Compound Symmetry Groups

by

Jinny Wu

to obtain the degree of Bachelor of Science
at the Delft University of Technology,
to be defended publicly on Wednesday June 25, 2025 at 16:00 PM.

Student number:	5751071	
Project duration:	April 22, 2025 – June 25, 2025	
Thesis committee:	Dr. N.D. (Nikolaas) Verhulst,	TU Delft, supervisor
	Dr.Ir. W.G.M. (Wolter) Groenevelt,	TU Delft, senior

Laymen's summary

The study of compound symmetry groups is a branch of mathematics that has only recently emerged. The study of these groups began with the *Gizmo Gears* puzzle designed by Douglas Engel. The goal of this puzzle is to rotate two overlapping gears until all pieces are in the correct place. The difficulty lies in the fact that if one gear is turned, a part of the other gear is also moved because they overlap. All the possible configurations the gears can take when rotated can be studied using a branch of mathematics called group theory. For a more general example of this puzzle, consider two overlapping disks, each of which can be rotated 90° degrees. As the disks are rotated, they are cut up into pieces, as rotating one disk also moves a piece of the other disk. On a surface level, compound symmetry groups can be described as a collection of all combinations of rotations that result in a new permutation of the disk pieces. As the overlap between the disks increases, more pieces are created under their rotations. From a certain amount of overlap, more formally called the critical radius, the number of pieces may even become infinite. Several estimates are given for where this critical radius lies, along with results that characterise the existence of such radii. Furthermore, compound symmetry groups can be generalised by considering an arbitrary amount of disks. For three or more disks, we only have numerical estimates for the critical radii, which are found using a brute-force approach.

Summary

In [4], compound symmetry groups are defined as "groups generated by a set of isometries of subspaces of a metric space." Here the main focus is on groups generated by discrete rotations of overlapping disks in the plane. The simplest case occurs when only two disks are involved, which are called two-disk systems. The groups corresponding to these systems are denoted as $GG_{n_1,n_2}(r)$, where n_1, n_2 represents the rotational symmetry of the left and right disk respectively, and r the radius of both disks. When the disks are rotated, they are partitioned into pieces. Below a certain radius of the disks, we can label the pieces and express the rotations as permutations, from which we can identify the group structure. However, after a certain radius, the group may become infinite, and we will call this radius the critical radius. We know that there exists some r for which $GG_{n_1,n_2}(r)$ is infinite if and only if $\mathrm{lcm}(n_1, n_2) \notin \{1, 2, 3, 4, 6\}$ (see theorem 2). For some $n = n_1 = n_2$ there exists an algebraic expression for the critical radius (see theorem 3), but for most n only numerical estimates are known. Two-disk systems can be extended to three-disk systems, and generalised to k-disk systems. Estimates of critical radii for three-disk systems can be found in table 4.1. These estimates are found using a brute-force approach.

Contents

1
Introduction

Group theory plays a prominent role in many areas of science, from crystallography to cryptography to topology. A particularly interesting concept in group theory is the symmetry group of a pattern in the plane, which consists of isometries (such as rotations or reflections) that leave the pattern unchanged. In this paper the general notion of these kinds of symmetry is expanded, by defining discrete rotations on disks that overlap in the Euclidean plane. The groups generated by these rotations are called compound symmetry groups. When the radii of the disks are increased, more overlap is created, giving rise to more complex group structures. From a certain radius, which we will call the critical radius, the group can even become infinite.

Compound symmetry groups which arise from two overlapping disks have been analysed in [4], where basic properties and the cardinality of these groups are discussed. Several questions that remain are the structure of finite compound symmetry groups and the behaviour of three (or more) overlapping disks. Thus, the goal of this thesis is to formalize the results from [4] and extend their research by considering the aforementioned questions.

In chapter 2, foundational concepts from group theory are recalled. Then, in chapter 3, terminology on compound symmetry groups is defined, and the behaviour of these groups within the framework of two overlapping disks is analysed. In chapter 4, the study on two-disk systems is extended by exploring k-disk systems, and estimates are given for the critical radii of three-disk systems. In chapter 5, numerical algorithms are explained concerning the generation of orbit patterns and the estimates for the critical radii. Lastly, in chapter 6 the results are stated and suggestions for future research are given.

2 Preliminaries

The study of compound symmetry groups is based on fundamental notions in group theory. In this chapter we recall some theoretical background knowledge from the fields of algebra and real analysis. In section 2.1 groups are defined. In section 2.2 symmetry groups are discussed, which are a stepping stone to the study of compound symmetry groups.

2.1. Groups

Definition 1. A *group* is a set G with an operation $G \times G \to G$, denoted as $(a, b) \mapsto a \circ b$, which satisfies the following three group axioms:

(G1) $\forall\, a, b, c \in G \quad a \circ (b \circ c) = (a \circ b) \circ c$ (Associativity)

(G2) $\exists\, e \in G: \quad \forall\, a \in G \quad e \circ a = a \circ e = a$ (Identity element)

(G3) $\forall\, a \in G \quad \exists\, a^* \in G: \quad a \circ a^* = a^* \circ a = e$ (Inverse element)

For the group operation it is conventional to use multiplicative notation. That means we will write $a \cdot b$ or ab instead of $a \circ b$

Definition 2. Let G_1 and G_2 be groups. A map $f : G_1 \to G_2$ is called a *homomorphism* if for all $a, b \in G_1$ it holds that $f(ab) = f(a)f(b)$. G_1 and G_2 are *isomorphic*, where we write $G_1 \cong G_2$, if there exists a bijective homomorphism from G_1 to G_2.

Definition 3. Let G be a group and $S = \{s_1, s_2, \ldots\} \subseteq G$. Then S is a *generating set* for G if for all $g \in G$ there exists $x_1, x_2, \ldots, x_n \in S$ and $\epsilon_1, \epsilon_2, \ldots, \epsilon_n \in \{-1, 1\}$ for some $n \in \mathbb{N}$ such that $g = x_1^{\epsilon_1} x_2^{\epsilon_2} \ldots x_n^{\epsilon_n}$. The elements of S are called *generators* of G and we write $G = \langle S \rangle$ or $G = \langle s_1, s_2, \ldots \rangle$

Example 1. (Cyclic groups) Examples of groups are $\mathbb{Z}$ and $\mathbb{Z}/n\mathbb{Z}$ with + as the operation. These groups are *cyclic*. That is, there exists $x \in G$ such that $G = \langle x \rangle$. We denote a cyclic group of n elements by C_n.

Definition 4. Let G be a group and X be a set. Then G *acts on* X if for all $g \in G$ and for all $x \in X$ there exists $g \circ x \in X$ such that:

(W1) $\forall\, x \in X \quad e \circ x = x$

(W2) $\forall\, g,h \in G, x \in X \quad (g \circ h) \circ x = g \circ (h \circ x)$

Definition 5. Let the group G act on the set X. Let $x \in X$. Then the set

$$Gx = \{g \circ x : g \in G\}$$

is called the *orbit* of x under G

Example 2. Let X be the unit disk and $G = C_4$ be the cyclic group of order 4 that acts on X by 90° counter-clockwise rotations. Then

$$G(1,0) = \{(1,0),(0,1),(-1,0),(0,-1)\}$$

and

$$G(0,0) = \{(0,0)\}$$

Theorem 1. *(Bézout's identity) Let $a, b \in \mathbb{Z}$. Then there exist $x, y \in \mathbb{Z}$ such that*

$$xa + yb = gcd(a,b)$$

Proof. Cf. e.g. [13], p.6. □

2.2. Symmetry groups

The symmetry group of an object consists of transformations that map the object to itself. To build these symmetry groups, we need to define metric spaces and groups of maps.

Definition 6. For any set X, the set

$$S(X) = \{f : X \to X \mid f \text{ is bijective}\}$$

with function composition as the operation forms a group, called a *symmetric group*.

Definition 7. A *metric space* is a couple (M,d), where M is a set and $d : M \times M \mapsto [0,\infty)$ is a *metric* on M. That is, d is a function that satisfies the following properties:

(a) $\forall\, x, y \in M \quad 0 \le d(x,y) < \infty$

(b) $d(x,y) = 0 \Leftrightarrow x = y$

(c) $\forall\, x, y, \in M \quad d(x,y) = d(y,x)$

(d) $\forall\, x, y, z \in M \quad d(x,y) \le d(x,z) + d(z,y)$

Note that every subset $A \subseteq M$ together with the metric d on M is again a metric space. (A,d) is also called a *subspace* of (M,d).

Definition 8. An *isometry* of a metric space (M,d) is a map $\sigma : M \to M$ such that for all $p,q \in M$ we have that

$$d(p,q) = d(\sigma(p),\sigma(q))$$

Example 3. (Isometries of the Euclidean plane) Let $(\mathbb{C},d)$ be a metric space, where $d : \mathbb{C} \to \mathbb{C}$ defined by $d(z_1,z_2) = |z_1 - z_2|$ is the Euclidean metric. Then examples of isometries are:

(a) translations: for $z_0 \in \mathbb{C}$, the map $t_{z_0} : \mathbb{C} \to \mathbb{C}$ given by $t_{z_0}(z) = z + z_0$ is called a translation over z_0.

(b) rotations of the plane around the origin: for $\varphi \in \mathbb{R}$, the map $\rho_\varphi : \mathbb{C} \to \mathbb{C}$ given by $\rho_\varphi(z) = e^{\varphi i} z$ rotates the plane counter-clockwise around the origin over an angle φ.

(c) reflections across a line through the origin: for $\varphi \in \mathbb{R}$, the map $\sigma_\varphi : \mathbb{C} \to \mathbb{C}$ given by $\sigma_\varphi(z) = e^{\varphi i}\overline{e^{-\varphi i} z}$ reflects the plane across the line through the origin that intersects the x-axis at an angle φ.

The set of all isometries of the Euclidean plane forms a group, denoted $E(\mathbb{R}^2)$. If we restrict ourselves to the isometries of the plane that fix the origin, we get the orthogonal group $O_2(\mathbb{R})$. An equivalent formulation of these isometries exists if we take $\mathbb{R}^2$ with the Euclidean metric instead of $\mathbb{C}$, which involves matrices. However, in the context of compound symmetry groups, it is more convenient to work in the complex plane.

Lemma 1. *The composition of rotations (about points which may be different) is either a rotation or a translation.*

Proof. Cf. e.g. [11], p.5. □

Corollary 1. *Consider a composition of rotations, and let φ be the sum of the individual rotation angles. If $\varphi = 0 \bmod 2\pi$, then the composition is a translation. Otherwise, the composition is a rotation with rotation angle φ.*

Proof. Cf. e.g. [11], p.5. □

Example 4. (Dihedral groups) The dihedral group D_n with $n \in \mathbb{Z}_{\geq 2}$, consists of the isometries of $\mathbb{C}$ that map a regular n-gon onto itself. In particular, if we take the origin as the centre of the n-gon and $(1,0)$ as one of its vertices, then

$$D_n = \{\rho_0, \rho_{2\pi/n}, \rho_{4\pi/n}, \ldots, \rho_{(2n-2)\pi/n}\} \cup \{\sigma_0, \sigma_{\pi/n}, \sigma_{2\pi/n}, \ldots, \sigma_{(n-1)\pi/n}\}$$

Note that in this case D_n consists of n rotations and n reflections.

Definition 9. A *lattice* in $\mathbb{R}^n$ is a subset

$$L = \{a_1\mathbf{v}_1 + a_2\mathbf{v}_2 + \cdots + a_n\mathbf{v}_n \mid a_1, a_2, \ldots, a_n \in \mathbb{Z}\}$$

where $\{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n\}$ is a basis of $\mathbb{R}^n$.

Definition 10. Let H be any subgroup of $E(\mathbb{R}^2)$. The *translation subgroup* $\mathrm{Trans}(H)$ consists of all translations in H. The *point group* $\mathrm{Point}(H)$ consists of all rotations and reflections in H. That is, $\mathrm{Point}(H) \cong H/\mathrm{Trans}(H)$[1].

