Orbiter User's Guide

Anton Betten

November 28, 2019

Abstract

The open source package Orbiter is devoted to the classification of combinatorial objects. Orbiter provides algorithms for effective handling of finite permutation groups and their orbits in various actions. This guide is targeted for users who are not familiar to programming. It describes how Orbiter is installed and used.

1 Introduction

Orbiter is a software package for the classification of combinatorial objects. This User's guide shows how Orbiter can be used. Orbiter is a library of C++ classes, together with a set of ready-to-use applications. There is no command line interface. The Orbiter applications can be invoked using the command line interface (for instance from Unix terminals). It is also possible to write shell scripts or makefiles.

The installation of Orbiter requires the following steps:

(a) Ensure that `git` and the C++ development suite are installed (`gnuc` and `make`). Windows users may have to install `cygwin` (plus the extra packages git, make, gnuc). Macintosh users may have to install the xcode development tools from the appstore (it is free). Linux users may have to install the development packages. Orbiter often produces latex reports. In order to compile these files, make sure you have latex installed (Orbiter programs run without it though).

(b) Clone the Orbiter source tree from github (abetten/orbiter). The commands are:

```
git clone <github-orbiter-path>
```

where `<github-orbiter-path>` has to be replaced by the actual address provided by github. To obtain this path, find Orbiter on github, then click on the green box that says "Clone or download" and copy the address into the clipboard by clicking the clipboard symbol (see Figure 1). Back in the terminal, you can paste this text after the `git clone` command.

(c) Issue the following commands to complete the download of submodules:

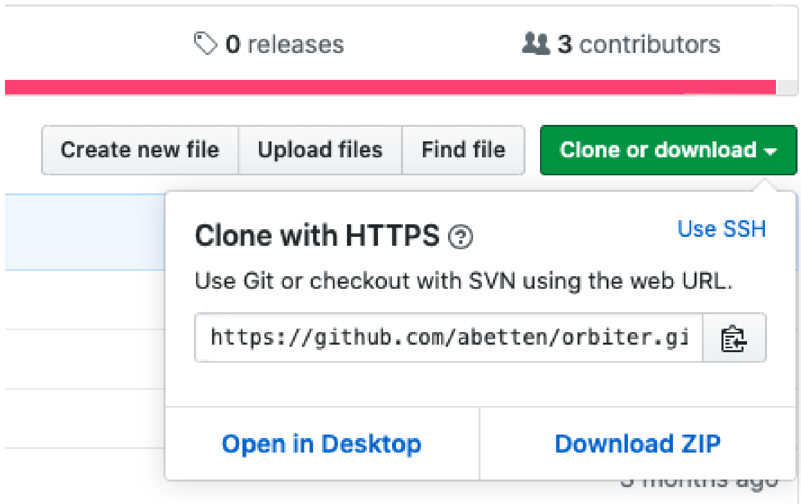


Figure 1: GitHub Clone or Download button

```
cd orbiter
git submodule init
git submodule update
```

(d) Issue the following commands to compile Orbiter using recursive makefiles:

```
make
make install
```

These two commands compile the Orbiter source tree and copy the executables to the subdirectory bin inside the Orbiter source tree. Compiling Orbiter will take a little while (5 minutes, depending on the speed of the machine). Depending on the compiler, some warnings will be produced, though none of them are serious. If an error appears, please check that you followed all the steps above (including the git submodule commands fro the previous steps). All executables will first be created in the subtree `ORBITER/src/apps` and will have the file extension `.out`. The `make install` command copies the executables to the bin subdirectory. A list of all executables is given in Appendix K. The Orbiter directory structure is shown in Appendix J.

2 Finite Fields

Finite fields and projective spaces over finite fields play an important role in Orbiter. The command

```
cheat_sheet_GF.out -q <q>
```

Subfield	Polynomial	Numerical rank
$\mathbb{F}_4$	$X^2 + X + 1$	7
$\mathbb{F}_8$	$X^3 + X + 1$	11

Table 1: The subfields of $\mathbb{F}_{64}$

creates a report for the field $\mathbb{F}_q$. The elements of the field $\mathbb{F}_q$ are represented in different ways. Suppose that $q = p^e$ for some prime p and some integer $q \geq 1$. The elements of $\mathbb{F}_q$ are mapped bijectively to the integers in the interval $[0, q-1]$, using the base-p representation. If $e = 1$, the map takes the residue class a mod p with $0 \leq a < p$ to the integer a. Otherwise, we write the field element as

$$\sum_{h=0}^{e-1} a_i \alpha^i$$

where α is the root of some irreducible polynomial $m(X)$ of degree e over $\mathbb{F}_p$ amd $0 \leq a_i < p$ for all i. The associated integer is obtained as

$$\sum_{h=0}^{e-1} a_i p^i.$$

This is the numerical rank of the polynomial. This representation takes 0 in $\mathbb{F}_q$ to the integer 0. Likewise, $1 \in \mathbb{F}_q$ is mapped to the integer 1. Arithmetic is done by considering the polynomials over $\mathbb{F}_p$ and modulo the irreducible polynomial $m(X)$ with root α. For instance, the field $\mathbb{F}_4$ is created using the polynomial $m(X) = X^2 + X + 1$. The elements are

$$0, \quad 1, \quad 2 = \alpha, \quad 3 = \alpha + 1.$$

Addition and multiplication tables are listed in the report in Appendix A. Orbiter maintains a small database of primitive (irreducible) polynomials for the purposes of creating finite fields. This means that the residue class of α is a primitive element of the field, where α is a root of the polynomial. Appendix A shows the report for the field $\mathbb{F}_4$.

Some Computer algebra systems (GAP [9] and Magma [5]) use Conway polynomials to generate finite fields. The reason for doing so is that Conway polynomials are consistent with respect to subfields. This means that the Conway polynomial for the subfield is always the correct one considering the field generated by a Conway polynomial. However, the search for Conway polynomials is tedious. To compute the next Conway polynomial, each of the lower degree Conway polynomials must be known. If desired, Orbiter provides an override option to choose a specified polynomial to create the finite field. An example for this technique will appear in Section 6. If field extensions are desired, it is advised to create a report for the largest field. The report will show the irreducible polynomials associated to each non-trivial subfield of this particular field. Then, the override option can be used to specify the appropriate polynomial whenever any of these subfields is needed. Table 1 shows the subfields of $\mathbb{F}_{64}$ generated by the polynomial $X^6 + X^5 + 1$ whose numerical rank is 97.

It is also possible to change the field generated by one polynomial to the field generated by another polynomial. To this end, write the elements terms of powers of primitive elements in each field. The isomorphism maps powers to corresponding powers.

3 Finite Projective Spaces

Finite projective spaces and their groups are essential objects in Orbiter. The projective space $\mathrm{PG}(n,q)$ is the set of non-zero subspaces of $\mathbb{F}_q^{n+1}$ ordered with respect to inclusion. The projective dimension of a subspace is always one less than the vector space dimension. So, a projective point is a vector subspace of dimension one. A projective line is a vector subspace of dimension two, etc. A point is written as $P(\mathbf{x})$ for some vector $\mathbf{x} = (x_0, \ldots, x_n)$ with $x_0, \ldots, x_n \in \mathbb{F}_q$, not all zero. For any non-zero element $\lambda \in \mathbb{F}_q$, $P(\lambda\mathbf{x})$ is the same point as $P(\mathbf{x})$. For $\mathbf{a} = (a_0, \ldots, a_n) \in \mathbb{F}_q$, not all zero, the symbol $[\mathbf{a}]$ represents the line

$$\{P(\mathbf{x}) \mid \mathbf{a} \cdot \mathbf{x} = \sum_{i=0}^{n} a_i x_i = 0\}.$$

For any non-zero element $\lambda \in \mathbb{F}_q$, $[\lambda\mathbf{a}] = [\mathbf{a}]$.

A collineation of a projective space π is a bijective mapping from the points of π to themselves which preserves collinearity. That is, a collineation φ maps any three collinear points P, Q, R to another collinear triple $\varphi(P), \varphi(Q), \varphi(R)$. The collineations form a group with respect to composition, the collineation group. If M is the matrix of an endomorphism, then Ψ_M is the induced map on projective space. By considering the homomorphism $M \mapsto \Psi_M$, the group $\mathrm{GL}(n+1,q)$ of invertible endomorphisms becomes a subgroup of the group of collineations of $\mathrm{PG}(n,q)$. This is the projectivity group $\mathrm{PGL}(n+1,q)$. It is isomorphic to $\mathrm{GL}(n+1,q)/\mathbb{F}_q^{\times}$. Another source of collineations is this: Let $\Phi \in \mathrm{Aut}(\mathbb{F}_q)$ be a field automorphism. Then Φ acts on projective space by sending $P(\mathbf{x})$ to $P(\mathbf{x}\Phi)$. This map is another type of collineation, called automorphic collineation. This way, $\mathrm{Aut}(\mathbb{F}_q)$ can be considered another subgroup of the group of collineations. If $q = p^h$ for some prime p and some integer h then

$$\Phi_0 : \mathbb{F}_q \to \mathbb{F}_q,\ x \mapsto x^p$$

is a generator for the cyclic group $C_h \simeq \mathrm{Aut}(\mathbb{F}_q)$. By the fundamental theorem of projective geometry, the collineation group of $\mathrm{PG}(n,q)$ ($n \geq 2$) is isomorphic to the semidirect product of the projectivity group and the automorphism group of the field. The collineation group is $\mathrm{P\Gamma L}(n+1,q) = \mathrm{PGL}(n+1,q) \ltimes \mathrm{Aut}(\mathbb{F}_q)$. We use the following notation for elements of $\mathrm{P\Gamma L}(n+1,q)$. Let Φ_0 be a generator for $\mathrm{Aut}(\mathbb{F}_q)$ and let $M \in \mathrm{GL}(n+1,q)$. The map

$$(\Psi_M, \Phi_0^k) : \mathrm{PG}(n,q) \to \mathrm{PG}(n,q),\ P(\mathbf{x}) \mapsto P(\mathbf{y}), \quad \mathbf{y} = (\mathbf{x} \cdot M)^{\Phi_0^k}$$

is denoted as

$$M_k. \tag{1}$$

The identity element is I_0, where I is the identity matrix and 0 is the residue class modulo h. The rules for multiplication and inversion in the collineation group are given as

$$\begin{aligned} M_k \cdot N_l &= \left(M^{\Phi^{-k} \mod h} \cdot N\right)_{k+l \mod h}, & (2)\\ \left(M_k\right)^{-1} &= \left(\left(M^{-1}\right)^{\Phi^{-k} \mod h}\right)_{-k \mod h}. & (3) \end{aligned}$$

