Discrete Groups and Computational Geometry

A thesis presented in partial fulfilment of the requirements

for the degree of

Doctor of Philosophy

in

Mathematics

at Massey University, Albany, New Zealand

Haydn Mark Cooper

2013

ABSTRACT

Let f and g be Möbius transformations with finite-orders p and q respectively. Further, let $\gamma = tr[f, g] - 2$, where $tr[f, g]$ is the trace of the commutator of f and g in the standard $SL(2, \mathbb{C})$ representation of Möbius transformations.

The group $G = \langle f, g \rangle$ is then defined, up to conjugacy, by the parameter set (p, q, γ), whenever $\gamma \neq 0$. If the group G is discrete and non-elementary, then it is a Kleinian group. Kleinian groups are intimately related to hyperbolic 3-orbifolds.

Here we develop a computer program that constructs a fundamental domain for such Kleinian groups. These constructions are undertaken directly from the parameters given above. We use this program to investigate, and add to, recent work on the classification of arithmetic Kleinian groups generated by two (finite-order) elliptic transformations.

ACKNOWLEDGEMENTS

The completion of this project is due, in no small part, to the help, encouragement and advice that I have received along the way. So I must give full credit to those I have met on this journey.

Foremost to my supervisors: Distinguished Professor Gaven Martin, for the ideas and encouragement in the development of this project along with his confident support throughout the many years since I entered graduate school; and Dr Winston Sweatman, for his constant guidance, encouragement and interest in every facet of my work. Without their support, and confidence in my ability to succeed, I'm sure none of this would have happened. On meeting them as an undergraduate student I never expected I would one day be researching alongside them; though I fear I may have used up all their patience, leaving none for their future students.

I can never forget my wonderful family, the support of my parents Peter and Alma, and my lovely wife Anna who imbued me with the impetus to sit down and actually get all this finished. Along with my children, Gabriella and Orson; to them, and their constant distraction, any and all work I ever undertake must surely be dedicated.

No journey would be complete without a dose of similarly weary travellers, without whom we would surely all go mad. So I extend a thanks to my friends and colleagues; especially those other postgrads who have slowly processed through Massey University's IIMS and NZIAS departments. As with all academic endeavour, little would be possible without the administrative staff, whose tireless assistance should never be forgotten, nor go without mention. I will not attempt to name the many of you who have had to endure my curious notions and quiet triumphs over these long years.

There is nothing more harrowing than a good education.

A hearty thank you to you all.

`Haydn M Cooper`

`March 2013`

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

Over the last few decades the theory of Kleinian groups has flourished through its deep connections to low dimensional topology and geometry. The highlight of these recent studies being G. Perelman's recent proof[1] of Thurston's geometrisation conjecture, which states that any compact 3-manifold can be decomposed canonically into submanifolds that have geometric structures; see [48]. These geometric structures fall into eight possible types, the most prevalent and least understood of which are the submanifolds with hyperbolic structures. The geometrization conjecture also provides an analogue, for 3-manifolds, of the uniformization theorem for surfaces.

This latter result, the uniformization theorem, is a noted achievement by mathematicians of the 19th century; implying, for instance, the Poincaré conjecture. Additionally, there have been many other recent advances in this field, including: the density conjecture, [2],[13]; the ending lamination conjecture, [12]; the surface subgroup conjecture, [35]; and the virtual Haken conjecture, [3].

While we do not discuss all these results here (nor offer any statements of precise theorems), together they provide a remarkably complete picture of the structure of hyperbolic isometry group actions and their quotient spaces. These quotient spaces being 3-dimensional hyperbolic manifolds and orbifolds (where the addition of a precise type of singular structure is allowed). The study of these group actions, the study of Kleinian groups, is the focus of this thesis.

Kleinian groups are non-elementary discrete groups of orientation-preserving isometries acting on hyperbolic 3-space. Non-elementary and discrete implying that the group does not have an abelian subgroup of finite index.

We consider hyperbolic 3-space, X, and the group of Möbius transformations $\mathcal{M}$, the fractional linear transformations acting on the extended complex plane $\hat{\mathbb{C}} = \mathbb{C} \cup \{\infty\}$. As a Lie group, the isometry group $Isom^+(X)$ is isomorphic to $\mathcal{M}$. A common model for hyperbolic 3-space is the upper half-space

$$U = \{(x, y, t) \in \mathbb{R}^3 \mid t > 0\}$$

[1] 2006 Fields Medal.

with a metric

$$ds^2 = \frac{dx^2 + dy^2 + dt^2}{t^2}.$$

With the $t = 0$ hyper-plane of $\mathbb{R}^3$ then being identified with $\mathbb{C}$, the complex plane, allowing us to equate ∂X with the Riemann sphere $\hat{\mathbb{C}}$. And Möbius transformations can take both a functional and a matrix form, often being presented as the projective special matrix group $PSL(2, \mathbb{C})$, of complex 2×2 matrices

$$\pm \begin{bmatrix} a & b \\ c & d \end{bmatrix}, \; ad - bc = 1$$

under matrix multiplication.

Work in classifying Kleinian groups began over 100 years ago with Klein's Erlanger program which aimed to demonstrate the deep links between geometry and group theory; and, as mentioned above, the past decades have seen considerable renewed interest in these groups underscoring the deep links between the study of groups, complex analysis and the classification of geometric spaces.

Of considerable relevance to us, among these recent works, are the computational studies of hyperbolic structures on knot and link complements, ideal triangulations, and generalisations of Poincaré's theorem for fundamental polygons. Particularly through J. Weeks' programme SnapPea[2], [59]; a demonstration of its computational application in the greater theory is manifested by the census of hyperbolic manifolds.

Before discussing the specifics of the work in this thesis, we demonstrate the basic links between groups and spaces with the simple Euclidean example of the torus (a 2-manifold). As shown in Figure 1.1.

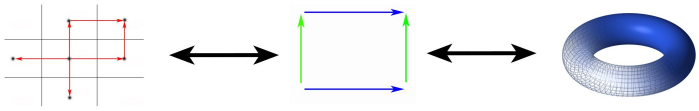


Fig. 1.1: The torus, T^2.

The torus is among the simplest of closed manifolds, being the quotient of Euclidean 2-space under the action of a group G generated by two linearly independent translations on the real-plane. This group action identifies a quadrilateral which tessellates the entire plane by copies of itself translated by the group - a *fundamental domain*, $\mathbb{E}^2/G$. The quotient map induced by the group action is an equivalence relation on the plane - two points are the same if there is a group element whose action maps one to the other. This equivalence relation then

[2] 1999 MacArthur "Genius" prize.

wraps up the boundary of a fundamental domain - giving something isometric to $T^2 = S^1 \times S^1$.

Simply switching the direction of the identification between two edges (in this torus example), the quotient space becomes considerably more complicated. In this situation it yields the Klein bottle, a non-orientable 2-manifold that cannot be embedded in Euclidean 3-space.

Working in 3-dimensions, and with the added complexity of hyperbolic space, gives rise to much more interesting, and complicated, tessellations and fundamental domains. For example, see the successful construction in Figure 1.3. This thesis focuses on the construction of such domains for a variety of applications.

Traditionally, in all but simple cases, the discreteness of a given subgroup of $\mathcal{M}$ is very difficult to determine, [30]; with there being a lack of criteria that are both sufficient and necessary for the condition. This has lead to classification work focusing on specific classes of groups. To this end, Fuchsian groups, the 2-dimensional counterparts to Kleinian groups, have long been classified; as have the elementary discrete groups. Similarly, manifolds and orbifolds are more or less classified for all but this hyperbolic case, [19]; with the study of Kleinian groups having largely focused on the torsion-free (manifold) cases, explicitly leaving the classification of hyperbolic 3-orbifolds as a special case to be dealt with later [42].

Recently, focus has turned to attempts at identifying universal constraints on the geometry of Kleinian groups, such as minimal covolume, the Margulis constant, collaring theorems and more. This work has led to intense study of spaces of two-generator Kleinian groups, further encouraged by the following theorem.

Theorem (Proposition 2 of [33]). *Let G be a non-elementary subgroup of $\mathcal{M}$. Then G is discrete if and only if every two-generator subgroup $\langle f, g\rangle$ is discrete for every $f, g \in G$.*

We focus on the two-generator subgroups, $G = \langle f, g\rangle$, of $\mathcal{M}$. These can be parameterised, up to conjugacy, by the ordered set of complex numbers

$$par(G) = (\beta, \beta', \gamma) \in \mathbb{C}^3,$$

where

$$\begin{aligned} \beta &= tr^2 f - 4, \\ \beta' &= tr^2 g - 4, \\ \gamma &= tr[f, g] - 2. \end{aligned}$$

These traces coming from the matrix representation of Möbius transformations.

This parameterisation of two-generator groups opens access to alternative means of studying these groups; and allows for the elegant expression of a number of discreteness results, for example we have the following theorem.

Theorem (Jørgensen's inequality, [33]). *If $\langle f, g\rangle$ is a Kleinian group, then*

$$|\beta| + |\gamma| \geq 1.$$

Through the use of Jørgensen's necessary requirement for discreteness, given above, and many similar "inequalities for discreteness", numerous results have been realised with respect to the parameter spaces of discrete groups, along with geometric descriptions of a group's generating elements; see [28], [57]. These studies typically begin by describing the space of two-generator discrete subgroups of $\mathcal{M}$ using these various generalisations of Jørgensen's inequality based around an intriguing family of polynomial trace identities in $SL(2, \mathbb{C})$, [25],[26].

These polynomials identify inequalities that hold except for certain specific parameter sets. We highlight the following simple example.

Theorem. *If $\langle f, g\rangle$ is a Kleinian group, then*

$$|1 + \beta| + |\gamma| \geq 1$$

unless $\gamma = 1 + \beta$ and $\langle f, g\rangle$ is Nielsen equivalent to a group generated by two elements of finite order 2 and 3.

From which the question arises: what are the γ-values for a Kleinian group generated by elements of finite order 2 and 3, particularly those with $\gamma = 1 + \beta$.

First, there are infinitely many such values, namely

$$\gamma = 1 - 4\sin^2(\pi/p) = 1 + \beta, \qquad p \geq 7.$$

These correspond to the $(2, 3, p)$ triangle groups (the groups generated by reflection in the sides of a hyperbolic triangle with interior angles $\pi/2, \pi/3$, and π/p), but there are others. A few of these groups are arithmetic groups (arithmetic groups being mentioned further below and discussed in Section 2.3.4), but the question arises as to how to determine discreteness for the remainder.

In this situation, (for this and other inequalities) we generally do not know *a priori* if a given parameter set corresponds to a discrete group, and in most cases it will not. However, in important cases it may well be discrete. Part of our work here is set up to help determine where possible, when these "exceptional points"

lead to discrete groups, by constructing a fundamental domain, calculating its volume (which may be infinite) and giving other clues as to its identity; the failure to construct such a domain otherwise suggesting that the group is not actually discrete.

Building on these inequalities it is possible to determine general *a priori* bounds, or "universal constraints", which hold under various assumptions; but will not generally hold for the extremal case. These assumptions often take the form of geometric requirements related, for example, to distances between elements of the singular set or between the axes of group elements. The challenge then is to try prove that if these assumptions fail, then specific geometric configurations must occur which produce a two-generator subgroup of a specific type.

This is achievable as the failure of these assumptions often confines the investigation to a small sub-space of two-generator discrete groups that is supposed to have been *a priori* completely described. This reduces possible extremal groups to a finite list of candidates for further consideration; often, many of which are arithmetic. Once it is shown that the examples which the *a priori* bounds do not cover are arithmetic, other techniques can be used to identify extremals from this finite list. In this way it it has been shown that most examples of small volume hyperbolic orbifolds and manifolds have an arithmetic structure; and, more generally, the same is true for other types of extremal problems. To summarise, the identification of extremal manifolds and orbifolds is predicated on the assumption that the extremals for these geometric problems are two-generator arithmetic Kleinian groups, and this is generally the case.

Working with finite lists of (often) arithmetic groups relates to another part of the overall research program on Kleinian groups, that is, the enumeration of all the two-generator arithmetic Kleinian groups. In this thesis we make a major advance in that programme with a "computational" classification of all the arithmetic groups generated by two elements of finite orders 2 and p, $p = 3, 4, 6$. These cases have previously been identified as the most difficult cases to deal with. These methods should ultimately deal with the remaining cases, but more research related to specific classes of algebraic integers within ellipses of the complex plane needs to be completed first.

If we consider the case of Kleinian groups generated by a pair f and g of finite order (elliptic) transformations, with orders p and q respectively, then

$$\begin{aligned} \beta &= tr^2 f - 4 &&= -4\sin^2(\pi/p), \\ \beta' &= tr^2 g - 4 &&= -4\sin^2(\pi/q), \end{aligned}$$

and there is then the alternative parameter set

$$(p, q, \gamma),$$

which we will consider closely in this thesis.

By focusing on specific p and q values, the results mentioned above have been combined to provide a growing description of the γ parameter space, [23]. Often these groups correspond to arithmetic lattices (nice, discrete groups) and their subgroups; the distinction between the groups and their proper subgroups being that arithmetic groups are of finite covolume.

This is another area in which our work contributes. The arithmetic criteria obtained are sufficient to show that a group is not only discrete, but a (discrete) subgroup of an arithmetic group. However, it provides no indication as to whether or not the group has finite covolume, nor the orbifold's underlying topology. Under Mostow rigidity, [49], (co)volume is an important topological invariant of these (groups and) orbifolds, so its identification from a possible γ-value is an important component in the facilitation of these identifications. Our computer programme (mentioned below) computes this volume, directly from γ. Once a possible candidate has been identified from this volume data (matched against a list of data on things like orbifold surgeries on the cusped census) explicit calculations can be made to confirm the correct identification of the group. Note though that it is possible for two different orbifolds to have the same volume.

All this arithmetic work, combined with other recent work, leads to a fairly complete descriptions of some γ-parameter spaces. These descriptions combine a free boundary, beyond which all γ-values correspond to free groups, [24]; an iterative disc-covering procedure, projecting discs based on Jørgensens's and similar inequalities, within which the group (p, q, γ) cannot be discrete with the possible exception of certain special points (as described in the example above), [28]; and a recent knot-based investigation that has begun to establish the structure of the fractal "free" boundary, [62]. The results of these are demonstrated for the $(3, 2, \gamma)$ groups in Figure 1.2.

These disc covering methods provide a plethora of parameter sets for groups generated by two elliptic transformations. As noted above, in some cases the γ parameters are analytically known to be discrete, while in other cases the question of discreteness remains. For example, the $(7, 2)$-commutator plane, analogous to that above, has no arithmetically determined points.

The attempted construction of a fundamental domain then allows the possible

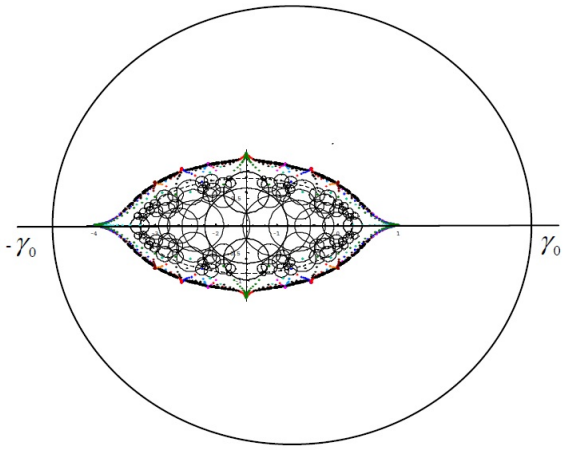


Fig. 1.2: The $(3, 2, \gamma)$ parameter space, $\mathbb{C}$.
(Figure 1.3 of [62].)

confirmation of the group as a discrete group. If a successful construction is achieved and a covolume is calculated, it can then be compared against the orbifold data arising from Dehn surgeries on hyperbolic knot complements. Thus this work offers one of the few possible ways to identify the structures of parameter slices when we do not expect many of the groups to be arithmetic.

Thus our project is part of this area of research, with a focus on computationally determining if a group is discrete and, if so, if it has finite covolume, through the generation of a fundamental domain, directly from a given parameter set. As a part of the overall investigation into the two-elliptic-generator case, we task ourselves the goal of providing an informative description of the construction of fundamental domains for the Klienian groups generated by two elliptic elements; deliberately allowing for an experimental approach to the construction of a group's fundamental domain. Applications of computational methods for the construction and study of fundamental domains were first demonstrated in [56]; an application which has revolutionised the approach to this area of study.

To this end, we have developed a computer program, SnappyD, for the construction of fundamental domains for these groups; the use of an earlier version, titled F-elliptic, has already appeared in the results of [62]. This program is based solidly on a specific set of subroutines found in the well used and otherwise adapted program SnapPea [59]; we have chosen to use this program as the basis of our own as its use in this field is well noted, and it contains an impressive level of documentation.

This work also developed from an investigation into the failures of SnapPea

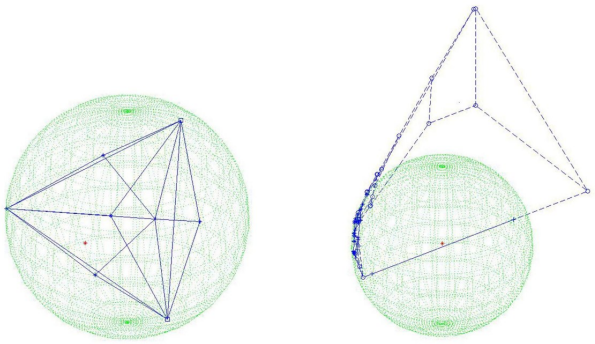


Fig. 1.3: A successful (left), and an unsuccessful (right), construction.

(based programs), when attempting to determine fundamental domains for Kleinian groups with infinite, as opposed to finite, covolume; see Figure 1.3. Thus a goal of this work is the computation and successful construction of the representative quotient spaces for such groups.

This computer program has then allowed us to construct and study fundamental domains for a large number of groups; groups which have featured in the several, recent publications related to the classification of arithmetic hyperbolic 3-orbifolds and investigation of Kleinian groups; see [23], [60] and [62]. The work here reinforces the results seen in these references, based on which we give the following result which relies on the "computational boundary" identified by Zhang in [62].

Theorem. *Let $G = \langle f, g \rangle$ be an arithmetic Kleinian group where f is an elliptic transformation of order 3, 4 or 6; and g an elliptic transformation of order 2. Then G is listed in the tables of Chapter 5.*

More specifically, we have reaffirmed the covolume results seen in [23] and [62]; adding values missing from their tables and confirming those values already seen. In doing this we give Theorem 5.3.3 in Section 5.3.1, an expansion of Theorem 8.2 of [23], and complete Table 1 of [60]. And, under the conjectures of [62], we further refine the complete list of two-generator arithmetic groups that are generated by an element of order 2 and order 3, 4 or 6; leaving only 18 possible members still in need of confirmation, all of which appear to lie extremely close to the previously mentioned fractal boundary.

In using these computational methods, we highlight the value of these programs and their ability to lend considerable insight into the methods used to construct domains in these, and related, applications. This insight is exemplified in our identification of pairs of (p, q, γ) and $(p, q, \beta-\gamma)$ parameter sets that demonstrate

parallax (Section 5.2.4) in their volumes. Results based on these groups (and the relevant fundamental domains) have been used in previous work; but as far as we are aware, this is the first instance that this parallax nature has been highlighted in relation to the classification of arithmetic groups.

Thesis Overview

We begin in Chapter 2 with a preliminary overview of Kleinian groups through an introductory treatment of Möbius transformations, before highlighting group properties and recent work on the classification of the Kleinian groups in [23], [60] and [62]. In doing this we establish our focus as related to the investigation of two-elliptic-generator Kleinian groups, and their (p, q)-commutator spaces. From there, we describe the geometric action of a discrete group, laying the framework for the computational methods used in subsequent chapters. This is achieved through a detailed description of fundamental domains and their existence through Dirichlet domains and fundamental polyhedrons.

This allows us to outline the `Dirichlet_` subroutines of the program SnapPea in Chapter 3. These routines form the basis of our program, so we attempt a thorough exposition of the underlying routines and spatial description on which the program runs, while taking care to avoid the minute trivialities of the implemented programming. Using this description of the basis routines, Chapter 4 details the considerations necessitated in their application to the two-elliptic-generator groups that we are interested in; with regard to achieving constructions for domains of infinite covolume, within the Euclidean representation of polyhedral structures.

We supplement these process descriptions with judicious use of figures and flowcharts. And, in Section 3.4.1, provide code-excerpts of the top-level processes from the `Dirichlet_` subroutines of SnapPea [59], along with a list of computational constants.

The resultant program, SnappyD, has then been used to construct domains for the groups given in [23], [60] and [62]; with detailed summary tables provided in Appendix C. In Appendix B we also provide a simple overview of the execution of the program, its variants and the contents of the support files.

In Chapter 5 a reduction and description of these tables is provided. This is begun with a detailed description of the results for the generalised arithmetic triangle groups of [60]; before we discuss the outlier data and similar results that our computations have uncovered. We summarise these results in Section 5.3, listing polynomial tables and covolume results as relevant to the classification of arithmetic groups. We end with the classification given in [62], and the

furthering of several group results given in [23].

Among these results we highlight the existence of a number of groups that display parallax in their covolume results, and groups that fail the cycle condition on Poincaré polyhedrons. These groups include those used in the work of recent references around which extra consideration may be needed. We also point to a possible categorisation of Fuchsian group output based on basepoint movements.

We have written this with an aim of providing a reference for any similar projects that other researchers may desire to embark upon; focusing on the overall geometric, and computational, aspects of the project. By doing this we hope to maintain a level of accessibility within the work, providing a thorough description of the relevant computational action of SnappyD (SnapPea).

As we deal with several models of hyperbolic geometry, we avoid detailing any model specifically; instead providing a brief overview of the four canonical models, their relation to one another and their representative isometry groups in Appendix A. Given this we note several key references, specifically the texts [10], [41] [44] and [55] which are considered core references for research into Kleinian groups. In addition to these, [41] is a definitive reference on arithmetic groups and [51] provides a foundation in topology.

2. THE GEOMETRY OF KLEINIAN GROUPS

Here we outline the properties of Kleinian groups, as relevant to our interests. The general results of this section are detailed across [10], [32], [44] and [55]; with an introductory account provided in most papers given in the bibliography.

2.1 Discrete Groups

Let X be a topological metric space and G a group of transformations on X; G is then discrete if the identity, I, of G is isolated in the topology of local uniform convergence. Any subgroup of a discrete group is also discrete.

For our purposes X will be a model of hyperbolic 3-space and G will be a subgroup of $Isom^+(X)$, the orientation-preserving isometries of X. We describe these background spaces in Appendix A.

Let x be any element of X, then:

- The stabilizer of x is the subgroup $G_x = \{g \in G : gx = x\}$; and
- The orbit of x is the set $G(x) = \{gx : g \in G\}$.

Any two points $x, y \in X$ are in the same orbit if and only if $G(x) = G(y)$. In this way the orbits partition X.

G is said to act discontinuously on X if and only if for every compact subset K of x, and for all but finitely many g in G,

$$g(K) \cap K = \emptyset.$$

If $g(K) \cap K = \emptyset$ for all $g \neq I$, then G is said to act properly[1] discontinuously on X.

If G acts discontinuously on X, then each stabilizer G_x is finite and each orbit $G(x)$ is a closed discrete subset of X.

Definition 2.1.1. *Let X be topological space and let G be a group of transformations acting discontinuously on X. Then a fundamental set S for G in X is a subset of X containing one, and only one, point from each orbit $G(x)$.*

[1] Freely discontinuous in some references.

Clearly, a fundamental set is a partition of X such that any two points x and y, in X, are in the same partition if, and only if, there exists a transformation g, in G, such that $gx = y$. This provides a natural quotient space, the orbit space $X/G = X/\sim$, where the quotient mapping is: $x \sim y$ if and only if there exists g in G, such that $g(x) = y$.

Let X and G be as above, and let $vol()$ be a volumetric measure on X. Then a subset D of X is referred to as a fundamental domain of G in X if and only if:

- D is a domain (an open connected subset of X);
- there exists a fundamental set S, such that $D \subset S \subset \overline{D}$; and
- the $vol(\partial D)$ is zero.

This last criterion is not always used in the definition for a fundamental domain; if a fundamental domain satisfies this criterion, then it is said to be *proper*, otherwise it is an *improper* fundamental domain.

Definition 2.1.2. *A fundamental domain D for G in X, is said to be locally finite if and only if for every compact subset K of X the set*

$$\{g \in G \mid g(\overline{D}) \cap K \neq \emptyset\}$$

is finite.

Fundamental domains provide a useable, geometric realisation of a quotient space. A group G is cocompact if its fundamental domain has compact closure; and is defined to have covolume equal to the volume of its fundamental domain.

2.2 Möbius Transformations

Consider the Riemann sphere, $\hat{\mathbb{C}} = \mathbb{C} \cup \{\infty\}$, and the fractional linear transformations, $f : \hat{\mathbb{C}} \mapsto \hat{\mathbb{C}}$,

$$f : z \mapsto \frac{az + b}{cz + d},$$

where $ad - bc = 1$. These mappings are known as Möbius transformations, and form a group under the composition of functions. We denote this group by $\mathcal{M}$.

This group of Möbius transformations is then the group of orientation-preserving transformations on $\hat{\mathbb{C}}$ that are the finite compositions of reflections in lines and circles. It follows that each element of $\mathcal{M}$ has a unique representation in, and that $\mathcal{M}$ is isomorphic to, the matrix group $PSL(2, \mathbb{C}) \approx SL(2, \mathbb{C})/\{\pm I\}$.

We will follow the general tradition of abusing notation, whereby we simultaneously consider a Möbius transformation as both a function and a matrix; and

further, we will consider matrices in $SL(2,\mathbb{C})$ in the understanding that they are a double cover of $PSL(2,\mathbb{C})$.

$$f = \pm \begin{bmatrix} a & b \\ c & d \end{bmatrix}, \; ad - bc = 1.$$

In this way we consider f to have a trace value $tr(f) = \pm(a+d)$, and a trace-square value $tr^2(f) = (a+d)^2$; noting that $tr(f)$ is dependant on the chosen matrix representation of f, where $tr^2(f)$ is not.

Through the Poincaré extension, below, these transformations can be extended to represent the orientation-preserving isometries on the upper half-space model of hyperbolic 3-space, see Appendix A. Thus the classification of Möbius transformations following in the rest of this section carries over onto the group of orientation-preserving isometries $Isom^+(X)$.

It is common in the literature to forgo the label $\mathcal{M}$ and refer directly to $PSL(2,\mathbb{C})$ as a representation of $Isom^+(X)$. We do not do this here as different models of hyperbolic 3-space have different, though isomorphic, representative matrix groups and we will not be immediately focusing on a single background model as is commonly the case.

2.2.1 The Poincaré Extension

We consider the extended real space $\hat{\mathbb{R}}^n = \mathbb{R}^n \cup \{\infty\}$, and note the equivalence between $\hat{\mathbb{R}}^2$ and $\hat{\mathbb{C}}$. The Möbius group on $\hat{\mathbb{R}}^n$ is then denoted $M(\hat{\mathbb{R}}^n)$ and is the group of orientation-preserving transformations that are the finite compositions of reflections in hyper-planes and hyper-spheres of $\hat{\mathbb{R}}^n$. The Möbius group on $X \subset \hat{\mathbb{R}}^n$, denoted $M(X)$, is then the subgroup of $M(\hat{\mathbb{R}}^n)$ preserving X.

By considering $\mathbb{R}^2$ to be the (x,y)-plane in $\mathbb{R}^3$ and by taking each reflecting line and circle, used to compose an element of $M(\hat{\mathbb{R}}^2)$, to be the intersection of $\mathbb{R}^2$ with a reflecting orthogonal plane or sphere, the action of $M(\hat{\mathbb{R}}^2)$ can be extended to act on $\hat{\mathbb{R}}^3$. This extended action preserves, and is symmetric on either side of, the (x,y)-plane; thus it is common to restrict consideration of this action to the upper half-space $\hat{U}^3 = \{(x,y,z) \in \mathbb{R}^3 : z \geq 0\} \cup \{\infty\}$ and note that $M(\hat{\mathbb{R}}^2) \approx M(\hat{\mathbb{U}}^3)$.

This construction is known as the Poincaré extension, and the group of orientation-preserving compositions of reflections in hyper-spheres and hyper-planes, in $\hat{\mathbb{R}}^n$, that preserve $\hat{U}^n$ is isomorphic to $M(\hat{\mathbb{R}}^{n-1})$. More specifically, $M(\hat{U}^3)$ is isomorphic to $\mathcal{M}$.

Note that the action of Möbius transformations, as extended onto $\hat{U}^3$ above, can

be easily determined by extending their fractional linear form with quaternion algebra, see [10]. Further, the boundary ∂U^3 is the extension of the (x, y)-plane, $\mathbb{R}^2 \cup \{\infty\} \approx \hat{\mathbb{C}}$; exposing the link between a group's action on a space's boundary and its action on the space within.

2.2.2 Conjugacy Classes and Trace Classification

Let f and g be elements of the group G. Then f is said to be conjugate (in G) to g if and only if there exists h (in G) such that $hgh^{-1} = f$. This is an equivalence relation, and it is common to consider the conjugacy classes of group elements as a means of considering transformations by their general "geometric" actions as opposed to explicit form.

Let k be any non-zero complex number, we define the Möbius transformation m_k, in function form, by

$$m_k(z) = kz \text{ (if } k \neq 1); \text{ and } m_1 = z + 1.$$

These Möbius transformations are referred to as standard forms. Every non-identity element of $\mathcal{M}$ is conjugate to one, and only one, of these standard forms.

$$gfg^{-1} = m_k, \text{ for some } g \in G \text{ and some } k \in \mathbb{C} \setminus \{0\}.$$

Hence these standard forms can be used to represent each of the conjugacy classes of $\mathcal{M}$.

Let f, g be any two elements of $\mathcal{M}$. Then f is conjugate to g if and only if $tr^2(f) = tr^2(g)$; and further

$$tr^2(m_k) = k + 2 + 1/k.$$

Thus it is common to classify the conjugacy classes of $\mathcal{M}$ by both their standard forms (geometric actions) and tr^2 values.

Definition 2.2.1. *Let f be any non-identity element of $\mathcal{M}$, then*

1. *f is parabolic if and only if $tr^2(f) = 4$*
   (equivalently f is conjugate to the standard form m_1);

2. *f is elliptic if and only if $tr^2(f) \in [0, 4)$*
   (equivalently f is conjugate to a standard form m_k, where $|k| = 1$, $k \neq 1$);

3. *f is hyperbolic if and only if $tr^2(f) \in (4, +\infty)$*
   (equivalently f is conjugate to a standard form m_k, where $|k| \neq 1$, $k \in \mathbb{R}$);

4. *f is loxodromic*[2] *if and only if* $tr^2(f) \notin [0, +\infty)$
   (equivalently f is conjugate to a standard form m_k, where $|k| \neq 1$, $k \notin \mathbb{R}$).

Notice that, when acting on $\hat{\mathbb{C}}$, a parabolic transformation has a single fixed point, and the other classes of transformations each have two fixed points. Acting in U^3, each non-parabolic transformation f has an axis, the unique geodesic that arcs between its fixed points on $\hat{\mathbb{C}}$.

For a more detailed description of Möbius transformations and their classification see [10], [44] and [55].

2.2.3 Elliptic Elements

We take particular interest in elliptic transformations, as they form a key part of the groups that we will be investigating.

Elliptic transformations are conjugate in $\mathcal{M}$ to transformations of the (standard) form

$$f : z \mapsto e^{i\theta} z,\ 0 < \theta < 2\pi.$$

This makes them rotations of θ about their axes, and thus comparable to the rotational transformations of Euclidean geometry.

Note that these transformations have standard matrix representation:

$$f = \begin{bmatrix} e^{i\frac{\theta}{2}} & 0 \\ 0 & e^{-i\frac{\theta}{2}} \end{bmatrix};$$

And so the trace-squared value for an elliptic element is

$$tr^2(f) = 4\cos^2(\theta/2).$$

Consider a non-identity transformation f in $\mathcal{M}$, and let n be the least positive integer such that $f^n = I$. Then f is said to be of (finite) order n; if no such n exists, then f has infinite order. All non-identity, orientation-preserving isometries with finite order are elliptic. An elliptic transformation is called a primitive (elliptic) transformation of order $n \geq 3$ if it is of order n and $tr^2(f) = 4\cos^2(\pi/n)$.

A group G is said to be torsion-free if it has no non-trivial elements of finite-order; thus a subgroup of $\mathcal{M}$ is torsion-free if and only if it contains no elliptic transformations.

[2] Usage of the term loxodromic is not consistent in the literature, where it is often extended to include the hyperbolic transformations.

2.2.4 Limit Sets and Ordinary Sets

Let G be a discrete subgroup of $\mathcal{M}$, considered to be acting on $\hat{\mathbb{C}}$, and let Λ_0 be the set of fixed points for all hyperbolic and loxodromic elements in G.

$$\Lambda_0 = \{z \in \hat{\mathbb{C}} \mid \text{there exists } f \in G, tr^2(f) \notin [0,4] \text{ such that } f(z) = z\}.$$

The limit set Λ is the closure of Λ_0 in $\hat{\mathbb{C}}$, and the complement of the limit set, $\Omega = \hat{\mathbb{C}} \setminus \Lambda$, is referred to as the ordinary set[3]. If G is torsion-free then it acts properly discontinuously on Ω, and discontinuously otherwise.

If Λ contains more than 2 points then G is a *non-elementary* group, otherwise G is an *elementary* group. Λ is the smallest non-empty G-invariant closed subset of $\hat{\mathbb{C}}$; and if G is non-elementary, then Λ is a perfect set[4].

2.3 Kleinian Groups

Our specific interest is in the non-elementary discrete subgroups of $\mathcal{M}$, these groups being collectively known as Kleinian groups[5].

Kleinian groups are intimately related to hyperbolic geometry; the group $\mathcal{M}$ being isomorphic to $Isom^+(X)$; where X is hyperbolic 3-space, as described in Appendix A. In fact the quotient space X/G, when G is a Kleinian group, is isometric to an orientable hyperbolic 3-orbifold; when G is torsion-free, then X/G is also referred to as an orientable hyperbolic 3-manifold.

It is noticeable that many references refer to orbifolds inclusively as manifolds, or to avoid them as an exceptional class of groups to be dealt with later; presumably this is due to the term, and relevance of, orbifolds not having been highlighted until work focusing on the classification of Kleinian groups and manifolds was well underway, [58]. See, for example, [42].

Let G be a Kleinian group, then, as in Section 2.1, G is said to be cocompact if and only if X/G is compact; and, similarly a Kleinian group G is said to have (in)finite covolume if and only if X/G has (in)finite volume with respect to its induced hyperbolic metric. A Kleinian group with finite covolume is non-cocompact if and only if it contains parabolic elements.

Kleinian groups are those non-elementary groups which act discontinuously on X.

[3] The ordinary set is also often refered to as the domain of discontinuity or the regular set.

[4] A set S is perfect if it contains all of its limit points; equivalently, S is perfect if S is closed and has no isolated points.

[5] In the literature, There is some variation over the definition of Kleinian groups; primarily in the restriction to non-elementary groups.

Theorem 2.3.1. *Let G be a non-elementary subgroup of $\mathcal{M}$, then G is discrete if and only if G acts discontinuously on X.*

Note that this theorem does not hold for the closure of X. If a group $G \subset \mathcal{M}$ acts discontinuously on X, then it is discrete; however a group may be discrete while not acting discontinuously on ∂X. For example, take $f(z) = 2z$, then $\langle f \rangle \subset \mathcal{M}$ is a discrete group that does not act discontinuously on $\mathbb{C}$. When we refer to the general action of an element it will generally be with respect to its extended action on $\overline{X}$.

2.3.1 Elementary, Fuchsian and Degenerate Groups

While not our specific focus, for the sake of completeness we also describe a number of other types of subgroups of $\mathcal{M}$.

Recall that a discrete group, G, is said to be elementary if the limit set Λ contains no more than two points, or, equivalently, if it has a finite orbit in $\mathbb{R}^3$; otherwise it is said to be non-elementary. If all elements in G have a common fixed point, then G is referred to as reducible; all reducible groups are elementary.

Further, if Ω is empty then a Kleinian group G is said to be of the first kind; otherwise it is of the second kind. If a Kleinian group has finite covolume, then it is of the first kind.

If each element in G preserves some disc or halfplane in $\hat{\mathbb{C}}$, then G is called a Fuchsian group; in which case, G is conjugate to a subgroup of $PSL(2, \mathbb{R})$. Fuchsian groups are a well studied set of groups that are intimately related to the isometries of hyperbolic 2-space in the same way Kleinian groups are to hyperbolic 3-space.

The set of Fuchsian groups has been classified, as have the elementary groups, see [10]. The torsion-free Kleinian groups have also been heavily studied; a good reference for this class of groups, following Poincaré's treatment of Fuchsian groups, can be found in [42].

There are numerous other classes of Kleinian groups. We highlight several for the case of G, where G is a finitely generated Kleinian group.

- G is a quasifuchsian group if Λ is a Jordan curve and there are no elements in G that interchange the components of Ω.
- G is an extended quasifuchsian group if Λ is a Jordan curve and there are elements in G that interchange the components of Ω.
- G is a degenerate group if Λ and Ω are both non-empty and connected.

- G is a web group if Ω contains infinitely many components each with a quasifuchsian group stabilizer.

Note that an extended quasifuchsian group has a quasifuchsian subgroup of index two. And the boundary of each component of Ω in a web group must be a Jordan curve.

Determining the discreteness of a given subgroup in $\mathcal{M}$ is generally a difficult problem, see [30]. There are many sufficient, or necessary, conditions; but no simple conditions that are both necessary and sufficient for discreteness. This leaves the classification of Kleinian groups an active area of research. Aiding these investigations are the notable properties of Kleinian groups to decompose when finitely generated; as investigated in [1], [61] and their references. Specifically, any finitely generated Kleinian groups can be "constructed" from a set of elementary groups, totally degenerate groups and web groups. More relevant to our work is the following well known result.

Theorem 2.3.2 (Proposition 2 of [33])**.** *The group G is a Kleinian group if and only if, for each pair $f, g \in G$, the two-generator group $\langle f, g\rangle \subset G$ is a Kleinian group.*

Given the results above, it is common to study general Kleinian groups through these more specific groups. We note that this research follows the above result in restricting our attention to certain types of two-generator Kleinian groups; specifically, our interest will be in groups generated by a pair of elliptic transformations, i.e. Kleinian groups with torsion, relating to hyperbolic 3-orbifolds.

2.3.2 Parameterisation of Two-Generator Groups

For any two elements $f, g \in \mathcal{M}$ we define the commutator $[f, g] = fgf^{-1}g^{-1}$, and note that $tr[f, g]$ is invariant under conjugation, matrix representation and alternating the order of f and g.

Using this, the trace-squared parameterisation given in Section 2.2 can be expanded to a parameterisation of two-generator groups.

Definition 2.3.3. *Let $G = \langle f, g\rangle$ be a two-generator subgroup of $\mathcal{M}$. Then we define the parameter set of G to be*

$$par(\langle f, g\rangle) = (\beta, \beta', \gamma)$$

where

$$\begin{array}{rcccl} \beta & = & \beta(f) & = & tr^2(f) - 4, \\ \beta' & = & \beta(g) & = & tr^2(g) - 4, \\ \gamma & = & \gamma(f, g) & = & tr[f, g] - 2. \end{array}$$

These parameters have been normalised (as compared to the individual parameters given in Section 2.2.2) so that f is parabolic if and only if $\beta(f) = 0$. Note that f and g then share a common fixed point if and only if $\gamma(f, g) = 0$. Recall that if the generators share a fixed point, then all elements of G share a common fixed point; in which case G is reducible and, thus, elementary.

In the literature it is common to have $par(\langle f, g\rangle) = (\gamma, \beta, \beta')$. We give the version above for consistency, as we will later use a modified parameter set of the form (p, q, γ), see Section 2.4.

Using this parameterisation we give the following well known theorem.

Theorem 2.3.4. *If $\gamma \neq 0$, then (β, β', γ) determines a two-generator group $\langle f, g\rangle \subset \mathcal{M}$ up to conjugacy.*

The proof of this, see [27], is found in demonstrating that an appropriate pair of generators can always be constructed from a given parameter set; in Section 4.2 we demonstrate such generators for the two-elliptic-generator case, see also [16].

By this parameterisation the space of two-generator subgroups of $\mathcal{M}$ ($\gamma \neq 0$) is linked to the complex space $\mathbb{C}^3$; the discreteness of the group $\langle f, g\rangle$ may then be considered in context of the point $(\beta, \beta', \gamma) \in \mathbb{C}^3$.

2.3.3 Projections of Discreteness

The following results demonstrate that we can project the parameter space $\mathbb{C}^3$ onto $\mathbb{C}^2$, mapping general two-generator Kleinian groups onto Kleinian groups where one generator is of order two, and that furthermore there is a symmetry in these spaces for each fixed β-value.

Theorem 2.3.5 (Lemma 2.26 in [28])**.** *If $(\beta, \beta' = \beta, \gamma = \gamma_0(\gamma_0 - \beta) \neq 0)$ are the parameters for a discrete group, then there exists a discrete group with parameters $(\beta, -4, \gamma_0)$.*

Theorem 2.3.6 (Lemma 2.29 in [28])**.** *If (β, β', γ) are the parameters of a discrete group where $\gamma \neq 0, \beta$, then there exists discrete groups with parameters $(\beta, -4, \gamma)$ and $(\beta, -4, \beta - \gamma)$.*

Note that $\beta' = -4$ implies that g is an elliptic element of order two.

The case where $\gamma = \beta \neq 0$ is dealt with below and relates to a special class of groups discussed in [43].

Theorem 2.3.7 (Lemma 2.31 in [28])**.** *If (β, β', γ) are the parameters of a discrete group $\langle f, g\rangle$ with $\gamma = \beta(f) \neq 0$, then either f is an elliptic transformation of order $2, 3, 4, 6$, or g is an elliptic transformation of order 2.*

These projections cover all possible sets of parameters, except when $\gamma \neq 0$; which, as noted above, correspond to elementary groups. Such elementary groups are discussed in detail in [63].

The above projections map Kleinian groups onto subgroups of their $\mathbb{Z}_2$ extension, its Lie product and both their conjugates. Thus inducing the symmetry seen in Figure 2.1 and other, similar figures; in fact, [23], [28] and [62] each note that in generating these figures, calculations have only been done in the first quadrant and they have then mapped them onto the other three quadrants using the above results.

In the case of $\beta = \beta'$, the projection onto a subgroup of the $\mathbb{Z}_2$ extension is geometrically equivalent to introducing an order-2 group element mapping the axis of f to the axis of g. As G is commonly an index 2 subgroup of this $\mathbb{Z}_2$ extension, the covolume of these extensions can be expected to be half that of the original group.

Later, in Chapter 5, we use these results to generate multiple additional representative generators for computational work.

2.3.4 Arithmetic Groups

Arithmetic groups, also known as arithmetic lattices, are a subclass of Kleinian groups, noted for the strong interplay between their geometric, group and number-theoretic properties. The text [41] discusses them in great detail, and gives the following definition.

Definition 2.3.8. *A finite covolume Kleinian group G is an arithmetic Kleinian group if and only if the following three conditions hold:*

1. *kG is a number field with exactly one complex-place.*
2. *trG consists of algebraic integers.*
3. *The quaternion algebra $AG \cong \left(\frac{a,b}{kG}\right)$ is ramified at all real-places of kG.*

If a Kleinian group with non-finite covolume satisfies conditions 1 - 3, then it is not an arithmetic group but is instead a proper subgroup of an arithmetic group.

For instance, the Bianchi groups are the groups $PSL(2, O_d)$, where d is a positive square-free integer. These groups represent the commensurability classes of the non-cocompact arithmetic Kleinian groups, see [41].

Arithmetic Kleinian groups can be obtained from quaternion algebras; and arithmetic properties have been determined which provide arithmetic criteria

sufficient to determine whether various two-generator subgroups of $\mathcal{M}$ are discrete, [23]. Recent work has focused on developing methods for the classification of the cocompact arithmetic Kleinian groups, most specifically in the case of groups with two-elliptic-generators.

Theorem 2.3.9 (Theorem 2.6 in [23]). *Let $G = \langle f, g \rangle$ be a Kleinian group, with f a primitive elliptic element of order $n \geq 3$, g an elliptic element of order 2, and $\gamma \neq 0, \beta$. Then G is a subgroup of an arithmetic Kleinian group if:*

- *$\mathbb{Q}(\gamma, \beta)$ has at most one complex place;*
- *γ is a root of a monic polynomial $p(z) \in \mathbb{Z}[\beta][z]$;*
- *if γ and $\overline{\gamma}$ are not real, then all other roots of $p(z)$ are real and lie in the interval $(\beta, 0)$;*
- *if γ is real, then all other roots of $p(z)$ are real and lie in the interval $(\beta, 0)$.*

If G is also of finite covolume, then G is an arithmetic Kleinian group.

In the above theorem, the requirement on the field $\mathbb{Q}(\gamma, \beta)$ to have at most one complex place can, for some cases, be described in terms of the polynomial $p(z)$. Specifically in the cases where $n = 3, 4, 6$ these criteria offer a much simpler description, see [23].

The interest in two-elliptic-generator arithmetic groups, noted in Theorem 2.3.9, is motivated by results proving that there are only finitely many arithmetic Kleinian groups generated by elliptic elements [39]. A full classification has already been determined for the non-cocompact case; and a description of 21 arithmetic generalised triangle groups, from the cocompact case, is given in [60]. The work here is related to a greater project on the classification of the two-elliptic-generator arithmetic groups.

2.4 The γ Parameter Space

Following Theorem 2.3.2 and the results of the previous section, it is common to focus on the investigation of two-generator Kleinian groups; and, more specifically, the nature and "geometry" of their parameter sets when considered as a subspace of $\mathbb{C}^3$.

The $(p, 2)$-Commutator Plane

We consider only the two-generator Kleinian groups $\langle f, g\rangle$ where f is a primitive elliptic of order p, and g is a primitive elliptic of order q. In this case

$$par(\langle f, g\rangle) = \left(-4\sin\left(\frac{\pi}{p}\right), -4\sin\left(\frac{\pi}{q}\right), \gamma\right);$$

and one can consider the alternative parameter set

$$par(\langle f, g\rangle) = (p, q, \gamma).$$

We follow the results of previous sections and consider the groups $\langle f, g\rangle$ where f is a primitive elliptic of order $p \geq 2$ and g is an elliptic of order 2. By fixing p, attention can be restricted to the γ parameter and its parameter-space $\mathbb{C}$; which is referred to as the $(p, 2)$-commutator plane. While all non-zero values of the $(p, 2)$-commutator plane may correspond to a two-generator group, only a subset of these will correspond to Kleinian groups.

Note that if $(2, 2, \gamma)$ corresponds to a discrete group, then it is either a cyclic or dihedral group; therefore the cases when $p = q = 2$ are commonly ignored.

In the case $p > 2$, given a point $\gamma \in \mathbb{C}^*$ one can then ask whether the parameter set $(p, 2, \gamma)$ corresponds to a Kleinian group? If it does, then does this group have finite covolume and is it arithmetic? And further, what can we say about the $(p, 2)$-commutator plane in general?

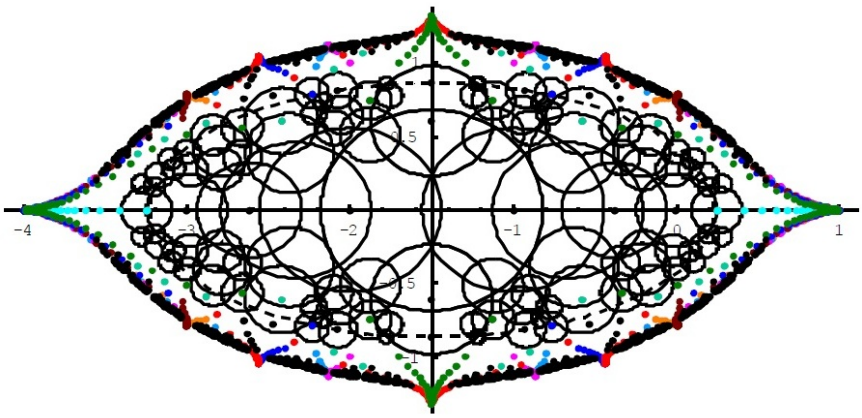


Fig. 2.1: $(3, 2)$-commutator plane with disc covering and fractal boundary. (Figure 3.16 of of [62].)

Investigating these commutator planes is an active area of research, see [62], [23] and the references therein. The remainder of this section will give a brief

overview of this research and explain the various results demonstrated on the $(3, 2)$-commutator plane shown in Figure 2.1; as seen in [62].

Comparable results and images have been shown for other $(p, 2)$-commutator planes, most specifically where $p = 3, ..., 7$. Similar images, and related results, can be found in [23] and [62].

We begin by highlighting two key features of all $(p, 2)$-commutator planes.

2.4.1 Symmetry

The reflective symmetry, seen in Figure 2.1, follows from the results of Section 2.3.3 which imply a symmetry in discreteness results, on the $(p, 2)$-commutator plane, about the lines $Im(\gamma) = 0$ and $Re(\gamma) = \frac{1}{2}\beta$.

That is, if one of the following parameter sets corresponds to a (arithmetic) Kleinian group:

- $(\beta, -4, \gamma)$,
- $(\beta, -4, \beta - \gamma)$,
- $(\beta, -4, \overline{\gamma})$,
- $(\beta, -4, \beta - \overline{\gamma})$;

then they all correspond to a (arithmetic) Kleinian group.

2.4.2 Bounds and Inequalities

By a fundamental domain construction argument based on the Klein combination theorems, see [1], [24], it is known that for each commutator plane there exists a real number $\gamma_0 \geq 0$ such that for all γ, if $|\gamma| > \gamma_0$, then $(p, 2, \gamma)$ corresponds to a free group. This γ_0-value is indicated by the circle seen in Figure 1.2.

Thus, when $|\gamma|$ is large enough, every parameter set $(p, 2, \gamma)$ corresponds to a two-generator discrete group $\langle f, g\rangle$ that is isomorphic to the free product of the two cyclic subgroups $\langle f\rangle$ and $\langle g\rangle$. In other words, $(p, 2, \gamma)$ corresponds to a group with presentation:

$$\langle f, g \mid f^p, g^2 = I\rangle.$$

In addition to this free boundary $|\gamma| > \gamma_0$, many other inequalities have been shown to exist between the various parameters. One of the earlier results in this area was Jørgensen's inequality, which provides a necessary condition for non-elementary Kleinian groups.

Theorem 2.4.1 (Jørgensen's Inequality, [33]). *If $\langle f, g\rangle$ is a Kleinian group, then its parameters satisfy the following inequality*

$$| \beta(f) | + | \gamma(f,g) | \geq 1.$$

This theorem has stimulated research in these inequalities, with many results having been determined with respect to the complex analysis of Möbius transformations; for example see [26], [27] and [57]. The inequality in Theorem 2.4.1 can easily be rearranged to have the form $|\gamma| \geq r$ for some positive real number r; but, more specific to our setting, [28] provides the following inequality.

Theorem 2.4.2. *If $\langle f, g\rangle$ is a Kleinian group, and f or g is an elliptic of order $n \geq 3$ and $\gamma \neq 0, \beta$, then*

$$| \gamma | \geq 2\cos\left(\frac{2\pi}{7}\right).$$

These inequalities, and the free boundary, can be used to begin to describe the space of two-elliptic-generator Kleinian groups.

2.4.3 Disc Covering

Recent work, demonstrated in [23] and the references therein, has seen the development of methods which utilize a variety of inequalities, like those mentioned above, and semi-group polynomials in the γ-parameter, to build a description of the γ-plane. This description comprises of an iterative series of boundaries upon the γ-values corresponding to discrete groups within a $(p, 2)$-commutator plane.

Let $G = \langle f, g\rangle$ be a discrete group where f and g are elliptic. Take

$$h = g \circ f^{m_1} \circ g^{-1} \circ f^{m_2} \circ g \circ ... \circ f^{m_n} \circ g^{(-1)^n},$$

where $m_i = \pm 1$. Then $\langle f, h\rangle$, as a subgroup of $\langle f, g\rangle$, is discrete; by Theorem 2.3.6, the group represented by $(\beta(f), -4, \gamma(f,h))$ is discrete; and, further, $\gamma(f,h)$ is given by a semigroup polynomial t in $\gamma(f,g)$ and $\beta(f)$.

Fixing p, the semigroup polynomial only has variable $\gamma(f,g)$, which gives rise to an iterative process for the derivation of successive γ-values in the $(p,2)$-commutator plane.

$$\gamma_{i+1} = t(\gamma_i).$$

This iterative process is described in [28], and cannot have 0 as a limit. Further it cannot iterate into the regions determined by inequalities, such as those given

in Section 2.4.2 above, to not hold discrete groups; with the exception of certain 'exceptional' values.

$$\begin{aligned} \langle f, g\rangle \text{ discrete} &\Rightarrow \langle f, h\rangle \text{ discrete} \\ \langle f, h\rangle \text{ not discrete} &\Rightarrow \langle f, g\rangle \text{ not discrete} \end{aligned}$$

If R is one of the regions where γ does not correspond to a discrete group, then neither does the set $t^{-1}(R)$. This provides a disc covering method whereby the only valid Kleinian group values of γ can be found in $\mathbb{C}\backslash\{\cup t^{-1}(R) : t \in \tau\}$, where τ is the set of such semigroup polynomials; with the exception of certain, hard to determine, exceptional points.

The results of this disc covering method for the $(3, 2)$-commutator plane, as undertaken in [23], are shown in Figure 2.1; specifically the discs and the points contained within them.

Determining the regions R, and their exceptional points, can be difficult; but [23] notes that there exist polynomials in the arithmetic data for which valid, discrete γ-values are a root. Using this, one can determine arithmetic conditions which guarantee the discreteness of a group. The arithmetic theorems recently developed then reduce the search for γ-values down to a test on polynomials.

These methods have been used to eliminate large regions of possible γ-values in a number of $(p, 2)$-commutator planes. See [23] and its references, specifically [28], for more details and images of other disc coverings.

2.4.4 Fractal Boundary

Using SnapPea (see Section 3.1) for the experimental construction of γ-words, and eliminating the determinably free elements, progress has recently been made in demonstrating a conjecturally fractal boundary beyond which all γ-values correspond to free groups, [62].

This has been done through a consideration of $(m, 0)$ Dehn surgery upon hyperbolic 2-bridge links. These links correspond to hyperbolic orbifolds groups generated by two order-m transformations. These groups are not free, so are contained within the $|\gamma_0|$ circle, but are generally outside the disc covering given above. This work has given rise to a "computational boundary", beyond which there are only free groups. In the $(p, 2)$-commutator planes this is conjectured to be a fractal Jordan curve, and is observationally comparable to the Riley slice.

The boundary is shown, for the $(3, 2)$ case, in Figure 2.1, and is given by the

individual points surrounding the disc covering.

This work has greatly improved upon the previously mentioned $|\gamma_0|$ bounds, and gives an indication towards the overall completeness of these disc covering methods.

2.5 Fundamental Polyhedrons

We return to focusing on the topological and geometric structures used to represent a group, and note that for the remainder of this chapter there will be the tacit assumption that X is a model of hyperbolic 3-space and G is a discrete subgroup of $Isom^+(X)$. However, it is worth noting that this discussion generalises easily to an n-dimensional case; and, as such, it is often given this way, see [20], [44] and [55]; similarly G can be generalised to include orientation-reversing isometries and other non-hyperbolic geometries can be used.

Further, as it can be hard to accurately represent this information in a visual manner, pictures provided in this section will generally be given as 2-dimensional cross-sections. Given this, pictures should be considered as general guides only. For a comparative reference, a full description of the 2-dimensional case can be found in [10].

2.5.1 Polyhedrons

Let the set P be comprised of a non-empty intersection of countably many, non-disjoint, open half-spaces in X, such that only finitely many of the hyper-planes, defining these half-spaces, meet any compact subset of X. Then P is an open convex polyhedron in X. Figure 2.2 shows several polyhedrons in the projective disc model of hyperbolic space.

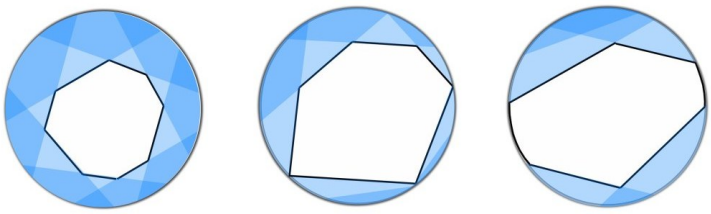


Fig. 2.2: Polyhedrons.

The closure of P is denoted $\overline{P}$, and we note that it has a natural cell decomposition which is given by the intersections of the above half-planes.

- The single dimension three cell is $\overline{P}$ itself.

- The dimension two cells are called the sides[6] of P, denoted s_i;
- The dimension one cells are referred to as the edges of P, denoted e_i; and
- The dimension zero cells are the vertices of P, denoted v_i.

Note that the sides are closed, convex subsets of the defining hyper-planes; edges the closed, geodesic line segments lying in the intersection of exactly two sides; and vertices the finite end-points of these edges, where 3 or more sides meet. The boundary of P in X, ∂P, is a union of the countably many sides of P.

2.5.2 Side-Pairings and Identification

An identification on a polyhedron P is a map assigning each side s with a side s', and an isometry g_s, such that:

- g_s maps s' onto s;
- $(s')' = s$ and $g_{s'} = (g_s)^{-1}$; and
- for each s, there is a neighbourhood U of s, such that

$$g_s(U \cap P) \cap P = \emptyset.$$

We refer to a polyhedron with identification as an identified polyhedron and we call the transformations g_s the side-pairing transformations of P, denoting the group they generate by G_S. Further we consider the notations $g_{s'}$ and g'_s to be interchangeable.

Notice that if $s = s'$, then the above conditions imply that g_s is of order two.

Definition 2.5.1 (reflection relations)**.** *Relations of the form $g_s^2 = 1$ are called the reflection relations of an identified polyhedron P.*

Let P^* be the identified polyhedron obtained by identifying the sides of P. Then we have the usual quotient map

$$\pi : \overline{P} \mapsto P^*$$

Where, for $x, y \in \overline{P}$, $\pi(x) = \pi(y)$ if there is a side-pairing transformation g such that $g(x) = y$. We refer to sets and points $A, B \subset \overline{P}$ as identified whenever $\pi(A) = \pi(B)$.

[6] In this, three-dimensional, case it is common to refer to the side of P as faces; we avoid this here for simple notational benefits.

Definition 2.5.2 (metric condition)**.** *Let P be a polyhedron. If $\pi^{-1}(x)$ is a finite set for each $x \in P^*$, then P is said to satisfy the metric condition.*

If P satisfies the metric condition, then the standard quotient metric ρ^* is a metric on P^*. The core property of the metric ρ^* is that

$$\rho^*(\pi(x), \pi(y)) \leq \rho(x, y),$$

for all x and y in $\overline{P}$. This ensures that if $\pi : \overline{P} \mapsto P^*$ is continuous, then P^* is complete if and only if it is complete in the metric $\rho*$.

Fundamental Polyhedron

Definition 2.5.3 (Fundamental Polyhedron)**.** *Let G be discrete, and let P be a polyhedron in X. Then P is a fundamental polyhedron for G if and only if:*

- *For every non-trivial $g \in G$, $g(P) \cap P = \emptyset$.*
- *For every $x \in X$, there is a $g \in G$ such that $g(x) \in \overline{P}$.*
- *P has an identification.*
- *P is locally finite.*

Clearly, if a polyhedron is a fundamental domain, then it is a fundamental polyhedron. It is not uncommon for the concept of polyhedrons to be generalised so that non-polyhedral domains might also be considered to have sides and an identification.

2.5.3 The Dirichlet Domain

Given a discrete group G there is a standard method of construction used to determine a fundamental polyhedron; this construction is called a Dirichlet domain.

Definition 2.5.4 (Dirichlet domain)**.** *Let ρ be a metric on X and let G be discrete. Then*

1. *Choose a point 0 in X, such that the stabilizer of 0, G_0, contains only the identity element, I.*
2. *For any element $g \in G \backslash \{I\}$, define the half-space $H_g(0)$ of points closer to 0 than to $g(0)$,*

$$H_g(0) = \{x \in X \mid \rho(x, 0) < \rho(x, g(0))\}.$$

3. Let $D(G,0)$ be the intersection of all such half-spaces,

$$D(G,0) = \cap_{g\in G, g\neq I} H_g(0).$$

Then $D(G,0)$ is the Dirichlet domain for G with basepoint 0.

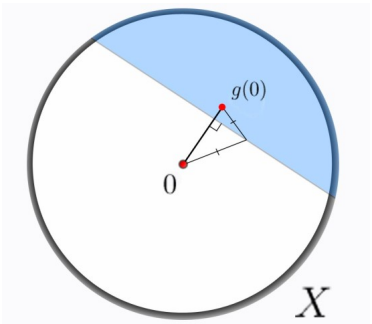


Fig. 2.3: The construction of a Dirchlet domain.

In this way $D(G,0)$ is comprised of the points of X closer to 0 than to any $g(0)$, $g \in G\backslash\{I\}$. It is clear that this construction process satisfies the intersection definition for polyhedrons given above, thus $D(G,0)$ is a polyhedron in X.

The elements of G also provide an identification on the boundary of $D(G,0)$, with the transformation g which "generates" a side s in $H_g(0)$, also being a side-pairing transformation, mapping s' onto s. Clearly $D(G,0)\cap g(D(G,0)) = \emptyset$ for all $g \in G\backslash\{I\}$, the closure of $D(G,0)$ contains a fundamental set for G and, as any compact set can only meet finitely many of its G images, $D(G,0)$ is locally finite; thus we have the following result.

Theorem 2.5.5. *The Dirichlet domain $D(G,0)$ is a fundamental polyhedron for the discrete group G.*

This Dirichlet process results in the construction of a fundamental domain for any discrete group G; and it is worth noting that any group G for which we can construct a fundamental domain is discrete.

2.6 The Geometry of Fundamental Polyhedrons

Any subsequent consideration of fundamental domains will be solely with respect to fundamental polyhedrons and Dirichlet domains. As such, we cover some of the additional properties of these constructions, with the specific intent of setting up terminology for discussing the Poincaré theorem, given below, and

the construction processes discussed in Chapter 3. See [31], [44] and our other references for further details on these constructions.

A fundamental polyhedron P is exact if and only if each side of P is equal to the intersection $\overline{P} \cap g(\overline{P})$ for some side-pairing transformation g. The Dirichlet domain of a discrete group is an exact fundamental polyhedron.

In a procedural sense, if $H_g(0)$, $g \in G$, determines a new side during the construction of a Dirichlet domain, then we call it a side-generating transformation. All side-pairing transformations are side-generating, but the set of side-pairing transformations may not contain all the side-generating transformations of the $D(G, 0)$ construction process.

For every point $x \in s$, $\rho(0, x) = \rho(g_s(0), x)$, and so $\rho(0, x) = \rho(g'_s(0), g'_s(x)) = \rho(0, g'_s(x))$. The element g_s, which generates a side s, also maps its side pair s' onto s, $g_s(s') = s$; from this it follows that if y is any point in $\overline{P}$, then $\rho(a, 0) = \rho(b, 0)$ for all $a, b \in \pi^{-1}(y)$.

Let x be any point in X, then we can draw a path from some point y in P to x, such that the path does not pass through the translates of any vertices or edges of P. There are a series of conjugates of side-generating transformations that will map any translate of P along this path onto the next translate of P. As these conjugating elements can be made as products of side-generating transformations, it is clear that P tessellates X. Thus we have $G_S = G$.

Theorem 2.6.1. *Let P be a fundamental polyhedron for the group G. Then the side-pairing transformations of P generate the group G.*

2.6.1 Cycle Transformations

Let P be a complete fundamental polyhedron, we consider the following procedure based on its identification:

1. Select an arbitrary edge e_1;
2. there are two sides meeting at e_1, label one s_1;
3. determine side s'_1 and the side-pairing transformation $g_1(s_1) = s'_1$;
4. let $e_2 = g_1(e_1)$;
5. continue to determine s_i, s'_i and g_i, for e_i $i \geq 2$

This gives rise to a sequence of edges $\{e_i\}$, side pairs $\{(s_i, s'_i)\}$, and of side-pairing transformations $\{g_i\}$. Note that $g_i = g'_{s_i} = g_{s'_i}$.
Define the period m as the least positive integer such that all three of the above sequences are periodic with period m. The sequence of edges $\{e_1, ..., e_m\}$ is called an edge cycle, which we will simply denote $\{e_1\}$.

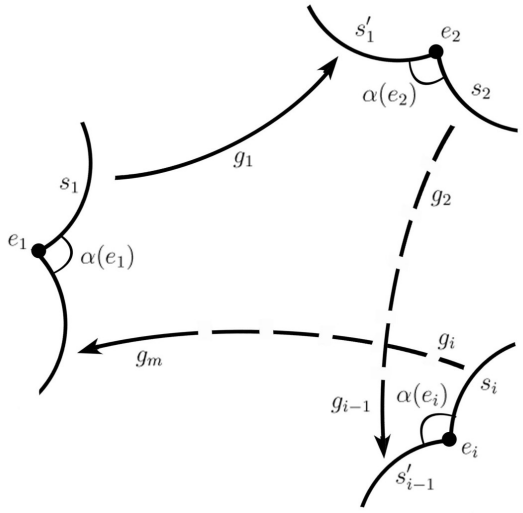


Fig. 2.4: An edge cycle.