Definition 11. A subgroup W of $E(\mathbb{R}^2)$ is called a *wallpaper group* if $\mathrm{Trans}(W)$ is a lattice and $\mathrm{Point}(W)$ is finite.

[1] Note that $\mathrm{Trans}(H)$ is a normal subgroup (for a proof, Cf. e.g. [1], p.2), so the quotient group is well defined.

Remark 1. Informally, a wallpaper group is the set of isometries acting on a *wallpaper pattern*, which is a two-dimensional design that repeats in two directions. A wallpaper pattern thus admits a lattice of translational symmetries, which is invariant under the wallpaper group. In other words, consider a lattice L induced by $\mathrm{Trans}(W)$ for some wallpaper group W. Then each lattice point in L gets mapped to another lattice point in L under W.

Remark 2. There are exactly 17 wallpaper groups up to isomorphism. A classification of these groups can be found in [10]. Wallpaper patterns corresponding to each group are shown in figure 2.1.

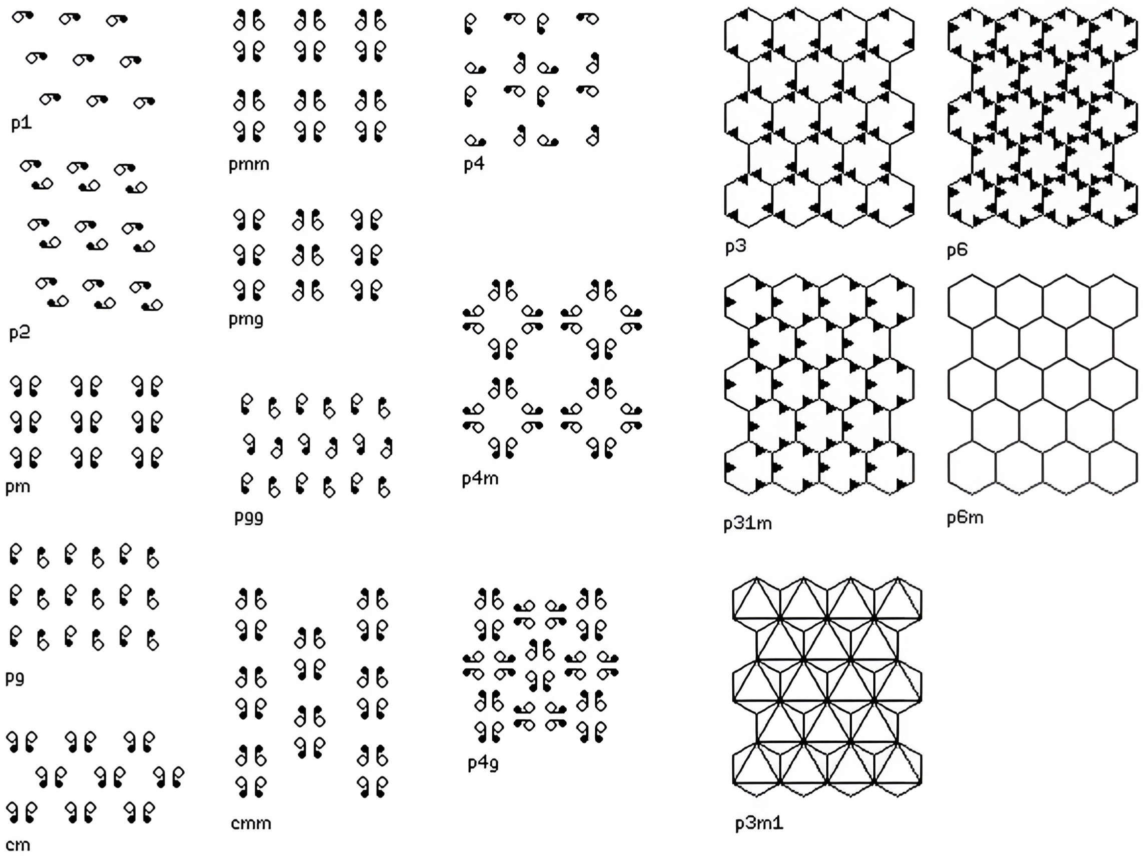


Figure 2.1: Wallpaper patterns for each of the 17 wallpaper groups [8].

3

Two-disk systems

With the preliminaries in place, we are ready to discuss compound symmetry groups. A way to visualise these groups is to consider two overlapping disks, which we call a two-disk system. When discrete rotations are defined on each disk, they are partitioned into segments when the rotations are applied. In section 3.1, we define the necessary terminology around compound symmetry groups. In section 3.2, the group structure of finite two-disk systems is discussed. Lastly, we will discuss properties of infinite two-disk systems in section 3.3.

3.1. Terminology

Definition 12. Let $(\mathbb{C}, d)$ be the complex plane with the Euclidean metric d and let $\{A_i\}_{i\in I}$ be a family of closed, not necessarily disjoint, subspaces of $\mathbb{C}$ for some index set I. For each $i \in I$, let $\sigma_i : \mathbb{C} \to \mathbb{C}$ be a function which is an isometry on $A_i \subseteq \mathbb{C}$ and the identity function on $\mathbb{C} \backslash A_i$. We will call this function a *partial isometry* on A_i. A *compound symmetry group* is defined as the group generated by the partial isometries σ_i on A_i, with function composition as the group operation: $fg(x) = g(f(x))$. More formally:

$$G = \langle \sigma_i | i \in I \rangle$$

Remark 3. The above definition can be generalised to a general metric space. However, in the current setting we are interested in groups generated by discrete rotations of two overlapping closed disks, which we have been calling a *two-disk system.* Therefore we set the following specifications:

(a) Fix $r_1, r_2 > 0$. Let A_1 and A_2 be two closed disks with centres $-1+0i$ and $1+0i$ and radii r_1 and r_2 respectively:

$$A_1 = \{z \in \mathbb{C} : |z+1| \leq r_1\}, \quad A_2 = \{z \in \mathbb{C} : |z-1| \leq r_2\}$$

(b) For the partial isometries we define $a : \mathbb{C} \to \mathbb{C}$ and $b : \mathbb{C} \to \mathbb{C}$ as the clockwise rotations of the left disk by $2\pi/n_1$ and right disk by $2\pi/n_2$ respectively, for some $n_1, n_2 \in \mathbb{Z}_{\geq 2}$.

More formally:

$$a(z) = \begin{cases} (z+1)e^{-2\pi i/n_1} - 1 & \text{if } z \in A_1 \\ z & \text{if } z \in \mathbb{C} \setminus A_1 \end{cases}$$
$$b(z) = \begin{cases} (z-1)e^{-2\pi i/n_2} + 1 & \text{if } z \in A_2 \\ z & \text{if } z \in \mathbb{C} \setminus A_2 \end{cases}$$

We denote the group with these specifications as

$$GG_{n_1,n_2}(r_1, r_2) = \langle a, b \rangle$$

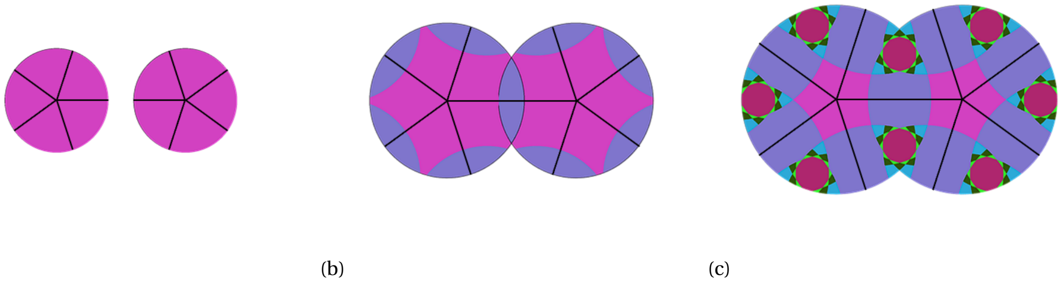
If $n_1 = n_2$ or $r_1 = r_2$, we use a single subscript. The radii can also be omitted to indicate a family of groups where the disks have equal radii but are not specified.

Remark 4. Recall that the group operation is defined as

$$fg(x) = g(f(x))$$

for $f, g \in G$. We choose this convention so that the rotations can be read left to right.

A visual representation of a two-disk system can be given by its *portrait.* A portrait shows the two disks in the Euclidean plane, where "regions that remain connected under all the group elements (*pieces*) are colored identically; the color is a function of the size of the orbit" [4]. Note that if we take $X = A_1 \cup A_2$, then a compound symmetry group can act on X. The orbit of $x \in X$ then consists of all possible images of x under the group elements. Figure 3.1 shows portraits of $GG_5(r)$ for different values of r. Figure 3.1a shows the portrait of $GG_5(0.8)$, where it is seen that the disks are disjoint. Figure 3.1b portrays $GG_5(1.2)$. Here the disks overlap, and In this case nine wedge pieces are added. In figure 3.1c, $r = 1.5$, which creates additional pieces with different orbits.



(a) (b) (c)

Figure 3.1: $GG_5(r)$ for (a) $r = 0.8$, (b) $r = 1.2$, (c) $r = 1.5$. Adapted from https://www.twodisks.org

Another method of visualising two-disk systems is to plot the orbit of specific points in the disks [4], which we will call the *orbit pattern.* Here we discretise the boundary into finitely many points and compute the orbit of each of these points under all group elements. If we plot the union of all the orbits, we get the orbit pattern as shown in figure 3.2.

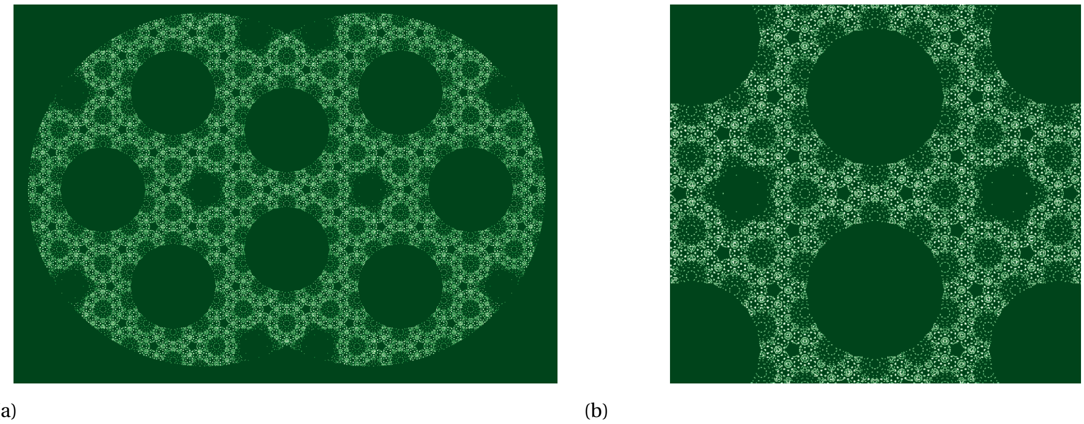


Figure 3.2: Orbits of discretized boundary of $GG_5(2.144)$. (a) shows the full orbits, (b) shows a magnified view.

Remark 5. When numerically computing the orbits, only the boundary points of the disks are considered, rather than a discretised version of the disks itself. This choice is made to reduce computation time, without significantly affecting any quantitative results. A detailed discussion on how these patterns are generated can be found in chapter 5.

3.2. Finite groups

A proportion of the compound symmetry groups are finite. Looking at the portraits in figure 3.1, group structure arises from the permutations of the pieces under the rotation of the disks. For a given portrait, we will see the group which acts on that portrait is a subgroup of a symmetric group[1]. For two disks with parameters n_1, n_2, and radii r_1, r_2 such that the disks do not overlap, there is no interaction between the disks. Therefore the group structure is the direct product of two cyclic groups:

$$GG_{n_1,n_2}(r_1, r_2) \cong C_{n_1} \times C_{n_2}$$

If the radii are chosen such that the disks overlap, the group structure becomes more complex. Recall that $G = \langle a, b \rangle$. To find the group structure given a certain portrait, we will proceed as follows:

1. Label each piece of the portrait.
2. Express the generators a and b in terms of the permutation of these pieces.
3. Find the group generated by these permutations using results from group theory or a computer algorithm.

