, Tr, Or, Ir and OT, containing all 3-geometric subgroups (up to conjugacy) of the automorphism group $\mathrm{Aut}(G)$ of a graph $G = (V, E)$ of types $\mathcal{T}, \mathcal{O}, \mathcal{I}, \mathcal{T}^*, \mathcal{O}^*, \mathcal{I}^*$ and $(\mathcal{O}|\mathcal{T})$ respectively.

We now have three algorithms that together, given a graph $G = (V, E)$, find all 3-geometric subgroups of its automorphism group $\mathrm{Aut}(G)$.

4.4   Representations for 3-geometric automorphism groups

Given a graph $G = (V, E)$ and a 3-geometric subgroup S of $\mathrm{Aut}(G)$ or a certain type T, we need to describe how to represent each element of S, by defining an injective homomorphism $\phi : S \to T$, see Theorem 1. As we have seen, the groups of type $\mathcal{C}_k, \mathcal{D}_k, \mathcal{T}, \mathcal{O}$ and $\mathcal{I}$ will be represented purely by rotations matrices, and the groups of type $\mathcal{C}_k, \mathcal{D}_k, \mathcal{T}, \mathcal{O}$ and $\mathcal{I}$ will have an additional element represented by the central inversion $-I$. To fully define ϕ, we only need to define it for the generators of S (note that we have defined the groups S by their generators as well in the algorithms of the previous subsection). We now describe per type how to represent the generators.

If $T = \mathcal{C}_k$, then $S = \langle g \rangle$ for some generator g. We can represent this group as a rotation around the z axis, so as a representation we take

$$\phi(g) = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix},$$

where $\theta = \frac{2\pi m}{k}$ for some m coprime to k. For $T = \mathcal{C}_k^*$, we found by Algorithm 5 the group $S = \langle g, a \rangle$, for which we define $\phi(g)$ as above and $\phi(a) = -I$, the central inversion.

If $T = \mathcal{D}_k$, then by Algorithm 6 we found a group $S = \langle g, a \rangle$, with $\mathrm{order}(g) = k$ and $\mathrm{order}(a) = 2$. Here we represent the action of g as a rotation around the z axis, as we did for the cyclic groups:

$$\phi(g) = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix},$$

where $\theta = \frac{2\pi m}{k}$ for some m coprime to k. Furthermore, we represent the action of a as a 2-fold rotation around an axis perpendicular to the z-axis, for this we choose the x-axis. Hence, we find

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

For a group of type $\mathcal{D}_k^*$ (so that, by Algorithm 6, $S = \langle g, a, c \rangle$), we find the same values for $\phi(g)$ and $\phi(a)$, and we define $\phi(c)$ to be the central inversion, $\phi(c) = -I$.

A group of type $\mathcal{T}$ will be represented by the rotational symmetries of a tetrahedron. By Algorithm 7 we found a group $S = \langle g, h \rangle$, where $\mathrm{order}(g) = 2$ and $\mathrm{order}(h) = 3$. Imagine we place a regular tetrahedron with vertices A, B, C, D in a 3-dimensional space with axes x, y, z. We do this such that vertex A is on the positive z-axis, vertices B, C, D form a triangle parallel to the (x, y)-plane with negative z-coordinates, vertex B is on the (x, z)-plane and the barycentre of the vertices is in the origin. Note that vertices C and D will each be on one side of the (x, z)-plane, which gives two options; it is irrelevant for now which option we choose. Now we see there is a 3-fold rotational symmetry around the z-axis. We will therefore represent the permutation h of S as a rotation over $\frac{2\pi}{3}$ around the z axis:

$$\phi(g) = \begin{pmatrix} -\frac{1}{2} & -\frac{\sqrt{3}}{2} & 0 \\ \frac{\sqrt{3}}{2} & -\frac{1}{2} & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Furthermore, we see there is also a 2-fold rotational symmetry around the line through the centre of edge (A, B) and the centre of edge C, D (note that this line also goes through the origin). Therefore we will represent the generator g of S as a rotation over π around this line. We have already seen rotation matrices around an axis, we can define the rotation around this line by a matrix multiplication of 3 such matrices; a rotation over $\frac{2\pi}{?}$ around the y-axis over $\frac{\pi}{2} - \arctan\sqrt{2}$ (placing the centres of edges (A, B) and (C, D) on the x-axis), a rotation over π around the x-axis, and the inverse of the first rotation (around the y-axis). If we work out this multiplication, we find:

$$\phi(h) = \begin{pmatrix} -\frac{1}{3} & 0 & \frac{2\sqrt{2}}{3} \\ 0 & -1 & 0 \\ \frac{2\sqrt{2}}{3} & 0 & \frac{1}{3} \end{pmatrix}.$$