Definition 2.6.2 (cycle transformation)**.** *The isometry $B_{e_1} = g_m \circ ... \circ g_1$ is called the cycle transformation at e_1.*

Note that $B_{e_1}(e_1) = e_1$, and if $e_i \in \{e_1\}$, then the cycle transformation at e_i is conjugate to the cycle transformation at e_1. In this way we need only consider conjugacy classes of edge cycles, each class corresponding to a distinct set of identified edges.

For each edge e_i in the cycle $\{e_1\}$, let $\alpha(e_i)$ be the (hyperbolic) dihedral angle made by the sides s'_{i-1} and s_i, measured from inside P.

Definition 2.6.3 (Cycle condition)**.** *A polyhedron P is said to satisfy the cycle condition if and only if for each edge cycle $\{e_1\}$, there is a positive integer v such that*

$$v \sum_{i=1}^{m} \alpha(e_i) = 2\pi.$$

If a polyhedron satisfies the Cycle condition, then each edge cycle has the relation $B^v = I$.

Definition 2.6.4 (Cycle relations)**.** *The relations of the form $B^v = I$ are called the cycle relations of the fundamental polyhedron P.*

2.6.2 Ideal Sides, Edges and Vertices

Consider a polyhedron $P \subset X$, with closure $\overline{P}$, as defined above.

Let P' be the closure of P with respect to $\overline{X} = X \cup \partial X$. Then $\overline{P} \subset P'$ and $\partial\overline{P} = \partial P \subset \partial P'$.

Here we are concerned with those points contained in the intersection $\partial P' \cap \partial X$, which we refer to as the ideal points of P. These points are highlighted red in Figure 2.5; however, note that these images are cross-sections of X so only clearly demonstrate the 2-dimensional case.

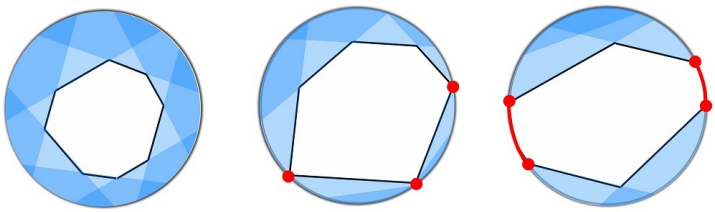


Fig. 2.5: The ideal components of a polyhedron.

It is common to extend the definition of sides, vertices and edges to include these ideal parts of the polyhedron P. We then refer to all non-ideal points in P', or P, as being finite.

An ideal side is a maximal 2-dimensional component of $\partial P' \cap \partial X$. As the definition of a polyhedron implies convexity, no two ideal sides can meet. An ideal side and a finite side are either: disjoint; meet at a curve on ∂X (ideal edge); or meet at a point on ∂X (ideal vertex). Note that it is possible for an ideal and finite side to meet at more than one distinct ideal vertex while sharing no ideal edges.

As above, an ideal edge is a maximal, 1-dimensional component in the intersection of an ideal side and a finite side of P. Due to the nature of hyper-planes in X, no two distinct finite faces can share an ideal edge.

The ideal vertices are then the endpoints in ∂X of any (finite or ideal) edges of P. An ideal vertex is referred to as proper if it is the endpoint of finite edges only, otherwise it is improper. There can be no similar categorisation of edges and faces.

If two finite sides of P meet at, and only at, a single ideal vertex v, then v is known as an infinite-edge. Cycles can be constructed for infinite-edges as they are for finite edges. This construction process either ends at an ideal edge, making it a chain; or completes a cycle. Any such cycle is referred to as an infinite-edge cycle.

Theorem 2.6.5 (Cusp Condition)**.** *A fundamental polyhedron is complete if all its infinite-edge cycle transformations are parabolic.*

We note that ideal sides have no side-pairings, as any side-pairing transformation g would infer a contradiction $P \cap g(P) \neq \emptyset$ or $g(P) \nsubseteq X$. However, ideal edges and ideal vertices can be considered to be identified, like their finite counterparts, under a continuous extension of the quotient map π.

2.7 Poincaré's Theorem

We end this chapter with a brief discussion on Poincaré's (fundamental) theorem for fundamental polyhedrons.

Definition 2.7.1. *Let P be a complete polyhedron with an identification satisfying the cycle condition. Then P is called a Poincaré polyhedron.*

Theorem 2.7.2 (Poincaré's Theorem)**.** *Let P be a Poincaré polyhedron, then:*

- *G_S is discrete;*
- *P is a fundamental polyhedron for G_S; and*
- *the cycle relations and the reflection relations (of P) form a complete set of relations for G_S.*

This is a subtle theorem that stands out as one of the few simple, sufficient criterions for discreteness; further, it provides a direct link between the fundamental domain of a group and its group presentation. As such, this theorem serves as a cornerstone to many areas of research and is outlined in many texts.

The proof is very rarely given explicitly for the three-dimensional case - we have not been able to find one. Maskit gives a detailed overview of the proof in 2-dimensions, before indicating how to extend it to 3-dimensions in [45]; here he also first demonstrated that a commonly held convexity condition was not required. In fact, while convexity is a part of our original descriptions of polyhedrons and the Dirichlet construction, it isn't necessary for any of the descriptions that followed.

The proof of Poincaré's theorem is now often given in a general n-dimensional form, as in [44], we note that [20] provides the most detailed and complete exposition available on the theorem; demonstrating it in it's most general form, as well as providing a review of the current literature on the topic.

The constructions, given in these Sections 2.5-2.7, provide a method from which quick and accurate computation of fundamental domains can be based, which is discussed further in Chapters 3 and 5; and in the canonical reference [56]

We note again that the Dirichlet domain is a function of both the group G and the basepoint 0. Shifting the basepoint will give a distinct construction, geometrically comparable to the Dirichlet domain for a group conjugate to G.

Additionally Poincaré polyhedrons are not restricted to polyhedrons with finite-covolume; and thus Poincaré's theorem for fundamental polyhedrons holds for domains with ideal vertices, edges and sides.

3. COMPUTING FUNDAMENTAL DOMAINS

In this chapter we discuss a computational framework for the construction of fundamental polyhedrons. The method outlined here is taken from the kernel of the computer program SnapPea [59], and is an implementation of the theory given in Chapter 2.

Later, in Chapter 4, we discuss our intended application and adaption of the processes given here, before using them to construct domains for discrete groups generated by two finite-order elliptic transformations in Chapter 5.

3.1 SnapPea

SnapPea is a computer program designed for the study of hyperbolic 3-manifold and knot theory. The primary developer is J. Weeks, who created the first version as part of his doctoral thesis. When we refer to this program we are referring to the modern kernel of the program - that is, the current computational core of the program, not the user interface or any other satellite parts of the program. SnapPea can be downloaded via [59].

The overall focus of this program is on the triangulation of manifolds (orbifolds), and from them the purposeful derivation of information relevant to the classification of groups and manifolds. It performs several main computations: taking a knot, or representative file, as an input; constructing a triangulation of the knot complement, describing the manifold as a gluing of tetrahedra; and determining the fundamental group and other manifold properties. From there it can be used to perform Dehn surgeries on the knot complement or use the fundamental group to construct a fundamental domain in hyperbolic 3-space, representative of the manifold or orbifold.

In this way the main focus of SnapPea revolves around computations on, and manipulations of, a triangulation data structure.

SnapPea is often considered to be comprised of two main parts: the routines relating to the determination of an ideal triangulation for a hyperbolic knot complement; and those relating to the computation of the canonical decomposition of this complement. However, for our purposes we consider the SnapPea

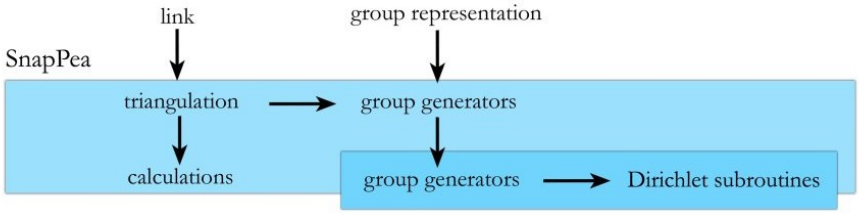


Fig. 3.1: Flow chart of the SnapPea program.

kernel to be divided into two different sections, those parts dealing with this triangulation and decomposition of a knot complement, and those dealing with the construction of a fundamental domain; the later of which we refer to collectively as `Dirichlet_` subroutines. These subroutines, which we detail below, are concerned with the construction of Dirichlet domains as we have described in Section 2.5.3.

3.1.1 The `Dirichlet_` subroutines

As stated, our interest is in SnapPea's computational processes that focus on the construction of fundamental domains in hyperbolic space.

These processes are a centralized group of programs written across several files, each possessing a different component of the computational process for the construction of a fundamental polyhedron. We note these files below.

- `Dirichlet.c` provides overall structure to the general construction processes; giving global organization, formatting input generators for the main construction procedure, accessing the routines for finalising the domains, and providing access to the core Dirichlet domain construction routines.
- `Dirichlet_basepoint.c` provides the routines for moving the basepoint of the Dirichlet domain in an attempt to maximise its injectivity radius.
- `Dirichlet_construction.c` contains the two main construction routines, the functions for cutting the domain representation, and the preliminary finalisation routines for confirming the resultant construction as a fundamental polyhedron.
- `Dirichlet_extras.c` provides the functions for adding all the "bells and whistles" to the Dirichlet domain once it has been computed. These determine the edge classes and calculate the hyperbolic volume of the fundamental polyhedron, Euler characteristic and other constants; and, in doing so, confirm the fundamental polyhedron.

- `Dirichlet_precision.c` provides some access to the standard error reducing functions of SnapPea for use in these subroutines.

The actual construction procedure is detailed below in Section 3.3.

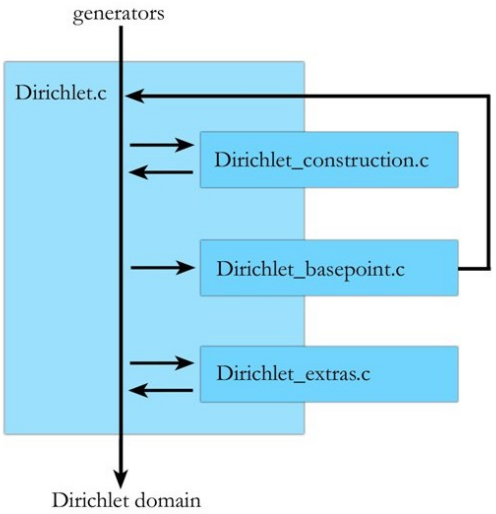


Fig. 3.2: Interaction of the `Dirichlet_` subroutines.

There are two additional `Dirichlet_` files that, while a part of the general `Dirichlet_` subroutines (by name), are not among those under our consideration, these are: `Dirichlet_rotate.c`, which helps the SnapPea user interface to keep track of the visual representation of a domain; and `Dirichlet_conversion.c`, which converts the Dirichlet domain back into a triangulation for use inline with the majority of SnapPea's functions.

In addition to the files given above, the `Dirichlet_` subroutines run off the routines and data structures provided by a number of other files; such as the matrix algebra and kernel files of SnapPea; and the file `winged_edge.h`, which provides the overall data structure for representing the polyhedron data. For our purposes we consider these additional files to be part of the greater, overall `Dirichlet_` subroutines.

Note that all the files mentioned above are freely available for download as part of the SnapPea kernel [59], and have been provided in PDF form in the SnappyD support files.

3.1.2 SnapPea as a Template

The code of the SnapPea kernel is freely available for use as a template for building programs upon. The most notable example of a program based on SnapPea

is probably the program Snap, [17]. A program providing an updated, object orientated, code that integrates the computational methods of SnapPea with the highly accurate computational techniques of the number theory program Pari, [53]. Snap maintains a similar triangulation focus to SnapPea, and it is worth noting that its developers encouraged the depth of development seen in parts of the modern SnapPea program.

Given the similar ideas, but greater accuracy of Snap, it is worth addressing the idea that using it as a basis over SnapPea would seem the wiser choice. Indeed, brief communication with one of the developers did indicate that Snap should also have infinite precision Dirichlet domain construction routines. However, the code has scarce documentation, especially when compared to the incredibly well documented SnapPea, while still maintaining a triangulation specific focus, so we felt working from SnapPea would be the wiser initial choice. As we discuss in Chapter 6, increasing the precision of these routines is an option for a later discussion.

Our choice of SnapPea was also encouraged by its noted successes, its construction procedures are even used in Snap as a double check. Additionally, the program has been used in other related research, see [62], and a goal at the onset of this project was determining why certain attempts to construct Dirichlet domains, using a program based on the `Dirichlet_` subroutines, were failing. This last issue is dealt with in Chapter 5.

3.1.3 Groups of Interest

It is our intention to modify the `Dirichlet_` subroutines to prevent the failure of the program in a variety of situations; we discuss these changes and issues in Chapter 4. Specifically, we will expand the routines to construct domains for both finite and non-finite constructions, directly from a modified parameter set (p, q, γ); while providing information on the construction process and structure of the fundamental polyhedron, providing a core upon which future investigations and adaptation may be based.

The triangulation process of SnapPea ensures that any group for which a domain construction attempt is made, will be both (approximately) discrete and have finite covolume. We are interested in the arithmetic, disc-covering and "free boundary" related groups as discussed in Section 2.4. In the case of arithmetic data we know that the group is discrete; but for other points, especially the exclusion points of the disc-covering, this is not necessarily the case. For investigative purposes, we require the program to be able to attempt domain constructions for both sets of groups.

3.2 Computing Fundamental Polyhedrons

In applying a computational theory to the construction of a fundamental polyhedron, there are several important issues that require consideration.

At its most basic level, as outlined in Section 2.5.3, the procedure for constructing a fundamental polyhedron follows three basic steps:

1. Start with
   - X, a model of hyperbolic 3-space;
   - G, a discrete group of isometries acting on X; and
   - 0, a suitable basepoint in X.
2. Determine $g(0)$ for a non-identity element g of G and remove ("cut off") all points in X closer to $g(0)$ than they are to 0.
3. Repeat step (2) for all non-identity elements of G.

However, based on this naive interpretation of the process, several considerations arise in the computation of a Dirichlet domain, namely: the accurate and easy representation of a polyhedron in hyperbolic 3-space; the representation, or storing, of group information in both an accessible and usable way; and an accurate cutting method with which to generate the polyhedron.

Additionally, in any case that is likely to be of interest, this is a countably infinite construction process. While in practice a final domain may often be found quickly, generally there is no guarantee that, after $n \in \mathbb{N}$ cutting steps, the Dirichlet domain for a group will be realised. Further, in a system without infinite precision, there is a likely chance that after a sufficiently large number of cutting steps, or generations of group elements, the accumulation of error will exceed the bounds of acceptable accuracy. Thus this construction process must be tempered with the realisation of possible failure.

In this section we describe the implementation of the background mathematical constructs used by the `Dirichlet_` subroutines, and note the practical limitations that have been kept in mind. The realisation of these limitations against the computational goals of this project are discussed in Chapter 5.

We note that for the remainder of this work, when we refer to polyhedrons, P_i, and half-spaces, H_g, we are generally referring to their respective closures; as distinct from the open sets defined in Sections 2.5.1 and 2.5.3. This is due to the computational representation of spaces being best based on closed sets. Within the context of this ongoing discussion, this should not cause any confusion.

3.2.1 Implementation of Hyperbolic Space

For the general purposes of describing a polyhedron, the projective disc model, $\mathbb{D}^3$ (see Appendix A), lends itself to being the obvious choice of background model as it avoids many of the extraneous properties inherent of hyperbolic space, specifically:

- the space is finitely bounded in $\mathbb{R}^3$, with $\mathbb{D}^3$ being identified as the unit 3-ball; and
- the geodesics of $\mathbb{D}^3$ are Euclidean straight-line segments in $\mathbb{R}^3$.

Additionally, it is noted in [37] that the isometry and transformation calculations in $\mathbb{D}^3$ do not take on the rational form they do in other models.

This allows (finite) Dirichlet domains to be described by the conventional linear data structures that are often used to represent Euclidean polyhedrons. Under this model, SnapPea uses a winged edge data structure to represent a polyhedron in $\mathbb{R}^3$, see [8]. This mesh construct is noted for a description which focuses on a description of the edge data, which aligns well with the edge properties outlined in Section 2.6.

Note that while the choice of $\mathbb{D}^3$, as a model of hyperbolic 3-space, gives access to the common, real linear algebra, description of a polyhedron and the real linear algebra inherent of the representative isometry group, $PSO(1,3)$, it is still necessary to accurately consider the polyhedral domain P as a hyperbolic structure; especially with relation to the action of hyperbolic isometries on the basepoint and alterations to the polyhedron.

However, with this linear representation, an issue arises with the Dirichlet domain construction process being required to start from an arbitrarily large domain. We take the basepoint $0 = \{0,0,0\} \subset \mathbb{R}^3$, and require an initial polyhedron P_0 from which the construction can begin. A group element, g, will then induce a cutting action represented by $P_0 \cap \{x : \rho(x,0) \leq \rho(x,g(0))\}$.

If we maintain the Euclidean representation of a polyhedron, then any attempt to use a regular n-gon (with arbitrarily large n) to approximate the boundary $\partial\mathbb{D}^3 \subset \mathbb{R}^3$, in a meaningful way, would be data intensive and likely still not general enough. Whereas using a non-polyhedral structure for P_0 introduces an undesirable level of complexity to the description of the initial construction. Given this, the `Dirichlet_` subroutines instead begin with a domain that exists beyond $\mathbb{D}^3$; to discuss this situation we define a new class of points.

Definition 3.2.1 (Classification of Points)**.** *Consider a model of hyperbolic* 3*-space X, embedded in a parent space Y ($\hat{\mathbb{R}}^{1,3}$ or $\hat{\mathbb{R}}^3$). Let y be any element of Y, then:*

- *y is finite if and only if it is contained in X;*
- *y is ideal if and only if it is contained in ∂X, the boundary of X with respect to Y; and*
- *y is hyperideal if and only if it is neither finite nor ideal.*

This definition extends the description of finite and ideal points given in Section 2.6.2. Considering only the projective disc model of hyperbolic 3-space, we can restate the definition in more measurable terms.

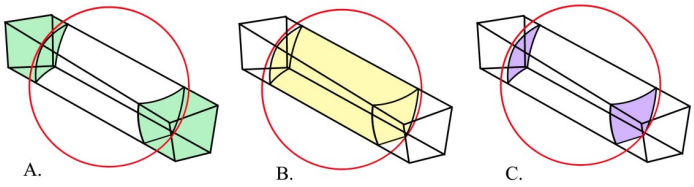


Fig. 3.3: The components of a polyhedron.
(A) hyperideal components; (B) finite component; and (C) ideal components.

Definition 3.2.2 (Classification of Points in $\mathbb{R}^3$)**.** *Let y be any point in* $\mathbb{R}^3$*, then:*

- *y is finite if and only if* $\|y\|_E^2 < 1$*;*
- *y is ideal if and only if* $\|y\|_E^2 = 1$*;*
- *y is hyperideal if and only if* $\|y\|_E^2 > 1$

In this computational setting, SnapPea cannot be quite so exact; the `Dirichlet_` subroutines consider a point y to be:

- finite if and only if $\|y\|_E^2 < 1 - 4 \times 10^{-7}$; and
- hyperideal if and only if $\|y\|_E^2 > 1 + 10^{-3}$.

Leaving a band of width 1.0004×10^{-3} within which a point is considered to be ideal.

We refer to any polyhedron in $\mathbb{R}^3$ with an interior that includes hyperideal parts as a hyperideal polyhedron, and any other non-empty polyhedron as a finite polyhedron. Note that a finite polyhedron is not prohibited from having ideal vertices or infinite (hyperbolic) volume; but it cannot have ideal edges or faces, as the linear representation of polyhedrons would then imply that there must be hyperideal parts to the polyhedron.

The Dirichlet domain construction then begins from a simple hyperideal polyhedron, a Euclidean cube set in $\mathbb{R}^3$:

$$P_0 = \{(x_1, x_2, x_3) \in \mathbb{R}^3 : \|x_1\|_E, \|x_2\|_E, \|x_3\|_E \leq 2\}.$$

Thus ensuring that any Dirichlet domain will be contained within this initial polyhedron.

However, there is a catch, by using this framework of linear representation, and hyperideal starting point, we restrict our possible solutions of accurately represented fundamental polyhedrons to those with only finite or proper ideal vertices. That is, as indicated above, these procedures cannot be used to accurately represent a domain with ideal faces.

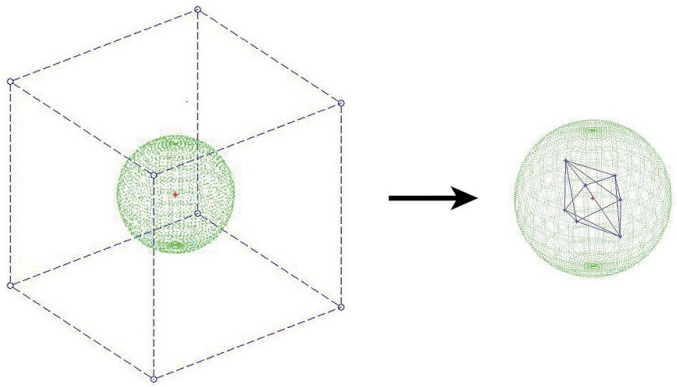


Fig. 3.4: The initial cube, P_0, and a resultant finite polyhedron P.

Within SnapPea, this hyperideal starting point and subsequent domain restriction is not an issue, as the generating groups arise from hyperbolic knot complements and so only domains whose polyhedrons are fully within the interior of $\mathbb{D}^3$, with the exception of a possibly countable number of ideal vertices, are considered valid constructions. In other words, in their application within SnapPea, there is an assumption that there exist elements g in G that will remove all hyperideal parts of the initial domain (this is something we are unable to assume in our work).

Note that in all of the above, the `Dirichlet_` subroutines actually consider $\mathbb{R}^3$ as the $x_1 = 1$ hyper-plane of the Minkowski space $\mathbb{R}^{1,3}$, so as to allow easy use of the $PSO(1,3)$ representation of isometry transformations. The individual group elements being contained in data structures to match the $PSO(1,3)$ data group, and generally being stored alongside their inverses so that if a face is generated, then so too is its identified mate.

3.2.2 Group Generation and Representation

As mentioned above, the $\mathbb{D}^3$ model of hyperbolic space uses the $PSO(1,3)$ matrix group to represent $Isom^+(X)$, see Appendix A. This is a group of 4×4 real matrices whose actions, when applied in vector form and restricted appropriately, act as hyperbolic isometries on the unit 3-ball. This real linear algebra fits with the Euclidean polyhedral representation and allows for more straight-forward calculations than might otherwise be possible.

However, the groups G commonly used to generate the Dirichlet domains are not finite, which can make working with them as a whole computationally intensive. In an attempt to avoid this issue, the `Dirichlet_` subroutines use a sequential method of generating group elements, making use of the fact that G is finitely generated. Let G be generated by generators g_j and take:

- G_0 to be the set of generators $\{g_j\}$;
- P_0 to be the initial cube structure, as defined above;
- for $i > 0$, G_i to be the set of side-generating elements of P_i together with their first-order products; and let

$$P_i = P_{i-1} \cap_{g \in G_{i-1}} H_g(0),$$

where

$$H_g(0) = \{x \mid \rho(x,0) \leq \rho(x, g(0))\}.$$

Then, in this way G_i and P_i can be constructed sequentially and, for all $i \geq 0$, we have

$$G_0 \to G_{i-1} \to G_i \subseteq G$$

$$P \subseteq P_i \subseteq P_{i-1} \subseteq P_0$$

Where $P = \overline{D(G,0)}$.

By working off the side-generating transformations, the `Dirichlet_` subroutines ensure that the group data remains manageable, and lets "needless" transformations be ignored.

However, as we note in Section 3.3.1, if $P_i = P_{i+1}$, or equivalently $G_i = G_{i+1}$, then this group construction method stops progressing. If construction stops, then as the generators, g_j, should appear in P_i as side-generating transformations and a superset of the intended domain has been constructed, geometric and topological data can be used to determine any further cuts that are necessary to realise the final polyhedron P.

Overall, this provides an efficient and effective method of group and domain generation; and in the case of groups that SnapPea deals with, quickly leads to a finite polyhedron containing the desired fundamental domain, if one is achievable.

Note that rather than waiting for the end of this sequential process, once a finite polyhedron has been realised the `Dirichlet_` subroutines move onto a more geometric construction method, making use of the side-pairing transformations and their nature within the polyhedron's identification. This is detailed further in Section 3.3.1.

3.2.3 Error

By working in $\mathbb{D}^3$, with matrices in $PSO(3,1)$, as opposed to the more commonly used $PSL(2,\mathbb{C})$, the `Dirichlet_` subroutines are able to use real linear algebra in both polyhedral and group representation. However, this comes with a risk of possibly permitting a faster accumulation of error. SnapPea handles error by a simple method of fixed error resolution; and number recognition through "safe" functions at certain steps.

The routines keep firm limits on this error by way of an error catch system that constantly checks for topological errors; ensuring that the constructions either remain valid to within specific bounds of accuracy or are discarded. This allows the program to work at high speed, and attempt constructions that may fail, regardless of other factors.

Overall, this provides a straight-forward method of attempting an infinite (though practically finite) process, without attempting to force the system beyond topological irregularities; focusing on an end result that makes use of the reciprocal nature of Dirichlet domains - if we achieve a successful construction $D(G,0)$, then the construction must be valid as its identification will in turn generate G.

Resolution

There are many individual error (or resolution) bounds within the program, such as the ideal and hyperideal approximations mentioned in Section 3.2.1. For reference purposes, we provide a brief list and description of the main error bounds in Section 3.4.2.

Separate from these is an alterable resolution setting: vertex epsilon, ε_V; this setting is used to determine the accuracy of vertex positions. Specifically, for

any vertices $u, v \in \mathbb{R}^3$, for construction purposes we have

$$v = u \Leftrightarrow \|v - u\|_E < \varepsilon_V.$$

An important feature to note here is that $\|.\|_E$ is a Euclidean measure, so does not take into account the hyperbolic length of potential edges. In this way, depending on the size of the individual faces that make up our desired domain, and the particular form of the group G, this setting can potentially have quite an impact on the final output in a successful construction process, depending on edge sizes and specific input generators.

The files in SnapPea note that constructions can generally be considered likely to fail if ε_V is not set within $(10^{-16}, 10^{-2})$, due to either being too imprecise or attempting to be precise beyond the accuracy of the general computer arithmetic. In Appendix C we test all our constructions over the full range of these ε_V settings. We also note that in later sections we refer to ε_V by the value k, where $\varepsilon_V = 10^{-k}$.

3.3 The Dirichlet Process

Here we detail the overall construction process, as undertaken by the `Dirichlet_` subroutines, using the space and structures as described above. We note that the general approach here is an attempt to find a fundamental polyhedron for a given group (from generating isometries), not the direct computation of a Dirichlet domain as described in Definition 2.5.4, of Section 2.5.3.

For additional reference, and to assist this discussion, we provide Figures 3.5 and 3.6; as well as pseudo-code overviews of the three main construction routines, see Section 3.4.1. Recall that the full code for these routines can be found in the downloadable SnapPea [59], or in the support files of SnappyD.

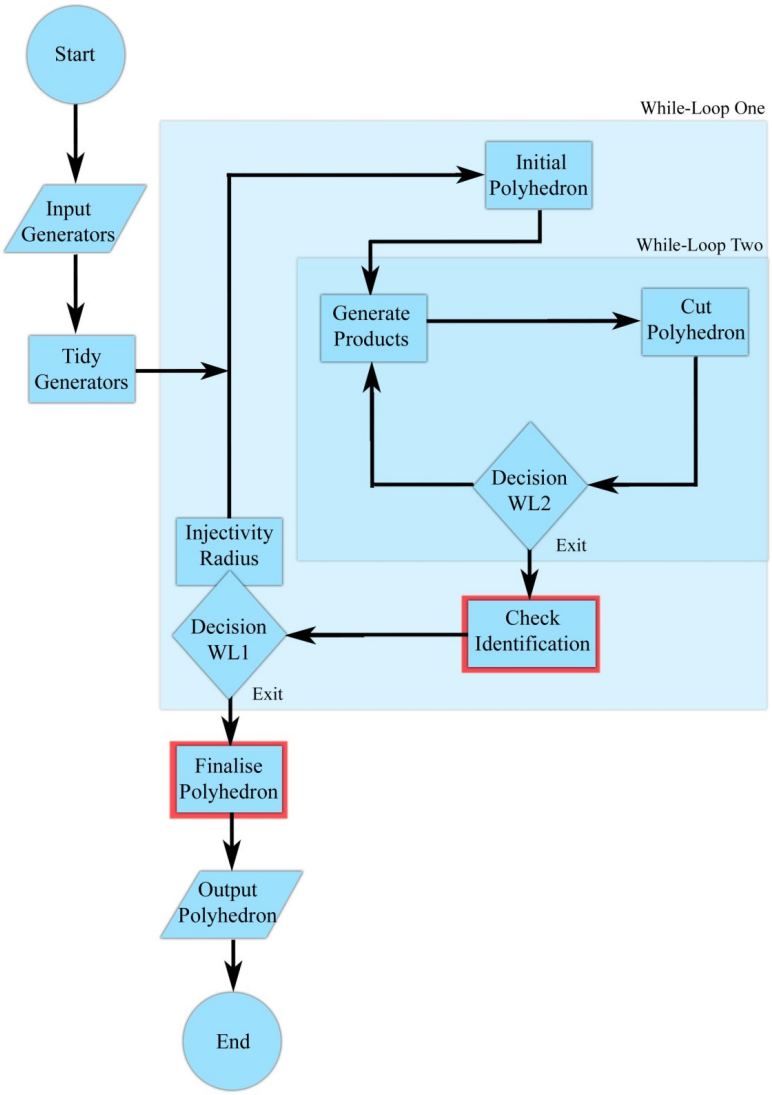
3.3.1 The Construction Procedure

The general process of construction is outlined in Figure 3.5. We break it down to several key steps, which we cover in detail below; note that this is an expanded, and case dependent, version of the basic Dirichlet domain construction process described in Sections 3.2.1 and 2.5.3.

Figure 3.6 demonstrates the physical alterations made to the polyhedron as it progresses through these seven steps.

Given an input list of generators:

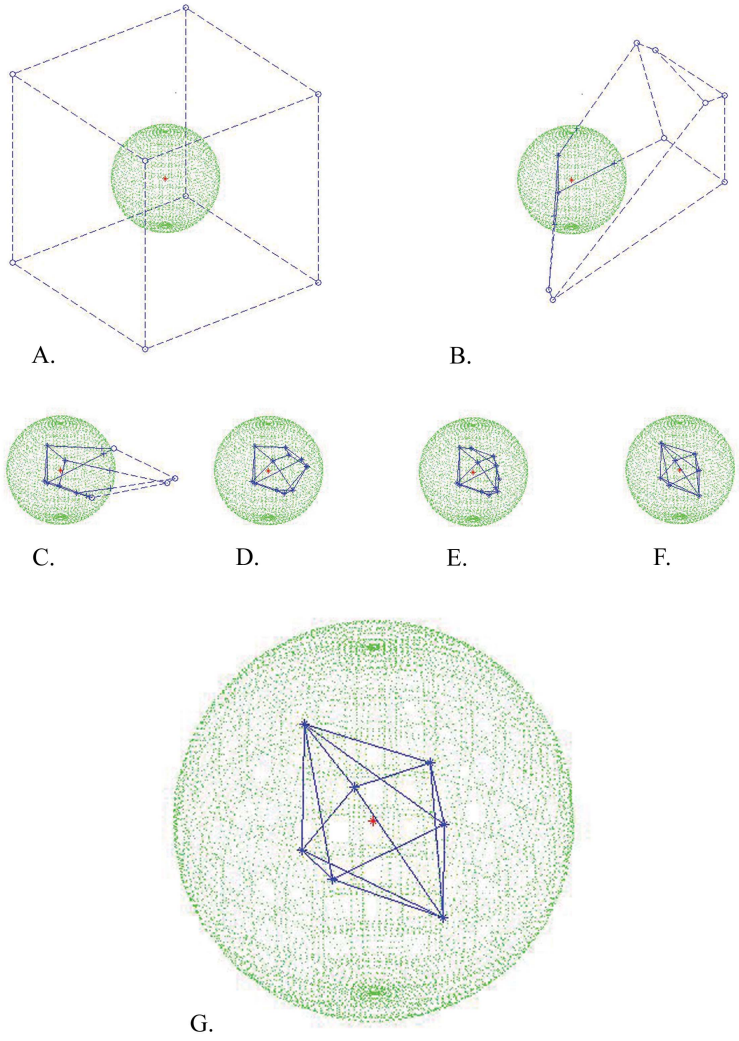
1. Check the input generators;



`Decision WL1`: Exit if the injectivity radius is maximised.
`Decision WL2`: Exit if the polyhedron is finite.

The details of red-boxed steps `Check Identification` and `Finalise Polyhedron` are shown in Figures 3.7 and 3.9 respectively.

Fig. 3.5: Flow chart of the construction procedure.



Step: (A) the initial cube; (B) the initial cuts; (C)-(D) bulk slicing; (E)-(F) maximising the injectivity radius; and (G) the finalised domain.

Fig. 3.6: The construction of a Dirichlet domain.

2. Create a hyperideal polyhedron P_0 (initial cube) and perform the initial, generator determined, cuts to make P_1;

3. Calculate the products of current side-generating transformations and cut with them (bulk slicing);

4. Repeat Step 3 until P_i is a finite polyhedron;

5. Check the identification on P_i;

6. Attempt to shift the basepoint and maximize the injectivity radius of P_i, if the basepoint moves, start from Step 2 again; and

7. Finalize the domain by determining the general constants and quotient map equivalence classes.

Note that this process has a double while-loop structure due to Steps 4 and 6; Step 4 being a crucial processing loop. As noted below there is also a loop structure in the side-paring routines of Step 5.

For code reference: `Dirichlet.c` handles Step 1, and calls the functions from `Dirichlet_basepoint.c` and `Dirichlet_extras.c` for Steps 6 and 7 respectively. Whereas the majority of the construction itself, taking place in Steps 2-5, is handled by `Dirichlet_construction.c`. See Section 3.4.1

Generators (Step 1)

The `Dirichlet_` subroutines are begun with the acceptance of a list of matrix group generators $g_1, ..., g_n \in PSO(1,3)$, which are immediately adapted into a more useable Matrix_pair data structure, that links a generator and its inverse, ensuring they are always accessible together.

However, before any attempt at construction is made, the program attempts to 'tidy' the generators. This is done to ensure that none of the generators fix the basepoint 0, and that the cuts induced by the different generators don't remove one another.

Ensuring that each generator induces a side-generating cut is important for the group tracking procedure, as detailed in Section 3.2.1. If a generator is identified as inducing a degenerate cut, then it is simplified by a composition with other generators to provide an equivalent, non-degenerate, generator set. This is the set $G_0 = \{g_j\}$.

Initial Cube and Cuts (Step 2)

With the starting generators confirmed, the construction can then begin with the initial cube P_0 - a cube embedded in the hyperideal space of $\mathbb{D}^3$, as discussed in Section 3.2.1; with the intention that this cube be trimmed down to a finite polyhedron.

The sides of this initial cube are not associated with any group elements, ensuring they are empty components of the general cutting process.

With a construction space initialised, the cube is then cut by the hyper-planes induced by the individual actions of the initial group generators. This provides the initial sides, and links the group generating process with the geometric cutting process.

$$P_1 = P_0 \cap_{\forall g_j} H_{g_j}(0).$$

Bulk Slicing (Steps 3 and 4)

This is the main construction process undertaken by the function `initial_polyhedron()` in the file `Dirichlet_construction.c`, see Section 3.4.1. We consider these steps to be the main constructive action of the `Dirichlet_` subroutines.

Here, a list G_i is created that contains all the side-generating transformations of the current domain P_i together with their first-order products; the polyhedron is then cut by the planes induced by these products to obtain P_{i+1}; and the process is repeated.

There are only two possible exits from this loop:

1. the successful construction of a finite polyhedron (a polyhedron fully contained in $\overline{\mathbb{D}^3}$); or

2. the accumulation of error beyond the acceptable bounds of resolution.

This process proceeds quickly, with a successful construction typically completing this step in well under ten iterations of the loop.

Identification (Step 5)

If the process makes it to Step 5, then the polyhedron P_i is an accurate representation of a superset (in $\overline{\mathbb{D}^3}$) of the intended Dirichlet domain. At this point the routine `compute_Dirichlet_domain()` calls the functions `check_faces()`, `verify_faces()` and `verify_group()`, to check and develop the identification

on the polyhedron[1]. These ensure that the construction is a fundamental polyhedron, as desired, and complete the general polyhedron construction process.

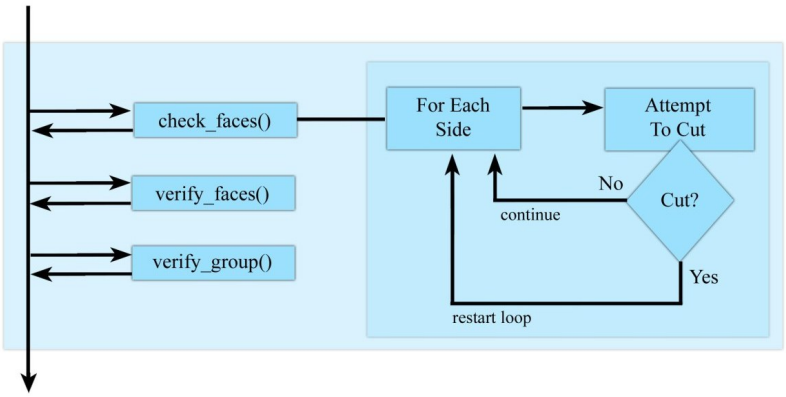


Fig. 3.7: Flow chart of the identification process.

The identification is checked in three stages, the first two ensuring that there is an identification on P_i, with the third confirming the constructed polyhedron as an actual fundamental domain for the group and not some finite-sheeted cover.

1. `check_faces()` runs through each side of the polyhedron, checking that the identification maps each side's edge into its mate;

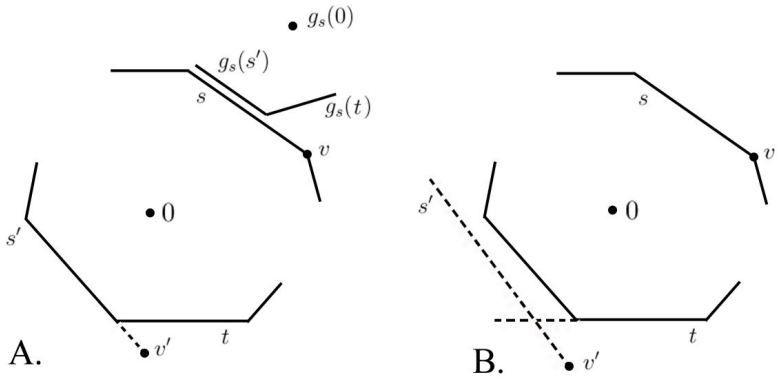


Fig. 3.8: Paring two sides.
(A) a side s with mate s'; and (B) a mateless side s.

   This is actually a secondary construction routine, attempting to "pare" (or "cut") each side with various transformations generated via its mate (paired side) and those sides adjoining its mate. The side-paring process is split into two cases: sides with mates on the polyhedron; and sides without.

[1] Recall that, for 3-dimensional polyhedrons the terms face and side are largely synonymous.

Case 1 Side s with side-generating transformation g_s and mate s'.
Consider s' and an adjacent side t. If there exists a point v' such that $v' \notin s'$ and $v' \notin H_t(0)$ but $g_s(v') \in s$, then take $v = g_s(v')$.

Case 2 Side s with side-generating transformation g_s but no mate.
Let $s' = g_s^{-1}(s)$, then as s' is not a side of the polyhedron there must exist some side t such that the cut induced by g_t would cut s'. Take $v \in s$, such that $v' = g_s'(v) \in s'$ would also be removed by g_t.

Then

$$\begin{aligned} \rho(0, v) &= \rho(g_s(0), v), \\ \rho(0, v') &> \rho(g_t(0), v'). \end{aligned}$$

from which it follows that

$$\rho(0, v) > \rho(g_s \circ g_t(0), v).$$

Thus the transformation $b = g_s \circ g_t$ will induce a cut on the polyhedron removing the part of s containing v.

In either case, determining the side t is not straight forward. Instead the procedure undertaken is to attempt to pare s with any transformation b induced by a valid side, t, of the polyhedron. In the mateless case a cut must be induced by some choice of side, and if one is not, then a topological error is reported. In the mated case, if no cuts are made, then it follows that $g_s(s') \subset s$.

This check processes through all sides of the polyhedron under a `for-loop` that is reset whenever a cut is made. As these cuts can introduce new sides into the polyhedron, this process continues until no more cuts can be made, thus ensuring that all faces have been checked.

In this way `check_faces()` ensures that each side is a subset of its mate under the side-pairing transformations, implying $g_s s' = s$ for all sides s.

2. `verify_faces()` provides an additional check to the result of `check_faces()` by looping through each side of the polyhedron and confirming that each pair of sides has the same number of edges; and

3. `verfiy_group()` then goes through each transformation in the generator list and attempts to reconstruct it from the side-generating transformations.

The combination of these three processes confirm that the polyhedron P_i has an identification. And as the set of side-generating transformations generate the elements of G_0, that identification generates the original group G. Thus $G_s = G$ and $P_i = P = D(G, 0)$.

Injectivity Radius (Step 6)

The injectivity radius, r, is the radius of the largest ball centred at 0, $B(0, r)$, that is fully contained within the polyhedron P; and is related to exponential maps, [18]. A maximised injectivity radius gives some indication towards there being a minimal number of sides, and increased symmetry, in the constructed polyhedron.

If the construction reaches this step, then it has been demonstrated that the input generators lead to the construction of a fundamental polyhedron. So `compute_Dirichlet_domain()` calls the `Dirichlet_basepoint.c` routines. This routine attempts to determine a conjugation of the set of side-generating transformations that will maximise the distance from the basepoint to the closest side, subject to the condition no other side becomes closer than it. This is a non-linear problem so calculations are done with a linearised version.

The program loops through the requisite calculations keeping track of the overall basepoint displacement. If the basepoint moves less than a minimal amount, then the calculated domain is considered to already have a maximised injectivity radius; and the program moves on to Step 7. Otherwise, attempts to move the basepoint continue until they reach the bounds of the linearisations accuracy, at which point the program returns to Step 2, recomputing the fundamental polyhedron using the full group of side-generating transformations as generators.

We note that there is an allowance within the calling of the domain construction routines to skip Step 6 and so avoid this component of the construction process.

Finalisation and Output (Step 7)

Once a final fundamental polyhedron has been constructed, the `compute_Dirichlet_domain()` routine passes the construction on to the `bells_and_whistles()` function of `Dirichlet_extras.c` for finalisation. The general process of this step is outlined in Figure 3.9.

At this point in the construction process, all that has been computed is a winged edge data structure, with abstract group elements associated with each side, such that this data can be combined to provide an identified polyhedron. In other words, the only known data is basic geometric "position" information and

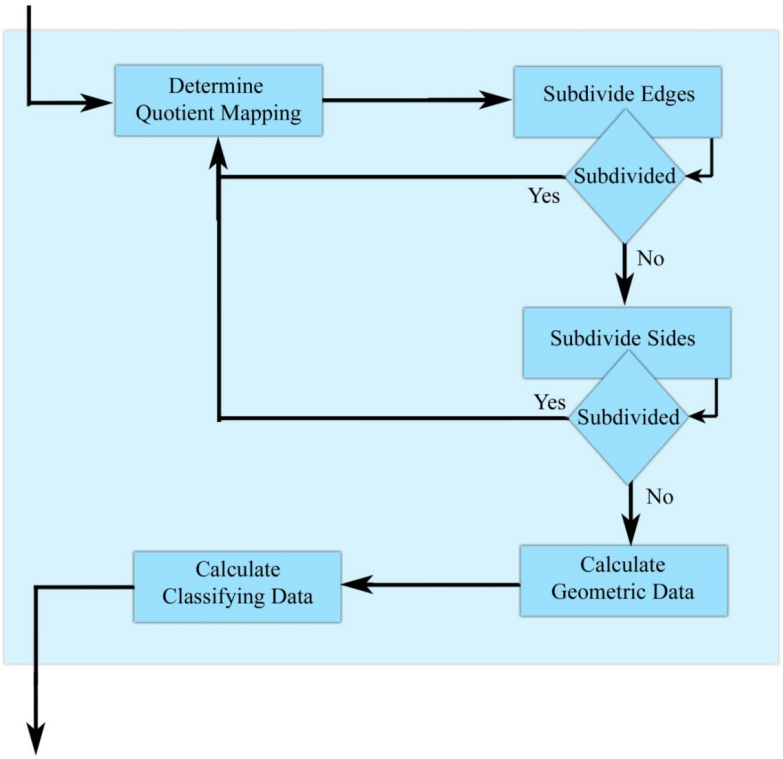


Fig. 3.9: Flow chart of the finalisation process.

individual side-pairing information. The goal is then to combine this information to provide a full identification (quotient mapping) on the polyhedron and calculate all relevant domain information.

1. **Quotient Mapping:**

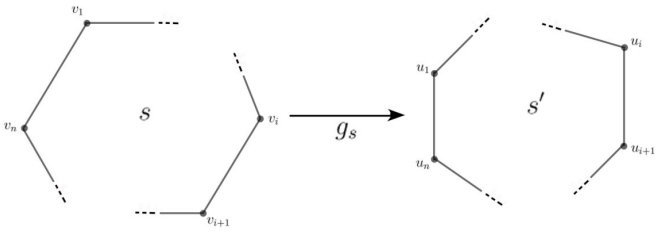
   The first step in this process is to classify all the sides, edges and vertices under their identified quotient mapping. This has already been done on the set of sides through the generation of side-pairs (under side-pairing transformations).

   Identifying the edge classes is not quite as straightforward with each side only being aware of a single, arbitrary, bounding edge, and needing to iterate through all possible matches between the bounding edges to identify the correct mappings. This is achieved by looking at the edges of side s, with sequential vertices labelled $v_1, ..., v_n$, and its pair s', with vertices labelled $u_1, ..., u_n$, and determining which $m \in \mathbb{N}$ gives the smallest sum

of distances between the vertices of mapped edge pairs.

$$\sum_{i=0}^{n} \|u_{i+m(modn)} - g_s(v_i)\|_E.$$

This then gives the set of quotient map identities seen Figure 3.10, and edge classes are then set accordingly.



$$u_{i+m(modn)} = g_s(v_i)$$
$$\pi(u_{i+m(modn)}) = \pi(v_i)$$

Fig. 3.10: Determining edge pairings.

During this edge matching step, the specifics of how each member of the edge class maps to its mates are also determined. Identifying vertices can then be achieved through identifying a vertex with all those it is mapped to under the recorded edge classes.

2. **Subdivision:**

Having specified the quotient map, the program then attempts to ensure that the singular set of the constructed orbifold is contained within the subcomplex of the polyhedron's cell construction. That is, it ensures no points of the singular set are part of the interior of a side. This is done in two steps:

- All edge classes containing an edge that maps to its inverted self are split in two, then a new quotient map is generated; and
- All sides that map to themselves are subdivided appropriately, then a new quotient map is generated.

The sub-division of a side s is split into three possible cases:

(A) g_s is a reflection through a point in the centre of s, in which case the side is split into multiple parts;

(B) g_s is a reflection in s, in which case no actual subdivision occurs; and

(C) g_s rotates one half of the side onto the other, in which case the side is split in two.

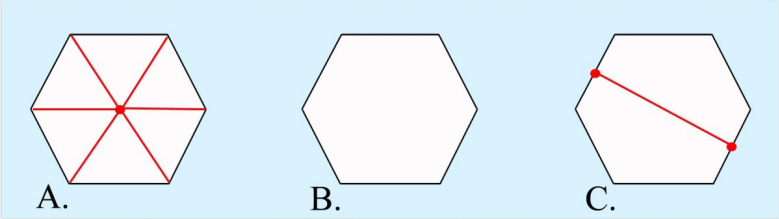


Fig. 3.11: Subdividing a side.
(A) reflection in a point; (B) reflection in the side; and
(C) rotation about an axis.

This last case is the only orientation-preserving case[2], and the side is split along the axis of the transformation g_s. See Figure 3.11.

3. **Calculating Geometric Properties:**

   With a full identification determined on the polyhedron, geometric data is then computed for the construction. This includes the calculation of side, edge and vertex distances and lengths, and calculating the variation in each of these measures within the individual edge and vertex classes - if this variation is too great the construction is abandoned and an error reported.

   The dihedral angles at each edge are also calculated, as are the solid angles at each vertex. This data is then used to determine the sum of angles in each class, and provide their singularity order; specifically, the real number v in equation

$$v \sum_{i=1}^{m} \alpha(e_i) = 2\pi$$

   is calculated for each edge class.

   The finalisation of the construction is then completed with the calculation of certain construction classifying data. The inner and outer radii, the geometric Euler characteristic, the spine radius and volume of the polyhedron are all calculated; along with the deviation of the construction. This deviation is the maximum of all the side-generating transformation deviations (from a $PSO(1,3)$ matrix), and is usually very small - between 10^{-12} and 10^{-15}; see `DEVIATION_EPSILON` in Section 3.4.2.

   As the volume of the polyhedron is of key interest in our work (as the group's covolume), we note that the volume is calculated via decomposing

[2] While our discussion, and interest, is focused on the situation involving only orientation-preserving transformations, SnapPea itself is not restricted to using only these isometries and so can be used to construct domains for non-orientable orbifolds.

the polyhedron into "birectangular tetrahedra" and then determining the volume of each tetrahedron individually by the Lobachevsky function.

Upon the successful completion of this step, the polyhedron has been fully described as a quotient space and fundamental polyhedron for the group G. The data for this fundamental polyhedron is then returned back through the calling program as a final output.

3.4 Code Excerpts

For additional reference, we end this Section with streamlined code excerpts for the four of the main routines found in the `Dirichlet_` subroutine files, see 3.1.1; followed by a brief, descriptive list of the main error bounds.

3.4.1 Code

The first three of these excerpts comprise the main construction routines, with the fourth being the top level finalisation routine.

The comments made with regard to each routine should be considered alongside the process outlined at the top of Section 3.3.1.

Dirichlet_from_generators_with_displacement

```
00  Dirichlet_from_generators_with_displacement(){
01      array_to_matrix_pair_list()
02      precise_generators()
03      simplify_generators()
04      if generator_fixes_basepoint()
05          conjugate_matrices()
06      while(true){
07          compute_Dirichlet_domain()
08          if (polyhedron == NULL)
09              return NULL;
10          if (maximize_injectivity_radius == true){
11              maximize_the_injectivity_radius()
12          }else{
13              if (Dirichlet_bells_and_whistles() != func_failed)
14                  return polyhedron;
15              else
16                  return NULL
17          }
18      }
19  }
```

This function is the main routine of the file `Dirichlet.c`, providing external access to the `Dirichlet_` subroutines and running checks on both the input and final polyhedron.

- Lines `01-05` comprise Step 1;
- Line `07` calls `compute_Dirichlet_domain` for Steps 2 through 5;
- Line `11` calls `maximise_the_injectivity_radius` for Step 6; with

- Lines 06-18 providing the associated loop; and
- Line 13 calls `Dirichlet_Bells_and_Whistles` for Step 7.

Within the SnapPea code there are multiple routines that provide external access to the `Dirichlet_` subroutines, but they all call this function.

compute_Dirichlet_domain

```
00  compute_Dirichlet_domain(){
01      initial_polyhedron()
02      if (check_faces() == func_failed)
03          return NULL;
04      count_cells()
05      sort_faces()
06      if (verify_faces() == func_failed)
07          return NULL;
08      if (verify_group() == func_failed)
09          return NULL;
10      rewrite_gen_list()
11      return polyhedron;
12  }
```

This function is called by `Dirichlet_from_generators_with_displacement()` to run Steps 2-5, and is the top routine of the file `Dirichlet_construction.c`. It provides access to the construction and initialisation routines for Steps 2-4 and then runs through Step 5.

- Line 01 calls `initial_polyhedron` for Steps 2 through 4;
- Line 04 is an error check, to ensure the construction is a polyhedron; and
- Lines 02, 06 and 08 comprise Step 5.

initial_polyhedron

```
00  initial_polyhedron(){
01      new_WEPolyhedron();
02      make_cube()
03      if (slice_polyhedron() == func_failed)
04          return NULL;
05      while (has_hyperideal_vertices() == true)}
06          compute_all_products()
07          if (slice_polyhedron() == func_failed)
08              return NULL;
09      }
10      return polyhedron;
11  }
```

This function is called by `compute_Dirichlet_domain()` to run Steps 2-4, and is the major initial construction routine of `Dirichlet_construction.c`.

- Line `02` is the first part of Step 2;
- Line `03` is the second part of Step 2;
- Lines `05-09` is the loop of Steps 3 and 4;
- Lines `06` and `07` being Step 3.

Dirichlet_bells_and_whistles

```
00  Dirichlet_bells_and_whistles(){
01     face_classes()
02     edge_classes()
03     vertex_classes()
04     subdivide_edges_where_necessary()
05     subdivide_faces_where_necessary()
06     dihedral_angles()
07     solid_angles()
08     if (vertex_distances() == func_failed)
09         return func_failed;
10     if (edge_distances() == func_failed)
11         return func_failed;
12     face_distances()
13     if (edge_lengths() == func_failed)
14         return func_failed;
15     compute_approx_volume()
16     compute_inradius()
17     compute_outradius()
18     compute_spine_radius()
19     compute_deviation()
20     compute_geometric_Euler_characteristic()
21     return func_OK;
22  }
```

This function is called by `Dirichlet_from_generators_with_displacement()` and is the primary routine of the file `Dirichlet_extras.c`, completing Step 7 by calling all the functions to finalise the polyhedron. We note that there are only three conditions where the construction might return failure, though there is the possibility of fatal errors occurring in, and being identified by, many of the other computations.

3.4.2 Constants

We also provide a list of the more noted computational constants used in the core `Dirichlet_` subroutine files. We categorise these based on parent file; and, where

applicable, we briefly describe the processes related to these constants. As with the code above, these constants should be considered alongside the processes and other references within this chapter.

In file: `Dirichlet.h`

`MATRIX_EPSILON` ($\varepsilon_M = 10^{-5}$).

For all computational purposes, two matrices $A = (a_{i,j})$ and $B = (b_{i,j})$ in $PSO(1,3)$ are considered equal if and only if $|a_{i,j} - b_{i,j}| < \varepsilon_M$ for all i, j.

In file: `Dirichlet.c`

`SIMPLIFY_EPSILON` ($\varepsilon_{simple} = 10^{-2}$).

Take matrices A and B in $PSO(1,3)$, A is considered simpler than B if and only if

$$\cosh \rho_H(A(0), 0) < \cosh \rho_H(B(0), 0) - \varepsilon_{simple}.$$

This measure is used to determine the potential improvement made in moving from generators $\{g, f\}$ to $\{g, gf\}$, in an attempt to avoid degeneracy in the generator set.

`FIXED_BASEPOINT_EPSILON` ($\varepsilon_{fix} = 10^{-6}$).

A transformation matrix $A = (a_{i,j})$ in $PSO(1,3)$ is considered to fix the basepoint if and only if $a_{0,0} < 1 + \varepsilon_{fix}$.

In file: `Dirichlet_construction.c`

`CUBE_SIZE` ($C = 2.0$).

This is half the width of the initial cube structure, as described in Section 3.2.1.

`ERROR_EPSILON` ($\varepsilon_{error} = 10^{-4}$).

If a non-identity matrix A, in $PSO(1,3)$, is generated, and if

$$\rho(0, A(0)) < \varepsilon_{error},$$

then an error is returned. By starting with group generators, and attempting to maximise the injectivity radius, the routines ensure that this only occurs if A is a (computationally) erroneous identity matrix.

`HYPERIDEAL_EPSILON` ($\varepsilon_{hyp} = 10^{-3}$).

A point x in $\mathbb{R}^{1,3}$ is considered to be hyperideal if and only if

$$\|x\|_L^2 > \varepsilon_{hyper}.$$

Recall that $\|x\|_L^2 = -x_0^2 + x_1^2 + x_2^2 + x_3^2$ and $x_0 = 1$.

`VERIFY_EPSILON` ($\varepsilon_{verify} = 10^{-4}$).

This replaces ε_{simple} in the function `verify_group()`, as it attempts to filter the set of side-generating transformations down to the original set of generators.

`DEVIATION_EPSILON` ($\varepsilon_{dev} = 10^{-3}$).

The deviation of a matrix in $PSO(1,3)$ is a measure of how much the inner-products, of row i and column j, deviate from the expected values (1,0, or -1). If an inner-product's deviation exceeds ε_{dev}, then the calculations have moved too far from $PSO(1,3)$ and so a failure is reported.

In file: `Dirichlet_extras.c`

`DIST_EPSILON` ($\varepsilon_D = 10^{-3}$).

Let g_s be a side-pairing transformation, if

$$|\rho(0, g_s(0)) - \rho(0, g_s'(0))| > \varepsilon_D,$$

then something has gone wrong and an error is reported.

`LENGTH_EPSILON` ($\varepsilon_L = 10^{-3}$).

The variance in edge lengths across an edge class must be no more than ε_L.

`IDEAL_EPSILON` ($\varepsilon_{ideal} = 4 \times 10^{-7}$).

A point x in $\mathbb{R}^{1,3}$, that is not hyperideal, is considered to be ideal if and only if

$$\|x\|_L^2 \geq -\varepsilon_{ideal}.$$

Recall that $\|x\|_L^2 = -x_0^2 + x_1^2 + x_2^2 + x_3^2$ and $x_0 = 1$.

4. IMPLEMENTATION OF THE DIRICHLET ROUTINES

It is our intention to use the routines, outlined in Chapter 3, in an attempt to compute fundamental polyhedrons for groups with two elliptic generators, as introduced in Section 2.4.

We will calculate these polyhedrons directly from their associated (modified) parameter sets:

$$(p, q, \gamma) \mapsto P = D(G, 0)$$

using a modified version of the Dirichlet subroutines.

The actual implementation of the routines in themselves is quite straight-forward, as the only input required is a list of group-generating elements in $PSO(1, 3)$ form.

The specifics of our method of generator construction are covered below in Section 4.2, but they are not a primary focus of this chapter. Here our interest is in how the Dirichlet subroutines, as found in SnapPea and detailed in Section 3.3, may not be generally suitable in application to our groups of interest, as mentioned in Section 3.1.3.

In this chapter we discuss our modifications to the code, which we implement through our new program SnappyD. The goal being to adapt these routines so that they are able to deal with domains represented by hyperideal constructions.

We discuss the results of our program in Chapter 5

4.1 SnappyD

A key part of this project has been exploring and adapting these routines to avoid the issues alluded to above, so that some results about our groups of interest might still be determined. In addition to this, as mentioned in Section 3.1.2, there was an underlying motivation of investigative work - there already being a local SnapPea based program that implemented the `Dirichlet_` subroutines, which failed for certain group constructions.

To this end, we have written a program that interfaces with the (now) modified `Dirichlet_` subroutines of SnapPea. We call our program SnappyD, in reference to SnapPea. In essence, SnappyD is a "wrapper" or "shell" program, providing access to these modified routines and outputting informative data relevant to the construction process.

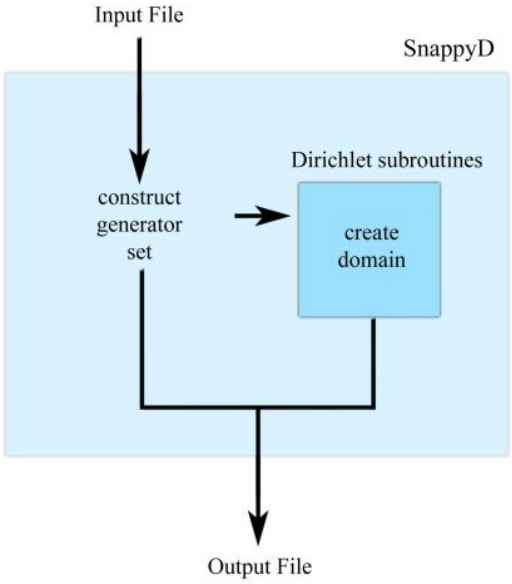


Fig. 4.1: Flow chart of the SnappyD program.

The program does not have a full user interface. Instead we have decided to run it off a series of text-files, with an input list of parameter sets, another input of selected output options and a single output file; as discussed in Appendix B.

Given our motivations and intended use of the program, we have added considerable construction tracking information that can be accessed through the output file. This has provided the information from which the polyhedron plots seen in this thesis have been constructed (using a Matlab based print function); as well as a great deal of insight into the construction process and any potential failures.

This chapter details how SnappyD constructs matrix group generators from a given modified parameter set, and how the `Dirichlet_` subroutines it uses have been modified from those in SnapPea. For more information on SnappyD see Appendix B and the support files.

4.2 Input Generator Sets

SnappyD takes an input parameter set, constructs matrices to represent the group generator set, and then passes them on to our modified version of the `Dirichlet_` subroutines. As such, we start with a discussion on our chosen method of matrix construction, as this informs our discussion on alternative parameter inputs in Section 4.2.2.

Note that our interest is solely in those discrete groups with two elliptic generators, both of finite order, as described in Section 2.4. In this case, for the primitive elliptic

generators f, g with order p, q respectively, we have

$$\begin{array}{rclcrcl} \beta & = & \beta(f) & = & tr^2(f) - 4 & = & -4\sin(\pi/p), \\ \beta' & = & \beta(g) & = & tr^2(g) - 4 & = & -4\sin(\pi/q). \end{array}$$

This allows us to consider modified parameter sets of the form (p, q, γ), as mentioned in Section 2.4.

4.2.1 Generators in $PSL(2, \mathbb{C})$

As noted in Section 2.3.2, each parameter set determines a two-generator group in $PSL(2, \mathbb{C})$ up to conjugacy. As such, there are infinitely many groups upon which each construction can be based on.

For computational purposes, it is appropriate to follow the common method of assuming one generator is a standard form; meaning one generator will have a (near) diagonal matrix representative in $PSL(2, \mathbb{C})$ ($PSO(1, 3)$). We choose the generator corresponding to the lead input parameter, p, to be of this form.

Under this assumption, the generators for the group G with modified parameter set $par(G) = (p, q, \gamma)$, are constructed in $SL(2, \mathbb{C})$ to be:

$$\langle f, g \rangle ,$$

with $f =$

$$\begin{bmatrix} e^{i\pi/p} & 0 \\ 0 & e^{-i\pi/p} \end{bmatrix} \tag{4.1}$$

and $g =$

$$\begin{bmatrix} \cos\left(\frac{\pi}{q}\right) + x & \frac{\gamma}{4}\csc^2\left(\frac{\pi}{p}\right) \\ 1 & \cos\left(\frac{\pi}{q}\right) - x \end{bmatrix}, \tag{4.2}$$

where

$$x = \sqrt{-\sin^2\left(\frac{\pi}{q}\right) - \frac{\gamma}{4}\csc^2\left(\frac{\pi}{p}\right)}. \tag{4.3}$$

A full overview and derivation for the construction of these generators can be found in [16]; though here M_g has been trivially altered from that given in the reference.

Standard Forms, Fixed Points and Conjugation

We note that when viewed as acting on $\hat{\mathbb{C}}$, f fixes the points 0 and ∞; and g has fixed points

$$z = \sqrt{-\sin^2\left(\frac{\pi}{q}\right) - \frac{\gamma}{4}\csc^2\left(\frac{\pi}{p}\right)} \pm i\sqrt{\sin^2\left(\frac{\pi}{q}\right)},$$

which are of the form $z = w + ui \pm vi$, where $x = w + ui$ as in Equation 4.3, and $w, u, v \in \mathbb{R}$.

While due to the complex square root in the initial term there are technically four possible values, the selection of these root pairs is made in the determination of the x term in generating matrix g, see Equations 4.2 and 4.3.

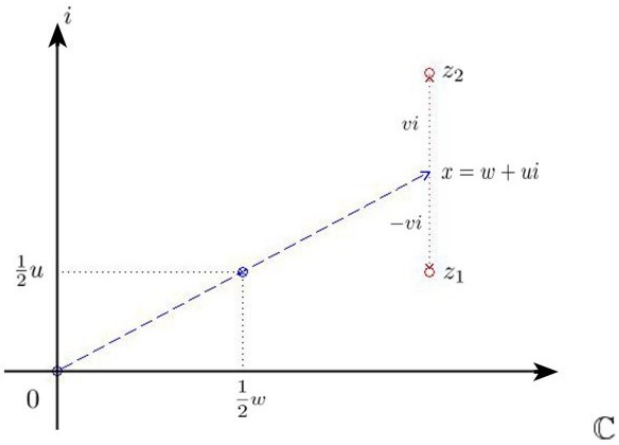


Fig. 4.2: The fixed points of f and g on $\mathbb{C}$.