For example, consider the portrait of $GG_2(1.5)$ in figure 3.3a, where each piece is labelled 1 up to and including 10. Note than even though visually there are five coloured regions, we will act as if there are ten to account for the two-rotational symmetry of the disks (later we will see that labelling the portrait using 7 numbers yields the same group). Applying a is equivalent to rotating the left disk clockwise by 180°. This means that the pieces are mapped in the following way:

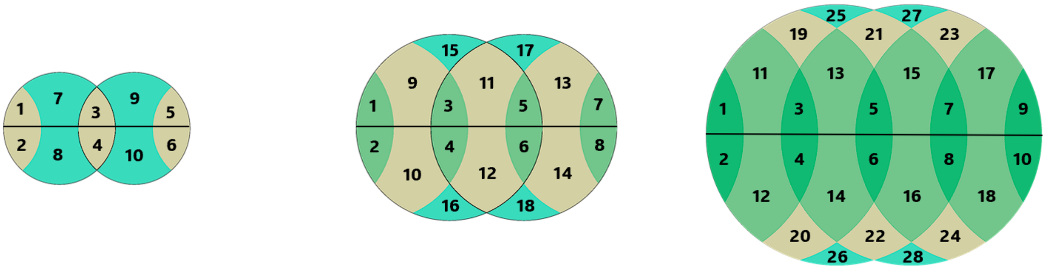
$$1 \mapsto 4, \quad 2 \mapsto 3, \quad 3 \mapsto 2, \quad 4 \mapsto 1, \quad 5 \mapsto 5, \quad 6 \mapsto 6, \quad 7 \mapsto 8, \quad 8 \mapsto, 7, \quad 9 \mapsto 9, \quad 10 \mapsto 10$$

[1] In fact, every group is isomorphic to a subgroup of a symmetric group. This is also known as Cayley's theorem.

In cycle notation this corresponds to the permutation $(1,4)(2,3)(7,8)$. Likewise applying b is equivalent to the permutation $(3,6)(4,5)(9,10)$. Therefore

$$GG_2(1.5) = \langle (1,4)(2,3)(7,8),\ (3,6)(4,5)(9,10) \rangle \cong D_6$$

One can show that the group is isomorphic to D_6 by writing out all the elements or using group-theoretical results. However, for more complicated portraits this becomes an involved task. Therefore we will resort to software to speed up the process. We will use SageMath [12], which is a free open source mathematics software which has the capability to identify a group given its generators (written as permutations).



(a) (b) (c)

Figure 3.3: $GG_2(r)$ for (a) $r = 1.5$, (b) $r = 2.5$, (c) $r = 3.5$. Adapted from https://www.twodisks.org

The same procedure can be applied to characterise the compound symmetry groups $GG_2(2.5)$ and $GG_2(3.5)$ acting on the portraits shown in figure 3.3b and 3.3c respectively. It can be verified that

$$GG_2(2.5) = \langle (1,6)(2,5)(3,4)(9,12)(10,11)(15,16), (3,8)(4,7)(5,6)(11,14)(12,13)(17,18) \rangle \cong D_{12}$$

and

$$GG_2(3.5) = \langle a, b \rangle \cong D_{60}$$

where

$$a = (1,8)(2,7)(3,6)(4,5)(11,16)(12,15)(13,14)(19,22)(20,21)(25,26)$$
$$b = (3,10)(4,9)(5,8)(6,7)(13,18)(14,17)(15,16)(21,24)(22,23)(27,28)$$

Note that the label we give each piece in the portrait does not matter; the pieces can be labelled in a different order. Also remark that for the portrait of $GG_2(1.5)$, the labelling shown in figure 3.4 yields the same group D_6. In this case the generators are $(1,2)(4,5)$ and $(2,3)(6,7)$. The reason these generators yield the same group is because the beige wedge pieces in figure 3.4 are not rotated around their own axes when the disks are rotated. This is not the case for the portrait of $GG_2(2.5)$ shown in figure 3.3b. For example, the green piece labelled 3 and 4 is rotated 180° around its own axis when the left disk is rotated. Since this does not happen for $GG_2(1.5)$, fewer labels can be used. In table 3.1, the characterised compound symmetry groups are stated again, along with several others.

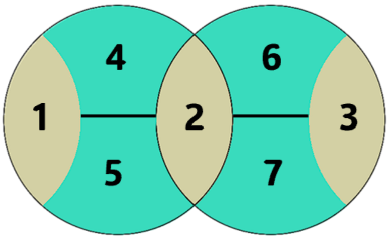


Figure 3.4: Labelling of $GG_2(1.5)$. Adapted from https://www.twodisks.org

Group	Isomorphic to
$GG_2(1.5)$	D_6
$GG_2(2.5)$	D_{12}
$GG_2(3.5)$	D_{60}
$GG_3(1.5)$	$C_3 \times C_3 \times A_5$
$GG_3(1.9)$	$C_3 \times C_3 \times A_5 \times S_7 \times S_7$
$GG_3(2)$	$A_5 \times S_7 \times S_7$
$GG_4(1.2)$	$C_4 \times A_7 \rtimes {}^2C_4$
$GG_5(1.2)$	$C_5 \times C_5 \times A_9$
$GG_6(1.1)$	$C_3 \times C_6 \times S_{11}$
$GG_7(1.1)$	$C_7 \times C_7 \times A_{13}$

Table 3.1: Characterisation of several compound symmetry groups.

For more complicated portraits, identifying its group structure becomes a computationally intensive process, such as the group acting on the portrait shown in figure 3.1c. Therefore, table 3.1 only show groups acting on relatively simple portraits.

3.3. Infinite groups

Many compound symmetry groups have an infinite number of elements for a large enough radius. Proof sketches of two theorems that characterise the infinite behaviour of compound symmetry groups have been given in [4]. We will attempt to formalise these proof sketches. We start with a definition.

Definition 13. Let GG_{n_1,n_2} be a family of groups with an infinite member. If there exists $r_c > 0$, such that $GG_{n_1,n_2}(r)$ is finite for $r < r_c$ and infinite for $r > r_c$, then r_c is called the *critical radius.*

Theorem 2. *There exists some r for which $GG_{n_1,n_2}(r)$ is infinite if and only if* $\operatorname{lcm}(n_1, n_2) \notin \{1,2,3,4,6\}$

[2] This notation is used for the semidirect product: https://mathworld.wolfram.com/SemidirectProduct.html

Proof. Suppose $\operatorname{lcm}(n_1, n_2) \notin \{1,2,3,4,6\}$. Define A_1, A_2, a, and b as in remark 3 and assume $r_1 = r_2 = r$.

Case 1: assume $n_1 = n_2 = n$. Choose $r > 0$ such that there exists $z_0 \in A_1 \cap A_2$ such that

$$\{a^{-k}(z_0) : k = 0, \dots, n-1\} \subseteq A_1 \cap A_2$$

That is, there is a point in the intersection of the two disks such that after repeatedly rotating the left disk counterclockwise, the resulting points stay in the intersection. We claim that the points

$$\{a^{-k}b^k(z_0) : k = 0, \dots, n-1\}$$

are the vertices of a regular n-gon. Indeed,

$$a^{-k}b^k(z_0) = b^k(a^{-k}(z_0)) = b^k((z_0+1)e^{2k\pi i/n} - 1) = ((z_0+1)e^{2k\pi i/n} - 2)e^{-2k\pi i/n} + 1 = \\ z_0 + 1 - 2e^{-2k\pi i/n} + 1 = z_0 + 2 - 2e^{-2k\pi i/n}$$

and note that the set $B_0 = \{z_0 + 2 - 2e^{-2k\pi i/n} : k = 0, \dots, n-1\}$ is a parametrization of the vertices of a regular n-gon with center $z_0 + 2$, circumradius $c_0 (= 2)$ and sidelength s_0. Call this n-gon P_0. Thus, the claim follows.

Subcase 1: $n > 6$. We generate translations between adjacent vertices of P_0. More precisely, if $a^{-k}b^k(z_0)$ is one vertex of P_0, then $a^{-(k+1)}b^{k+1}(z_0)$ is its clockwise adjacent vertex. Thus the operation

$$(a^{-k}b^k)^{-1}a^{-(k+1)}b^{k+1} = b^{-k}a^k a^{-k-1}b^{k+1} = b^{-k}a^{-1}bb^k \tag{3.1}$$

maps one vertex of P_0 to its clockwise adjacent one. Note that they are indeed translations, as the sum of the individual rotation angles is zero (see lemma 1 and corollary 1), and that for every $k \in \{1, \dots, n-1\}$ we get a different operation. Thus, there are n distinct translations, each corresponding to a different side of P_0. We can interpret these operations as vectors: take $n = 7$ as an example. In figure 3.5 (left) it can be seen that each side of the heptagon corresponds to a vector translation as defined in equation (3.1). These vector translations are shown more explicitly in figure 3.6.

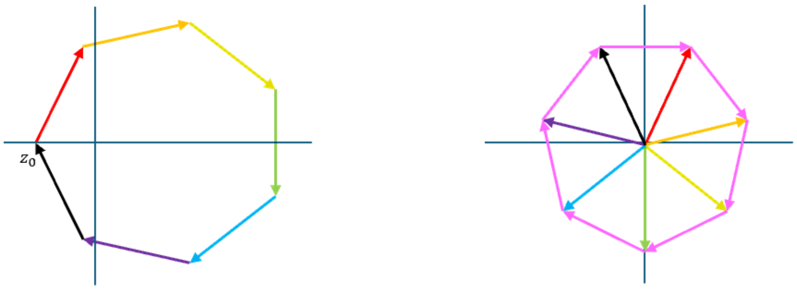


Figure 3.5: P_0 (left) and P_1 (right).

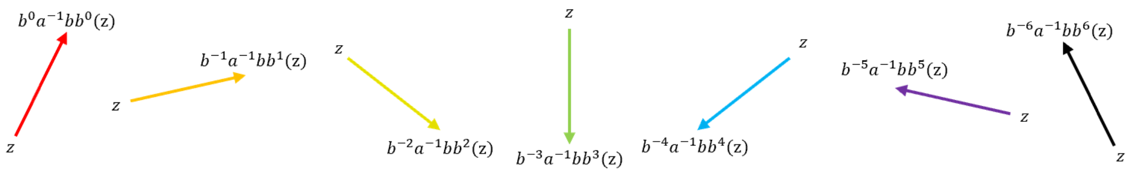


Figure 3.6: Translation vectors of P_0 for $n = 7$.

Now consider the set

$$B_1 = \{b^{-k}a^{-1}bb^k(0) : k = 0, \dots, n-1\}$$

which contains the n distinct images of these translations applied to the origin. It can be seen that this set also forms an n-gon which we will call P_1 (see figure 3.5 (right)). Continuing in this fashion, we can generate translations from one vertex to an adjacent vertex in P_1 (as indicated by the pink vectors in figure 3.5). and subsequently create the set

$$B_2 = \{b^{-k-1}ab^{-1}a^{-1}b^{k+2}(0) : k = 0, \dots, n-1\}$$

which contains the vertices of a polygon which we will call P_2. In figure 3.5, this corresponds to applying the pink translation vectors to the origin. Note the operators are again translations, as they are a composition of translations (alternatively one can check the sum of the individual rotation angles is zero). The above description marks the beginning of a general procedure: given a polygon P_m, create a polygon P_{m+1} by:

1. Generating the n translations from each vertex to its clockwise-adjacent vertex (to be interpreted as translation vectors).