Note that the order of $(\phi(g)\phi(h))$ is indeed 3, such that this defines a correct representation of the rotational symmetries of a regular tetrahedron. Again, for type $\mathcal{T}^*$ we add the central inversion as representation for a (as in Algorithm 7); $\phi(a) = -I$.

Now consider a group of type $\mathcal{O}$. It will be represented by the rotational symmetries of a regular octahedron or a cube (its polyhedral dual). As in the previous subsection, we will consider the cube. Place a cube with edges of length $\sqrt{2}$ in a 3-dimensional space with axes x, y, z. Do this in such a way that the vertices have coordinates $(0, \pm 1, \pm\frac{1}{\sqrt{2}})$ and $(\pm 1, 0, \pm\frac{1}{\sqrt{2}})$. Now we see that there is a 2-fold rotational symmetry around the x-axis, which we will use to represent the generator g (as defined in Algorithm 7):

$$\phi(g) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{pmatrix}.$$

Furthermore we search for a 3-fold rotational symmetry for the representation $\phi(h)$, such that the order of $\phi(g)\phi(h)$ is 4. We find this by taking a rotation over $\frac{2\pi}{3}$ around the line through coordinates $(-1, 0, -\frac{1}{\sqrt{2}})$ and $(1, 0, \frac{1}{\sqrt{2}})$. Similar to what we did for the tetrahedron, we find the matrix for this my multiplying three matrices, those representing a rotation over $-\arctan\frac{1}{\sqrt{2}}$ around the y-axis, a rotation over $\frac{2\pi}{3}$ around the x-axis and the inverse of the first rotation (around the y-axis). Working out this multiplication, we find

$$\phi(h) = \begin{pmatrix} 0 & -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} & -\frac{1}{2} & \frac{1}{2} \\ -\frac{1}{\sqrt{2}} & -\frac{1}{2} & \frac{1}{2} \end{pmatrix}.$$

Again, for type $\mathcal{O}^*$ we represent a as found in Algorithm 7 by the central inversion, $\phi(a) = -I$.

4.5 Displaying a 3-geometric automorphism group

To generate a drawing of a graph $G = (V, E)$ that displays a 3-geometric subgroup H of the automorphism group $\mathrm{Aut}(G)$, given a representation ϕ, we use the same principles as for 2 dimensions, and extend them to three dimensions. Therefore, for each orbit representative v_i we first find the stabiliser H_{v_i}. If it equals the group H itself, we have found the one fixed point, so we will place it at the origin. If H_{v_i} can be generated by one (non-trivial) generator, we draw v_i such that it is fixed by $\phi(H_{v_i})$. To make sure we draw each orbit at a different distance to the origin, we take a vector of length i. If H_{v_i} is generated by 2 generators, then one of these generators must be a reflection, say r. We now draw v_i at a point of distance i to the origin such that it is fixed by r, and not fixed by any other element in a subgroup of $\phi(H)$ containing r. If H_{v_i} is the trivial group, we draw it such that it is not fixed by any (non-trivial) element of $\phi(H)$, so that it will indeed have a regular orbit. This gives is the following general algorithm to define a drawing:

Algorithm 8. *Displaying a 3-geometric automorphism group* **Input:** A 3-geometric subgroup H of the automorphism group $\mathrm{Aut}(G)$ of a graph $G = (V, E)$, a representation $\phi : H \to O_3(\mathbb{R})$.
Output: A drawing D that displays H as symmetries.