The fixed points, or axes, of the isometies f and g, acting on $\mathbb{U}^3$, are then the geodesic arcs between these fixed points. In the case of f this is the positive component of the x_3-axis, which includes the Dirichlet basepoint $(0, 0, 1)$ (in $\mathbb{U}^3$).

As stated in Section 3.3, the `Dirichlet_` subroutines attempt to ensure that the generating matrices do not contain the basepoint in their set of fixed points, so all of our generator sets will be conjugated on entry into the routines. We had previously attempted a conjugation during generator construction that would move the basepoint to $\frac{1}{2}(w + ui)$; but, in testing, this gave no decrease in measurable error, while increasing the overall manipulation of the generator set, see Section 4.5.3. Thus we have abandoned the alteration, leaving the code available in the back-up source code for possible later use.

Precision

An additional benefit of the f and g representation, shown in Equations 4.1 and 4.2, is the accuracy in the matrix entries.

Of the 16 components of the matrix entries in f and g, 6 are numerical constants and another 4 are trigonometric constants; further, of the remaining 6 components, 2 feature these constants additively, 2 are γ multiples of a trigonometric constant and all solely feature terms involving these constants and γ. In this way, all the entries are derived from the real constants 1, 0, $\sin(\pi/n)$, $\cos(\pi/n)$, $\sin^2(\pi/n)$, $\csc^2(\pi/n)$, and a complex term γ.

Within a general computing system[1] we could compute all this to an arbitrary level of accuracy; but the `Dirichlet_` subroutines work in double precision. Thus, these constants have all been calculated individually to double precision for $2 \leq n \leq 50$ and stored within the generator construction routines of SnappyD.

Thus, for the purposes of this project, we limit acceptable elliptic order values to $2 \leq p, q \leq 50$ and also disallow the elementary cases $p = q = 2$.

$PSO(1,3)$ Representation

While this discussion has focused on the $PSL(2,\mathbb{C})$ representation of $Isom^+(X)$, we note that it is $PSO(1,3)$ matrices that are used in the `Dirichlet_` subroutines, see Section 3.2.1. It was our original intention to construct generators straight into $PSO(1,3)$, see [16], in a bid to boost initial accuracy, however the gains here also appeared inconsequential.

Instead we construct generators in $PSL(2,\mathbb{C})$ and then map them into $PSO(1,3)$ using the routines already present in SnapPea, see Appendix A. This leaves the generator construction process more easily open to later alteration and the immediate display of generators more applicable to common references.

4.2.2 Parameter Sets

Recall that each modified parameter set describes a conjugacy class of two-generator groups and so, by the method of generator construction given above, the projection

$$(p, q, \gamma) \mapsto \langle f, g \rangle$$

produces one specific pair of group generators, out of an infinitude of possible pairs.

While in general not all parameter sets correspond to a discrete group, for our purposes we assume all those we work with do; each set of parameters having been derived through arithmetic or other means, as seen in Chapter 2.

Considering that, as noted in Section 3.2.3, the `Dirichlet_` subroutines only proceed while error does not exceed the bounds of acceptable accuracy, the use of multiple computationally distinct generator sets will provide a level of redundancy in our overall calculations. This should not only provide additional likelihood of success, but also further insight into the robustness and accuracy of the constructions based around each group.

Rather than directly manipulating our generator constructions, we can instead use variations of the parameter set to construct equivalent, or closely related, group generators that are often computationally distinct from those already available.

[1] That is to say, in a computational computer program like Maple or Mathematica.

Parameterisation

We consider a modified parameter set, of the form (p, q, γ), from which generating matrices are constructed under the assumption that the lead generating element will be a standard form; with all three parameters then influencing the construction of the second generator. However, as

$$\langle f, g\rangle = \langle g, f\rangle$$

and

$$tr[f, g] = tr[g, f]$$

the parameter sets (p, q, γ) and (q, p, γ) must both represent the same conjugacy class of groups in $Isom^+(X)$.

This is not surprising, as the usual projection onto an order n commutator plane is arbitrary with respect to generator order. In this situation, it follows that if $p \neq q$, then the set (q, p, γ) will generally lead to the construction of an equivalent, and computationally distinct, set of generating matrices.

Similarly, the γ parameter of each set will typically appear as the complex root of a polynomial; these roots occur in pairs, and each group parameterised by one of these complex conjugates provides another computationally distinct set of generators.

This infers that while a parameter set determines a group up to conjugacy, for our constructions the particular representation of the parameter set determines the particular member of the conjugacy class that is generated. Thus, for our purposes, when $p \neq q$ and $Im(\gamma) \neq 0$, we have four computationally distinct, but equivalent, parameter sets:

$$(p, q, \gamma), \tag{4.4}$$
$$(q, p, \gamma), \tag{4.5}$$
$$(p, q, \overline{\gamma}), \tag{4.6}$$
$$(q, p, \overline{\gamma}). \tag{4.7}$$

With $p = q$ and $Im(\gamma) = 0$ each reducing the number of variant options by a factor of two.

$\mathbb{Z}_2$ extension

In the case where $p = q$ there is no distinction between the parameter sets 4.4 and 4.5, or 4.6 and 4.7. However, by using the $\mathbb{Z}_2$ extension as detailed in Section 2.3.3, new groups can be constructed which are generated by an order p $(= q)$ element, an order 2 element, and have a gamma parameter γ_0 determined by the equation

$$\gamma = \gamma_0(\gamma_0 + 4\sin^2(\pi/p)).$$

This gives four new parameter sets:

$$(p, 2, \gamma_0) \tag{4.8}$$

$$(p, 2, \overline{\gamma_0}) \tag{4.9}$$

$$(2, p, \gamma_0) \tag{4.10}$$

$$(2, p, \overline{\gamma_0}) \tag{4.11}$$

The groups represented by these new parameter sets should be distinct from the original group, but remain members of the same commensurability class of groups. Specifically, these groups can be expected to have half the covolume of the original group; if they don't, then they should have the same volume, indicating that the particular order-two generator is an element of the original group.

Parameter Sets

In this way, from any given parameter set, (p, q, γ), for a two-elliptic-generator group:

- If $\gamma \notin \mathbb{R}$, then we can consider constructions generated by four (two if $p = q$) initial matrix generator sets, taken from the computationally distinct parameters sets 4.4 - 4.7; and
- If $p = q$, then we can also consider these alongside the constructions generated by the parameter sets 4.8 - 4.11, giving a total of 6 initial matrix generator sets to work from.

In the case the $\gamma \in \mathbb{R}$, these numbers are halved.

Following this idea, we use multiple generator sets, noting that while it is hoped that each of the distinct but equivalent generator sets will give the same result, it is neither uncommon nor unexpected for there to be some discrepancy, particularly for large values of p, q or γ. We highlight some of these variations in Chapter 5.

We note that the $\gamma \in \mathbb{R}$ case, and indeed the general case of $\beta, \beta', \gamma \in \mathbb{R}$, has been classified [29]; and so are not a part of our general focus.

4.3 Failures in Implementation

When it works, the domain construction procedure outlined in Chapter 3 is fast, completing in a matter of seconds; with the main construction loop (while loop two, see Figure 3.5) generally terminating in well under ten iterations for relatively simple domains. However, it is not hard to find discrete groups in our class of interest where the initial construction fails to finish within 10^6 iterations; and, if we artificially terminate the loop and attempt to continue with the construction after a 'reasonable' number of iterations, then the domain fails topological checks and the program fails in completing the domain's construction.

This non-completion of while-loop two is quite distinct from the standard construction failures outlined in Section 3.2.1. Those failures comprising of a series of controlled

exits that are induced by geometric or group theoretic incongruities that are often due to an accumulation of error. While failure on artificially exiting the loop is not unexpected, as loop exit-criteria have not been satisfied, getting to a point where artificial exits are required is an issue.

There are two primary reasons for these continuing loops:

1. the desired construction is a hyperideal polyhedron; or
2. effective cutting action is not being undertaken.

The first issue is of key interest, as the system of the entire `Dirichlet_` subroutines is built around the assumption of a domain represented by a finite polyhedron. Identifying this case, while preventing a loop induced by the secondary cause, is the focus of this section.

Below we discuss our expanded description of domains to further classify these hyperideal issues, before detailing the alterations we have then been able to make to the `Dirichlet_` subroutines; this has allowed us to construct representations of domains with ideal sides using hyperideal polyhedrons.

4.3.1 Hyperideal Issue

As the `Dirichlet_` subroutines are not designed to deal with hyperideal polyhedrons, adapting the routines to work with them is a primary concern.

When the attempted construction of such a representative polyhedron occurs, it is often observable in the evolution of a domain structure. This is demonstrated in the non-completing construction seen in Figure 4.3. The sides that remain consistently throughout the shown construction are lying on the initial cube and it is clear that there is an issue in cutting off hyperideal regions in the domain; this construction process never exits while-loop two due to domain's hyperideal components. Compare this with Figure 3.6.

This is seen in most relevant constructions that fail to complete, and it does not seem inappropriate to assume that almost all non-completing constructions (as opposed to those constructions that fail) stem from the desired domain requiring hyperideal representation.

As noted in Section 3.2.1, the `Dirichlet_` subroutines make no provision for a fundamental domain with ideal sides, as any such domain will have a hyperideal representation in the polyhedron structure underlying the routines. The assumption of domain with only finite or proper vertices being a defining feature of the construction process, most specifically in the exit requirement for while loop two, see Section 3.3.1.

This is a serious obstacle in our use of the code; as we have seen in Section 2.4, there are infinitely many groups that fit in this incompatible category that are in our class of interest.

As the key issue we need to work around involves the type (finite, ideal or hyperideal) of

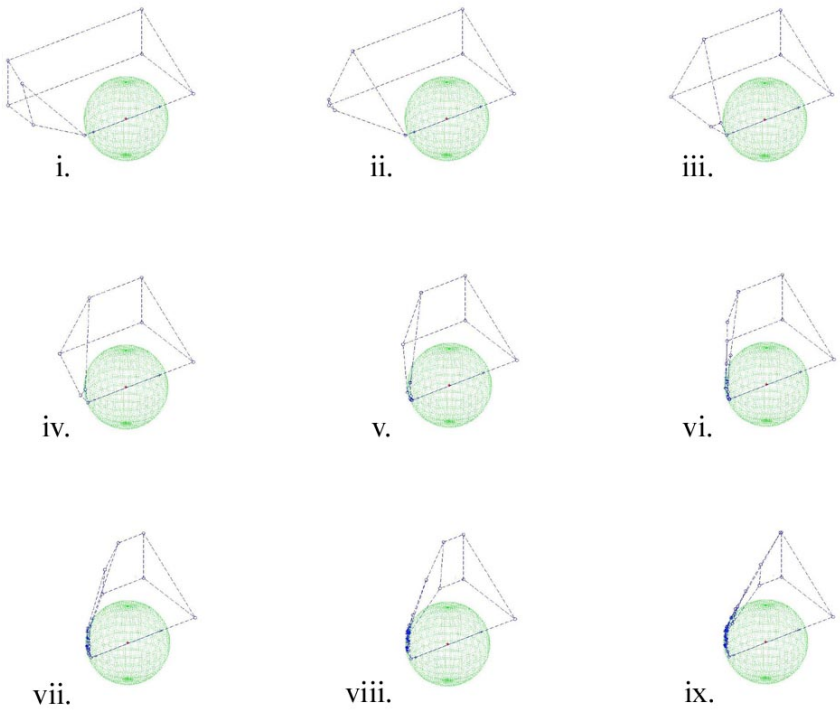


Fig. 4.3: Hyperideal construction failure.

each point in the polyhedron, we have implemented a classification over the developing polyhedron.

Classification of the Hyperideal Parts

At every major step in the construction or finalisation of the polyhedron (outside of the basic cutting steps), we classify the current polyhedron with respect to its finite and hyperideal components. In the `Dirichlet_` subroutines there already exists a Boolean function for determining whether or not a given polyhedron is hyperideal, based on the vertices, but this work expands on it greatly.

We classify the polyhedron in four steps, each relating to an individual component:

- We identify the nature of each vertex as either finite, ideal or hyperideal; using the measures given in Section 3.2.1. This classification of vertices provides the details upon which the remaining classification is processed.
- Each edge, e, is classified relative to three points, the two endpoints v_1 and v_2, and v_0, the point on the edge closest to the origin (basepoint):
  - If v_0, v_1 and v_2 are hyperideal, then e is hyperideal;
  - If v_1 and v_2 are both non-hyperideal, then e is finite; or

  - If either of v_1 and v_2 are hyperideal while v_0 is finite, then e is retracted.

  Note that as v_0 is at least as close to the origin as v_1 and v_2, it is not possible for v_0 to be hyperideal if either of v_1 and v_2 are not.

- A side, s, is classified first by whether or not it has a side-generating transformation, and then by the classification of its bounding edges.

  - If s has no associated side-generating transformation, then it is a false side to be ignored for all construction and classification purposes.

  Otherwise:

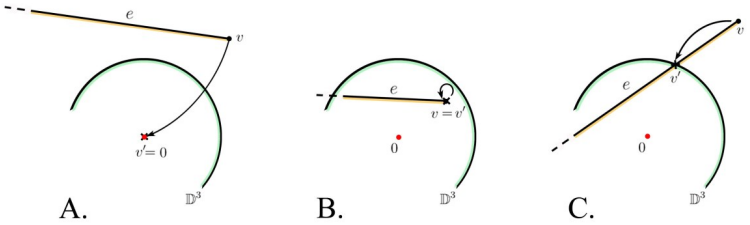
  - If s has only finite edges, then it is finite.
  - If s has retracted edges, then it is retracted; and
  - If s has only hyperideal edges it is (bounded) hyperideal;

The polyhedron P_i itself can then be classified by a check of whether or not it has hyperideal vertices, and whether or not it has retracted edges.

Retraction Points

We have also expanded the description contained in the edge data-structure to include the retraction point for each of its end points. If the end vertex is finite (or ideal) then the retraction point matches it. If hyperideal, but retractable, it holds the ideal point that the vertex would retract to - this point is the intersection of the edge with the unit sphere. Retraction points are calculated by the bisection method, using SnapPea's measures for ideal and hyperideal points to determine an edge's ideal intersection point.

If the edge is marked hyperideal, then it has no finite component and the retractions are marked to be the basepoint, 0. This mirrors a general classification of hyperideal components where we give them zero position, length, volume, etc.



The retraction point v', on edge e, of vertex v.
Where e is: (A) a hyperideal edge;
(B) a finite edge; and (C) a retracted edge.

Fig. 4.4: Retraction points and edge classification.

For non-hyperideal edges, the segment between the retraction points is the "real" useable component of the edge.

We note that retraction points are marked with respect to edges; in this way each vertex may have multiple, distinct retraction points. Further, no finite vertex may

exist at the basepoint, as this would infer that the basepoint is fixed by some group element; so hyperideal retractions being set to this point adds no ambiguity to the construction.

The retraction points are stored in the edge data-structure only. As the polyhedron is the convex hull of its vertices, and each vertex is the intersection of multiple edges of sides, the retraction information cannot be coherently stored in the vertex construct. Vertices do, however, keep track of whether of not they are linked to a retraction point on an edge.

False Sides

In classifying the sides as finite, retractable or hyperideal; we also assumed that hyperideal sides are those without group action, this is not an entirely valid assumption, as there could theoretically exist generated sides whose finite components have been wholly removed by a subsequently generated side; but this is preferable to removing any potentially useable data from the polyhedral data structure.

As far as we are aware this situation has not arisen, and if it were to, then it would either: cause an error in the `check_faces()` function, denying a successful construction over outputting incorrect results; or, have no effect on the construction or its validity.

Infinite Edges

Through this classification we have been able to alter the processes used throughout the program, so as to take into account the expanded nature of the desired polyhedral constructions. However, given the highly specific nature of infinite edges (see Section 2.6.2), and the fuzzy ideal boundary inherent in the `Dirichlet_` system, we have decided not to expand our classification to attempt such a definitive classification of infinite edges; if any such points exist.

It is worth noting that if infinite edges do not exist then the domain is guaranteed complete and potential infinite edges should be visible in the final observable geometric structure; should there ever arise any concern over them. Because of this, we do not confirm the completeness of hyperideal domains. This is a minor hole in the program, but we feel it is justified; putting us in line with many references that do not give a full treatment to the completeness condition, as noted in [20].

4.3.2 Loop Exits

With the above hyperideal classification of the constructing polyhedron, we also need an appropriate method of avoiding the aforementioned infinite-loop in while-loop two. Outside of general failure, the loop has one exit condition "`the polyhedron is finite`", so a more comprehensive set of exit criteria is required.

There are two reasons for an apparently non-completing loop:

- No new cuts are being made ($G_{i+1} = G_i \Leftrightarrow P_{i+1} = P_i$) and the current domain P_i is hyperideal.

- Small cuts are being slowly made (error may be in effect here).

Faced with a potential infinite loop, the program needs a way to determine whether the intended construction has reached its end, while still hyperideal in nature, or if the process should be continued in search of a finite end point.

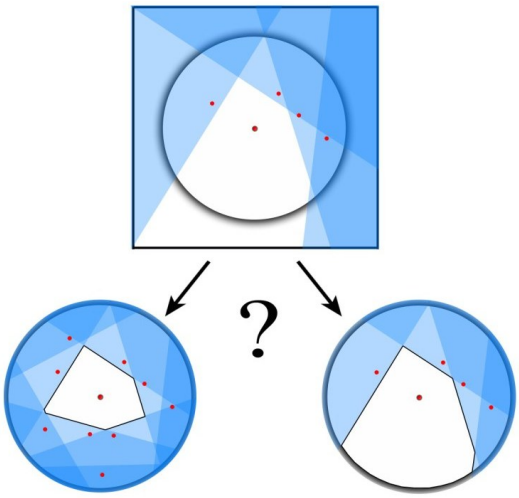


Fig. 4.5: Hyperideal construction decision

As most constructions conclude in fewer than 10 iterations of the basic construction loop, we begin by simply setting a catch-all limit of 50 loops; under the assumption that if cuts are being made at this point, then they are most likely due to an accumulation of error in the matrix representation of group elements. If this is not the case, then it should be noticeable from the output tracking information; and, in either case, the continuing construction procedures in the identification routines will hopefully be able to salvage a valid construction.

For example, this ongoing loop issue can occur at the extreme ends of valid vertex epsilon settings, provided that an accumulation of error does not default the construction first. And a similar, resolution-dependant, infinite loop has been noticed in the side-paring routines when resolution is set too low. We have not yet identified any other valid cases in our groups of interest.

The more likely situation, in falling into an infinite loop, is that the construction is not progressing as no more new cuts are being generated; this implies no new group elements are being added to the working group. Once this happens, no new cuts can be generated.

At this point either the construction is a failure or the domain is hyperideal, and a method of distinguishing between the two cases is required. It is not our desire to force the construction process beyond its normal limits, so we identify when this situation is happening and attempt one extra round of calculating group-element products (twice

in one loop) in an attempt to generate new cutting actions. If this doesn't work, this step of the construction is abandoned and while-loop two is exited.

The idea with this is that we don't want to force the routines through to unnecessary errors; and if we don't make the construction gains we need, then the identification routines may progress the construction further. If not, then there are other generator sets that may succeed. Originally we had set out to generate higher level products[2] when progress was not being made, but this quickly becomes computationally intensive - constructing the m-fold products of side-generating transformations for a polyhedron with n sides involves the construction of m^n matrices in $PSO(1,3)$; a very modest 10 sided figure requires 128000 arithmetic operations to compute the third order products.

In Loop Classification

In the interests of ensuring that the system does not exit while-loop two without as full a construction as possible, especially in the hyperideal case, we have also incorporated the identification routines, discussed in Section 3.3.1 and modified in Section 4.4.1 below, into the loop.

In some ways this is quite a distinct alteration to the flow of the original routines, but the overall effect in this alteration is minimal - simply reducing the number of loops often needed, while allowing for an increased trackable depth within the program. Allowing us to maintain a better record of when key construction points are achieved.

In this new position, these routines are given no ability to fail the construction, acting only as a supplementary set of construction routines and adding some additional options for tracking success within the domain generation process.

With the confirmation of a fundamental domain now available in loop, we leave the original confirmations of while-loop one in place, as a catch-all for any exit and a visible reconfirmation. It should be noted that these alterations of the loop process do not guarantee the success of the overall construction, but do prevent the most common form of infinite loop that can occur in the program.

The modified while-loop two is outlined in Figure 4.6. We summarise the new loop exit criteria and their interaction with the identification routines.

Case 1 `Exit conditions satisfied:`

Loop manually stopped (50 iterations); or finite polyhedron.

`Follow on interaction:`

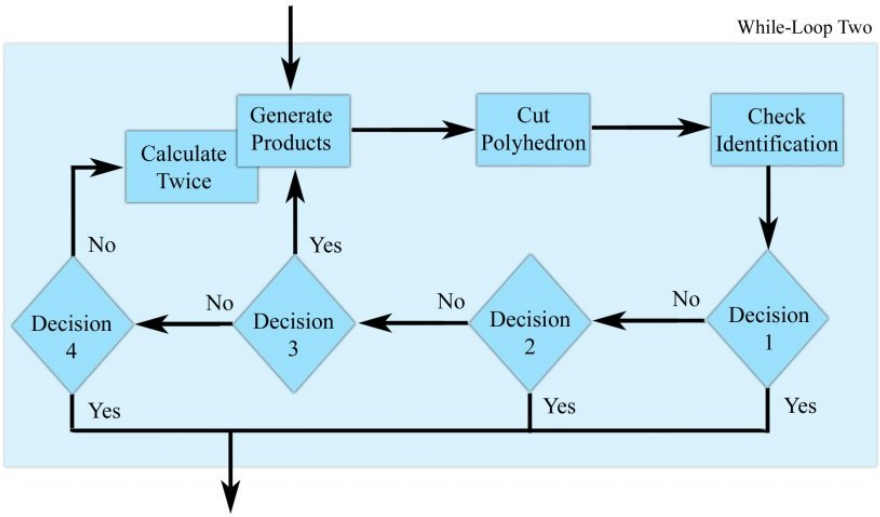
The identification is checked.

If successful, the domain is finalised.

Case 2 `Exit conditions satisfied:`

Cutting has stopped.

[2] We have generally run SnappyD with two variants: one using the double round of product calculations and one that simply exits the loop. But there is an additional variant that will sequentially work up to calculating n-fold products; we do not use this version as it has so far given minimal gains for a considerable increase in computing time. See Appendix B



Decision 1: Is the polyhedron finite?
Decision 2: Is the number of iterations greater than 50?
Decision 3: Is effective cutting action still being made?
Decision 4: Have we attempted to compute second level products?

Fig. 4.6: The modification of while-loop two

(The identification may have been confirmed in loop.)

`Follow on interaction:`

The identification is (double) checked.

If successful, the domain is finalised.

From which it is clear that we have maintained the original integrity of the `Dirichlet_` subroutines while expanding the while-loop two processes to deal with the hyperideal case. Further, the inclusion of identification routines, to allow for the tracking of construction successes allows the process to determine whether the construction itself is actually complete.

As mentioned previously, we have identified one other potentially infinite loop occurrence, in the `check_faces()` component of the identification checking routines, but this is either due to resolution settings or greater errors. We have considered working to also prevent this loop (however rare), but felt that there was no obvious way around it that would not have other potential effects upon the greater class of successful computations.

4.4 Finalisation

Having expanded the main construction and loop processes to develop and classify hyperideal polyhedrons, it is then necessary to expand the identification, basepoint and other finalisation routines to also work with these constructions.

The classification and finalisation routines come after exiting while-loop two in the

construction process. In the standard system this would mean that the polyhedron is finite; but in our modified routines this is not necessarily the case. The first step is to demonstrate the identification on the (hyperideal) polyhedron, confirming a fundamental domain; this may have already been demonstrated in the modified while-loop two, as discussed above. Once this has been done, then the basepoint can be maximised; before developing a quotient map and determining constants of interest.

4.4.1 Identification

Checking an identification on the hyperideal polyhedron requires some additional consideration; with each of the separate verification programs needing to be altered to take into account the existence of hyperideal (or, specifically, transformation-less) sides. To this end we incorporate a new function `classify_polyhedron()` into the routines. This function generates a classification over the polyhedron, as outlined in Section 4.3.1, and is called prior to each iteration of the identification routines.

The individual components of these routines, as in SnapPea, are covered in Section 3.3.1 (step 5). The two "key" identification routines (`check_faces()` and `verify_groups()`) need theoretically minor alteration; having to now skip consideration of those components of the polyhedron related to its hyperideal sides.

Then, with the above classification over the polyhedron, `verify_faces()` can investigate the full stratification of edge types around each side and confirm the numbers in each non-hyperideal class matches across the side-pairings.

If these routines are successful, then it is confirmed that the construction represents a Dirichlet domain, $D(G, 0)$.

4.4.2 Basepoint Restriction

The movement of the basepoint is handled by the `Dirichlet_basepoint.c` routines and is the feature of while-loop one[3]; see Section 3.3.1 (step 6). It is worth noting that the actual basepoint, used to begin the construction, is always centred at 0 in the projective model of hyperbolic space; the group is actually conjugated to affect an artificial movement of the basepoint.

Now that ideal sides are a possibility, the routines used to maximise the injectivity radius of the polyhedron become a concern. Recall that, for the purposes of this construction, the injectivity radius is the hyperbolic distance from the basepoint to the closest side. While in the case of a finite polyhedron there is no issue in this maximisation system, with a hyperideal polyhedron there is the possibility that the lack of a finite side in some direction may allow for the basepoint to be moved towards a limit point on the sphere at infinity. That is, there is a possibility that the injectivity radius will tend to infinity for hyperideal polyhedrons.

[3] As noted in Section 3.3.1, moving the basepoint, and avoiding while-loop one, is an option within the standard `Dirichlet_` routines. We have versions of SnappyD that allow for either option; see Appendix B.

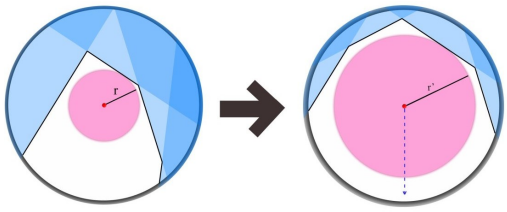


Fig. 4.7: Basepoint issues with hyperideal polyhedrons.

As motion of the basepoint proceeds in a series of small increments, we sum these steps in an attempt to track the overall movement. We consider $\tilde{b}$, the sum of Euclidean movements induced upon the base point, this does not accurately represent the position of the basepoint but does give some indication as to its relative position. This is not restricted to only the movements induced by the basepoint routines, also including any minor resetting of the basepoint that may be affected by a manipulation of the initial generator set. In this way all constructions, even non-maximised, will show some form of basepoint manipulation.

We then allow access to the basepoint routines only while $\|\tilde{b}\|_E^2 \leq 0.7$. If the bound is hit or exceeded, then it is noted in the output file. In these circumstances it may be best to work with, or compare output alongside that of, a non-maximised version of the domain.

While exceeding this bound does not strictly indicate an issue with the construction, it is generally indicative of a case of potential interest, especially if this is seen across multiple generating sets for the same conjugacy class of groups. These results are discussed further in Section 5.2.5.

4.4.3 "Bells and Whistles" Finalisation

These routines are all discussed in Section 3.3.1 (step 7); and, as in SnapPea, if the construction process reaches this point, then the polyhedron is a Dirichlet domain for the group G.

Quotient Mapping

The processes that generate a quotient map on the polyhedron run off a comprehensive edge-pairing routine, based upon the known side-pairings. For sides and vertices, limiting the routine's access only to those components with finite parts works well. But the dependence on a winged edge data structure and the additional complexity of edges, now with finite, retractable or hyperideal natures, makes several extra levels of consideration necessary to the edge-pairing routines.

The standard process for determining the edge-pairings is to look at a side, (double) check that they have the same number of edges and work around them from an easily calculated starting point. The core process in this method is still usable, but requires expansion to take into account the purely hyperideal edges, of which any number can

exist, which arise in retracted sides and that can have no part in the identification and generation of edge classes. Further, as edges may not necessarily have finite, or otherwise comparable, end points we instead have to focus on the individual retraction points of each edge.

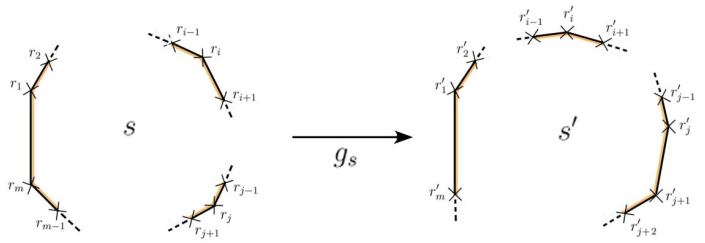


Fig. 4.8: Pairing edges on retracted sides.

This is a much more complicated process; but works well. Recall that all hyperideal components are given 0 measure and the retraction points on hyperideal edges are set to the basepoint. This allows the process to focus on processing the retracted vertices, without possibility of data corruption through the consideration of false points. See Figure 4.8

Further, the data from this process can be accessed in the output script and provides an interesting means of visualising the development of the quotient mapping.

The Bisection of Edges and Sides

With an accurate quotient mapping the bisection of edges and sides, to ensure that the singular set of the orbifold is contained within the vertex structure, is still possible.

The groups we use are all orientation-preserving; this means that edges may need to be bisected as usual, but the only possible side-splitting situation is that linked to an order two rotation about an axis contained in the side. As our generators are of finite-order, sides that map to themselves, and thus require bisection, are not uncommon.

If a side is finite, then the standard bisection routines work. However, if it is a retraction or bounded with only hyperideal edges, then there is the potential for some issues to arise. Namely, in these situations it is entirely possible that the edge structure will be such that no obvious cut is actionable, see B, C, D in Figure 4.9; and compare with Figure 3.11.

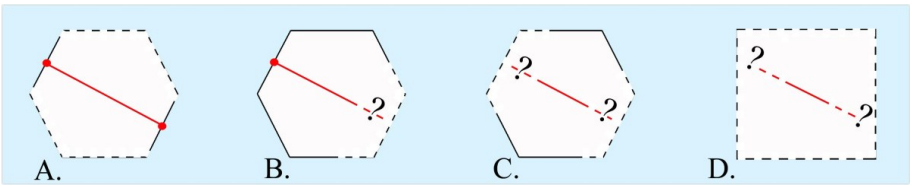


Fig. 4.9: Bisecting the sides of hyperideal polyhedrons.

Generally, determining a non-obvious bisection line should not be too difficult; simply requiring the determination of the associated transformation's axis. But effecting such a bisection is a major alteration to the standard format of the polyhedron's winged edge structure. As these bisections are a largely cosmetic issue in our appreciation of the domains, and do not affect the overall nature of the construction, we skip the bisection of a side if it is of type B, C or D seen in Figure 4.9.

If any bisections are skipped, they are noted in the output file. The only effect of forgoing the bisection of a side, is in the tracking of group data, as noted in Section 4.5.3 below.

Other Data

In following with the rest of these modifications, the final domain data is calculated as per the standard `Dirichlet_` subroutines, utilizing our classification of components to avoid enacting computations on hyperideal components.

The lengths and distances across mapping classes are all calculated by finite components, along with the dihedral angles and singularity orders. But for classification data, we focus solely on the calculation of the volume and deviation; excluding the other geometric classifying constants, like spine-radius and radii, which lie outside the general domain confirmation processes.

This provides the data we are interested in, leaving the other data routines to be adapted and incorporated at a later date.

4.5 Word Tracking

As seen above, the overall program constructs a polyhedron and, if finite, gives its hyperbolic volume. This volume (or lack of volume in the hyperideal case) being our primary interest. But the finalisation routines develop a lot of information that is otherwise unused in the construction.

More specifically, the routines also develop a full quotient mapping over the polyhedron, and determine the singularity orders about each relevant point. This information is related to Poincaré's Theorem and, with a word description of group elements, can provide an additional level of insight into our groups of interest and the construction process used.

4.5.1 Words

Within SnapPea there exist group representation routines that attempt to build a representation from the triangulation of a manifold. These routines are not available, nor applicable, to the `Dirichlet_` subroutines; however, by adding a word generation and tracking system, we can generate a word for each constructed group element, a word for each of the polyhedron's cycle transformations and build some form of group representation.

Let g be any element of G, then we use $w(g)$ to denote the reduced word for g. Words are held in SnappyD using a simple integer array; which, due to storage concerns, we limit to being 126 letters long. This system is built around there being only two generators, but it can very easily be expanded to deal with more. The generators are the only length one words and the identity the only length zero word.

Whenever relevant group elements are manipulated, the corresponding words are also generated. However, if computing $f \circ g$ and

$$length(w(f)) + length(w(g)) > 126,$$

then $w(f \circ g)$ is set to the empty word and an error in word tracking is reported in the output script.

There is no easy way to incorporate the word data-structure into the matrix representation, so instead it is imbedded within the common matrix pair structure used for tracking matrices, which stores a group element alongside its inverse. Where a matrix is stored individually, as in noting side-generating transformations, data structures have been expanded to also hold the word. This keeps relevant information together and ensures both matrix and word representations can be easily referenced.

In this way the word tracking system sits outside of the modified `Dirichlet_` subroutines and in no way interacts with the actual computations. To keep things simple, this word tracking is always in effect; but is only displayed in the output script when desired.

4.5.2 Edge Cycles

With the stored word information we can generate words for each cycle transformation in the finalised domain. This is achieved by looping through the edge class $\{e_i\}$

$$e_i \mapsto e_{i+1} \mapsto ... \mapsto e_{i+m};$$

and noting the side-generating transformations

$$g_i^{-1} \mapsto g_{i+1}^{-1} \mapsto ... \mapsto g_{i+m}^{-1}.$$

This information can then be used to generate the word of the inverse cycle transformation:

$$w(B_{e_i}^{-1}) = w(g_i^{-1} \circ g_{i+1}^{-1} \circ ... \circ g_{i+m}^{-1}) = w(g_i^{-1})w(g_{i+1}^{-1})...w(g_{i+m}^{-1}).$$

Combined with an edge's singularity order n, we can then determine the words for the cycle relations $B_{e_i}^{-n} = I$.

The program avoids direct calculation and storage of these larger words and instead outputs them directly, accompanied by the relevant singularity order. Note that at this point the cycle condition has not been shown to be satisfied; the information relevant

to the checks for this condition are given alongside the display of the singularity order.

4.5.3 Applications and Poincaré's Theorem

As we do not confirm the completeness condition, see Section 4.3.1, and lack a means of classifying infinite edges, we cannot technically use the Poincaré Theorem. However assuming the condition is satisfied we can still look at the data presented critically. In either case the information provided by this tracking of words is of general interest.

For such interests, we have created an output option that works under the assumption of satisfying Poincaré's Theorem and displays each side-generating transformation (generators) and each edge cycle (relations) to provide a pseudo-presentation of the group; while confirming the cycle condition. There are, however, still issues that must be recognised when considering this output:

- The "presentation" may not be complete, with unactionable side bisections being unable to effect the relevant edge cycles. In other words, certain reflection relations may be left out of the list of relations, though this will be noted in the general output.
- There is no in-built method for dealing with the word problem in this program. Further, each variant generator set uses the same letters to represent different elements, so a direct comparison between groups can be especially difficult.

For these reasons we consider this representation output as basic observational data and will not use it in our work in classifying groups. Despite these issues, this still allows for considerable insight into the construction process.

For example, it was through this output that we uncovered the additional complexities that the simplification routines were inducing in the initial generating sets, in response to the basepoint avoiding conjugation highlighted in Section 4.2.1. Under this preliminary conjugation it was not uncommon to have the input generators $\{f, g\}$, being adapted into forms similar to $\{f, f^7gf^{-8}\} = f^7\{f, gf^{-1}\}f^{-7}$. Whereas these adaptations have not been seen to occur for generator sets that have not had this preliminary conjugation applied to them.

5. COMPUTATIONAL GEOMETRY

We have used our program, SnappyD, for the attempted construction of fundamental domains relating to a large number of groups. In this chapter we provide an overview of the related computational results, and their application.

As a preliminary investigation, we give a detailed overview of a well-known set of groups in Section 5.1; where we highlight a number of observable results. In Section 5.2, we detail aspects of the computational nature of our results, highlighting the properties of what might be considered the outliers in our output data. We then summarise the results from all our constructions in Section 5.3 with reference to the classification of arithmetic groups.

Specifically, we use SnappyD to construct fundamental domains for a number of groups derived from, and related to, those discussed in Section 2.4, paying particular interest to the volume of such constructions. These groups are either derived from arithmetic data, and so are known *a priori* to be discrete; or arise as exceptional points in the disc covering procedure, where discreteness may be unconfirmed.

Full output tables, comprising the base data for all successful constructions can be found in Appendix C. A small portion of the results in the appendix are discussed here directly, but they are all referenced with respect to the classification of groups.

Results

The overall process of SnappyD and the general output are described in detail in Chapters 3 and 4. Our particular interest in its application is in determining the covolume of an orbifold given its associated parameter set; or, alternatively, determining that the orbifold is associated with a hyperideal construction (as described in Section 4.3.1), which the system then denotes as "free". These are generally not free groups and will instead be among the types of Kleinian groups listed in Section 2.3.1; this result indicating a demonstration of discreteness where computational results were previously unavailable.

To help ensure the achievement of results we base our construction on multiple computational parameter sets, as described in Section 4.2.2, and attempt these individual constructions over a range of error resolutions (vertex epsilon, see Section 3.2.3). Assuming a successful construction, this then provides a variety of information regarding the rigidity of the construction and its comparative complexity.

As the domain generated for any given G, its conjugates and, potentially, its $\mathbb{Z}_2$ ex-

tension should all share a superficially equivalent complexity, we do not undertake a rigorous investigation of the individual constructions and avoid a full description of the related calculations. Instead focusing on the overall group set G.

We consider the volume as our end result, qualifying it by deviation in the constructed polyhedron. Recall that the deviation is a measure of the distance a side-generating transformation is from $PSO(1,3)$. These results, when combined with information relevant to the source of the parameter set, then allow us to determine the discreteness and possible arithmeticity of the group.

Discrete Groups

In a general setting, there is no assumption that a given modified parameter set (p, q, γ) corresponds to a discrete group; in which case there can be no assumption of success. But, in the case of subgroups of arithmetic groups, all the parameter sets that we use correspond directly with a discrete, if not arithmetic, group. Through this we know that, ignoring the considerations of error and precision, a fundamental polyhedron can be constructed for each of these groups.

With this knowledge we can attempt the construction of a valid domain, and reaffirm the constructions against other earlier results.

We split our interest between to two general sources, specifically the group data for two-elliptic-generator groups relating to both the arithmetically determinable points and the exceptional points within the "computational boundary" data; both described in Chapter 2 and taken from references [23], [60] and [62].

As these groups arise from discreteness or arithmeticity calculations; the γ-values are each derived from a relevant polynomial. Where possible, we have listed these polynomials in the relevant sections of this chapter. We otherwise reference the groups by parameter; generally considering the parameter set to be representative of the class of equivalent parameter sets as per Section 4.2.2.

5.1 Example: Generalised Triangle Groups

As an initial example we focus on the groups given in Table 1 of [60]. This is a table of data on arithmetic generalised triangle groups, and we presume the few volumes given there are taken from a reference table.

These groups have presentations of the form

$$\langle f, g \mid f^p = g^q = w^r = I \rangle$$

where w is some word in f and g and r is some integer power. They arise from arithmetic computations and have an associated γ-polynomial as given in Table 5.1 below.

We pick this set of groups as [60] consists of a detailed description of them with group

presentations, arithmetic details and a description of each of the associated orbifolds. Additionally they are finite with volumes given for only 6 out of the 21 given groups. Our results allow us to complete the description of covolumes and also provide a more directly insightful output.

The data given in the original table corresponds to 21 groups that occur in seven sets of three, which we refer to as triples. The first entry of each triple is a $(3,3)$-group with the following two entries being $(3,2)$-group, $\mathbb{Z}_2$ extensions; with no additional consideration being given to complex conjugates, as is common in the literature. Here we denote each of the triples by an index n.

n	(p,q)	γ-polynomial	γ	V_0	covolume
1	$(3,3)$	$2+6z+4z^2+z^3$	$-1.77184 \pm 1.11514i$	0.2646	0.264774439883296
	$(3,2)$	$1+7z+5z^2+z^3$	$-2.41964 \pm 0.60629i$	0.1323	0.132387219941707
	$(3,2)$	$2+4z+4z^2+z^3$	$-0.58036 \pm 0.60629i$	0.1323	0.132318149806017
2	$(3,3)$	$2+4z+2z^2+z^3$	$-0.68055 \pm 1.63317i$	0.6616	0.661714937315199
	$(3,2)$	$1+z+3z^2+z^3$	$-0.11535 \pm 0.58974i$	0.3308	0.330857468657719
	$(3,2)$	$2+10z+6z^2+z^3$	$-2.88465 \pm 0.58974i$	0.3308	0.330717107428080
3	$(3,3)$	$1+2z+2z^2+2z^3+z^4$	$0.19927 \pm 1.58951i$	–	0.982766244039437
	$(3,2)$	$1+21z+21z^2+8z^3+z^4$	$-3.13846 \pm 0.48506i$	–	0.492361631010942
	$(3,2)$	$1+3z+3z^2+4z^3+z^4$	$0.13846 \pm 0.48506i$	–	0.491484440296781
4	$(3,3)$	$2+30z+30z^2+24z^3+8z^4+z^5$	$-2.80606 \pm 1.15645i$	–	1.090072770048278
	$(3,2)$	$1+22z+22z^2+18z^3+7z^4+z^5$	$-0.89704 \pm 0.95897i$	–	0.545209773747604
	$(3,2)$	$2+32z+32z^2+24z^3+8z^4+z^5$	$-2.10296 \pm 0.95897i$	–	0.544786544898183
5	$(3,3)$	$2+4z+4z^2+2z^4+z^5$	$0.84236 \pm 1.35530i$	–	1.232257018354058
	$(3,2)$	$2+2z+2z^2+10z^3+6z^4+z^5$	$0.29843 \pm 0.37680i$	–	0.616139999303372
	$(3,2)$	$1+34z+34z^2+28z^3+9z^4+z^5$	$-3.29843 \pm 0.37680i$	–	0.615425771551932
6	$(3,3)$	$1+14z+51z^2+66z^3+38z^4+10z^5+z^6$	$-3.19690 \pm 0.90182i$	–	1.696240561334381
	$(3,2)$	$1+30z+79z^2+80z^3+40z^4+10z^5+z^6$	$-1.92469 \pm 1.06173i$	–	0.847936929164838
	$(3,2)$	$1+12z+34z^2+40z^3+25z^4+8z^5+z^6$	$-1.07531 \pm 1.06173i$	–	0.848229498305300
7	$(3,3)$	$2+10z+8z^2+2z^3+2z^4+z^5$	$0.68088 \pm 1.73330i$	–	2.446924195119881
	$(3,2)$	$1+z6z^3+5z^4+z^5$	$0.27988 \pm 0.48692i$	–	1.223512318246351
	$(3,2)$	$2+28z+54z^2+36z^3+10z^4+z^5$	$-3.27988 \pm 0.48692i$	–	1.221755633824927

Tab. 5.1: Generalised triangle groups.

We provide an updated table in Table 5.1; listing the previous covolume, V_0, where available, and our new calculated covolume. This table does not list the individual volume for each of the various computational parameter sets, instead displaying the volume associated with the lowest deviation for each conjugate pair. As with all tables in this thesis, we only give the γ-value to 5 decimal places, whereas our general computations are undertaken at 15 decimal places, to fit with the double precision used in the program.

5.1.1 Output Tables

As this is an initial example, here we give the summary output information related to the fundamental polyhedrons constructed for these groups.

In Table 5.2 we give the output of SnappyD, summarised by:

- n, the set of groups under consideration.
- (p,q), the respective order of generators input into the program.
- γ, the γ-parameter used, reduced to 5 decimal places.

- ε_V, is the band of vertex resolution k-values for which the construction holds; each resolution setting is of the form 10^{-k}, for integer k. Note that we only test within the bounds suggested by SnapPea: $3 \leq k \leq 15$.
- deviation, the largest deviation of a side-generating transformation in the construction.
- B_0, the base point in $\mathbb{R}^3$, moved via conjugation under the "maximise injectivity radius" routines.
- $||B_0||_E^2$, the square of the Euclidean distance that the basepoint has been moved (to 2 d.p.).
- $S/V/E$, S being the total number of sides, V the vertices and E the edges, that the final polyhedron has. Note that in the general situation, $S/V/E$ only considers those sides, vertices and edges that are either finite or retractable; in this case, however, there are only finite sides, vertices and edges.
- covolume, the dominant hyperbolic volume of the particular construction.

The larger summary tables given in the Appendix C follow an identical structure to that given here.

For this example, in Table 5.3, we also give the summary of output information for the `nCPT_NM` variant of SnappyD, see Appendix B. This variant does not maximise the injectivity radius, so offers a comparison of basepoint effects. Where applicable, this table also fits the format given above.

5.1.2 Variance and Rigidity

The results displayed in Table 5.1 are consistent with those seen elsewhere and, aside from slight variations due to precision, fit well with the general theory. However, one of the more interesting results, shown in both Tables 5.2 and 5.3, is the rigidity of these constructions compared against this slight variance in the individual results.

Notably, for each computational parameter set, if a polyhedral construction holds for a value of ε_V, then any successful construction from the same parameter set gives the same polyhedral form - basepoint, $S/V/E$ counts and volume all remain unchanged. The one exception to this are the data tremors mentioned below and discussed in Section 5.2.3. Against this, the same repetition of form is generally not seen across the computational parameter sets within a parameter class; outside of that affected by the basepoint routines.

The effects of moving the basepoint are highlighted in comparing Tables 5.2 and 5.3. Firstly, of primary interest in relation to the focus of our calculations, maximising the injectivity radius generally decreases the spread in volumes over a set of equivalent parameter sets by an order of magnitude or more. And further, as might be expected, the basepoint routines have a homogenising effect on the various polyhedral forms, as expressed through the side, edge and vertex counts.

n	(p,q)	γ	ε_V	deviation	B	$\|\|B\|\|_E^2$	S/V/E	covolume
1	(3,3)	$-1.77184 + 1.11514i$	$3 \leftrightarrow 13$	2.55×10^{-15}	$(-0.04, 0.31, 0.21)$	0.14	10/7/15	0.264774439883296
		$-1.77184 - 1.11514i$	$4 \leftrightarrow 14$	5.33×10^{-15}	$(0.24, 0.31, 0.17)$	0.18	14/12/24	0.264774439883378
	(3,2)	$-2.41964 + 0.60629i$	$3 \leftrightarrow 14$	2.22×10^{-15}	$(0.19, 0.25, 0.14)$	0.12	14/12/24	0.132387219941707
		$-2.41964 - 0.60629i$	$4 \leftrightarrow 14$	2.44×10^{-15}	$(0.25, 0.14, 0.22)$	0.13	8/9/15	0.132387219941693
		$-0.58036 + 0.60629i$	$4 \leftrightarrow 13$	1.73×10^{-14}	$(-0.17, 0.27, -0.02)$	0.10	22/24/44	0.132318149806017
		$-0.58036 - 0.60629i$	$4 \leftrightarrow 13$	2.98×10^{-14}	$(-0.17, 0.25, 0.11)$	0.10	22/24/44	0.132403276354546
	(2,3)	$-2.41964 + 0.60629i$	$3 \leftrightarrow 14$	2.44×10^{-15}	$(0.14, 0.21, -0.14)$	0.08	14/12/24	0.132387219941619
		$-2.41964 - 0.60629i$	$3 \leftrightarrow 14$	2.66×10^{-15}	$(0.14, 0.21, 0.14)$	0.08	14/12/24	0.132387219941758
		$-0.58036 + 0.60629i$	$5 \leftrightarrow 12$	2.30×10^{-14}	$(0.05, 0.21, 0.24)$	0.10	22/24/44	0.132426403921074
		$-0.58036 - 0.60629i$	$5 \leftrightarrow 12$	2.41×10^{-14}	$(0.15, 0.19, 0.15)$	0.08	22/24/44	0.132158766596323
2	(3,3)	$-0.68055 + 1.63317i$	$5 \leftrightarrow 13$	1.53×10^{-14}	$(0.19, 0.45, 0.19)$	0.27	16/14/28	0.661714937315159
		$-0.68055 - 1.63317i$	$5 \leftrightarrow 13$	1.42×10^{-14}	$(0.34, 0.40, 0.24)$	0.34	16/14/28	0.661714937315199
	(3,2)	$-0.11535 - 0.58974i$	$6 \leftrightarrow 13$	3.17×10^{-13}	$(-0.11, 0.30, 0.19)$	0.14	24/32/54	0.330857468657992
		$-0.11535 + 0.58974i$	$4 \leftrightarrow 13$	2.13×10^{-14}	$(-0.11, 0.35, -0.04)$	0.14	24/32/54	0.330857468657719
		$-2.88465 - 0.58974i$	$4 \leftrightarrow 14$	2.71×10^{-14}	$(0.33, 0.03, 0.36)$	0.24	22/24/44	0.330802402206134
		$-2.88465 + 0.58974i$	$4 \leftrightarrow 14$	5.02×10^{-14}	$(0.33, 0.29, 0.22)$	0.24	22/24/44	0.330264970254204
	(2,3)	$-0.11535 - 0.58974i$	$4 \leftrightarrow 12$	5.85×10^{-14}	$(0.30, 0.24, 0.19)$	0.18	24/32/54	0.330857468657576
		$-0.11535 + 0.58974i$	$4 \leftrightarrow 12$	2.89×10^{-14}	$(0.21, 0.26, 0.26)$	0.18	24/32/54	0.330857468657535
		$-2.88465 - 0.58974i$	$4 \leftrightarrow 13$	2.89×10^{-14}	$(0.29, 0.22, 0.23)$	0.18	22/24/44	0.330788586067824
		$-2.88465 + 0.58974i$	$5 \leftrightarrow 13$	1.47×10^{-14}	$(-0.04, 0.08, 0.40)$	0.17	22/24/44	0.330717107428080
3	(3,3)	$0.19927 + 1.58951i$	$5 \leftrightarrow 11$	1.35×10^{-14}	$(0.31, 0.51, 0.14)$	0.38	20/24/42	0.982766244039437
		$0.19927 - 1.58951i$	$6 \leftrightarrow 13$	1.97×10^{-13}	$(0.42, 0.39, 0.33)$	0.44	20/24/42	0.982766244039443
	(3,2)	$-3.13846 + 0.48506i$	$4 \leftrightarrow 14$	9.77×10^{-15}	$(-0.28, 0.09, 0.39)$	0.24	18/18/34	0.492361631010875
		$-3.13846 - 0.48506i$	$4 \leftrightarrow 14$	9.33×10^{-15}	$(-0.28, 0.28, 0.29)$	0.24	18/18/34	0.492361631011032
		$0.13846 + 0.48506i$	$4 \leftrightarrow 13$	6.95×10^{-14}	$(-0.08, 0.39, -0.05)$	0.16	24/32/54	0.491693878116127
		$0.13846 - 0.48506i$	$4 \leftrightarrow 13$	6.14×10^{-14}	$(-0.08, 0.31, 0.24)$	0.16	24/32/54	0.491677333327298
	(2,3)	$-3.13846 + 0.48506i$	$4 \leftrightarrow 13$	7.44×10^{-15}	$(0.07, 0.06, 0.36)$	0.14	18/18/34	0.492361631010930
		$-3.13846 - 0.48506i$	$4 \leftrightarrow 13$	3.55×10^{-15}	$(0.34, 0.27, 0.23)$	0.24	18/18/34	0.492361631010942
		$0.13846 + 0.48506i$	$4 \leftrightarrow 12$	9.04×10^{-14}	$(-0.26, 0.21, 0.26)$	0.18	24/32/54	0.491677333327583
		$0.13846 - 0.48506i$	$4 \leftrightarrow 13$	4.00×10^{-14}	$(-0.19, 0.33, 0.20)$	0.18	24/32/54	0.491484440296781
4	(3,3)	$-2.80606 + 1.15645i$	$4 \leftrightarrow 13$	1.61×10^{-14}	$(0.31, 0.41, 0.35)$	0.38	18/18/34	1.090072770048188
		$-2.80606 - 1.15645i$	$6 \leftrightarrow 12$	8.88×10^{-15}	$(0.58, 0.24, 0.46)$	0.61	18/23/39	1.090072770048278
	(3,2)	$-0.89704 + 0.95897i$	$5 \leftrightarrow 13$	7.66×10^{-15}	$(0.14, 0.43, 0.06)$	0.21	18/16/32	0.544958718160841
		$-0.89704 - 0.95897i$	$5 \leftrightarrow 13$	1.73×10^{-14}	$(0.16, 0.43, -0.04)$	0.21	18/16/32	0.544962897261796
		$-2.10296 + 0.95897i$	$4 \leftrightarrow 13$	1.38×10^{-14}	$(0.15, 0.37, 0.13)$	0.18	18/17/33	0.545036385024126
		$-2.10296 - 0.95897i$	$5 \leftrightarrow 14$	7.11×10^{-15}	$(0.45, 0.30, 0.28)$	0.38	18/17/33	0.544739451493277
	(2,3)	$-0.89704 + 0.95897i$	$3 \leftrightarrow 14$	3.33×10^{-15}	$(0.08, 0.34, 0.05)$	0.13	18/16/32	0.545209773747604
		$-0.89704 - 0.95897i$	$5 \leftrightarrow 14$	6.55×10^{-15}	$(0.12, 0.35, 0.15)$	0.16	18/16/32	0.544974108710528
		$-2.10210 + 0.95897i$	$5 \leftrightarrow 13$	6.44×10^{-15}	$(0.23, 0.07, 0.37)$	0.19	18/17/33	0.544786544898183
		$-2.10210 - 0.95897i$	$4 \leftrightarrow 13$	8.22×10^{-15}	$(0.19, 0.33, 0.08)$	0.15	18/17/33	0.545036385024131
5	(3,3)	$0.84236 + 1.35530i$	$4 \leftrightarrow 12$	4.85×10^{-14}	$(0.37, 0.54, 0.10)$	0.45	20/26/44	1.232257018353899
		$0.84236 - 1.35530i$	$4 \leftrightarrow 12$	3.62×10^{-14}	$(0.45, 0.37, 0.39)$	0.50	20/26/44	1.232257018354058
	(3,2)	$0.29843 + 0.37680i$	$4 \leftrightarrow 11$	3.21×10^{-13}	$(-0.07, 0.41, -0.06)$	0.17	26/36/60	0.616139999305901
		$0.29843 - 0.37680i$	$4 \leftrightarrow 13$	1.71×10^{-13}	$(-0.07, 0.31, 0.27)$	0.17	26/36/60	0.616139999304282
		$-3.29843 + 0.37680i$	$5 \leftrightarrow 13$	2.36×10^{-13}	$(-0.30, 0.09, 0.44)$	0.29	24/26/48	0.616097893053077
		$-3.29843 - 0.37680i$	$5 \leftrightarrow 13$	1.45×10^{-14}	$(-0.31, 0.32, 0.32)$	0.30	24/26/48	0.615425771551932
	(2,3)	$0.29843 + 0.37680i$	$4 \leftrightarrow 12$	7.59×10^{-14}	$(-0.23, 0.24, 0.26)$	0.18	26/36/60	0.616139999303372
		$0.29843 - 0.37680i$	$3 \leftrightarrow 12$	1.12×10^{-13}	$(-0.18, 0.33, 0.22)$	0.19	26/36/60	0.616024211234735
		$-3.29843 + 0.37680i$	$5 \leftrightarrow 14$	4.62×10^{-14}	$(0.14, 0.05, 0.40)$	0.18	24/26/48	0.616128509177734
		$-3.29843 - 0.37680i$	$3 \leftrightarrow 13$	1.91×10^{-14}	$(0.40, 0.29, 0.29)$	0.33	24/26/48	0.616128509177068
6	(3,3)	$-3.19690 + 0.90182i$	$6 \leftrightarrow 11$	2.19×10^{-13}	$(-0.33, 0.27, 0.54)$	0.47	20/20/38	1.696240561334349
		$-3.19690 - 0.90182i$	$5 \leftrightarrow 13$	6.10×10^{-14}	$(-0.33, 0.28, 0.54)$	0.48	20/20/38	1.696240561334381
	(3,2)	$-1.92469 + 1.06173i$	$4 \leftrightarrow 13$	5.00×10^{-14}	$(0.14, 0.39, 0.13)$	0.19	22/24/44	0.848365358302074
		$-1.92469 - 1.06173i$	$4 \leftrightarrow 12$	5.66×10^{-15}	$(0.55, 0.21, 0.38)$	0.49	22/26/46	0.847936929164838
		$-1.07530 + 1.06173i$	$4 \leftrightarrow 13$	2.29×10^{-13}	$(0.20, 0.45, 0.07)$	0.25	26/28/52	0.847904607792574
		$-1.07530 - 1.06173i$	$5 \leftrightarrow 14$	2.99×10^{-13}	$(0.27, 0.44, 0.13)$	0.28	26/28/52	0.847904607792901
	(2,3)	$-1.92469 + 1.06173i$	$4 \leftrightarrow 14$	8.44×10^{-15}	$(0.16, 0.12, 0.37)$	0.18	20/20/38	0.847979373190956
		$-1.92469 - 1.06173i$	$3 \leftrightarrow 13$	5.46×10^{-14}	$(0.19, 0.35, 0.08)$	0.16	22/24/44	0.847921705316328
		$-1.07531 + 1.06173i$	$5 \leftrightarrow 14$	2.16×10^{-13}	$(-0.05, 0.21, 0.40)$	0.21	26/28/52	0.849015226143611
		$-1.07531 - 1.06173i$	$5 \leftrightarrow 14$	1.80×10^{-13}	$(0.19, 0.37, 0.15)$	0.20	26/28/52	0.848229498305300
7	(3,3)	$0.68088 + 1.73330i$	$5 \leftrightarrow 12$	3.06×10^{-13}	$(0.57, 0.60, 0.10)$	0.69	24/30/52	2.446924195119881
		$0.68088 - 1.73330i$	$5 \leftrightarrow 13$	8.67×10^{-13}	$(0.68, 0.40, 0.42)$	0.80	24/30/52	2.446924195121926
	(3,2)	$0.27988 + 0.48692i$	$4 \leftrightarrow 12$	4.98×10^{-13}	$(0.16, 0.50, -0.04)$	0.28	26/34/58	1.222531911485409
		$0.27988 - 0.48692i$	$4 \leftrightarrow 12$	2.95×10^{-13}	$(0.16, 0.50, 0.03)$	0.28	26/34/58	1.222412331605630
		$-3.27988 + 0.48692i$	$6 \leftrightarrow 12$	2.07×10^{-12}	$(-0.55, 0.07, 0.47)$	0.53	28/30/56	1.220360017152392
		$-3.27988 - 0.48692i$	$5 \leftrightarrow 13$	1.64×10^{-12}	$(-0.32, 0.32, 0.35)$	0.33	28/30/56	1.221755633824927
	(2,3)	$0.27988 + 0.48692i$	$5 \leftrightarrow 12$	7.88×10^{-13}	$(0.10, 0.40, 0.13)$	0.19	26/36/60	1.221708487472495
		$0.27988 - 0.48692i$	$4 \leftrightarrow 12$	2.44×10^{-13}	$(0.12, 0.40, 0.18)$	0.21	26/34/58	1.223512318246351
		$-3.27988 + 0.48692i$	$4 \leftrightarrow 13$	1.71×10^{-12}	$(0.14, 0.07, 0.43)$	0.21	28/30/56	1.221624951511321
		$-3.27988 - 0.48692i$	$5 \leftrightarrow 13$	1.09×10^{-12}	$(0.48, 0.32, 0.53)$	0.61	28/30/56	1.220360017153322

Tab. 5.2: Summary of the SnappyD output data used to build Table 5.1.

n	(p,q)	γ	ε_V	deviation	S/V/E	covolume
1	(3, 3)	$-1.77184 + 1.11514i$	$3 \leftrightarrow 13$	1.55×10^{-15}	14/12/24	0.264687258680185
		$-1.77184 - 1.11514i$	$4 \leftrightarrow 14$	5.11×10^{-15}	16/16/30	0.264686441306019
	(3, 2)	$-2.41964 + 0.60629i$	$3 \leftrightarrow 14$	2.44×10^{-15}	14/12/24	0.132129725818503
		$-2.41964 - 0.60629i$	$4 \leftrightarrow 14$	2.44×10^{-15}	18/18/34	0.132128103188614
		$-0.58036 + 0.60629i$	$4 \leftrightarrow 13$	1.35×10^{-14}	20/24/42	0.132432240182155
		$-0.58036 - 0.60629i$	$4 \leftrightarrow 13$	8.66×10^{-15}	24/32/54	0.132131747290004
	(2, 3)	$-2.41964 + 0.60629i$	$3 \leftrightarrow 14$	9.99×10^{-16}	14/12/24	0.132387219941671
		$-2.41964 - 0.60629i$	$3 \leftrightarrow 14$	1.78×10^{-15}	16/16/30	0.132322895571381
		$-0.58036 + 0.60629i$	$5 \leftrightarrow 12$	1.42×10^{-14}	24/32/54	0.131911327267540
		$-0.58036 - 0.60629i$	$5 \leftrightarrow 12$	3.03×10^{-14}	20/24/42	0.132060396011615
2	(3, 3)	$-0.68055 + 1.63317i$	$5 \leftrightarrow 13$	2.15×10^{-14}	20/24/42	0.661552728945180
		$-0.68055 - 1.63317i$	$5 \leftrightarrow 14$	1.29×10^{-14}	20/24/42	0.661069296768815
	(3, 2)	$-0.11535 - 0.58974i$	$6 \leftrightarrow 13$	2.60×10^{-14}	24/32/54	0.330777803760144
		$-0.11535 + 0.58974i$	$4 \leftrightarrow 13$	1.89×10^{-14}	22/28/48	0.330777690867818
		$-2.88465 - 0.58974i$	$4 \leftrightarrow 14$	6.02×10^{-14}	32/42/72	0.330927423235746
		$-2.88465 + 0.58974i$	$4 \leftrightarrow 14$	4.49×10^{-14}	30/40/68	0.329017346148485
		$-0.11535 - 0.58974i$	$4 \leftrightarrow 12$	4.07×10^{-14}	22/28/48	0.330051212280796
		$-0.11535 + 0.58974i$	$4 \leftrightarrow 12$	2.14×10^{-14}	22/28/48	0.330312643854552
		$-2.88465 - 0.58974i$	$4 \leftrightarrow 13$	2.75×10^{-14}	24/28/50	0.331565307952308
		$-2.88465 + 0.58974i$	$5 \leftrightarrow 15$	6.13×10^{-14}	28/34/60	0.330751336927118
3	(3, 3)	$0.19927 + 1.58951i$	$5 \leftrightarrow 11$	2.89×10^{-14}	26/36/60	0.984599506000193
		$0.19927 - 1.58951i$	$6 \leftrightarrow 14$	5.66×10^{-14}	22/28/48	0.987652431633644
	(3, 2)	$-3.13846 + 0.48506i$	$4 \leftrightarrow 14$	2.11×10^{-14}	28/34/60	0.491997114458778
		$-3.13846 - 0.48506i$	$4 \leftrightarrow 14$	1.40×10^{-14}	30/38/66	0.491563198867146
		$0.13846 + 0.48506i$	$4 \leftrightarrow 13$	5.68×10^{-14}	24/34/56	0.491747854413174
		$0.13846 - 0.48506i$	$4 \leftrightarrow 13$	5.87×10^{-14}	24/32/54	0.490133650431142
	(2, 3)	$-3.13846 + 0.48506i$	$4 \leftrightarrow 13$	3.44×10^{-15}	24/28/50	0.492447868444507
		$-3.13846 - 0.48506i$	$4 \leftrightarrow 13$	7.99×10^{-15}	32/42/72	0.493054678033088
		$0.13846 + 0.48506i$	$4 \leftrightarrow 12$	5.48×10^{-14}	24/32/54	0.491574570296174
		$0.13846 - 0.48506i$	$4 \leftrightarrow 13$	2.42×10^{-14}	24/32/54	0.489965849808521
4	(3, 3)	$-2.80606 + 1.15645i$	$4 \leftrightarrow 13$	8.38×10^{-14}	32/46/76	1.089918280135662
		$-2.80606 - 1.15645i$	$6 \leftrightarrow 12$	8.66×10^{-14}	32/48/78	1.088779735890397
	(3, 2)	$-0.89704 + 0.95897i$	$3 \leftrightarrow 13$	6.77×10^{-15}	20/22/40	0.544426169529836
		$-0.89704 - 0.95897i$	$3 \leftrightarrow 13$	1.82×10^{-14}	20/22/40	0.544922154168940
		$-2.10296 + 0.95897i$	$4 \leftrightarrow 13$	6.37×10^{-14}	28/36/62	0.547169909744077
		$-2.10296 - 0.95897i$	$4 \leftrightarrow 14$	1.06×10^{-13}	24/28/50	0.547057918815104
	(2, 3)	$-0.89704 + 0.95897i$	$3 \leftrightarrow 14$	2.33×10^{-15}	18/16/32	0.544172613086960
		$-0.89704 - 0.95897i$	$3 \leftrightarrow 14$	5.66×10^{-15}	20/20/38	0.545351262916556
		$-2.10296 + 0.95897i$	$4 \leftrightarrow 14$	7.17×10^{-14}	28/32/58	0.544259874629925
		$-2.10296 - 0.95897i$	$4 \leftrightarrow 13$	4.53×10^{-14}	26/28/52	0.544484565914210
5	(3, 3)	$0.84236 + 1.35530i$	$4 \leftrightarrow 12$	2.73×10^{-14}	24/34/56	1.232133992072095
		$0.84236 - 1.35530i$	$4 \leftrightarrow 12$	2.56×10^{-14}	20/24/42	1.228000819263353
	(3, 2)	$0.29843 + 0.37680i$	$4 \leftrightarrow 11$	2.52×10^{-13}	26/38/62	0.614057325941823
		$0.29843 - 0.37680i$	$4 \leftrightarrow 13$	1.45×10^{-13}	26/36/60	0.615400480567670
		$-3.29843 + 0.37680i$	$5 \leftrightarrow 14$	2.61×10^{-13}	34/44/76	0.615955840645185
		$-3.29843 - 0.37680i$	$5 \leftrightarrow 13$	4.22×10^{-14}	34/44/76	0.616826022797327
	(2, 3)	$0.29843 + 0.37680i$	$4 \leftrightarrow 12$	6.58×10^{-14}	26/36/60	0.615547856614611
		$0.29843 - 0.37680i$	$3 \leftrightarrow 12$	7.03×10^{-14}	26/36/60	0.616354916916405
		$-3.29843 + 0.37680i$	$6 \leftrightarrow 14$	9.19×10^{-14}	36/46/80	0.616620247901536
		$-3.29843 - 0.37680i$	$4 \leftrightarrow 13$	3.04×10^{-14}	30/38/66	0.617052387498835
6	(3, 3)	$-3.19690 + 0.90182i$	$5 \leftrightarrow 11$	1.47×10^{-13}	32/46/76	1.700543453502957
		$-3.19690 - 0.90182i$	$5 \leftrightarrow 13$	1.09×10^{-13}	32/46/76	1.699810630880004
	(3, 2)	$-1.92469 + 1.06173i$	$4 \leftrightarrow 14$	5.75×10^{-14}	26/34/58	0.850210625627002
		$-1.92469 - 1.06173i$	$5 \leftrightarrow 12$	9.33×10^{-15}	22/26/46	0.850328904734173
		$-1.07531 + 1.06173i$	$4 \leftrightarrow 14$	2.27×10^{-13}	28/32/58	0.849073425906248
		$-1.07531 - 1.06173i$	$5 \leftrightarrow 14$	4.16×10^{-13}	28/34/60	0.848298916169056
	(2, 3)	$-1.92469 + 1.06173i$	$4 \leftrightarrow 14$	2.66×10^{-14}	24/26/48	0.849584527813553
		$-1.92469 - 1.06173i$	$3 \leftrightarrow 13$	2.31×10^{-14}	20/20/38	0.850777178404936
		$-1.07531 + 1.06173i$	$5 \leftrightarrow 14$	2.58×10^{-13}	28/30/56	0.848157500807782
		$-1.07531 - 1.06173i$	$5 \leftrightarrow 14$	1.43×10^{-13}	28/32/58	0.849911973939790
7	(3, 3)	$0.68088 + 1.73330i$	$5 \leftrightarrow 12$	5.68×10^{-13}	28/38/64	2.447545319087347
		$0.68088 - 1.73330i$	$5 \leftrightarrow 14$	1.24×10^{-12}	28/36/62	2.440210655861061
	(3, 2)	$0.27988 + 0.48692i$	$4 \leftrightarrow 12$	3.45×10^{-13}	24/34/56	1.223947900719701
		$0.27988 - 0.48692i$	$4 \leftrightarrow 12$	2.17×10^{-13}	26/36/60	1.226976667567070
		$-3.27988 + 0.48692i$	$6 \leftrightarrow 14$	6.49×10^{-12}	42/58/98	1.225912173877403
		$-3.27988 - 0.48692i$	$5 \leftrightarrow 14$	3.96×10^{-12}	46/64/108	1.223013782735435
	(2, 3)	$0.27988 + 0.48692i$	$5 \leftrightarrow 12$	3.87×10^{-13}	28/40/66	1.223300684685046
		$0.27988 - 0.48692i$	$4 \leftrightarrow 12$	1.42×10^{-13}	28/40/66	1.225205491656666
		$-3.27988 + 0.48692i$	$4 \leftrightarrow 13$	3.01×10^{-12}	36/46/80	1.223621083476920
		$-3.27988 - 0.48692i$	$5 \leftrightarrow 13$	1.89×10^{-12}	44/62/104	1.223864938465404

Tab. 5.3: Non-maximised output for the data in Table 5.2

We highlight these effects in Table 5.4; giving, for each set of parameters, $sp(Vol)$, the spread of output volumes, and n(S/V/E), the number of distinct S/V/E counts; for both the maximised (CPT) and unmaximised (nCTP_NM) output polyhedrons.

n	(p,q)	nCPT_NM		CPT	
		n(S/V/E)	$sp(Vol)$	n(S/V/E)	$sp(Vol)$
1	$(3,3)$ $(3,2)$, $(2,3)$	5	8.2×10^{-7} 5.2×10^{-4}	4	8.2×10^{-14} 2.3×10^{-4}
2	$(3,3)$ $(3,2)$, $(2,3)$	7	4.8×10^{-4} 2.5×10^{-3}	3	4.0×10^{-14} 5.9×10^{-4}
3	$(3,3)$ $(3,2)$, $(2,3)$	8	3.1×10^{-3} 3.1×10^{-3}	3	6.0×10^{-15} 8.8×10^{-4}
4	$(3,3)$ $(3,2)$, $(2,3)$	8	1.1×10^{-3} 3.0×10^{-3}	4	9.0×10^{-14} 4.7×10^{-4}
5	$(3,3)$ $(3,2)$, $(2,3)$	7	4.1×10^{-3} 3.0×10^{-3}	5	8.4×10^{-13} 7.1×10^{-4}
6	$(3,3)$ $(3,2)$, $(2,3)$	8	7.3×10^{-4} 2.6×10^{-3}	4	3.2×10^{-14} 1.1×10^{-3}
7	$(3,3)$ $(3,2)$, $(2,3)$	9	7.3×10^{-3} 4.0×10^{-3}	4	2.0×10^{-12} 3.2×10^{-3}

Tab. 5.4: The effect of maximising the basepoint.