2. Applying these n translations on the origin, which form the vertices of P_{m+1}.

Denote the circumradius of P_m by c_m and its side length by l_m. We claim that $c_{m+1} < c_m$. Indeed, recall that the relation between the side length and circumradius of a n-gon is given by $l_m = 2c_m \sin(\frac{\pi}{n})$ [9]. Then, since $n > 6$, we have

$$l_m = 2c_m \sin\left(\frac{\pi}{n}\right) < 2c_m \sin\left(\frac{\pi}{6}\right) = 2c_m \cdot \frac{1}{2} = c_m$$

Since $c_{m+1} = l_m$ by construction, the claim follows. We can therefore create an infinite amount of polygons via this construction. Each vertex of P_m with $m \in \mathbb{N}$ is the image of an operation applied to the origin. Since no two vertices overlap, we conclude the group must be infinite.

subcase 2 $n = 5$. In this case the side length of a pentagon is larger than its circumradius, so we need a different approach. We know that $\{a^{-k}b^k(z_0) : k = 0, \dots, 4\}$ are the vertices of a pentagon P_0 (see figure 3.7 (left)). Now let $t_k = b^{-k}a^{-1}bb^k$ be the translations from one vertex to an adjacent one as defined as before, also seen in figure 3.8. Instead of constructing B_1 as in subcase 1, we make B_1 as follows (see figure 3.7 (right)):

$$B_1 = \{t_0(0), t_0t_2(0), t_0t_2t_4(0), t_0t_2t_4t_1(0), t_0t_2t_4t_1t_3(0)\}$$

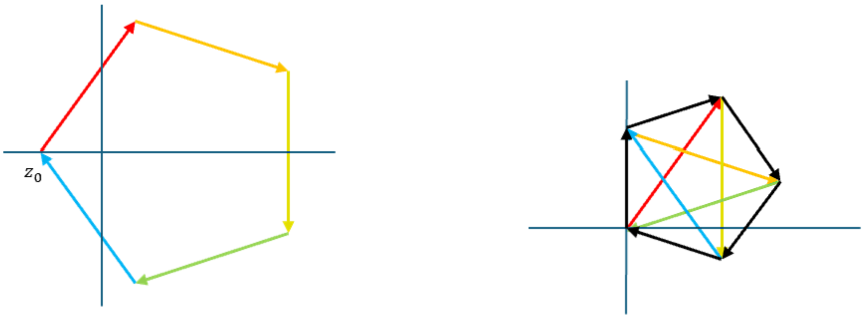


Figure 3.7: P_0 (left) and P_1 (right).

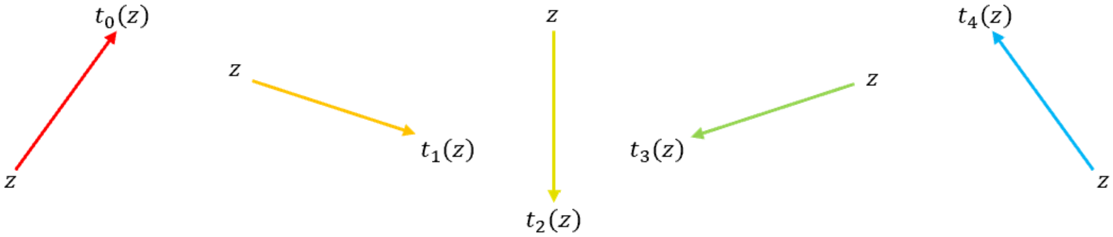


Figure 3.8: Translation vectors of P_0 for $n = 5$.

We start in the origin and keep applying consecutive translations to make a pentagram. The vertices of this pentagram form a pentagon P_1, and the process is repeated. The general procedure is similar as before: given a pentagon P_m, create a pentagon P_{m+1} by

1. Generating the 5 translations from each vertex to its clockwise-adjacent vertex (to be interpreted as translation vectors).

2. Applying these 5 translations consecutively in a specific order to the origin to make a pentagram, whose vertices form a pentagon P_{m+1}.

It is clear that P_{m+1} has a smaller circumradius than P_m. Using a similar reasoning as in subcase 1, we conclude that the group is infinite.

Case 2: $n_1 \neq n_2$. We claim $(a^{-1}b)^p$ and $(ba^{-1})^p$ are rotations by $2\pi/\text{lcm}(n_1, n_2)$ around two different centres for some $p \in \mathbb{Z}$. These rotations can play the role of a and b in the above proof. We prove the claim for $(a^{-1}b)^p$; the claim for $(ba^{-1})^p$ follows by symmetry. Note that $a^{-1}b$ is a rotation over an angle $2\pi/n_1 - 2\pi/n_2$ (see lemma 1 and corollary 1). We need to show that there exist $p, q \in \mathbb{Z}$ such that

$$p\left(\frac{2\pi}{n_1} - \frac{2\pi}{n_2}\right) = \frac{2\pi}{\text{lcm}(n_1, n_2)} + q2\pi$$

where we are accounting for the 2π-periodicity of the exponential. Using that for all $n_1, n_2 \in \mathbb{N}$, $n_1 n_2 = \text{gcd}(n_1, n_2)\text{lcm}(n_1, n_2)$, this is equivalent to finding p and q with

$$(-p - qn_2)n_1 + pn_2 = \text{gcd}(n_1, n_2) \tag{3.2}$$

See appendix A for a detailed calculation. By Bézout's identity (see theorem 1), there exist integers $x = -p - qn_2$ and $y = p$ such that $xn_1 + yn_2 = \gcd(n_1, n_2)$. From this it follows that $p \in \mathbb{Z}$, but q is not necessarily integer. However, we can multiply both sides of equation 3.2 with an integer k so that q is also integer. The case $(ba^{-1})^p$ is similar. This proves the "if" implication of theorem 2.

Conversely, suppose $\operatorname{lcm}(n_1, n_2) \in \{1,2,3,4,6\}$. The case $r < 2$ is trivial, so let $r \geq 2$ be arbitrary. Note that if $n_1 = 1$ or $n_2 = 1$, the group is trivially finite. We first prove the case for $n_1 = n_2 = 3$. The other cases are nearly identical. We start by showing the orbits of all points in the disks are finite.

Suppose $n_1 = n_2 = 3$. Recall that a and b are partial isometries of A_1 and A_2 respectively. Now define $\alpha : \mathbb{C} \to \mathbb{C}$ and $\beta : \mathbb{C} \to \mathbb{C}$ as follows:

$$\alpha(z) = (z+1)e^{-2\pi i/3} - 1$$
$$\beta(z) = (z-1)e^{-2\pi i/3} + 1$$

That is, α and β are rotations with the same rotation angles as a and b, but now they are defined as isometries acting on the entirety of $\mathbb{C}$. It holds that

$$\langle \alpha, \beta \rangle = \langle \alpha, \alpha^{-1}\beta \rangle$$

Indeed, it is clear that $\alpha, \alpha^{-1}\beta \in \langle \alpha, \beta \rangle$, so therefore $\langle \alpha, \alpha^{-1}\beta \rangle \subseteq \langle \alpha, \beta \rangle$. Conversely, it is also clear that $\alpha, \beta \in \langle \alpha, \alpha^{-1}\beta \rangle$, so therefore $\langle \alpha, \beta \rangle \subseteq \langle \alpha, \alpha^{-1}\beta \rangle$. Note that $\alpha^{-1}\beta$ is a translation by corollary 1, and therefore $\langle \alpha, \alpha^{-1}\beta \rangle$ is precisely the wallpaper group p_3 [7]. Recall from definition 11 that $\operatorname{Trans}(p_3)$ is a lattice, and from remark 1 that each lattice point gets mapped to another lattice point under p_3.

Now let $p \in A_1 \cup A_2$ be arbitrary. Let L be the lattice induced by p_3 which contains p as a lattice point. We claim for all $g \in GG_3(r)$, $g(p)$ is a lattice point in L. Recall that we can write $g = \sigma_1 \sigma_2 \ldots \sigma_m$ with $\sigma_i \in \{a, b\}$. We prove the claim by induction on m. For the base case, take $m = 1$. Then either $g = a$ or $g = b$. If $g = a$, then

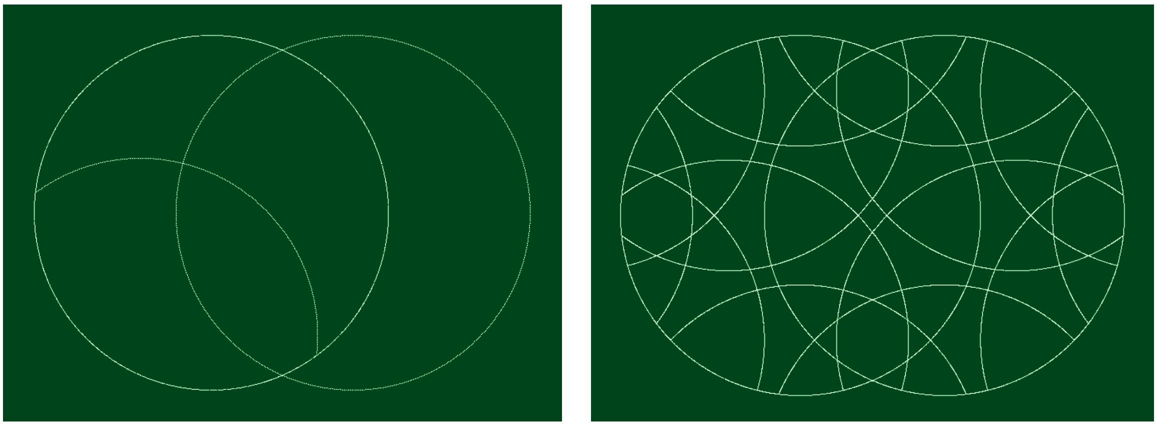
$$a(p) = \begin{cases} \alpha(p) & \text{if } p \in A_1 \\ p & \text{if } p \notin A_1 \end{cases}$$

Note that p is a lattice point and $\alpha(p)$ is a lattice point as $\alpha \in p_3$ (see remark 1). Thus $a(p)$ is a lattice point of L. The case $g = b$ is identical. For the induction hypothesis, let k be arbitrary and suppose $g_k(p) = \sigma_1 \sigma_2 \ldots \sigma_k(p)$ is a lattice point in L. For the induction step, consider $g_{k+1} = \sigma_1 \sigma_2 \ldots \sigma_k \sigma_{k+1}$. Then either $\sigma_{k+1} = a$ or $\sigma_{k+1} = b$. If $\sigma_{k+1} = a$, then

$$g_{k+1}(p) = a(g_k(p)) \begin{cases} \alpha(g_k(p)) & \text{if } p \in A_1 \\ g_k(p) & \text{if } p \notin A_1 \end{cases}$$

Note $g_k(p)$ is a lattice point by the induction hypothesis, so $\alpha(g_k(p))$ is also a lattice point as $\alpha \in p_3$ (see remark 1). Thus $g_{k+1}(p)$ is a lattice point. The case $\sigma_{k+1} = b$ is identical. By induction, the claim follows.

Consequently, the orbit of p is a subset of L within $A_1 \cup A_2$. Since $A_1 \cup A_2$ is bounded, we conclude that the orbit of p is finite.



(a) (b)

Figure 3.9: Orbit pattern of $GG_3(2.5)$. (a) shows the union of the boundaries and the boundaries after only applying a. (b) shows the whole orbit pattern.

Now consider the disk boundaries $\partial A_1 \cup \partial A_2$. By construction, the boundaries between pieces in a portrait are precisely the orbits of the disk boundaries (see figure 3.9) [4]. For example, when the left disk is rotated, a circular arc from the right disk gets rotated as well (see figure 3.9a). Each point in the boundary has finite orbit, so the orbit of $\partial A_1 \cup \partial A_2$ consists of finitely many circular arcs[3]. These arcs partition the disks into pieces (see figure 3.9b). The compound symmetry group consists of all possible permutations of these pieces under the rotation of the disks. Since there are finitely many pieces, there are also finitely many permutations of them. Thus, the group is finite.