> $v_1, v_2, \ldots, v_n \leftarrow$ Representatives of the orbits of H acting on V.
> **for** $i \leftarrow 1$ **to** n **do**
> > $H_{v_i} \leftarrow$ Stabiliser of v_i
> > **if** $H_{v_i} = H$ **then**
> > > $D(v_i) \leftarrow$ The origin
> > **else if** $\phi(H_{v_i})$ is generated by one generator **then**
> > > $D(v_i) \leftarrow$ A eigenvector of $\phi(H_{v_i})$ of length i for eigenvalue 1.
> > **else if** $\phi(H_{v_i})$ is generated by two generators **then**
> > > $r \leftarrow$ The reflection generating $\phi(H_{v_i})$
> > > $D(v_i) \leftarrow$ A vector of length i, fixed by r, and not fixed by any other element in a subgroup of $\phi(H)$ containing r
> > **else if** $\phi(H_{v_i})$ is the trivial group **then**
> > > $D(v_i) \leftarrow$ A vector of length i not in the fixed point space of any non-trivial element of $\phi(H)$
> > **end if**
> > **for** All v in the orbit of v_i **do**
> > > $h \leftarrow$ The permutation such that $v = hv_i$
> > > $D(v) \leftarrow \phi(h)D(v_i)$
> > **end for**
> **end for**

The exact details for this algorithm are different for the different types of groups, and there is still quite some freedom in choice, in for instance the ordering of the orbit representatives. In particular, a vertex with the trivial group as stabiliser gives a lot of choice.

5   Conclusion

Given a graph $G = (V, E)$, we have seen what n-geometric subgroups of the automorphism group $\text{Aut}(G)$ are, and how they can be used to provide nice symmetrical drawings of G. For $n = 2, 3$ we have seen how to find all geometric subgroups, using group-theoretic arguments. Then, given such a geometric group, we have seen how to define a drawing that displays this group. We have implemented these methods using **Sage**.

6 Neat examples

6.1 The Petersen Graph; an extensive example

The Petersen graph is a famous example and counterexample in graph theory. It is a strongly regular graph. It is often drawn as a pentagram within a pentagon, where every corner points of the pentagram is connected with a corner point of the pentagon, as in Figure 10d. In this drawing, it clearly displays a rotation over $\frac{2\pi}{5}$, along with a reflection. So with the knowledge of this article, we know that the automorphism group of the Petersen graph contains a 2-geometric dihedral subgroup of order 10. There are however more symmetries that can be displayed, using different drawings.

In Figure 9 we see 3 different drawings of the Petersen, which are a direct result of our own implementation (see Appendix). All 3 drawings display a different 2-geometric automorphism group that is cyclic. Figure 9a and Figure 9b display groups of order 2, their representations are realised by a reflection in the x-axis (the central horizontal line). Both of the groups have more than one fixed point, which is why they cannot be represented by a rotation. We can see that the drawing in Figure 9a is not strict; vertex 2 is drawn on edge {1,6}. As we can see from this picture, 0, 1, 2 and 6 are the fixed points of the cyclic group that is being displayed. Since in the subgraph on 0, 1, 2 and 6, vertex 1 has degree 3, we find that it is not possible to find a strict drawing that displays this cyclic group, as we have seen in Section 3.3.2. The drawing in Figure 9b however is strict. In Figure 9c we see a drawing that displays a cyclic group of order 3, represented by a rotation over $\frac{2\pi}{3}$. This drawing is strict.
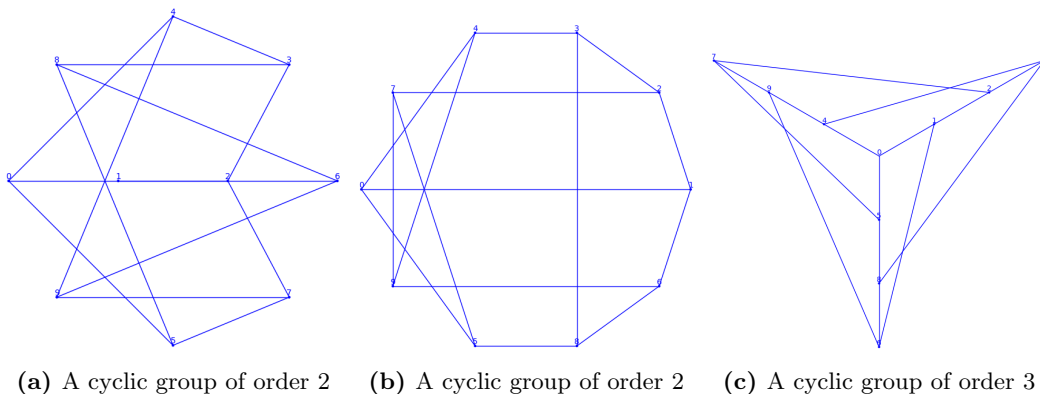