For later use, we record one well-known fact about the action of the projectivity group: The projectivity group acts transitively on the set of frames ($n+1$ points, no n of which are contained in a hyperplane). The order of $\mathrm{P\Gamma L}(n,q)$ is

$$|\mathrm{P\Gamma L}(n,q)| = \frac{h\prod_{k=0}^{n-1}(q^n - q^k)}{q-1}.$$

A correlation is a one-to-one mapping between the set of points and the set of hyperplanes which reverses incidence. So, if ρ is a correlation and P is a point and ℓ is a hyperplane then P^ρ is a hyperplane and ℓ^ρ is a point and

$$\ell^\rho \in P^\rho \iff P \in \ell.$$

A correlation of order two is called polarity. The standard polarity is the map

$$\rho : \mathcal{P} \leftrightarrow \mathcal{L},\ P(\mathbf{x}) \leftrightarrow [\mathbf{x}].$$

The command

```
cheat_sheet_PG.out -n <n> -q <q>
```

creates a report for the projective geometry $\mathrm{PG}(n,q)$. Appendix B shows such a report for $\mathrm{PG}(2,4)$. It is important to note that Orbiter has enumerators for points and subspaces of $\mathrm{PG}(n,q)$. The point enumerator allows to represent the points using the integer interval $[0, \theta_n(q)-1]$, where

$$\theta_n(q) = \frac{q^{n+1}-1}{q-1}.$$

The points in projective geometry are the one-dimensional subspaces.

In order to enumerate the points, right-normalized representatives are considered. There is one important convention. In projective geometry, a frame is a special set of vectors. Specifically, in $\mathrm{PG}(n,q)$, a frame is a set of $n+1$ vectors, no n of which are contained in a hyperplane. The standard frame consists of all points represented by vectors of the form

$$\mathbf{e}_i = (0,\ldots,0,1,0,\ldots,0),$$

with a one in the ith coordinate, together with the all-one vector. The Orbiter enumerator for projective points assigns the numbers

$$0, 1, \ldots, n-1, n$$

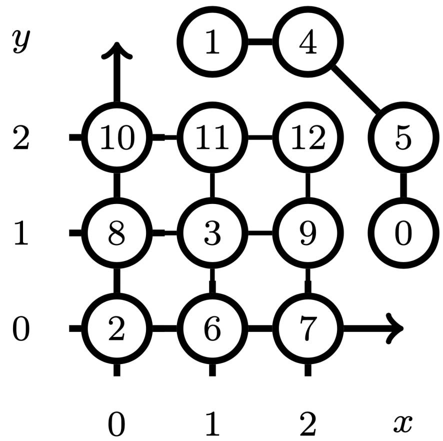


Figure 2: The projective plane $\mathrm{PG}(2,3)$

$P_0 = (1,0,0)$	$P_4 = (1,1,0)$	$P_8 = (0,1,1)$	$P_{12} = (2,2,1)$
$P_1 = (0,1,0)$	$P_5 = (2,1,0)$	$P_9 = (2,1,1)$	
$P_2 = (0,0,1)$	$P_6 = (1,0,1)$	$P_{10} = (0,2,1)$	
$P_3 = (1,1,1)$	$P_7 = (2,0,1)$	$P_{11} = (1,2,1)$	

Table 2: The 13 points of $\mathrm{PG}(2,3)$

to the frame. All other vectors are assigned higher numbers. This is illustrated in Figure 2 for the projective plane $\mathrm{PG}(2,3)$. We use capital letters for homogeneous coordindates, in this case (X,Y,Z). We use lowercase letters for cartesian coordinates in the affine plane $Z \neq 0$. A point (X,Y,Z) with $Z \neq 0$ determines the affine point (x,y) where

$$x = \frac{X}{Z}, \quad y = \frac{Y}{Z}.$$

The x/y grid shows the affine plane $Z \neq 0$. The bent line at the top right corner represents the line at infinity with equation $Z = 0$. The points of the frame are labeled $0,1,2$ and 3. The first two points lie on the line at infinity, where the horizontal and vertical lines intersect, respectively. The point 2 is the origin of the x/y plane, and 3 is the unit point $(x,y) = (1,1)$. Once these four points have been labeled, the remaining points are labeled as well. This is done by starting with the remaining points on the line at infinity, and then labeling the remaining points in the x,y plane one horizontal row at a time. All told, we have the 13 points listed in Table 2. The 13 lines are shown in Table 3. A line is represented by a 2×3 matrix. The line is the rowspan of the matrix, considered as projective set.

The reason for this special labeling of the points is that the frame plays a special role in projective geometry. The projective linear group is transitive on frames. Many combinatorial objects in projective space will therefore be equivalent to one containing the standard frame. If the lecicographic ordering on subsets is used to pick orbit representatives, then the standard frame is automatically part of the object (for such objects). This is often convenient.

$$L_0 = \begin{bmatrix} 100 \\ 010 \end{bmatrix} \quad L_4 = \begin{bmatrix} 101 \\ 010 \end{bmatrix} \quad L_8 = \begin{bmatrix} 102 \\ 010 \end{bmatrix} \quad L_{12} = \begin{bmatrix} 010 \\ 001 \end{bmatrix}$$

$$L_1 = \begin{bmatrix} 100 \\ 011 \end{bmatrix} \quad L_5 = \begin{bmatrix} 101 \\ 011 \end{bmatrix} \quad L_9 = \begin{bmatrix} 102 \\ 011 \end{bmatrix}$$

$$L_2 = \begin{bmatrix} 100 \\ 012 \end{bmatrix} \quad L_6 = \begin{bmatrix} 101 \\ 012 \end{bmatrix} \quad L_{10} = \begin{bmatrix} 102 \\ 012 \end{bmatrix}$$

$$L_3 = \begin{bmatrix} 100 \\ 001 \end{bmatrix} \quad L_7 = \begin{bmatrix} 110 \\ 001 \end{bmatrix} \quad L_{11} = \begin{bmatrix} 120 \\ 001 \end{bmatrix}$$

Table 3: The 13 lines of $\mathrm{PG}(2,3)$

Examples of combinatorial objects which contain a frame are ovals in projective planes and MDS-codes.

There are important group actions associated with projective spaces. For $n \geq 2$, the automorphism group of $\mathrm{PG}(n,q)$ is the collineation group $\mathrm{P\Gamma L}(n+1,q)$. This group acts on the set of points. There is an associateed action on the hyperplanes preserving incidence. This is the contragredient action. It is related to the dual coordinates for hyperplanes.

4 Algebraic Sets

A very important notion in projective geometry is that of algebraic sets. A set of points V in $\mathrm{PG}(n,q)$ is algebraic if there is a set of homogeneous polynomials $p_1,\ldots,p_r$ whose roots over $\mathbb{F}_q$ are the given set. In this case, we write $V = \mathbf{v}(p_1,\ldots,p_r)$. The set V is often called the variety of $p_1,\ldots,p_r$.

Conversely, given a set of points V in $\mathrm{PG}(n,q)$, the ideal $I(V)$ is the set of homogeneous polynomials in $\mathbb{F}_q[X_0,\ldots,X_n]$ which vanish on all of V. This set is an ideal in the polynomial ring. In fact, it is a pricipal ideal, meaning that it is generated by one element only. Orbiter has ways to compute the variety of a polynomial ideal and to compute a generator for the ideal of a set.

Interestingly, in $\mathrm{PG}(n,q)$, every set is algebraic of degree at most $(n+1)(q-1)$ [10]. An algorithm to compute the ideal of a set is given in [6], based on earlier work described in [2]. Sets which are not algebraic can be found in projective geometries over infinite fields.

Table 4 shows the Orbiter monomial orderings for degrees $2, 3$ and 4 in a plane. Suppose we are interested in $\mathbb{F}_{11}$ rational points of the elliptic curve $y^2 = x^3 + x + 3$. We write $x^3 + 3 - y^2 + x = 0$. Homogenizing yields $X^3 + 3Z^3 - Y^2Z + XZ = 0$. Using X_0, X_1, X_2 instead of X, Y, Z yields

$$X_0^3 + 3X_2^3 + 10X_1^2X_2 + X_0X_2^2 = 0.$$

h	monomial	vector
0	X_0^2	$(2,0,0)$
1	X_1^2	$(0,2,0)$
2	X_2^2	$(0,0,2)$
3	X_0X_1	$(1,1,0)$
4	X_0X_2	$(1,0,1)$
5	X_1X_2	$(0,1,1)$

h	monomial	vector
0	X_0^3	$(3,0,0)$
1	X_1^3	$(0,3,0)$
2	X_2^3	$(0,0,3)$
3	$X_0^2X_1$	$(2,1,0)$
4	$X_0^2X_2$	$(2,0,1)$
5	$X_0X_1^2$	$(1,2,0)$
6	$X_1^2X_2$	$(0,2,1)$
7	$X_0X_2^2$	$(1,0,2)$
8	$X_1X_2^2$	$(0,1,2)$
9	$X_0X_1X_2$	$(1,1,1)$

h	monomial	vector
0	X_0^4	$(4,0,0)$
1	X_1^4	$(0,4,0)$
2	X_2^4	$(0,0,4)$
3	$X_0^3X_1$	$(3,1,0)$
4	$X_0^3X_2$	$(3,0,1)$
5	$X_0X_1^3$	$(1,3,0)$
6	$X_1^3X_2$	$(0,3,1)$
7	$X_0X_2^3$	$(1,0,3)$
8	$X_1X_2^3$	$(0,1,3)$
9	$X_0^2X_1^2$	$(2,2,0)$
10	$X_0^2X_2^2$	$(2,0,2)$
11	$X_1^2X_2^2$	$(0,2,2)$
12	$X_0^2X_1X_2$	$(2,1,1)$
13	$X_0X_1^2X_2$	$(1,2,1)$
14	$X_0X_1X_2^2$	$(1,1,2)$

Table 4: The Orbiter ordering of monomials of degree $2,3$ and 4 in a plane

Using the indexing of monomials from Table 4, we record the following pairs (a,i) where a is the coefficient and i is the index of the monomial

$$(1,0),\ (3,2),\ (10,6),\ (1,7).$$

This is concatenated to the sequence $1,0,3,2,10,6,1,7$. The Orbiter command

```
create_object.out -v 2 -q 11 -n 2 \
    -projective_variety "EC" 3 "1,0,3,2,10,6,1,7"
```

creates the algebraic set associated to the cubic curve $y^2 = x^3 + x + 3$ in $\mathrm{PG}(2,11)$. It turns out that there are exactly 18 points over $\mathbb{F}_{11}$ (cf. Figure 3).