The other cases are almost identical. The only difference is that in each case, the group generated by the global rotations with rotation angles $2\pi/n_1$ and $2\pi/n_2$ are isomorphic to a different wallpaper group. It is not hard to show the remaining isomorphisms, which are given in table 3.2:

		wallpaper
n_1	n_2	group [7]
2	4	$\langle \alpha, \beta \rangle \cong p_4$
4	4	$\langle \alpha, \beta \rangle \cong p_4$
2	6	$\langle \alpha, \beta \rangle \cong p_6$
3	6	$\langle \alpha, \beta \rangle \cong p_6$
6	6	$\langle \alpha, \beta \rangle \cong p_6$

Table 3.2: Corresponding wallpaper groups for various n_1 and n_2 such that $\mathrm{lcm}(n_1, n_2) \in \{1,2,3,4,6\}$.

For the case $n_1 = n_2 = 2$, we have that $\langle \alpha, \beta \rangle \cong p_2$, but here p_2 is not a wallpaper group. Instead it is a *frieze group*, which is similar to a wallpaper group, but has translational symmetry in only one direction instead of two [6]. At first glance this seems out of the ordinary, but this frieze group and all of the aforementioned wallpaper groups have in common that their generators consist of a rotation and a translation.

□

[3] A formal continuity argument is required here. However, the intuition behind this claim should be clear.

The "only if" direction of theorem 2 states that if $\operatorname{lcm}(n_1, n_2) \in \{1,2,3,4,6\}$, $GG_{n_1,n_2}(r)$ is finite for all r. In section 3.2 finite groups are constructed by expressing the rotations as permutations of the pieces of the portrait. When the disks increase in radii, more pieces are created, and therefore there exists no upper bound for the size of these particular finite groups (except for the identity group $GG_1(r)$). We formalise this observation in corollary 2.

Corollary 2. *Suppose* $\operatorname{lcm}(n_1, n_2) \in \{2,3,4,6\}$. *Then for all* $M \in \mathbb{N}$, *there exists* $r > 0$ *such that* $|GG_{n_1,n_2}(r)| \geq M$. *In other words, when considering the compound symmetry groups* $GG_{n_1,n_2}(r)$ *that are finite for all* r*, there exists no upper bound for their size.*

Proof. Let $M \in \mathbb{N}$ be arbitrary. Consider $a^{-1}b(z)$ for some $z \in A_1 \cap A_2$, and recall that $a^{-1}b$ is a translation (see corollary 1) on $A_1 \cap A_2$. We can choose r such that

$$a^{-1}(z), a^{-1}b(z), a^{-1}ba^{-1}(z), \ldots, \left(a^{-1}b\right)^M(z) \in A_1 \cap A_2$$

So there are at least M elements in $GG_{n_1,n_2}(r)$. So $\left|GG_{n_1,n_2}(r)\right| \geq M$.

□

Theorem 3. *GG_5 is infinite at* $r = \sqrt{3+\varphi}$

Proof. Let $\zeta_n = e^{2\pi i/n}$ and define the points $E = \zeta_5 - \zeta_5^2$, $F = 1 - \zeta_5 + \zeta_5^2 - \zeta_5^3$, $G = 2F - E$, $E' = -E$, $F' = -F$, $G' = -G$. See figure 3.10. Note that F, F', G, and G' lie on line segment EE' and that $|E+1| = r$. See appendix A for a detailed calculation. Now define the following three translations:

1. line segment $E'F'$ is transformed by $a^{-2}b^{-1}a^{-1}b^{-1}$ to line segment GF
2. line segment $F'G'$ is transformed by $abab^2$ to line segment FE
3. line segment $G'E$ is transformed by $abab^{-1}a^{-1}b^{-1}$ to line segment $E'G$

The sum of the individual rotation angles of transformations 1, 2, and 3 are -2π, 2π, and 0 respectively. Thus, these transformations are indeed translations by lemma 1 and corollary 1. Moreover, these translations translate any portion of line segment EE' piecewise onto itself, and one can check that during these transformations no point on a segment leaves the intersection of the two disks [4].

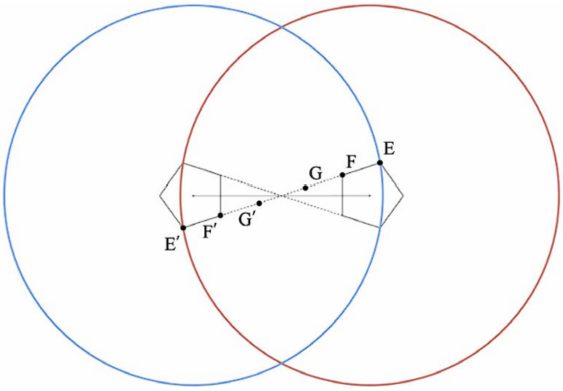


Figure 3.10: Geometric construction for critical radius [4].

The first two transformations are translations of length $|F-F'|$, and the third transformation is a translation of length $|E-G|$. With some analysis we derive that $|E-E'|/|F-F'| = \varphi$ and $|E-E'|/|E-G| = 1+\varphi$ (see appendix A). Therefore, any point on line segment EE' has infinite orbit. Indeed, suppose on the contrary there exists a point on EE' such that it has finite orbit. Then there exists a combination of aforementioned translations such that the net distance of these translations is zero. In other terms:

$$a|F-F'| + b|E-G| = 0 \Leftrightarrow$$
$$-\frac{a}{b} = \frac{|E-G|}{|F-F'|} = \frac{|E-G|}{|F-F'|} \cdot \frac{|E-E'|}{|E-E'|} = \frac{\varphi}{1+\varphi} = \varphi - 1$$

for some $a, b \in \mathbb{Z}$, which contradicts the fact that $\varphi - 1$ is irrational [5]. So every point on EE' has infinite orbit. So $GG_5(\sqrt{3+\varphi})$ is infinite. □

According to [4], the critical radius of GG_5 is exactly $\sqrt{3+\varphi}$. However, "for most n, all we have is our numerical estimates."

4

k-disk systems

In the previous chapter compound symmetry groups in the context of two-disk systems are discussed. In this chapter, this system is broadened by considering $k \in \mathbb{N}$ disks which can be placed freely in the Euclidean plane. In section 4.1, we expand the terminology we have defined in chapter 3. In section 4.2 properties of three-disk systems are discussed. Lastly, in section 4.3, some examples of k-disk systems are given.

4.1. Terminology

We will still work with compound symmetry groups as given in definition 12. Recall that the specifications set in remark 3 characterise two-disk systems. This characterisation can be broadened to a general definition.

Definition 14. A *k-disk system* is defined as follows: let A_1, A_2,..., A_k be k closed disks with centres $C = \{c_1, c_2, \dots, c_k\} \subseteq \mathbb{C}$ and radii $R = \{r_1, r_2, \dots r_k\} \subseteq \mathbb{R}_{>0}$ respectively:

$$A_1 = \{z \in \mathbb{C} : |z - c_1| \leq r_1\}, \quad A_2 = \{z \in \mathbb{C} : |z - c_2| \leq r_2\}, \quad \dots, \quad A_k = \{z \in \mathbb{C} : |z - c_k| \leq r_k\}$$

For the partial isometries define $\sigma_i : \mathbb{C} \to \mathbb{C}$ as the clockwise rotation of A_i by $2\pi/n_i$ with $n_i \in N = \{n_1, n_2, \dots, n_k\} \subseteq \mathbb{Z}_{\geq 2}$. More formally:

$$\sigma_i(z) = \begin{cases} (z - c_i)e^{-2\pi i/n_i} + c_i & \text{if } z \in A_i \\ z & \text{if } z \in \mathbb{C} \backslash A_i \end{cases}$$

We denote the group with these specifications as

$$G^k_{C,N}(R) = \langle \sigma_1, \sigma_2, \dots, \sigma_k \rangle$$

If $n_1 = n_2 = \cdots = n_k$ or $r_1 = r_2 = \cdots = r_k$, a single value is written. As before, the radii can be omitted to indicate a family of groups where the disks have equal radii but are not specified. For two-disk systems, the notation $GG_{n_1,n_2}(r_1, r_2)$ is still used.

4.2. Three-disk systems

For three-disk systems, we expand on two-disk systems by choosing the centre of the third disk such that all three centres are equidistant. We therefore take $C = \{-1+0i, 1+0i, 0+i\sqrt{3}\}$, which will be used throughout this section.

Finite groups can be characterized as in section 3.2 by labelling the pieces of the portrait of a k-disk system. When expressing the generators as permutations, the group structure can be derived. For example, consider the portrait of $G^3_{C,3}(2/\sqrt{3})$ as shown in figure 4.1. By writing the generators as permutations we get that

$$G^3_{C,3}\left(\frac{2}{\sqrt{3}}\right) = \langle a, b, c\rangle = C_3 \times C_3 \times A_5$$

where

$$\begin{aligned} a &= (1,2,3)(15,19,17)(20,18,16) \\ b &= (4,5,6)(14,23,21)(20,24,22) \\ c &= (7,8,9)(11,13,15)(12,14,10) \end{aligned}$$

The group structure is found computationally using SageMath [12].

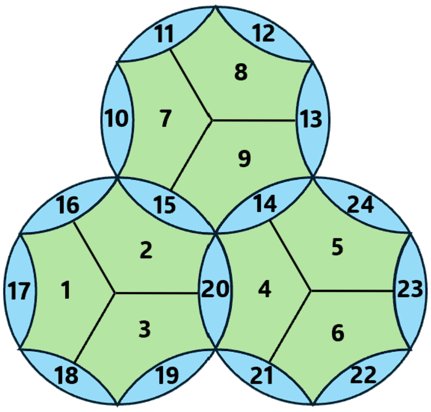


Figure 4.1: Labelled portrait of $G^3_{C,3}(2/\sqrt{3})$.

Like two-disk systems, there also exist families of groups corresponding to three-disk systems with a critical radius. We will discuss the critical radii of $G^3_{C,n}$ for various n. It is uncertain for which n there exists an algebraic expression for the critical radii, but numerical estimates can be found by considering the orbit length of points on the disk boundaries. The numerical estimates can be found in table 4.1, which are computed up to three decimal places. These estimates are found using a brute-force algorithm, where the algorithm assumes that a group is infinite if it discovers a point whose orbit length exceeds a predefined threshold. A detailed discussion on how these estimates are computed is found in chapter 5.

n	Estimate
4	1.999
5	1.376
7	1.137
8	1.151
9	1.122
10	1.086

n	Estimate
11	1.056
12	1.376
13	1.040
14	1.044
15	1.044
16	1.034

Table 4.1: Critical radii for $G^3_{C,n}$.

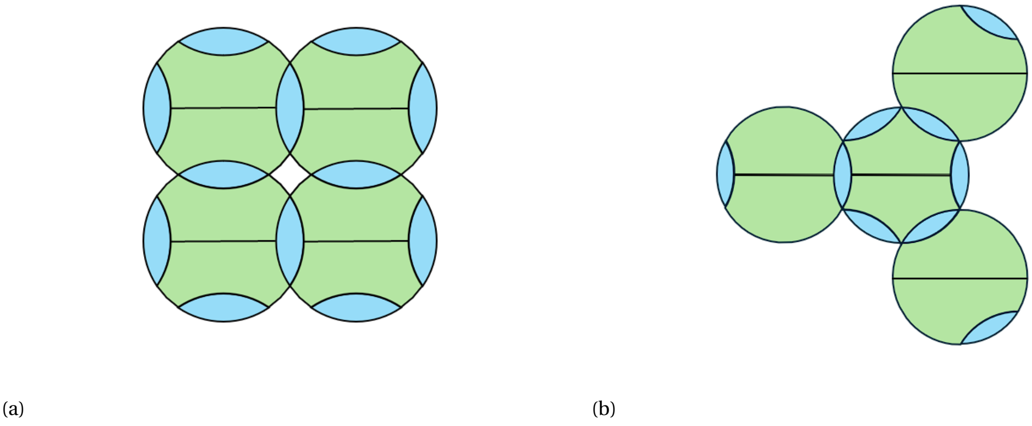
Note that in table 4.1, the estimates for each n are lower than those of the two-disk systems stated in [4]. Also note that no estimates are given for $n \in \{1,2,3,6\}$. This is because computationally, for large radii, no point is found within the disks that exceeds the orbit threshold. Therefore, it is conjectured that for these n, the groups are finite for all $r > 0$. We will state a conjecture similar to theorem 2.