(a) A cyclic group of order 2   **(b)** A cyclic group of order 2   **(c)** A cyclic group of order 3

Figure 9: Several drawings of the Petersen graph displaying different 2-geometric cyclic groups

Now take a look at Figure 10. Here we see several drawings of the Petersen graph, displaying different dihedral groups. The drawings in Figure 10a and 10b display two different dihedral groups of order 6, by a rotation over $\frac{2\pi}{3}$ and a reflection. Neither of these drawings are strict. In Figure 10a we see again the same problem as described in Section 3.3.2. For instance the vertices 0, 1, 2 and 6 must be drawn on one line, as they have the same stabiliser. In the subgraph containing exactly those vertices, vertex 1 has degree 3, making it impossible to find a strict drawing. In Figure 10b we see a different problem. The edges {2,7}, {3,8} and {6,9} are drawn such that they go through vertex 0.

In Figure 10c we see a drawing displaying a 2-geometric dihedral group of order 10, represented by a rotation over $\frac{2\pi}{5}$ and a reflection. This drawing is clearly strict. Using our algorithm, we find that the Petersen graph has a total of 3 different 2-geometric dihedral groups (up to conjugacy). However, we have not yet found the famous representation as can be seen in Figure 10d. Recall that in Algorithm 3, in step 9 we can choose any