Table 5 shows the Orbiter monomial orderings for degrees 2 and 3 in $\mathrm{PG}(3,q)$.

5 Permutation Groups and Stabilizer Chains

Finite permutation groups are stored on a computer using a stabilizer chain (or Sims chain). This is because groups can be very large, so storing all elements is prohibitive. The idea is to replace the set of elements by a data structure which allows to access each group element in a deterministic fashion. In particular, the data structure tells us the order of the group. For the general theory of permutation group algorithms, see [18, 11, 8], for instance. The

h	monomial	vector
0	X_0^2	$(2,0,0,0)$
1	X_1^2	$(0,2,0,0)$
2	X_2^2	$(0,0,2,0)$
3	X_3^2	$(0,0,0,2)$
4	X_0X_1	$(1,1,0,0)$
5	X_0X_2	$(1,0,1,0)$
6	X_0X_3	$(1,0,0,1)$
7	X_1X_2	$(0,1,1,0)$
8	X_1X_3	$(0,1,0,1)$
9	X_2X_3	$(0,0,1,1)$

h	monomial	vector
0	X_0^3	$(3,0,0,0)$
1	X_1^3	$(0,3,0,0)$
2	X_2^3	$(0,0,3,0)$
3	X_3^3	$(0,0,0,3)$
4	$X_0^2X_1$	$(2,1,0,0)$
5	$X_0^2X_2$	$(2,0,1,0)$
6	$X_0^2X_3$	$(2,0,0,1)$
7	$X_0X_1^2$	$(1,2,0,0)$
8	$X_1^2X_2$	$(0,2,1,0)$
9	$X_1^2X_3$	$(0,2,0,1)$
10	$X_0X_2^2$	$(1,0,2,0)$
11	$X_1X_2^2$	$(0,1,2,0)$
12	$X_2^2X_3$	$(0,0,2,1)$
13	$X_0X_3^2$	$(1,0,0,2)$
14	$X_1X_3^2$	$(0,1,0,2)$
15	$X_2X_3^2$	$(0,0,1,2)$
16	$X_0X_1X_2$	$(1,1,1,0)$
17	$X_0X_1X_3$	$(1,1,0,1)$
18	$X_0X_2X_3$	$(1,0,1,1)$
19	$X_1X_2X_3$	$(0,1,1,1)$

Table 5: The Orbiter ordering of monomials of degree 2 and 3 in three-space

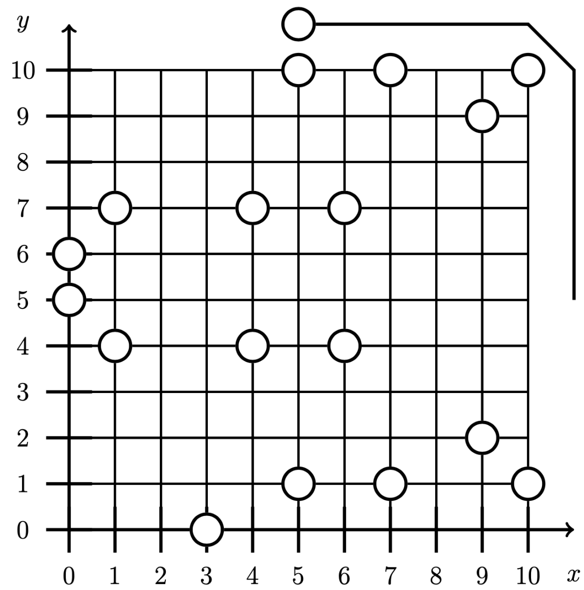


Figure 3: Elliptic curve $y^2 \equiv x^3 + x + 3 \bmod 11$

two important notions are that of a base and a strong generating set. We will summarize the main technique and give an example.

The way that a stabilizer chain is set up is by repeated application of the orbit-stabilizer theorem from basic algebra. A point is chosen and its orbit is computed. The subgroup which fixes the point is smaller (unless the point is fix, which is easy to avoid), and the subgroup is represented in a recursive fashion in the same way. This process is iterated until at one point the stabilizer is trivial. For a permutation group, this is always the case. The sequence of points whose orbits which have been stabilizer is called a base. The ith point in the base is called the ith base point. The orbit of the ith base point is called the ith basic orbit. The elements of the group fall into disjoint cosets of the stabilizer of the first base point. The subgroup again falls into disjoint cosets of the stabilizer of the next base point and so forth. The elements in the non-tivial cosets are partitioned just like the subgroup, using a fixed coset representative. In the end, each group element is discribed by a unique choice of cosets. The coset representatives can be computed using Schreier trees. Because of the orbit-stabilizer theorem, the order of the group is the product of the lengths of the basic orbits. Assume that $B = (b_1, \ldots, b_k)$ is a base for a permutation group G. The ith stabilizer subgroup in the subgroup chain is

$$G^{(i)} = G_{b_1,\ldots,b_{i-1}},$$

the subgroup which stabilizes each of the first $i - 1$ base points. The convention is that $G^{(1)} = G$. This way, the stabilizer chain is

$$G = G^{(1)} \geq G^{(2)} \geq G^{(3)} \geq \cdots \geq G^{(k+1)} = 1.$$

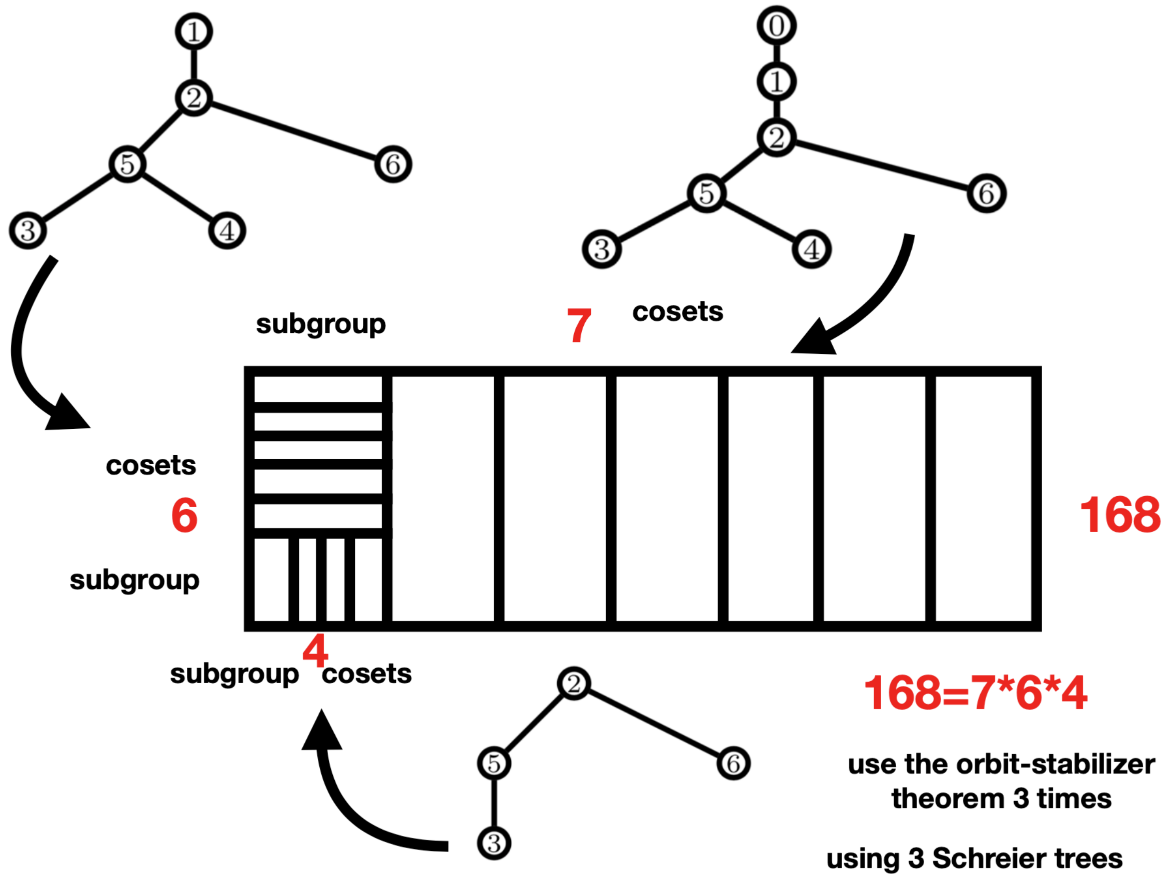


Figure 4: The stabilizer chain for PSL(3, 2)

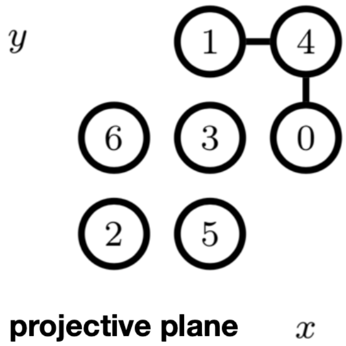
Let us consider an example. The group PSL(3, 2) of order 168 has a stabilizer chain of length 3, with respect to the base points

$$b_1 = P(1,0,0), \quad b_2 = P(0,1,0), \quad b_3 = P(0,0,1).$$

Figure 4 shows the partition of PSL(3, 2) which results from this choice of base points. The first basic orbit has length 7. The cosets of the stabilizer partition the group into 7 sets, represented by the large vertical blocks. The atabilizer of the first base point is partitioned according to the orbit of the second base point, which has length 6. These are the 6 blocks on the far left. The block in the lower left represents the stabilizer of the first two base points. This subgroup is partitioned into an orbit of length 4 of the third base point, whose stabilizer in the previous group is trivial. This corresponds to writing the group order as