Presumably, if the basepoint maximisation routines could continue indefinitely, did not require linearisation of functions or were working in a system with higher accuracy, then we would see greater homogeneity in the basepoint distance, volume and $S/V/E$ results. The basepoint routines also effect the other data used to summarise the output; specifically, for the 70 sets of constructions above, basepoint maximisation leads to 41 increases and only 14 decreases in the size of the ε_V-bands; whereas, due to increased matrix manipulation it results in 31 decreases to 38 increases in the polyhedron deviations.

Another interesting observation is variance in the deviation results and the ε_V-bands under which the constructions hold. Other than all constructions holding under a common vertex epsilon subinterval of $10^{-6} > \varepsilon_V > 10^{-11}$ and deviations generally remaining in the same order of magnitude, there are no obvious relationships between these results within a parameter class; although structural considerations encourage the supposition of one.

5.1.3 Tremors in Covolume and Deviation

For the sake of clarity and space, in building the above tables, we have ignored one particular form of outlier that is seen in our output. Specifically, the differing output values that may occur across the successful ε_V-values relating to a single input parameter set.

In doing this we have removed subtle variations in the results experienced in 2 of the 70 listed computational parameter sets. For the standard SnappyD output, these results are visible in the greater appendix tables; but, for the sake of completeness in this example, we also give them here in Table 5.5.

Note that in Table 5.5, as with the tables seen in the appendix, we list the overall

n	(p,q)	γ	ε_V	deviation	B	$\|\|B\|\|_E^2$	S/E/V	covolume
5	$(3,3)$	$0.84236+1.35530i$	$4\leftrightarrow 12$	4.85×10^{-14}	$(0.37,0.54,0.10)$	0.45	20/26/44	1.232257018353899
		$0.84236-1.35530i$	$4\leftrightarrow 12$	3.62×10^{-14}	$(0.45,0.37,0.39)$	0.50	20/26/44	1.232257018354058
	$(3,2)$	$0.29843+0.37680i$	$4\leftrightarrow 11$	3.21×10^{-13}	$(-0.07,0.41,-0.06)$	0.17	26/36/60	0.616139999305901
		$0.29843-0.37680i$	$4\leftrightarrow 13$	1.71×10^{-13}	$(-0.07,0.31,0.27)$	0.17	26/36/60	0.616139999304282
		$-3.29843+0.37680i$	$5\leftrightarrow 13$	2.36×10^{-13}	$(-0.30,0.09,0.44)$	0.29	24/26/48	0.616097893053077
		$-3.29843-0.37680i$	$5\leftrightarrow 13$	1.45×10^{-14}	$(-0.31,0.32,0.32)$	0.30	24/26/48	0.615425771551932
	$(2,3)$	$0.29843+0.37680i$	$4\leftrightarrow 12$	7.59×10^{-14}	$(-0.23,0.24,0.26)$	0.18	26/36/60	0.616139999303372
		$0.29843-0.37680i$	$3\leftrightarrow 12$	1.12×10^{-13}	$(-0.18,0.33,0.22)$	0.19	26/36/60	0.616024211234735
		$-3.29843+0.37680i$	$5\leftrightarrow 14$ 5	4.62×10^{-14} 4.86×10^{-14}	$(0.14,0.05,0.40)$	0.18	24/26/48	0.616128509177734 0.616128509177205
		$-3.29843-0.37680i$	$3\leftrightarrow 13$	1.91×10^{-14}	$(0.40,0.29,0.29)$	0.33	24/26/48	0.616128509177068
2	$(3,3)$	$-0.68055+1.63317i$	$5\leftrightarrow 13$	2.15×10^{-14}	-	-	20/24/42	0.661552728945180
		$-0.68055-1.63317i$	$5\leftrightarrow 14$	1.29×10^{-14}	-	-	20/24/42	0.661069296768815
	$(3,2)$	$-0.11535-0.58974i$	$6\leftrightarrow 13$	2.60×10^{-14}	-	-	24/32/54	0.330777803760144
		$-0.11535+0.58974i$	$4\leftrightarrow 13$	1.89×10^{-14}	-	-	22/28/48	0.330777690867818
		$-2.88465-0.58974i$	$4\leftrightarrow 14$	6.02×10^{-14}	-	-	32/42/72	0.330927423235746
		$-2.88465+0.58974i$	$4\leftrightarrow 14$	4.49×10^{-14}	-	-	30/40/68	0.329017346148485
	$(2,3)$	$-0.11535-0.58974i$	$4\leftrightarrow 12$	4.07×10^{-14}	-	-	22/28/48	0.330051212280796
		$-0.11535+0.58974i$	$4\leftrightarrow 12$	2.14×10^{-14}	-	-	22/28/48	0.330312643854552
		$-2.88465-0.58974i$	$4\leftrightarrow 13$	2.75×10^{-14}	-	-	24/28/50	0.331565307952308
		$-2.88465+0.58974i$	$5\leftrightarrow 15$ 15	6.13×10^{-14} 3.91×10^{-14}	-	-	28/34/60	0.330751336927118 0.330751336927133

Tab. 5.5: Full output for parameters 5 of Table 5.2 and 2 of Table 5.3.

construction result in the top line for each computational set, with outliers within this overall result listed below. We call computational outliers of this form (data) tremors. These occur semi-regularly and are discussed further in Section 5.2.3.

5.1.4 Rigidity Under Loss of Precision

As a final part of this example, we also look at the devolution of the construction under a loss of input precision. As mentioned above, our generated output results are taken from an input γ-parameter known to a precision of 15 decimal places. Interestingly, as this precision is reduced the constructed polyhedron is still maintained.

In Table 5.6, we demonstrate this for the $(3,3)$ parameter sets of the $n=1$ triple. Here, as precision is lost the width of successful ε_V values decreases until a precision of 6 decimal places is reached; below which the construction fails.

d.p.	$(3,3,-1.771844506346038+1.115142508039937i)$			$(3,3,-1.771844506346038-1.115142508039937i)$		
	ε_V	deviation	covolume	ε_V	deviation	covolume
15	$3-13$	2.55×10^{15}	0.264774439883296	$4-14$	5.33×10^{15}	0.264774439883378
14	$3-14$	3.22×10^{15}	0.264774439883291	$4-14$	5.77×10^{15}	0.264774439883487
13	$3-12$	5.11×10^{15}	0.264774439882926	$4-12$	6.22×10^{15}	0.264774439882894
12	$3-11$	1.67×10^{15}	0.264774439882821	$4-11$	7.33×10^{15}	0.264774439882191
11	$3-10$	3.44×10^{15}	0.264774439905987	$4-9$	3.66×10^{15}	0.264774439975595
10	$3-8$	2.00×10^{15}	0.264774440241217	$4-8$	6.00×10^{15}	0.264774441376238
9	$3-8$	3.77×10^{15}	0.264774436152756	$4-8$	4.77×10^{15}	0.264774441318465
8	$3-7$	2.00×10^{15}	0.264774458561606	$4-6$	5.33×10^{15}	0.264774430084478
7	$3-6$	1.89×10^{15}	0.264774393214161	$4-6$	4.44×10^{15}	0.264774482652009
6	$3-6$	1.67×10^{15}	0.264775688660788	4	3.55×10^{15}	0.264770068097691

Tab. 5.6: The devolution of the $(3,3,\gamma)$ output as precision descreases.

Notable here is the gradual reduction in the number of agreed decimal places in the output volume of the individual parameter sets; a change which seems to relate linearly to the reduction in precision. But of further interest is the deviations reaction to the loss of precision. While, in this example, the deviation maintains its order of magni-

tude, it also seems to move randomly between increasing and decreasing, while ending on a distinct improvement in both cases. Similar results are seen in the equivalent output of the other groups related to the triple.

For further comparison we provide Table 5.7 which highlights the vertex epsilon bands over which the $\mathbb{Z}_2$ extension related constructions hold.

d.p.	$\gamma = -2.419643377607081 + 0.6062907292071 99i$				$\gamma = -0.580356622392919 + 0.6062907292071 99i$			
	$(3,2,\gamma)$	$(2,3,\gamma)$	$(3,2,\overline{\gamma})$	$(2,3,\overline{\gamma})$	$(3,2,\gamma)$	$(2,3,\gamma)$	$(3,2,\overline{\gamma})$	$(2,3,\overline{\gamma})$
15	$3 \mapsto 14$	$3 \mapsto 14$	$4 \mapsto 14$	$3 \mapsto 14$	$4 \mapsto 13$	$5 \mapsto 12$	$4 \mapsto 13$	$5 \mapsto 12$
14	$3 \mapsto 14$	$3 \mapsto 14$	$4 \mapsto 14$	$3 \mapsto 14$	$4 \mapsto 14$	$5 \mapsto 13$	$4 \mapsto 13$	$5 \mapsto 12$
13	$3 \mapsto 13$	$3 \mapsto 13$	$4 \mapsto 13$	$3 \mapsto 13$	$4 \mapsto 12$	$5 \mapsto 12$	$4 \mapsto 12$	$5 \mapsto 12$
12	$3 \mapsto 13$	$3 \mapsto 12$	$4 \mapsto 12$	$3 \mapsto 13$	$4 \mapsto 11$	$5 \mapsto 11$	$4 \mapsto 11$	$5 \mapsto 11$
11	$3 \mapsto 11$	$3 \mapsto 11$	$4 \mapsto 11$	$3 \mapsto 11$	$4 \mapsto 10$	$5 \mapsto 10$	$4 \mapsto 10$	$5 \mapsto 10$
10	$3 \mapsto 10$	$3 \mapsto 10$	$4 \mapsto 10$	$3 \mapsto 10$	$4 \mapsto 9$	$5 \mapsto 9$	$4 \mapsto 9$	$5 \mapsto 9$
9	$3 \mapsto 8$	$3 \mapsto 8$	$4 \mapsto 8$	$3 \mapsto 8$	$4 \mapsto 7$	$5 \mapsto 8$	$4 \mapsto 7$	$5 \mapsto 8$
8	$3 \mapsto 8$	$3 \mapsto 8$	$4 \mapsto 8$	$3 \mapsto 7$	$4 \mapsto 7$	$5 \mapsto 7$	$4 \mapsto 7$	$5 \mapsto 7$
7	$3 \mapsto 6$	$3 \mapsto 6$	$4 \mapsto 7$	$3 \mapsto 6$	$4 \mapsto 6$	$5 \mapsto 6$	$4 \mapsto 6$	$5 \mapsto 6$
6	$3 \mapsto 5$	$3 \mapsto 5$	$4 \mapsto 5$	$3 \mapsto 5$	4		4	
5	$3 \mapsto 5$	$3 \mapsto 5$	4	$3 \mapsto 4$				
4	3	3						

Tab. 5.7: The devolution of ε_V-bands as precision decreases.

What is most striking in this table is:

- For $\gamma = -2.419643377607081 + 0.606290729207199i$, $(3,2,\gamma)$ and $(2,3,\gamma)$, and $(3,2,\overline{\gamma})$ and $(2,3,\overline{\gamma})$ hold for approximately the same ε_V-bands; but there is no apparent relation between the vertex epsilon bands of (p,q,γ) and $(p,q,\overline{\gamma})$.
- Whereas for $\gamma = -0.580356622392919 + 0.606290729207199i$, there is a notable comparison in the vertex epsilon bands of (p,q,γ) and $(p,q,\overline{\gamma})$; but not between $(3,2,\gamma)$ and $(2,3,\gamma)$, or $(3,2,\overline{\gamma})$ and $(2,3,\overline{\gamma})$.

Due to time and space constraints, we do not similarly investigate other groups though, as noted in Chapter 6, the variations in accuracy (under resolution) in geometric constructions is a potentially interesting area of research.

5.2 Computational Results

We have used all the groups from references [60], [62] and [23] as input for the SnappyD program, the output results for which are given in the tables of Appendix C.

In this section we highlight various aspects noted in the resulting output files, specifically with regard to the outliers within what are generally highly consistent computational results. Considering these we then summarise our actual results in Section 5.3.

We note that the tables given in this section are cropped out of the larger tables of Appendix C, which should be seen for additional reference.

5.2.1 Failures

We begin with Table 5.8, where we provide a list of the group parameter information for which SnappyD was unable to generate a fundamental polyhedron. In this case we give the full input length of the first-quadrant γ-values.

(p, q)	γ
$(3, 2)$	$0.161691418362448 + 0.636709481679928i$
$(3, 2)$	$0.273409138442041 + 0.563821092829119i$
$(3, 2)$	$-1.063559204522075 + 1.164380499099818i$
$(3, 2)$	$-0.822388727866991 + 1.112527792130389i$
$(3, 2)$	$0.681519354760437 + 0.147287943028609i$
$(3, 2)$	$0.571863231390706 + 0.252351616795528i$
$(3, 2)$	$0.531012602661534 + 0.273494396101797i$
$(3, 2)$	$-0.176111557895748 + 0.891627543858532i$
$(3, 2)$	$-1.231365882100651 + 1.181179136574646i$
$(3, 2)$	$-0.943438327815293 + 1.141151187720406i$
$(3, 2)$	$0.597265244764000 + 0.223225966457038i$
$(3, 2)$	$0.476225374351834 + 0.335893920570061i$
$(3, 2)$	$0.295161963992842 + 0.539127458455876i$
$(3, 2)$	$0.325799049368975 + 0.511380200203660i$
$(4, 2)$	$0.815157802391803 + 0.712418803346697i$
$(4, 2)$	$0.581529360457850 + 0.939156034693663i$
$(4, 2)$	$-0.303676609148679 + 1.435949864109956i$
$(4, 2)$	$-0.828352852975573 + 1.576686092327405i$

Tab. 5.8: Construction failures.

The 'total' construction failures for each of these parameter sets is due to a simple and rapid accumulation of numerical error; which faults the construction process in the early cutting routines.

It is notable that each of these points is taken from [62]. Specifically Tables 5.6, 5.7 and 5.11; where they are given as points with no corresponding γ-polynomials. Each of these groups should correspond to arithmetic data but it is not known whether the group is an arithmetic group or a proper subgroup of one; indeed, they may even correspond to actual free groups. We did not have the polynomial reference to re-generate these numbers ourselves, so we could suppose that this is related to some clerical error in the tables, akin to that noted in Appendix Section C.4.

In either case, it is a positive sign that the error in these failures seems consistent. A future expansion in accuracy, or other adaptation to the code, might hopefully remedy any issues in these constructions (if there are any).

5.2.2 Cycle Condition Failures

Further to the construction failures listed in Section 5.2.1, 5 of our groups have one or more parameter sets whose final polyhedron fails to satisfy the cycle condition. We give the output details of these parameter sets in Table 5.9; using an orange box to mark those outputs failing the condition.

It is worth noting that the cycle condition is not tested in SnapPea's `Dirichlet_`

(p,q)	γ	ε_V	deviation	B	$\|\|B\|\|_E^2$	S/V/E	covolume
$(3,2)$	$-1.50000-0.60666i$	$3\leftrightarrow 13$	2.89×10^{-15}	$(-0.03,0.24,0.05)$	0.06	12/9/19	0.077777592962837
	$-1.50000+0.60666i$	$3\leftrightarrow 13$	2.66×10^{-15}	$(-0.03,0.24,-0.05)$	0.06	12/9/19	0.077777592962811
$(2,3)$	$-1.50000-0.60666i$	$4\leftrightarrow 13$	8.66×10^{-15}	$(0.02,0.27,0.10)$	0.08	18/16/32	0.039015451336356
	$-1.50000+0.60666i$	$3\leftrightarrow 14$	failure. possibly generates a finite-sheeted cover				
$(3,2)$	$-0.52842-0.78122i$	$5\leftrightarrow 14$	2.53×10^{-14}	$(0.04,0.34,0.18)$	0.15	22/24/44	0.363085747245742
	$-0.52842+0.78122i$	$4\leftrightarrow 13$	1.83×10^{-14}	$(0.04,0.37,-0.06)$	0.15	22/24/44	0.362805250204902
	$-2.47158-0.78122i$	$4\leftrightarrow 13$	1.50×10^{-15}	$(0.38,0.17,0.29)$	0.26	8/9/15	0.362887228283480
	$-2.47158+0.78122i$	$4\leftrightarrow 13$	3.00×10^{-15}	$(0.30,0.33,0.17)$	0.23	14/12/24	0.362887228283415
$(2,3)$	$-0.52842-0.78122i$	$4\leftrightarrow 13$	2.40×10^{-14}	$(-0.05,0.39,0.21)$	0.20	22/24/44	0.363031557094651
	$-0.52842+0.78122i$	$4\leftrightarrow 13$	2.78×10^{-14}	$(-0.10,0.26,0.18)$	0.11	22/24/44	0.363024421723659
	$-2.47158-0.78122i$	$3\leftrightarrow 13$	2.89×10^{-15}	$(0.28,0.26,0.20)$	0.18	14/12/24	0.362887228283405
	$-2.47158+0.78122i$	$3\leftrightarrow 14$	2.55×10^{-15}	$(0.28,0.26,-0.20)$	0.18	14/12/24	0.362887228283461
$(5,2)$	$0.11803-0.60666i$	$4\leftrightarrow 13$	4.22×10^{-14}	$(-0.27,0.17,0.18)$	0.13	18/16/32	0.039105939852173
	$0.11803+0.60666i$	$4\leftrightarrow 12$ 4	4.64×10^{-14} 4.55×10^{-14}	$(-0.27,0.22,0.11)$	0.13	18/16/32	0.039120554210565 0.039120554210676
	$-1.50000-0.60666i$	$3\leftrightarrow 12$	6.77×10^{-15}	$(-0.13,0.39,0.25)$	0.23	16/16/30	0.234301713689990
	$-1.50000+0.60666i$	$3\leftrightarrow 12$	failure. possibly generates a finite-sheeted cover				
$(2,5)$	$0.11803-0.60666i$	$3\leftrightarrow 13$	2.49×10^{-14}	$(-0.16,0.02,0.15)$	0.05	18/16/32	0.039051905406329
	$0.11803+0.60666i$	$3\leftrightarrow 13$ 3, 4 5	2.49×10^{-14} 1.27×10^{-14} failure.	$(-0.17,0.03,0.14)$	0.05	10/8/16	0.039050285615583 0.039050285615487
	$-1.50000-0.60666i$	$3\leftrightarrow 14$	failure. possibly generates a finite-sheeted cover				
	$-1.50000+0.60666i$	$3\leftrightarrow 14$	failure. possibly generates a finite-sheeted cover				
$(3,2)$	$-0.07087+0.66435i$	$4\leftrightarrow 13$	7.84×10^{-14}	$(0.07,0.42,-0.07)$	0.19	24/32/54	0.595189968898689
	$-0.07087-0.66435i$	$4\leftrightarrow 12$	3.46×10^{-14}	$(0.07,0.42,0.06)$	0.19	24/32/54	0.595189968897844
	$-2.92913+0.66435i$	$5\leftrightarrow 12$	1.00×10^{-12}	$(0.47,0.14,0.26)$	0.30	22/20/40	0.297303264846344
	$-2.92913-0.66435i$	$5\leftrightarrow 13$	5.37×10^{-14}	$(0.39,0.03,0.41)$	0.32	22/24/44	0.594206188118980
$(2,3)$	$-0.07087+0.66435i$	$4\leftrightarrow 13$	1.01×10^{-13}	$(-0.13,0.22,0.30)$	0.15	24/32/54	0.595419291729511
	$-0.07087-0.66435i$	$4\leftrightarrow 13$	8.83×10^{-14}	$(-0.03,0.39,0.24)$	0.21	24/32/54	0.595419291729334
	$-2.92913+0.66435i$	$5\leftrightarrow 13$	2.32×10^{-14}	$(-0.02,0.09,0.46)$	0.22	22/24/44	0.594663573037919
	$-2.92913-0.66435i$	$4\leftrightarrow 13$	8.88×10^{-15}	$(0.36,0.24,0.27)$	0.26	22/24/44	0.594566134203106
$(4,2)$	$-1.00000+1.73205i$	$5\leftrightarrow 13$	4.24×10^{-14}	$(0.32,0.98,-0.01)$	1.06	16/12/28	Free?
	$-1.00000-1.73205i$	$4\leftrightarrow 13$	2.43×10^{-14}	$(0.65,0.66,0.04)$	0.85	16/12/28	Free?
$(2,4)$	$-1.00000+1.73205i$	$3\leftrightarrow 14$	5.86×10^{-14}	$(0.13,0.32,0.95)$	1.01	18/18/36	Free?
	$-1.00000-1.73205i$	$4\leftrightarrow 13$	5.72×10^{-14}	$(0.79,0.35,0.55)$	1.05	18/18/36	Free?

Tab. 5.9: Groups that fail the cycle condition.

subroutines. Which is relevant as the data in this table demonstrates an inconsistency within the results of a nature less stable than the Parallax data described in Section 5.2.4.

Of notable importance with these groups is that, with one exception, each contains at least one construction that passes the cycle condition. And so we can ignore constructions that fail the cycle condition when considering our summary output. Similarly, the failure in each of these groups is caused by an edge cycle $\{e_i\}$ such that

$$2.5\sum_{i=1}^{m}\alpha(e_i)=2\pi.$$

The one exception to this is the sole $(4,2,\gamma)$ group; for which, as seen in Table 5.9, each construction fails the condition. However, the non-maximised output does pass the condition in its $(2,4)$ forms. We note that this group exceeds the basepoint restrictions as per Section 5.2.5; however due to the general failure of the cycle condition, we do not take it as a valid consideration within that section.

This group also stands out for failing the cycle condition due to an edge cycle $\{e_i\}$

with angle sum given as

$$\sum_{i=1}^{m} \alpha(e_i) = 0.$$

This edge cycle being a single edge with finite component comprising of a single vertex. In this way the group stands out as a possible example of a constructed polyhedron with infinite edge, as described in Section 2.6.2.

In Figure 5.1 we display the constructed polyhedron for the $(4, 2, -1 + 1.73205i)$ form of this group; highlighting the edge in question.

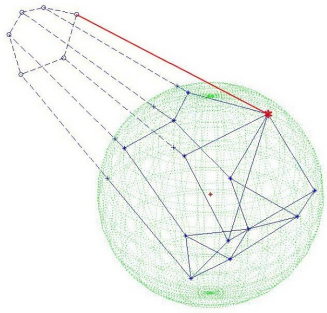


Fig. 5.1: Fundamental polyhedron for $(4, 2, -1 + 1.73205i)$.
With the edge, and finite vertex marked in red.

This is the only example of this type of cycle condition failure seen in the groups we have tested in this thesis. It is notable that the group does pass the condition in some non-maximised forms; so further investigation of these forms may be of interest. Until then, with confirmation of a valid construction in at least one form, we are confident in the general "Free?" (hyperideal) result.

5.2.3 Tremors

As already noted in Section 5.1.3, several of our input parameter sets display what we refer to as data tremors. As tremors are quite common, the two given in Table 5.5 of that section serve as examples; with all others being visible in the tables given in Appendix C.

A data tremor is a slight variation in the output for a single computational parameter set, as the selected vertex epsilon values change. Often this is just a minor shift in the deviation and covolume values; but occasionally it can feature multiple changes across the band of tested epsilon vertex values and also include construction failures. While an obvious and interesting occurrence, these are not true errors like those outlined in Sections 5.2.1 and 5.2.2.

These tremors are specific to an individual construction and are induced by the interaction of vertex generation components of the cutting routines and ε_V-values. Let

$F(P_i, G_i, \varepsilon_V)$ represent the construction process

$$P_i \mapsto P_{i+1},$$

as outlined in Section 3.2.2, working under a resolution of ε_V. Then, let v_0 be a vertex in P_i and v_1 be a vertex induced by the cutting action of G_i, such that

$$10^{-(k+1)} < ||v_0 - v_1||_E^2 < 10^{-k}.$$

Now consider $F(P_i, G_i, \varepsilon_V) = P_{i+1}$.

- If $\varepsilon_V \geq 10^{-k}$, then a $F(P_i, G_i, \varepsilon_V) = P_{i+1}$ is constructed containing only v_0, as a representative of both v_0 and v_1; and,
- if $\varepsilon_V \leq 10^{-(k+1)}$, then a $F(P_i, G_i, \varepsilon_V) = P_{i+1}$ is constructed containing both v_0 and v_1 as distinct vertices of the polyhedron.

In this way, as construction attempts are made over the breadth of vertex epsilon values they move between distinguishing and not distinguishing the vertices v_0 and v_1. The generation of tremors is also dependent on the original group-generating elements and the current basepoint position; so neither construction possibility directly invalidates the other.

Tremors can be induced by any cutting action in the construction process; with polyhedral differences being overwritten by later cutting, paring or basepoint maximising actions. In this way tremors are merely a minor numerical variation in the group G_i data, inducing a minor fluctuation in the final polyhedron.

5.2.4 Parallax

The SnappyD output for a small number of groups displays what we refer to as ($\mathbb{Z}_2$) parallax.

In these cases there is a distinct difference in the constructed domain and output volume for the different constructions within each class of parameter sets. Specifically, the parameter sets $(p, 2, \gamma)$ and $(p, 2, \beta-\gamma)$, of a parameter class, correspond to distinct groups. This can occur when the order two element, in the theorems of Section 2.3.3, is already a member of $\langle f, g\rangle = G$; in which case $\langle f, h\rangle$ is equal to G, rather than an index 2 subgroup of G.

These situations can be identified by sets of input parameter with two (valid) output volumes V_1 and V_2 which correspond to a case where $V_1 = 0.5V_2$; with $(p, 2, \gamma)$ and $(p, 2, \beta - \gamma)$ being members of the same commensurability class. We refer to V_2 as the parallax volume. Table 5.10 highlights all the groups displaying parallax that we have encountered.

As these two volumes split the results for each group in two, we only list the best result for each V_1 and V_2. Here we mark out the parallax deviation and covolume in blue, and the used data in yellow; this matches the tables given in Appendix C. It is

(p,q)	γ	ε_V	deviation	B	$\|\|B\|\|_E^2$	S/V/E	covolume
$(3,2)$	$-2.07638 \pm 0.81470i$	$4 \leftrightarrow 13$	3.64×10^{-14}	$(0.09, 0.20, 0.08)$	0.06	26/26/50	0.127456265703298
	$-0.92362 \pm 0.81470i$	$3 \leftrightarrow 13$	3.33×10^{-15}	$(0.02, 0.35, 0.07)$	0.13	18/16/32	0.255445410566101
$(3,2)$	$0.06115 \pm 0.38830i$	$4 \leftrightarrow 13$	2.65×10^{-13}	$(-0.13, 0.20, -0.06)$	0.06	24/22/44	0.102661071637195
	$-3.06115 \pm 0.38830i$	$3 \leftrightarrow 14$	2.66×10^{-15}	$(-0.23, 0.24, 0.22)$	0.16	18/18/34	0.205514735563156
$(3,2)$	$0.36778 \pm 0.23154i$	$4 \leftrightarrow 12$	1.55×10^{-12}	$(0.26, 0.13, 0.39)$	0.24	28/26/52	0.192584612983471
	$-3.36778 \pm 0.23154i$	$4 \leftrightarrow 12$	7.44×10^{-15}	$(0.18, 0.12, 0.32)$	0.15	20/20/38	0.385610022684362
$(3,2)$	$-2.74540 \pm 0.74953i$	$5 \leftrightarrow 12$	5.70×10^{-13}	$(0.45, 0.15, 0.25)$	0.28	28/30/56	0.263536300108507
	$-0.25460 \pm 0.74953i$	$5 \leftrightarrow 13$	1.78×10^{-14}	$(0.06, 0.42, -0.03)$	0.18	22/26/46	0.526530016209333
$(3,2)$	$-1.08698 \pm 0.98787i$	$3 \leftrightarrow 13$	9.25×10^{-13}	$(-0.02, 0.21, 0.14)$	0.07	28/30/56	0.264308750156759
	$-1.91302 \pm 0.98787i$	$4 \leftrightarrow 13$	7.44×10^{-15}	$(0.12, 0.10, 0.34)$	0.14	20/20/38	0.529035831989798
$(3,2)$	$-2.48848 \pm 1.00187i$	$4 \leftrightarrow 9$	2.29×10^{-11}	$(0.33, 0.17, 0.28)$	0.21	38/38/74	0.770435534681183
	$-0.51152 \pm 1.00187i$	$6 \leftrightarrow 13$	3.08×10^{-12}	$(0.19, 0.48, 0.04)$	0.27	32/36/66	1.539578485870570
$(4,2)$	$-0.12256 \pm 0.74486i$	$3 \leftrightarrow 13$	2.32×10^{-14}	$(0.22, 0.23, -0.06)$	0.11	20/18/36	0.137344777947632
	$-1.87744 \pm 0.74486i$	$3 \leftrightarrow 13$	1.89×10^{-15}	$(0.13, 0.29, 0.13)$	0.12	14/12/24	0.274956314143212
$(4,2)$	$-1.59369 \pm 1.19616i$	$4 \leftrightarrow 12$	2.52×10^{-14}	$(0.42, 0.37, 0.22)$	0.36	26/28/52	0.447434850606938
	$-0.40631 \pm 1.19616i$	$3 \leftrightarrow 13$	1.45×10^{-14}	$(0.24, 0.40, 0.05)$	0.22	20/22/40	0.894660969880485
$(4,2)$	$0.78810 \pm 0.40136i$	$3 \leftrightarrow 12$	4.49×10^{-13}	$(-0.15, 0.35, 0.31)$	0.24	24/22/44	0.347567297638236
	$-2.78810 \pm 0.40136i$	$4 \leftrightarrow 13$	1.12×10^{-14}	$(-0.05, 0.36, 0.39)$	0.28	20/22/40	0.695106728358357
$(5,2)$	$0.25278 \pm 0.85077i$	$4 \leftrightarrow 13$	1.72×10^{-14}	$(0.08, 0.43, 0.29)$	0.28	20/18/36	0.313085454214014
	$-1.63475 \pm 0.85077i$	$4 \leftrightarrow 14$	3.77×10^{-15}	$(-0.15, 0.52, 0.34)$	0.41	14/12/24	0.626170908428058
$(5,2)$	$-1.30017 \pm 1.28803i$	$4 \leftrightarrow 10$	1.43×10^{-12}	$(-0.41, 0.39, 0.44)$	0.51	26/28/52	0.700364591063245
	$-0.08180 \pm 1.28803i$	$3 \leftrightarrow 13$	2.05×10^{-14}	$(0.30, 0.43, 0.24)$	0.33	20/22/40	1.402192567477373
$(6,2)$	$0.50000 \pm 0.86603i$	$3 \leftrightarrow 12$	1.51×10^{-14}	$(-0.23, -0.03, 0.33)$	0.16	22/21/42	0.507438590492248
	$-1.50000 \pm 0.86603i$	$3 \leftrightarrow 13$	5.00×10^{-15}	$(0.18, 0.36, 0.20)$	0.20	16/14/30	1.016300073515976

Tab. 5.10: The groups displaying parallax.

notable that in every case listed, the parallax results coincide with a lower deviation, presumably due to parallax groups having a simpler structure.

However, in selecting the data to use in the tables of Section 5.3 we ignore parallax output. We do this for two primary reasons: when dealing with $\mathbb{Z}_2$ data we are expecting covolumes half that of the original group G, and they fit better into the general theory if they do so; and the general research area has an interest in smallest covolumes.

It is worth noting that these groups have been seen in other references, and in these cases it is commonly a parallax volume that has been used to represent the volume for the class of parameter sets. As such this is the first time we have seen this parallax property mentioned in the literature. See Section 5.3.1.

5.2.5 Basepoint Tracking and Fuchsian Groups

In the tables given in Appendix C, summarising the output details of each construction, we have marked out the basepoint motions that hit or exceed the limits we have placed.

The tracking of each basepoint is based on the sum of Euclidean, as opposed to hyperbolic, distances that the basepoint is moved; so cannot be used as an accurate measure for the actual position of a basepoint. However, any parameter set where the basepoint consistently exceeds the limitations on its movement has the potential to be of relevance.

As the basepoint is moved in non-constant increments, it is possible for its movement

to exceed the bound we have placed on it. But it is uncommon for all computational parameter sets related to a group to exceed this limit; the outlier groups for which this occurs are discussed below.

Outside of these groups, it is notable that these more extreme, individual basepoint movements occur predominantly for input (p, q) with $p > q$, especially for the larger p-values. The number of extreme movements when $p > q$ being greater, by roughly a factor of 4, than the number when $p < q$. This is not surprising as this value corresponds to the order of the lead input generator which, as per Section 4.2.1, is constructed as a standard form with axis running near the basepoint. This infers that the initial basepoint will be in the proximity of a $2\pi/p$ interior angle of the polyhedral construction, increasing the likelihood of injectivity radius expansion as it moves away from this axis.

Fuchsian Groups

In Table 5.11 below, we give the groups for which all successful output constructions have their basepoint movement stopped by the limit we impose. These groups are listed by a representative parameter set and their basepoint movements $b_i = ||B||_E^2$ for each of the 4 computational parameter sets.

(p, q)	γ-value	b_1	b_2	b_3	b_4
$(3, 2)$	$-1.00000 + 0.00000i$	1.01	1.25	1.19	1.19
$(3, 2)$	$-0.38197 + 0.00000i$	1.18	0.88	0.99	0.78
$(3, 2)$	$0.24698 + 0.00000i$	0.82	1.08	0.84	0.79
$(4, 2)$	$-1.00000 + 0.00000i$	1.11		1.11	
$(4, 2)$	$0.61803 + 0.00000i$	1.03	0.83	0.92	0.96
$(5, 2)$	$-0.38197 + 0.00000i$	1.17	0.83	1.16	1.08
$(5, 2)$	$0.61803 + 0.00000i$	1.03	0.93	1.07	0.76
$(6, 2)$	$1.00000 + 0.00000i$	-	0.78	0.96	0.85
$(7, 2)$	$0.24698 + 0.00000i$	0.98	1.00	1.11	0.99
$(7, 2)$	$1.24698 + 0.00000i$	-	0.85	1.00	0.70

Tab. 5.11: Fuchsian basepoint movements.

What is noticeable here is how the 0.70 limit on the basepoint movement is generally exceeded by no small factor.

In each of the examples where this occurs, the group in question corresponds to those identified as, or potential, Fuchsian in [23]. It is not surprising that such movements occur in the construction of polyhedral domains for Fuchsian groups, as such constructions must be the projection between one of more copies of the group's fundamental polygon on $\partial\mathbb{H}^3$; giving rise to the situation generalised in Figure 4.7.

In Figure 5.2 we demonstrate the basepoint movement of these Fuchsian groups in comparison to other hyperideal constructions.

Based on these observations we postulate that if the constructions for each computational parameter set, related to a group G, are successful and exceed the basepoint limit, then G is a Fuchsian group.

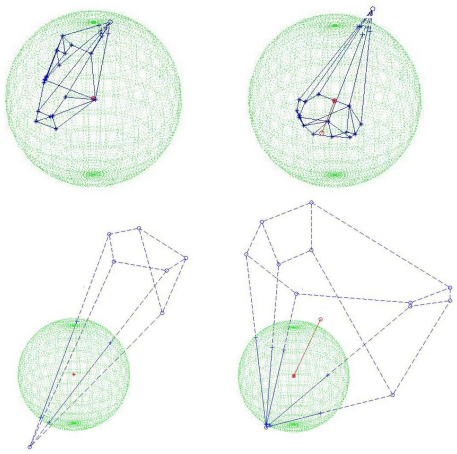


Fig. 5.2: Domain maximisation and basepoint movement.
For a hyperideal polyhedron (top) and Fuchsian polyhedron (bottom).

5.3 Polynomial Tables

The group data we have used in our investigations is arithmetic in nature; based on this, our computational results, and on the results of the source references, we give the following theorems.

Theorem 5.3.1. *The groups given in Table 5.1 and Tables 5.12-5.16 are arithmetic and have covolumes approximated by the values given.*

Theorem 5.3.2. *The groups in Tables 5.17 - 5.18 have infinite covolume and are proper subgroups of arithmetic groups*

Noting that each parameter set (p, q, γ) given in a table also corresponds to groups $(p, q, \overline{\gamma})$, $(p, q, \beta - \gamma)$ and $(p, q, \beta - \overline{\gamma})$. The relevant tables are listed below.

5.3.1 Classification of Groups

Under our tabulated results, combined with our indication of Fuchsian groups, we consider the volume related results of [23].

In Tables 5.19 and 5.20 we compare these results with those arising from our calculations. Noting that the red boxed γ-values in Table 5.20 correspond to groups indicated to be Fuchsian as per Section 5.2.5.

In both Tables 5.19 and 5.20 we mark out new results in yellow. Further note that those volumes of Table 5.19 that are marked out in blue differ from those given in [23]; these groups exhibit parallax volumes and in these cases one form of $\mathbb{Z}_2$-extension obtains the minimum volume arithmetic group of which both forms are embedded in, this volume being given by V_∞. As noted in Section 5.2.4, we are aware of the existence of this parallax, but always select the lower volume for our results.

n	(p,q)	γ-polynomial	γ	covolume
1	$(3,2)$	$z^4+4z^3+4z^2+z+1$	$0.00755+0.51312i$	0.327149654332475
2	$(3,2)$	$z^5+6z^4+13z^3+14z^2+8z+1$	$-0.66222+0.89978i$	0.515836186082985
3	$(3,2)$	$z^5+8z^4+23z^3+27z^2+11z+1$	$-0.18484+0.71242i$	0.517468087113118
4	$(3,2)$	$z^4+6z^3+13z^2+10z+1$	$-0.79289+0.97832i$	0.686008759135839
5	$(3,2)$	$z^4+8z^3+21z^2+19z+4$	$0.21021+0.41375i$	0.463893546593001
6	$(3,2)$	$z^4+6z^3+13z^2+11z+1$	$-1.29342+1.00144i$	0.463972819748495
7	$(3,2)$	$z^5+9z^4+29z^3+38z^2+16z+1$	$0.02268+0.62320i$	0.646287064285721
8	$(3,2)$	$z^5+6z^4+12z^3+11z^2+7z+1$	$-0.37053+0.84016i$	0.647079332036773
9	$(3,2)$	z^3+2z^2+z+1	$-0.12256+0.74486i$	0.785999075910039
10	$(3,2)$	$z^4+4z^3+6z^2+5z+1$	$-0.75187+1.03398i$	1.145197409428369
11	$(3,2)$	$z^5+5z^4+6z^3-z^2-z+1$	$0.34815+0.31570i$	0.567165208527598
12	$(3,2)$	$z^5+7z^4+19z^3+25z^2+14z+1$	$-1.35087+1.05848i$	0.567285975023829
13	$(3,2)$	$z^5+7z^4+19z^3+23z^2+10z+1$	$-1.02127+1.12212i$	1.894601048848805
14	$(3,2)$	$z^4+5z^3+8z^2+6z+3$	$-0.34861+0.75874i$	0.432564598947726
15	$(3,2)$	$2z^2+2z+2$	$-1.00000+1.00000i$	0.610519526694357
16	$(3,2)$	$z^5+7z^4+18z^3+23z^2+17z+5$	$-0.60186+0.93867i$	0.711117203273613
17	$(3,2)$	$z^5+5z^4+7z^3+3z^2+2z+1$	$0.11005+0.57190i$	0.712543560125077
18	$(3,2)$	$z^5+8z^4+25z^3+37z^2+23z+3$	$-1.86240+1.07589i$	0.865996064707863
19	$(3,2)$	z^3+3z^2+2z+2	$-0.23931+0.85787i$	1.255231836333305
20	$(3,2)$	z^3+4z^2+6z+2	$-1.22816+1.11514i$	1.059048949359549
21	$(3,2)$	z^3+4z^2+5z+4	$-0.65219+1.02885i$	1.498880413762323
22	$(3,2)$	$z^5+6z^4+11z^3+6z^2+2z+3$	$0.17229+0.58559i$	1.181360578218381
23	$(3,2)$	$z^5+6z^4+12z^3+11z^2+8z+3$	$-0.13972+0.82586i$	1.750517907450094
24	$(3,2)$	z^4+3z^3-2z+1	$0.46746+0.27759i$	1.019779352592997
25	$(3,2)$	$z^4+6z^3+12z^2+7z+1$	$-0.41847+0.93916i$	1.018809897806248
26	$(3,2)$	z^3+3z^2+4z+1	$-1.34116+1.16154i$	1.319305550522633
27	$(3,2)$	z^3+2z^2+1	$0.10278+0.66546i$	1.709353487322167
28	$(3,2)$	$z^4+3z^3+z^2-z+1$	$0.33909+0.44663i$	1.472909330019834
29	$(3,2)$	$z^4+6z^3+12z^2+9z+1$	$-1.50000+0.60666i$	0.039015451336356
30	$(3,2)$	$z^4+5z^3+7z^2+3z+1$	$-0.21190+0.40136i$	0.040890367870668
31	$(3,2)$	z^3+4z^2+4z+2	$-0.58036+0.60629i$	0.132387219941707
32	$(3,2)$	z^3+5z^2+8z+5	$-1.12256+0.74486i$	0.157117893796128
33	$(3,2)$	z^3+3z^2+2z+1	$-0.33764+0.56228i$	0.156983968194460
34	$(3,2)$	$z^4+5z^3+8z^2+6z+1$	$-0.92362+0.81470i$	0.127456265703298
35	$(3,2)$	z^2+3z+3	$-1.50000+0.86603i$	0.338313868802930
36	$(3,2)$	z^3+4z^2+5z+3	$-0.76721+0.79255i$	0.263008715157447
37	$(3,2)$	$z^4+5z^3+6z^2+1$	$0.06115+0.38830i$	0.102661071637195
38	$(3,2)$	$z^6+8z^5+24z^4+35z^3+28z^2+12z+1$	$-0.52842+0.78122i$	0.362887228283480
39	$(3,2)$	z^3+3z^2+z+2	$-0.11535+0.58974i$	0.330717107428080

Tab. 5.12: The arithmetic groups of the $(3,2)$-commutator plane.

Together, these results complete the volume classification of the groups in [23]. Specifically, the results here allow us to complete the holes within Table 11 of [23], and so extended Theorem 8.2, of [23].

Theorem 5.3.3 (Revised Theorem 8.2 of [23])**.** *The following two-generator Kleinian groups $G_{p,n}$, generated by elliptics of order p and 2, are arithmetic.*

- $G_{3,n}$ *for* $n=3,...,14$, $n\neq 12$,
- $G_{4,n}$ *for* $n=2,...,13$, $n\neq 10,11$,
- $G_{5,n}$ *for* $n=2,...,11$, $n\neq 5$,
- $G_{6,n}$ *for* $n=1,...,8$, $n\neq 4,7$.

We also note that Table 5.18 and the failed constructions of Table 5.8 comprise the totality of unknown, unconfirmed volumes listed in Tables 5.5, 5.6, 5.7 and 5.11 of [62]; our earlier results for the groups of [62] having already been included in that reference. Further, groups 3 and 10 in Table 5.5 and 10 in Table 5.6 of [62] are shown to be among our noted parallax groups $n=3$, 6 and 15 in Table 5.16, respectively.

n	(p,q)	γ-polynomial	γ	covolume
1	$(4,2)$	z^3+z^2+1	$0.23279+0.79255i$	0.595004149184967
2	$(4,2)$	z^3-z+1	$0.66236+0.56228i$	0.824071575741213
3	$(4,2)$	z^2-z+1	$0.50000+0.86603i$	1.524595659895754
4	$(4,2)$	z^3+z^2+2z+1	$-0.21508+1.30714i$	2.122090588879277
5	$(4,2)$	$z^4+z^3-3z^2-z+3$	$1.06115+0.38830i$	1.542632159257580
6	$(4,2)$	$z^4+2z^3+2z^2+3z+1$	$0.03640+1.21238i$	1.540788174525126
7	$(4,2)$	$z^4+z^3-z^2+z+1$	$0.65139+0.75874i$	1.624021910597929
8	$(4,2)$	z^4-4z^2+z+3	$1.36778+0.23154i$	2.900382924576496
9	$(4,2)$	$z^4+3z^3+5z^2+5z+1$	$-0.57943+1.45743i$	2.889771482503296
10	$(4,2)$	$z^4+2z^3+z^2+2z+1$	$0.20711+0.97832i$	1.028760806843382
11	$(4,2)$	z^2+z+2	$-0.50000+1.32288i$	1.332406780851792
12	$(4,2)$	z^2+z+2	$0.34781+1.02885i$	2.232710498631728
13	$(4,2)$	z^3+3z^2+5z+4	$-0.77330+1.46771i$	1.927861496441281
14	$(4,2)$	z^3+2z^2+2z+3	$-0.09473+1.28374i$	2.597157107493283
15	$(4,2)$	z^3-2z+2	$0.88465+0.58974i$	1.985525637343016
16	$(4,2)$	z^2+z+1	$-1.50000+0.86603i$	0.253384811296644
17	$(4,2)$	z^3+2z^2+z+1	$-0.12256+0.74486i$	0.137344777947632
18	$(4,2)$	z^2+2z+2	$-1.00000+1.00000i$	0.457982797088671
19	$(4,2)$	z^3+3z^2+4z+3	$-0.65884+1.16154i$	0.593932061794999
20	$(4,2)$	z^3+z^2+1	$0.23279+0.79255i$	0.595004149184967
21	$(4,2)$	z^3+2z^2+2z+2	$-0.22816+1.11514i$	0.793839731308609
22	$(4,2)$	z^3+z^2-z+1	$0.41964+0.60629i$	0.264774439883719
23	$(4,2)$	z^2+1	$0.00000+1.00000i$	0.915965594176532
24	$(4,2)$	$z^4+3z^3+4z^2+4z+1$	$-0.40630+1.19616i$	0.447434850606938
25	$(4,2)$	$z^4+z^3-2z^2+1$	$0.78810+0.40136i$	0.347567297638236

Tab. 5.13: The arithmetic groups in the $(4,2)$-commutator plane.

n	(p,q)	γ-polynomial	γ	covolume
1	$(5,2)$	$z^2-\beta z+1$	$-0.69098+0.72287i$	0.093325539506778
2	$(5,2)$	$z(z-\beta-1)^2-\beta-1$	$0.11803+0.60666i$	0.039050285615487
3	$(5,2)$	$z^2-(\beta+1)z+1$	$-1.19098+0.98159i$	0.468603427380289
4	$(5,2)$	$z(z-\beta-1)^2+1$	$0.25278+0.85077i$	0.313085454214014
5	$(5,2)$	$z^3-(2\beta+1)z^2+(\beta^2+\beta+2)z-2\beta-1$	$-0.38197+1.27202i$	0.861236641862642
6	$(5,2)$	$z^3-(2\beta+2)z^2+(\beta^2+2\beta+2)z-\beta$	$0.11803+1.16963i$	1.119270315426469
7	$(5,2)$	$z^4-(2\beta+1)z^3+(\beta^2+\beta+2)z^2-2\beta z+1$	$-0.08180+1.28803i$	0.700364591063245
8	$(5,2)$	$z^3-(2\beta+3)z^2+(\beta^2+3\beta+2)z+1$	$0.61803+0.78615i$	0.861241520146132
9	$(5,2)$	$z(z-\beta)(z-\beta-2)^2+1$	$0.87764+0.58260i$	0.454271480250933

Tab. 5.14: The arithmetic groups in the $(5,2)$-commutator plane.

n	(p,q)	γ-polynomial	γ	covolume
1	$(6,2)$	z^3-2z^2+2	$1.41964+0.60629i$	1.853421079184026
2	$(6,2)$	z^3-z^2+z+1	$0.77184+1.11514i$	1.851412053612527
3	$(6,2)$	z^3-2z^2+z+1	$1.23279+0.79255i$	2.110523747092161
4	$(6,2)$	z^3-3z^2+2z+1	$1.66236+0.56228i$	4.402234623856832
5	$(6,2)$	z^3+z+1	$0.34116+1.16154i$	1.320702203534967
6	$(6,2)$	z^2+2	$0.00000+1.41421i$	2.007682006682812
7	$(6,2)$	z^2-2z+2	$1.00000+1.00000i$	3.053218647256478
8	$(6,2)$	z^3+z^2+3z+1	$-0.31945+1.63317i$	2.646859749260644
9	$(6,2)$	z^2-z+2	$0.50000+1.32288i$	3.549442254748726
10	$(6,2)$	z^2+z+1	$-0.50000+0.86603i$	0.253735401602561
11	$(6,2)$	z^2+1	$0.00000+1.00000i$	0.610643729451522
12	$(6,2)$	z^2-z+1	$0.50000+0.86603i$	0.507438590492248
13	$(6,2)$	z^3+z^2+2z+1	$-0.21508+1.30714i$	1.019497428415326
14	$(6,2)$	z^3+z+1	$0.34116+1.16154i$	1.320702203534967
15	$(6,2)$	z^3-z^2+1	$0.87744+0.74486i$	1.022004856907895

Tab. 5.15: The arithmetic groups in the $(6,2)$-commutator plane.

n	(p,q)	γ	covolume
1	$(3,2)$	$-0.50000+0.86603i$	0.676627737605571
2	$(3,2)$	$0.36778+0.23154i$	0.192584612983471
3	$(3,2)$	$-0.25460+0.74953i$	0.263536300108507
4	$(3,2)$	$0.47582+0.16460i$	0.220219588908876
5	$(3,2)$	$0.23676+0.31257i$	0.154318773285096
6	$(3,2)$	$-0.07087+0.66435i$	0.297303264846344
7	$(3,2)$	$-1.08698+0.98787i$	0.264308750156759
8	$(3,2)$	$0.02764+0.73661i$	2.606687956763813
9	$(3,2)$	$-0.33909+0.92881i$	1.786256005477690
10	$(3,2)$	$-1.40631+1.19616i$	1.473780096936488
11	$(3,2)$	$-0.56154+1.01758i$	1.886983797195308
12	$(3,2)$	$-0.42847+1.00664i$	2.457416519588215
13	$(3,2)$	$-0.87763+1.11400i$	3.438382428925763
14	$(3,2)$	$-0.02256+0.77896i$	2.904334127793920
15	$(3,2)$	$-0.51152+1.00187i$	0.770435534681183
16	$(3,2)$	$0.61068+0.18781i$	1.959655953058214
17	$(4,2)$	$1.00755+0.51312i$	3.667339431536167
18	$(4,2)$	$0.17660+1.20282i$	3.308662886740566
19	$(4,2)$	$-0.78492+1.30714i$	0.824868942429759

Tab. 5.16: Additional arithmetic points.

n	(p,q)	γ-polynomial	γ
1	$(3,2)$	$z+1$	$-1.00000+0.00000i$
2	$(3,2)$	z^2+3z+1	$-0.38197+0.00000i$
3	$(3,2)$	z^3+4z^2+3z-1	$0.24698+0.00000i$
4	$(4,2)$	$z+1$	$-1.00000+0.00000i$
5	$(4,2)$	z^2+z-1	$0.61803+0.00000i$
6	$(4,2)$	$z^4+4z^3+7z^2+6z+1$	$-1.00000+1.27202i$
7	$(5,2)$	$z-\beta-1$	$-0.38197+0.00000i$
8	$(5,2)$	$z-\beta-2$	$0.61803+0.00000i$
9	$(5,2)$	$z^2-\beta z+2$	$-0.69098+1.23391i$
10	$(6,2)$	z^2+z+2	$-0.50000+1.32288i$
11	$(6,2)$	$z-1$	$1.00000+0.00000i$
12	$(7,2)$	$z-\beta-1$	$0.24698+0.00000i$
13	$(7,2)$	$z^2-\beta z+1$	$-0.37651+0.92641i$
14	$(7,2)$	$z-\beta-2$	$1.24698+0.00000i$

Tab. 5.17: The hyperideal polynomial groups.

n	(p,q)	γ	covolume
1	$(3,2)$	$-1.50000+0.99849i$	hyperideal
2	$(3,2)$	$-1.50000+1.07899i$	hyperideal
3	$(3,2)$	$-1.50000+1.16963i$	hyperideal
4	$(3,2)$	$-1.50000+1.21740i$	hyperideal
5	$(3,2)$	$-1.50000+1.30625i$	hyperideal
6	$(3,2)$	$-1.50000+1.27647i$	hyperideal
7	$(4,2)$	$-1.00000+1.41421i$	hyperideal
8	$(4,2)$	$-1.00000+1.55377i$	hyperideal
9	$(4,2)$	$-1.00000+1.65289i$	hyperideal
10	$(4,2)$	$-1.00000+1.73205i$	hyperideal

Tab. 5.18: Additional hyperideal points.

(p,q)	γ	covolume	V_0	V_∞
$(3,2)$	$-1.50000+0.60666i$	0.039015451336356	0.0390	0.0390
$(3,2)$	$-0.21190+0.40136i$	0.040890367870668	0.0408	0.0408
$(3,2)$	$-0.58036+0.60629i$	0.132387219941707	0.1323	0.0661
$(3,2)$	$-1.12256+0.74486i$	0.157117893796128	0.1571	0.0785
$(3,2)$	$-0.33764+0.56228i$	0.156983968194460	0.1571	0.0785
$(3,2)$	$-0.92362+0.81470i$	0.127456265703298	0.1274	0.1274
$(3,2)$	$-1.50000+0.86603i$	0.338313868802930	0.3383	0.0845
$(3,2)$	$-0.76721+0.79255i$	0.263008715157447	0.2638	0.0659
$(3,2)$	$0.06115+0.38830i$	0.102661071637195	0.2056	0.1028
$(3,2)$	$-0.52842+0.78122i$	0.362887228283480	?	?
$(3,2)$	$-0.11535+0.58974i$	0.330717107428080	0.3308	0.1654
$(4,2)$	$-1.50000+0.86603i$	0.253384811296644	0.2537	0.1268
$(4,2)$	$-0.12256+0.74486i$	0.137344777947632	0.1374	0.1374
$(4,2)$	$-1.00000+1.00000i$	0.457982797088671	0.4579	0.2289
$(4,2)$	$-0.65884+1.16154i$	0.593932061794999	0.5936	0.2968
$(4,2)$	$0.23279+0.79255i$	0.595004149184967	0.5936	0.2968
$(4,2)$	$-0.22816+1.11514i$	0.793839731308609	0.7943	0.0661
$(4,2)$	$0.41964+0.60629i$	0.264774439883719	0.2647	0.0661
$(4,2)$	$0.00000+1.00000i$	0.915965594176532	0.9159	0.1526
$(4,2)$	$-0.40630+1.19616i$	0.447434850606938	0.8951	0.4475
$(4,2)$	$0.78810+0.40136i$	0.347567297638236	0.3475	0.3475
$(5,2)$	$-0.69098+0.72287i$	0.093325539506778	0.0933	0.0933
$(5,2)$	$0.11803+0.60666i$	0.039050285615487	0.0390	0.0390
$(5,2)$	$-1.19098+0.98159i$	0.468603427380289	0.4686	0.0390
$(5,2)$	$0.25278+0.85077i$	0.313085454214014	?	?
$(5,2)$	$-0.38197+1.27202i$	0.861236641862642	0.8612	0.0717
$(5,2)$	$0.11803+1.16963i$	1.119270315426469	1.1199	0.0933
$(5,2)$	$-0.08180+1.28803i$	0.700364591063245	?	?
$(5,2)$	$0.61803+0.78615i$	0.861241520146132	0.8612	0.0717
$(5,2)$	$0.87764+0.58260i$	0.454271480250933	?	?
$(6,2)$	$-0.50000+0.86603i$	0.253735401602561	0.2537	0.0845
$(6,2)$	$0.00000+1.00000i$	0.610643729451522	0.6106	0.3053
$(6,2)$	$0.50000+0.86603i$	0.507438590492248	0.5074	0.1691
$(6,2)$	$-0.21508+1.30714i$	1.019497428415326	1.0212	0.5106
$(6,2)$	$0.34116+1.16154i$	1.320702203534967	1.3193	0.3298
$(6,2)$	$0.87744+0.74486i$	1.022004856907895	1.0212	0.5106

Tab. 5.19: The arithmetic $G_{p,n}$ groups.

Of particular relevance here though, is that these few new results (of Table 5.18) all correspond to hyperideal polyhedrons; this demonstrates that there are a number of non-Arithmetic groups within the list of groups given in Tables 5.5, 5.6 and 5.11 of [62]. This may not be entirely unexpected, but stands out among the data given in the reference.

We end by noting that, by Conjectures 5.1.2, 5.2.2 and 5.3.2 of [62] where a "computational boundary" has been identified for the space of free groups, the arithmetic two-generator Kleinian groups generated by an order 2 element and either an order 3, 4 or 6 element are all included in the tables of this Chapter, specifically the groups given in Table 5.1, Tables 5.12-5.16, and possibly Table 5.8 (pending confirmation of volume).

(p,q)	γ	V_0	V_∞
$(3,2)$	$-1.00000+0.00000i$	S_4	S_4
$(3,2)$	$-0.38197+0.00000i$	A_4	A_5
$(3,2)$	$0.24698+0.00000i$	$Fuch$	$Fuch$
$(4,2)$	$-1.00000+0.00000i$	S_4	S_4
$(4,2)$	$0.61803+0.00000i$	$Fuch$	$Fuch$
$(4,2)$	$-1.00000+1.27202i$	∞	0.0717
$(5,2)$	$-0.38197+0.00000i$	A_5	A_5
$(5,2)$	$0.61803+0.00000i$	$Fuch$	$Fuch$
$(5,2)$	$-0.69098+1.23391i$	∞	0.4610
$(6,2)$	$-0.50000+1.32288i$	∞	0.5555
$(6,2)$	$1.00000+0.00000i$	$Fuch$	$Fuch$
$(7,2)$	$0.24698+0.00000i$	$-$	$Fuch$
$(7,2)$	$-0.37651+0.92641i$	$-$	$?$
$(7,2)$	$1.24698+0.00000i$	$-$	$Fuch$

Tab. 5.20: The hyperideal $G_{p,n}$ groups.

6. CLOSING REMARKS

In this work we have made dramatic alterations to the `Dirichlet_` subroutines of the commonly used computational package SnapPea [59]; expanding and refocusing the input data it runs on and offering a depth of insight into the computational processes used in the construction of fundamental polyhedrons, for hyperbolic 3-orbifolds, that we have not seen elsewhere in the literature.

We have then used our new program, SnappyD, to construct and study fundamental domains for the groups listed in several recent publications related to the classification of arithmetic hyperbolic 3-orbifolds and investigation of Kleinian groups; [23], [60] and [62]. These constructions reinforce the results seen in these references and extend them in important ways.

Most specifically, we have reaffirmed the covolume results seen in [23] and [62]; adding values missing from the tables and confirming those values already seen.

In doing this we give Theorem 5.3.3 in Section 5.3.1, an expansion of Theorem 8.2 of [23], and complete Table 1 of [60]. In addition to this, under the conjectures of [62] we also further refine the complete list of two-generator arithmetic groups that are generated by one element of order 2 and one element of order 3, 4 or 6; leaving only 18 possible members still awaiting confirmation.