Conjecture 1. *Let $G^3_{C,N}$ be a family of groups corresponding to a three-disk system with $C = \{-1+0i, 1+0i, 0+i\sqrt{3}\}$ and $N = \{n_1, n_2, n_3\}$. Then there exists some r for which $G^3_{C,N}(r)$ is infinite if and only if $\mathrm{lcm}(n_1, n_2, n_3) \notin \{1,2,3,6\}$.*

4.3. General disk systems

Up until now only disk systems with equidistant centres have been analysed. In the Euclidean plane however, no more than three points can be equidistant. This constraint makes it unnatural to extend our analysis to systems with more disks. Nevertheless, it is still worth investigating whether the group structure changes when the distance condition is relaxed. We will explore some examples.

Example 5. (4-disk systems) Consider the two 4-disk systems as shown in figure 4.2. The portraits in figure 4.2a and 4.2b correspond to $G^4_{C_1,2}(r_1)$ and $G^4_{C_2,2}(r_2)$ respectively, with $C_1 = \{-1-i, -1+i, 1-i, 1+i\}$, $C_2 = \{0, -2, 1+i\sqrt{3}, 1-i\sqrt{3}\}$ and $r_1, r_2 > 0$ such that 'wedge pieces' are created (indicated by the blue regions). By labelling the pieces and expressing the rotations as permutations, one can verify that the portrait shown in figure 4.2a corresponds to the group D_6 and the portrait shown in figure 4.2b corresponds to $C_6 \times S_4$. Having different centers for the disks may thus result in different groups.


(a)

(b)

Figure 4.2: Two configurations of a 4-disk system corresponding to (a) $G^4_{C_1,2}(r_1)$ and (b) $G^4_{C_2,2}(r_2)$.

Example 6. (Critical radius) Consider the same two 4-disk systems as shown in figure 4.2, but now with each disk having 4-rotational symmetry instead of 2. Denote the family of these groups as $G^4_{C_1,4}$ and $G^4_{C_2,4}$. We will try to numerically estimate the critical radius of both systems by using a brute-force approach (explained in more detail in chapter 5). For $G^4_{C_2,4}$ the critical radius is estimated to be $r = 1.946$. However, for $G^4_{C_1,4}$, the values $1 \le r \le 10$ were checked, but none of these yielded a point that exceeded the orbit threshold. A plausible hypothesis is that $G^4_{C_1,4}$ is finite for all $r > 0$.

5

Numerical approaches

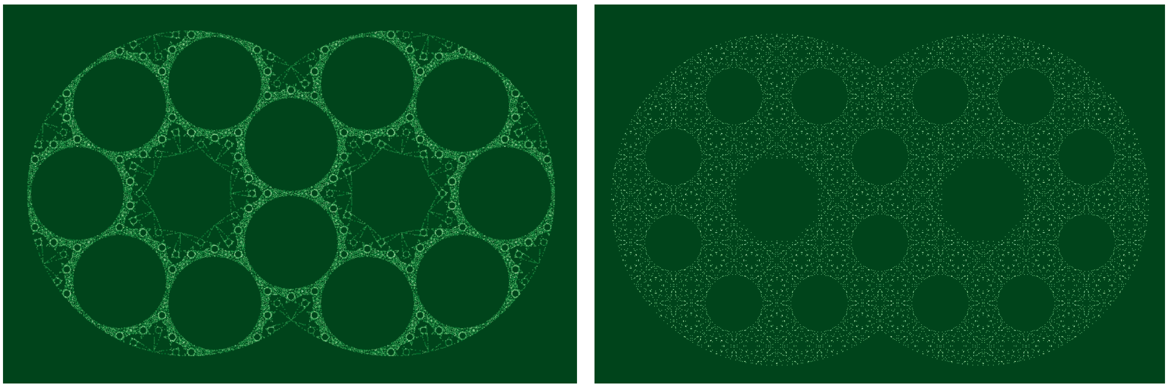
Several algorithms are implemented in Python to generate orbit patterns and estimate critical radii. In section 5.1 we will discuss how the various orbit patterns of compound symmetry groups are generated. In section 5.2 algorithms for estimating critical radii are discussed. The code can be found in appendix B.

5.1. Orbit patterns

At its core, an orbit pattern shows the orbits traced by the boundary of the disks. The first step in generating such a pattern is to discretize the boundary of each disk into finitely many points. In the code this is done by the `create_boundary` method and the amount of points is given by `num_points`. For each of the discretized points, we calculate its orbit using the `get_orbit` method: for a given start point (in this case a boundary point), we compute its image under the clockwise and counterclockwise rotation of the two disks, and add those points to the orbit. Then, for those images, we calculate their images again, where we do not add a point to the orbit if it has already been encountered before. This process is repeated until all the points in the orbit are found (in which case the orbit is finite), or until a predefined threshold `cut_off` for the length of the orbit is reached. Since many orbits have infinite length, the latter ensures `get_orbit` terminates in a finite amount of time. We will choose `cut_off` in such a way that, if the orbit reaches this length, we can reasonably assume that it is infinite. The pseudocode of `get_orbit` is given by algorithm 1, where we use the notation for k-disk systems as defined in definition 14.

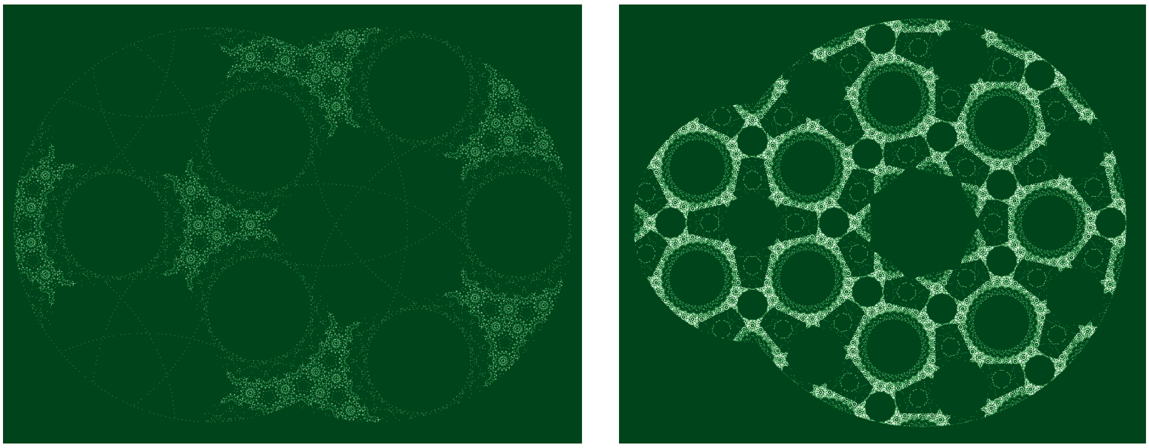
Algorithm 1 Procedure for calculating the orbit of a given start point.

```
procedure GET_ORBIT($z$, cut_off)
    orbit = $\{z\}$
    $m \leftarrow 0$
    while $m <$ |orbit| and |orbit| $<$ cut_off do    ▷ Terminates when all points are found or cut_off is reached
        for $i \leftarrow 1$ to $k$ do
            if orbit[$m$] $\in A_i$ then
                orbit $\leftarrow$ orbit $\cup \{\sigma_i$(orbit[$m$])$\} \cup \{\sigma_i^{-1}$(orbit[$m$])$\}$
            end if
        end for
        $m \leftarrow m + 1$
    end while
    return orbit
end procedure
```



(a) (b)

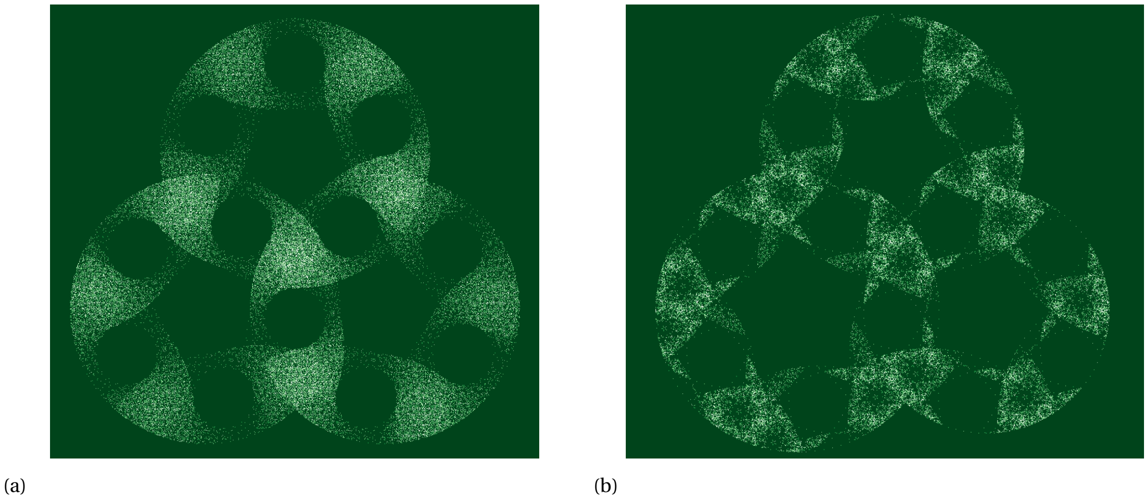
Figure 5.1: Orbit patterns of (a) $GG_7(1.6)$, and (b) $GG_8(1.6)$.



(a) (b)

Figure 5.2: Orbit patterns of (a) $GG_{3,5}(2.4)$, and (b) $GG_{4,7}(1.4, 2.4)$.

After the orbits of all boundary points are computed, they are stored in one array and then mapped onto a bitmap, which is an image in which each pixel can have one of two colours. The colours are chosen arbitrarily. Calculating the orbit of every boundary point is computationally expensive. Therefore, several optimizations have been made. Firstly, the `joblib` package [2] is used to allow for parallel computing of the orbits, considerably shortening the runtime. Secondly, the `KDTree` class from the `scipy.spatial` library [3] is used. This class helps with rapid nearest neighbour look-up, which is used to efficiently map the orbits onto the bitmap. The full code can be found in appendix B. Note that the code treats disks as instances from a class, so it can be used easily to analyse an arbitrary number of disks.



(a) (b)

Figure 5.3: Orbit patterns of (a) $G^3_{C,5}(1.5)$, and (b) $G^3_{C,N}(1.5, 1.3, 1.4)$ with $C = \{-1, 1, i\sqrt{3}\}$ and $N = \{5, 6, 7\}$.

5.2. Critical radii

To estimate the critical radius of a family of groups G^k_n, we compute the length of the orbit of the discretised boundary points of the disk (using `create_boundary`), which is similar to the generation of orbit patterns described above. The main method is `find_radius` which, as the names suggests, estimates the critical radius of G^k_n. It does so by incrementally increasing the radius of the disks, until there exists a boundary point which reaches the predefined `cut_off` threshold, from which we then can assume that the orbit is infinite. We only consider the boundary points as it is computationally less expensive than discretizing the entire disks. See algorithm 2 for the pseudocode.

A naive approach would be to discretize an interval in which we believe the critical radius is present and then check every value in that interval. A more optimised approach would be to estimate the critical radius up to, for example, one decimal place (which is less expensive to compute), and then use that estimate as a starting point to compute the radius up to two decimal places. In this manner, refining the critical radius by adding another decimal of precision becomes more efficient as we start from an estimate that is already close to the actual value. This approach can be used by the `find_radius` method by decreasing the `step` size each time a new estimate is found. Furthermore, runtime is also reduced by the `joblib` package [2], which is used again to parallelly compute the orbits across multiple CPU cores.