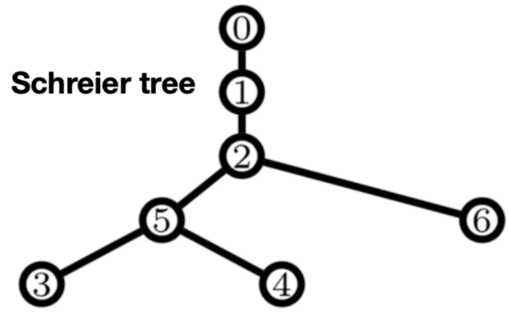
$$168 = 7 \cdot 6 \cdot 4.$$

A closer look at the first basic orbit is provided by Figure 5. We see the Orbiter labeling of points of PG(2, 2), the chosen generators, and the resulting Schreier tree. This tree provides



i	P_i
0	$(1,0,0)$
1	$(0,1,0)$
2	$(0,0,1)$
3	$(1,1,1)$
4	$(1,1,0)$
5	$(1,0,1)$
6	$(0,1,1)$

set of points



$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

group generators

Figure 5: The first basic orbit in the stabilizer chain for $\mathrm{PSL}(3,2)$

the unique coset representatives for the cosets of the stabilizer of the point 0. Orbiter allows to access group elements using an enumerator. The elements of the group G are represented by the integer interval $[0, |G|-1]$. The figure also shows the Schreier trees which represent the basic orbits.

A strong generating set for a group G with base B is a set of generators S such that

$$G^{(i)} = \langle S \cap G^{(i)} \rangle.$$

This means that S has sufficiently many elements so that it contains generators for each subgroup in the stabilizer chain. Orbiter uses a base for $\mathrm{P\Gamma L}(n,q)$ described in [2]. Notice that the same base for G works for every subgroup of G. For a homomorphic image, we may need to stabilizer chain: one for the group induced and one for the kernel of the action. Since every collineation group is a subgroup of $\mathrm{P\Gamma L}(n,q)$, we can represent every collineation group using the base described in [2]. Interestingly, the base for $\mathrm{P\Gamma L}(n,q)$ described in [2] is very closely related to a frame in the underlying projective space. Since we enumerate points in $\mathrm{PG}(n,q)$ so that the points in a frame come first, this implies that most of the points in the base for $\mathrm{P\Gamma L}(n,q)$ have very small ranks. This is convenient for working with these groups.

Command	Arguments	Group
-GL	n, q	$\mathrm{GL}(n, q)$
-GGL	n, q	$\Gamma\mathrm{L}(n, q)$
-SL	n, q	$\mathrm{SL}(n, q)$
-SSL	n, q	$\Sigma\mathrm{L}(n, q)$
-PGL	n, q	$\mathrm{PGL}(n, q)$
-PGGL	n, q	$\mathrm{P}\Gamma\mathrm{L}(n, q)$
-PSL	n, q	$\mathrm{PSL}(n, q)$
-PSSL	n, q	$\mathrm{P}\Sigma\mathrm{L}(n, q)$
-AGL	n, q	$\mathrm{AGL}(n, q)$
-AGGL	n, q	$\mathrm{A}\Gamma\mathrm{L}(n, q)$
-ASL	n, q	$\mathrm{ASL}(n, q)$
-ASSL	n, q	$\mathrm{A}\Sigma\mathrm{L}(n, q)$

Table 6: Basic types of Orbiter matrix groups

6 Linear Groups

There are many ways to create linear and semilinear groups in Orbiter. The groups are created as matrices over finite fields, together with a suitable permutation representation. The elements of finite fields are represented as integers as described in Section 2.
The creation of linear groups from the command line is done using the

```
-linear <group-description> <optional: modifier> -end
```

option. Several Orbiter applications require a group to be specified in this way. Let us look at the specific commands that are possible. The `<group-description>` starts with the main type, which can be one of the commands listed in Table 6. The executable `linear_group.out` can be used to create a matrix group. The group description can be extended by optional modifiers, listed in Table 7. For instance,

```
linear_group.out -v 3 -linear -PGGL 3 4 -end  \
    -report \
    -sylow
```

creates $\mathrm{P}\Gamma\mathrm{L}(3, 4)$. A report can be found in Appendix D. Because of the option `-sylow`, the report includes information about Sylow subgroups. Let us look at a sporadic simple group. The command

```
linear_group.out -v 2  \
    -linear -PGL 7 11 -Janko1 -end \
    -report
```

Modifier	Arguments	Meaning
-Janko1		first Janko group (needs $\mathrm{PGL}(7, 11)$)
-wedge		action on the exterior square
-PGL2OnConic		induced action of $\mathrm{PGL}(2, q)$ on the conic in the plane $\mathrm{PG}(2, q)$
-monomial		subgroup of monomial matrices
-diagonal		subgroup of diagonal matrices
-null_polarity_group		null polarity group
-symplectic_group		symplectic group
-singer	k	subgroup of index k in the Singer cycle
-singer_and_frobenius	k	subgroup of index k in the Singer cycle, extended by the Frobenius automorphism of $\mathbb{F}_{q^n}$ over $\mathbb{F}_q$
-subfield_structure_action	s	action by field reduction to the subfield of index s
-subgroup_from_file	f l	read subgroup from file f and give it the label l
-borel_subgroup_upper		Borel subgroup of upper triangular matrices
-borel_subgroup_lower		Borel subgroup of lower triangular matrices
-identity_group		identity subgroup
-on_k_subspaces	k	induced action on k dimensional subspaces
-orthogonal	ϵ	orthogonal group O^ϵ, with $\epsilon \in \{\pm 1\}$ when n is even
-subgroup_by_generators	l o n str(1) ... str(n)	Generate a subgroup from generators. The label "l" is used to denote the subgroup; o is the order of the subgroup; n is the number of generators and str(1), ..., str(n) are the generators for the subgroup in string representation.

Table 7: Modifiers for creating matrix groups

Nice generators:

$$\begin{bmatrix} 1 & 1 & 4 \\ 6 & 8 & 1 \\ 7 & 5 & 8 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 \\ 1 & 7 & 7 \\ 5 & 6 & 3 \end{bmatrix}$$

Group action $\mathrm{PGL}(3, 11)$ of degree 133
Group order 21
tl=$7, 3, 1, 1,$
Base: $(0, 1, 2, 3)$
Strong generators for a group of order 21:

$$\begin{bmatrix} 1\,0\,0 \\ 1\,7\,7 \\ 5\,6\,3 \end{bmatrix}, \begin{bmatrix} 1\,1\,4 \\ 6\,8\,1 \\ 7\,5\,8 \end{bmatrix}$$

1,0,0,1,7,7,5,6,3,
1,1,4,6,8,1,7,5,8,

Table 8: The Group generated by a power of a Singer cycle and a Frobenius automorphism

creates the first Janko group as a subgroup of $\mathrm{PGL}(7, 11)$. A latex report is shown in Appendix C. Let us look at another group. The Singer subgroup in $\mathrm{GL}(n, q)$ is a subgroup of order $(q^n - 1)$ acting transitively on the nonzero vectors of $\mathbb{F}_q^n$. The image in $\mathrm{PGL}(n, q)$ is a cyclic group of order $(q^n - 1)/(q - 1)$ acting transitively on the points of the associated projective space. We consider the Singer subgroup of $\mathrm{PGL}(3, 11)$. This is a cyclic subgroup of order 133. We consider the 19th power of the Singer cycle, together with the Frobenius automorphism for $\mathbb{F}_{11^3}$ over $\mathbb{F}_{11}$, to generate a group of order 21. The following command can be used to create this group.

```
linear_group.out -v 3 -linear -PGL 3 11 \
    -singer_and_frobenius 19 -end  \
    -report
```

Table 8 shows the report generated for this group of order 21. Orbiter, through its interface to Magma [5], can compute the conjugacy classes of groups. For instance, the command

```
linear_group.out -v 6 -linear -PSL 3 2 \
    -end  -classes
```

can be used to create a report about the conjugacy classes of the simple group $\mathrm{PSL}(3, 2)$. The report is shown in Appendix E.

It is possible to use the group that was created to do other tasks as described in Table 9.

It is possible to create groups and subgroups using generators directly from the command line. For instance, it is known that the quaternion group is generated by the following

Modifier	Arguments	Meaning
-orbits_on_subsets	k	Compute orbits on k-subsets
-orbits_on_points		Compute orbits in the action that was created
-orbits_of	i	Compute orbit of point i in the action that was created
-stabilizer		Compute the stabilizer of the orbit representative (needs -orbits_on_points)
-draw_poset		Draw the poset of orbits (needs -orbits_on_subsets)
-classes		Compute a report of the conjugacy classes of elements (needs Magma [5])
-normalizer		Compute the normalizer (needs Magma [5]; needs a group with a subgroup)
-report		Produce a latex report about the group
-sylow		Include Sylow subgroups in the report (needs -report)
-print_elements		Produce a printout of all group elements
-print_elements_tex		Produce a latex report of all group elements
-group_table		Produce the group table (needs -report)
-orbits_on_set_system_from_file	fname f l	reads the csv file "fname" and extract sets from columns $[f, ..., f + l - 1]$
-orbit_of_set_from_file	fname	reads a set from the text file "fname" and computes orbits on the elements of the set
-multiply	str1 str2	Creates group elements from str1 and str2 and multiplies
-inverse	str	Creates a group element from str and computes its inverse

Table 9: Tasks that can be performed for a group

Element 0 / 8 of order 1:

$$\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

Element 1 / 8 of order 4:

$$\begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix}$$

Element 2 / 8 of order 2:

$$\begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$$

Element 3 / 8 of order 4:

$$\begin{bmatrix} 1 & 2 \\ 2 & 2 \end{bmatrix}$$

Element 4 / 8 of order 4:

$$\begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix}$$

Element 5 / 8 of order 4:

$$\begin{bmatrix} 0 & 1 \\ 2 & 0 \end{bmatrix}$$

Element 6 / 8 of order 4:

$$\begin{bmatrix} 2 & 2 \\ 2 & 1 \end{bmatrix}$$

Element 7 / 8 of order 4:

$$\begin{bmatrix} 0 & 2 \\ 1 & 0 \end{bmatrix}$$

Table 10: The elements of the quaternion group inside $\mathrm{SL}(2,3)$

generators (taken from Wikipaedia):

$$i = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}, \quad j = \begin{bmatrix} -1 & 1 \\ 1 & 1 \end{bmatrix}, \quad k = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$$

as a subgroup of $\mathrm{SL}(2,3)$. The Orbiter command

```
linear_group.out -v 3 -linear -SL 2 3 \
    -subgroup_by_generators "quaternion" "8" 3 \
    "1,1,1,2" \
    "2,1,1,1" \
    "0,2,1,0" \
    -end \
    -print_elements_tex \
    -group_table \
    -report
```

creates the group. Notice that -1 must be written as 2, considering the remarks about the representation of field elements in Section 2, recalling the fact that we are in $\mathbb{F}_3$. The command produces the list of group elements shown in Table 10. The group table is shown in Table 11.