In using these computational methods, we highlight the value of these programs and their ability to lend considerable insight into computational methods used to construct domains in these, and related, applications. This insight is exemplified in our identification of both:

- constructed polyhedrons that fail the cycle condition (Section 5.2.1); and
- pairs of (p, q, γ) and $(p, q, \beta - \gamma)$ parameter sets that demonstrate parallax in their volumes (Section 5.2.4).

Results based on these groups (and the constructed polyhedrons) have been used in previous work; but as far as we are aware, this is the first instance that their nature has been highlighted in relation to the classification of arithmetic groups.

From here, a greater analytical exploration of the outlier results would be of considerable interest. Especially this parallax property, and whether it can be determined from, or acts to classify, parameter sets; and exploring the conjectured relationship between Fuchsian groups and large basepoint movement (Section 5.2.5).

As an experimental project, working with fixed precision, accuracy is an issue here and increasing the accuracy of these routines is something to be considered; along

with attempting to confirm the arithmeticity of the last 18 groups. However, in the meantime we see this limited precision as having considerable merit for the quick and experimental approach it allows. Especially with regards to explorations related to things like the exceptional data points resulting from the disc-covering procedure of [28]; γ-values that relate to groups that may or may not be discrete. This provides a level of computational insight into how group elements generate domains that we believe, with further refinement, could lead to interesting studies comparable to those on the iterative actions of Fuchsian and quasi-Fuchsian groups, [50].

Related to all this is the question of discreteness and the role of a representation in finite precision with relation to results. Under finite (or even arbitrary) precision all the groups that we are primarily concerned with (namely groups generated by two elliptic transformations which are not free on those generators) are analytically finite, as all γ parameters are isolated.

Considering the future expansion of this program, it would be of value to incorporate a robust procedure for determining if a given group is not discrete, possibly through the addition tests based around Jørgensen's inequalities and a computational reference for the known disc coverings of the γ parameter space (and other similar spaces) already studied. It would also be of benefit to further develop the available machinery in an attempt to generate a more thorough description of the boundary (or at least a computational approximation of the boundary) for the space of free groups. Such a combination of methods would advantage the more visually exciting areas of these subjects, such as: the techniques utilized in generating the burgeoning description of $(p, 2)$-commutator planes [28],[62]; the iterative mapping of spaces under group actions [50]; and this description of quotient spaces.

In considering the general construction proceedures, we also echo the sentiment of [20] in encouraging an attempt at building a more expansive and general collection of functions for the construction and study of fundamental domains. The majority of the functions used in SnappyD (SnapPea), and comparable programs, take a general form; which, under modern programming techniques with the encapsulation of data and alternative forms of data typing, should be amenable to a much greater range of applications. Together this would allow for an expansion of scope from hyperbolic 3-orbifolds, to geometric manifolds in general; and more complicated situations.

Though, in considering future computational work, we are curious as to the continued use of multiple separate code packages commonly being used, e.g. Pari, Geoview; and wonder if it would not be better to write our programs in one of the more common mathematical programming suites out there[1], where the implementation of symbolic computation, and other complexities, has already been handled.

[1] For example, Mathematica provides comprehensive function databases for these applications, including access to a computer algebra system for symbolic computation and a variety of highly developed graphical output, and user interface, options.

BIBLIOGRAPHY

[1] W. Abikoff and B. Maskit, *Geometric Decompositions of Kleinian Groups*, Amer. J. of Math., Vol. 99, No. 4,(1977), pp. 687-697.

[2] I. Agol, *Tameness of Hyperbolic 3-Manifolds*, arXiv:0405568, (2004).

[3] I. Agol, D. Groves, and J. Manning, *The virtual Haken conjecture*, arXiv:1204.2810, (2012).

[4] L.V. Ahlfors, *Finitely Generated Kleinian Groups*, Amer. J. of Math., Vol. 86, No. 2, (1964), pp. 413-429.

[5] L.V. Ahlfors, *Fundamental Polyhedrons and Limit Point Sets of Kleinian Groups*, Proc. N. A. S., Vol. 55, No. 2, (1966), pp. 251-254.

[6] L.V. Ahlfors, *The Structure of a Finitely Generated Kleinian Group*, Acta Mathematica, July 1969, Vol. 122, No. 1, (1969), pp. 1-17.

[7] H. Akiyoshi, M., Sakuma, M., Wada and Y., Yamashita, *Punctured Torus Groups and 2-Bridge Knot Groups (I)*, Lecture Notes in Mathematics, Springer, 2007.

[8] M. Ashikhmin, S. Marschner and P. Shirley, *Fundamentals of Computer Graphics*, 3rd Revised Ed., A K Peters/CRC Press, (2009).

[9] A. F. Beardon and B. Maskit, *The Limit Points of Kleinian Groups and Finite Sided Fundamental Polyhedra*, Acta Mathematica, Vol. 132, No. 1, (1974), pp. 1-12.

[10] A. F. Beardon, *The Geometry of Discrete Groups*, Springer-Verlag, (1983).

[11] A. Borel, *Commensurability Classes and Volumes of Hyperbolic 3-Manifolds*, Ann. Scuola Norm. Sup., Vol. 8, No. 1, (1981), pp. 1-33.

[12] J. Brock, R. Canary, and Y. Minsky, *The Classification of Kleinian Surface Groups, II: The Ending Lamination Conjecture*, arXiv preprint math/0412006, (2004).

[13] D. Calegari, and D. Gabai, *Shrinkwrapping and the Taming of Hyperbolic 3-Manifolds*, J. Amer. Math. Soc., Vol. 19, No. 2 (2006), pp. 385-446.

[14] J. T. Callahan, *The Arithmetic and Geometry of Two-Generator Kleinian Groups*, PhD Thesis, University of Texas at Austin, (2009).

[15] J. Callahan and A. Reid, *Hyperbolic Structures on Knot Complements* Chaos Solitons and Fractals, Vol. 9, No. 4, (1998), pp. 705-738.

[16] H. Cooper, *Two Generator Discrete Groups of Isometries and their Representation*, MSc Thesis, Massey University, (2008).

[17] D. Coulson, O.A. Goodman, C.D. Hodgson, and W.D. Neumann, *Computing Arithmetic Invariants of* 3*-Manifolds*, Experiment. Math., Vol. 9, No. 1, (2000), pp. 127-152.

[18] M. P. Do Carmo, *Riemannian Geometry*, Birkhäuser Boston, (1992).

[19] W.D. Dunbar, *Geometric Orbifolds*, Revista Mathematica, Vol. 1, No. 1, (1988), pp. 67-99.

[20] D. Epstein and C. Petronio, *An Exposition Of Poincaré's Polyhedron Theorem*, L'Enseignement Mathématique, Vol. 40, (1994), pp. 113-170.

[21] R. Frigerio and B. Martelli, *Countable Groups are Mapping Class Groups of Hyperbolic* 3*-Manifolds*, Math. Res. Lett., Vol. 13, No. 6, (2006), pp. 897-910.

[22] D. Gabai, R. Meyerhoff and N. Thurston, *Homotopy Hyperbolic* 3*-Manifolds are Hyperbolic*, Ann. of Math., Vol. 157, No. 2, (2003), 335-431.

[23] F. W. Gehring, C. Maclachlan, G. J. Martin and A.W. Reid, *Arithmeticity, Discreteness and Volume*, Trans. Amer. Math. Soc., Vol. 349, No. 9, (1997), pp. 3611-3643.

[24] F.W. Gehring, C. Maclachlan and G.J. Martin, *On the Discreteness of the Free Product of Finite Cyclic Groups*, Department of Mathematics, The University of Auckland, New Zealand, 1997.

[25] F. Gehring and G. Martin, *Iteration Theory and Inequalities for Kleinian Groups*, Bull. Am. Math. Soc., Vol. 21, (1989), pp. 57-63.

[26] F.W. Gehring and G.J. Martin, *Inequalities for Möbius Transformations and Discrete Groups*, J. reine angew. Math., Vol. 418, (1991), pp. 31-76 .

[27] F. W. Gehring and G. J. Martin, *Stability and Extremality in Jørgensen's Inequality*, Complex Variables, Vol. 12, (1989), pp. 277-282.

[28] F. W. Gehring and G. J. Martin, *Commutators, Collars and the Geometry of Möbius Groups*, J. Anal. Math., Vol. 63, No. 1, (1994), pp. 175-218.

[29] F.W. Gehring, J.P. Gilmann and G.J. Martin, *Kleinian Groups with Real Parameters*, Communications in Contemporary Mathematics, Vol. 3, No. 02, (2001), pp. 163-186.

[30] J.P. Gilman, *Algorithms, Complexity and Discreteness in* $PSL(2,\mathbb{C})$, Journal D'Analyse Mathématique, Vol. 73, No. 1, (1997), pp. 91-114.

[31] L. Greenberg, *Fundamental Polyhedra for Kleinian Groups*, Ann. of Math., 2nd Series, Vol. 84, No. 3, (1966), pp. 433-441.

[32] P. Harpe, *Topics in Geometric Group Theory*, University of Chicago Press, (2000).

[33] T. Jørgensen, *On Discrete Groups of Möbius Transformations*, Amer. J. of Math., Vol. 98, No. 3, (1976), pp. 739-749.

[34] T. Jørgensen and A. Marden, *Algebraic and Geometric Convergence of Kleinian Groups*, Math. Scand., Vol. 66, (1990), pp. 47-72.

[35] J. Kahn and V. Markovic, *Immersing Almost Geodesic Surfaces in a Closed Hyperbolic Three Manifold*, arXiv:0910.5501 (2009).

[36] K. S. Lam, *Topics in Contemporary Mathematical Physics*, World Scientific Publishing, (2003).

[37] M. Lipyanskiy, *A Computer-Assisted Application of Poincaré's Fundamental Polyhedron Theorem*, preprint, (2002).

[38] C. Maclachlan and G. Martin, *All Kleinian Groups with two Elliptic Generators whose Commutator is Elliptic*, Math. Proc. Camb. Phil. Soc., Vol. 135, No. 3, (2003), pp. 413-420.

[39] C. Maclachlan and G. Martin, 2-*Generator Arithmetic Kleinian Groups*, J. Reine Angew. Math., Vol. 511, (1999), pp. 95-117.

[40] C. Maclachlan and A. W. Reid, *Commensurability Classes of Arithmetic Kleinian Groups and their Fuchsian Subgroups*, Math. Proc. Camb. Phil. Soc., Vol. 102, No. 02, (1987), pp. 251-257.

[41] C. Maclachlan, and A. Reid, *The Arithmetic of Hyperbolic* 3-*Manifolds*, Springer-Verlag, (2003).

[42] A. Marden, *The Geometry of Finitely Generated Kleinian Groups*, Ann. of Math., Vol. 99, No. 2, (1974), pp. 383-462.

[43] B. Maskit, *Some Special* 2-*Generator Kleinian Groups*, Proc. of the Amer. Math. Soc., Vol. 106, No. 1, (1989), pp. 175-186.

[44] B. Maskit, *Kleinian Groups*, Springer-Verlag, (1980).

[45] B. Maskit, *On Poincaré's Theorem for Fundamental Polygons*, Adv. in Math., Vol. 7, No. 3, (1971), pp. 219-230.

[46] J. Milnor, *Hyperbolic Geometry: The First 150 Years*, Bull. of the Amer. Math. Soc. (New Series), Vol. 6, Num. 1, (1982).

[47] J.W. Morgan, *Recent Progress on the Poincaré Conjecture and the Classification of* 3-*Manifolds*, Bull. of the Amer. Math. Soc. (New Series), Vol. 42, No. 1, (2004), pp. 57-78.

[48] J. Morgan and G. Tian, *Ricci Flow and the Poincaré Conjeture*, Amer. Math. Soc., Vol. 3, (2007).

[49] G. Mostow, *Strong Rigidity of Locally Symmetric Spaces*, Annals of Math. Studies, Princeton University Press, Vol. 78, (1973).

[50] D. Mumford, C. Series, and D. Wright, *Indra's Pearls: The vision of Felix Klein*, Cambridge University Press, (2002).

[51] J.R. Munkres, *Topology*, Prentice Hall, (2000).

[52] P.J. Nicholls, *Ford and Dirichlet Regions for Discrete Groups of Hyperbolic Motions*, Trans. of the Amer. Math. Soc., Vol. 282, No. 1, (1984), pp. 355-365.

[53] *Pari*, Université Bordeaux, available from `pari.math.u-bordeaux.fr/`.

[54] D. Pedoe, *Geometry, A Comprehensive Course*, Dover Publications, (1988).

[55] J. G. Ratcliffe, *Foundations of Hyperbolic Manifolds*, Springer-Verlag, (1994).

[56] R. Riley, *Applications of a Computer Implementation of Poincaré's Theorem on Fundamental Polyhedra*, Math. Comp., Vol. 40, No. 162, (1983), pp. 607-632.

[57] D. Tan, *On Two-Generator Discrete Groups of Möbius Transformations*, Proc. of the Amer. Math. Soc., Vol. 106, No. 3, (1989), pp. 763-770.

[58] W. Thurston, *Three Dimensional Manifolds, Kleinian Groups and Hyperbolic Geometry*, Bull. of the Amer. Math. Soc. (New Series), Vol 6, No. 2, (1982).

[59] J. Weeks, *SnapPea*, available from `www.geometrygames.org/SnapPea/`.

[60] A.G.T Williams, *Arithmeticity of Orbifold Generalised Triangle Groups*, J. of Pure and Applied Algebra, Vol. 177, No. 3, (2003), pp. 309-322.

[61] H. Yamamoto, *Constructibility and Geometric Finiteness of Kleinian Groups*, Tôhoku Math. J., Vol. 32, (1980), pp. 353-362.

[62] Q. Zhang, *Two Elliptic Generator Kleinian Groups*, PhD Thesis, Massey University, (2010).

[63] Q. Zhang, *Parameters of the Two Generator Discrete Elementary Groups*, MSc Thesis, Massey University, (2006).

APPENDIX

A. HYPERBOLIC 3-SPACE

The primary focus of this thesis is a computational discussion of fundamental domains for hyperbolic 3-orbifolds; or, more directly, for the action of two-elliptic-generator groups acting on hyperbolic space. Here we provide a brief overview of the primary models of hyperbolic 3-space and some of the relationships between them and their isometries.

A.1 The Four Models of Hyperbolic 3-space

Hyperbolic geometry is one of the seven canonical geometries described by Thurston [58], and is represented by a 3-dimensional real vector space with a constant negative curvature of -1. Below we outline the four canonical models of hyperbolic 3-space.

Note that in the main text we have denoted generic hyperbolic 3-space, unspecific of model, by X with metric ρ.

The Hyperboloid Model

This first model is the only one not commonly embedded in a subset of $\mathbb{R}^3$, instead being embedded in Lorentzian 4-space. Also known as Minkowski Space, this is a four dimensional real vector space under the indefinite Lorentzian norm $\|x\|_L^2$, and is commonly denoted $\mathbb{R}^{1,3}$.

$$\|x\|_L^2 = -x_0^2 + \sum_{i=1}^{3} x_i^2.$$

Note that the $x_0 = 0$ hyperspace of $\mathbb{R}^{1,3}$ is $\mathbb{R}^3$.

The hyperboloid model, $\mathbb{H}^3$, is then the positive sheet of the unit hyperboloid

$$H^3 = \{x \in \mathbb{R}^{1,3} : \|x\|_L^2 = -1,\ x_0 > 0\},$$

with metric ρ_H given by

$$\cosh \rho_H(x, y) = -x_0 y_0 + x_1 y_1 + x_2 y_2 + x_3 y_3.$$

The positive light cone of $\mathbb{R}^{1,3}$ is the set

$$L = \{x \in \mathbb{R}^{1,3} : \|x\|_L^2 = 0,\ x_0 > 0\}.$$

Note that the positive sheet of H^3 is contained in the interior of L. The linear transfor-

mations preserving the Lorentzian norm and the interior of the positive light cone form $PO(1,3)$, the positive Lorentz group[1], this group represents the full set of isometries on $\mathbb{H}^3$; and $PSO(1,3)$, the positive special Lorentz group[2] (the index two subgroup of elements with determinant one), represents the orientation-preserving isometries.

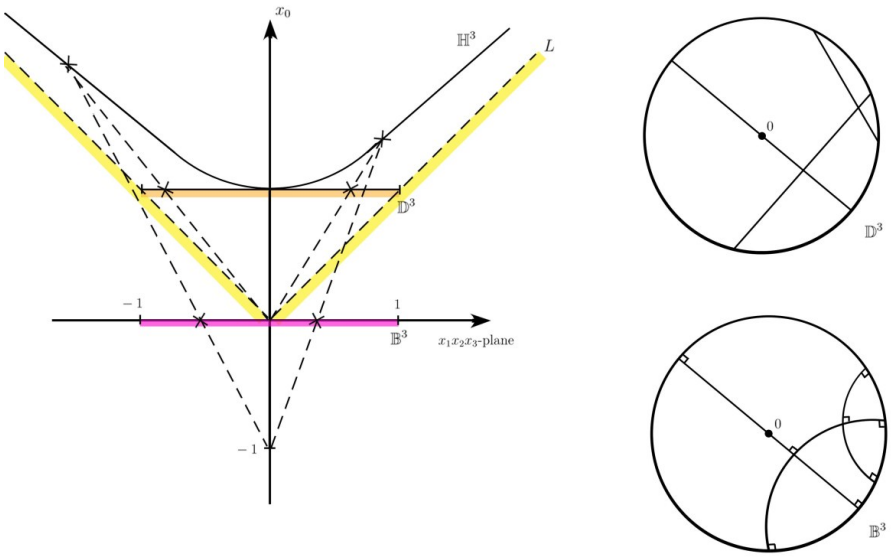


Fig. A.1: Cross-section of $\mathbb{H}^3$ and the projections onto $\mathbb{D}^3$ and $\mathbb{B}^3$.

The Klein Ball Model

One can rescale $\mathbb{H}^3$, by its vector components in $\mathbb{R}^{1,3}$, so that $x_0 = 1$; or, equivalently, project $\mathbb{H}^3$ toward the origin and onto the unit 3-ball, centred in the $x_0 = 1$ hyperspace. This gives the Klein ball (or projective disc) model of hyperbolic 3-space, denoted $\mathbb{D}^3$, which has a metric ρ_D given by

$$\cosh \rho_D(x,y) = \frac{1 - x.y}{\sqrt{1 - \|x\|_E^2}\sqrt{1 - \|y\|_E^2}}.$$

Where x, and x and y are considered vector elements of the ball $B(0,1)$, in $\mathbb{R}^3$ rather than $\mathbb{R}^{1,3}$; this is often explained through a further projection of the unit 3-ball centred in $x_0 = 1$ onto the unit 3-ball centred in $x_0 = 0$.

The retraction from $\mathbb{H}^3$ to $\mathbb{D}^3$ also retracts the action of $PSO(1,3)$, which is then also the representation group for $Isom^+(\mathbb{D}^3)$. It is a noted property of this space that the geodesics are Euclidean line segments in $\mathbb{R}^3$.

[1] This is also often denoted $O^+(1,3)$
[2] This is also often denoted $SO^+(1,3)$

The Poincaré Disc Model

Alternatively, projecting $\mathbb{H}^3$ towards $(-1,0,0,0)$ and onto the unit 3-ball, centred in the $x_0 = 0$ hyperspace, gives rise to the conformal ball model, $\mathbb{B}^3$, which has a metric ρ_B defined by

$$\cosh \rho_B(x,y) = 1 + \frac{2\|x-y\|_E^2}{(1-\|x\|_E^2)(1-\|y\|_E^2)}.$$

The geodesics in this space are then the circular arcs orthogonal to the boundary S^2, and isometries are then the compositions of reflections in spheres orthogonal to this boundary. Thus $Isom^+(\mathbb{B}^3)$ is a Möbius group.

Reflecting this model in the sphere $S((0,0,1),\sqrt{2})$ followed by the $x_3 = 0$ plane gives the upper half-space model, $\mathbb{U}^3$, which is seen in Section 2.2 and is commented on further below. This pair of reflections conjugate the isometry group of Möbius transformations, demonstrating that $Isom^+(\mathbb{B}^3)$ is represented by $\mathcal{M}$ or, equivalently, $PSL(2,\mathbb{C})$.

This model makes obvious the relation of ∂X to the Riemann sphere.

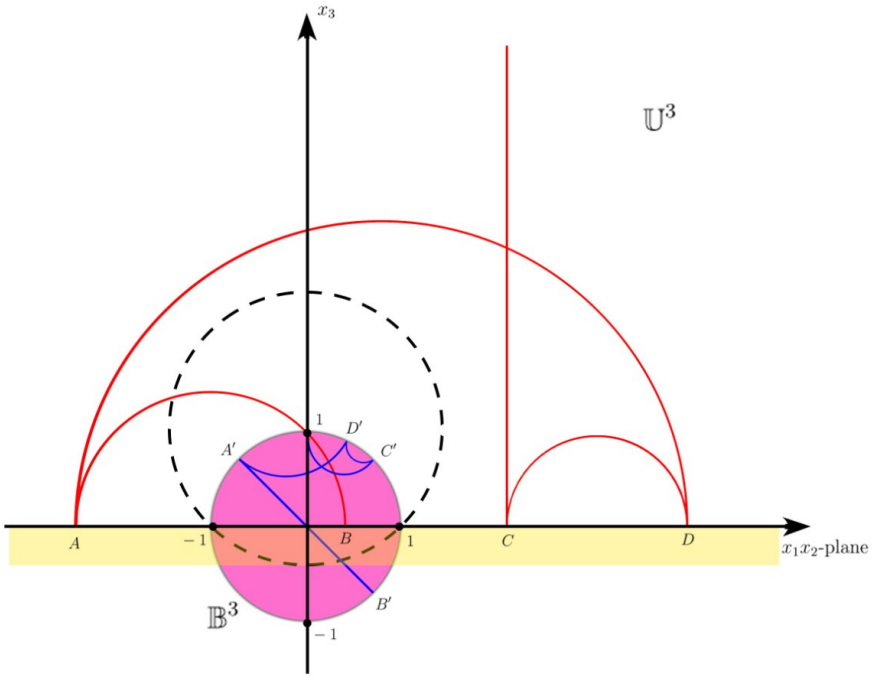


Fig. A.2: The mapping between conformal models.

The Upper Half-Space Model

The upper half-space model, $\mathbb{U}^3$, is mentioned in Section 2.2 and is the space

$$U^3 = \{x \in \mathbb{R}^3 \mid x_3 > 0\},$$

with the metric ρ_U, given by

$$\cosh \rho_U(x, y) = 1 + \frac{\|x - y\|_E^2}{2x_3 y_3}.$$

But it is worth noting that in many references, the metric is given in an alternative, differential form, as seen in Chapter 1.

$$ds = \frac{dx_1^2 + dx_2^2 + dx_3^2}{x_3^2}.$$

The Euclidean boundary of $\mathbb{U}^3$, or sphere at infinity, is

$$\partial\mathbb{U}^3 = \{x \in \mathbb{R}^3 \mid x_3 = 0\} \cup \{\infty\}.$$

And the geodesics of $\mathbb{U}^3$ are the intersection of Euclidean lines and circular arcs, orthogonal to the boundary, with U^3. Similarly, each geodesic plane is the intersection of U^3 with a Euclidean hemisphere (or plane) orthogonal to $\partial\mathbb{U}^3$; isometries being reflections in these spheres. In this way, an equivalence between $Isom^+(X)$ and $\mathcal{M}$, the Möbius transformations preserving the upper half-space, is clear.

The two models $\mathbb{B}^3$ and $\mathbb{U}^3$ are known as the conformal models, as both spaces accurately represent hyperbolic angles in their Euclidean embeddings.

A.2 Orientation-Preserving Isometries

As highlighted above and in Section 2.2, there is a deep interaction between the geometry of hyperbolic space, the group of Möbius transformations acting on the upper half space U^3, and the Möbius group of fractional linear transformations acting on the extended complex plane.

That is, each non-identity orientation-preserving isometry acting on $\mathbb{U}^3$ has a unique representation in $PSL(2, \mathbb{C})$

$$\pm \begin{bmatrix} a & b \\ c & d \end{bmatrix}, \; ad - bc = 1.$$

Which is conjugate to a single standard form, these conjugacy classes being parameterised by the non-zero complex parameter k

- $M_1 = \pm \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}$; and
- $M_k = \pm \begin{bmatrix} \sqrt{k} & 0 \\ 0 & 1/\sqrt{k} \end{bmatrix}$, where $k \notin \{0, 1\}$.

With the square of these matrix traces being invariant under conjugation this allows for a trace-squared classification that can also be related back to the geometric action of the isometry.

Definition A.2.1 (Definition 2.2.1, Section 2.2.2)**.** *Let f be any element of $\mathcal{M}\backslash\{I\}$, then*

1. *f is parabolic if and only if $tr^2(f) = 4$;*
2. *f is elliptic if and only if $tr^2(f) \in [0, 4)$;*
3. *f is hyperbolic if and only if $tr^2(f) \in (4, +\infty)$;*
4. *f is loxodromic if and only if $tr^2(f) \notin [0, +\infty)$.*

Mapping $PSL(2, \mathbb{C})$ to $PSO(1, 3)$

Let $\mathbb{V}$ denote the set of all 2×2 complex Hermitian matrices[3], then $\mathbb{V}$ is a 4-dimensional real vector space with basis:

$$B_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}; B_2 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}; B_3 = \begin{bmatrix} 0 & i \\ -i & 0 \end{bmatrix}; B_4 = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}.$$

And basis decomposition

$$M = \begin{bmatrix} \alpha & \gamma \\ \overline{\gamma} & \beta \end{bmatrix} = \frac{\alpha+\beta}{2}B_1 + \Re(\gamma)B_2 + \Im(\gamma)B_3 + \frac{\alpha-\beta}{2}B_4.$$

The function $\|x\|_V = \sqrt{-det(x)}$, defines an indefinite norm on $\mathbb{V}$, under which the vectors B_1, B_2, B_3, B_4 provide a Lorentz-orthonormal basis, indicating that the inner-product space $\mathbb{V}$ is isometric to $\mathbb{R}^{1,3}$. It follows that we can embed $\mathbb{H}^3$ in $\mathbb{V}$ in the same manner it is embedded in $\mathbb{R}^{1,3}$.

Consider the mapping:

$$\begin{aligned} \Phi &: A \mapsto A\mathbb{V}A^* \\ \Phi(A)(X) &: X \mapsto AXA^* \end{aligned}$$

Where $A \in SL(2, \mathbb{C})$. Notice that $\Phi(A)(X) = \Phi(-A)(X)$. It is easily shown that $\Phi(A)(X)$ is a linear function preserving the orientation and norm on $\mathbb{V}$; further, each $\Phi(A)(X)$ is a unique orientation-preserving isometry on $\mathbb{H}^3$, as embedded in $\mathbb{V}$.

Thus the function Φ is an injection of $SL(2, \mathbb{C})$ into $Isom^+(\mathbb{H}^3)$. With the above basis decomposition, elements in $PSO(1, 3)$ can then be generated that have actions on $\mathbb{R}^{1,3}$ equivalent to the action of $\Phi(A)$ on $\mathbb{V}$.

In this way we can lift Φ to a mapping from $SL(2, \mathbb{C})$ to $PSO(1, 3)$, carrying the classification above into $PSO(1, 3)$. A reversal of this construction is not difficult and provides a map from $PSO(1, 3)$ into $SL(2, \mathbb{C})/\{I, -I\}$.

[3] complex square matrices $M = (m_{ij})$ such that $m_{ij} = \overline{m}_{ji}$

A.3 Orientation-Reversing Isometries

The classification of orientation-reversing isometries is rarely covered in the literature. The best classification we have found is in the SnapPea source file `complex_length.c`, [59], where the lack of reference material is also noted.

Orientation-reversing transformations are dealt with in much the same way as orientation-preserving; being the composition of a single reflection in a plane and an orientation-preserving transformation (possibly the identity). For classification purposes this is usually achieved via a single reflection in the $x_2 = 0$ hyper-plane; in the conformal models this leads to fractional linear transformations on $\overline{z}$ instead of z,

$$z \mapsto \frac{a\overline{z} + b}{c\overline{z} + d}, \text{ where } ad - bc = 1.$$

The transformations can then be classified into conjugacy classes under their orientation-preserving component.

It should be noted that orientation-reversing loxodromic transformations do not exist.

A.4 Notes

The four models are commonly divided into two sets: the conformal models $\mathbb{U}^3$ and $\mathbb{B}^3$; and the projective models $\mathbb{H}^3$ and $\mathbb{D}^3$. In the general literature, notation for the specific models is not consistent, with it being common to simply use $\mathbb{H}^3$ or H^3 to denote hyperbolic space as pertaining to each individual text's particular model of interest.

The conformal models are the most common; prevalent for their links to complex analysis and inversive geometry, alongside their conformal nature and the simple parameterization they exhibit. Whereas the projective models find more relation to mathematical physics, specifically the hyperboloid model with Lorentzian space being the background of space-time calculations. The parallel between the definition of the hyperboloid model ($\|x\|_H^2 = -1$) and the sphere ($\|x\|_E^2 = 1$) is also noted; similarly the generation of the Klein model is comparable to the gnomic projection of the sphere.

Given the Euclidean representation of its geodesics, the Klein disc model finds special use in geometric representation; as seen in our use of it (following SnapPea). In fact, both disc models are often used for demonstrative and artistic displays (see the work of Escher, **www.mcescher.com**).

For a more detailed overall description on these models and the classification of Möbius transformations see [10], [44] and [55]. For a detailed discussion on the mapping given above, and explicit forms of these mapped matrices, see [16]; additionally, algorithms for converting matrices between $PSO(1,3)$ and $PSL(2,\mathbb{C})$ can be found in the SnapPea file `matrix_conversion.c`, [59].

B. SUPPORT FILES

This work centres around the development and use of the computer program SnappyD. As such, this thesis should come with an attached CD, or other accessible data-file, containing the computer program itself along with several variants and a number of support files.

This section briefly covers the execution of the SnappyD program and the contents of the support files.

B.1 SnappyD

SnappyD is a computer program developed for the construction and investigation of fundamental domains in reference to hyperbolic 3-orbifolds. It is named in reference to the Dirichlet subroutines of Jeffrey Week's SnapPea program, [59], upon which it is solidly based.

Where other work seems to focus on expanding the accuracy of SnapPea's overall computations, with a focus on the accurate triangulation of link complements, or providing domain construction routines for a narrow class of highly accurate groups; SnappyD follows the construction of Dirichlet domains within the accuracy of double-precision arithmetic, and allows for the successful construction of domains that would be otherwise unacceptable within the confines of the common linear representation of hyperbolic polyhedrons.

SnapPea and SnappyD are covered in Chapters 3 and 4, respectively.

B.1.1 Using the Program

To facilitate fast use of the program, SnappyD works as a simple executable file, reading from two input files and writing to a single output file. These three files are all of notepad (`.txt`) format and are summarised:

- The input file "`snappyD_input.txt`" provides a list of parameter sets, and several other settings for each individual construction to be based upon. Each line in the file is of the form:

  $$n \quad E_V \quad p \quad q \quad RE(\gamma) \quad IM(\gamma)$$

  Where:

  - n is an integer reference number for tracking the results of an individual group or set of groups.
  - $E_V \in [3, 15]$ is an integer, setting vertex epsilon $\varepsilon_V = 10^{-E_V}$; and
  - (p, q, γ) is the computational parameter set of interest;

  The input file can contain any number of groups, but each must be given on a separate line, with no additional formatting.

- The input file "`snappyD_print_defaults.txt`" contains only one meaningful line of data; an array of 1's and 0's that are used to indicate the form and complexity of output desired.

  This array comprises the first line in the file and is the only line read by SnappyD; the remainder of the file providing a description of the various options. This file is not necessary for the success of the program, but without it the output will only be given in its simplest form.

- The output file is titled "`snappyD_OUTPUT_XX.txt`", where XX is a series of letters to describe the particular variant of SnappyD used, see Section B.1.2 below. The output file is re-generated each time (the specific variant of) the program is used.

Originally we had considered designing a simple user interface window. But both building it, and using it, seemed unnecessarily time consuming.

To use the program, all that is strictly needed is the executable file and a "`snappyD_input.txt`" file. As mentioned the output file is reconstructed each execution, and will over-write any file with the same name; if you wish to save the output, then relabeling it is highly encouraged.

B.1.2 Variants

There are a number of variants to the SnappyD program. The core version we use is `SnappyD_CPT.exe`, which attempts to construct a polyhedron as outlined in Sections 4.3 and 4.4. Additionally there are:

1. `SnappyD_CPT_NM.exe` - which does not maximise the injectivity radius;
2. `SnappyD_nCPT.exe` - which does not attempt to "compute products twice" before leaving while-loop two (see Section 4.3.2);
3. `SnappyD_nCPT_NM.exe` - which combines the alterations of the two previous variants; and
4. `SnappyD_CPe_NM.exe` - which does not maximise the injectivity radius, but will attempt to sequentially compute more products in while-loop two, up to an internally set limit (currently 5).

The core, `SnappyD_CPT.exe`, version of SnappyD is the one we have used for the majority of our work, with the other variants existing mostly for comparison. The support files only include actual executables for variants 3 and 4 above; though code is provided for the others variants.

As outlined in Section 4.3.2, `SnappyD_CPe_NM.exe` can take considerably longer to progress constructions than the other variants. For this reason we have purposely removed its access to the basepoint manipulation routines; and we suggest running it only on small data sets and moderate precision settings ($\varepsilon_V = 10^{-7}$). We note that unlike the other variants, `SnappyD_CPe_NM.exe` will exit while-loop 2 if the identification is confirmed on the constructed polyhedron.

We also provide the executable `Snappy.exe`, which reads from the input file "`snappy_input.txt`". The input file is identical to the SnappyD input, except that it does not take an E_V input, instead running at with default $\varepsilon_V = 10^{-7}$. This is a simple "wrapper" of the original Dirichlet subroutines as found in SnapPea, the only alteration that has been made is capping while-loop two after 50 iterations. This program writes to the file "`snappy_output.txt`"; and there are no output options for this program - it simply lists the parameter set and either the constructions volume or "FAIL".

B.2 Progress Tracking and Output

The output for SnappyD is written into the file "`snappyD_OUTPUT_XX.txt`", where "`XX`" denotes the variant used. The exact details written in the output can vary dramatically, depending on the "`snappyD_print_defaults.txt`" settings: from a simple statement of the parameters with a volume if successful, or "FAILS" otherwise; to a full step-by-step description of the entire process - detailing the separate functions entered, the group elements being generated, and each attempt at classifying and verifying the polyhedron.

The only output that can never be turned off is the announcement of fatal errors in the construction process.

This output runs off a comprehensive and descriptive progress tracking system that we have built into the code. Progress through the construction is tracked by a series of Boolean values - allowing successes and errors to be announced when they occur; along with optional tracking of individual construction processes, right down to detailing the flow through the individual routines.

As noted earlier, this program is quite fast, and a simple output selection is ideal for streaming through large lists of generator inputs.

A full description of the various output options would take considerable space, so we instead refer the reader to the "`snappyD_print_defaults.txt`" file; and suggest attempting to use the program to complete constructions across various "print" settings - using a single parameter set.

B.3 Reference Information

In addition to the program, the support files include a variety of files relevant to the information given in this thesis.

Reference Data

The data-files, both input and output, for all tables seen in this thesis are given in the support files. These have been given in a simple output style focusing on the interests displayed here.

Files for generating the individual tables should also be available, in `.txt`, `.pdf` and `.tex` formats.

Reference Code

Along with the executables, the support files should also include the code used to generate the core program and its variants. Like SnapPea they are all written in the `C` programming language.

For additional reference we also provide the base code for the SnapPea kernel (code), both in a readable `.pdf` and a usable `.c` (and `.h`) formats.

The SnapPea kernel is also available online and is free to distribute; [59].

C. TABLES

Here we provide detailed tables from our use of SnappyD in constructing the fundamental domains for groups generated by two elliptic transformations. The results from these tables are discussed, and applied, in Chapter 5.

Using SnappyD, we have investigated each of the groups listed in references [23], [60] and [62]; through the attempted construction of their fundamental polyhedrons. For ease in cross-referencing, in the tables given here we have maintained the group order as per the tables seen in the individual references the groups are taken from.

The tables here display information as discussed in Section 5; listing the parameter sets, basepoint movement, deviation and volume if finite; with a "Free" note for hyperideal constructions[1].

All results shown are for the standard version of SnappyD; the generated files for which can be found in the SnappyD support files. A discussion on the various versions of SnappyD can be found in Appendix B.1.2.

Table Structure

The structure of the tables given in this Appendix follows the form seen in Section 5.1.1. Specifically, each row of the table corresponds to the constructions based upon one particular set of "computational" input parameters. With the parameter sets corresponding to each inter-related class of groups being separated into their own box of rows within the greater table.

In some circumstances a row will contain more than one line of data, this corresponds to the data "tremors" discussed in Section 5.2.3. In these cases, the overall or dominant result for the particular parameter set is given in the top line, with additional lines listing the outlier results and their corresponding values.

The columns describe these constructions accordingly:

- n - a reference to the particular class of group parameters that is under consideration.
- (p, q) - the respective order of generators, as input into the program.
- γ - the specific γ-parameter used. These are given only to 5 decimal places, due to space constraints; the actual γ-values used are input at full double precision,

[1] As mentioned in Chapter 5, hyperideal constructions are generally not free and are instead likely to be among those groups listed in Section 2.3.1.

having been calculated from the polynomials seen in the tables of Chapter 5, where possible.

- ε_V - the band of vertex resolution k-values for which the construction holds; each resolution setting is of the form 10^{-k}, for integer k. See Section 3.2.3. Note that we test all values within the bounds suggested by SnapPea: $3 \leq k \leq 15$.
- deviation - the largest deviation of a side-pairing (generating) transformation found in the final constructed polyhedron. See Section 3.3.1.
- B_0 - the sum of Euclidean movements induced on the basepoint in an attempt to maximise the injectivity radius, as discussed in Section 4.4.2.
- $||B_0||_E^2$ - the Euclidean measure of distance the tracked basepoint has moved (given to 2 decimal places).
- $S/V/E$ - the total number of (real) sides, vertices and edges seen in the final polyhedron. These counts do not include purely hyperideal components.
- covolume - the hyperbolic volume of the particular construction.

For most purposes, attention can be directed to the second and third rows, consisting of the parameter data; and the last row, which states the volume of the construction. If the final construction is a hyperideal polyhedron, then instead of listing a volume it is marked as "Free?" to match the general SnappyD output; as noted in the text, these groups do not generally correspond to actual free groups. Where there is a failure in obtaining a successful construction, the row is marked appropriately and to save space, where possible, multiple failing rows have been compounded; this should not add any ambiguity to the results.

Marked Results

Certain components in each table have also been highlighted. Most noticeably, the volumes used in the main tables of Chapter 5 have been blocked in yellow; these correspond to the construction within a set of inter-related parameters that displays the lowest deviation. Where multiple results share this deviation, the more successful construction (determined by spread of ε_V-values) is highlighted.

We also circle specific data with coloured boxes.

- Deviations marked in a yellow box correspond to the lowest deviation within the class of inter-related parameter sets, disregarding parallax outliers; if a parallax outlier is the lowest deviation, then it is marked in a blue box along with its corresponding volume.
- Volumes corresponding to constructions that fail the cycle condition are marked out by an orange box.
- Basepoint distance measures that equal or exceeded their bounds, as given in Section 4.4.2, are boxed in red.

The relevance of these results is covered in Chapter 5.

C.1 Orbifold Generalised Triangle Groups

These are the groups used in the detailed example that comprises Section 5.1. Here, in Table C.1, we detail the general output under the standard SnappyD; we do this for the sake of completeness of the Appendix, with the information given here having already appeared in the mentioned section. We do not list any greater tables detailing the devolution or other results noted previously; the output details for which should be available in the support files.

Note that these groups can also be found in Table 5.3 of [62]; but here they are taken from [60].

Unlike the other tables in this Appendix, we have highlighted multiple volumes in each class of parameters; each volume corresponding to one used in Table 5.1. This data is also unique for being the only data set that we covered in this thesis that contains parameter sets where $p = q$.

C.2 Disc Covering Groups

These groups are taken from [23] and correspond to investigations on commutator planes. In the reference, a set of groups, $G_{p,n}$, is given for each of the five $(p, 2)$-commutator planes, $p = 3, ..., 7$.

We give output for these groups in Tables C.2 - C.9, separating them into $G_{p,n}$ sets as per the reference.

Note that some of these groups are also seen in the results of [62].

C.3 Boundary Groups

These groups are those taken from the tables of [62]; we consider all groups, with the exception of Tables 5.2, 5.3, 5.10 and 5.14, as the group data provided in those tables already appears in references [23] and [60], so has been covered above.

We detail these groups in Tables C.10 - C.26. Note that the referencing systems in several of the tables given in [62] are related to knot descriptions, which we forgo here; reverting to a standard numerical reference for all groups.

Within these Tables there are still 10 individual parameter sets that have already been covered in the disc-covering groups of [23]. We have not removed them from these tables, so as to preserve the structure of referencing and comparison purposes, and instead reference the above results from within the table.

C.4 Notes

- The finite constructions of Tables C.2 - C.8, C.10 - C.15, C.21, C.23, C.25 and C.26 are summarised in Tables 5.12 - 5.15 of Section 5.3.

- The hyperideal constructions of Tables C.2 - C.9, C.16 - C.19 and C.24 are summarised in Table 5.17.
- All other finite and hyperideal constructions are covered in Tables 5.16 and 5.18 respectively.
- The group parameter sets over which SnappyD fails to output a successfully constructed domain are covered briefly in Section 5.2.1.
- In displaying the γ-values in [62] to a chosen number of significant figures, the author seems to have made some minor errors that we have corrected in our tables.

n	(p,q)	γ	ε_V	deviation	B	$\|\|B\|\|_E^2$	S/V/E	covolume
1	$(3,3)$	$-1.77184+1.11514i$	$3\leftrightarrow 13$	2.55×10^{-15}	$(-0.04,0.31,0.21)$	0.14	10/7/15	0.264774439883296
		$-1.77184-1.11514i$	$4\leftrightarrow 14$	5.33×10^{-15}	$(0.24,0.31,0.17)$	0.18	14/12/24	0.264774439883378
	$(3,2)$	$-2.41964+0.60629i$	$3\leftrightarrow 14$	2.22×10^{-15}	$(0.19,0.25,0.14)$	0.12	14/12/24	0.132387219941707
		$-2.41964-0.60629i$	$4\leftrightarrow 14$	2.44×10^{-15}	$(0.25,0.14,0.22)$	0.13	8/9/15	0.132387219941693
		$-0.58036+0.60629i$	$4\leftrightarrow 13$	1.73×10^{-14}	$(-0.17,0.27,-0.02)$	0.10	22/24/44	0.132318149806017
		$-0.58036-0.60629i$	$4\leftrightarrow 13$	2.98×10^{-14}	$(-0.17,0.25,0.11)$	0.10	22/24/44	0.132403276354546
	$(2,3)$	$-2.41964+0.60629i$	$3\leftrightarrow 14$	2.44×10^{-15}	$(0.14,0.21,-0.14)$	0.08	14/12/24	0.132387219941619
		$-2.41964-0.60629i$	$3\leftrightarrow 14$	2.66×10^{-15}	$(0.14,0.21,0.14)$	0.08	14/12/24	0.132387219941758
		$-0.58036+0.60629i$	$5\leftrightarrow 12$	2.30×10^{-14}	$(0.05,0.21,0.24)$	0.10	22/24/44	0.132426403921074
		$-0.58036-0.60629i$	$5\leftrightarrow 12$	2.41×10^{-14}	$(0.15,0.19,0.15)$	0.08	22/24/44	0.132158766596323
2	$(3,3)$	$-0.68055+1.63317i$	$5\leftrightarrow 13$	1.53×10^{-14}	$(0.19,0.45,0.19)$	0.27	16/14/28	0.661714937315159
		$-0.68055-1.63317i$	$5\leftrightarrow 13$	1.42×10^{-14}	$(0.34,0.40,0.24)$	0.34	16/14/28	0.661714937315199
	$(3,2)$	$-0.11535-0.58974i$	$6\leftrightarrow 13$	3.17×10^{-13}	$(-0.11,0.30,0.19)$	0.14	24/32/54	0.330857468657992
		$-0.11535+0.58974i$	$4\leftrightarrow 13$	2.13×10^{-14}	$(-0.11,0.35,-0.04)$	0.14	24/32/54	0.330857468657719
		$-2.88465-0.58974i$	$4\leftrightarrow 14$	2.71×10^{-14}	$(0.33,0.03,0.36)$	0.24	22/24/44	0.330802402206134
		$-2.88465+0.58974i$	$4\leftrightarrow 14$	5.02×10^{-14}	$(0.33,0.29,0.22)$	0.24	22/24/44	0.330264970254204
	$(2,3)$	$-0.11535-0.58974i$	$4\leftrightarrow 12$	5.85×10^{-14}	$(0.30,0.24,0.19)$	0.18	24/32/54	0.330857468657576
		$-0.11535+0.58974i$	$4\leftrightarrow 12$	2.89×10^{-14}	$(0.21,0.26,0.26)$	0.18	24/32/54	0.330857468657535
		$-2.88465-0.58974i$	$4\leftrightarrow 13$	2.89×10^{-14}	$(0.29,0.22,0.23)$	0.18	22/24/44	0.330788586067824
		$-2.88465+0.58974i$	$5\leftrightarrow 13$	1.47×10^{-14}	$(-0.04,0.08,0.40)$	0.17	22/24/44	0.330717107428080
3	$(3,3)$	$0.19927+1.58951i$	$5\leftrightarrow 11$	1.35×10^{-14}	$(0.31,0.51,0.14)$	0.38	20/24/42	0.982766244039437
		$0.19927-1.58951i$	$6\leftrightarrow 13$	1.97×10^{-13}	$(0.42,0.39,0.33)$	0.44	20/24/42	0.982766244039443
	$(3,2)$	$-3.13846+0.48506i$	$4\leftrightarrow 14$	9.77×10^{-15}	$(-0.28,0.09,0.39)$	0.24	18/18/34	0.492361631010875
		$-3.13846-0.48506i$	$4\leftrightarrow 14$	9.33×10^{-15}	$(-0.28,0.28,0.29)$	0.24	18/18/34	0.492361631011032
		$0.13846+0.48506i$	$4\leftrightarrow 13$	6.95×10^{-14}	$(-0.08,0.39,-0.05)$	0.16	24/32/54	0.491693878116127
		$0.13846-0.48506i$	$4\leftrightarrow 13$	6.14×10^{-14}	$(-0.08,0.31,0.24)$	0.16	24/32/54	0.491677333327298
	$(2,3)$	$-3.13846+0.48506i$	$4\leftrightarrow 13$	7.44×10^{-15}	$(0.07,0.06,0.36)$	0.14	18/18/34	0.492361631010930
		$-3.13846-0.48506i$	$4\leftrightarrow 13$	3.55×10^{-15}	$(0.34,0.27,0.23)$	0.24	18/18/34	0.492361631010942
		$0.13846+0.48506i$	$4\leftrightarrow 12$	9.04×10^{-14}	$(-0.26,0.21,0.26)$	0.18	24/32/54	0.491677333327583
		$0.13846-0.48506i$	$4\leftrightarrow 13$	4.00×10^{-14}	$(-0.19,0.33,0.20)$	0.18	24/32/54	0.491484440296781
4	$(3,3)$	$-2.80606+1.15645i$	$4\leftrightarrow 13$	1.61×10^{-14}	$(0.31,0.41,0.35)$	0.38	18/18/34	1.090072770048188
		$-2.80606-1.15645i$	$6\leftrightarrow 12$	8.88×10^{-15}	$(0.58,0.24,0.46)$	0.61	18/23/39	1.090072770048278
	$(3,2)$	$-0.89704+0.95897i$	$5\leftrightarrow 13$	7.66×10^{-15}	$(0.14,0.43,0.06)$	0.21	18/16/32	0.544958718160841
		$-0.89704-0.95897i$	$5\leftrightarrow 13$	1.73×10^{-14}	$(0.16,0.43,-0.04)$	0.21	18/16/32	0.544962897261796
		$-2.10296+0.95897i$	$4\leftrightarrow 13$	1.38×10^{-14}	$(0.15,0.37,0.13)$	0.18	18/17/33	0.545036385024126
		$-2.10296-0.95897i$	$5\leftrightarrow 14$	7.11×10^{-15}	$(0.45,0.30,0.28)$	0.38	18/17/33	0.544739451493277
	$(2,3)$	$-0.89704+0.95897i$	$3\leftrightarrow 14$	3.33×10^{-15}	$(0.08,0.34,0.05)$	0.13	18/16/32	0.545209773747604
		$-0.89704-0.95897i$	$5\leftrightarrow 14$	6.55×10^{-15}	$(0.12,0.35,0.15)$	0.16	18/16/32	0.544974108710528
		$-2.10210+0.95897i$	$5\leftrightarrow 13$	6.44×10^{-15}	$(0.23,0.07,0.37)$	0.19	18/17/33	0.544786544898183
		$-2.10210-0.95897i$	$4\leftrightarrow 13$	8.22×10^{-15}	$(0.19,0.33,0.08)$	0.15	18/17/33	0.545036385024131
5	$(3,3)$	$0.84236+1.35530i$	$4\leftrightarrow 12$	4.85×10^{-14}	$(0.37,0.54,0.10)$	0.45	20/26/44	1.232257018353899
		$0.84236-1.35530i$	$4\leftrightarrow 12$	3.62×10^{-14}	$(0.45,0.37,0.39)$	0.50	20/26/44	1.232257018354058
	$(3,2)$	$0.29843+0.37680i$	$4\leftrightarrow 11$	3.21×10^{-13}	$(-0.07,0.41,-0.06)$	0.17	26/36/60	0.616139999305901
		$0.29843-0.37680i$	$4\leftrightarrow 13$	1.71×10^{-13}	$(-0.07,0.31,0.27)$	0.17	26/36/60	0.616139999304282
		$-3.29843+0.37680i$	$5\leftrightarrow 13$	2.36×10^{-13}	$(-0.30,0.09,0.44)$	0.29	24/26/48	0.616097893053077
		$-3.29843-0.37680i$	$5\leftrightarrow 13$	1.45×10^{-14}	$(-0.31,0.32,0.32)$	0.30	24/26/48	0.615425771551932
	$(2,3)$	$0.29843+0.37680i$	$4\leftrightarrow 12$	7.59×10^{-14}	$(-0.23,0.24,0.26)$	0.18	26/36/60	0.616139999303372
		$0.29843-0.37680i$	$3\leftrightarrow 12$	1.12×10^{-13}	$(-0.18,0.33,0.22)$	0.19	26/36/60	0.616024211234735
		$-3.29843+0.37680i$	$5\leftrightarrow 14$ 5	4.62×10^{-14} 4.86×10^{-14}	$(0.14,0.05,0.40)$	0.18	24/26/48	0.616128509177734 0.616128509177205
		$-3.29843-0.37680i$	$3\leftrightarrow 13$	1.91×10^{-14}	$(0.40,0.29,0.29)$	0.33	24/26/48	0.616128509177068
6	$(3,3)$	$-3.19690+0.90182i$	$6\leftrightarrow 11$	2.19×10^{-13}	$(-0.33,0.27,0.54)$	0.47	20/20/38	1.696240561334349
		$-3.19690-0.90182i$	$5\leftrightarrow 13$	6.10×10^{-14}	$(-0.33,0.28,0.54)$	0.48	20/20/38	1.696240561334381
	$(3,2)$	$-1.92469+1.06173i$	$4\leftrightarrow 13$	5.00×10^{-14}	$(0.14,0.39,0.13)$	0.19	22/24/44	0.848365358302074
		$-1.92469-1.06173i$	$4\leftrightarrow 12$	5.66×10^{-15}	$(0.55,0.21,0.38)$	0.49	22/26/46	0.847936929164838
		$-1.07530+1.06173i$	$4\leftrightarrow 13$	2.29×10^{-13}	$(0.20,0.45,0.07)$	0.25	26/28/52	0.847904607792574
		$-1.07530-1.06173i$	$5\leftrightarrow 14$	2.99×10^{-13}	$(0.27,0.44,0.13)$	0.28	26/28/52	0.847904607792901
	$(2,3)$	$-1.92469+1.06173i$	$4\leftrightarrow 14$	8.44×10^{-15}	$(0.16,0.12,0.37)$	0.18	20/20/38	0.847979373190956
		$-1.92469-1.06173i$	$3\leftrightarrow 13$	5.46×10^{-14}	$(0.19,0.35,0.08)$	0.16	22/24/44	0.847921705316328
		$-1.07531+1.06173i$	$5\leftrightarrow 14$	2.16×10^{-13}	$(-0.05,0.21,0.40)$	0.21	26/28/52	0.849015226143611
		$-1.07531-1.06173i$	$5\leftrightarrow 14$	1.80×10^{-13}	$(0.19,0.37,0.15)$	0.20	26/28/52	0.848229498305300
7	$(3,3)$	$0.68088+1.73330i$	$5\leftrightarrow 12$	3.06×10^{-13}	$(0.57,0.60,0.10)$	0.69	24/30/52	2.446924195119881
		$0.68088-1.73330i$	$5\leftrightarrow 13$	8.67×10^{-13}	$(0.68,0.40,0.42)$	0.80	24/30/52	2.446924195121926
	$(3,2)$	$0.27988+0.48692i$	$4\leftrightarrow 12$	4.98×10^{-13}	$(0.16,0.50,-0.04)$	0.28	26/34/58	1.222531911485409
		$0.27988-0.48692i$	$4\leftrightarrow 12$	2.95×10^{-13}	$(0.16,0.50,0.03)$	0.28	26/34/58	1.222412331605630
		$-3.27988+0.48692i$	$6\leftrightarrow 12$	2.07×10^{-12}	$(-0.55,0.07,0.47)$	0.53	28/30/56	1.220360017152392
		$-3.27988-0.48692i$	$5\leftrightarrow 13$	1.64×10^{-12}	$(-0.32,0.32,0.35)$	0.33	28/30/56	1.221755633824927
	$(2,3)$	$0.27988+0.48692i$	$5\leftrightarrow 12$	7.88×10^{-13}	$(0.10,0.40,0.13)$	0.19	26/36/60	1.221708487472495
		$0.27988-0.48692i$	$4\leftrightarrow 12$	2.44×10^{-13}	$(0.12,0.40,0.18)$	0.21	26/34/58	1.223512318246351
		$-3.27988+0.48692i$	$4\leftrightarrow 13$	1.71×10^{-12}	$(0.14,0.07,0.43)$	0.21	28/30/56	1.221624951511321
		$-3.27988-0.48692i$	$5\leftrightarrow 13$	1.09×10^{-12}	$(0.48,0.32,0.53)$	0.61	28/30/56	1.220360017153322

Tab. C.1: Data from [60].

n	(p,q)	γ	ε_V	deviation	B	$\|\|B\|\|_E^2$	S/V/E	covolume
1	$(3,2)$	$-1.00000+0.00000i$	$3\leftrightarrow 14$	1.07×10^{-14}	$(0.73,0.48,-0.49)$	1.01	5/1/5	Free?
		$-2.00000+0.00000i$	$3\leftrightarrow 14$	3.11×10^{-15}	$(0.78,0.28,0.75)$	1.25	5/1/5	Free?
	$(2,3)$	$-1.00000+0.00000i$	$3\leftrightarrow 14$	5.66×10^{-15}	$(0.80,0.53,0.51)$	1.19	5/1/5	Free?
		$-2.00000+0.00000i$	$3\leftrightarrow 13$	8.10×10^{-15}	$(0.78,0.53,0.54)$	1.19	5/1/5	Free?
2	$(3,2)$	$-0.38197+0.00000i$	$3\leftrightarrow 13$	6.77×10^{-15}	$(0.76,0.19,0.75)$	1.18	5/1/5	Free?
		$-2.61803+0.00000i$	$3\leftrightarrow 13$	4.66×10^{-15}	$(0.78,-0.18,0.50)$	0.88	5/1/5	Free?
	$(2,3)$	$-0.38197+0.00000i$	$3\leftrightarrow 13$	9.99×10^{-15}	$(0.71,0.51,0.48)$	0.99	5/1/5	Free?
		$-2.61803+0.00000i$	$3\leftrightarrow 14$	9.44×10^{-15}	$(0.84,0.17,0.19)$	0.78	5/1/5	Free?
3	$(3,2)$	$-1.50000-0.60666i$	$3\leftrightarrow 13$	2.89×10^{-15}	$(-0.03,0.24,0.05)$	0.06	12/9/19	0.077777592962837
		$-1.50000+0.60666i$	$3\leftrightarrow 13$	2.66×10^{-15}	$(-0.03,0.24,-0.05)$	0.06	12/9/19	0.077777592962811
	$(2,3)$	$-1.50000-0.60666i$	$4\leftrightarrow 13$	8.66×10^{-15}	$(0.02,0.27,0.10)$	0.08	18/16/32	0.039015451336356
		$-1.50000+0.60666i$	$3\leftrightarrow 14$	failure. possibly generates a finite-sheeted cover				
4	$(3,2)$	$-0.21190+0.40136i$	$3\leftrightarrow 13$	9.41×10^{-14}	$(0.04,0.24,0.16)$	0.09	22/20/40	0.040851193893052
			3	7.66×10^{-14}				0.040825387267666
			13	8.68×10^{-14}				0.040851193892787
		$-0.21190-0.40136i$	$4\leftrightarrow 12$	3.32×10^{-13}	$(0.11,0.30,0.06)$	0.11	22/20/40	0.040890367881093
		$-2.78810+0.40136i$	$5\leftrightarrow 13$	1.40×10^{-14}	$(0.23,0.09,0.16)$	0.09	22/20/40	0.040848491993216
		$-2.78810-0.40136i$	$4\leftrightarrow 14$	9.33×10^{-15}	$(0.26,0.17,0.07)$	0.10	22/20/40	0.040890367870632
			14	8.33×10^{-15}				0.040890367870668
	$(2,3)$	$-0.21190+0.40136i$	$3\leftrightarrow 12$	3.33×10^{-13}	$(0.07,0.22,0.25)$	0.12	22/20/40	0.040848492004980
			3	4.28×10^{-13}				0.040822685385170
		$-0.21190-0.40136i$	$4\leftrightarrow 13$	1.44×10^{-13}	$(-0.01,-0.03,0.26)$	0.07	22/20/40	0.040890367872991
		$-2.78810+0.40136i$	$3\leftrightarrow 13$	1.55×10^{-14}	$(0.05,0.10,-0.21)$	0.06	22/20/40	0.040884964081833
		$-2.78810-0.40136i$	$4\leftrightarrow 13$	4.42×10^{-14}	$(0.05,0.10,0.21)$	0.06	22/20/40	0.040884964083423
5	$(3,2)$	$-0.58036-0.60629i$	$4\leftrightarrow 13$	2.98×10^{-14}	$(-0.17,0.25,0.11)$	0.10	22/24/44	0.132403276354546
		$-0.58036+0.60629i$	$4\leftrightarrow 13$	1.73×10^{-14}	$(-0.17,0.27,-0.02)$	0.10	22/24/44	0.132318149806017
		$-2.41964-0.60629i$	$4\leftrightarrow 14$	2.44×10^{-15}	$(0.25,0.14,0.22)$	0.13	8/9/15	0.132387219941693
		$-2.41964+0.60629i$	$3\leftrightarrow 14$	2.22×10^{-15}	$(0.19,0.25,0.14)$	0.12	14/12/24	0.132387219941707
	$(2,3)$	$-0.58036-0.60629i$	$5\leftrightarrow 12$	2.41×10^{-14}	$(0.15,0.19,0.15)$	0.08	22/24/44	0.132158766596323
		$-0.58036+0.60629i$	$5\leftrightarrow 12$	2.30×10^{-14}	$(0.05,0.21,0.24)$	0.10	22/24/44	0.132426403921074
		$-2.41964-0.60629i$	$3\leftrightarrow 14$	2.66×10^{-15}	$(0.14,0.21,0.14)$	0.08	14/12/24	0.132387219941758
		$-2.41964+0.60629i$	$3\leftrightarrow 14$	2.44×10^{-15}	$(0.14,0.21,-0.14)$	0.08	14/12/24	0.132387219941619
6	$(3,2)$	$-1.12256-0.74486i$	$3\leftrightarrow 14$	8.66×10^{-15}	$(0.01,0.30,0.06)$	0.10	20/22/40	0.157135965589727
		$-1.12256+0.74486i$	$3\leftrightarrow 14$	9.10×10^{-15}	$(-0.03,0.30,0.08)$	0.10	20/22/40	0.157403816793161
		$-1.87744-0.74486i$	$4\leftrightarrow 13$	9.66×10^{-15}	$(0.01,0.26,-0.11)$	0.08	24/26/48	0.157117893796116
		$-1.87744+0.74486i$	$3\leftrightarrow 13$	9.77×10^{-15}	$(0.01,0.27,0.11)$	0.08	24/26/48	0.157117893796155
	$(2,3)$	$-1.12256-0.74486i$	$4\leftrightarrow 14$	8.88×10^{-15}	$(-0.08,0.26,0.08)$	0.08	20/22/40	0.157117893796278
			14	8.10×10^{-15}				0.157117893796126
		$-1.12256+0.74486i$	$3\leftrightarrow 13$	6.11×10^{-15}	$(-0.08,0.26,-0.08)$	0.08	20/22/40	0.157117893796012
		$-1.87744-0.74486i$	$3\leftrightarrow 13$	8.66×10^{-15}	$(0.00,0.25,0.04)$	0.06	24/26/48	0.157117893796077
		$-1.87744+0.74486i$	$3\leftrightarrow 13$	4.66×10^{-15}	$(0.01,0.25,-0.00)$	0.06	24/26/48	0.157117893796128
7	$(3,2)$	$-0.33764-0.56228i$	$3\leftrightarrow 12$	9.41×10^{-14}	$(0.16,0.20,0.20)$	0.11	22/26/46	0.157285227171242
			3	1.21×10^{-13}				0.157285227171591
			4	failure.				
		$-0.33764+0.56228i$	$4\leftrightarrow 12$	1.80×10^{-13}	$(0.18,0.26,-0.05)$	0.11	24/28/50	0.157058545714813
		$-2.66236-0.56228i$	$4\leftrightarrow 14$	1.09×10^{-14}	$(0.26,0.02,0.30)$	0.16	20/22/40	0.156850042593146
		$-2.66236+0.56228i$	$4\leftrightarrow 14$	1.67×10^{-14}	$(0.26,0.25,0.18)$	0.16	20/22/40	0.156983968194752
	$(2,3)$	$-0.33764-0.56228i$	$4\leftrightarrow 13$	4.27×10^{-13}	$(0.18,0.24,0.09)$	0.10	22/26/46	0.157201560484605
		$-0.33764+0.56228i$	$4\leftrightarrow 13$	5.60×10^{-13}	$(0.10,0.16,0.27)$	0.11	24/28/50	0.156960920774735
			13	2.93×10^{-13}				0.156960920774842
		$-2.66236-0.56228i$	$4\leftrightarrow 13$	3.33×10^{-15}	$(0.20,0.19,0.19)$	0.11	20/22/40	0.156983968194460
		$-2.66236+0.56228i$	$4\leftrightarrow 14$	8.88×10^{-15}	$(0.20,0.18,-0.19)$	0.11	20/22/40	0.156983968194785
8	$(3,2)$	$-0.92362+0.81470i$	$3\leftrightarrow 13$	3.33×10^{-15}	$(0.02,0.35,0.07)$	0.13	18/16/32	0.255445410566101
			3				16/15/29	0.255512516393925
			4	failure.				
		$-0.92362-0.81470i$	$4\leftrightarrow 13$	3.66×10^{-15}	$(-0.00,0.31,0.11)$	0.11	18/16/32	0.255177950088320
		$-2.07638+0.81470i$	$4\leftrightarrow 13$	2.05×10^{-13}	$(0.17,0.12,0.21)$	0.09	26/28/52	0.127673032901201
		$-2.07638-0.81470i$	$4\leftrightarrow 13$	3.64×10^{-14}	$(0.09,0.20,0.08)$	0.06	26/26/50	0.127456265703298
	$(2,3)$	$-0.92362+0.81470i$	$3\leftrightarrow 14$	8.79×10^{-15}	$(-0.05,0.28,0.05)$	0.08	18/16/32	0.254944838148437
		$-0.92362-0.81470i$	$5\leftrightarrow 14$	6.13×10^{-15}	$(-0.02,0.29,0.11)$	0.10	18/16/32	0.254967508911403
		$-2.07638+0.81470i$	$3\leftrightarrow 13$	3.13×10^{-13}	$(0.19,0.18,0.11)$	0.08	26/28/52	0.127612863244017
			3	3.19×10^{-13}				0.127612863244156
		$-2.07638-0.81470i$	$3\leftrightarrow 13$	1.21×10^{-13}	$(0.04,0.11,0.18)$	0.05	26/28/52	0.127736089451518

Tab. C.2: $G_{3,n}$ data from [23] Part I.