$m < k$ coprime to k. For Figure 10c, we used $m = 1$. However, as we have seen in Section 2.6, choosing a different representation may influence the drawing. Now, if we use $m = 3$ instead (note that $k = 5$, so that this is the only other option), we find the quite famous drawing of the Petersen graph as shown in Figure 10d.
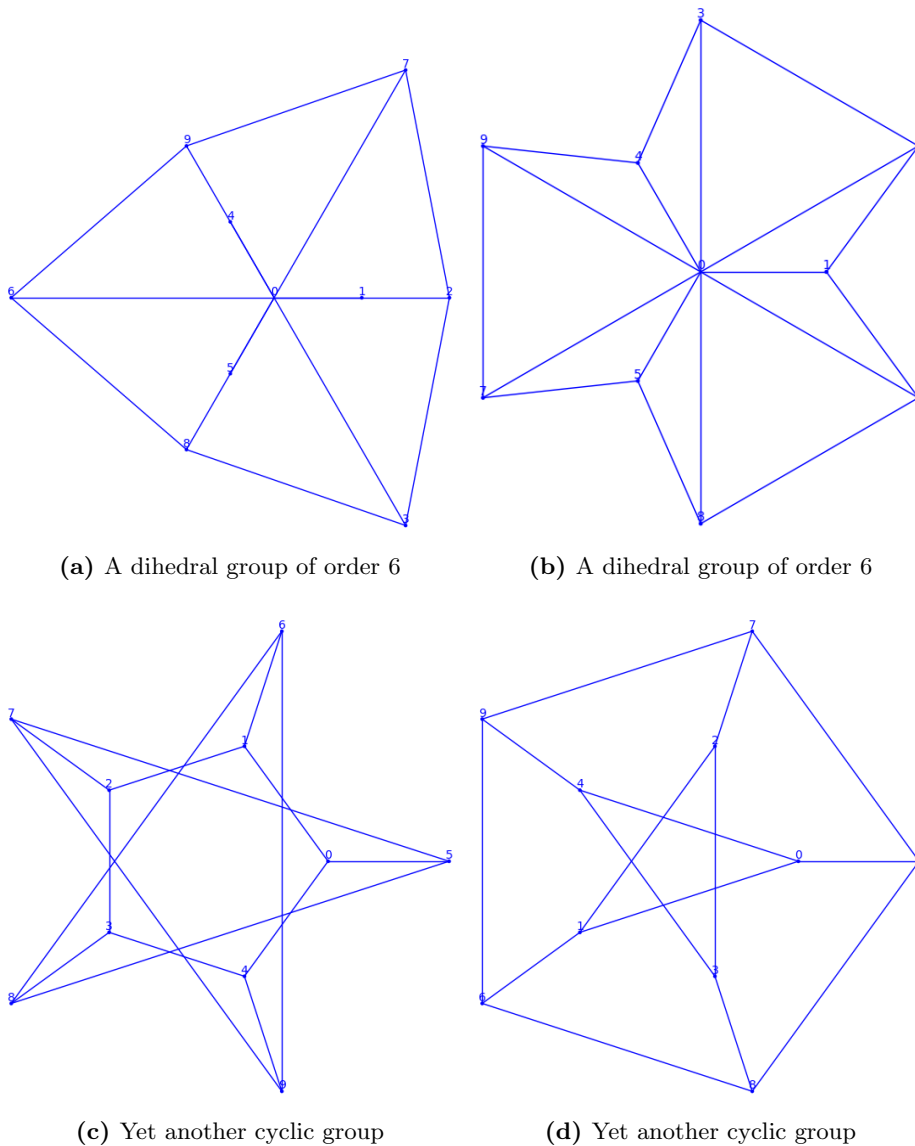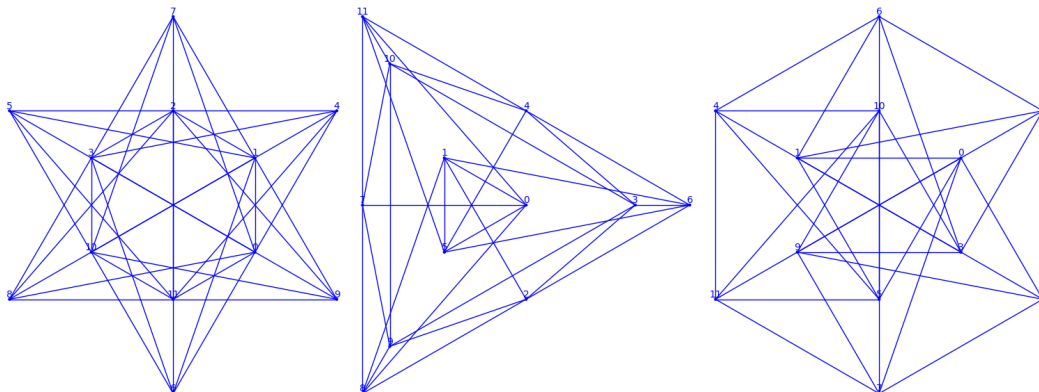

(a) A dihedral group of order 6

(b) A dihedral group of order 6

(c) Yet another cyclic group

(d) Yet another cyclic group

Figure 10: Several drawings of the Petersen graph displaying different 2-geometric dihedral groups

6.2 2-Dimensional examples

In Figure 11 we see three drawings of the Icosahedral graph (representing the vertices and edges of a icosahedron) displaying 2-geometric dihedral groups of order 12, 6 and 6 respectively. They are represented by rotations and a reflection. Only the drawing in

Figure 11b is strict. The icosahedral graph has 12 vertices and 30 edges.

(a) A dihedral group of order 12 (b) A dihedral group of order 6 (c) A dihedral group of order 6

Figure 11: Several drawings of the Icosahedral graph displaying different 2-geometric dihedral groups

In Figure 12 we see a drawing of the Schläfli graph, displaying a dihedral group of order 18. The Schläfli graph is a strongly regular graph with 27 vertices and 216 edges.
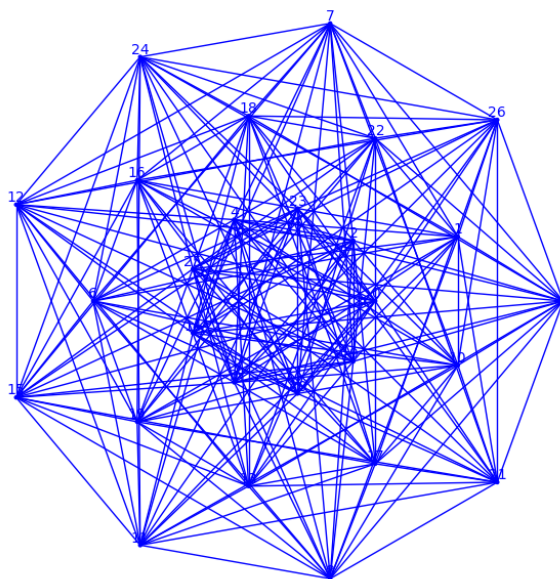