There are two ways to create the orthogonal group. First, we can create them as subgroups of the associated general linear groups. This will create the action on the projective space.

0	1	2	3	4	5	6	7
1	2	3	0	5	6	7	4
2	3	0	1	6	7	4	5
3	0	1	2	7	4	5	6
4	7	6	5	2	1	0	3
5	4	7	6	3	2	1	0
6	5	4	7	0	3	2	1
7	6	5	4	1	0	3	2

Table 11: The group table of the quaternion group

The `orthogonal_group.out` application can be used if the action on the singular points is desired. For instance,

```
orthogonal_group.out -v 2 -epsilon 1 -d 6 -q 2 -report
```

creates $\mathrm{PGO}^+(6,2)$, including the report shown in Appendix F.

Sometimes, the generators depend on specific choices made for the finite field. For instance, if the field if a true extension field over its prime field, the choice of the polynomial matters. This is particularly relevant if generators are taken from other sources. For instance, the electronic Atlas of finite simple groups [21] lists generators for $U_3(3)$ as 3×3 matrices over the field $\mathbb{F}_9$ using the following short Magma [5] program:

```
F<w>:=GF(9);
x:=CambridgeMatrix(1,F,3,[
"164",
"506",
"851"]);
y:=CambridgeMatrix(1,F,3,[
"621",
"784",
"066"]);
G<x,y>:=MatrixGroup<3,F|x,y>;
```

The generators are given using the Magma command `CambridgeMatrix`, which allows for more efficient coding of field elements. The field elements are coded as base-3 integers (like in Orbiter) with respect to the Magma version of $\mathbb{F}_9$. Magma uses Conway polynomials to generate finite fields which are not of prime order. The Conway polynomial for $\mathbb{F}_9$ can be determined using the following Magma command (which can be typed into the Magma online calculator at [20])

```
F<w>:=GF(9);
print DefiningPolynomial(F);
```

which results in

```
$.1^2 + 2*$.1 + 2
```

which is the Magma way of printing the polynomial $X^2 + 2X + 2$. To have Orbiter use this polynomial, the `-override_polynomial` option can be used. First, the polynomial is identified with the vector of coefficients $(1, 2, 2)$ which is then read as base-3 representation of an integer as

$$(1, 2, 2) = 1 \cdot 3^2 + 2 \cdot 3 + 2 = 17.$$

The Orbiter command

```
linear_group.out -v 3 -linear -override_polynomial "17" -PGL 3 9 \
    -subgroup_by_generators "U_3_3" "6048" 2 \
    "1,6,4, 5,0,6, 8,5,1" \
    "6,2,1, 7,8,4, 0,6,6" \
    -end  \
    -report
```

can then be used to create the group. Notice how the generators are encoded almost like in the Magma command, except that commas are used to separate entries. The Orbiter report for this group in shown in Appendix G.

For a slightly more challenging example, let us create the group Co_3 (Conway's third group). The group is a subgroup of $PGL(22, 2)$. We use the generators found in [19]. The command has been reformatted slightly. Each matrix should be written in one row.

```
linear_group.out -v 3 -linear -PGL 22 2 \
    -subgroup_by_generators "Co3" "495766656000" 2 \
    "1,1,0,1,1,1,0,0,0,1,0,0,0,0,0,1,0,1,0,0,0,0,
    1,1,1,1,0,1,0,1,1,1,1,1,0,1,0,0,0,0,1,0,1,1,
    0,0,0,0,0,0,1,0,0,0,0,0,0,1,0,0,0,1,0,1,0,1,
    1,1,1,1,1,0,0,1,1,0,1,1,0,0,0,1,0,0,1,1,1,0,
    0,1,0,1,0,1,0,0,0,0,0,0,0,0,1,0,0,1,1,1,0,1,
    0,0,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,1,0,1,0,1,
    0,0,1,0,0,0,0,0,0,0,0,0,0,1,0,0,0,1,0,1,0,1,
    0,0,0,1,0,0,0,0,1,1,0,0,0,0,0,0,1,1,1,1,1,1,
    1,1,1,0,1,0,0,1,0,0,1,1,0,1,0,0,0,1,0,0,1,1,
    0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,1,0,1,0,1,
    0,0,0,0,0,0,0,0,0,0,1,0,0,1,0,0,0,1,0,1,0,1,
    0,1,1,0,1,1,1,1,1,1,0,1,0,0,1,1,1,0,1,1,1,1,
    0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,1,0,1,0,1,
    0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,1,0,1,
    0,0,0,0,0,0,0,0,0,1,0,0,0,1,0,0,0,1,0,1,0,1,
    0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,1,1,1,0,1,
    0,0,0,1,0,0,0,1,1,0,0,0,0,0,1,0,0,1,1,0,1,0,
    0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,1,0,1,
    0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,1,0,1,0,1,0,1,
    0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,1,0,1,0,0,
    0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,1,0,1,1,1,
    0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,1,0,0,0,1," \
```

```
"0,1,0,1,0,0,0,0,1,0,1,1,1,0,1,0,1,1,1,1,1,1,
0,1,1,0,0,1,0,1,0,0,0,1,1,1,1,0,1,1,0,0,0,0,
0,0,1,1,0,1,0,0,0,0,1,1,1,1,1,1,0,1,0,1,1,1,
0,0,0,1,1,0,1,1,1,0,0,0,1,0,1,1,0,1,0,0,1,1,
1,0,1,0,0,1,0,0,0,0,1,0,0,0,0,1,0,1,1,1,1,0,
1,1,0,1,0,0,0,0,0,0,0,0,1,0,1,0,1,0,0,0,1,1,
1,1,0,0,1,0,1,0,1,0,0,0,1,1,1,1,0,1,0,1,0,1,
1,0,0,0,1,1,0,1,0,0,1,1,0,1,0,1,0,1,0,1,0,1,
0,1,0,0,1,1,0,0,0,1,0,1,0,0,0,0,0,0,0,1,1,1,
1,1,0,0,0,0,0,0,1,0,1,0,0,1,0,1,0,1,0,0,1,0,
0,1,0,1,1,1,0,1,1,0,0,1,1,1,0,0,0,0,0,1,0,1,
0,1,0,1,1,1,1,1,0,1,0,1,0,0,1,1,1,1,1,0,0,1,
1,0,0,0,0,1,0,1,0,1,0,1,0,1,0,1,0,1,0,0,0,1,
0,0,0,1,0,1,0,0,0,0,1,1,1,1,0,0,1,0,0,1,1,1,
0,0,1,1,0,1,0,0,1,0,1,1,1,0,1,1,0,0,1,1,1,1,
0,1,0,0,1,1,0,0,1,0,1,1,0,0,1,1,1,1,1,0,1,0,
1,1,0,1,0,1,1,0,0,1,1,1,1,1,0,1,1,0,0,0,1,1,
0,1,0,0,1,0,1,0,0,1,0,0,1,0,0,0,1,0,0,0,0,1,
1,1,0,0,1,0,1,1,0,0,0,0,1,0,0,1,1,1,0,0,1,1,
0,1,0,1,1,1,0,1,1,0,0,1,0,1,0,0,0,0,0,0,0,1,
0,0,0,0,0,0,1,1,0,1,1,1,1,0,0,0,1,0,1,1,1,0,
1,1,0,1,1,0,1,0,1,0,1,0,1,1,1,0,0,0,0,1,0,1," \
-end
```

The group is created in about 35 seconds. The lengths of the basic orbits are:

$$37950,\ 324,\ 56,\ 45,\ 16.$$

The orbit length distribution is the following:

$$11178,\ 37950,\ 1536975,\ 2608200.$$

This shows that there are two short orbits and two long orbits. By luck, we picked one of the short orbits for the first basic orbit. This was good because that way the stabilizer chain has many short orbits. This makes the stabilizer chain more efficient.

7 Orbits on subspaces

The `subspace_orbits_main.out` application computes the orbits of a group on the lattice of subspaces of a finite vector space.