n	(p,q)	γ	ε_V	deviation	B	$\|\|B\|\|_E^2$	S/V/E	covolume
9	$(3,2)$	$-1.50000+0.86603i$	$3\leftrightarrow 13$	2.33×10^{-15}	$(-0.01,0.30,0.11)$	0.10	16/13/28	0.338313868802930
		$-1.50000-0.86603i$	$4\leftrightarrow 14$	3.11×10^{-15}	$(0.13,0.34,0.09)$	0.14	12/8/19	0.338313868803489
	$(2,3)$	$-1.50000+0.86603i$	$3\leftrightarrow 13$	4.88×10^{-15}	$(-0.02,0.25,0.11)$	0.08	16/13/28	0.338313868802763
		$-1.50000-0.86603i$	$3\leftrightarrow 13$	5.11×10^{-15}	$(-0.02,0.27,0.05)$	0.08	16/13/28	0.338313868802010
10	$(3,2)$	$-0.76721-0.79255i$	$3\leftrightarrow 13$ 13	1.64×10^{-14} 1.64×10^{-14}	$(-0.02,0.31,0.12)$	0.11	20/20/38	0.263723173448867 0.263723173448867
		$-0.76721+0.79255i$	$4\leftrightarrow 14$	7.33×10^{-15}	$(-0.05,0.33,0.07)$	0.12	22/26/46	0.263487375455115
		$-2.23279-0.79255i$	$5\leftrightarrow 13$	1.53×10^{-14}	$(0.34,0.22,0.24)$	0.23	22/26/46	0.263518203239260
		$-2.23279+0.79255i$	$4\leftrightarrow 14$ 4	3.20×10^{-14} 2.80×10^{-14}	$(0.16,0.32,0.13)$	0.14	24/34/56	0.263688965427039 0.263688965427042
	$(2,3)$	$-0.76721-0.79255i$	$4\leftrightarrow 13$	1.08×10^{-14}	$(-0.04,0.30,0.01)$	0.09	22/26/46	0.263394331688615
		$-0.76721+0.79255i$	$4\leftrightarrow 13$	1.52×10^{-14}	$(-0.04,0.29,-0.01)$	0.09	22/26/46	0.263394331688655
		$-2.23279-0.79255i$	$5\leftrightarrow 14$	9.10×10^{-15}	$(0.16,0.28,0.10)$	0.11	24/34/56	0.262917569081549
		$-2.23279+0.79255i$	$5\leftrightarrow 14$	6.00×10^{-15}	$(0.16,0.28,-0.11)$	0.11	24/34/56	0.263008715157447
11	$(3,2)$	$0.06115+0.38830i$	$4\leftrightarrow 12$	1.01×10^{-12}	$(0.05,0.38,0.09)$	0.15	24/22/44	0.102843008210627
		$0.06115-0.38830i$	$4\leftrightarrow 13$ 13	7.67×10^{-13} 6.73×10^{-13}	$(0.10,0.29,0.22)$	0.14	24/22/44	0.102696887119289 0.102696887119277
		$-3.06115+0.38830i$	$4\leftrightarrow 13$	2.66×10^{-15}	$(0.26,0.27,0.19)$	0.17	18/18/34	0.205514735563116
		$-3.06115-0.38830i$	$3\leftrightarrow 14$	2.66×10^{-15}	$(-0.23,0.24,0.22)$	0.16	18/18/34	0.205514735563156
	$(2,3)$	$0.06115+0.38830i$	$4\leftrightarrow 13$	2.65×10^{-13}	$(-0.13,0.20,-0.06)$	0.06	24/22/44	0.102661071637195
		$0.06115-0.38830i$	$3\leftrightarrow 13$ 3	3.74×10^{-13} 3.16×10^{-13}	$(-0.11,0.17,0.11)$	0.05	24/22/44	0.102961237259604 0.106095154776647
		$-3.06115+0.38830i$	$4\leftrightarrow 14$ 4	5.11×10^{-15} 5.33×10^{-15}	$(0.01,0.04,0.28)$	0.08	18/18/34	0.205571485365434 0.205571485365421
		$-3.06115-0.38830i$	$4\leftrightarrow 14$	4.22×10^{-15}	$(0.24,0.24,0.18)$	0.14	18/18/34	0.205579755283018
12	$(3,2)$	$0.24698+0.00000i$	$3\leftrightarrow 14$	2.44×10^{-15}	$(0.63,0.48,-0.43)$	0.82	5/0/5	Free?
		$-3.24698+0.00000i$	$3\leftrightarrow 13$	2.32×10^{-14}	$(-1.01,0.15,0.18)$	1.08	11/7/17	Free?
	$(2,3)$	$0.24698+0.00000i$	$3\leftrightarrow 13$	7.44×10^{-15}	$(0.63,0.49,0.45)$	0.84	5/0/5	Free?
		$-3.24698+0.00000i$	$3\leftrightarrow 13$	1.13×10^{-14}	$(0.04,0.13,0.88)$	0.79	11/7/17	Free?
13	$(3,2)$	$-0.52842-0.78122i$	$5\leftrightarrow 14$	2.53×10^{-14}	$(0.04,0.34,0.18)$	0.15	22/24/44	0.363085747245742
		$-0.52842+0.78122i$	$4\leftrightarrow 13$	1.83×10^{-14}	$(0.04,0.37,-0.06)$	0.15	22/24/44	0.362805250204902
		$-2.47158-0.78122i$	$4\leftrightarrow 13$	1.50×10^{-15}	$(0.38,0.17,0.29)$	0.26	8/9/15	0.362887228283480
		$-2.47158+0.78122i$	$4\leftrightarrow 13$	3.00×10^{-15}	$(0.30,0.33,0.17)$	0.23	14/12/24	0.362887228283415
	$(2,3)$	$-0.52842-0.78122i$	$4\leftrightarrow 13$	2.40×10^{-14}	$(-0.05,0.39,0.21)$	0.20	22/24/44	0.363031557094651
		$-0.52842+0.78122i$	$4\leftrightarrow 13$	2.78×10^{-14}	$(-0.10,0.26,0.18)$	0.11	22/24/44	0.363024421723659
		$-2.47158-0.78122i$	$3\leftrightarrow 13$	2.89×10^{-15}	$(0.28,0.26,0.20)$	0.18	14/12/24	0.362887228283405
		$-2.47158+0.78122i$	$3\leftrightarrow 14$	2.55×10^{-15}	$(0.28,0.26,-0.20)$	0.18	14/12/24	0.362887228283461
14	$(3,2)$	$-0.11535-0.58974i$	$6\leftrightarrow 13$	3.17×10^{-13}	$(-0.11,0.30,0.19)$	0.14	24/32/54	0.330857468657992
		$-0.11535+0.58974i$	$4\leftrightarrow 13$	2.13×10^{-14}	$(-0.11,0.35,-0.04)$	0.14	24/32/54	0.330857468657719
		$-2.88465-0.58974i$	$4\leftrightarrow 14$	2.71×10^{-14}	$(0.33,0.03,0.36)$	0.24	22/24/44	0.330802402206134
		$-2.88465+0.58974i$	$4\leftrightarrow 14$	5.02×10^{-14}	$(0.33,0.29,0.22)$	0.24	22/24/44	0.330264970254204
	$(2,3)$	$-0.11535-0.58974i$	$4\leftrightarrow 12$	5.85×10^{-14}	$(0.30,0.24,0.19)$	0.18	24/32/54	0.330857468657576
		$-0.11535+0.58974i$	$4\leftrightarrow 12$	2.89×10^{-14}	$(0.21,0.26,0.26)$	0.18	24/32/54	0.330857468657535
		$-2.88465-0.58974i$	$4\leftrightarrow 13$	2.89×10^{-14}	$(0.29,0.22,0.23)$	0.18	22/24/44	0.330788586067824
		$-2.88465+0.58974i$	$5\leftrightarrow 13$	1.47×10^{-14}	$(-0.04,0.08,0.40)$	0.17	22/24/44	0.330717107428080

Tab. C.3: $G_{3,n}$ data from [23] Part II.

n	(p,q)	γ	ε_V	deviation	B	$\|\|B\|\|_E^2$	S/V/E	covolume
1	$(4,2)$	$-1.00000+0.00000i$	$3\leftrightarrow 14$	2.00×10^{-15}	$(0.75,0.30,0.68)$	1.11	5/1/5	Free?
	$(2,4)$	$-1.00000+0.00000i$	$3\leftrightarrow 14$	8.77×10^{-15}	$(0.76,0.47,0.55)$	1.11	5/1/5	Free?
2	$(4,2)$	$-0.50000+0.86603i$	$4\leftrightarrow 13$	1.29×10^{-14}	$(0.07,0.42,-0.03)$	0.18	20/22/40	0.253735401602386
		$-0.50000-0.86603i$	$4\leftrightarrow 13$	9.44×10^{-15}	$(0.08,0.42,0.05)$	0.18	20/22/40	0.253598511561355
		$-1.50000+0.86603i$	$3\leftrightarrow 13$	4.55×10^{-15}	$(0.18,0.41,0.04)$	0.20	18/22/38	0.253041353515199
		$-1.50000-0.86603i$	$4\leftrightarrow 13$	5.55×10^{-15}	$(0.38,0.31,0.25)$	0.31	18/22/38	0.252720823519394
	$(2,4)$	$-0.50000+0.86603i$	$5\leftrightarrow 13$	7.66×10^{-15}	$(-0.08,0.28,0.07)$	0.09	20/22/40	0.253542824010740
		$-0.50000-0.86603i$	$3\leftrightarrow 13$	2.00×10^{-14}	$(-0.07,0.30,0.05)$	0.10	20/22/40	0.253338508765728
		$-1.50000+0.86603i$	$4\leftrightarrow 14$	6.22×10^{-15}	$(0.08,0.30,-0.00)$	0.10	18/22/38	0.253712610730981
		$-1.50000-0.86603i$	$3\leftrightarrow 14$ 3	3.33×10^{-15} 4.66×10^{-15}	$(0.02,0.29,0.07)$	0.09	18/22/38	0.253384811296644 0.253384811296590
3	$(4,2)$	$-0.12256-0.74486i$	$3\leftrightarrow 13$	2.71×10^{-14}	$(-0.21,0.30,0.14)$	0.16	16/13/27	0.137289264368554
		$-0.12256+0.74486i$	$3\leftrightarrow 13$ $3,4,5$	7.66×10^{-14} 7.84×10^{-14}	$(-0.05,0.25,0.18)$	0.10	22/22/42	0.137353485832864 0.137353485832673
		$-1.87744-0.74486i$	$3\leftrightarrow 14$	4.22×10^{-15}	$(-0.27,0.37,0.23)$	0.27	14/12/24	0.274956314143136
		$-1.87744+0.74486i$	$5\leftrightarrow 13$	3.55×10^{-15}	$(0.28,0.43,0.07)$	0.27	14/12/24	0.274956314143194
	$(2,4)$	$-0.12256-0.74486i$	$5\leftrightarrow 13$	1.07×10^{-13}	$(0.08,0.17,0.15)$	0.06	20/18/36	0.137439859735903
		$-0.12256+0.74486i$	$3\leftrightarrow 13$	2.32×10^{-14}	$(0.22,0.23,-0.06)$	0.11	20/18/36	0.137344777947632
		$-1.87744-0.74486i$	$3\leftrightarrow 13$	1.89×10^{-15}	$(0.13,0.29,0.13)$	0.12	14/12/24	0.274956314143212
		$-1.87744+0.74486i$	$4\leftrightarrow 14$	2.44×10^{-15}	$(0.17,0.30,-0.11)$	0.13	14/12/24	0.274956314143310
4	$(4,2)$	$-1.00000+1.00000i$	$4\leftrightarrow 13$	5.11×10^{-15}	$(0.14,0.44,0.01)$	0.21	16/13/28	0.457982797089828
		$-1.00000-1.00000i$	$4\leftrightarrow 13$	3.11×10^{-15}	$(0.30,0.44,0.19)$	0.31	10/7/16	0.457982797088671
	$(2,4)$	$-1.00000+1.00000i$	$3\leftrightarrow 14$	1.13×10^{-14}	$(-0.04,0.27,0.17)$	0.10	16/13/28	0.457982797089235
		$-1.00000-1.00000i$	$3\leftrightarrow 7$ $8\leftrightarrow 13$	6.11×10^{-15} 6.33×10^{-15}	$(0.08,0.49,0.21)$	0.29	10/7/16	0.457982797088716 0.457982797088756
5	$(4,2)$	$-0.65884-1.16154i$	$3\leftrightarrow 14$	1.13×10^{-14}	$(0.42,0.49,0.24)$	0.47	22/28/48	0.594047553937928
		$-0.65884+1.16154i$	$4\leftrightarrow 13$	6.66×10^{-15}	$(0.36,0.56,-0.09)$	0.46	22/28/48	0.593932061794999
		$-1.34116-1.16154i$	$4\leftrightarrow 13$	3.13×10^{-14}	$(0.25,0.51,-0.03)$	0.33	24/26/48	0.593035916591978
		$-1.34116+1.16154i$	$3\leftrightarrow 12$ 3	4.41×10^{-14} 7.31×10^{-14}	$(0.25,0.51,0.02)$	0.33	24/26/48	0.593035916591825 0.593035916591793
	$(2,4)$	$-0.65884-1.16154i$	$3\leftrightarrow 14$	1.44×10^{-14}	$(0.17,0.38,0.17)$	0.20	22/28/48	0.593687497734380
		$-0.65884+1.16154i$	$4\leftrightarrow 14$	1.12×10^{-14}	$(0.19,0.38,-0.17)$	0.21	22/28/48	0.593688668377331
		$-1.34116-1.16154i$	$4\leftrightarrow 14$	5.42×10^{-14}	$(0.13,0.37,0.08)$	0.16	24/26/48	0.592911657244731
		$-1.34116+1.16154i$	$3\leftrightarrow 14$ 3	6.28×10^{-14} 6.08×10^{-14}	$(0.19,0.37,0.04)$	0.18	24/26/48	0.593687497734524 0.593687497734493
6	$(4,2)$	$0.23279-0.79255i$	$5\leftrightarrow 12$	1.88×10^{-14}	$(0.17,0.50,0.13)$	0.29	22/26/46	0.593261429781721
		$0.23279+0.79255i$	$4\leftrightarrow 11$	5.50×10^{-14}	$(0.13,0.31,0.38)$	0.26	22/28/48	0.593639175240615
		$-2.23279-0.79255i$	$5\leftrightarrow 14$	2.18×10^{-14}	$(-0.27,0.47,0.31)$	0.39	22/28/48	0.594063100273875
		$-2.23279+0.79255i$	$5\leftrightarrow 13$	1.10×10^{-14}	$(-0.26,0.28,0.49)$	0.39	22/28/48	0.595004149184967
	$(2,4)$	$0.23279-0.79255i$	$4\leftrightarrow 13$	4.69×10^{-14}	$(0.03,0.36,0.11)$	0.14	22/26/46	0.593261429782327
		$0.23279+0.79255i$	$3\leftrightarrow 14$	4.24×10^{-14}	$(0.02,0.34,0.12)$	0.13	22/28/48	0.592869862079440
		$-2.23279-0.79255i$	$4\leftrightarrow 14$	1.71×10^{-14}	$(0.33,0.35,0.21)$	0.28	22/28/48	0.593689839020260
		$-2.23279+0.79255i$	$4\leftrightarrow 14$	1.75×10^{-14}	$(0.33,0.34,-0.22)$	0.27	22/28/48	0.593689839020190
7	$(4,2)$	$-0.22816-1.11514i$	$4\leftrightarrow 12$	2.44×10^{-14}	$(0.35,0.44,0.31)$	0.41	22/26/46	0.793775329716258
		$-0.22816+1.11514i$	$4\leftrightarrow 12$	2.04×10^{-14}	$(0.29,0.53,-0.10)$	0.38	24/32/54	0.793730924087971
		$-1.77184-1.11514i$	$4\leftrightarrow 11$	1.08×10^{-13}	$(-0.34,0.45,0.34)$	0.44	22/30/50	0.794323319649402
		$-1.77184+1.11514i$	$4\leftrightarrow 12$	1.26×10^{-13}	$(0.39,0.56,0.03)$	0.46	22/30/50	0.794323319649266
	$(2,4)$	$-0.22816-1.11514i$	$4\leftrightarrow 14$	4.95×10^{-14}	$(0.08,0.37,0.17)$	0.17	24/32/54	0.794272642251702
		$-0.22816+1.11514i$	$3\leftrightarrow 14$	1.86×10^{-14}	$(0.11,0.33,0.23)$	0.18	22/26/46	0.793839731308609
		$-1.77184-1.11514i$	$4\leftrightarrow 13$	5.28×10^{-14}	$(0.30,0.39,0.14)$	0.26	22/30/50	0.794323319650378
		$-1.77184+1.11514i$	$4\leftrightarrow 14$	7.48×10^{-14}	$(0.30,0.38,-0.15)$	0.26	22/30/50	0.794323319650373
8	$(4,2)$	$0.41964-0.60629i$	$4\leftrightarrow 13$ 4 13	4.00×10^{-13} 1.65×10^{-13} 1.63×10^{-13}	$(-0.19,0.32,0.27)$	0.21	22/20/40	0.264774439885714 0.264774439884122 0.264774439884123
		$0.41964+0.60629i$	$4\leftrightarrow 13$	2.42×10^{-13}	$(0.01,0.33,0.18)$	0.14	20/17/35	0.264774439884227
		$-2.41964-0.60629i$	$4\leftrightarrow 13$	1.81×10^{-13}	$(-0.23,0.28,0.31)$	0.23	22/20/40	0.264736844559787
		$-2.41964+0.60629i$	$4\leftrightarrow 13$	6.08×10^{-14}	$(-0.23,0.31,0.29)$	0.23	22/20/40	0.264774439883397
	$(2,4)$	$0.41964-0.60629i$	$3\leftrightarrow 12$ 3	3.94×10^{-13} 3.41×10^{-13}	$(0.39,0.27,0.12)$	0.24	22/20/40	0.264774439886526 0.264774439886159
		$0.41964+0.60629i$	$3\leftrightarrow 12$	2.90×10^{-13}	$(0.15,0.19,0.29)$	0.15	20/17/35	0.264774439884908
		$-2.41964-0.60629i$	$3\leftrightarrow 13$	4.77×10^{-14}	$(0.00,0.13,0.32)$	0.12	22/20/40	0.264774439883719
		$-2.41964+0.60629i$	$4\leftrightarrow 12$	1.78×10^{-13}	$(-0.07,0.29,0.37)$	0.22	22/20/40	0.264774439884830

Tab. C.4: $G_{4,n}$ data from [23] Part I.

n	(p,q)	γ	ε_V	deviation	B	$\|\|B\|\|_E^2$	S/V/E	covolume
9	$(4,2)$	$0.00000+1.00000i$	$4\leftrightarrow 12$	2.72×10^{-14}	$(0.40,0.50,-0.25)$	0.47	20/23/42	0.915965594175611
		$0.00000-1.00000i$	$4\leftrightarrow 13$	2.15×10^{-14}	$(0.40,0.42,0.38)$	0.48	20/23/42	0.915965594176800
		$-2.00000+1.00000i$	$4\leftrightarrow 13$	1.43×10^{-14}	$(0.46,0.57,0.05)$	0.54	14/10/24	0.915965594175439
		$-2.00000-1.00000i$	$4\leftrightarrow 13$	4.66×10^{-15}	$(-0.33,0.49,0.33)$	0.46	14/10/24	0.915965594176536
	$(2,4)$	$0.00000+1.00000i$	$4\leftrightarrow 13$	3.70×10^{-14}	$(-0.04,0.22,0.37)$	0.19	20/23/42	0.915965594176029
		$0.00000-1.00000i$	$4\leftrightarrow 13$ 4	3.22×10^{-13} 2.69×10^{-14}	$(0.14,0.46,0.28)$	0.31	20/23/42	0.915965594178955 0.915965594179460
		$-2.00000+1.00000i$	$4\leftrightarrow 13$	6.33×10^{-15}	$(0.36,0.37,-0.20)$	0.31	14/10/24	0.915965594176399
		$-2.00000-1.00000i$	$3\leftrightarrow 14$	4.66×10^{-15}	$(0.36,0.38,0.19)$	0.31	14/10/24	0.915965594176532
10	$(4,2)$	$0.61803+0.00000i$	$3\leftrightarrow 13$ 13	7.11×10^{-15} 3.72×10^{-14}	$(0.76,0.48,0.47)$	1.03	5/0/5	Free?
		$-2.61803+0.00000i$	$4\leftrightarrow 13$	2.62×10^{-14}	$(-0.80,0.31,0.31)$	0.83	11/7/17	Free?
	$(2,4)$	$0.61803+0.00000i$	$3\leftrightarrow 13$	6.22×10^{-15}	$(0.68,0.46,0.49)$	0.92	5/0/5	Free?
		$-2.61803+0.00000i$	$3\leftrightarrow 13$	1.80×10^{-14}	$(-0.01,0.21,0.96)$	0.96	11/7/17	Free?
11	$(4,2)$	$-1.00000-1.27202i$	$4\leftrightarrow 11$	1.37×10^{-13}	$(0.23,0.36,0.18)$	0.22	10/7/16	Free?
		$-1.00000+1.27202i$	$4\leftrightarrow 13$	3.74×10^{-14}	$(0.41,0.41,0.14)$	0.36	10/7/16	Free?
	$(2,4)$	$-1.00000-1.27202i$	$5\leftrightarrow 13$	3.91×10^{-14}	$(0.12,0.20,0.26)$	0.12	10/7/16	Free?
		$-1.00000+1.27202i$	$4\leftrightarrow 13$	2.93×10^{-14}	$(0.18,0.12,0.25)$	0.11	8/6/13	Free?
12	$(4,2)$	$-0.40631+1.19616i$	$5\leftrightarrow 12$	2.16×10^{-14}	$(0.43,0.57,-0.12)$	0.53	20/22/40	0.897162578942312
		$-0.40631-1.19616i$	$5\leftrightarrow 12$	2.35×10^{-14}	$(0.42,0.58,0.09)$	0.53	20/22/40	0.896781554188068
		$-1.59369+1.19616i$	$4\leftrightarrow 12$ 4	2.75×10^{-14} 2.52×10^{-14}	$(0.42,0.37,0.22)$	0.36	26/28/52	0.447434850607097 0.447434850606938
		$-1.59369-1.19616i$	$6\leftrightarrow 12$	3.90×10^{-14}	$(0.42,0.22,0.37)$	0.36	26/28/52	0.447402486364538
	$(2,4)$	$-0.40631+1.19616i$	$3\leftrightarrow 13$	1.45×10^{-14}	$(0.24,0.40,0.05)$	0.22	20/22/40	0.894660969880485
		$-0.40631-1.19616i$	$3\leftrightarrow 14$ 3	2.44×10^{-14} 2.26×10^{-14}	$(0.23,0.40,0.20)$	0.25	20/22/40	0.896719777668525 0.896719777668514
		$-1.59369+1.19616i$	$4\leftrightarrow 13$ 4	7.35×10^{-13} 5.24×10^{-13}	$(0.11,0.17,-0.27)$	0.11	26/28/52	0.447434850610119 0.447434850615054
		$-1.59369-1.19616i$	$4\leftrightarrow 12$	1.72×10^{-13}	$(0.11,0.18,0.26)$	0.11	26/28/52	0.447434850609249
13	$(4,2)$	$0.78810+0.40136i$	$3\leftrightarrow 12$ 3	4.49×10^{-13} 7.54×10^{-13}	$(-0.15,0.35,0.31)$	0.24	24/22/44	0.347567297638236 0.347567297638396
		$0.78810-0.40136i$	$4\leftrightarrow 12$	9.87×10^{-13}	$(-0.12,0.45,0.17)$	0.24	24/22/44	0.347568126904567
		$-2.78810+0.40136i$	$4\leftrightarrow 13$	1.12×10^{-14}	$(-0.05,0.36,0.39)$	0.28	20/22/40	0.695106728358357
		$-2.78810-0.40136i$	$3\leftrightarrow 12$	2.04×10^{-14}	$(-0.16,0.49,0.26)$	0.33	18/18/34	0.694418848157637
	$(2,4)$	$0.78810+0.40136i$	$3\leftrightarrow 11$ 3	9.50×10^{-13} 9.47×10^{-13}	$(0.02,0.50,0.25)$	0.31	24/22/44	0.347568126901567 0.347568126901511
		$0.78810-0.40136i$	$3\leftrightarrow 11$ 3 4	7.84×10^{-13} 7.85×10^{-13} failure.	$(0.11,0.58,0.20)$	0.39	24/22/44	0.347567297637630 0.347567297637376
		$-2.78810+0.40136i$	$3\leftrightarrow 14$	3.69×10^{-14}	$(0.12,0.27,0.27)$	0.16	20/22/40	0.695481297423900
		$-2.78810-0.40136i$	$4\leftrightarrow 13$	1.73×10^{-14}	$(0.25,0.31,0.21)$	0.20	20/22/40	0.695279250172051

Tab. C.5: $G_{4,n}$ data from [23] Part II.

n	(p,q)	γ	ε_V	deviation	B	$\|\|B\|\|_E^2$	S/V/E	covolume
1	$(5,2)$	$-0.38197+0.00000i$	$3\leftrightarrow 13$	6.88×10^{-15}	$(0.79,0.63,0.39)$	1.17	5/1/5	Free?
		$-1.00000+0.00000i$	$3\leftrightarrow 12$	4.66×10^{-15}	$(0.81,0.32,0.28)$	0.83	5/1/5	Free?
	$(2,5)$	$-0.38197+0.00000i$	$3\leftrightarrow 13$	5.55×10^{-15}	$(0.77,0.46,0.59)$	1.16	5/1/5	Free?
		$-1.00000+0.00000i$	$3\leftrightarrow 14$	3.66×10^{-15}	$(0.93,0.34,0.31)$	1.08	5/1/5	Free?
2	$(5,2)$	$-0.69098+0.72287i$	$3\leftrightarrow 12$	3.89×10^{-15}	$(0.10,0.34,0.08)$	0.13	10/8/16	0.093325539506778
			3	5.88×10^{-15}				0.093325539506677
		$-0.69098-0.72287i$	$4\leftrightarrow 12$	4.77×10^{-15}	$(0.04,0.31,0.13)$	0.11	10/8/16	0.093325539506687
	$(2,5)$	$-0.69098+0.72287i$	$5\leftrightarrow 13$	1.49×10^{-14}	$(0.04,0.16,-0.12)$	0.04	10/8/16	0.093325539506920
		$-0.69098-0.72287i$	$3\leftrightarrow 13$	1.29×10^{-14}	$(0.04,0.17,0.12)$	0.04	10/8/16	0.093325539506970
3	$(5,2)$	$0.11803-0.60666i$	$4\leftrightarrow 13$	4.22×10^{-14}	$(-0.27,0.17,0.18)$	0.13	18/16/32	0.039105939852173
		$0.11803+0.60666i$	$4\leftrightarrow 12$	4.64×10^{-14}	$(-0.27,0.22,0.11)$	0.13	18/16/32	0.039120554210565
			4	4.55×10^{-14}				0.039120554210676
		$-1.50000-0.60666i$	$3\leftrightarrow 12$	6.77×10^{-15}	$(-0.13,0.39,0.25)$	0.23	16/16/30	0.234301713689990
		$-1.50000+0.60666i$	$3\leftrightarrow 12$	failure. possibly generates a finite-sheeted cover				
	$(2,5)$	$0.11803-0.60666i$	$3\leftrightarrow 13$	2.49×10^{-14}	$(-0.16,0.02,0.15)$	0.05	18/16/32	0.039051905406329
		$0.11803+0.60666i$	$3\leftrightarrow 13$	2.49×10^{-14}	$(-0.17,0.03,0.14)$	0.05	10/8/16	0.039050285615583
			3, 4	1.27×10^{-14}				0.039050285615487
			5	failure.				
		$-1.50000-0.60666i$	$3\leftrightarrow 14$	failure. possibly generates a finite-sheeted cover				
		$-1.50000+0.60666i$	$3\leftrightarrow 14$	failure. possibly generates a finite-sheeted cover				
4	$(5,2)$	$-0.19098+0.98159i$	$4\leftrightarrow 13$	1.17×10^{-14}	$(0.29,0.53,-0.17)$	0.39	20/22/40	0.468603427380087
		$-0.19098-0.98159i$	$4\leftrightarrow 12$	1.57×10^{-14}	$(0.28,0.46,0.30)$	0.38	20/22/40	0.468603427380127
		$-1.19098+0.98159i$	$4\leftrightarrow 13$	1.91×10^{-14}	$(-0.23,0.46,0.33)$	0.38	18/22/38	0.468603427379948
		$-1.19098-0.98159i$	$3\leftrightarrow 13$	1.67×10^{-14}	$(-0.23,0.45,0.35)$	0.38	18/22/38	0.468603427380023
			13	8.19×10^{-14}				0.468603427379797
	$(2,5)$	$-0.19098+0.98159i$	$3\leftrightarrow 13$	6.48×10^{-14}	$(0.05,0.34,0.06)$	0.12	20/22/40	0.468603427380566
			3	2.11×10^{-14}				0.468603427380367
		$-0.19098-0.98159i$	$3\leftrightarrow 13$	2.82×10^{-14}	$(0.06,0.35,0.05)$	0.13	20/22/40	0.468419988135056
		$-1.19098+0.98159i$	$3\leftrightarrow 13$	8.33×10^{-15}	$(0.18,0.35,0.01)$	0.15	18/22/38	0.468603427380267
		$-1.19098-0.98159i$	$3\leftrightarrow 13$	6.77×10^{-15}	$(0.05,0.34,0.10)$	0.13	18/22/38	0.468603427380289
5	$(5,2)$	$0.61803+0.00000i$	$3\leftrightarrow 12$	7.33×10^{-15}	$(0.75,0.58,0.35)$	1.03	5/0/5	Free?
		$-2.00000+0.00000i$	$3\leftrightarrow 13$	8.22×10^{-15}	$(-0.61,0.73,0.13)$	0.93	7/6/12	Free?
			3, 4	9.10×10^{-15}				
			10, 11, 12	1.42×10^{-14}				
			13	2.72×10^{-14}				
			5, 9	failure.				
	$(2,5)$	$0.61803+0.00000i$	$3\leftrightarrow 13$	1.17×10^{-14}	$(0.75,0.45,0.54)$	1.07	5/0/5	Free?
		$-2.00000+0.00000i$	$3\leftrightarrow 14$	1.10×10^{-14}	$(-0.25,-0.05,0.83)$	0.76	11/9/19	Free?
			3, 4	8.66×10^{-15}			7/6/12	
			5	failure.				
			$6\leftrightarrow 9$	6.44×10^{-15}	$(-0.25,-0.05,0.83)$	0.76	7/6/12	Free?
6	$(5,2)$	$0.25278-0.85077i$	$3\leftrightarrow 11$	8.85×10^{-14}	$(0.06,0.50,0.13)$	0.27	20/18/36	0.313340838688514
			3	6.96×10^{-14}				0.313340838689695
		$0.25278+0.85077i$	$4\leftrightarrow 13$	1.72×10^{-14}	$(0.08,0.43,0.29)$	0.28	20/18/36	0.313085454214014
		$-1.63475-0.85077i$	$4\leftrightarrow 14$	3.77×10^{-15}	$(-0.15,0.52,0.34)$	0.41	14/12/24	0.626170908428058
		$-1.63475+0.85077i$	$3\leftrightarrow 13$	8.77×10^{-15}	$(-0.15,0.47,0.41)$	0.41	14/12/24	0.626170908427892
	$(2,5)$	$0.25278-0.85077i$	$3\leftrightarrow 13$	1.31×10^{-13}	$(-0.25,0.21,0.21)$	0.15	20/18/36	0.312700441258862
		$0.25278+0.85077i$	$3\leftrightarrow 12$	1.63×10^{-13}	$(-0.28,-0.04,0.29)$	0.16	22/22/42	0.313085454215978
		$-1.63475-0.85077i$	$3\leftrightarrow 13$	4.00×10^{-15}	$(0.18,0.34,0.17)$	0.18	14/12/24	0.626170908427952
		$-1.63475+0.85077i$	$3\leftrightarrow 13$	6.00×10^{-15}	$(0.18,0.34,-0.18)$	0.17	14/12/24	0.626170908427911
7	$(5,2)$	$-0.69098+1.23391i$	$5\leftrightarrow 12$	4.44×10^{-14}	$(-0.41,0.57,0.25)$	0.55	16/13/28	Free?
		$-0.69098-1.23391i$	$5\leftrightarrow 12$	1.22×10^{-14}	$(0.43,0.61,0.13)$	0.57	16/13/28	Free?
		$-0.69098+1.23391i$	$5\leftrightarrow 12$	2.15×10^{-14}	$(-0.41,0.57,0.25)$	0.55	16/13/28	Free?
		$-0.69098-1.23391i$	$5\leftrightarrow 11$	2.39×10^{-14}	$(0.43,0.61,0.13)$	0.57	16/13/28	Free?
	$(2,5)$	$-0.69098+1.23391i$	$3\leftrightarrow 13$	5.00×10^{-15}	$(-0.12,0.18,0.42)$	0.22	12/11/22	Free?
		$-0.69098-1.23391i$	$3\leftrightarrow 13$	3.64×10^{-14}	$(0.10,0.39,0.12)$	0.18	16/13/28	Free?
		$-0.69098+1.23391i$	$3\leftrightarrow 13$	4.71×10^{-15}	$(-0.12,0.18,0.42)$	0.22	12/11/22	Free?
		$-0.69098-1.23391i$	$3\leftrightarrow 13$	3.60×10^{-14}	$(0.10,0.39,0.12)$	0.18	16/13/28	Free?
8	$(5,2)$	$-0.38197+1.27202i$	$4\leftrightarrow 13$	1.48×10^{-13}	$(0.52,0.60,-0.22)$	0.68	22/28/48	0.860875274165978
		$-0.38197-1.27202i$	$4\leftrightarrow 13$	2.26×10^{-14}	$(0.57,0.52,0.37)$	0.73	22/28/48	0.861225344288923
			4, 5	1.67×10^{-14}			20/25/43	0.861236641862642
		$-1.00000+1.27202i$	$3\leftrightarrow 13$	3.11×10^{-14}	$(-0.27,0.59,0.32)$	0.52	24/26/48	0.859899854837779
		$-1.00000-1.27202i$	$4\leftrightarrow 14$	7.99×10^{-14}	$(-0.27,0.56,-0.34)$	0.50	24/26/48	0.859899854837823
	$(2,5)$	$-0.38197+1.27202i$	$5\leftrightarrow 13$	1.29×10^{-14}	$(-0.05,0.23,0.37)$	0.19	20/25/43	0.861241520146001
		$-0.38197-1.27202i$	$5\leftrightarrow 13$	1.71×10^{-14}	$(0.16,0.40,0.17)$	0.21	22/28/48	0.860875274165283
		$-1.00000+1.27202i$	$4\leftrightarrow 14$	5.71×10^{-14}	$(0.25,0.41,0.08)$	0.24	24/26/48	0.859899854837891
			14	2.80×10^{-14}				0.859899854837936
		$-1.00000-1.27202i$	$3\leftrightarrow 14$	1.33×10^{-13}	$(0.16,0.41,0.10)$	0.20	24/26/48	0.860960287124459

Tab. C.6: $G_{5,n}$ data from [23] Part I.

n	(p,q)	γ	ε_V	deviation	B	$\|\|B\|\|_E^2$	S/V/E	covolume
9	$(5,2)$	$0.11803-1.16963i$	$4\leftrightarrow 12$ 12	5.74×10^{-14} 2.97×10^{-13}	$(0.59,0.45,0.46)$	0.77	22/30/50	1.118581444567214 1.118581444565920
		$0.11803+1.16963i$	$4\leftrightarrow 11$	9.98×10^{-14}	$(0.60,0.57,0.33)$	0.79	22/30/50	1.119298555873805
		$-1.50000-1.16963i$	$4\leftrightarrow 11$	2.96×10^{-13}	$(-0.21,0.59,0.44)$	0.59	22/30/50	1.120064501687885
		$-1.50000+1.16963i$	$5\leftrightarrow 11$	1.57×10^{-13}	$(-0.21,0.57,0.46)$	0.58	22/30/50	1.119989940730108
	$(2,5)$	$0.11803-1.16963i$	$3\leftrightarrow 13$	3.67×10^{-14}	$(0.11,0.39,0.20)$	0.21	24/32/54	1.119270315426469
		$0.11803+1.16963i$	$4\leftrightarrow 13$	8.07×10^{-14}	$(0.19,0.36,0.27)$	0.24	22/30/50	1.118814103894236
		$-1.50000-1.16963i$	$4\leftrightarrow 14$ 14	4.82×10^{-14} 5.62×10^{-14}	$(0.33,0.41,0.18)$	0.31	22/30/50	1.120222529291118 1.120222529291008
		$-1.50000+1.16963i$	$5\leftrightarrow 14$	2.10×10^{-13}	$(0.33,0.41,-0.19)$	0.31	22/30/50	1.120222529291431
10	$(5,2)$	$-0.08180-1.28803i$	$4\leftrightarrow 11$	6.79×10^{-14}	$(0.60,0.50,0.41)$	0.78	20/22/40	1.401249917568806
		$-0.08180+1.28803i$	$5\leftrightarrow 11$	1.70×10^{-13}	$(0.59,0.60,-0.27)$	0.78	20/22/40	1.401249917567046
		$-1.30017-1.28803i$	$5\leftrightarrow 12$	2.68×10^{-12}	$(-0.50,0.35,0.44)$	0.57	26/28/52	0.700666393700143
		$-1.30017+1.28803i$	$4\leftrightarrow 10$ 4	1.87×10^{-12} 1.43×10^{-12}	$(-0.41,0.39,0.44)$	0.51	26/28/52	0.700364591055182 0.700364591063245
	$(2,5)$	$-0.08180-1.28803i$	$3\leftrightarrow 13$	2.05×10^{-14}	$(0.30,0.43,0.24)$	0.33	20/22/40	1.402192567477373
		$-0.08180+1.28803i$	$3\leftrightarrow 14$	4.17×10^{-14}	$(0.36,0.43,0.04)$	0.32	20/22/40	1.401249917569314
		$-1.30017-1.28803i$	$4\leftrightarrow 12$	1.75×10^{-12}	$(0.11,0.22,0.30)$	0.15	26/28/52	0.699824630962773
		$-1.30017+1.28803i$	$5\leftrightarrow 12$	2.18×10^{-12}	$(0.11,0.22,-0.30)$	0.15	26/28/52	0.699824630999953
11	$(5,2)$	$0.61803+0.78615i$	$4\leftrightarrow 11$ 4 5	2.23×10^{-13} 2.23×10^{-13} failure.	$(0.38,0.57,0.24)$	0.53	22/28/48	0.860708833608866 0.860708833608874
		$0.61803-0.78615i$	$4\leftrightarrow 11$	1.25×10^{-13}	$(0.43,0.55,0.31)$	0.59	22/28/48	0.861241520145068
		$-2.00000+0.78615i$	$3\leftrightarrow 13$ 13	5.96×10^{-14} 1.38×10^{-13}	$(-0.12,0.46,0.54)$	0.51	20/25/43	0.861241520145816 0.861241520145561
		$-2.00000-0.78615i$	$4\leftrightarrow 13$	6.33×10^{-14}	$(-0.13,0.64,0.31)$	0.52	20/25/43	0.861241520146301
	$(2,5)$	$0.61803+0.78615i$	$3\leftrightarrow 12$	3.16×10^{-14}	$(0.15,0.38,0.12)$	0.18	22/28/48	0.860320286827674
		$0.61803-0.78615i$	$4\leftrightarrow 12$ 13	1.19×10^{-13} 7.02×10^{-14}	$(0.16,0.39,0.13)$	0.20	22/28/48	0.861241520145928 0.861241520145697
		$-2.00000+0.78615i$	$4\leftrightarrow 13$	2.00×10^{-14}	$(-0.27,0.15,0.40)$	0.26	20/25/43	0.861241520146132
		$-2.00000-0.78615i$	$3\leftrightarrow 13$	3.26×10^{-14}	$(0.34,0.45,0.28)$	0.39	20/25/43	0.861241520146243
12	$(5,2)$	$0.87764+0.58260i$	$4\leftrightarrow 12$ 4	3.64×10^{-13} 2.33×10^{-13}	$(0.15,0.56,0.18)$	0.37	22/20/40	0.454752582526266 0.454752582525471
		$0.87764-0.58260i$	$4\leftrightarrow 13$	3.35×10^{-13}	$(0.19,0.52,0.29)$	0.39	22/20/40	0.454519626171802
		$-2.25960+0.58260i$	$5\leftrightarrow 13$	6.62×10^{-14}	$(-0.01,0.58,0.11)$	0.35	22/20/40	0.454752582524517
		$-2.25960-0.58260i$	$4\leftrightarrow 12$	4.30×10^{-13}	$(-0.05,0.49,0.34)$	0.36	22/20/40	0.454519626172170
	$(2,5)$	$0.87764+0.58260i$	$4\leftrightarrow 13$	7.77×10^{-14}	$(-0.23,0.04,0.33)$	0.16	22/20/40	0.454752582524330
		$0.87764-0.58260i$	$3\leftrightarrow 13$	4.05×10^{-13}	$(-0.31,0.28,0.22)$	0.22	22/20/40	0.454272752254004
		$-2.25960+0.58260i$	$4\leftrightarrow 12$	2.83×10^{-14}	$(-0.09,0.34,0.36)$	0.25	22/20/40	0.454271480250933
		$-2.25960-0.58260i$	$3\leftrightarrow 13$	7.66×10^{-14}	$(-0.03,0.22,0.30)$	0.14	22/20/40	0.454729363580044

Tab. C.7: $G_{5,n}$ data from [23] Part II.

n	(p,q)	γ	ε_V	deviation	B	$\|\|B\|\|_E^2$	S/V/E	covolume
1	$(6,2)$	$-0.50000+0.86603i$	$3\leftrightarrow 13$	7.55×10^{-15}	$(0.32,0.24,0.44)$	0.36	10/7/16	0.253735401603465
			3	7.77×10^{-15}				0.253735401603176
			4	8.44×10^{-15}				0.253735401602411
		$-0.50000-0.86603i$	$3\leftrightarrow 13$	8.66×10^{-15}	$(0.32,0.24,0.44)$	0.36	10/7/16	0.253735401603135
	$(2,6)$	$-0.50000+0.86603i$	$3\leftrightarrow 13$	6.22×10^{-15}	$(0.17,0.07,0.19)$	0.07	10/8/17	0.253735401602576
			3, 4					0.253735401602561
			5	failure.				
		$-0.50000-0.86603i$	$4\leftrightarrow 13$	1.82×10^{-14}	$(0.23,0.20,0.17)$	0.12	10/7/16	0.253735401603777
2	$(6,2)$	$0.00000+1.00000i$	$4\leftrightarrow 13$	1.15×10^{-14}	$(0.46,0.52,0.41)$	0.64	20/22/40	0.610643729451075
		$0.00000-1.00000i$	$4\leftrightarrow 13$	1.14×10^{-14}	$(0.46,0.49,0.41)$	0.61	20/22/40	0.610124675044900
		$-1.00000+1.00000i$	$6\leftrightarrow 13$	2.98×10^{-14}	$(-0.09,0.65,0.28)$	0.51	18/22/38	0.610643729451021
			13	5.22×10^{-14}				0.610643729451050
		$-1.00000-1.00000i$	$4\leftrightarrow 12$	2.82×10^{-14}	$(-0.09,0.55,0.44)$	0.51	18/22/38	0.610643729451040
	$(2,6)$	$0.00000+1.00000i$	$3\leftrightarrow 13$	2.35×10^{-14}	$(0.12,0.36,0.06)$	0.15	20/22/40	0.609758967962758
			3	1.31×10^{-14}				0.609758967962676
		$0.00000-1.00000i$	$3\leftrightarrow 13$	1.17×10^{-14}	$(-0.12,0.34,0.15)$	0.15	20/22/40	0.610691628099879
		$-1.00000+1.00000i$	$3\leftrightarrow 14$	1.55×10^{-14}	$(0.04,0.36,-0.12)$	0.14	18/22/38	0.610643729451591
		$-1.00000-1.00000i$	$4\leftrightarrow 14$	6.66×10^{-15}	$(0.04,0.36,0.11)$	0.14	18/22/38	0.610643729451522
3	$(6,2)$	$0.50000+0.86603i$	$4\leftrightarrow 10$	6.49×10^{-13}	$(0.32,0.55,0.36)$	0.53	20/17/36	0.507512535466457
		$0.50000-0.86603i$	$4\leftrightarrow 10$	1.89×10^{-13}	$(0.31,0.43,0.49)$	0.52	20/17/36	0.507470803201674
			4	1.43×10^{-13}				0.507470803203489
		$-1.50000+0.86603i$	$4\leftrightarrow 12$	2.00×10^{-14}	$(0.02,0.67,0.38)$	0.60	16/14/30	1.015497614363542
		$-1.50000-0.86603i$	$4\leftrightarrow 12$	1.35×10^{-14}	$(-0.04,0.71,0.25)$	0.57	14/10/24	1.014941606407466
	$(2,6)$	$0.50000+0.86603i$	$3\leftrightarrow 12$	1.51×10^{-14}	$(-0.23,-0.03,0.33)$	0.16	22/21/42	0.507438590492248
		$0.50000-0.86603i$	$3\leftrightarrow 13$	1.85×10^{-13}	$(-0.20,0.24,0.25)$	0.16	20/17/36	0.508115372507426
			3	1.87×10^{-13}				0.508115372507441
		$-1.50000+0.86603i$	$3\leftrightarrow 14$	6.22×10^{-15}	$(0.17,0.36,-0.20)$	0.20	16/14/30	1.016300073516832
		$-1.50000-0.86603i$	$3\leftrightarrow 13$	5.00×10^{-15}	$(0.18,0.36,0.20)$	0.20	16/14/30	1.016300073515976
4	$(6,2)$	$-0.50000+1.32288i$	$5\leftrightarrow 12$	3.33×10^{-14}	$(-0.28,0.79,0.21)$	0.75	16/13/28	Free?
		$-0.50000-1.32288i$	$4\leftrightarrow 12$	2.02×10^{-14}	$(-0.29,0.76,-0.27)$	0.73	16/13/28	Free?
	$(2,6)$	$-0.50000+1.32288i$	$3\leftrightarrow 13$	3.00×10^{-15}	$(-0.08,0.21,0.47)$	0.27	15/16/30	Free?
			3	3.55×10^{-15}			12/11/22	Free?
			4, 5, 6	3.00×10^{-15}				Free?
			12, 13	9.33×10^{-15}			17/19/35	Free?
			6	failure.				
		$-0.50000-1.32288i$	$4\leftrightarrow 13$	9.88×10^{-15}	$(0.38,0.66,0.34)$	0.70	15/16/30	Free?
			4, 5	4.88×10^{-15}			12/11/22	Free?
			12, 13	1.34×10^{-14}			17/19/35	Free?
			6	failure.				
5	$(6,2)$	$-0.21508-1.30714i$	$4\leftrightarrow 13$	1.18×10^{-13}	$(-0.32,0.79,-0.05)$	0.73	26/38/62	1.021768522986962
		$-0.21508+1.30714i$	$4\leftrightarrow 11$	3.72×10^{-13}	$(-0.36,0.76,0.19)$	0.75	26/36/60	1.021347614330418
		$-0.78492-1.30714i$	$4\leftrightarrow 13$	7.08×10^{-14}	$(-0.27,0.70,0.41)$	0.73	22/26/46	1.019497428415570
		$-0.78492+1.30714i$	$5\leftrightarrow 11$	1.38×10^{-13}	$(-0.10,0.80,0.22)$	0.70	24/26/48	1.021266309673058
	$(2,6)$	$-0.21508-1.30714i$	$4\leftrightarrow 14$	2.00×10^{-13}	$(0.13,0.41,0.17)$	0.22	26/38/62	1.022378515079977
		$-0.21508+1.30714i$	$5\leftrightarrow 13$	4.31×10^{-14}	$(0.00,0.29,0.33)$	0.20	22/28/48	1.021444941021198
		$-0.78492-1.30714i$	$3\leftrightarrow 12$	6.68×10^{-14}	$(0.17,0.43,0.11)$	0.23	24/26/48	1.021266309674068
		$-0.78492+1.30714i$	$4\leftrightarrow 13$	1.58×10^{-14}	$(0.15,0.28,0.34)$	0.22	22/26/46	1.019497428415326
			13	1.51×10^{-13}				1.019497428414627
6	$(6,2)$	$0.34116-1.16154i$	$5\leftrightarrow 12$	1.51×10^{-13}	$(-0.48,0.74,0.22)$	0.82	26/36/60	1.320409963222722
		$0.34116+1.16154i$	$4\leftrightarrow 11$	2.40×10^{-13}	$(-0.49,0.53,0.55)$	0.82	26/36/60	1.321558562482194
		$-1.34116-1.16154i$	$4\leftrightarrow 13$	1.77×10^{-13}	$(-0.03,0.72,0.52)$	0.79	22/30/50	1.319305550520877
		$-1.34116+1.16154i$	$4\leftrightarrow 13$	1.56×10^{-13}	$(-0.11,0.80,0.33)$	0.75	22/30/50	1.318904026482595
	$(2,6)$	$0.34116-1.16154i$	$4\leftrightarrow 13$	8.26×10^{-14}	$(0.11,0.40,0.23)$	0.22	26/36/60	1.322625363588137
		$0.34116+1.16154i$	$4\leftrightarrow 13$	4.51×10^{-14}	$(0.22,0.36,0.30)$	0.27	22/30/50	1.320702203534967
		$-1.34116-1.16154i$	$6\leftrightarrow 13$	7.86×10^{-14}	$(0.33,0.42,0.20)$	0.33	22/30/50	1.319305550520717
		$-1.34116+1.16154i$	$5\leftrightarrow 13$	4.60×10^{-14}	$(0.33,0.41,-0.22)$	0.33	22/30/50	1.319305550520641
7	$(6,2)$	$1.00000+0.00000i$	$3\leftrightarrow 15$	failure. possibly generates a finite-sheeted cover				
		$-2.00000+0.00000i$	$3\leftrightarrow 12$	1.62×10^{-14}	$(0.01,0.85,0.23)$	0.78	11/9/19	Free?
	$(2,6)$	$1.00000+0.00000i$	$3\leftrightarrow 14$	4.88×10^{-15}	$(0.70,0.44,0.54)$	0.96	5/0/5	Free?
		$-2.00000+0.00000i$	$3\leftrightarrow 14$	1.40×10^{-14}	$(-0.24,-0.09,0.88)$	0.85	11/9/19	Free?
			3, 4, 5, 6	5.77×10^{-15}			7/6/12	Free?
			7	failure.				
8	$(6,2)$	$0.87744-0.74486i$	$5\leftrightarrow 14$	6.93×10^{-14}	$(-0.43,0.67,0.31)$	0.72	22/30/50	1.021266309675690
		$0.87744+0.74486i$	$4\leftrightarrow 11$	5.82×10^{-14}	$(-0.47,0.71,0.23)$	0.78	20/26/44	1.022578858868845
		$-1.87744-0.74486i$	$6\leftrightarrow 13$	2.73×10^{-13}	$(0.06,0.79,0.24)$	0.69	26/36/60	1.021317792812498
		$-1.87744+0.74486i$	$4\leftrightarrow 13$	1.73×10^{-13}	$(0.20,0.68,0.35)$	0.63	26/38/62	1.020123408160045
	$(2,6)$	$0.87744-0.74486i$	$3\leftrightarrow 13$	2.82×10^{-14}	$(0.23,0.41,0.14)$	0.24	20/26/44	1.022004856907895
			13	4.35×10^{-14}				1.022004856907774
		$0.87744+0.74486i$	$4\leftrightarrow 12$	4.35×10^{-14}	$(0.23,0.40,0.12)$	0.23	22/30/50	1.021266309674652
		$-1.87744-0.74486i$	$3\leftrightarrow 13$	4.07×10^{-14}	$(0.17,0.37,0.19)$	0.20	26/38/62	1.020786170671771
		$-1.87744+0.74486i$	$4\leftrightarrow 13$	2.58×10^{-13}	$(-0.12,0.31,0.28)$	0.18	26/38/62	1.021187860075532

Tab. C.8: $G_{6,n}$ data from [23].

n	(p,q)	γ	ε_V	deviation	B	$\|\|B\|\|_E^2$	S/V/E	covolume
1	$(7,2)$	$0.24698+0.00000i$	$3\leftrightarrow 13$	6.11×10^{-15}	$(0.70,0.60,0.35)$	0.98	5/0/5	Free?
		$-1.00000+0.00000i$	$4\leftrightarrow 13$ 13	9.66×10^{-15} 2.09×10^{-14}	$(-0.88,0.39,0.29)$	1.00	11/7/17	Free?
	$(2,7)$	$0.24698+0.00000i$	$3\leftrightarrow 14$ 14	5.11×10^{-15} 1.19×10^{-14}	$(0.75,0.45,0.58)$	1.11	5/0/5	Free? Free?
		$-1.00000+0.00000i$	$3\leftrightarrow 14$	5.77×10^{-15}	$(-0.23,0.07,0.97)$	0.99	11/7/17	Free?
2	$(7,2)$	$-0.37651+0.92641i$	$4\leftrightarrow 13$ $4,5$	2.26×10^{-14} 2.53×10^{-14}	$(-0.18,0.46,0.45)$	0.45	10/7/16	Free?
		$-0.37651-0.92641i$	$4\leftrightarrow 12$	1.11×10^{-14}	$(-0.12,0.53,0.34)$	0.40	18/15/32	Free?
	$(-2,7)$	$-0.37651+0.92641i$	$5\leftrightarrow 12$	2.22×10^{-14}	$(-0.30,-0.14,0.26)$	0.18	18/15/32	Free?
		$-0.37651-0.92641i$	$5\leftrightarrow 12$	2.55×10^{-14}	$(-0.30,0.13,0.26)$	0.18	18/15/32	Free?
3	$(7,2)$	$1.24698+0.00000i$	$3\leftrightarrow 15$	failure. possibly generates a finite-sheeted cover				
		$-2.00000+0.00000i$	$4\leftrightarrow 12$	8.06×10^{-14}	$(0.02,0.87,0.31)$	0.85	11/9/19	Free?
	$(2,7)$	$1.24698+0.00000i$	$3\leftrightarrow 13$	7.66×10^{-15}	$(0.71,0.44,0.55)$	1.00	5/0/5	Free?
		$-2.00000+0.00000i$	$3\leftrightarrow 13$	6.22×10^{-15}	$(-0.22,-0.07,0.81)$	0.70	11/9/19	Free?

Tab. C.9: $G_{7,n}$ data from [23].

n	(p,q)	γ	ε_V	deviation	B	$\|\|B\|\|_E^2$	S/V/E	covolume
1	(3, 2)	$-2.66236+0.56228i$	$4\leftrightarrow 14$	1.67×10^{-14}	$(0.26, 0.25, 0.18)$	0.16	20/22/40	0.156983968194752
		$-2.66236-0.56228i$	$4\leftrightarrow 14$	1.09×10^{-14}	$(0.26, 0.02, 0.30)$	0.16	20/22/40	0.156850042593146
		$-0.33764+0.56228i$	$4\leftrightarrow 12$	1.16×10^{-13}	$(0.18, 0.26, -0.05)$	0.11	24/28/50	0.157058545714459
		$-0.33764-0.56228i$	$3\leftrightarrow 12$ 3	1.11×10^{-13} 1.40×10^{-13}	$(0.16, 0.20, 0.20)$	0.11	22/26/46	0.157285227171474 0.157285227171736
			4	failure.				
	(2, 3)	$-2.66236+0.56228i$	$4\leftrightarrow 14$	8.88×10^{-15}	$(0.20, 0.18, -0.19)$	0.11	20/22/40	0.156983968194785
		$-2.66236-0.56228i$	$4\leftrightarrow 13$	3.33×10^{-15}	$(0.20, 0.19, 0.19)$	0.11	20/22/40	0.156983968194460
		$-0.33764+0.56228i$	$4\leftrightarrow 12$	3.60×10^{-14}	$(0.10, 0.16, 0.27)$	0.11	24/28/50	0.156960920774048
		$-0.33764-0.56228i$	$4\leftrightarrow 13$	4.85×10^{-13}	$(0.18, 0.24, 0.09)$	0.10	22/26/46	0.157201560484887
2	(3, 2)	$-1.87744+0.74486i$	$3\leftrightarrow 13$	9.33×10^{-15}	$(0.01, 0.27, 0.11)$	0.08	24/26/48	0.157117893796149
		$-1.87744-0.74486i$	$4\leftrightarrow 14$	9.33×10^{-15}	$(0.01, 0.26, -0.11)$	0.08	24/26/48	0.157117893796187
		$-1.12256+0.74486i$	$3\leftrightarrow 14$	8.66×10^{-15}	$(-0.03, 0.30, 0.08)$	0.10	20/22/40	0.157403816793093
		$-1.12256-0.74486i$	$3\leftrightarrow 14$	9.33×10^{-15}	$(0.01, 0.30, 0.06)$	0.10	20/22/40	0.157135965589722
	(2, 3)	$-1.87744+0.74486i$	$3\leftrightarrow 13$	3.11×10^{-15}	$(0.01, 0.25, -0.00)$	0.06	24/26/48	0.157117893796125
		$-1.87744-0.74486i$	$3\leftrightarrow 13$	7.99×10^{-15}	$(0.00, 0.25, 0.04)$	0.06	24/26/48	0.157117893796084
		$-1.12256+0.74486i$	$3\leftrightarrow 13$	5.88×10^{-15}	$(-0.08, 0.26, -0.08)$	0.08	20/22/40	0.157117893796242
		$-1.12256-0.74486i$	$4\leftrightarrow 13$	8.10×10^{-15}	$(-0.08, 0.26, 0.08)$	0.08	20/22/40	0.157117893796344
3	(3, 2)	$0.00755+0.51312i$	$5\leftrightarrow 12$	6.66×10^{-14}	$(-0.23, 0.00, 0.32)$	0.16	24/32/54	0.326923948367612
		$0.00755-0.51312i$	$4\leftrightarrow 11$	7.06×10^{-14}	$(-0.23, 0.28, 0.17)$	0.16	24/32/54	0.326895403085588
		$-3.00755+0.51312i$	$5\leftrightarrow 13$	6.66×10^{-15}	$(0.31, 0.29, 0.21)$	0.23	24/32/54	0.327149654332462
		$-3.00755-0.51312i$	$4\leftrightarrow 13$	2.11×10^{-14}	$(0.30, 0.03, 0.36)$	0.22	24/32/54	0.327149654332549
	(2, 3)	$0.00755+0.51312i$	$4\leftrightarrow 12$	1.27×10^{-13}	$(-0.27, 0.26, 0.11)$	0.15	24/32/54	0.326923948368145
		$0.00755-0.51312i$	$4\leftrightarrow 11$	5.91×10^{-14}	$(-0.26, 0.28, 0.09)$	0.15	24/30/52	0.326482313024283
		$-3.00755+0.51312i$	$4\leftrightarrow 13$ 4	1.13×10^{-14} 1.15×10^{-14}	$(0.01, 0.07, 0.35)$	0.13	24/32/54	0.327149654332475 0.327149654332484
		$-3.00755-0.51312i$	$4\leftrightarrow 14$	1.35×10^{-14}	$(0.28, 0.24, 0.21)$	0.18	24/32/54	0.327149654332518
4	(3, 2)	$-0.66222+0.89978i$	$3\leftrightarrow 14$	8.82×10^{-14}	$(0.04, 0.40, 0.06)$	0.16	28/34/60	0.517316315432455
		$-0.66222-0.89978i$	$3\leftrightarrow 13$	5.66×10^{-14}	$(0.21, 0.36, 0.23)$	0.23	24/24/46	0.515836186082985
		$-2.33778+0.89978i$	$4\leftrightarrow 14$	9.53×10^{-14}	$(0.27, 0.37, 0.15)$	0.23	26/38/62	0.516551679156900
		$-2.33778-0.89978i$	$5\leftrightarrow 14$ 5	9.57×10^{-14} 1.05×10^{-13}	$(0.44, 0.23, 0.30)$	0.34	24/30/52	0.517265636050405 0.517265636050411
	(2, 3)	$-0.66222+0.89978i$	$4\leftrightarrow 13$	6.75×10^{-14}	$(0.07, 0.35, 0.03)$	0.13	28/34/60	0.517270575096117
		$-0.66222-0.89978i$	$4\leftrightarrow 13$	1.18×10^{-13}	$(0.01, 0.33, 0.13)$	0.12	28/34/60	0.516823508294407
		$-2.33778+0.89978i$	$5\leftrightarrow 14$	9.15×10^{-14}	$(0.28, 0.30, -0.17)$	0.20	26/38/62	0.517091710314763
		$-2.33778-0.89978i$	$4\leftrightarrow 14$	1.40×10^{-13}	$(0.33, 0.23, 0.26)$	0.23	24/32/54	0.517371884527065
5	(3, 2)	$-2.81516+0.71242i$	$5\leftrightarrow 12$	2.00×10^{-14}	$(0.42, 0.32, 0.25)$	0.35	24/26/48	0.517468087113118
		$-2.81516-0.71242i$	$5\leftrightarrow 13$	2.43×10^{-13}	$(0.42, 0.03, 0.41)$	0.34	24/26/48	0.517468087115553
		$-0.18484+0.71242i$	$5\leftrightarrow 13$	7.19×10^{-14}	$(-0.03, 0.40, 0.03)$	0.16	24/30/52	0.517265636050197
		$-0.18484-0.71242i$	$4\leftrightarrow 13$	5.80×10^{-14}	$(0.04, 0.34, 0.22)$	0.17	24/32/54	0.517383226173917
	(2, 3)	$-2.81516+0.71242i$	$5\leftrightarrow 14$	8.42×10^{-14}	$(-0.07, 0.10, 0.51)$	0.27	24/26/48	0.517468087113785
		$-2.81516-0.71242i$	$3\leftrightarrow 14$	7.22×10^{-14}	$(0.37, 0.22, 0.29)$	0.27	24/26/48	0.517468087113621
		$-0.18484+0.71242i$	$4\leftrightarrow 13$	2.56×10^{-14}	$(-0.09, 0.27, 0.23)$	0.13	24/32/54	0.517030228001522
		$-0.18484-0.71242i$	$4\leftrightarrow 14$ 14	4.45×10^{-14} 4.42×10^{-14}	$(-0.01, 0.35, 0.01)$	0.12	24/30/52	0.517064988010753 0.517064988010771
6	(3, 2)	$-2.20711+0.97832i$	$4\leftrightarrow 13$	6.13×10^{-13}	$(0.21, 0.38, 0.14)$	0.21	36/44/78	0.682937609993004
		$-2.20711-0.97832i$	$4\leftrightarrow 13$	2.87×10^{-13}	$(0.49, 0.29, 0.31)$	0.42	34/44/76	0.685854167198917
		$-0.79289+0.97832i$	$3\leftrightarrow 13$ 3	6.28×10^{-14} 5.33×10^{-14}	$(0.15, 0.44, 0.06)$	0.22	28/40/66	0.685358137796434 0.685358137796378
		$-0.79289-0.97832i$	$6\leftrightarrow 13$	4.88×10^{-14}	$(0.14, 0.38, 0.17)$	0.19	26/30/54	0.687624377722874
	(2, 3)	$-2.20711+0.97832i$	$4\leftrightarrow 13$	2.58×10^{-13}	$(0.25, 0.33, -0.12)$	0.19	36/44/78	0.681939340447133
		$-2.20711-0.97832i$	$4\leftrightarrow 14$	2.22×10^{-13}	$(0.25, 0.34, 0.11)$	0.19	36/44/78	0.681939340447074
		$-0.79289+0.97832i$	$4\leftrightarrow 13$	4.43×10^{-14}	$(0.09, 0.34, 0.12)$	0.14	26/30/54	0.686611589632577
		$-0.79289-0.97832i$	$5\leftrightarrow 13$	2.91×10^{-14}	$(0.12, 0.36, 0.15)$	0.17	28/40/66	0.686008759135839
7	(3, 2)	$-3.21021+0.41375i$	$4\leftrightarrow 14$	8.84×10^{-14}	$(-0.27, 0.08, 0.40)$	0.24	24/30/52	0.463638985399381
		$-3.21021-0.41375i$	$4\leftrightarrow 14$ 14	1.05×10^{-13} 6.73×10^{-14}	$(-0.28, 0.30, 0.29)$	0.25	24/30/52	0.463638985399384 0.463638985399429
		$0.21021+0.41375i$	$4\leftrightarrow 12$	1.41×10^{-13}	$(-0.20, 0.36, -0.01)$	0.17	26/36/60	0.463543631310823
		$0.21021-0.41375i$	$4\leftrightarrow 12$	9.66×10^{-14}	$(-0.21, 0.28, 0.21)$	0.17	26/36/60	0.462317766141817
	(2, 3)	$-3.21021+0.41375i$	$4\leftrightarrow 13$ 13	2.26×10^{-14} 6.81×10^{-14}	$(0.09, 0.05, 0.36)$	0.14	24/30/52	0.463638985399296 0.463638985399254
		$-3.21021-0.41375i$	$4\leftrightarrow 13$	2.07×10^{-14}	$(0.37, 0.29, 0.39)$	0.38	22/24/44	0.463893546593001
		$0.21021+0.41375i$	$4\leftrightarrow 12$	1.22×10^{-13}	$(-0.24, 0.28, 0.12)$	0.15	26/36/60	0.461782733883807
		$0.21021-0.41375i$	$4\leftrightarrow 13$	8.53×10^{-14}	$(-0.23, 0.30, 0.10)$	0.15	26/36/60	0.463638985399803

Tab. C.10: Data from Table 3.1 of [62] Part I.

n	(p,q)	γ	ε_V	deviation	B	$\|\|B\|\|_E^2$	S/V/E	covolume
8	$(3,2)$	$-1.70658+1.00144i$	$7\leftrightarrow 13$	8.84×10^{-14}	$(0.08,0.35,0.12)$	0.14	28/32/58	0.463291347321593
		$-1.70658-1.00144i$	$4\leftrightarrow 13$	6.51×10^{-14}	$(0.08,0.35,-0.12)$	0.14	28/32/58	0.463291347321527
		$-1.29342+1.00144i$	$4\leftrightarrow 13$	2.95×10^{-14}	$(0.25,0.40,0.10)$	0.23	22/24/44	0.463972819748543
		$-1.29342-1.00144i$	$4\leftrightarrow 13$	2.82×10^{-14}	$(0.20,0.38,0.14)$	0.21	22/24/44	0.463493123342520
	$(2,3)$	$-1.70658+1.00144i$	$4\leftrightarrow 14$	1.07×10^{-13}	$(0.12,0.32,0.04)$	0.12	28/32/58	0.464944033504322
		$-1.70658-1.00144i$	$5\leftrightarrow 14$	1.17×10^{-13}	$(0.10,0.32,0.06)$	0.11	28/32/58	0.465170051293689
		$-1.29342+1.00144i$	$3\leftrightarrow 13$	4.88×10^{-14}	$(0.06,0.15,0.34)$	0.14	22/24/44	0.463884802808912
		$-1.29342-1.00144i$	$3\leftrightarrow 13$	2.02×10^{-14}	$(0.21,0.31,0.19)$	0.18	22/24/44	0.463972819748495
9	$(3,2)$	$-3.02268+0.62320i$	$4\leftrightarrow 14$	1.97×10^{-13}	$(0.39,0.33,0.25)$	0.32	30/36/64	0.647382627247724
		$-3.02268-0.62320i$	$5\leftrightarrow 14$	1.30×10^{-13}	$(-0.36,0.27,0.31)$	0.30	30/36/64	0.647382627247248
		$0.02268+0.62320i$	$4\leftrightarrow 13$	7.93×10^{-14}	$(0.05,0.44,-0.00)$	0.20	26/34/58	0.647085571543276
		$0.02268-0.62320i$	$4\leftrightarrow 13$	1.67×10^{-13}	$(0.09,0.43,0.10)$	0.20	26/34/58	0.647085571544311
	$(2,3)$	$-3.02268+0.62320i$	$5\leftrightarrow 13$	3.97×10^{-14}	$(0.02,0.09,0.44)$	0.20	30/36/64	0.646287064285721
		$-3.02268-0.62320i$	$4\leftrightarrow 14$ 14	1.95×10^{-13} 1.73×10^{-13}	$(0.37,0.25,0.26)$	0.26	30/36/64	0.647387982377141 0.647387982377091
		$0.02268+0.62320i$	$4\leftrightarrow 12$	1.10×10^{-13}	$(-0.00,0.36,-0.02)$	0.13	26/36/60	0.646684596070416
		$0.02268-0.62320i$	$4\leftrightarrow 13$	1.27×10^{-13}	$(-0.03,0.33,0.19)$	0.15	26/34/58	0.647085571543479
10	$(3,2)$	$-0.37053+0.84016i$	$5\leftrightarrow 13$	1.07×10^{-13}	$(0.13,0.45,0.03)$	0.22	30/40/68	0.646819936948037
		$-0.37053-0.84016i$	$4\leftrightarrow 12$	5.55×10^{-14}	$(0.19,0.41,0.19)$	0.24	30/40/68	0.647079332036773
		$-2.62947+0.84016i$	$5\leftrightarrow 11$	9.69×10^{-14}	$(0.40,0.36,0.21)$	0.34	26/34/58	0.647085571542674
		$-2.62947-0.84016i$	$5\leftrightarrow 13$	1.06×10^{-13}	$(0.39,-0.02,0.43)$	0.34	26/34/58	0.647085571542876
	$(2,3)$	$-0.37053+0.84016i$	$4\leftrightarrow 13$ 4	8.04×10^{-14} 8.13×10^{-14}	$(-0.07,0.25,0.38)$	0.21	30/40/68	0.646819936947746 0.646819936947737
		$-0.37053-0.84016i$	$4\leftrightarrow 13$	3.04×10^{-13}	$(0.08,0.36,0.18)$	0.17	30/40/68	0.646819936948211
		$-2.62947+0.84016i$	$4\leftrightarrow 14$	2.46×10^{-13}	$(-0.15,0.09,0.51)$	0.29	26/34/58	0.647085571542869
		$-2.62947-0.84016i$	$4\leftrightarrow 14$	1.95×10^{-13}	$(0.38,0.27,0.26)$	0.28	26/34/58	0.647085571542963
11	$(3,2)$	$-2.87744+0.74486i$	$5\leftrightarrow 14$ $5,6$	7.15×10^{-14}	$(0.46,0.34,0.27)$	0.40	26/32/56	0.786130813717339 0.786130813717339
			7	failure.				
		$-2.87744-0.74486i$	$4\leftrightarrow 13$	3.16×10^{-13}	$(0.45,0.03,0.44)$	0.40	26/32/56	0.786225296258149
		$-0.12256+0.74486i$	$5\leftrightarrow 13$	3.78×10^{-13}	$(0.06,0.44,0.02)$	0.20	30/38/66	0.785863578105783
		$-0.12256-0.74486i$	$4\leftrightarrow 13$	2.05×10^{-13}	$(0.19,0.39,0.27)$	0.26	28/38/64	0.785794396515464
	$(2,3)$	$-2.87744+0.74486i$	$4\leftrightarrow 13$ 4	5.36×10^{-13} 5.34×10^{-13}	$(-0.04,0.11,0.54)$	0.30	26/32/56	0.786222380061525 0.786222380061523
		$-2.87744-0.74486i$	$4\leftrightarrow 13$	3.38×10^{-13}	$(0.41,0.23,0.30)$	0.32	26/32/56	0.785957346960051
		$-0.12256+0.74486i$	$4\leftrightarrow 13$	1.52×10^{-13}	$(0.08,0.38,0.04)$	0.15	30/38/66	0.785999075910518
		$-0.12256-0.74486i$	$4\leftrightarrow 12$ 4	2.54×10^{-14}	$(-0.01,0.34,0.18)$	0.15	30/38/66	0.785999075910039 0.785999075910034
			5	failure.				