Figure 12: A drawing of the Schläfli graph, displaying a dihedral group of order 18

Now let's take a look at a larger graph. The M22 graphis a strongly regular graph with 77 vertices and 616 edges. Using our algorithm, we find that its automorphism contains (among others) a 2-geometric dihedral subgroup of order 22. The drawing in Figure 13 displays this group.
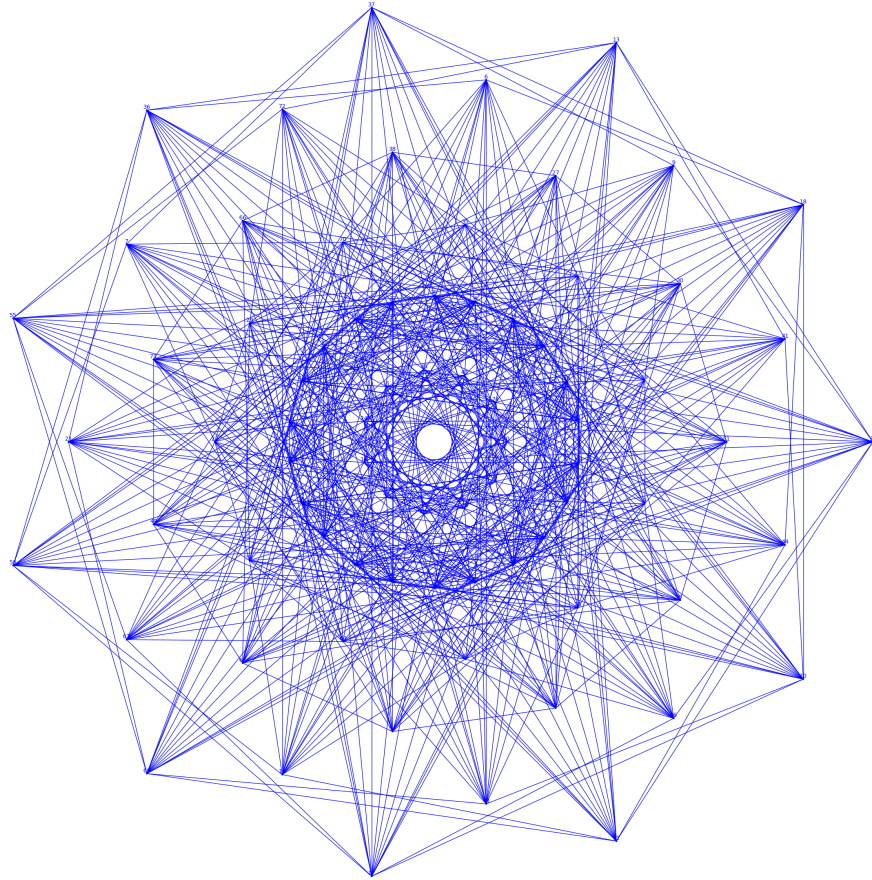
Figure 13: A drawing of the M22 graph, displaying a dihedral group of order 22

References

[1] D. Abelson, S. Hong and D. Taylor, A Group-Theoretic Method for Drawing Graphs Symmetrically.

[2] P. Eades and X. Lin, Spring Algorithms and Symmetries, *Theoretical Computer Sciences*, 240, pp. 379-405, 2000.