Suppose we want to classify the subspaces in $\mathrm{PG}(3,2)$ under the action of the orthogonal group. The orthogonal group is the stabilizer of a quadric. In $\mathrm{PG}(3,2)$ there are two distinct nondegenerate quadrics, $\mathcal{Q}^+(3,2)$ and $\mathcal{Q}^-(3,2)$. The $\mathcal{Q}^+(3,2)$ quadric is a finite version of the quadric given by the equation

$$x_0x_1 + x_2x_3 = 0,$$

and depicted over the real numbers in Figure 6. $\mathrm{PG}(3,2)$ has 15 points:

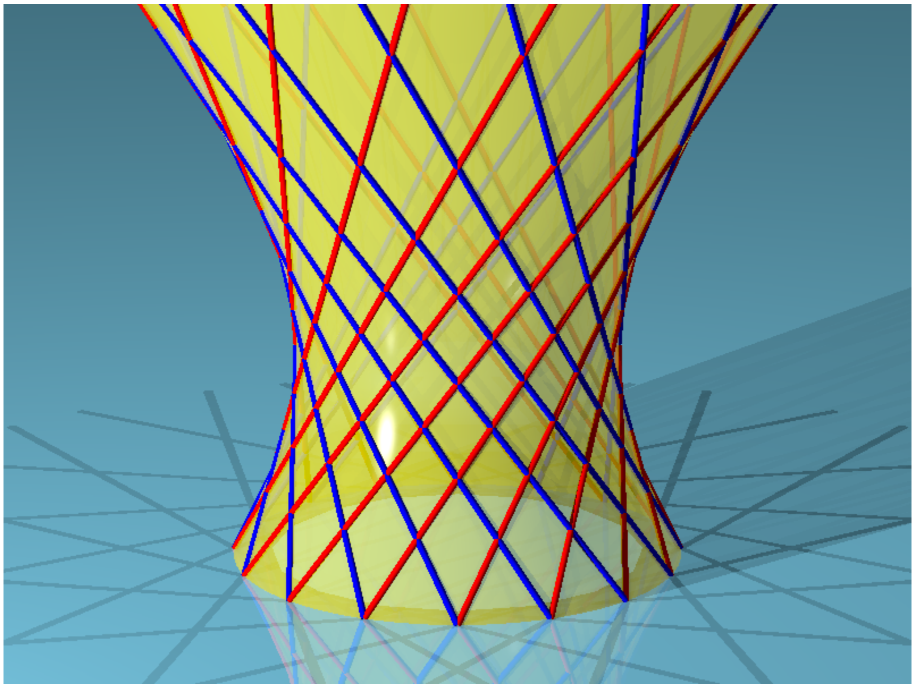


Figure 6: The hyperbolic quadric in affine space $\mathbb{R}^3$

$P_0 = (1,0,0,0)$	$P_4 = (1,1,1,1)$	$P_8 = (1,1,1,0)$	$P_{12} = (0,0,1,1)$
$P_1 = (0,1,0,0)$	$P_5 = (1,1,0,0)$	$P_9 = (1,0,0,1)$	$P_{13} = (1,0,1,1)$
$P_2 = (0,0,1,0)$	$P_6 = (1,0,1,0)$	$P_{10} = (0,1,0,1)$	$P_{14} = (0,1,1,1)$
$P_3 = (0,0,0,1)$	$P_7 = (0,1,1,0)$	$P_{11} = (1,1,0,1)$	

The $\mathcal{Q}^+(3,2)$ quadric given by the equation above consists of the nine points

$$P_0, P_1, P_2, P_3, P_4, P_6, P_7, P_9, P_{10}.$$

The quadric is stabilized by the group $\mathrm{PGO}^+(4,2)$ of order 72. The command

```
subspace_orbits_main.out -v 5 \
    -depth 4 -group -PGL 4 2 -orthogonal 1 -end \
    -draw_poset -embedded \
```

produces a classification of all subspaces of $\mathrm{PG}(3,2)$ under $\mathrm{PGO}^+(4,2)$. A Hasse diagram of the classification is shown in Figure 7. Let us try to understand this output a little bit. Every node stands for one isomorphism class of orbits of the orthogonal group on subspaces. The number before the semicolon refers to the orbit representative at that node. The number after the semicolon gives the order of the stabilizer of the node. The node at the top represents the zero subspace, with a stabilizer of order 72 (the full group). Every node below this represents a non-trivial subspace. Each subspace is described using the numerical representation of the basis elements, according to the labeling of points that was given above. In order to make

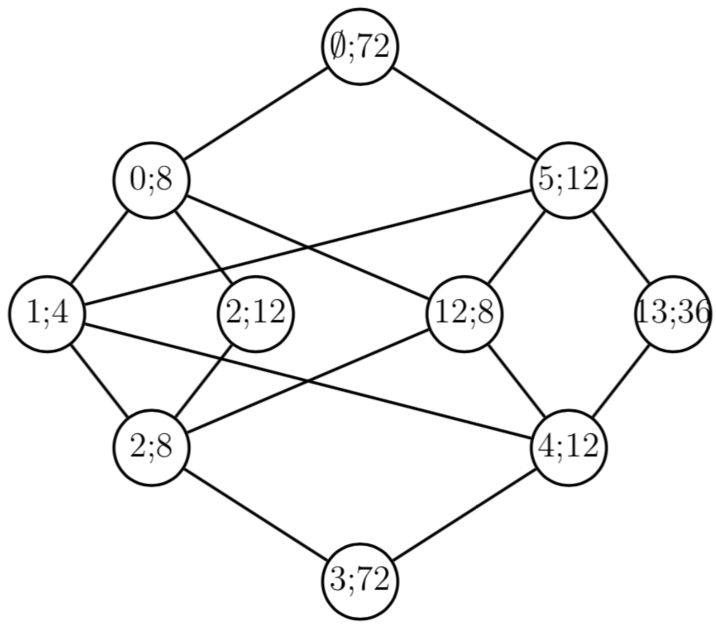


Figure 7: Hasse-diagram of the types of subspaces of $\mathrm{PG}(3,2)$

the presentation more compact, only the index of the last of the basis vectors is listed at each node. The other basis vectors can be recovered by following the leftmost path to the root. For instance, the node at the very bottom is labeled by 3, representing P_3. The other basis elements are P_0, P_1, P_2 because $0, 1, 2$ are the labels encountered along the unique leftmost path to the root. Since $P_0, \ldots, P_3$ represent the four unit vectors, it is clear that the bottom node represents the whole space $\mathrm{PG}(3,2)$. The stabilizer is the full group, of order 72. The two nodes at level one represent the two types of points. P_0 represents points on the quadric (with a point stabilizer of order 9), and P_5 represents the points off the quadric (with a point stabilizer of order 12). The middle node has 4 orbits. Reading left to right, these nodes represent the following orbits on lines:

(a) Secant lines. Such lines have two points on the quadric and $q-1$ points off the quadric. A representative is the line P_0P_1. These lines give rise to hyperbolic pairs.

(b) Totally isotropic lines. These are lines contained in the quadric (these correspond to the colored lines in Fig. 6). A representative is the line P_0P_2.

(c) Tangent lines. Such lines have exactly one point on the quadric. A representative is the line P_0P_{12}.

(d) External lines. Such lines contain no quadric point. A representative is the line P_5P_{13}.

There are two types of planes:

(a) Planes which intersect the quadric in two totally isotropic lines. A representative is the plane $P_0P_1P_2$.

Application	Purpose
all_cliques.out	Finds all cliques in a graph
all_cycles.out	Finds all cycles in a graph
all_rainbow_cliques.out	Finds all rainbow-cliques in a vertex colored graph
cayley.out	Computes Cayley graphs
cayley_sym_n.out	Computes Cayley graphs of Sym(n)
colored_graph.out	Exports Orbiter colored graphs to Maple [15] or Magma [5]
create_graph.out	Create an Orbiter graph using command line arguments
create_layered_graph_file.out	Create Orbiter layered graph file from previously computed poset data
draw_colored_graph.out	Draws a clored graph; can perform other tasks as well
draw_graph.out	Draws a graph
graph.out	Classifies graphs and tournaments using poset classification
johnson_table.out	Creates a table of Johnson graph parameters
layered_graph_main.out	draw posets
nauty.out	Simple interface to Nauty [16]
rainbow_cliques.out	Finds all rainbow-cliques in a vertex colored graph
srg.out	Creates a table of parameters of small strongly regular graphs
treedraw.out	Draws trees from previously created tree files

Table 12: Orbiter Applications for Graph Theory

(b) Planes which intersect the quadric in a conic. A representative is the plane $P_0P_1P_4$.

8 Graph Theory

Many applications in Orbiter are devoted to graph theory. They are listed in Table 12.
For instance, the command

```
cayley_sym_n.out -v 1 -n <n> -coxeter
```

creates the Cayley graph on Sym(n) with respect to the Coxeter generators. The graphs for Sym(4) and Sym(5) are shown in Figure 8. The drawings were created using the command

```
draw_colored_graph.out -v 1 -file Cayley_Sym_4_coxeter.colored_graph
    -aut -on_circle  -embedded -scale 0.25 -line_width 0.5
```

For these drawings, the elements in the groups are totally ordered according to the indexing associated with a chosen stabilizer chain. In each case, the base is the sequence of integers $0, \dots, n-1$ where $n = 4, 5$, respectively.

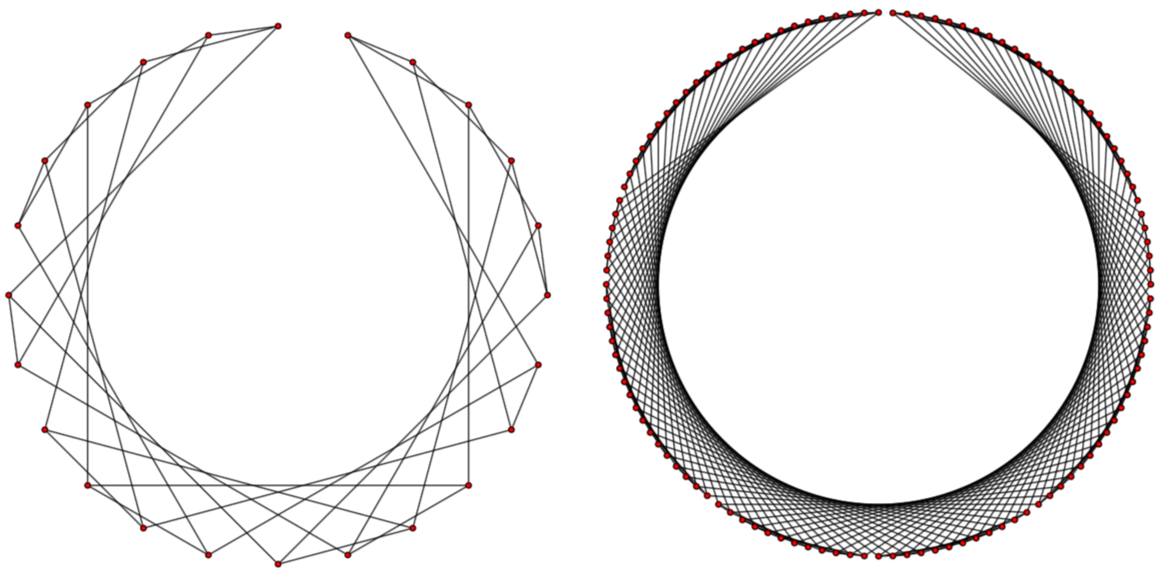


Figure 8: Cayley graphs for Sym(4) and Sym(5)

The executable `create_graph.out` can be used to create a graph. The description of the graph is through command line options, using keywords as listed in Table 13. For instance,

```
create_graph.out -v 2 -save graph_J520.bin -graph -Johnson 5 2 0 -end
```

creates $J(5,2,0)$, also known as the Petersen graph.

```
create_graph.out -v 2 -save graph_P13.bin -graph -Paley 13 -end
```

creates the Paley graph of order 13.