Tab. C.11: Data from Table 3.1 of [62] Part II.

n	(p,q)	γ	ε_V	deviation	B	$\|\|B\|\|_E^2$	S/V/E	covolume
12	$(3,2)$	$-2.24813+1.03398i$	$7\leftrightarrow 14$	4.07×10^{-12}	$(0.26,0.41,0.15)$	0.26	34/42/74	1.141541830593926
			14	3.37×10^{-12}				1.141541830593913
		$-2.24813-1.03398i$	$5\leftrightarrow 13$	2.50×10^{-12}	$(0.53,0.07,0.43)$	0.47	30/38/66	1.150966806397200
			5	2.84×10^{-12}				1.150966806397164
		$-0.75187+1.03398i$	$4\leftrightarrow 13$	2.62×10^{-12}	$(0.20,0.47,0.06)$	0.26	36/44/78	1.144557251570631
		$-0.75187-1.03398i$	$5\leftrightarrow 14$	4.00×10^{-12}	$(0.20,0.40,0.19)$	0.24	36/42/76	1.144289443644866
			14	2.06×10^{-12}				1.144289443644726
	$(2,3)$	$-2.24813+1.03398i$	$5\leftrightarrow 12$	5.68×10^{-13}	$(0.30,0.35,-0.14)$	0.23	34/42/74	1.145197409428369
		$-2.24813-1.03398i$	$8\leftrightarrow 13$	7.73×10^{-13}	$(0.43,0.14,0.36)$	0.33	30/38/66	1.147770689218986
		$-0.75187+1.03398i$	$5\leftrightarrow 14$	1.50×10^{-12}	$(0.14,0.36,0.15)$	0.17	36/42/76	1.144776469725223
		$-0.75187-1.03398i$	$6\leftrightarrow 13$	2.74×10^{-12}	$(0.18,0.38,0.17)$	0.21	36/44/78	1.145096650762742
13	$(3,2)$	$-3.34815+0.31570i$	$4\leftrightarrow 14$	1.10×10^{-13}	$(-0.34,0.32,0.26)$	0.29	30/40/68	0.567218210159849
			4	1.09×10^{-13}				0.567218210159845
		$-3.34815-0.31570i$	$5\leftrightarrow 14$	5.84×10^{-14}	$(-0.29,0.35,0.26)$	0.27	26/32/56	0.567165208527812
		$0.34815+0.31570i$	$5\leftrightarrow 12$	1.01×10^{-13}	$(-0.18,0.38,-0.03)$	0.18	26/38/62	0.567165208528317
		$0.34815-0.31570i$	$4\leftrightarrow 12$	6.25×10^{-14}	$(-0.20,0.28,0.24)$	0.18	26/38/62	0.567165208527376
			4	6.58×10^{-14}				0.567165208527338
			5	failure.				
	$(2,3)$	$-3.34815+0.31570i$	$5\leftrightarrow 13$	1.19×10^{-14}	$(0.10,0.03,0.38)$	0.16	26/32/56	0.567165208527598
		$-3.34815-0.31570i$	$4\leftrightarrow 14$	3.51×10^{-14}	$(0.38,0.36,0.30)$	0.36	26/32/56	0.567165208527755
			14	2.19×10^{-14}				0.567165208527715
		$0.34815+0.31570i$	$3\leftrightarrow 12$	3.47×10^{-13}	$(-0.22,0.30,0.12)$	0.15	26/38/62	0.567049507229398
			3	3.57×10^{-13}				0.567049507229602
			4,5,6	failure.				
		$0.34815-0.31570i$	$4\leftrightarrow 12$	2.38×10^{-13}	$(-0.21,0.31,0.11)$	0.16	26/38/62	0.566948496846089
14	$(3,2)$	$-1.64913+1.05848i$	$4\leftrightarrow 13$	2.49×10^{-14}	$(0.09,0.37,0.12)$	0.16	26/32/56	0.567285975023820
		$-1.64913-1.05848i$	$4\leftrightarrow 13$	2.18×10^{-14}	$(0.09,0.37,-0.13)$	0.16	26/32/56	0.567285975023829
		$-1.35087+1.05848i$	$3\leftrightarrow 13$	2.51×10^{-14}	$(0.36,0.42,0.13)$	0.32	26/30/54	0.567162499322202
			3,4				22/23/43	0.567164292397292
			5	failure.				
		$-1.35087-1.05848i$	$4\leftrightarrow 13$	5.48×10^{-14}	$(0.23,0.43,0.05)$	0.24	26/28/52	0.567165208528485
	$(2,3)$	$-1.64913+1.05848i$	$3\leftrightarrow 14$	1.33×10^{-13}	$(0.14,0.33,0.05)$	0.13	26/32/56	0.567285975024011
		$-1.64913-1.05848i$	$4\leftrightarrow 13$	5.72×10^{-14}	$(0.12,0.33,0.07)$	0.13	26/32/56	0.567165208527762
		$-1.35087+1.05848i$	$3\leftrightarrow 13$	7.79×10^{-14}	$(0.18,0.14,0.37)$	0.19	26/30/54	0.567162499322480
			3,4				22/23/43	0.567164292397572
			5	failure.				
		$-1.35087-1.05848i$	$4\leftrightarrow 14$	1.01×10^{-13}	$(0.32,0.30,0.25)$	0.25	26/30/54	0.567162499322576
			4				22/23/43	0.567164292397665
			5	failure.				
15	$(3,2)$	$-1.97873+1.12212i$	$6\leftrightarrow 11$	2.86×10^{-11}	$(0.16,0.44,0.15)$	0.24	54/82/134	1.893377029749469
		$-1.97873-1.12212i$	$6\leftrightarrow 11$	1.16×10^{-11}	$(0.50,0.38,0.27)$	0.46	38/50/86	1.885892195233329
		$-1.02127+1.12212i$	$5\leftrightarrow 13$	4.84×10^{-11}	$(0.25,0.49,0.07)$	0.31	46/64/108	1.886872709877064
		$-1.02127-1.12212i$	$5\leftrightarrow 13$	3.19×10^{-11}	$(0.32,0.47,0.11)$	0.34	46/64/108	1.880774913810047
	$(2,3)$	$-1.97873+1.12212i$	$5\leftrightarrow 13$	6.09×10^{-11}	$(0.23,0.12,0.41)$	0.23	38/50/86	1.884891575291061
		$-1.97873-1.12212i$	$5\leftrightarrow 13$	1.57×10^{-10}	$(0.24,0.37,0.09)$	0.20	54/82/134	1.895705583407503
		$-1.02127+1.12212i$	$6\leftrightarrow 11$	8.40×10^{-12}	$(-0.03,0.24,0.45)$	0.26	46/64/108	1.894601048848805
		$-1.02127-1.12212i$	$5\leftrightarrow 13$	3.92×10^{-11}	$(0.25,0.40,0.17)$	0.25	46/64/108	1.883406819520532

Tab. C.12: Data from Table 3.1 of [62] Part III.

n	(p,q)	γ	ε_V	deviation	B	$\|\|B\|\|_E^2$	S/V/E	covolume
1	$(3,2)$	$-0.76721-0.79255i$	$3\leftrightarrow 13$	1.64×10^{-14}	$(-0.02,0.31,0.12)$	0.11	20/20/38	0.263723173448867
		$-0.76721+0.79255i$	$4\leftrightarrow 14$	7.33×10^{-15}	$(-0.05,0.33,0.07)$	0.12	22/26/46	0.263487375455115
		$-2.23279-0.79255i$	$5\leftrightarrow 13$	1.53×10^{-14}	$(0.34,0.22,0.24)$	0.23	22/26/46	0.263518203239260
		$-2.23279+0.79255i$	$4\leftrightarrow 14$ 4	3.20×10^{-14} 2.80×10^{-14}	$(0.16,0.32,0.13)$	0.14	24/34/56	0.263688965427039 0.263688965427042
	$(2,3)$	$-0.76721-0.79255i$	$4\leftrightarrow 13$	1.08×10^{-14}	$(-0.04,0.30,0.01)$	0.09	22/26/46	0.263394331688615
		$-0.76721+0.79255i$	$4\leftrightarrow 13$	1.52×10^{-14}	$(-0.04,0.29,-0.01)$	0.09	22/26/46	0.263394331688655
		$-2.23279-0.79255i$	$5\leftrightarrow 14$	9.10×10^{-15}	$(0.16,0.28,0.10)$	0.11	24/34/56	0.262917569081549
		$-2.23279+0.79255i$	$5\leftrightarrow 14$	6.00×10^{-15}	$(0.16,0.28,-0.11)$	0.11	24/34/56	0.263008715157447
2	$(3,2)$	$-0.34861+0.75874i$	$4\leftrightarrow 14$	3.09×10^{-14}	$(0.02,0.40,0.04)$	0.16	26/34/58	0.432564598948254
		$-0.34861-0.75874i$	$4\leftrightarrow 13$	4.22×10^{-14}	$(0.07,0.39,0.09)$	0.17	26/34/58	0.432548106138239
		$-2.65139+0.75874i$	$4\leftrightarrow 14$	2.33×10^{-14}	$(0.37,0.33,0.21)$	0.28	26/34/58	0.432564598947727
		$-2.65139-0.75874i$	$4\leftrightarrow 13$	4.84×10^{-14}	$(0.36,-0.00,0.39)$	0.28	26/34/58	0.432414119827482
	$(2,3)$	$-0.34861+0.75874i$	$4\leftrightarrow 13$	4.22×10^{-14}	$(-0.04,0.31,-0.15)$	0.12	26/34/58	0.432564598948507
		$-0.34861-0.75874i$	$4\leftrightarrow 13$	6.64×10^{-14}	$(-0.04,0.32,0.15)$	0.13	26/34/58	0.432564598949059
		$-2.65139+0.75874i$	$5\leftrightarrow 14$	2.94×10^{-14}	$(0.33,0.24,-0.25)$	0.22	26/34/58	0.432564598947876
		$-2.65139-0.75874i$	$4\leftrightarrow 13$	8.10×10^{-15}	$(0.33,0.24,0.24)$	0.22	26/34/58	0.432564598947726
3	$(3,2)$	$-1.00000+1.00000i$	$3\leftrightarrow 14$	1.09×10^{-13}	$(0.15,0.43,0.07)$	0.21	26/30/54	0.610778931489704
		$-1.00000-1.00000i$	$4\leftrightarrow 14$	2.16×10^{-13}	$(0.21,0.43,0.06)$	0.23	26/30/54	0.611172970066301
		$-2.00000+1.00000i$	$5\leftrightarrow 13$	7.99×10^{-14}	$(0.13,0.39,0.13)$	0.18	30/42/70	0.610009854824228
		$-2.00000-1.00000i$	$5\leftrightarrow 13$	2.38×10^{-14}	$(0.42,0.33,0.25)$	0.35	24/30/52	0.610519526694357
	$(2,3)$	$-1.00000+1.00000i$	$4\leftrightarrow 14$ 14	2.08×10^{-13} 9.57×10^{-14}	$(0.14,0.35,-0.15)$ $(0.14,0.35,-0.15)$	0.16 0.16	26/30/54 26/30/54	0.610632545608902 0.610632545608938
		$-1.00000-1.00000i$	$3\leftrightarrow 13$	7.06×10^{-14}	$(0.14,0.35,0.14)$	0.16	26/30/54	0.610797931045570
		$-2.00000+1.00000i$	$3\leftrightarrow 14$ 3	8.50×10^{-14}	$(0.18,0.09,0.36)$	0.17	24/30/52	0.610519526695058 0.610519526695014
			4	failure.				
		$-2.00000-1.00000i$	$4\leftrightarrow 14$	2.48×10^{-13}	$(0.18,0.33,0.07)$	0.15	30/42/70	0.610416962118645
4	$(3,2)$	$-0.60186+0.93867i$	$4\leftrightarrow 13$	1.83×10^{-13}	$(0.09,0.42,0.06)$	0.19	28/34/60	0.710901322036743
		$-0.60186-0.93867i$	$4\leftrightarrow 14$	1.05×10^{-13}	$(0.23,0.40,0.24)$	0.27	26/28/52	0.711606386948809
		$-2.39814+0.93867i$	$5\leftrightarrow 13$	6.84×10^{-14}	$(0.33,0.39,0.17)$	0.29	24/34/56	0.711338778934043
		$-2.39814-0.93867i$	$5\leftrightarrow 13$	5.84×10^{-14}	$(0.39,-0.04,0.43)$	0.34	26/38/62	0.711117203273613
	$(2,3)$	$-0.60186+0.93867i$	$4\leftrightarrow 14$	2.30×10^{-13}	$(0.11,0.37,0.07)$	0.15	28/34/60	0.712315086378573
		$-0.60186-0.93867i$	$4\leftrightarrow 13$	3.78×10^{-13}	$(0.05,0.35,0.14)$	0.14	28/34/60	0.711763881371836
		$-2.39814+0.93867i$	$5\leftrightarrow 14$	2.06×10^{-13}	$(0.34,0.31,-0.21)$	0.25	24/34/56	0.711338778934606
		$-2.39814-0.93867i$	$5\leftrightarrow 13$	7.03×10^{-14}	$(0.38,0.27,0.26)$	0.28	26/38/62	0.710938595008224
5	$(3,2)$	$0.11005+0.57190i$	$5\leftrightarrow 13$	1.87×10^{-13}	$(-0.02,0.42,-0.00)$	0.18	26/36/60	0.710510137161106
		$0.11005-0.57190i$	$4\leftrightarrow 13$	1.84×10^{-13}	$(0.07,0.35,0.28)$	0.20	26/38/62	0.710579473464925
		$-3.11005+0.57190i$	$5\leftrightarrow 14$	2.72×10^{-13}	$(0.39,0.34,0.25)$	0.33	28/40/66	0.712180070353679
		$-3.11005-0.57190i$	$5\leftrightarrow 14$	2.03×10^{-13}	$(-0.33,0.28,0.32)$	0.29	28/40/66	0.712180070353544
	$(2,3)$	$0.11005+0.57190i$	$4\leftrightarrow 13$	1.19×10^{-13}	$(-0.07,0.28,0.27)$	0.16	26/38/62	0.710579473464162
		$0.11005-0.57190i$	$4\leftrightarrow 12$	1.06×10^{-13}	$(0.00,0.37,0.01)$	0.14	26/36/60	0.710753951013632
		$-3.11005+0.57190i$	$4\leftrightarrow 13$	8.37×10^{-14}	$(0.06,0.08,0.42)$	0.18	28/40/66	0.712543560125077
		$-3.11005-0.57190i$	$5\leftrightarrow 13$	2.45×10^{-13}	$(0.37,0.27,0.25)$	0.28	28/40/66	0.711612113759270
6	$(3,2)$	$-1.86240+1.07589i$	$5\leftrightarrow 14$	1.20×10^{-12}	$(0.13,0.39,0.13)$	0.18	32/48/78	0.866178421227291
		$-1.86240-1.07589i$	$4\leftrightarrow 14$	1.09×10^{-12}	$(0.13,0.38,-0.13)$	0.18	32/48/78	0.866178421227852
		$-1.13760+1.07589i$	$4\leftrightarrow 13$	1.49×10^{-13}	$(0.21,0.46,0.08)$	0.26	28/36/62	0.865996064707863
		$-1.13760-1.07589i$	$5\leftrightarrow 14$	5.05×10^{-13}	$(0.28,0.43,0.17)$	0.29	28/36/62	0.864674947308066
	$(2,3)$	$-1.86240+1.07589i$	$3\leftrightarrow 13$	2.82×10^{-13}	$(0.14,0.13,0.37)$	0.17	28/34/60	0.866237338993355
		$-1.86240-1.07589i$	$4\leftrightarrow 14$ 14	8.87×10^{-13} 6.66×10^{-13}	$(0.18,0.35,0.07)$	0.16	32/48/78	0.865353485572168 0.865353485572135
		$-1.13760+1.07589i$	$5\leftrightarrow 13$	1.74×10^{-13}	$(-0.00,0.21,0.37)$	0.18	28/36/62	0.864774791781850
		$-1.13760-1.07589i$	$4\leftrightarrow 13$	3.30×10^{-13}	$(0.21,0.38,0.15)$	0.21	28/36/62	0.864164682655471

Tab. C.13: Data from Table 5.1 of [62] Part I.

n	(p,q)	γ	ε_V	deviation	B	$\|\|B\|\|_E^2$	S/V/E	covolume
7	$(3,2)$	$-0.23931 - 0.85787i$	$5 \leftrightarrow 13$	3.71×10^{-12}	$(0.20, 0.39, 0.26)$	0.26	38/60/96	1.252307176564980
		$-0.23931 + 0.85787i$	$4 \leftrightarrow 13$	1.97×10^{-12}	$(0.24, 0.51, 0.01)$	0.32	32/44/74	1.252928202049658
		$-2.76069 - 0.85787i$	$6 \leftrightarrow 12$	1.43×10^{-11}	$(0.48, -0.00, 0.47)$	0.45	38/48/84	1.250321943401754
		$-2.76069 + 0.85787i$	$5 \leftrightarrow 13$	9.92×10^{-12}	$(0.50, 0.38, 0.26)$	0.46	38/48/84	1.250713782739033
	$(2,3)$	$-0.23931 - 0.85787i$	$5 \leftrightarrow 13$	9.84×10^{-13}	$(0.19, 0.39, 0.22)$	0.24	32/44/74	1.255231836333305
		$-0.23931 + 0.85787i$	$5 \leftrightarrow 13$	8.22×10^{-12}	$(0.17, 0.41, 0.03)$	0.20	38/60/96	1.251442419208005
		$-2.76069 - 0.85787i$	$5 \leftrightarrow 13$	4.29×10^{-12}	$(0.46, 0.26, 0.32)$	0.37	38/48/84	1.251828263901439
		$-2.76069 + 0.85787i$	$5 \leftrightarrow 13$	4.56×10^{-12}	$(-0.10, 0.12, 0.59)$	0.37	38/48/84	1.252338870907436
8	$(3,2)$	$-1.77184 + 1.11514i$	$5 \leftrightarrow 14$	5.55×10^{-13}	$(0.14, 0.39, 0.13)$	0.19	32/44/74	1.067793082470478
		$-1.77184 - 1.11514i$	$4 \leftrightarrow 13$	7.93×10^{-13}	$(0.14, 0.39, -0.13)$	0.19	32/44/74	1.067737300243070
		$-1.22816 + 1.11514i$	$4 \leftrightarrow 14$	7.33×10^{-13}	$(0.29, 0.47, 0.09)$	0.31	28/34/60	1.059048949359196
		$-1.22816 - 1.11514i$	$4 \leftrightarrow 11$	3.44×10^{-13}	$(0.29, 0.46, -0.10)$	0.31	28/34/60	1.059048949359549
	$(2,3)$	$-1.77184 + 1.11514i$	$4 \leftrightarrow 13$	5.49×10^{-13}	$(0.20, 0.36, 0.01)$	0.17	32/44/74	1.067412724658020
		$-1.77184 - 1.11514i$	$4 \leftrightarrow 14$	1.42×10^{-12}	$(0.18, 0.35, 0.08)$	0.16	32/44/74	1.060315838908867
		$-1.22816 + 1.11514i$	$5 \leftrightarrow 13$	7.64×10^{-13}	$(0.10, 0.19, 0.38)$	0.19	28/34/60	1.059072575346030
		$-1.22816 - 1.11514i$	$3 \leftrightarrow 14$	6.60×10^{-13}	$(0.28, 0.37, 0.19)$	0.25	28/34/60	1.059072575345849
9	$(3,2)$	$-0.65219 - 1.02885i$	$5 \leftrightarrow 13$	8.60×10^{-12}	$(0.41, 0.38, 0.31)$	0.41	34/46/78	1.483691461545163
		$-0.65219 + 1.02885i$	$6 \leftrightarrow 14$	8.50×10^{-12}	$(0.18, 0.47, 0.05)$	0.26	38/54/90	1.486637828881297
		$-2.34781 - 1.02885i$	$5 \leftrightarrow 12$	7.12×10^{-12}	$(0.48, -0.01, 0.46)$	0.44	38/50/86	1.496113936056134
		$-2.34781 + 1.02885i$	$8 \leftrightarrow 13$	2.18×10^{-11}	$(0.32, 0.42, 0.16)$	0.31	62/98/158	1.491181485968302
	$(2,3)$	$-0.65219 - 1.02885i$	$5 \leftrightarrow 13$	9.50×10^{-12}	$(0.16, 0.38, 0.17)$	0.20	38/54/90	1.485461477905392
		$-0.65219 + 1.02885i$	$5 \leftrightarrow 13$	8.99×10^{-12}	$(0.18, 0.22, 0.39)$	0.23	34/46/78	1.486979860104871
		$-2.34781 - 1.02885i$	$6 \leftrightarrow 12$	3.66×10^{-12}	$(0.44, 0.23, 0.32)$	0.35	38/50/86	1.498880413762323
		$-2.34781 + 1.02885i$	$5 \leftrightarrow 12$	8.60×10^{-12}	$(0.35, 0.34, -0.19)$	0.28	62/98/158	1.494723230945138
			5				62/92/152	1.493844293827972
			6, 7	failure.				
10	$(3,2)$	$0.17229 + 0.58559i$	$5 \leftrightarrow 12$	1.14×10^{-13}	$(0.12, 0.48, -0.02)$	0.24	32/44/74	1.178776902865389
		$0.17229 - 0.58559i$	$4 \leftrightarrow 12$	5.12×10^{-13}	$(0.25, 0.44, 0.24)$	0.32	30/42/70	1.181360578216781
		$-3.17229 + 0.58559i$	$5 \leftrightarrow 13$	1.82×10^{-12}	$(-0.34, 0.12, 0.44)$	0.32	32/46/76	1.177589893407057
		$-3.17229 - 0.58559i$	$5 \leftrightarrow 14$	2.30×10^{-12}	$(-0.35, 0.30, 0.34)$	0.33	32/46/76	1.177440816718236
	$(2,3)$	$0.17229 + 0.58559i$	$4 \leftrightarrow 12$	1.01×10^{-12}	$(0.04, 0.29, 0.34)$	0.20	30/42/70	1.181360578218381
		$0.17229 - 0.58559i$	$4 \leftrightarrow 12$	4.83×10^{-12}	$(0.03, 0.36, 0.22)$	0.18	32/44/74	1.179957019878848
		$-3.17229 + 0.58559i$	$4 \leftrightarrow 13$	2.73×10^{-12}	$(0.09, 0.08, 0.45)$	0.22	32/46/76	1.179576630255273
		$-3.17229 - 0.58559i$	$6 \leftrightarrow 13$	1.65×10^{-12}	$(0.42, 0.28, 0.28)$	0.33	32/46/76	1.179374721462420
11	$(3,2)$	$-0.13972 + 0.82586i$	$5 \leftrightarrow 13$	2.89×10^{-11}	$(0.21, 0.50, 0.01)$	0.30	38/52/88	1.754207433479665
		$-0.13972 - 0.82586i$	$5 \leftrightarrow 13$	2.66×10^{-11}	$(0.25, 0.40, 0.29)$	0.31	38/54/90	1.748677559087148
		$-2.86028 + 0.82586i$	$6 \leftrightarrow 13$	2.61×10^{-11}	$(0.60, 0.39, 0.26)$	0.58	42/64/104	1.757144139163060
		$-2.86028 - 0.82586i$	$7 \leftrightarrow 10$	4.39×10^{-11}	$(0.58, -0.02, 0.49)$	0.57	42/64/104	1.757711925963126
	$(2,3)$	$-0.13972 + 0.82586i$	$5 \leftrightarrow 12$	5.84×10^{-11}	$(0.15, 0.38, 0.23)$	0.22	38/54/90	1.753904062661501
		$-0.13972 - 0.82586i$	$5 \leftrightarrow 13$	5.59×10^{-11}	$(0.15, 0.39, 0.22)$	0.22	38/52/88	1.750828480010552
			5	4.85×10^{-11}				1.750828480015106
			6	failure.				
		$-2.86028 + 0.82586i$	$6 \leftrightarrow 11$	1.23×10^{-11}	$(-0.10, 0.11, 0.67)$	0.47	42/64/104	1.750517907450094
		$-2.86028 - 0.82586i$	$5 \leftrightarrow 12$	2.36×10^{-11}	$(0.51, 0.27, 0.41)$	0.50	42/64/104	1.758382402555404

Tab. C.14: Data from Table 5.1 of [62] Part II.

n	(p,q)	γ	ε_V	deviation	B	$\|\|B\|\|_E^2$	S/V/E	covolume
1	$(-3,2)$	$0.46746+0.27759i$	$5\leftrightarrow 13$	6.37×10^{-13}	$(0.08,0.48,-0.07)$	0.24	34/52/84	1.019779352598105
		$0.46746-0.27759i$	$5\leftrightarrow 13$	5.76×10^{-13}	$(0.11,0.47,0.12)$	0.25	34/52/84	1.019118210708849
		$-3.46746+0.27759i$	$5\leftrightarrow 13$	3.75×10^{-13}	$(-0.33,0.04,0.50)$	0.36	28/36/62	1.019983204951585
		$-3.46746-0.27759i$	$4\leftrightarrow 14$	3.33×10^{-13}	$(-0.34,0.39,0.32)$	0.37	28/36/62	1.021066214332981
	$(-2,3)$	$0.46746+0.27759i$	$4\leftrightarrow 13$	1.49×10^{-12}	$(0.01,0.40,-0.01)$	0.16	32/50/80	1.019562109651294
			13	5.63×10^{-13}				1.019562109639764
		$0.46746-0.27759i$	$5\leftrightarrow 13$	6.31×10^{-14}	$(-0.03,0.34,0.25)$	0.18	34/52/84	1.019779352593147
			13	6.24×10^{-14}				1.019779352592997
		$-3.46746+0.27759i$	$5\leftrightarrow 13$	3.28×10^{-13}	$(0.18,0.02,0.45)$	0.23	28/36/62	1.020338996345516
		$-3.46746-0.27759i$	$5\leftrightarrow 13$	5.03×10^{-13}	$(0.46,0.39,0.37)$	0.50	28/36/62	1.020542848704024
2	$(-3,2)$	$-2.58153+0.93916i$	$5\leftrightarrow 12$	3.15×10^{-13}	$(0.45,0.41,0.21)$	0.41	34/52/84	1.018809897806248
		$-2.58153-0.93916i$	$5\leftrightarrow 13$	8.12×10^{-13}	$(0.42,-0.05,0.47)$	0.40	34/52/84	1.018746472843271
		$-0.41847+0.93916i$	$5\leftrightarrow 13$	7.27×10^{-13}	$(0.27,0.50,-0.05)$	0.33	30/38/66	1.019168423089651
		$-0.41847-0.93916i$	$5\leftrightarrow 12$	1.01×10^{-12}	$(0.29,0.42,0.28)$	0.34	30/38/66	1.017587293822094
	$(-2,3)$	$-2.58153+0.93916i$	$5\leftrightarrow 13$	1.09×10^{-12}	$(-0.18,0.10,0.56)$	0.35	34/52/84	1.017007935103982
		$-2.58153-0.93916i$	$5\leftrightarrow 13$	1.23×10^{-12}	$(0.43,0.30,0.27)$	0.35	34/52/84	1.018393643850323
		$-0.41847+0.93916i$	$5\leftrightarrow 12$	6.99×10^{-13}	$(0.06,0.26,0.37)$	0.20	30/38/66	1.019337403809232
		$-0.41847-0.93916i$	$5\leftrightarrow 13$	8.28×10^{-13}	$(0.21,0.45,0.27)$	0.32	30/38/66	1.019238041091295
3	$(-3,2)$	$-1.65884-1.16154i$	$6\leftrightarrow 13$	1.14×10^{-11}	$(0.14,0.40,-0.13)$	0.20	40/56/94	1.308505501629080
		$-1.65884+1.16154i$	$6\leftrightarrow 13$	1.31×10^{-11}	$(0.14,0.41,0.13)$	0.20	40/56/94	1.308866192606601
		$-1.34116-1.16154i$	$4\leftrightarrow 13$	1.97×10^{-12}	$(0.62,0.42,-0.22)$	0.62	32/42/72	1.319305550522220
		$-1.34116+1.16154i$	$5\leftrightarrow 13$	2.85×10^{-12}	$(0.62,0.43,0.21)$	0.62	32/42/72	1.319305550523099
	$(-2,3)$	$-1.65884-1.16154i$	$6\leftrightarrow 14$	1.11×10^{-11}	$(0.18,0.36,0.08)$	0.17	40/56/94	1.319474166893606
		$-1.65884+1.16154i$	$3\leftrightarrow 14$	3.79×10^{-12}	$(0.08,0.17,0.38)$	0.18	36/46/80	1.320408605566066
		$-1.34116-1.16154i$	$5\leftrightarrow 12$	2.73×10^{-12}	$(0.54,0.21,0.39)$	0.48	32/42/72	1.319305550522260
		$-1.34116+1.16154i$	$4\leftrightarrow 13$	1.91×10^{-12}	$(0.41,0.14,0.54)$	0.48	32/42/72	1.319305550522633
4	$(-3,2)$	$0.10278-0.66546i$	$6\leftrightarrow 12$	9.45×10^{-12}	$(0.21,0.39,0.31)$	0.29	38/62/98	1.709314096976351
		$0.10278+0.66546i$	$5\leftrightarrow 13$	1.29×10^{-11}	$(0.29,0.54,-0.03)$	0.38	36/52/86	1.709274223642092
		$-3.10278-0.66546i$	$6\leftrightarrow 13$	3.77×10^{-11}	$(-0.39,0.29,0.35)$	0.37	46/68/112	1.708422738797249
		$-3.10278+0.66546i$	$6\leftrightarrow 12$	7.24×10^{-12}	$(0.45,0.36,0.28)$	0.41	46/68/112	1.709353487322167
	$(-2,3)$	$0.10278-0.66546i$	$5\leftrightarrow 13$	1.33×10^{-11}	$(0.21,0.40,0.27)$	0.28	36/52/86	1.709344065589438
		$0.10278+0.66546i$	$5\leftrightarrow 13$	8.80×10^{-12}	$(0.18,0.43,0.07)$	0.22	38/62/98	1.709191960284029
		$-3.10278-0.66546i$	$5\leftrightarrow 11$	7.34×10^{-12}	$(0.43,0.27,0.29)$	0.35	46/68/112	1.708508743153964
		$-3.10278+0.66546i$	$5\leftrightarrow 12$	4.14×10^{-11}	$(0.06,0.10,0.49)$	0.25	46/68/112	1.709394610761968
5	$(-3,2)$	$0.33909+0.44663i$	$5\leftrightarrow 12$	7.73×10^{-12}	$(0.16,0.50,-0.05)$	0.28	40/56/94	1.472893660595506
		$0.33909-0.44663i$	$4\leftrightarrow 10$	3.91×10^{-12}	$(0.28,0.48,0.22)$	0.36	36/50/84	1.474832589950734
			5	failure.				
		$-3.33909+0.44663i$	$6\leftrightarrow 13$	6.70×10^{-12}	$(-0.43,0.10,0.49)$	0.43	36/48/82	1.472712205090256
		$-3.33909-0.44663i$	$5\leftrightarrow 13$	4.50×10^{-12}	$(-0.44,0.34,0.36)$	0.44	36/48/82	1.474788761845281
	$(-2,3)$	$0.33909+0.44663i$	$4\leftrightarrow 12$	7.20×10^{-12}	$(0.08,0.32,0.34)$	0.23	36/50/84	1.475037573340682
		$0.33909-0.44663i$	$6\leftrightarrow 11$	1.41×10^{-11}	$(0.06,0.37,0.25)$	0.20	40/56/94	1.472188300244882
		$-3.33909+0.44663i$	$4\leftrightarrow 11$	3.40×10^{-12}	$(0.14,0.07,0.55)$	0.33	36/48/82	1.474788761839685
		$-3.33909-0.44663i$	$6\leftrightarrow 12$	2.70×10^{-12}	$(0.48,0.31,0.42)$	0.50	36/48/82	1.472909330019834

Tab. C.15: Data from Table 5.4 of [62].

n	(p,q)	γ	ε_V	deviation	B	$\|\|B\|\|_E^2$	S/V/E	covolume
1	$(3,2)$	$-0.50000+0.86603i$	$4\leftrightarrow 14$	2.20×10^{-14}	$(0.05,0.41,0.05)$	0.17	22/23/44	0.676398849581584
		$-0.50000-0.86603i$	$4\leftrightarrow 13$	2.55×10^{-14}	$(0.16,0.38,0.22)$	0.21	22/23/44	0.676760804700557
		$-2.50000+0.86603i$	$3\leftrightarrow 13$	3.55×10^{-15}	$(0.36,0.37,0.19)$	0.30	14/10/24	0.676627737605910
		$-2.50000-0.86603i$	$4\leftrightarrow 14$	4.88×10^{-15}	$(0.45,0.18,0.33)$	0.35	8/7/15	0.676627737607669
	$(2,3)$	$-0.50000+0.86603i$	$3\leftrightarrow 13$	3.24×10^{-14}	$(0.01,0.30,0.20)$	0.13	22/23/44	0.676563238051973
		$-0.50000-0.86603i$	$3\leftrightarrow 13$	1.57×10^{-14}	$(0.01,0.33,0.14)$	0.13	22/23/44	0.676563238050312
		$-2.50000+0.86603i$	$3\leftrightarrow 14$	3.55×10^{-15}	$(0.35,0.28,-0.24)$	0.25	14/10/24	0.676627737605571
		$-2.50000-0.86603i$	$3\leftrightarrow 13$	5.66×10^{-15}	$(0.35,0.28,0.23)$	0.26	14/10/24	0.676627737605229
2	$(3,2)$	$0.36778+0.23154i$	$4\leftrightarrow 12$	1.55×10^{-12}	$(0.26,0.13,0.39)$	0.24	28/26/52	0.192584612983471
		$0.36778-0.23154i$	$5\leftrightarrow 12$	4.22×10^{-12}	$(0.26,0.27,0.32)$	0.24	28/26/52	0.192406713793864
		$-3.36778+0.23154i$	$4\leftrightarrow 14$	1.55×10^{-14}	$(-0.23,0.13,0.37)$	0.21	20/20/38	0.385365362960497
			4					0.385365362960489
			5	failure.				
		$-3.36778-0.23154i$	$3\leftrightarrow 14$	3.06×10^{-14}	$(-0.23,0.24,0.31)$	0.21	20/20/38	0.385610022684702
	$(2,3)$	$0.36778+0.23154i$	$4\leftrightarrow 12$	4.76×10^{-12}	$(0.02,0.26,0.21)$	0.11	28/26/52	0.192773545478196
		$0.36778-0.23154i$	$3\leftrightarrow 12$	2.48×10^{-12}	$(0.02,0.23,0.13)$	0.07	28/26/52	0.192220119252556
			3	2.44×10^{-12}				0.192220119251988
			12	2.27×10^{-12}				0.192220119242490
		$-3.36778+0.23154i$	$4\leftrightarrow 12$	7.44×10^{-15}	$(0.18,0.12,0.32)$	0.15	20/20/38	0.385610022684362
		$-3.36778-0.23154i$	$4\leftrightarrow 14$	1.29×10^{-14}	$(0.31,0.25,0.26)$	0.23	20/20/38	0.385120703236589
3	$(3,2)$	$-0.25460+0.74953i$	$3\leftrightarrow 13$	4.17×10^{-14}	$(0.06,0.43,0.02)$	0.19	22/26/46	0.526530016209690
			3					0.526530016209692
			4	failure.				
		$-0.25460-0.74953i$	$5\leftrightarrow 13$	1.78×10^{-14}	$(0.06,0.42,-0.03)$	0.18	22/26/46	0.526530016209333
		$-2.74540+0.74953i$	$4\leftrightarrow 11$	1.28×10^{-12}	$(0.44,0.13,0.26)$	0.28	28/30/56	0.263454475747302
		$-2.74540-0.74953i$	$5\leftrightarrow 12$	5.70×10^{-13}	$(0.45,0.15,0.25)$	0.28	28/30/56	0.263536300108507
	$(2,3)$	$-0.25460+0.74953i$	$3\leftrightarrow 13$	6.03×10^{-14}	$(-0.02,0.33,0.10)$	0.12	24/28/50	0.526545516107715
		$-0.25460-0.74953i$	$4\leftrightarrow 13$	4.56×10^{-14}	$(0.01,0.34,0.15)$	0.14	22/26/46	0.526530016209657
		$-2.74540+0.74953i$	$5\leftrightarrow 11$	3.95×10^{-12}	$(0.12,-0.05,0.33)$	0.12	28/28/54	0.263272758034494
		$-2.74540-0.74953i$	$3\leftrightarrow 12$	8.45×10^{-13}	$(0.22,0.11,0.36)$	0.19	28/30/56	0.263259271398095
			3	8.64×10^{-13}				0.263259271398280
4	$(3,2)$	$-1.50000+0.99849i$	$3\leftrightarrow 13$	1.19×10^{-14}	$(0.21,0.21,0.20)$	0.13	10/7/16	Free?
		$-1.50000-0.99849i$	$4\leftrightarrow 12$	4.20×10^{-14}	$(0.02,0.24,0.07)$	0.06	10/7/16	Free?
	$(2,3)$	$-1.50000+0.99849i$	$4\leftrightarrow 13$	9.88×10^{-15}	$(0.14,0.10,0.21)$	0.08	10/7/16	Free?
		$-1.50000-0.99849i$	$5\leftrightarrow 13$	3.95×10^{-14}	$(0.10,0.15,0.21)$	0.08	10/7/16	Free?
5	$(3,2)$	$-1.50000+1.07899i$	$3\leftrightarrow 12$	9.10×10^{-15}	$(0.09,0.38,0.12)$	0.17	16/13/28	Free?
		$-1.50000-1.07899i$	$4\leftrightarrow 13$	4.77×10^{-15}	$(0.26,0.41,0.14)$	0.26	12/8/19	Free?
	$(2,3)$	$-1.50000+1.07899i$	$3\leftrightarrow 14$	1.24×10^{-14}	$(0.11,0.32,0.12)$	0.13	16/13/28	Free?
		$-1.50000-1.07899i$	$3\leftrightarrow 14$	1.20×10^{-14}	$(0.11,0.34,0.08)$	0.13	16/13/28	Free?
6	$(3,2)$	$-1.50000+1.16963i$	$3\leftrightarrow 14$	1.69×10^{-14}	$(0.14,0.41,0.12)$	0.20	16/13/28	Free?
		$-1.50000-1.16963i$	$4\leftrightarrow 14$	4.44×10^{-15}	$(0.32,0.44,0.16)$	0.33	12/8/19	Free?
	$(2,3)$	$-1.50000+1.16963i$	$3\leftrightarrow 13$	5.55×10^{-15}	$(0.17,0.35,0.12)$	0.17	16/13/28	Free?
		$-1.50000-1.16963i$	$3\leftrightarrow 12$	1.98×10^{-14}	$(0.17,0.36,0.09)$	0.17	16/13/28	Free?
7	$(3,2)$	$-1.50000+1.21740i$	$4\leftrightarrow 14$	5.88×10^{-15}	$(0.16,0.43,0.12)$	0.23	16/13/28	Free?
		$-1.50000-1.21740i$	$6\leftrightarrow 10$	5.77×10^{-15}	$(0.36,0.46,0.17)$	0.37	12/8/19	Free?
	$(2,3)$	$-1.50000+1.21740i$	$3\leftrightarrow 13$	3.66×10^{-15}	$(0.04,0.20,0.39)$	0.20	12/8/19	Free?
		$-1.50000-1.21740i$	$3\leftrightarrow 13$	1.38×10^{-14}	$(0.20,0.38,0.10)$	0.19	16/13/28	Free?

Tab. C.16: Data from Table 5.5 of [62] Part I.

n	(p,q)	γ	ε_V	deviation	B	$\|\|B\|\|_E^2$	S/V/E	covolume
8	$(3,2)$	$0.47582+0.16460i$	$5\leftrightarrow 11$	1.11×10^{-11}	$(0.32,0.17,0.42)$	0.31	30/28/56	0.220140673239553
		$0.47582-0.16460i$	$4\leftrightarrow 11$	2.99×10^{-12}	$(0.33,0.27,0.36)$	0.31	30/28/56	0.220140544640750
		$-3.47582+0.16460i$	$3\leftrightarrow 11$	4.74×10^{-13}	$(-0.26,0.25,0.19)$	0.17	30/28/56	0.220007282470588
		$-3.47582-0.16460i$	$3\leftrightarrow 11$	1.79×10^{-13}	$(-0.28,0.22,0.23)$	0.18	30/28/56	0.220219588908876
	$(2,3)$	$0.47582+0.16460i$	$3\leftrightarrow 12$	1.07×10^{-11}	$(0.08,0.27,0.21)$	0.12	30/28/56	0.219957247078344
		$0.47582-0.16460i$	$4\leftrightarrow 12$ 3	1.72×10^{-11} 8.78×10^{-12}	$(\ 0.08,\ 0.25,\ 0.16)$	0.09	30/28/56	0.220219589056842 0.220219588953623
		$-3.47582+0.16460i$	$5\leftrightarrow 12$ 3	1.12×10^{-12} 1.13×10^{-12}	$(\ 0.13,\ 0.20,\ 0.36)$	0.19	30/28/56	0.220140673225395 0.220140673225399
		$-3.47582-0.16460i$	$3\leftrightarrow 4$ $5\leftrightarrow 11$	1.03×10^{-12} 1.04×10^{-12}	$(\ 0.15,\ 0.16,\ 0.33)$	0.15	30/28/56	0.220007282472575 0.220007282472604
9	$(3,2)$	$0.23676+0.31257i$	$5\leftrightarrow 12$	8.68×10^{-13}	$(0.16,0.42,0.05)$	0.21	26/24/48	0.154287388430275
		$0.23676-0.31257i$	$4\leftrightarrow 12$	3.78×10^{-12}	$(0.20,0.28,0.28)$	0.19	26/24/48	0.154364842410065
		$-3.23676+0.31257i$	$3\leftrightarrow 11$ 3	2.62×10^{-14} 7.31×10^{-14}	$(-0.39,0.23,0.12)$	0.22	26/24/48	0.154047973303354 0.154047973303744
		$-3.23676-0.31257i$	$5\leftrightarrow 11$	2.40×10^{-14}	$(-0.39,0.15,0.22)$	0.22	26/24/48	0.154318773285096
	$(2,3)$	$0.23676+0.31257i$	$4\leftrightarrow 12$	7.57×10^{-13}	$(-0.06,0.24,0.21)$	0.11	26/24/48	0.153776007581217
		$0.23676-0.31257i$	$3\leftrightarrow 12$ 3	1.17×10^{-12} 1.10×10^{-12}	$(-0.04,0.22,0.10)$	0.06	26/24/48	0.154375705696126 0.154375705689934
		$-3.23676+0.31257i$	$4\leftrightarrow 12$	2.72×10^{-13}	$(0.03,0.20,0.45)$	0.25	26/24/48	0.154287388416193
		$-3.23676-0.31257i$	$3\leftrightarrow 12$	7.77×10^{-14}	$(0.09,0.09,0.41)$	0.19	26/24/48	0.153513110895098
10	$(3,2)$	$-0.07087+0.66435i$	$4\leftrightarrow 13$	7.84×10^{-14}	$(0.07,0.42,-0.07)$	0.19	24/32/54	0.595189968898689
		$-0.07087-0.66435i$	$4\leftrightarrow 12$	3.46×10^{-14}	$(0.07,0.42,0.06)$	0.19	24/32/54	0.595189968897844
		$-2.92913+0.66435i$	$5\leftrightarrow 12$	1.00×10^{-12}	$(0.47,0.14,0.26)$	0.30	22/20/40	0.297303264846344
		$-2.92913-0.66435i$	$5\leftrightarrow 13$	5.37×10^{-14}	$(0.39,0.03,0.41)$	0.32	22/24/44	0.594206188118980
	$(2,3)$	$-0.07087+0.66435i$	$4\leftrightarrow 13$	1.01×10^{-13}	$(-0.13,0.22,0.30)$	0.15	24/32/54	0.595419291729511
		$-0.07087-0.66435i$	$4\leftrightarrow 13$	8.83×10^{-14}	$(-0.03,0.39,0.24)$	0.21	24/32/54	0.595419291729334
		$-2.92913+0.66435i$	$5\leftrightarrow 13$	2.32×10^{-14}	$(-0.02,0.09,0.46)$	0.22	22/24/44	0.594663573037919
		$-2.92913-0.66435i$	$4\leftrightarrow 13$	8.88×10^{-15}	$(0.36,0.24,0.27)$	0.26	22/24/44	0.594566134203106
11	$(3,2)$	$-1.08698+0.98787i$	$7\leftrightarrow 12$ 3 4	1.20×10^{-12} 1.31×10^{-12} 1.20×10^{-12}	$(0.05,0.26,0.15)$	0.09	28/30/56 24/23/45	0.264160360839894 0.264135365269606 0.264043393927288
			$5,6$	failure.				
		$-1.08698-0.98787i$	$4\leftrightarrow 12$	1.21×10^{-12}	$(0.16,0.22,0.21)$	0.12	28/30/56	0.264618617416783
		$-1.91302+0.98787i$	$5\leftrightarrow 13$	4.11×10^{-14}	$(0.11,0.36,0.13)$	0.16	22/24/44	0.529237234824297
		$-1.91302-0.98787i$	$4\leftrightarrow 12$	1.50×10^{-14}	$(0.47,0.21,0.34)$	0.38	22/26/46	0.529237234824279
	$(2,3)$	$-1.08698+0.98787i$	$7\leftrightarrow 11$ 4	1.27×10^{-11}	$(-0.02,0.21,-0.14)$	0.07	28/30/56 24/23/45	0.264308750169820 0.264283754601990
			$5,6$	failure.				
		$-1.08698-0.98787i$	$3\leftrightarrow 13$ 3	1.03×10^{-12}	$(-0.02,0.21,0.14)$	0.07	28/30/56 24/23/45	0.264308750156561 0.264283754576702
			$4,5,6$	failure.				
			13	9.25×10^{-13}	$(-0.02,0.21,0.14)$	0.07	28/30/56	0.264308750156759
		$-1.91302+0.98787i$	$4\leftrightarrow 13$	7.44×10^{-15}	$(0.12,0.10,0.34)$	0.14	20/20/38	0.529035831989798
		$-1.91302-0.98787i$	$4\leftrightarrow 14$	3.42×10^{-14}	$(0.14,0.32,0.06)$	0.13	22/24/44	0.529237234824353

Tab. C.17: Data from Table 5.5 of [62] Part II.

n	(p,q)	γ	ε_V	deviation	B	$\|\|B\|\|_E^2$	S/V/E	covolume
1	$(3,2)$	$0.02764+0.73661i$	$6\leftrightarrow 9$	9.68×10^{-11}	$(0.32,0.55,-0.03)$	0.41	40/58/96	2.606687956763813
		$0.02764-0.73661i$	$5\leftrightarrow 12$	1.40×10^{-10}	$(0.36,0.43,0.33)$	0.43	40/58/96	2.614768903946696
		$-3.02764+0.73661i$	$8\leftrightarrow 9$	3.25×10^{-10}	$(0.48,0.37,0.28)$	0.45	56/80/134	2.626708787863783
		$-3.02764-0.73661i$	$7\leftrightarrow 11$	1.02×10^{-09}	$(-0.44,0.29,0.37)$	0.41	56/80/134	2.620881815776255
	$(2,3)$	$0.02764+0.73661i$	$6\leftrightarrow 12$	6.72×10^{-10}	$(0.14,0.33,0.40)$	0.29	40/58/96	2.615056055121278
		$0.02764-0.73661i$	$5\leftrightarrow 13$ 13	5.17×10^{-10} 5.18×10^{-10}	$(0.25,0.41,0.27)$	0.30	40/58/96	2.618423360997099 2.618423360997209
		$-3.02764+0.73661i$	$6\leftrightarrow 10$	1.56×10^{-09}	$(0.03,0.11,0.54)$	0.30	56/80/134	2.603440024219396
		$-3.02764-0.73661i$	$6\leftrightarrow 10$	2.13×10^{-09}	$(0.45,0.26,0.31)$	0.37	56/80/134	2.624609572288991
2	$(3,2)$	$-0.33909+0.92881i$	$6\leftrightarrow 13$	3.24×10^{-11}	$(0.28,0.53,0.01)$	0.35	40/58/96	1.786629473177491
		$-0.33909-0.92881i$	$5\leftrightarrow 12$	1.20×10^{-11}	$(0.34,0.46,0.25)$	0.39	40/58/96	1.786256005477690
		$-2.66091+0.92881i$	$6\leftrightarrow 12$	4.60×10^{-11}	$(0.48,0.40,0.23)$	0.45	40/58/96	1.781620783377264
		$-2.66091-0.92881i$	$6\leftrightarrow 12$	1.09×10^{-10}	$(0.46,-0.04,0.48)$	0.44	40/58/96	1.791590990342503
	$(2,3)$	$-0.33909+0.92881i$	$7\leftrightarrow 12$	4.73×10^{-11}	$(0.07,0.30,0.43)$	0.28	40/58/96	1.786256005501931
		$-0.33909-0.92881i$	$4\leftrightarrow 13$	3.95×10^{-11}	$(0.24,0.41,0.23)$	0.27	40/58/96	1.786629473184663
		$-2.66091+0.92881i$	$5\leftrightarrow 13$ 5	5.73×10^{-11} 8.65×10^{-11}	$(-0.14,0.11,0.58)$	0.37	40/58/96	1.792459082163101 1.792459082164200
		$-2.66091-0.92881i$	$5\leftrightarrow 12$	7.18×10^{-11}	$(0.42,0.11,0.38)$	0.34	46/70/114	1.787425136044603
3	$(3,2)$	$-1.40631+1.19616i$	$5\leftrightarrow 13$	5.55×10^{-12}	$(0.61,0.48,0.17)$	0.64	34/44/76	1.473170600500694
		$-1.40631-1.19616i$	$5\leftrightarrow 13$	9.48×10^{-12}	$(0.61,0.47,-0.19)$	0.63	34/44/76	1.473194395414834
		$-1.59369+1.19616i$	$5\leftrightarrow 10$	3.68×10^{-12}	$(0.15,0.42,0.12)$	0.22	38/50/86	1.476011079834294
		$-1.59369-1.19616i$	$5\leftrightarrow 14$	4.41×10^{-12}	$(0.38,0.44,0.19)$	0.37	42/54/94	1.474680608065723
	$(2,3)$	$-1.40631+1.19616i$	$4\leftrightarrow 13$	9.58×10^{-13}	$(0.42,0.12,0.46)$	0.41	34/44/76	1.473780096939082
		$-1.40631-1.19616i$	$5\leftrightarrow 10$	3.51×10^{-12}	$(0.55,0.28,0.37)$	0.52	34/44/76	1.473780096936488
		$-1.59369+1.19616i$	$5\leftrightarrow 14$	6.98×10^{-12}	$(0.07,0.19,0.39)$	0.19	42/54/94	1.474327445118291
		$-1.59369-1.19616i$	$5\leftrightarrow 13$	9.81×10^{-12}	$(0.19,0.37,0.09)$	0.18	38/50/86	1.474327445110911
4	$(3,2)$	$-0.56154+1.01758i$	$6\leftrightarrow 13$	3.03×10^{-11}	$(0.18,0.48,0.05)$	0.26	38/50/86	1.886983797195308
		$-0.56154-1.01758i$	$5\leftrightarrow 13$	4.11×10^{-11}	$(0.34,0.43,0.28)$	0.38	40/54/92	1.912190055689327
		$-2.43846+1.01758i$	$7\leftrightarrow 13$	8.73×10^{-11}	$(0.47,0.41,0.21)$	0.44	58/94/150	1.918222504276216
		$-2.43846-1.01758i$	$6\leftrightarrow 13$	1.52×10^{-10}	$(0.44,-0.06,0.48)$	0.43	58/94/150	1.919398019822496
	$(2,3)$	$-0.56154+1.01758i$	$4\leftrightarrow 13$	6.15×10^{-11}	$(0.10,0.26,0.38)$	0.22	40/54/92	1.916994266198379
		$-0.56154-1.01758i$	$6\leftrightarrow 13$	3.17×10^{-11}	$(0.15,0.38,0.18)$	0.20	38/50/86	1.906273264858948
		$-2.43846+1.01758i$	$6\leftrightarrow 12$	6.13×10^{-11}	$(0.45,0.28,-0.30)$	0.37	58/94/150	1.910046253239740
		$-2.43846-1.01758i$	$6\leftrightarrow 11$	4.99×10^{-11}	$(0.45,0.30,0.28)$	0.38	58/94/150	1.911148458736251
5	$(3,2)$	$-0.42847+1.00664i$	$5\leftrightarrow 12$	1.55×10^{-09}	$(0.25,0.52,0.03)$	0.33	46/60/104	2.457689880923794
		$-0.42847-1.00664i$	$5\leftrightarrow 12$	8.84×10^{-10}	$(0.38,0.44,0.30)$	0.43	44/56/98	2.446921256901292
		$-2.57153+1.00664i$	$6\leftrightarrow 12$	2.18×10^{-09}	$(0.49,0.44,0.22)$	0.48	52/86/136	2.446523234407060
		$-2.57153-1.00664i$	$6\leftrightarrow 12$	9.06×10^{-10}	$(0.46,-0.07,0.50)$	0.46	52/86/136	2.451739445687724
	$(2,3)$	$-0.42847+1.00664i$	$4\leftrightarrow 12$ 4	1.17×10^{-09} 1.16×10^{-09}	$(0.13,0.28,0.40)$	0.26	44/56/98	2.440184222920381 2.440184222921835
		$-0.42847-1.00664i$	$5\leftrightarrow 12$	3.93×10^{-10}	$(0.22,0.40,0.21)$	0.26	46/60/104	2.457416519588215
		$-2.57153+1.00664i$	$6\leftrightarrow 11$	1.41×10^{-09}	$(-0.24,0.07,0.60)$	0.42	52/86/136	2.447761959340231
		$-2.57153-1.00664i$	$6\leftrightarrow 12$	1.56×10^{-09}	$(0.48,0.32,0.29)$	0.42	52/86/136	2.455684611354928
6	$(3,2)$	$-0.87763+1.11400i$	$8\leftrightarrow 11$	3.81×10^{-08}	$(0.58,0.48,0.14)$	0.59	62/104/164	3.403866068741425
		$-0.87763-1.11400i$	$3\leftrightarrow 15$	failure.				
		$-2.12237+1.11400i$	$3\leftrightarrow 15$	failure.				
		$-2.12237-1.11400i$	$8\leftrightarrow 12$	6.35×10^{-08}	$(0.72,0.27,0.40)$	0.75	72/102/172	3.440750479073212
	$(2,3)$	$-0.87763+1.11400i$	$3\leftrightarrow 15$	failure.				
		$-0.87763-1.11400i$	$7\leftrightarrow 11$	6.43×10^{-08}	$(0.51,0.28,0.36)$	0.46	62/104/164	3.433832791680822
		$-2.12237+1.11400i$	$3\leftrightarrow 15$	failure.				
		$-2.12237-1.11400i$	$7\leftrightarrow 12$	3.34×10^{-08}	$(0.54,-0.13,0.63)$	0.70	74/104/176	3.438382428925763
7	$(3,2)$	$-1.50000+1.30625i$	$4\leftrightarrow 13$	5.45×10^{-15}	$(0.21,0.46,0.12)$	0.27	16/13/28	Free?
		$-1.50000-1.30625i$	$4\leftrightarrow 13$	1.09×10^{-14}	$(0.21,0.46,-0.13)$	0.27	16/13/28	Free?
	$(2,3)$	$-1.50000+1.30625i$	$3\leftrightarrow 14$	4.00×10^{-15}	$(0.09,0.23,0.42)$	0.24	12/8/19	Free?
		$-1.50000-1.30625i$	$3\leftrightarrow 14$	1.87×10^{-14}	$(0.26,0.40,0.11)$	0.24	16/13/28	Free?
8	$(3,2)$	$0.16169\pm0.63671i$	$3\leftrightarrow 15$	failure.				
		$-3.16169\pm0.63671i$	$3\leftrightarrow 15$	failure.				
	$(2,3)$	$0.16169\pm0.63671i$	$3\leftrightarrow 15$	failure.				
		$-3.16169\pm0.63671i$	$3\leftrightarrow 15$	failure.				

Tab. C.18: Data from Table 5.6 of [62] Part I.

n	(p,q)	γ	ε_V	deviation	B	$\|\|B\|\|_E^2$	S/V/E	covolume
9	$(3,2)$	$-0.02256 + 0.77896i$	$6 \leftrightarrow 12$	8.18×10^{-10}	$(0.24, 0.52, -0.01)$	0.33	48/70/116	2.904334127793920
		$-0.02256 - 0.77896i$	$6 \leftrightarrow 12$	2.49×10^{-09}	$(0.35, 0.42, 0.34)$	0.42	46/68/112	2.879763283402163
		$-2.97744 \pm 0.77896i$	$3 \leftrightarrow 15$			failure.		
	$(2,3)$	$-0.02256 + 0.77896i$	$5 \leftrightarrow 11$	6.58×10^{-09}	$(0.14, 0.31, 0.38)$	0.26	46/68/112	2.869851292152975
		$-0.02256 - 0.77896i$	$6 \leftrightarrow 12$	6.17×10^{-09}	$(0.18, 0.40, 0.24)$	0.24	48/70/116	2.885077608625386
		$-2.97744 + 0.77896i$	$7 \leftrightarrow 10$	8.22×10^{-10}	$(0.01, 0.12, 0.57)$	0.34	52/70/120	2.903307418976409
		$-2.97744 - 0.77896i$	$6 \leftrightarrow 12$	1.28×10^{-08}	$(0.47, 0.25, 0.33)$	0.40	52/70/120	2.885498461760693
10	$(3,2)$	$-0.51152 + 1.00187i$	$6 \leftrightarrow 13$	3.08×10^{-12}	$(0.19, 0.48, 0.04)$	0.27	32/36/66	1.539578485870570
		$-0.51152 - 1.00187i$	$4 \leftrightarrow 13$	5.56×10^{-13}	$(0.33, 0.43, 0.28)$	0.37	32/36/66	1.539039589792024
		$-2.48848 + 1.00187i$	$4 \leftrightarrow 9$	2.29×10^{-11}	$(0.33, 0.17, 0.28)$	0.21	38/38/74	0.770435534681183
		$-2.48848 - 1.00187i$	$7 \leftrightarrow 10$	1.33×10^{-10}	$(0.33, 0.14, 0.29)$	0.21	38/38/74	0.772699345216488
	$(2,3)$	$-0.51152 + 1.00187i$	$3 \leftrightarrow 13$	3.55×10^{-12}	$(0.09, 0.26, 0.38)$	0.22	32/36/66	1.538156498680045
		$-0.51152 - 1.00187i$	$5 \leftrightarrow 13$	3.82×10^{-12}	$(0.15, 0.38, 0.18)$	0.20	32/36/66	1.541135159292971
		$-2.48848 + 1.00187i$	$5 \leftrightarrow 9$	1.88×10^{-08}	$(0.21, 0.14, -0.25)$	0.12	38/38/74	0.771200674029689
		$-2.48848 - 1.00187i$	$5 \leftrightarrow 9$	3.44×10^{-09}	$(0.21, 0.14, 0.25)$	0.12	38/38/74	0.771200667171090
11	$(3,2)$	$-1.50000 + 1.27647i$	$4 \leftrightarrow 14$	1.64×10^{-14}	$(0.19, 0.45, 0.12)$	0.26	16/13/28	Free?
		$-1.50000 - 1.27647i$	$4 \leftrightarrow 13$	2.20×10^{-14}	$(0.19, 0.45, -0.13)$	0.26	16/13/28	Free?
	$(2,3)$	$-1.50000 + 1.27647i$	$3 \leftrightarrow 13$	3.22×10^{-15}	$(0.07, 0.22, 0.41)$	0.22	12/8/19	Free?
		$-1.50000 - 1.27647i$	$3 \leftrightarrow 14$	7.43×10^{-15}	$(0.24, 0.39, 0.11)$	0.22	16/13/28	Free?