Algorithm 2 Brute-force estimation of the critical radius.

```
procedure FIND_RADIUS(start, stop, step, num_points, cut_off)
    for $R \leftarrow$ start to stop step step do
        for $i \leftarrow 1$ to $k$ do                              ▷ Assign radius $R$ to all disks
            $r_i = R$
        end for
        boundary = create_boundary(num_points)          ▷ Discretizes disk boundaries
        orbits $\leftarrow \{\}$
        for $z$ in boundary do                 ▷ Compute orbit of every boundary point
            orbit = get_orbit($z$, cut_off)
            orbits $\leftarrow$ orbits $\cup$ {orbit}
        end for
        for orbit in orbits do                            ▷ If cut_off is reached, return $R$
            if |orbit| $\geq$ cut_off then
                return $R$
            end if
        end for
    end for
    return None                                ▷ if cut_off is never reached, return None
end procedure
```

The natural question that arises is how accurate the computed estimates are. To obtain an indication of the accuracy, we use `critical_radius` to approximate the critical radii of two-disk systems and compare these estimates to those found in [4]. Here we discretize each disk into 20 points, take the orbit threshold at 1.000.000, and estimate the radius up to 6 decimal places. It is mentioned in [4] that "points were found with a minimum of 10 billion destinations, in some cases up to 10 trillion." See table 5.1.

n	`critical_radius`	Estimates by [4]
5	2.147940	2.148961
7	1.623600	1.623574
8	1.708243	1.711411
9	1.408573	1.408482
10	1.540012	1.543357
11	1.290629	1.290582
12	1.375706	1.376547
13	1.214101	1.213594
14	1.196373	1.196554
15	1.163295	1.163276
16	1.148509	1.148470

Table 5.1: Critical radii for GG_n.

Note in table 5.1 that for each n, almost all estimates are similar up to two decimal places, from which then they start to differ. This is expected, since the permitted orbit length in [4] is several orders of magnitude larger than the threshold used by our own method; `critical_radius` cannot afford such a large threshold due to our limited computational power. Nevertheless, this comparison does show that `critical_radius` yields reasonable approximations. It can be confidently said that they are accurate up to two decimal places. Consequently, it can be assumed that the estimates given in table 4.1 are also accurate up to two decimal places.

6 Conclusion and discussion

The research on compound symmetry groups is still in its early stages. In this thesis, we formalised the results in [4] while also exploring other areas within the study of these groups. In chapter 3 compound symmetry groups acting on two-disk systems are analysed. Here a formal proof of theorem 2 and 3 is given, whose proof sketches are explained in [4]. Additionally, a method for identifying the group structure of finite compound symmetry groups is developed: first the pieces of the portrait are labelled, and then the generators are written as permutations from which the group structure can be identified. In chapter 4 our research is expanded by exploring k-disk systems. For three-disk systems in particular, numerical estimates for their critical radii are shown in table 4.1, which are believed to be accurate up to two decimal places. Lastly, in chapter 5, algorithms for generating orbit patterns and approximating the critical radii are described. Orbit patterns are generated by tracing the orbits of discretised boundary points. Estimates for the critical radii are computed by incrementally checking an interval of potential radii, and determining the first radius which contains a boundary points whose orbit exceeds a large enough threshold (and can thus be considered infinite).

There is still much more investigation to be done on compound symmetry groups. The question remains as to whether conjecture 1 holds. Moreover, it is unknown if algebraic expressions exist for the critical radii of three-disk systems, and the study of k-disk systems remains preliminary. Additionally, within the described numerical algorithms only the disk boundaries are considered because of limited computational power. Future research could focus on proving conjecture 1, and whether the theorems on the infinite behaviour of two-disk systems can be generalised to k-disk systems. Another direction for future research can be the development of more efficient numerical algorithms, for example, for approximating the critical radii.

A

Detailed calculations

Theorem 2

$$\frac{2\pi}{n_1} - \frac{2\pi}{n_2} = 2\pi\frac{n_2 - n_1}{n_1 n_2} = \frac{2\pi}{\mathrm{lcm}(n_1, n_2)}\frac{n_2 - n_1}{\mathrm{gcd}(n_1, n_2)}$$

We need to show that there exists $p, q \in \mathbb{Z}$ such that

$$p\frac{2\pi}{\mathrm{lcm}(n_1, n_2)}\frac{n_2 - n_1}{\mathrm{gcd}(n_1, n_2)} = \frac{2\pi}{\mathrm{lcm}(n_1, n_2)} + q2\pi$$

With some algebraic manipulation we get the following

$$\begin{aligned}
\frac{p}{\mathrm{lcm}(n_1, n_2)}\frac{n_2 - n_1}{\mathrm{gcd}(n_1, n_2)} = \frac{1}{\mathrm{lcm}(n_1, n_2)} + q \Leftrightarrow \\
\frac{p}{\mathrm{lcm}(n_1, n_2)}(n_2 - n_1) = \frac{\mathrm{gcd}(n_1, n_2)}{\mathrm{lcm}(n_1, n_2)} + q\mathrm{gcd}(n_1, n_2) \Leftrightarrow \\
\frac{p}{\mathrm{lcm}(n_1, n_2)}(n_2 - n_1) - q\mathrm{gcd}(n_1, n_2) = \frac{\mathrm{gcd}(n_1, n_2)}{\mathrm{lcm}(n_1, n_2)} \Leftrightarrow \\
p(n_2 - n_1) - qn_1 n_2 = \mathrm{gcd}(n_1, n_2) \Leftrightarrow \\
(-p - qn_2)n_1 + pn_2 = \mathrm{gcd}(n_1, n_2)
\end{aligned}$$

Theorem 3

$$
\begin{aligned}
|E+1| &= \\
|e^{2\pi i/5} - e^{4\pi i/5} + 1| &= \\
\left|\cos(\frac{2\pi}{5}) + i\sin(\frac{2\pi}{5}) - (\cos(\frac{4\pi}{5}) + i\sin(\frac{4\pi}{5})) + 1\right| &= \\
\left|\frac{1}{4}(\sqrt{5}-1) + i\sqrt{\frac{5}{8}+\frac{\sqrt{5}}{8}} - \frac{1}{4}(-1-\sqrt{5}) - i\sqrt{\frac{5}{8}-\frac{\sqrt{5}}{8}} + 1\right| &= \\
\left|\frac{1}{2}\sqrt{5} + 1 + i\left(\sqrt{\frac{5}{8}+\frac{\sqrt{5}}{8}} - \sqrt{\frac{5}{8}-\frac{\sqrt{5}}{8}}\right)\right| &= \\
\sqrt{\left(\frac{1}{2}\sqrt{5}+1\right)^2 + \left(\sqrt{\frac{5}{8}+\frac{\sqrt{5}}{8}} - \sqrt{\frac{5}{8}-\frac{\sqrt{5}}{8}}\right)^2} &= \\
\sqrt{\frac{5}{4} + \sqrt{5} + 1 + \frac{5}{4} - 2\sqrt{\left(\frac{5}{8}\right)^2 - \left(\frac{\sqrt{5}}{8}\right)^2}} &= \\
\sqrt{\frac{7}{2} + \sqrt{5} - 2\frac{\sqrt{5}}{4}} &= \\
\sqrt{3 + \frac{1+\sqrt{5}}{2}} &= \\
\sqrt{3+\varphi}
\end{aligned}
$$

$$
\begin{aligned}
&|E| = \\
&|e^{2\pi i/5} - e^{4\pi i/5}| = \\
&\left|\cos(\frac{2\pi}{5}) + i\sin(\frac{2\pi}{5}) - (\cos(\frac{4\pi}{5}) + i\sin(\frac{4\pi}{5}))\right| = \\
&\left|\frac{1}{4}(\sqrt{5}-1) + i\sqrt{\frac{5}{8}+\frac{\sqrt{5}}{8}} - \frac{1}{4}(-1-\sqrt{5}) - i\sqrt{\frac{5}{8}-\frac{\sqrt{5}}{8}}\right| = \\
&\left|\frac{1}{2}\sqrt{5} + i\left(\sqrt{\frac{5}{8}+\frac{\sqrt{5}}{8}} - \sqrt{\frac{5}{8}-\frac{\sqrt{5}}{8}}\right)\right| = \\
&\sqrt{\left(\frac{1}{2}\sqrt{5}\right)^2 + \left(\sqrt{\frac{5}{8}+\frac{\sqrt{5}}{8}} - \sqrt{\frac{5}{8}-\frac{\sqrt{5}}{8}}\right)^2} = \\
&\sqrt{\frac{5}{4}+\frac{5}{4} - 2\sqrt{\left(\frac{5}{8}\right)^2 - \left(\frac{\sqrt{5}}{8}\right)^2}} = \\
&\sqrt{\frac{5}{2} - 2\frac{\sqrt{5}}{4}} = \\
&\sqrt{\frac{5-\sqrt{5}}{2}}
\end{aligned}
$$

$$
\begin{aligned}
&|F| = \\
&|1 - e^{2\pi i/5} + e^{4\pi i/5} - e^{6\pi i/5}| = \\
&\left|1 - \cos\left(\frac{2\pi}{5}\right) + \cos\left(\frac{4\pi}{5}\right) - \cos\left(\frac{6\pi}{5}\right) + i\left(-\sin\left(\frac{2\pi}{5}\right)\right) + \sin\left(\frac{4\pi}{5}\right) - \sin\left(\frac{6\pi}{5}\right)\right| = \\
&\left|1 - \frac{1}{4}\left(\sqrt{5}-1\right) + \frac{1}{4}\left(-\sqrt{5}-1\right) - \frac{1}{4}\left(-\sqrt{5}-1\right) + i\left(-\sqrt{\frac{5}{8}+\frac{\sqrt{5}}{8}} + \sqrt{\frac{5}{8}-\frac{\sqrt{5}}{8}} + \sqrt{\frac{5}{8}-\frac{\sqrt{5}}{8}}\right)\right| = \\
&\left|1 - \frac{1}{4}\left(\sqrt{5}-1\right) + i\left(2\sqrt{\frac{5}{8}-\frac{\sqrt{5}}{8}} - \sqrt{\frac{5}{8}+\frac{\sqrt{5}}{8}}\right)\right| = \\
&\sqrt{\left(\frac{5}{4}-\frac{1}{4}\sqrt{5}\right)^2 + \left(2\sqrt{\frac{5}{8}-\frac{\sqrt{5}}{8}} - \sqrt{\frac{5}{8}+\frac{\sqrt{5}}{8}}\right)^2} = \\
&\sqrt{\frac{15-5\sqrt{5}}{8} + \frac{25-3\sqrt{5}}{8} - 4\sqrt{\left(\frac{5}{8}\right)^2 - \left(\frac{\sqrt{5}}{8}\right)^2}} = \\
&\sqrt{\frac{40-8\sqrt{5}}{8} - 4\frac{\sqrt{5}}{4}} = \\
&\sqrt{5-2\sqrt{5}}
\end{aligned}
$$

$$
\begin{aligned}
\frac{|E-E'|}{|F-F'|} = \frac{2|E|}{2|F|} = \frac{|E|}{|F|} &= \\
\frac{\sqrt{\frac{5-\sqrt{5}}{2}}}{\sqrt{5-2\sqrt{5}}} = \sqrt{\frac{5-\sqrt{5}}{2\left(5-2\sqrt{5}\right)}} = \sqrt{\frac{5-\sqrt{5}}{10-4\sqrt{5}} \cdot \frac{10+4\sqrt{5}}{10+4\sqrt{5}}} &= \\
\sqrt{\frac{30+10\sqrt{5}}{20}} = \sqrt{\frac{3}{2}+\frac{1}{2}\sqrt{5}} = \sqrt{\left(\frac{1+\sqrt{5}}{2}\right)^2} &= \\
\varphi
\end{aligned}
$$