Key	Arguments	Meaning
-Johnson	$n\ k\ s$	Johnson graph
-Paley	q	Paley graph
-Sarnak	$p\ q$	Lubotzky-Phillips-Sarnak graph [14]
-Schlaefli	q	Schlaefli graph
-Shrikhande		Shrikhande graph
-Winnie_Li	$q\ i$	Winnie-Li graph [13]
-Grassmann	$n\ k\ q\ r$	Grassmann graph

Table 13: Types of graphs

9 Coding Theory

A central problem in coding theory is to determine the set of inequivalent optimal linear codes. Orbiter can help with this. Let us discuss this problem in some more details.

A linear $[n,k]$-code $\mathcal{C}$ over $\mathbb{F}_q$ is a k-dimensional subspace of $\mathbb{F}_q^n$. The code is said to have minimum distance d if

$$\min_{\substack{c,c' \in \mathcal{C} \\ c \neq c'}} d(c,c') = d$$

where $d(\mathbf{x},\mathbf{y})$ is the Hamming matric on $\mathbb{F}_q^n$, which counts the number of positions where $\mathbf{x}$ and $\mathbf{y}$ differ. A code code has both k and d large with respect to n. There are theoretical bounds for what can be achieved. However, achieving this or coming close to it is often challenging. The notion of isometry with respect to the Hamming metric leads to a notion of equivalence of codes. Two codes are equivalent if the coordinates of the vectors in one code can be computed (simultaneously) so as to obtain the second code. The automorphism group is the set of isometry maps from one code to itself.

The classification problem of optimal codes in coding theory is the problem of determining the equivalence classes of codes for a given set of values of n and k with a lower bound on d. We wish to use Orbiter for solving this problem for small instances.

Orbiter reduces the problem of classifying $[n,k,\geq d]$ codes over $\mathbb{F}_q$ to an equivalent problem in finite geometry. According to [2], the equivalence classes of $[n,k,\geq d]$ codes over $\mathbb{F}_q$ for $d \geq 3$ are in canonical one-to-one correspondence to the sets of size n in $\mathrm{PG}(n,k-1,q)$ with the property that any set of size at most $d-1$ is linearly dependent. Let $\Lambda_{m,s}(q)$ be the the poset of subsets of $\mathrm{PG}(m,q)$ such that any set of s or less points is independent. The group $G = \mathrm{P\Gamma L}(m+1,q)$ acts on this poset. For $m = n-k-1$ and $s = d-1$, the orbits of G on sets in $\Lambda_{m,s}(q)$ of size n are in canonical one-to-one correspondence to the $[n,k,\geq d]$ codes over $\mathbb{F}_q$.

The Orbiter command

```
codes.out -v 2 -n <n> -k <k> -q <q> -d <d> -lex
```

can be used to classify the $[n,k,\geq d]$ codes over $\mathbb{F}_q$. For instance, the command

```
codes.out -v 2 -n 8 -k 4 -q 2 -d 4 -lex
```

classifies the $[8,4,\geq 4]$ codes over $\mathbb{F}_2$. It turns out that there is exactly one such code, the $[8,4,4]$ code known as the extended Hamming code. Using the group $\mathrm{PGL}(4,2)$ acting on the poset $\Lambda_{3,3}(2)$, Orbiter produced the poset of orbits shown in Figure 9. In this diagram, the numbers stand for Orbiter numbers of points in $\mathrm{PG}(3,2)$. All nodes except for the root node have a number attached to it. The node represent subsets. In order to determine the set associated to a node, follow the path from the root node to the node and collect the points according to their labels. The root node represents the empty set. The $[8,4,4]$-code is represented by the set $\{0,1,2,3,8,11,13,14\}$. The fact that there is only one node at level 8 in the poset of orbits tells us that the code is unique up to equivalence. Orbiter also produces a report about the classification. For this, the somewhat more complicated command

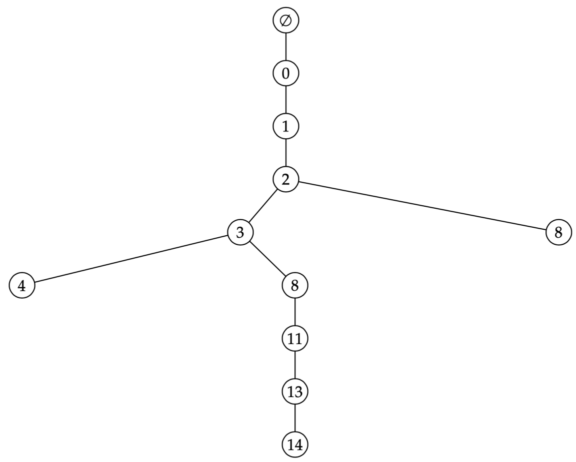


Figure 9: Orbits of PGL$(4, 2)$ on the poset $\Lambda_{3,3}(2)$,

```
codes.out -v 2 -n 8 -k 4 -q 2 -d 4 -w -lex \
    -draw_poset \
    -export_schreier_trees \
    -tools_path <path to Orbiter applications> \
    -report \
    -report_schreier_trees
    latex codes_linear_n8_k4_q2_d4.tex
    dvips codes_linear_n8_k4_q2_d4.dvi -o
    open codes_linear_n8_k4_q2_d4.ps
```

can be used (the last command is Macintosh specific; it opens the postscript file on screen). The `<path to Orbiter applications>` need to be replaced by the path to the Orbiter applications. The report can be seen in Appendix H. Let us look at the code. The elements of the set $\{0, 1, 2, 3, 8, 11, 13, 14\}$ are points in PG$(3, 2)$. The point labeling for PG$(3, 2)$ is shown in Appendix H. We write the coordinate vectors in the columns of a matrix H like so:

$$H = \begin{bmatrix} 1 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 \end{bmatrix}.$$

This matrix is the parity check matrix H of the code $\mathcal{C}$. This means that the words of the code are the vectors c such that $c \cdot H^\top = 0$. Observe that the vectors that we put in the columns of H all have odd weight. They are in fact the points of the hyperplane $x + y + z + w = 0$. This shows that the stabilizer of the code which is the stabilizer of the set is equal to AGL$(3, 2)$, a group of order 1344.

Application	Purpose
analyze_projective_code.out	Examine properties of a projective code
andre.out	Creates translation planes using the André [1] and Bruck-Bose [7] construction and examines properties
canonical_form.out	Computes the canonical forms of objects in $\mathrm{PG}(n, q)$ together with their stabilizers through graph canonization using Nauty [16]
cheat_sheet_PG.out	Creates a report about $\mathrm{PG}(n, q)$, see Section 3
classify_cubic_curves.out	Classifies cubic curves in $\mathrm{PG}(2, q)$
desarguesian_spread.out	Creates the dearguesian spread
determine_conic.out	Determines the equation of a conic in $\mathrm{PG}(2, q)$ given 5 points
determine_cubic.out	Determines the equation of a cubic in $\mathrm{PG}(2, q)$ given 9 points
determine_quadric.out	Determines the equation of a quadric in $\mathrm{PG}(3, q)$ given 9 points
hermitian_points.out	Ranking and unranking of points on the Hermitian surface
hermitian_spreads_main.out	Classification of Hermitian spreads
orthogonal_points.out	Ranking and unranking of points on a quadric
points.out	Ranking and unranking of points
polar.out	Creates the maximal singular subspaces in orthogonal polar spaces
process.out	File processing of points sets in $\mathrm{PG}(n, q)$

Table 14: Orbiter Applications for Projective Geometry

10 Projective Geometry

Many applications in Orbiter are devoted to problems in projective geometry. Table 14 contains a list of these.

The `canonical\_form.out` application can be used to classify sets in projective space and to compute their collineation stabilizer. The application gets its input from either files or the command lines using the -input argument. Here is an example. We consider the following three Orbiter commands:

```
create_object.out -v 5 -object -q 11 \
    -elliptic_curve 1 3 -end \
    -save E_b1_c3.txt
process.out -v 2 -job \
    -draw_points_in_plane EC_11_1_3 -q 11 -n 2 \
    -fname_base_out EC_11_1_3 -embedded \
    -input -file_of_points E_b1_c3.txt \
    -end \
```

```
    -end
canonical_form.out -v 10 -n 2 -q 11 \
    -input \
    -file_of_points E_b1_c3.txt \
    -end \
    -classify_nauty \
    -prefix PG_2_11_EC \
    -save PG_2_11_EC_classified \
    -report
```

The first command creates an elliptic curve over a finite field. Specifically, it creates the curve whose affine equation is

$$y^2 \equiv x^3 + x + 3 \bmod 11.$$

The curve turns out to have exactly 18 points in $\mathrm{PG}(2, 11)$. The second command produces a picture of the point set in $\mathrm{PG}(2, 11)$, shown in Figure 3. The third command invokes the `canonical_form.out` application to compute the collineation stabilizer. This is done by turning the plane into a graph, encoding the set of points by modifying the graph, and computing the canonical form and automorphism group of the graph using Nauty [16]. The number of vertices is twice the number of points in the plane plus two. So, in this example the graph has $2(133+1) = 268$ vertices. The application shows that the curve has a collineation stabilizer of order 6, generated by

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 10 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 1 & 8 \\ 5 & 9 & 5 \\ 8 & 1 & 1 \end{bmatrix}.$$

The full report is shown in Appendix I.10.