Tab. C.19: Data from Table 5.6 of [62] Part II.

n	(p,q)	γ	ε_V	deviation	B	$\|\|B\|\|_E^2$	S/V/E	covolume
1	$(3,2)$	$0.27341 \pm 0.56382i$	$3 \leftrightarrow 14$			failure.		
	$(3,2)$	$-3.27341 \pm 0.56382i$	$3 \leftrightarrow 15$			failure.		
	$(2,3)$	$0.27341 \pm 0.56382i$	$3 \leftrightarrow 15$			failure.		
	$(2,3)$	$-3.27341 \pm 0.56382i$	$3 \leftrightarrow 15$			failure.		
2	$(3,2)$	$-1.06356 \pm 1.16438i$	$3 \leftrightarrow 15$			failure.		
	$(3,2)$	$-1.93644 \pm 1.16438i$	$3 \leftrightarrow 15$			failure.		
	$(2,3)$	$-1.06356 \pm 1.16438i$	$3 \leftrightarrow 15$			failure.		
	$(2,3)$	$-1.93644 \pm 1.16438i$	$3 \leftrightarrow 15$			failure.		
3	$(3,2)$	$-0.82239 \pm 1.11253i$	$3 \leftrightarrow 15$			failure.		
	$(3,2)$	$-2.17761 \pm 1.11253i$	$3 \leftrightarrow 15$			failure.		
	$(2,3)$	$-0.82239 \pm 1.11253i$	$3 \leftrightarrow 15$			failure.		
	$(2,3)$	$-2.17761 \pm 1.11253i$	$3 \leftrightarrow 15$			failure.		
4	$(3,2)$	$0.68152 \pm 0.14729i$	$3 \leftrightarrow 15$			failure.		
	$(3,2)$	$-3.68152 \pm 0.14729i$	$3 \leftrightarrow 15$			failure.		
	$(2,3)$	$0.68152 \pm 0.14729i$	$3 \leftrightarrow 15$			failure.		
	$(2,3)$	$-3.68152 \pm 0.14729i$	$3 \leftrightarrow 15$			failure.		
5	$(3,2)$	$0.57186 \pm 0.25235i$	$3 \leftrightarrow 15$			failure.		
	$(3,2)$	$-3.57186 \pm 0.25235i$	$3 \leftrightarrow 15$			failure.		
	$(2,3)$	$0.57186 \pm 0.25235i$	$3 \leftrightarrow 15$			failure.		
	$(2,3)$	$-3.57186 \pm 0.25235i$	$3 \leftrightarrow 15$			failure.		
6	$(3,2)$	$0.61068 + 0.18781i$	$5 \leftrightarrow 11$	4.00×10^{-11}	$(0.33, 0.53, -0.18)$	0.42	42/64/104	1.968366319874948
	$(3,2)$	$0.61068 - 0.18781i$	$5 \leftrightarrow 11$	6.12×10^{-11}	$(0.31, 0.38, 0.42)$	0.41	42/64/104	1.968161596960865
	$(3,2)$	$-3.61068 + 0.18781i$	$5 \leftrightarrow 9$	4.92×10^{-11}	$(-0.32, 0.17, 0.51)$	0.38	42/64/104	1.988929607692590
	$(3,2)$	$-3.61068 - 0.18781i$	$5 \leftrightarrow 9$	1.95×10^{-11}	$(-0.37, 0.46, 0.29)$	0.44	42/64/104	1.959655953058214
	$(2,3)$	$0.61068 + 0.18781i$	$4 \leftrightarrow 11$	7.05×10^{-11}	$(0.15, 0.37, 0.35)$	0.28	42/64/104	1.964645380021100
	$(2,3)$		6			failure.		
	$(2,3)$	$0.61068 - 0.18781i$	$4 \leftrightarrow 10$	1.87×10^{-10}	$(0.17, 0.44, 0.39)$	0.38	42/64/104	1.964079146075944
	$(2,3)$	$-3.61068 + 0.18781i$	$6 \leftrightarrow 11$	3.55×10^{-11}	$(0.33, 0.14, 0.45)$	0.33	42/64/104	1.967925349483594
	$(2,3)$	$-3.61068 - 0.18781i$	$5 \leftrightarrow 11$	1.04×10^{-09}	$(0.48, 0.30, 0.39)$	0.47	42/64/104	1.969474111261852
7	$(3,2)$	$0.53101 \pm 0.27349i$	$3 \leftrightarrow 15$			failure.		
	$(3,2)$	$-3.53101 \pm 0.27349i$	$3 \leftrightarrow 15$			failure.		
	$(2,3)$	$0.53101 \pm 0.27349i$	$3 \leftrightarrow 15$			failure.		
	$(2,3)$	$-3.53101 \pm 0.27349i$	$3 \leftrightarrow 15$			failure.		
8	$(3,2)$	$-0.17611 \pm 0.89163i$	$3 \leftrightarrow 15$			failure.		
	$(3,2)$	$-2.82389 \pm 0.89163i$	$3 \leftrightarrow 15$			failure.		
	$(2,3)$	$-0.17611 \pm 0.89163i$	$3 \leftrightarrow 15$			failure.		
	$(2,3)$	$-2.82389 \pm 0.89163i$	$3 \leftrightarrow 15$			failure.		
9	$(3,2)$	$-1.23137 \pm 1.18118i$	$3 \leftrightarrow 15$			failure.		
	$(3,2)$	$-1.76863 \pm 1.18118i$	$3 \leftrightarrow 15$			failure.		
	$(2,3)$	$-1.23137 \pm 1.18118i$	$3 \leftrightarrow 15$			failure.		
	$(2,3)$	$-1.76863 \pm 1.18118i$	$3 \leftrightarrow 15$			failure.		
10	$(3,2)$	$-0.94344 \pm 1.14115i$	$3 \leftrightarrow 15$			failure.		
	$(3,2)$	$-2.05656 \pm 1.14115i$	$3 \leftrightarrow 15$			failure.		
	$(2,3)$	$-0.94344 \pm 1.14115i$	$3 \leftrightarrow 15$			failure.		
	$(2,3)$	$-2.05656 \pm 1.14115i$	$3 \leftrightarrow 15$			failure.		
11	$(3,2)$	$0.59727 \pm 0.22323i$	$3 \leftrightarrow 15$			failure.		
	$(3,2)$	$-3.59727 \pm 0.22323i$	$3 \leftrightarrow 15$			failure.		
	$(2,3)$	$0.59727 \pm 0.22323i$	$3 \leftrightarrow 15$			failure.		
	$(2,3)$	$-3.59727 \pm 0.22323i$	$3 \leftrightarrow 15$			failure.		
12	$(3,2)$	$0.47623 \pm 0.33589i$	$3 \leftrightarrow 15$			failure.		
	$(3,2)$	$-3.47623 \pm 0.33589i$	$3 \leftrightarrow 15$			failure.		
	$(2,3)$	$0.47623 \pm 0.33589i$	$3 \leftrightarrow 15$			failure.		
	$(2,3)$	$-3.47623 \pm 0.33589i$	$3 \leftrightarrow 15$			failure.		
13	$(3,2)$	$0.29516 \pm 0.53913i$	$3 \leftrightarrow 15$			failure.		
	$(3,2)$	$-3.29516 \pm 0.53913i$	$3 \leftrightarrow 15$			failure.		
	$(2,3)$	$0.29516 \pm 0.53913i$	$3 \leftrightarrow 15$			failure.		
	$(2,3)$	$-3.29516 \pm 0.53913i$	$3 \leftrightarrow 15$			failure.		
14	$(3,2)$	$0.32580 \pm 0.51138i$	$3 \leftrightarrow 15$			failure.		
	$(3,2)$	$-3.32580 \pm 0.51138i$	$3 \leftrightarrow 15$			failure.		
	$(2,3)$	$0.32580 \pm 0.51138i$	$3 \leftrightarrow 15$			failure.		
	$(2,3)$	$-3.32580 \pm 0.51138i$	$3 \leftrightarrow 15$			failure.		

Tab. C.20: Data from Table 5.7 of [62].

n	(p,q)	γ	ε_V	deviation	B	$\|\|B\|\|_E^2$	S/V/E	covolume
1	$(4,2)$	$-0.50000+0.86603i$	$4\leftrightarrow 13$	1.29×10^{-14}	$(0.07,0.42,-0.03)$	0.18	20/22/40	0.253735401602386
		$-0.50000-0.86603i$	$4\leftrightarrow 13$	9.44×10^{-15}	$(0.08,0.42,0.05)$	0.18	20/22/40	0.253598511561355
		$-1.50000+0.86603i$	$3\leftrightarrow 13$	4.55×10^{-15}	$(0.18,0.41,0.04)$	0.20	18/22/38	0.253041353515199
		$-1.50000-0.86603i$	$4\leftrightarrow 13$	5.55×10^{-15}	$(0.38,0.31,0.25)$	0.31	18/22/38	0.252720823519394
	$(2,4)$	$-0.50000+0.86603i$	$5\leftrightarrow 13$	7.66×10^{-15}	$(-0.08,0.28,0.07)$	0.09	20/22/40	0.253542824010740
		$-0.50000-0.86603i$	$3\leftrightarrow 13$	2.00×10^{-14}	$(-0.07,0.30,0.05)$	0.10	20/22/40	0.253338508765728
		$-1.50000+0.86603i$	$4\leftrightarrow 14$	6.22×10^{-15}	$(0.08,0.30,-0.00)$	0.10	18/22/38	0.253712610730981
		$-1.50000-0.86603i$	$3\leftrightarrow 14$	3.33×10^{-15}	$(0.02,0.29,0.07)$	0.09	18/22/38	0.253384811296644
			3	4.66×10^{-15}				0.253384811296590
2	$(4,2)$	$0.23279+0.79255i$	$4\leftrightarrow 11$	5.50×10^{-14}	$(0.13,0.31,0.38)$	0.26	22/28/48	0.593639175240615
		$0.23279-0.79255i$	$5\leftrightarrow 12$	1.88×10^{-14}	$(0.17,0.50,0.13)$	0.29	22/26/46	0.593261429781721
		$-2.23279+0.79255i$	$5\leftrightarrow 13$	1.10×10^{-14}	$(-0.26,0.28,0.49)$	0.39	22/28/48	0.595004149184967
		$-2.23279-0.79255i$	$5\leftrightarrow 14$	2.18×10^{-14}	$(-0.27,0.47,0.31)$	0.39	22/28/48	0.594063100273875
	$(2,4)$	$0.23279+0.79255i$	$3\leftrightarrow 14$	4.24×10^{-14}	$(0.02,0.34,0.12)$	0.13	22/28/48	0.592869862079440
		$0.23279-0.79255i$	$4\leftrightarrow 13$	4.69×10^{-14}	$(0.03,0.36,0.11)$	0.14	22/26/46	0.593261429782327
		$-2.23279+0.79255i$	$4\leftrightarrow 14$	1.75×10^{-14}	$(0.33,0.34,-0.22)$	0.27	22/28/48	0.593689839020190
		$-2.23279-0.79255i$	$4\leftrightarrow 14$	1.71×10^{-14}	$(0.33,0.35,0.21)$	0.28	22/28/48	0.593689839020260
3	$(4,2)$	$-1.34116+1.16154i$	$3\leftrightarrow 12$	4.41×10^{-14}	$(0.25,0.51,0.02)$	0.33	24/26/48	0.593035916591825
			3	7.31×10^{-14}				0.593035916591793
		$-1.34116-1.16154i$	$4\leftrightarrow 13$	3.13×10^{-14}	$(0.25,0.51,-0.03)$	0.33	24/26/48	0.593035916591978
		$-0.65884+1.16154i$	$4\leftrightarrow 13$	6.66×10^{-15}	$(0.36,0.56,-0.09)$	0.46	22/28/48	0.593932061794999
		$-0.65884-1.16154i$	$3\leftrightarrow 14$	1.13×10^{-14}	$(0.42,0.49,0.24)$	0.47	22/28/48	0.594047553937928
	$(2,4)$	$-1.34116+1.16154i$	$3\leftrightarrow 14$	6.28×10^{-14}	$(0.19,0.37,0.04)$	0.18	24/26/48	0.593687497734524
			3	6.08×10^{-14}				0.593687497734493
		$-1.34116-1.16154i$	$4\leftrightarrow 14$	5.42×10^{-14}	$(0.13,0.37,0.08)$	0.16	24/26/48	0.592911657244731
		$-0.65884+1.16154i$	$4\leftrightarrow 14$	1.12×10^{-14}	$(0.19,0.38,-0.17)$	0.21	22/28/48	0.593688668377331
		$-0.65884-1.16154i$	$3\leftrightarrow 14$	1.44×10^{-14}	$(0.17,0.38,0.17)$	0.20	22/28/48	0.593687497734380
4	$(4,2)$	$0.66236+0.56228i$	$5\leftrightarrow 13$	1.10×10^{-13}	$(0.18,0.40,0.37)$	0.32	22/30/50	0.825295376298228
		$0.66236-0.56228i$	$4\leftrightarrow 13$	3.89×10^{-14}	$(0.20,0.52,0.20)$	0.35	22/30/50	0.824578962449851
		$-2.66236+0.56228i$	$4\leftrightarrow 12$	3.20×10^{-14}	$(-0.23,0.47,0.26)$	0.34	24/28/50	0.824071575741213
		$-2.66236-0.56228i$	$4\leftrightarrow 14$	7.39×10^{-14}	$(-0.26,0.59,0.15)$	0.43	24/30/52	0.824403065414145
	$(2,4)$	$0.66236+0.56228i$	$4\leftrightarrow 12$	1.07×10^{-13}	$(0.07,0.37,0.14)$	0.16	22/30/50	0.825510669974842
		$0.66236-0.56228i$	$4\leftrightarrow 13$	4.17×10^{-14}	$(0.07,0.38,0.13)$	0.17	22/30/50	0.824521209676969
		$-2.66236+0.56228i$	$4\leftrightarrow 13$	8.84×10^{-14}	$(-0.20,0.06,0.43)$	0.23	24/30/52	0.823496863377182
		$-2.66236-0.56228i$	$4\leftrightarrow 14$	8.55×10^{-14}	$(0.42,0.58,0.35)$	0.64	24/30/52	0.823979602632310
5	$(4,2)$	$0.50000+0.86603i$	$4\leftrightarrow 13$	5.80×10^{-13}	$(0.61,0.47,0.46)$	0.80	26/34/58	1.523565545131137
		$0.50000-0.86603i$	$4\leftrightarrow 11$	2.78×10^{-13}	$(0.61,0.39,0.51)$	0.78	26/34/58	1.523908885312456
		$-2.50000+0.86603i$	$5\leftrightarrow 11$	4.76×10^{-13}	$(-0.39,0.21,0.64)$	0.61	28/34/60	1.523293304604914
		$-2.50000-0.86603i$	$4\leftrightarrow 12$	4.40×10^{-14}	$(-0.41,0.63,0.27)$	0.64	28/34/60	1.524595659895754
	$(2,4)$	$0.50000+0.86603i$	$4\leftrightarrow 13$	8.73×10^{-13}	$(0.18,0.31,0.42)$	0.31	26/34/58	1.522827749800601
		$0.50000-0.86603i$	$4\leftrightarrow 12$	1.13×10^{-12}	$(0.16,0.39,0.26)$	0.25	34/46/78	1.523846644121271
		$-2.50000+0.86603i$	$5\leftrightarrow 13$	8.32×10^{-13}	$(-0.25,0.06,0.58)$	0.40	28/34/60	1.522412409614590
		$-2.50000-0.86603i$	$4\leftrightarrow 13$	9.14×10^{-13}	$(0.55,0.52,0.41)$	0.75	28/34/60	1.524178139252510
6	$(4,2)$	$-0.21508+1.30714i$	$6\leftrightarrow 11$	9.94×10^{-13}	$(0.51,0.61,-0.16)$	0.66	44/58/100	2.122090588879277
		$-0.21508-1.30714i$	$6\leftrightarrow 10$	7.00×10^{-12}	$(0.52,0.47,0.37)$	0.64	38/48/84	2.119372694824350
		$-1.78492+1.30714i$	$6\leftrightarrow 11$	1.52×10^{-12}	$(-0.36,0.37,0.51)$	0.53	42/62/102	2.125201114257589
		$-1.78492-1.30714i$	$5\leftrightarrow 12$	6.96×10^{-12}	$(-0.35,0.49,0.42)$	0.54	42/62/102	2.124077667438180
	$(2,4)$	$-0.21508+1.30714i$	$5\leftrightarrow 13$	1.10×10^{-11}	$(0.34,0.43,0.18)$	0.33	38/48/84	2.119305381101686
		$-0.21508-1.30714i$	$5\leftrightarrow 14$	6.60×10^{-12}	$(0.31,0.43,0.23)$	0.33	44/58/100	2.124295589694652
		$-1.78492+1.30714i$	$6\leftrightarrow 12$	2.87×10^{-11}	$(0.40,0.42,-0.17)$	0.37	42/62/102	2.120955568937968
		$-1.78492-1.30714i$	$5\leftrightarrow 12$	8.96×10^{-12}	$(0.47,0.18,0.42)$	0.43	38/52/88	2.122102670376362

Tab. C.21: Data from Table 5.8 of [62] Part I.

n	(p,q)	γ	ε_V	deviation	B	$\|\|B\|\|_E^2$	S/V/E	covolume
7	$(4,2)$	$1.06115+0.38830i$	$4\leftrightarrow 13$	1.74×10^{-13}	$(0.57,0.52,0.40)$	0.75	26/38/62	1.545465473698826
		$1.06115-0.38830i$	$5\leftrightarrow 12$	1.64×10^{-13}	$(0.39,0.56,0.28)$	0.54	30/46/74	1.542632159257580
			12	4.87×10^{-12}				1.542632159259560
		$-3.06115+0.38830i$	$4\leftrightarrow 10$	6.96×10^{-13}	$(-0.24,0.62,0.23)$	0.49	28/30/56	1.542891480916546
		$-3.06115-0.38830i$	$5\leftrightarrow 11$	5.14×10^{-13}	$(-0.25,0.69,0.04)$	0.54	28/32/58	1.542645122925050
	$(2,4)$	$1.06115+0.38830i$	$3\leftrightarrow 13$	5.15×10^{-13}	$(0.18,0.35,0.39)$	0.31	26/38/62	1.542645122928808
		$1.06115-0.38830i$	$3\leftrightarrow 13$	6.87×10^{-13}	$(0.06,0.37,0.30)$	0.23	30/46/74	1.542854953734646
		$-3.06115+0.38830i$	$5\leftrightarrow 13$	9.10×10^{-13}	$(0.24,0.30,0.32)$	0.25	38/48/84	1.542004392351739
		$-3.06115-0.38830i$	$5\leftrightarrow 12$	4.41×10^{-12}	$(0.36,0.34,0.27)$	0.32	38/48/84	1.542145896625610
8	$(4,2)$	$0.03640+1.21238i$	$5\leftrightarrow 11$	1.85×10^{-13}	$(0.51,0.60,-0.18)$	0.64	30/36/64	1.540788174525126
		$0.03640-1.21238i$	$5\leftrightarrow 10$	3.63×10^{-13}	$(0.66,0.48,0.53)$	0.94	30/38/66	1.542645122913982
		$-2.03640+1.21238i$	$4\leftrightarrow 12$	1.90×10^{-12}	$(0.61,0.65,0.03)$	0.80	26/38/62	1.542302706507817
		$-2.03640-1.21238i$	$5\leftrightarrow 13$	4.91×10^{-13}	$(-0.42,0.55,0.42)$	0.65	26/38/62	1.545386182393748
	$(2,4)$	$0.03640+1.21238i$	$4\leftrightarrow 12$	5.57×10^{-13}	$(0.24,0.35,0.38)$	0.33	30/38/66	1.542645122929170
		$0.03640-1.21238i$	$4\leftrightarrow 12$	1.02×10^{-11}	$(0.28,0.42,0.24)$	0.32	30/36/64	1.539457592499659
			13	7.33×10^{-13}				1.539457592494465
		$-2.03640+1.21238i$	$6\leftrightarrow 13$	2.11×10^{-13}	$(-0.30,0.12,0.55)$	0.41	26/38/62	1.542645122925198
		$-2.03640-1.21238i$	$4\leftrightarrow 13$	8.96×10^{-13}	$(0.52,0.42,0.25)$	0.51	26/38/62	1.545720979684452
9	$(4,2)$	$0.65139+0.75874i$	$4\leftrightarrow 12$	8.65×10^{-14}	$(0.64,0.38,0.53)$	0.83	30/40/68	1.624021910597929
			4	9.13×10^{-14}				1.624021910598127
			5,6	failure.				
		$0.65139-0.75874i$	$5\leftrightarrow 11$	4.40×10^{-13}	$(0.60,0.60,0.25)$	0.78	30/40/68	1.623182497757532
			5	4.40×10^{-13}				1.623182497757539
			6,7,8	failure.				
		$-2.65139+0.75874i$	$4\leftrightarrow 13$	4.95×10^{-13}	$(-0.33,0.15,0.66)$	0.57	30/38/66	1.626212883740653
		$-2.65139-0.75874i$	$5\leftrightarrow 12$	1.33×10^{-13}	$(-0.35,0.66,0.21)$	0.61	30/38/66	1.625477965973462
	$(2,4)$	$0.65139+0.75874i$	$4\leftrightarrow 13$	5.39×10^{-13}	$(0.15,0.39,0.53)$	0.46	30/40/68	1.622813060220924
		$0.65139-0.75874i$	$4\leftrightarrow 12$	2.16×10^{-12}	$(0.31,0.42,0.32)$	0.37	30/40/68	1.623508874386314
		$-2.65139+0.75874i$	$5\leftrightarrow 13$	9.57×10^{-13}	$(-0.22,0.05,0.54)$	0.34	30/38/66	1.625086429747067
		$-2.65139-0.75874i$	$4\leftrightarrow 13$	1.65×10^{-12}	$(0.55,0.59,0.41)$	0.82	30/38/66	1.626631271664573
			4	1.07×10^{-12}				1.626631271663298
10	$(4,2)$	$1.36778+0.23154i$	$5\leftrightarrow 8$	1.92×10^{-10}	$(0.59,0.55,0.39)$	0.80	44/72/114	2.896525584118602
		$1.36778-0.23154i$	$6\leftrightarrow 9$	2.95×10^{-11}	$(0.69,0.58,0.36)$	0.94	38/60/96	2.892098842548785
		$-3.36778+0.23154i$	$6\leftrightarrow 9$	3.34×10^{-10}	$(-0.24,0.74,0.16)$	0.63	46/60/104	2.892265006742646
		$-3.36778-0.23154i$	$5\leftrightarrow 8$	3.76×10^{-10}	$(-0.21,0.73,0.15)$	0.60	44/58/100	2.895735473733071
	$(2,4)$	$1.36778+0.23154i$	$5\leftrightarrow 12$	1.99×10^{-11}	$(0.19,0.37,0.39)$	0.33	44/72/114	2.900382924576496
		$1.36778-0.23154i$	$3\leftrightarrow 12$	4.40×10^{-11}	$(0.36,0.42,0.40)$	0.47	38/60/96	2.889967487870522
		$-3.36778+0.23154i$	$4\leftrightarrow 12$	1.72×10^{-10}	$(0.36,0.35,0.30)$	0.34	54/82/134	2.896693414681468
			4	1.72×10^{-10}				2.896693414681705
			5	failure.				
		$-3.36778-0.23154i$	$5\leftrightarrow 12$	2.92×10^{-10}	$(0.41,0.37,0.27)$	0.38	54/82/134	2.895921415505758
11	$(4,2)$	$-0.57943+1.45743i$	$6\leftrightarrow 12$	2.52×10^{-11}	$(0.58,0.64,-0.14)$	0.76	48/62/108	2.889771482503296
			12	3.19×10^{-11}				2.889771482506005
		$-0.57943-1.45743i$	$6\leftrightarrow 12$	6.07×10^{-11}	$(0.71,0.57,0.33)$	0.94	46/58/102	2.883055816118886
		$-1.42057+1.45743i$	$5\leftrightarrow 12$	5.87×10^{-10}	$(0.42,0.61,-0.02)$	0.55	40/64/102	2.888631582046883
		$-1.42057-1.45743i$	$5\leftrightarrow 9$	2.11×10^{-10}	$(0.42,0.61,-0.00)$	0.55	40/64/102	2.888357706163286
	$(2,4)$	$-0.57943+1.45743i$	$6\leftrightarrow 11$	8.88×10^{-11}	$(0.09,0.30,0.56)$	0.41	46/58/102	2.891037328603279
		$-0.57943-1.45743i$	$5\leftrightarrow 11$	4.67×10^{-10}	$(0.48,0.46,0.26)$	0.50	46/58/102	2.887948649383576
		$-1.42057+1.45743i$	$4\leftrightarrow 12$	9.45×10^{-11}	$(0.34,0.43,-0.14)$	0.32	40/64/102	2.891340099023110
		$-1.42057-1.45743i$	$5\leftrightarrow 12$	2.91×10^{-11}	$(0.34,0.44,0.13)$	0.32	40/64/102	2.891141431027691

Tab. C.22: Data from Table 5.8 of [62] Part II.

n	(p,q)	γ	ε_V	deviation	B	$\|\|B\|\|_E^2$	S/V/E	covolume
1	$(4,2)$	$-0.22816+1.11514i$	$4\leftrightarrow 12$	2.04×10^{-14}	$(0.29,0.53,-0.10)$	0.38	24/32/54	0.793730924087971
		$-0.22816-1.11514i$	$4\leftrightarrow 12$	2.44×10^{-14}	$(0.35,0.44,0.31)$	0.41	22/26/46	0.793775329716258
		$-1.77184+1.11514i$	$4\leftrightarrow 12$	1.26×10^{-13}	$(0.39,0.56,0.03)$	0.46	22/30/50	0.794323319649266
		$-1.77184-1.11514i$	$4\leftrightarrow 11$	1.08×10^{-13}	$(-0.34,0.45,0.34)$	0.44	22/30/50	0.794323319649402
	$(2,4)$	$-0.22816+1.11514i$	$3\leftrightarrow 14$	1.86×10^{-14}	$(0.11,0.33,0.23)$	0.18	22/26/46	0.793839731308609
		$-0.22816-1.11514i$	$4\leftrightarrow 14$	4.95×10^{-14}	$(0.08,0.37,0.17)$	0.17	24/32/54	0.794272642251702
		$-1.77184+1.11514i$	$4\leftrightarrow 14$	7.48×10^{-14}	$(0.30,0.38,-0.15)$	0.26	22/30/50	0.794323319650373
		$-1.77184-1.11514i$	$4\leftrightarrow 13$	5.28×10^{-14}	$(0.30,0.39,0.14)$	0.26	22/30/50	0.794323319650378
2	$(4,2)$	$0.20711+0.97832i$	$4\leftrightarrow 12$	3.80×10^{-14}	$(0.46,0.57,-0.20)$	0.58	24/30/52	1.028680367759286
		$0.20711-0.97832i$	$4\leftrightarrow 13$	1.80×10^{-14}	$(0.51,0.53,0.28)$	0.62	24/30/52	1.028760806843382
		$-2.20711+0.97832i$	$4\leftrightarrow 12$	8.73×10^{-14}	$(-0.30,0.32,0.54)$	0.49	24/30/52	1.028760806843202
		$-2.20711-0.97832i$	$4\leftrightarrow 12$	1.64×10^{-13}	$(-0.32,0.52,0.36)$	0.50	24/30/52	1.028504497181299
	$(2,4)$	$0.20711+0.97832i$	$4\leftrightarrow 13$	5.44×10^{-14}	$(-0.05,0.33,0.54)$	0.40	24/30/52	1.028504497182124
		$0.20711-0.97832i$	$4\leftrightarrow 13$	4.50×10^{-14}	$(0.21,0.40,0.25)$	0.27	24/30/52	1.028697470061227
		$-2.20711+0.97832i$	$5\leftrightarrow 13$	1.23×10^{-13}	$(-0.20,0.09,0.47)$	0.27	24/30/52	1.028810003857120
		$-2.20711-0.97832i$	$4\leftrightarrow 14$	8.79×10^{-14}	$(0.43,0.39,0.23)$	0.39	24/30/52	1.028873340639614
3	$(4,2)$	$-0.50000+1.32288i$	$4\leftrightarrow 12$	5.04×10^{-14}	$(0.54,0.62,-0.14)$	0.69	28/36/62	1.332406780851792
		$-0.50000-1.32288i$	$4\leftrightarrow 13$ 13	5.64×10^{-14} 8.22×10^{-14}	$(0.58,0.59,0.20)$	0.73	28/36/62	1.332638039347383 1.332638039347364
		$-1.50000+1.32288i$	$4\leftrightarrow 12$	3.52×10^{-13}	$(0.37,0.58,0.00)$	0.48	30/44/72	1.333967006828866
		$-1.50000-1.32288i$	$4\leftrightarrow 12$	2.87×10^{-13}	$(0.37,0.58,-0.02)$	0.48	30/44/72	1.333967006828922
	$(2,4)$	$-0.50000+1.32288i$	$4\leftrightarrow 13$	7.10×10^{-13}	$(-0.08,0.29,0.57)$	0.41	28/36/62	1.333352290650703
		$-0.50000-1.32288i$	$4\leftrightarrow 14$ 14	1.44×10^{-13} 1.41×10^{-13}	$(0.36,0.43,0.23)$	0.37	28/36/62	1.333372391725309 1.333372391725283
		$-1.50000+1.32288i$	$5\leftrightarrow 13$	4.23×10^{-13}	$(0.28,0.41,-0.12)$	0.27	30/44/72	1.333837383963351
		$-1.50000-1.32288i$	$4\leftrightarrow 12$	1.15×10^{-13}	$(0.28,0.42,0.11)$	0.27	30/44/72	1.333837383963306
4	$(4,2)$	$0.34781+1.02885i$	$6\leftrightarrow 10$	2.91×10^{-12}	$(0.58,0.60,-0.23)$	0.75	32/44/74	2.232892791283357
		$0.34781-1.02885i$	$6\leftrightarrow 10$	3.12×10^{-12}	$(0.54,0.42,0.46)$	0.67	40/68/106	2.229129242949574
		$-2.34781+1.02885i$	$5\leftrightarrow 10$	2.76×10^{-12}	$(-0.34,0.37,0.59)$	0.61	48/68/114	2.228185242921229
		$-2.34781-1.02885i$	$7\leftrightarrow 10$	2.07×10^{-11}	$(-0.36,0.56,0.42)$	0.62	48/68/114	2.226313685909248
	$(2,4)$	$0.34781+1.02885i$	$5\leftrightarrow 13$	7.14×10^{-12}	$(0.09,0.27,0.41)$	0.25	40/68/106	2.229754821992193
		$0.34781-1.02885i$	$4\leftrightarrow 13$	2.62×10^{-12}	$(0.41,0.44,0.31)$	0.46	32/44/74	2.232710498631728
		$-2.34781+1.02885i$	$6\leftrightarrow 11$	2.97×10^{-12}	$(-0.15,0.11,0.56)$	0.35	48/68/114	2.228627856967081
		$-2.34781-1.02885i$	$6\leftrightarrow 12$	1.32×10^{-11}	$(0.54,0.39,0.30)$	0.53	48/68/114	2.231108255566736
5	$(4,2)$	$-0.77330+1.46771i$	$5\leftrightarrow 13$	1.12×10^{-12}	$(0.61,0.63,-0.11)$	0.77	30/40/68	1.926198482790508
		$-0.77330-1.46771i$	$5\leftrightarrow 13$	2.68×10^{-13}	$(0.64,0.62,0.20)$	0.83	30/40/68	1.926630884489936
		$-1.22670+1.46771i$	$6\leftrightarrow 13$	3.32×10^{-12}	$(0.40,0.60,-0.02)$	0.52	34/48/80	1.921819864166848
		$-1.22670-1.46771i$	$5\leftrightarrow 13$	5.88×10^{-12}	$(0.40,0.60,0.00)$	0.53	34/48/80	1.919045852349486
	$(2,4)$	$-0.77330+1.46771i$	$4\leftrightarrow 13$	4.11×10^{-13}	$(0.33,0.19,0.61)$	0.51	30/40/68	1.926727950702606
		$-0.77330-1.46771i$	$4\leftrightarrow 13$ 13	2.48×10^{-12} 1.39×10^{-12}	$(0.58,0.32,0.39)$	0.60	30/40/68	1.926727950701057 1.926727950703559
		$-1.22670+1.46771i$	$4\leftrightarrow 12$	1.98×10^{-13}	$(0.18,0.18,0.50)$	0.31	32/44/74	1.927861496441281
		$-1.22670-1.46771i$	$4\leftrightarrow 12$ 4	4.11×10^{-13} 5.31×10^{-13}	$(0.30,0.43,0.13)$	0.29	34/48/80	1.925812729834557 1.925812729834050
86	$(4,2)$	$-0.09473+1.28374i$	$5\leftrightarrow 11$	1.61×10^{-12}	$(0.50,0.60,-0.16)$	0.64	40/60/98	2.597157107493283
		$-0.09473-1.28374i$	$6\leftrightarrow 10$	1.75×10^{-12}	$(0.69,0.42,0.46)$	0.87	34/52/84	2.597642037600054
		$-1.90527+1.28374i$	$6\leftrightarrow 12$	9.56×10^{-12}	$(0.52,0.64,0.01)$	0.68	36/54/88	2.608247243662376
		$-1.90527-1.28374i$	$8\leftrightarrow 11$	3.14×10^{-12}	$(-0.36,0.52,0.42)$	0.58	36/54/88	2.613640484296453
	$(2,4)$	$-0.09473+1.28374i$	$6\leftrightarrow 13$ 4 5	7.72×10^{-12} 7.61×10^{-12} failure.	$(0.32,0.24,0.49)$	0.40	38/60/96	2.598191623666021 2.598191623665997
		$-0.09473-1.28374i$	$5\leftrightarrow 13$	5.14×10^{-12}	$(0.29,0.43,0.23)$	0.32	40/60/98	2.595359034906877
		$-1.90527+1.28374i$	$6\leftrightarrow 12$	6.00×10^{-11}	$(-0.31,0.12,0.48)$	0.33	36/54/88	2.611707676528477
		$-1.90527-1.28374i$	$7\leftrightarrow 12$	6.58×10^{-11}	$(0.50,0.31,0.35)$	0.48	40/58/96	2.600005617840727
7	$(4,2)$	$0.88465+0.58974i$	$6\leftrightarrow 9$	1.79×10^{-12}	$(0.68,0.52,0.43)$	0.92	30/42/70	1.986069411118363
		$0.88465-0.58974i$	$5\leftrightarrow 10$	3.59×10^{-13}	$(0.60,0.53,0.35)$	0.77	30/42/70	1.985525637343016
		$-2.88465+0.58974i$	$6\leftrightarrow 11$	5.29×10^{-13}	$(-0.28,0.17,0.71)$	0.61	30/42/70	1.988902948387202
		$-2.88465-0.58974i$	$4\leftrightarrow 10$	4.04×10^{-12}	$(-0.32,0.71,0.24)$	0.66	30/42/70	1.987396444211165
	$(2,4)$	$0.88465+0.58974i$	$4\leftrightarrow 13$	4.10×10^{-13}	$(0.26,0.35,0.43)$	0.38	30/42/70	1.984732456421766
		$0.88465-0.58974i$	$4\leftrightarrow 11$	3.87×10^{-13}	$(0.33,0.57,0.52)$	0.70	30/42/70	1.984987636520065
		$-2.88465+0.58974i$	$5\leftrightarrow 13$	1.74×10^{-12}	$(-0.10,0.04,0.57)$	0.33	30/42/70	1.985674551422753
		$-2.88465-0.58974i$	$6\leftrightarrow 13$	2.96×10^{-12}	$(0.55,0.51,0.40)$	0.73	30/42/70	1.984728379613146

Tab. C.23: Data from Table 5.9 of [62].

n	(p,q)	γ	ε_V	deviation	B	$\|\|B\|\|_E^2$	S/V/E	covolume
1	$(4,2)$	$-1.00000+1.41421i$	$5\leftrightarrow 14$	1.35×10^{-14}	$(0.38,0.58,-0.04)$	0.48	16/13/28	Free?
		$-1.00000-1.41421i$	$5\leftrightarrow 13$	2.12×10^{-14}	$(0.38,0.58,0.02)$	0.49	16/13/28	Free?
	$(2,4)$	$-1.00000+1.41421i$	$3\leftrightarrow 13$	3.77×10^{-15}	$(0.04,0.22,0.46)$	0.26	10/7/16	Free?
		$-1.00000-1.41421i$	$3\leftrightarrow 13$	1.13×10^{-14}	$(0.25,0.42,0.14)$	0.25	16/13/28	Free?
2	$(4,2)$	$-1.00000+1.55377i$	$5\leftrightarrow 13$	5.06×10^{-14}	$(0.46,0.62,-0.06)$	0.60	16/13/28	Free?
		$-1.00000-1.55377i$	$4\leftrightarrow 13$	3.31×10^{-14}	$(0.47,0.62,0.04)$	0.61	16/13/28	Free?
	$(2,4)$	$-1.00000+1.55377i$	$3\leftrightarrow 14$	4.44×10^{-15}	$(0.12,0.25,0.51)$	0.33	10/7/16	Free?
		$-1.00000-1.55377i$	$3\leftrightarrow 13$	2.55×10^{-14}	$(0.34,0.44,0.16)$	0.34	16/13/28	Free?
3	$(4,2)$	$1.00755+0.51312i$	$7\leftrightarrow 11$	9.08×10^{-12}	$(0.61,0.52,0.42)$	0.82	52/76/126	3.667339431536167
		$1.00755-0.51312i$	$8\leftrightarrow 12$	4.91×10^{-10}	$(0.61,0.58,0.31)$	0.80	48/66/112	3.681087139097583
		$-3.00755+0.51312i$	$6\leftrightarrow 10$	6.04×10^{-09}	$(-0.27,0.24,0.75)$	0.70	64/98/160	3.679014906160397
		$-3.00755-0.51312i$	$6\leftrightarrow 11$	2.25×10^{-09}	$(-0.29,0.74,0.33)$	0.74	64/98/160	3.672306327492297
	$(2,4)$	$1.00755+0.51312i$	$6\leftrightarrow 12$	1.46×10^{-10}	$(0.20,0.35,0.41)$	0.33	52/76/126	3.675159456805927
		$1.00755-0.51312i$	$5\leftrightarrow 12$	6.87×10^{-11}	$(0.42,0.43,0.38)$	0.51	50/70/118	3.683332586060244
		$-3.00755+0.51312i$	$6\leftrightarrow 11$	2.74×10^{-09}	$(0.02,0.02,0.58)$	0.34	64/98/160	3.682522626324831
		$-3.00755-0.51312i$	$6\leftrightarrow 12$	5.21×10^{-10}	$(0.66,0.56,0.45)$	0.96	64/98/160	3.686890741856108
4	$(4,2)$	$0.17660+1.20282i$	$6\leftrightarrow 12$ 12	3.71×10^{-10} 2.14×10^{-10}	$(0.57,0.61,-0.21)$	0.75	38/54/90	3.316021637460442 3.316021637488303
		$0.17660-1.20282i$	$5\leftrightarrow 12$	7.19×10^{-11}	$(0.70,0.44,0.48)$	0.92	36/50/84	3.309439624044029
		$-2.17660+1.20282i$	$5\leftrightarrow 9$	9.24×10^{-11}	$(-0.39,0.39,0.60)$	0.66	50/76/124	3.315537713648098
		$-2.17660-1.20282i$	$5\leftrightarrow 9$	1.63×10^{-10}	$(-0.40,0.57,0.44)$	0.68	50/76/124	3.316268299807699
	$(2,4)$	$0.17660+1.20282i$	$5\leftrightarrow 12$	2.45×10^{-11}	$(0.59,0.21,0.62)$	0.77	36/50/84	3.308662886740566
		$0.17660-1.20282i$	$4\leftrightarrow 13$ 13	5.26×10^{-11} 2.55×10^{-11}	$(0.59,0.41,0.38)$	0.65	36/50/84	3.314732294986595 3.314732294987029
		$-2.17660+1.20282i$	$6\leftrightarrow 11$	2.99×10^{-10}	$(-0.24,0.12,0.58)$	0.41	50/76/124	3.319760346031557
		$-2.17660-1.20282i$	$6\leftrightarrow 13$	3.24×10^{-10}	$(0.56,0.43,0.28)$	0.58	50/76/124	3.309158288660671
5	$(4,2)$	$0.81516\pm 0.71242i$	$3\leftrightarrow 15$	failure.				
		$-2.81516\pm 0.71242i$	$3\leftrightarrow 15$	failure.				
	$(2,4)$	$0.81516\pm 0.71242i$	$3\leftrightarrow 15$	failure.				
		$-2.81516\pm 0.71242i$	$3\leftrightarrow 15$	failure.				
6	$(4,2)$	$0.58153\pm 0.93916i$	$3\leftrightarrow 15$	failure.				
		$-2.58153\pm 0.93916i$	$3\leftrightarrow 15$	failure.				
	$(2,4)$	$0.58153\pm 0.93916i$	$3\leftrightarrow 15$	failure.				
		$-2.58153\pm 0.93916i$	$3\leftrightarrow 15$	failure.				
7	$(4,2)$	$-0.30368\pm 1.43595i$	$3\leftrightarrow 15$	failure.				
		$-1.69632\pm 1.43595i$	$3\leftrightarrow 15$	failure.				
	$(2,4)$	$-0.30368\pm 1.43595i$	$3\leftrightarrow 15$	failure.				
		$-1.69632\pm 1.43595i$	$3\leftrightarrow 15$	failure.				
8	$(4,2)$	$-0.82835\pm 1.57669i$	$3\leftrightarrow 15$	failure.				
		$-1.17165\pm 1.57669i$	$3\leftrightarrow 15$	failure.				
	$(2,4)$	$-0.82835\pm 1.57669i$	$3\leftrightarrow 15$	failure.				
		$-1.17165\pm 1.57669i$	$3\leftrightarrow 15$	failure.				
9	$(4,2)$	$-1.00000+1.65289i$	$5\leftrightarrow 12$	1.42×10^{-13}	$(0.85,0.60,-0.01)$	1.08	16/13/28	Free?
		$-1.00000-1.65289i$	$4\leftrightarrow 14$	1.91×10^{-14}	$(0.84,0.59,0.09)$	1.07	18/19/36	Free?
	$(2,4)$	$-1.00000+1.65289i$	$3\leftrightarrow 12$	4.11×10^{-15}	$(0.18,0.27,0.54)$	0.40	10/7/16	Free?
		$-1.00000-1.65289i$	$4\leftrightarrow 13$	8.90×10^{-14}	$(0.77,0.42,0.58)$	1.10	18/19/36	Free?
10	$(4,2)$	$-1.00000+1.73205i$	$5\leftrightarrow 13$	4.24×10^{-14}	$(0.32,0.98,-0.01)$	1.06	16/12/28	Free?
		$-1.00000-1.73205i$	$4\leftrightarrow 13$	2.43×10^{-14}	$(0.65,0.66,0.04)$	0.85	16/12/28	Free?
	$(2,4)$	$-1.00000+1.73205i$	$3\leftrightarrow 14$	5.86×10^{-14}	$(0.13,0.32,0.95)$	1.01	18/18/36	Free?
		$-1.00000-1.73205i$	$4\leftrightarrow 13$	5.72×10^{-14}	$(0.79,0.35,0.55)$	1.05	18/18/36	Free?
11	$(4,2)$	$-0.78492+1.30714i$	$4\leftrightarrow 13$	2.35×10^{-14}	$(0.47,0.58,-0.08)$	0.56	24/28/50	0.824868942429759
		$-0.78492-1.30714i$	$4\leftrightarrow 13$	2.43×10^{-14}	$(0.46,0.58,0.11)$	0.56	24/28/50	0.824311173803825
		$-1.21508+1.30714i$	$4\leftrightarrow 12$	2.82×10^{-14}	$(0.31,0.55,-0.01)$	0.40	24/30/52	0.824697952334939
		$-1.21508-1.30714i$	$5\leftrightarrow 11$	7.80×10^{-14}	$(0.31,0.55,-0.01)$	0.40	24/30/52	0.824751919464297
	$(2,4)$	$-0.78492+1.30714i$	$4\leftrightarrow 13$ $4,5$	9.61×10^{-14} 2.89×10^{-14}	$(0.04,0.24,0.39)$	0.21	24/28/50	0.823655908723790 0.823655908723696
		$-0.78492-1.30714i$	$3\leftrightarrow 13$	8.04×10^{-14}	$(0.29,0.45,0.21)$	0.33	24/28/50	0.824214675045511
		$-1.21508+1.30714i$	$4\leftrightarrow 14$	1.12×10^{-13}	$(0.25,0.40,0.08)$	0.22	24/30/52	0.824574862965058
		$-1.21508-1.30714i$	$4\leftrightarrow 14$	9.30×10^{-14}	$(0.19,0.40,0.10)$	0.20	24/30/52	0.824533637845718

Tab. C.24: Data from Table 5.11 of [62].

n	(p,q)	γ	ε_V	deviation	B	$\|\|B\|\|_E^2$	S/V/E	covolume
1	$(6,2)$	$0.00000+1.00000i$	$4\leftrightarrow 13$	1.15×10^{-14}	$(0.46,0.52,0.41)$	0.64	20/22/40	0.610643729451075
		$0.00000-1.00000i$	$4\leftrightarrow 13$	1.14×10^{-14}	$(0.46,0.49,0.41)$	0.61	20/22/40	0.610124675044900
		$-1.00000+1.00000i$	$6\leftrightarrow 13$	2.98×10^{-14}	$(-0.09,0.65,0.28)$	0.51	18/22/38	0.610643729451021
			13	5.22×10^{-14}				0.610643729451050
		$-1.00000-1.00000i$	$4\leftrightarrow 12$	2.82×10^{-14}	$(-0.09,0.55,0.44)$	0.51	18/22/38	0.610643729451040
	$(2,6)$	$0.00000+1.00000i$	$3\leftrightarrow 13$	2.35×10^{-14}	$(0.12,0.36,0.06)$	0.15	20/22/40	0.609758967962758
			3	1.31×10^{-14}				0.609758967962676
		$0.00000-1.00000i$	$3\leftrightarrow 13$	1.17×10^{-14}	$(-0.12,0.34,0.15)$	0.15	20/22/40	0.610691628099879
		$-1.00000+1.00000i$	$3\leftrightarrow 14$	1.55×10^{-14}	$(0.04,0.36,-0.12)$	0.14	18/22/38	0.610643729451591
		$-1.00000-1.00000i$	$4\leftrightarrow 14$	6.66×10^{-15}	$(0.04,0.36,0.11)$	0.14	18/22/38	0.610643729451522
2	$(6,2)$	$0.87744+0.74486i$	$4\leftrightarrow 11$	5.82×10^{-14}	$(-0.47,0.71,0.23)$	0.78	20/26/44	1.022578858868845
		$0.87744-0.74486i$	$5\leftrightarrow 14$	6.93×10^{-14}	$(-0.43,0.67,0.31)$	0.72	22/30/50	1.021266309675690
		$-1.87744+0.74486i$	$4\leftrightarrow 13$	1.73×10^{-13}	$(0.20,0.68,0.35)$	0.63	26/38/62	1.020123408160045
		$-1.87744-0.74486i$	$6\leftrightarrow 13$	2.73×10^{-13}	$(0.06,0.79,0.24)$	0.69	26/36/60	1.021317792812498
	$(2,6)$	$0.87744+0.74486i$	$4\leftrightarrow 12$	4.35×10^{-14}	$(0.23,0.40,0.12)$	0.23	22/30/50	1.021266309674652
		$0.87744-0.74486i$	$3\leftrightarrow 13$	2.82×10^{-14}	$(0.23,0.41,0.14)$	0.24	20/26/44	1.022004856907895
			13	4.35×10^{-14}				1.022004856907774
		$-1.87744+0.74486i$	$4\leftrightarrow 13$	2.58×10^{-13}	$(-0.12,0.31,0.28)$	0.18	26/38/62	1.021187860075532
		$-1.87744-0.74486i$	$3\leftrightarrow 13$	4.07×10^{-14}	$(0.17,0.37,0.19)$	0.20	26/38/62	1.020786170671771
3	$(6,2)$	$-0.21508+1.30714i$	$4\leftrightarrow 11$	3.72×10^{-13}	$(-0.36,0.76,0.19)$	0.75	26/36/60	1.021347614330418
		$-0.21508-1.30714i$	$4\leftrightarrow 13$	1.18×10^{-13}	$(-0.32,0.79,-0.05)$	0.73	26/38/62	1.021768522986962
		$-0.78492+1.30714i$	$5\leftrightarrow 11$	1.38×10^{-13}	$(-0.10,0.80,0.22)$	0.70	24/26/48	1.021266309673058
		$-0.78492-1.30714i$	$4\leftrightarrow 13$	7.08×10^{-14}	$(-0.27,0.70,0.41)$	0.73	22/26/46	1.019497428415570
	$(2,6)$	$-0.21508+1.30714i$	$5\leftrightarrow 13$	4.31×10^{-14}	$(0.00,0.29,0.33)$	0.20	22/28/48	1.021444941021198
		$-0.21508-1.30714i$	$4\leftrightarrow 14$	2.00×10^{-13}	$(0.13,0.41,0.17)$	0.22	26/38/62	1.022378515079977
		$-0.78492+1.30714i$	$4\leftrightarrow 13$	1.58×10^{-14}	$(0.15,0.28,0.34)$	0.22	22/26/46	1.019497428415326
			13	1.51×10^{-13}				1.019497428414627
		$-0.78492-1.30714i$	$3\leftrightarrow 12$	6.68×10^{-14}	$(0.17,0.43,0.11)$	0.23	24/26/48	1.021266309674068
4	$(6,2)$	$1.41964+0.60629i$	$4\leftrightarrow 11$	6.80×10^{-12}	$(-0.10,0.78,0.35)$	0.74	30/48/76	1.853822926081037
			4	6.79×10^{-12}				1.853822926081040
			11					1.853822926081656
			10	failure.				
		$1.41964-0.60629i$	$5\leftrightarrow 11$	1.98×10^{-12}	$(-0.40,0.65,0.62)$	0.97	26/38/62	1.853113400207439
		$-2.41964+0.60629i$	$5\leftrightarrow 12$	1.22×10^{-12}	$(0.04,0.86,0.31)$	0.83	28/34/60	1.849394026788523
		$-2.41964-0.60629i$	$5\leftrightarrow 12$	3.20×10^{-13}	$(0.04,0.86,0.31)$	0.83	32/42/72	1.853421079185258
	$(2,6)$	$1.41964+0.60629i$	$4\leftrightarrow 13$	1.11×10^{-13}	$(0.10,0.44,0.57)$	0.53	28/40/66	1.849081037072025
		$1.41964-0.60629i$	$4\leftrightarrow 13$	7.19×10^{-14}	$(0.21,0.40,0.35)$	0.33	28/40/66	1.853421079184026
		$-2.41964+0.60629i$	$4\leftrightarrow 12$	6.56×10^{-13}	$(0.11,0.36,0.32)$	0.25	36/48/82	1.853815964173367
		$-2.41964-0.60629i$	$5\leftrightarrow 13$	1.59×10^{-12}	$(0.32,0.40,0.26)$	0.33	36/48/82	1.853815964175094
5	$(6,2)$	$0.77184+1.11514i$	$5\leftrightarrow 12$	1.27×10^{-13}	$(-0.46,0.68,0.47)$	0.90	26/32/56	1.853421079183917
		$0.77184-1.11514i$	$5\leftrightarrow 12$	3.38×10^{-13}	$(-0.45,0.81,0.18)$	0.88	28/36/62	1.854318201353658
		$-1.77184+1.11514i$	$5\leftrightarrow 12$	7.75×10^{-12}	$(0.01,0.86,0.31)$	0.84	30/48/76	1.852704566782116
		$-1.77184-1.11514i$	$5\leftrightarrow 13$	3.37×10^{-12}	$(0.01,0.87,0.31)$	0.85	28/40/66	1.853421079182651
	$(2,6)$	$0.77184+1.11514i$	$4\leftrightarrow 12$	1.87×10^{-12}	$(0.26,0.36,0.38)$	0.34	28/36/62	1.855268147965626
		$0.77184-1.11514i$	$4\leftrightarrow 13$	4.85×10^{-13}	$(0.23,0.42,0.29)$	0.31	26/32/56	1.851437303822129
		$-1.77184+1.11514i$	$6\leftrightarrow 12$	1.03×10^{-13}	$(-0.26,0.22,0.48)$	0.35	28/40/66	1.851412053612527
		$-1.77184-1.11514i$	$4\leftrightarrow 13$	6.15×10^{-13}	$(0.47,0.43,0.28)$	0.49	28/40/66	1.852848782731610
6	$(6,2)$	$1.23279+0.79255i$	$5\leftrightarrow 9$	1.13×10^{-11}	$(-0.15,0.74,0.37)$	0.70	34/48/80	2.104888866208937
		$1.23279-0.79255i$	$5\leftrightarrow 9$	6.59×10^{-12}	$(-0.47,0.73,0.49)$	0.99	26/34/58	2.112230922031761
		$-2.23279+0.79255i$	$5\leftrightarrow 12$	2.92×10^{-12}	$(0.02,0.85,0.16)$	0.75	34/50/82	2.108523889564296
		$-2.23279-0.79255i$	$5\leftrightarrow 12$	2.73×10^{-12}	$(0.02,0.86,0.26)$	0.81	34/48/80	2.109921205240231
	$(2,6)$	$1.23279+0.79255i$	$4\leftrightarrow 13$	3.36×10^{-13}	$(0.41,0.42,0.32)$	0.45	28/36/62	2.110523747092161
		$1.23279-0.79255i$	$3\leftrightarrow 13$	1.35×10^{-11}	$(0.32,0.53,0.45)$	0.58	28/36/62	2.110692420015884
			3	1.38×10^{-11}				2.110692420016515
			13	7.86×10^{-13}				2.110692420000852
		$-2.23279+0.79255i$	$5\leftrightarrow 12$	1.93×10^{-12}	$(0.03,0.34,0.34)$	0.23	34/50/82	2.109257715349852
		$-2.23279-0.79255i$	$4\leftrightarrow 13$	2.57×10^{-12}	$(0.32,0.40,0.25)$	0.32	34/50/82	2.109558614713941
7	$(6,2)$	$1.66236+0.56228i$	$5\leftrightarrow 10$	8.45×10^{-9}	$(-0.01,0.85,0.32)$	0.82	46/64/108	4.398582444311185
		$1.66236-0.56228i$	$6\leftrightarrow 7$	9.04×10^{-9}	$(-0.42,0.76,0.55)$	1.06	40/54/92	4.396495857865980
		$-2.66236+0.56228i$	$7\leftrightarrow 10$	1.56×10^{-10}	$(0.11,0.80,0.34)$	0.77	54/84/136	4.393018786595057
		$-2.66236-0.56228i$	$6\leftrightarrow 9$	2.34×10^{-10}	$(0.02,0.87,0.31)$	0.85	46/66/110	4.400822885695283
			9	7.16×10^{-10}				4.400822886121945
	$(2,6)$	$1.66236+0.56228i$	$4\leftrightarrow 12$	7.37×10^{-11}	$(0.56,0.31,0.54)$	0.70	42/56/96	4.402234623856832
		$1.66236-0.56228i$	$6\leftrightarrow 12$	1.36×10^{-10}	$(0.33,0.42,0.39)$	0.44	64/96/158	4.408328044588912
		$-2.66236+0.56228i$	$5\leftrightarrow 11$	6.76×10^{-10}	$(-0.05,0.21,0.55)$	0.35	50/70/118	4.403179323627133
		$-2.66236-0.56228i$	$5\leftrightarrow 11$	2.64×10^{-9}	$(0.56,0.53,0.55)$	0.90	48/68/114	4.401909752798479

Tab. C.25: Data from Table 5.12 of [62].

n	(p,q)	γ	ε_V	deviation	B	$\|\|B\|\|_E^2$	S/V/E	covolume
1	$(6,2)$	$0.34116 + 1.16154i$	$4 \leftrightarrow 11$	2.40×10^{-13}	$(-0.49, 0.53, 0.55)$	0.82	26/36/60	1.321558562482194
		$0.34116 - 1.16154i$	$5 \leftrightarrow 12$	1.51×10^{-13}	$(-0.48, 0.74, 0.22)$	0.82	26/36/60	1.320409963222722
		$-1.34116 + 1.16154i$	$4 \leftrightarrow 13$	1.56×10^{-13}	$(-0.11, 0.80, 0.33)$	0.75	22/30/50	1.318904026482595
		$-1.34116 - 1.16154i$	$4 \leftrightarrow 13$	1.77×10^{-13}	$(-0.03, 0.72, 0.52)$	0.79	22/30/50	1.319305550520877
	$(2,6)$	$0.34116 + 1.16154i$	$4 \leftrightarrow 13$	4.51×10^{-14}	$(0.22, 0.36, 0.30)$	0.27	22/30/50	1.320702203534967
		$0.34116 - 1.16154i$	$4 \leftrightarrow 13$	8.26×10^{-14}	$(0.11, 0.40, 0.23)$	0.22	26/36/60	1.322625363588137
		$-1.34116 + 1.16154i$	$5 \leftrightarrow 13$	4.60×10^{-14}	$(0.33, 0.41, -0.22)$	0.33	22/30/50	1.319305550520641
		$-1.34116 - 1.16154i$	$6 \leftrightarrow 13$	7.86×10^{-14}	$(0.33, 0.42, 0.20)$	0.33	22/30/50	1.319305550520717
2	$(6,2)$	$0.00000 + 1.41421i$	$5 \leftrightarrow 13$	7.93×10^{-13}	$(-0.50, 0.54, 0.51)$	0.80	30/44/72	2.008903021734428
		$0.00000 - 1.41421i$	$5 \leftrightarrow 11$	2.79×10^{-12}	$(-0.36, 0.85, 0.04)$	0.86	28/36/62	2.007682006681319
		$-1.00000 + 1.41421i$	$5 \leftrightarrow 12$	5.70×10^{-13}	$(-0.01, 0.84, 0.25)$	0.77	28/38/64	2.008067163155648
		$-1.00000 - 1.41421i$	$5 \leftrightarrow 12$	2.21×10^{-13}	$(-0.35, 0.88, 0.23)$	0.96	26/36/60	2.007584103872233
	$(2,6)$	$0.00000 + 1.41421i$	$4 \leftrightarrow 13$	1.50×10^{-13}	$(-0.07, 0.37, 0.65)$	0.57	28/36/62	2.007682006682812
		$0.00000 - 1.41421i$	$4 \leftrightarrow 14$ 14	1.66×10^{-13} 1.55×10^{-13}	$(0.49, 0.46, 0.30)$	0.54	28/36/62	2.007682006683356 2.007682006683051
		$-1.00000 + 1.41421i$	$4 \leftrightarrow 14$	3.46×10^{-13}	$(0.31, 0.45, -0.16)$	0.32	28/38/64	2.007687025048551
		$-1.00000 - 1.41421i$	$5 \leftrightarrow 14$ 14	3.38×10^{-13} 2.28×10^{-13}	$(0.31, 0.46, 0.15)$	0.32	28/38/64	2.007687025047486 2.007687025046759
3	$(6,2)$	$1.00000 + 1.00000i$	$4 \leftrightarrow 10$	1.26×10^{-11}	$(-0.39, 0.80, 0.32)$	0.89	32/46/76	3.055778380625268
		$1.00000 - 1.00000i$	$5 \leftrightarrow 10$	8.00×10^{-12}	$(-0.48, 0.73, 0.43)$	0.95	30/46/74	3.052092045493262
		$-2.00000 + 1.00000i$	$6 \leftrightarrow 13$ 13	1.28×10^{-10} 4.66×10^{-11}	$(0.10, 0.81, 0.34)$	0.77	38/54/90	3.048964087845691 3.048964087743448
			12	failure.				
		$-2.00000 - 1.00000i$	$5 \leftrightarrow 12$	5.73×10^{-11}	$(0.02, 0.87, 0.31)$	0.85	38/52/88	3.048270916651866
	$(2,6)$	$1.00000 + 1.00000i$	$4 \leftrightarrow 13$	2.78×10^{-12}	$(0.13, 0.30, 0.46)$	0.31	38/64/100	3.054044210834238
		$1.00000 - 1.00000i$	$5 \leftrightarrow 12$	9.87×10^{-13}	$(0.46, 0.44, 0.37)$	0.55	30/46/74	3.053218647256478
		$-2.00000 + 1.00000i$	$6 \leftrightarrow 12$	1.18×10^{-11}	$(-0.19, 0.20, 0.52)$	0.35	36/51/85	3.053293429162713
		$-2.00000 - 1.00000i$	$5 \leftrightarrow 13$	2.06×10^{-12}	$(0.54, 0.45, 0.33)$	0.60	36/51/85	3.055401540750088
4	$(6,2)$	$-0.31945 + 1.63317i$	$5 \leftrightarrow 10$ 10	2.20×10^{-11} 6.49×10^{-11}	$(-0.21, 0.94, 0.03)$	0.93	28/34/60	2.646717797379673 2.646717797033200
		$-0.31945 - 1.63317i$	$4 \leftrightarrow 12$	9.54×10^{-12}	$(-0.22, 0.94, -0.10)$	0.95	28/34/60	2.646861970382090
		$-0.68055 + 1.63317i$	$5 \leftrightarrow 13$	5.85×10^{-12}	$(0.02, 0.85, 0.16)$	0.74	38/54/90	2.643405336574825
		$-0.68055 - 1.63317i$	$6 \leftrightarrow 13$	6.11×10^{-12}	$(-0.36, 0.83, 0.55)$	1.12	32/46/76	2.646336153435846
	$(2,6)$	$-0.31945 + 1.63317i$	$5 \leftrightarrow 10$	1.34×10^{-12}	$(0.22, 0.27, 0.59)$	0.46	30/40/68	2.646695166978066
		$-0.31945 - 1.63317i$	$4 \leftrightarrow 13$	6.55×10^{-13}	$(0.65, 0.59, 0.39)$	0.93	30/40/68	2.646859749260644
		$-0.68055 + 1.63317i$	$5 \leftrightarrow 13$	9.35×10^{-13}	$(0.20, 0.21, 0.60)$	0.44	32/44/74	2.646859749267347
		$-0.68055 - 1.63317i$	$4 \leftrightarrow 11$	2.27×10^{-12}	$(0.38, 0.48, 0.18)$	0.41	38/54/90	2.648010914887581
5	$(6,2)$	$0.50000 + 1.32288i$	$5 \leftrightarrow 12$ 5	2.13×10^{-11} 1.75×10^{-11}	$(-0.42, 0.64, 0.48)$	0.81	40/62/100	3.551014475800329 3.551014475800596
		$0.50000 - 1.32288i$	$6 \leftrightarrow 12$	1.13×10^{-10}	$(-0.45, 0.83, 0.29)$	0.98	38/60/96	3.553374419561017
		$-1.50000 + 1.32288i$	$5 \leftrightarrow 10$	3.51×10^{-11}	$(0.00, 0.86, 0.31)$	0.84	38/60/96	3.551123647189390
		$-1.50000 - 1.32288i$	$7 \leftrightarrow 10$ 10	1.14×10^{-10} 3.28×10^{-10}	$(-0.01, 0.80, 0.34)$	0.76	38/60/96	3.547460225579417 3.547460225069884
	$(2,6)$	$0.50000 + 1.32288i$	$5 \leftrightarrow 13$	1.00×10^{-11}	$(0.36, 0.28, 0.56)$	0.52	34/52/84	3.549442254748726
		$0.50000 - 1.32288i$	$5 \leftrightarrow 13$	1.49×10^{-11}	$(0.35, 0.44, 0.29)$	0.40	40/64/102	3.555072579853269
		$-1.50000 + 1.32288i$	$7 \leftrightarrow 13$	2.29×10^{-11}	$(0.51, 0.44, -0.27)$	0.54	38/58/94	3.553351881088147
		$-1.50000 - 1.32288i$	$4 \leftrightarrow 12$ 4	4.79×10^{-11} 3.44×10^{-11}	$(0.52, 0.41, 0.32)$	0.55	42/68/108	3.554390070486208 3.554390070475067

Tab. C.26: Data from Table 5.13 of [62].