[3] W. Ledermann, *Introduction to Group Theory*, Longman, 1973

[4] L. Grove and C. Benson, *Finite Reflection Groups*, Springer, 1971

[5] GAP, `http://www.gap-system.org`

[6] Sage, `http://www.sagemath.org`

[7] nauty, `http://pallini.di.uniroma1.it`

[8] bliss, `http://www.tcs.hut.fi/Software/bliss/`

[9] Saucy, `http://vlsicad.eecs.umich.edu/BK/SAUCY/`

Appendix

A Sage source code for 2 dimensions

```
# AUXILIARY FUNCTIONS
# FUNCTION "fixp": returns fixed points of permutation in cycle notation
def fixp(cycles):
    c = cycles.cycle_tuples(singletons=True)
    f = []
    for i in range(len(c)):
        if len(c[i])==1:
            f.append(c[i])
    return f


# FUNCTION "eq_cycles": checks if all (non-singleton) cycles have length m
def eq_cycles(cycles,m):
    c = map(len,cycles.cycle_tuples())
    if len(c)==0:
        return True
    else:
        return c.count(m)==len(c)



# input graph G
#G = graphs.PetersenGraph()
#G = graphs.CompleteGraph(4)
G = graphs.IcosahedralGraph()
#G = graphs.M22Graph()
#G = graphs.SchlaefliGraph()



# ALGORITHM: Find all 2-geometric subgroups of automorphism group Aut(G) of graph G
# INPUT: Graph G
# OUTPUT: List Cks of cyclic and Dks of dihedral 2-geometric subgroups of Aut(G)

# INITIATE
[AutG,orbitsG] = G.automorphism_group(return_group=True, orbits=True)
V = G.vertices()

# FIND 2-GEOMETRIC CYCLIC GROUPS
# find upper bound (up) for order of 2-geometric cyclic group
ub = max(map(len,orbitsG))
# find representatives "repG" for the conjugacy classes of Aut(G)
repG = AutG.conjugacy_classes_representatives()
# Put all elements (of order 2) AND (order m>2 with at most one fixed vertex and all
 other cycles of length m) in "Cks"
# for next algorithm: also require elements of order 2 to fix at most one vertex, put
 in "Cks2"
Cks = []
Cks2 = []
for i in range(len(repG)):
```

```
        m = order(repG[i])
        if m>=2 and m<=ub and len(fixp(repG[i]))<=1 and eq_cycles(repG[i],m):
            Cks.append(PermutationGroup([repG[i]],domain=V))
            Cks2.append(PermutationGroup([repG[i]],domain=V))
        elif m==2:
            Cks.append(PermutationGroup([repG[i]],domain=V))


# FIND 2-GEOMETRIC DIHEDRAL GROUPS (uses previous algorithm)
# use Cks2 as found in previous algorithm
# For all g in Cks2, compute normalizer "N" of <g> in AutG
Dks = []
for i in range(len(Cks2)):
    N = AutG.normalizer(Cks2[i])
    Ncr = N.conjugacy_classes_representatives()
    Ncr2 = []
    for j in range(len(Ncr)): # "Ncr2" conjugacy class representatives of order 2 in N
        if Ncr[j].order()==2:
            Ncr2.append(Ncr[j])
    for j in range(len(Ncr2)):
        if AutG(Ncr2[j]).cycles()!=[] and AutG(Ncr2[j])!=AutG(Cks2[i].gen()) and
           ((AutG(Ncr2[j])*AutG(Cks2[i].gen()))^2).cycles()==[]:
            Dks.append(PermutationGroup([Cks2[i].gen(),Ncr2[j]],domain=V)) #accept those
               elements for which (ga^2)=e and a!=e


# print cyclic and dihedral 2-geometric subgroups of Aut(G)
len(Cks); Cks; len(Dks); Dks




# input
H=Dks[8]

# ALGORITHM: Choose representation phi for 2-geometric automorphism group H
# INPUT: 2-geometric automorphism group H
# OUTPUT: Representation phi for 2-geometric automorphism group H
fixH = H.fixed_points()
if len(fixH)>1:
    h=H.gen()
    phi = {}
    phi[h]=matrix([[1,0],[0,-1]])  # use x-axis as reflection line
else:
    Hgens=H.gens()
    phi = {}
    if len(Hgens)==1:
        reflectionfound = True   #cyclic group, no reflection needed
    else:
        reflectionfound = False  #in case of dihedral group, one reflection needed
    for i in range(len(Hgens)):
        if reflectionfound==False and Hgens[i].order()==2:
            phi[Hgens[i]]=matrix([[1,0],[0,-1]])
```

```
                reflectionfound = True
        else:
            k=Hgens[i].order()
            m=1                           # choose m coprime with k, multipe choices!
                1 always works
            theta = 2*pi*m/k
            phi[Hgens[i]]=matrix([[cos(theta),-sin(theta)],[sin(theta),cos(theta)]])
    for i in range(Hgens[0].order()):
        phi[Hgens[0]^i]=phi[Hgens[0]]^i        # define phi for all powers of the
         first generator
        if len(Hgens)==2:                      # if there is a second generator
            for j in range(Hgens[1].order()-1):
                phi[Hgens[0]^i*Hgens[1]^(j+1)]=phi[Hgens[1]]^(j+1)*phi[Hgens[0]]^i
                    # define phi for all




# ALGORITHM: Define drawing D that displays the 2-geometric automorphism group H
# INPUT: 2-geometric automorphism group H (domain H = vertices G). (if |fix(H)|>1,
 assumes reflection in x-axis as representation)
# OUTPUT: drawing D defined on the domain of H (should equal vertices of G) that
 displays H
D={}
fixH = H.fixed_points()
lenfixH = len(fixH)
orbH=H.orbits()
if lenfixH>1:
    orbreprH=[]
    for i in range(len(orbH)):
        if len(orbH[i])==2:
            orbreprH.append(orbH[i][0])
    m=len(orbreprH)
    h=H.gen()
    d=vector([1,0])                       # assume reflection in x-axis as
     representation
    for i in range(lenfixH):              # place fixed points on line
        D[fixH[i]]=(i-1/2*(lenfixH-1))*d
    for j in range(m):                    # place orbits around reflection axis
     (in a circle)
        thetaj=pi*(j+1)/(m+1)
        D[orbreprH[j]]=1/2*(lenfixH-1)*matrix([[cos(thetaj),-sin(thetaj)],
         [sin(thetaj),cos(thetaj)]])*d
        D[h(orbreprH[j])]=phi[h]*D[orbreprH[j]]
else:
    Hgens = H.gens()
    rotation = max(map(order,Hgens))
    originfound = false          # checks if a point has already been placed at origin
    for i in range(len(orbH)):   # for all orbit representatives
        if originfound:          # ensures distances between consecutive orbit circles
         are all 1
            i2=i
```

```
            else:
                i2=i+1
        v = orbH[i][0]
        stab = H.stabilizer(v)    # compute stabilizer
        stabgens = stab.gens()    # find generators of stabilizer (max 2)
        if stab==H:               # holds for a maximum of one vertex (the fixed point)
            D[v]=vector([0,0])
            originfound = true
        elif stabgens[0].cycles()==[]:
            D[v]=i2*vector([cos(pi/(2*rotation)),sin(pi/(2*rotation))])
        elif len(stabgens)==1:
            stabphi = phi[stabgens[0]]
            stabphieigv = stabphi.eigenvalues()
            for j in range(len(stabphieigv)):
                if type(stabphieigv[j])==type(cos(1)):      # if stapphieigv[j] is
                 a (trigonometric) expression
                    stabphieigv[j] = stabphieigv[j].simplify_trig()    # simplify
                     eigenvectors to find value 1
            eigv1=stabphi.eigenvectors_right()[stabphieigv.index(1)][1][0] # find
             eigenvector with value 1
            D[v]=i2*eigv1/eigv1.norm()                              # of length i2
        for j in range(len(orbH[i])-1):  # for all u in H-orbit of v
            for l in range(H.order()):    # find h in H such that hv=u
                if H[l](v)==orbH[i][j+1]:
                    D[orbH[i][j+1]]=phi[H[l]]*D[v]
                    break
```

```
# ALGORITHM: plot drawing D of graph G
# INPUT: graph G, drawing D (defined on G.vertices())
# OUTPUT: drawing L with edges as lines and vertices as labeled points
L=line((0,0))
for v in G.vertices():
    L=L+point(D[v])+text(v,D[v],vertical_alignment='bottom')
for e in range(len(G.edges())):
    L=L+line([D[G.edges()[e][0]],D[G.edges()[e][1]]])
L.set_aspect_ratio(1)
L.axes(false)
L
```