$$
\begin{aligned}
|E-F| &= \\
|2e^{\frac{2}{5}\pi i} - 2e^{\frac{4}{5}\pi i} - 1 + e^{\frac{6}{5}\pi i}| &= \\
\left|2\cos\left(\frac{2\pi}{5}\right) - 2\cos\left(\frac{4\pi}{5}\right) + \cos\left(\frac{6\pi}{5}\right) - 1 + i\left(2\sin\left(\frac{2\pi}{5}\right)\right) - 2\sin\left(\frac{4\pi}{5}\right) + \sin\left(\frac{6\pi}{5}\right)\right| &= \\
\left|\frac{1}{2}(\sqrt{5}-1) - \frac{1}{2}(-1-\sqrt{5}) + \frac{1}{4}(-1-\sqrt{5}) + i\left(2\sqrt{\frac{5}{8}+\frac{\sqrt{5}}{8}} - 2\sqrt{\frac{5}{8}-\frac{\sqrt{5}}{8}} - \sqrt{\frac{5}{8}-\frac{\sqrt{5}}{8}}\right)\right| &= \\
\left|\frac{3}{4}\sqrt{5} - \frac{5}{4} + i\left(2\sqrt{\frac{5}{8}+\frac{\sqrt{5}}{8}} - 3\sqrt{\frac{5}{8}-\frac{\sqrt{5}}{8}}\right)\right| &= \\
\sqrt{\left(\frac{3}{4}\sqrt{5}-\frac{5}{4}\right)^2 + \left(2\sqrt{\frac{5}{8}+\frac{\sqrt{5}}{8}} - 3\sqrt{\frac{5}{8}-\frac{\sqrt{5}}{8}}\right)^2} &= \\
\sqrt{\frac{35-15\sqrt{5}}{8} + \frac{65-5\sqrt{5}}{8} - 12\sqrt{\left(\frac{5}{8}\right)^2 - \left(\frac{\sqrt{5}}{8}\right)^2}} &= \\
\sqrt{\frac{25-5\sqrt{5}}{2} - 12\frac{\sqrt{5}}{4}} &= \\
\sqrt{\frac{25-11\sqrt{5}}{2}}
\end{aligned}
$$

$$
\begin{aligned}
\frac{|E-E'|}{|E-G|} = \frac{2|E|}{2|E-F|} = \frac{|E|}{|E-F|} &= \\
\frac{\sqrt{\frac{5-\sqrt{5}}{2}}}{\sqrt{\frac{25-11\sqrt{5}}{2}}} = \sqrt{\frac{5-\sqrt{5}}{25-11\sqrt{5}}} = \sqrt{\frac{5-\sqrt{5}}{25-11\sqrt{5}} \cdot \frac{25+11\sqrt{5}}{25+11\sqrt{5}}} &= \\
\sqrt{\frac{70+30\sqrt{5}}{20}} = \sqrt{\frac{7}{2}+\frac{3}{2}\sqrt{5}} = \sqrt{\left(\frac{3+\sqrt{5}}{2}\right)^2} &= \\
1+\varphi
\end{aligned}
$$

B

Code

```
# -*- coding: utf-8 -*-
"""
Created on Sat May 31 13:28:03 2025

@author: jinny
"""

import numpy as np
import matplotlib.pyplot as plt
from scipy.spatial import KDTree
from joblib import Parallel, delayed

class Disk:
    def __init__(self, n, center, radius = 1):
        self.n = n
        self.center = np.array(center)
        self.radius = radius
        self.rotation_matrix = np.array(
                                [[np.cos(2*np.pi/self.n),
                                 ↪  -np.sin(2*np.pi/self.n)],
                                 [np.sin(2*np.pi/self.n),
                                  ↪  np.cos(2*np.pi/self.n)]]
                                )

    # discretizes disk boundary into finitely many points
    def create_boundary(self, num_points):
        angles = np.linspace(0, 2*np.pi, num_points)
        points = np.stack([self.center[0] + self.radius *
         ↪  np.cos(angles),
                           self.center[1] + self.radius *
                            ↪  np.sin(angles)], axis = 1)
        return points
    # rotate point clockwise
    def rotate_cw(self, point):
        translation_to_origin = point - self.center
        rotation = self.rotation_matrix @ translation_to_origin.T
        return rotation + self.center

    # rotate point anti-clockwise
    def rotate_acw(self, point):
        translation_to_origin = point - self.center
        rotation = self.rotation_matrix.T @ translation_to_origin.T
        return rotation + self.center

class DiskSystem:
    def __init__(self, *Disks):
        self.disks = list(Disks)
        self.amount = len(list(Disks))

    # concatenate discretized boundary points of all disks into one array
    def create_boundary(self, num_points):
        boundary = []
        for disk in self.disks:
            boundary.append(disk.create_boundary(num_points))
        return np.concatenate(boundary)

    # compute orbit of start_point
    def get_orbit(self, start_point, cut_off = 1000):
        orbit = [np.array(start_point)]
        seen = {tuple(np.round(start_point, 5))}

        k = 0
        while k < len(orbit) and len(orbit) < cut_off:
            for disk in self.disks:
                if np.dot(orbit[k] - disk.center, orbit[k] - disk.center) <= disk.radius**2:
                    for rotate in [disk.rotate_cw, disk.rotate_acw]:
                        point = rotate(orbit[k])
                        rounded = tuple(np.round(point, 5))
                        if rounded not in seen:
                            orbit.append(point)
                            seen.add(rounded)
            k += 1
        return np.array(orbit)

    # find critical radius by checking an interval of radii
```

```
def find_radius(self, start, stop, step, num_points, cut_off):
    for R in np.arange(start, stop, step):
        for disk in self.disks:
            disk.radius = R
        boundary = self.create_boundary(num_points)
        orbits = Parallel(n_jobs =
         ↪  -1)(delayed(self.get_orbit)(start_point, cut_off) for
         ↪  start_point in boundary)
        for orbit in orbits:
            if len(orbit) >= cut_off:
                for disk in self.disks:
                    disk.radius = R - step
                boundary1 = self.create_boundary(50)
                orbits1 = Parallel(n_jobs =
                 ↪  -1)(delayed(self.get_orbit)(start_point,
                 ↪  cut_off) for start_point in boundary1)
                lengths = [len(orbit) for orbit in orbits1]
                print(max(lengths))
                return R
    return None

# find critical radius up to a certain decimal
def critical_radius(self, precision, num_points, cut_off):
    start = 1
    steps = [0] + [10**(-k) for k in range(1, precision + 1)]
    for k in range(1, len(steps)):
        try:
            start = self.find_radius(start - steps[k-1], 3,
             ↪  steps[k], num_points, cut_off)
        except TypeError:
            print('N = ',self.disks[0].n,'cut_off point not reached;
             ↪  the group is considered finite')
            break
    if start is not None:
        print('N = ',self.disks[0].n,'cut_off point reached at r
         ↪  =',start, '; orbit is considered infinite')


# define image grid for portrait generation
@staticmethod
def coord_grid(xmin, xmax, ymin, ymax, density):
    x = np.linspace(xmin, xmax, int(xmax - xmin) * density)
    y = np.linspace(ymin, ymax, int(ymax - ymin) * density)

    x_grid, y_grid = np.meshgrid(x,y)
    return np.stack([x_grid, y_grid], axis = -1)
    # generates bitmap portrait
    def bitmap_portrait(self, num_points, cut_off, density, xmin, xmax,
     ↪  ymin, ymax):
        boundary = self.create_boundary(num_points)
        orbits = Parallel(n_jobs =
         ↪  -1)(delayed(self.get_orbit)(start_point, cut_off) for
         ↪  start_point in boundary)
        grid = self.coord_grid(xmin, xmax, ymin, ymax, density)
        flattened_output = np.reshape(grid, (-1, 2))
        tree = KDTree(flattened_output)

        height, width = grid.shape[:2]
        image = np.zeros((height, width))

        all_points = np.vstack(orbits)
        _, indices = tree.query(all_points)

        for index in indices:
            i, j = np.unravel_index(index, (height, width))
            image[i,j] = 1
        return image

# example
N = 4
r= 1.8

d1 = Disk(N, [0,0],r)
d2 = Disk(N, [-2,0],r)
d3 = Disk(N, [1, np.sqrt(3)],r)
d4 = Disk(N, [1, -np.sqrt(3)],r)

NDisk = DiskSystem(d1,d2,d3,d4)

critrad_orbit_cutoff = 1000000
critrad_disk_numpoints = 20

for N in range(1,17):
    for disk in NDisk.disks:
        disk.n = N
        disk.rotation_matrix = np.array(
                                [[np.cos(2*np.pi/N),
                                 ↪  -np.sin(2*np.pi/N)],
                                  [np.sin(2*np.pi/N),
                                   ↪  np.cos(2*np.pi/N)]]
                                )
    NDisk.critical_radius(3, critrad_disk_numpoints,
     ↪  critrad_orbit_cutoff)

image_orbit_cutoff = 1000
image_disk_numpoints = 100
image_density = 300
xmin, xmax, ymin, ymax = -4, 4, -3, 3

image = NDisk.bitmap_portrait(image_disk_numpoints, image_orbit_cutoff,
 ↪  image_density, xmin, xmax, ymin, ymax)
plt.figure()
plt.imshow(image, cmap = "Greens_r", origin = 'lower', extent=[xmin,
 ↪  xmax, ymin, ymax])
plt.show()
```

Bibliography

[1] Elijah Blum and Edmund Lam. Wallpaper groups. Technical report, Independent reading report for MATH 4340, Honors Algebra (expository notes), May 2024. Available at `https://blumelijah.github.io/WallpaperGroups.pdf`.

[2] Joblib developers. *Joblib: running Python functions as pipeline jobs*, 2024. URL `https://joblib.readthedocs.io`. Version 1.4.2. Available at: `https://joblib.readthedocs.io/`.

[3] Virtanen et al. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020. doi: 10.1038/s41592-019-0686-2.

[4] Robert A. Hearn, William Kretschmer, Tomas Rokicki, Benjamin Streeter, and Eric Vergo. Two-disk compound symmetry groups. *Bridges 2023 Conference Proceedings*, 2023. doi: https://archive.bridgesmathart.org/2023/bridges2023-29.pdf.

[5] Ron Knott. The golden section ratio: Phi, 1996-2023. URL `https://r-knott.surrey.ac.uk/Fibonacci/phi.html#section7.1`.

[6] Tyler Landau. Classifcations of frieze groups and an introduction to crystallographic groups, May 2019. URL `https://www.whitman.edu/documents/Academics/Mathematics/2019/Landau-Balof.pdf`.

[7] University of Manchester. Symmetry in nature – chapter 3: Wallpaper patterns, 2020. URL `https://personal.maths.manchester.ac.uk/jm/wiki/uploads/SymmetryInNature/Chapter3.pdf`.

[8] Quadibloc. Basic Tilings: The 17 Wallpaper Groups. `http://www.quadibloc.com/math/tilint.htm`, 2025.

[9] Math Open Reference. Radius of a regular polygon, 2011. URL `https://www.mathopenref.com/polygonradius.html`.

[10] Vivek Sasse. Classification of the 17 wallpaper groups. Research report, University of Chicago Mathematics REU, 2020.

[11] Scott Sutherland. Isometries, 2014. URL `https://www.math.stonybrook.edu/~scott/mat515.fall14/isometries.pdf`. Lecture notes from MAT515, Stony Brook University.

[12] The Sage Developers. *SageMath, the Sage Mathematics Software System (Version 10.6)*, 2025. `https://www.sagemath.org`.

[13] J. Top and J.S. Müller. Group theory, 2018. URL `https://www.rug.nl/staff/steffen.muller/lecture_notes_group_theory.pdf`. Lecture notes from 2nd year bachelor mathematics, University of Groningen.