11 Cubic Surfaces

Orbiter can classify cubic surfaces with 27 lines over finite fields. There are several different approaches to classify cubic surfces over finite fields with 27 lines. One approach is described in [4] and relies on Schlaefli's notion of a double six as a substructure [17]. Another approach is through non-conical six-arcs in a plane, as described in [12]. Both approaches have been implemented in Orbiter. It is not clear which approach is best, but at the moment the one with double sixes seems to perform better, so we will look at it next. The whole purpose of the construction algorithm is to produce the equations of surfaces. In order to do so, the notion of double sixes of lines in $\mathrm{PG}(3, q)$ is helpful. A double six determines a unique surfaces but a surface may have several double sixes associated to it. The classification algorithms sorts out the relationship between the isomorphism types of double sixes and the isomorphism types of cubic surfaces. In order to classify all double sixes, yet another substructure is considered. These are the five-plus-ones. They consist of 5 lines with a common transversal. Orbiter's poset classification algorithm is used to classify the five-plus-ones. Also, Orbiter will sort out the isomorphism classes of double sixes based on their relation to the five-plus-ones. In order to classify the five-plus-ones, the related Klein quadric is considered. Lines in $\mathrm{PG}(3,q)$ correspond to points on the Klein quadric. Thus, the five-plus-one configurations of lines correspond to certain configurations of points on the Klein quadric.

Let us look at the suitable Orbiter command. We fix a finite field $\mathbb{F}_q$ of order q. We issue the command

```
surface_classify.out -v 2 -linear -PGGL 4 <q> -wedge -end
```

with `<q>` replaced by the actual value of q. The Orbiter application `surface_classify.out` will classify the surfaces with 27 lines over the field $\mathbb{F}_q$ with respect to the collineation group $\mathrm{P\Gamma L}(4,q)$. If desired, it is possible to use

```
surface_classify.out -v 2 -linear -PGL 4 <q> -wedge -end
```

to perform the same classification with respect to the projectivity group $\mathrm{PGL}(4,q)$. For instance, if q is prime, there is no reason to ask for $\mathrm{P\Gamma L}(4,q)$ since the two groups are isomorphic and the group $\mathrm{PGL}(4,q)$ is more efficient. The algorithm produces a set of representatives of the isomorphism types of cubic surfaces with 27 lines in $\mathrm{PG}(3,q)$. It does to by picking an equation for each representative of an isomorphism type of surfaces. Unfortunatley, the equations that are chosen by the classification algorithm to represent the isomorphim types of surfaces are not very revealing to humans. This is because Orbiter's poset classification algorithm picks the surface by picking configurations of lines in $\mathrm{PG}(3,q)$ and lines in turn are chosen using the lexicographic ordering on subsets based on the enumerator for lines. The lines are labeled using an indexing function of the wedge product $\bigwedge V$ where $V \simeq \mathbb{F}_q^4$ is the vector space underlying $\mathrm{PG}(3,q)$. The indexing of the elements of the wedge product $\bigwedge \mathbb{F}_q^4$ depends on the indexing of the points on the $Q^+(5,q)$ quadric, because $\bigwedge \mathbb{F}_q^4$ and $Q^+(5,q)$ correspond in a canonical way. Because $\mathrm{PGL}(4,q)$ acts transitively on the lines of $\mathrm{PG}(3,q)$, the first line can be chosen arbitrarily. Orbiter picks the line

$$\ell_0 = \mathbf{L}\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

to be the first line in a five-plus-one configuration. The remaining five lines are supposed to intersect this line. For this, the stabilizer of the line ℓ_0 is considered in the action on the lines which intersect ℓ_0. This is an instance of a poset classification problem. The group of the stabilizer of ℓ_0, and the set of the set of subsets of size at most 5 of all pairwise disjoint lines intersecting ℓ_0.

Let us consider an example. The command

```
surface_classify.out -v 2 -linear -PGGL 4 4 -wedge -end
```

classifies the cubic surfaces over the field $\mathbb{F}_4$ under the action of the collineation group $\mathrm{P\Gamma L}(4,4)$. Orbiter shows that there is exactly one such surface. There are multiple output files of the surface classification program. For the case $q = 4$, the following files are generated (for different values of q, the files change accordingly):

(a) `neighbors_4.csv` This file contains a list of all lines in $\mathrm{PG}(3,q)$ which intersect the line $\mathbf{L}\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$. Five of these lines are chosen to form a five-plus-one together with ℓ_0.

(b) `fiveplusone_4.csv` contains a summary of the poset classification of five-plus-one configurations. The indexing of lines in the file is the same as the one shown in the file `neighbors_4.csv`.

(c) `Double_sixes_q4.data` is a binary file which contains the classification of double sixes in $\mathrm{PG}(3,4)$.

(d) `Double_sixes_q4.tex` is a latex file which reports the classification of five-plus-ones and the classification of double sixes in human readable format.

(e) `Surfaces_q4.data` is a binary file which contains the classification of surfaces with 27 lines in $\mathrm{PG}(3,q)$ (here, $q=4$).

(f) `surface_4.cpp` is a C++ source code file which contains the data about the classification in a form suitable for inclusion in the Orbiter source tree. In fact, this file has already been included into Orbiter.

(g) `memory_usage.csv` is a file which records the time and memory used during execution of the program `surface_classify.out`.

The `-report` option can be used to create a report of the classified surfaces. So, for instance

```
surface_classify.out -v 2 -linear -PGGL 4 4 -wedge -end -report
```

produces a latex report of the surface in $\mathrm{PG}(3,4)$. In this example, the file `Surfaces_q4.tex` will be created. The `-recognize` option can be used to identify a given surface in the list produced by the classification. For instance,

```
surface_classify.out -v 2 -linear -PGGL 4 8 -wedge -end \
    -recognize -q 8 -by_coefficients "1,6,1,8,1,11,1,13,1,19" -end
```

identifies the surface (cf. Table 5)

$$X_0^2X_3 + X_1^2X_2 + X_1X_2^2 + X_0X_3^2 + X_1X_2X_3 = 0 \tag{4}$$

in the classification of surfaces over the field $\mathbb{F}_8$. This means that an isomorphism from the given surface to the surface in the list is computed. Also, the generators of the automorphism group of the given surface are computed, using the known generators for the automorphism group of the surface in the classification. For instance, executing the command above yields a group of order 576 generated by the following elements. Recall that we use the notation for collineations introduced in (1) in Section 3.

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}_2, \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix}_2, \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \alpha^5 & 0 & 1 & 0 \\ 0 & \alpha^3 & 0 & 1 \end{bmatrix}_0,$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{bmatrix}_1, \begin{bmatrix} 1 & 0 & 0 & 0 \\ \alpha^6 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & \alpha & 0 & 1 \end{bmatrix}_1$$

The isomorphism to the surface number 0 in the classified list is given as

$$\begin{bmatrix} 1 & 4 & 4 & 0 \\ 6 & 0 & 0 & 0 \\ 6 & 2 & 0 & 1 \\ 7 & 0 & 4 & 0 \end{bmatrix}_0 . \tag{5}$$

Besides classification, there are two further ways to create surfaces in Orbiter. The first is a built-in catalogue of cubic surfaces with 27 lines for small finite fields $\mathbb{F}_q$ (at the moment, $q \leq 97$ is required). The second is a way of creating members of known infinite families. Both are facilitated using the `create_surface_main.out` command. For instance,

```
create_surface_main.out -v 2 -description -family_S 3 -q 13 -end
```

creates the member of the Hilbert-Cohn/Vossen surface described in [4] with parameter $a = 3$ and $b = 1$ over the field $\mathbb{F}_{13}$. The command

```
create_surface_main.out -v 2 -description -q 4 -catalogue 0 -end
```

creates the unique cubic surface with 27 lines over the field $\mathbb{F}_4$ which is stored under the index 0 in the catalogue. It is possible to apply a transformation to the surface created by the `create_surface_main.out` command. Suppose we are interested in the surface over $\mathbb{F}_8$ created in (4). We know that this surface can be mapped to the surface number 0 in the catalogue of cubic surfaces over $\mathbb{F}_8$ by the group element in (5). It is possible to create surface 0 over $\mathbb{F}_8$ using the `create_surface_main.out` command, and to apply the inverse transformation to recover the surface whose equation was given in (4). For instance, the command

```
create_surface_main.out -v 2 -description -q 8 -catalogue 0 -end \
    -transform_inverse "1,4,4,0,6,0,0,0,6,2,0,1,7,0,4,0,0"
```

does exactly that. The surface number 0 over $\mathbb{F}_8$ is created, and the transformation (5) is applied in reverse. Notice how the command `-transform_inverse` accepts the transformation matrix in row-major ordering, with the field automorphism as additional element. The purpose of doing this command is that `create_surface_main.out` creates a report about the surface, which contains detailed information about the surface (for instance about the automorphism group and the action of it). Sometimes, these reports are more useful if the surface equation is the one that we wish to consider, rather than the equation that Orbiter's classification algorithm chose. The option `-transform` works similarly, except that the transformation is not inverted. Many repeats and combinations of the `-transform` and `-transform_inverse` options are possible. The transformations are applied in the order in which the commands are given on the command line.

12 Acknowledgements

I thank Sajeeb Roy Chowdhury for help with shallow Schreier trees and clique finding. I thank Abdullah AlAzemi from Kuwait University for help with interfacing Nauty.

A The Field $\mathbb{F}_4$

polynomial: $X^2 + X + 1 = 7$
$Z_i = \log_\alpha(1 + \alpha^i)$

i	γ_i	$-\gamma_i$	γ_i^{-1}	$\log_\alpha(\gamma_i)$	α^i	Z_i	$\phi(\gamma_i)$	$T(\gamma_i)$	$N(\gamma_i)$
0	$0 = 0$	0	DNE	DNE	1	DNE	0	0	0
1	$1 = 1$	1	1	3	2	2	1	0	1
2	$\alpha = \alpha$	2	3	1	3	1	3	1	1
3	$\alpha + 1 = \alpha^2$	3	2	2	1	DNE	2	1	1

+	0	1	2	3
0	0	1	2	3
1	1	0	3	2
2	2	3	0	1
3	3	2	1	0

+	0	1	α	α^2
0	0	1	α	α^2
1	1	0	α^2	α
α	α	α^2	0	1
α^2	α^2	α	1	0

$\cdot$	1	2	3
1	1	2	3
2	2	3	1
3	3	1	2

$\cdot$	1	α	α^2
1	1	α	α^2
α	α	α^2	1
α^2	α^2	1	α

B Cheat Sheet PG$(2, 4)$

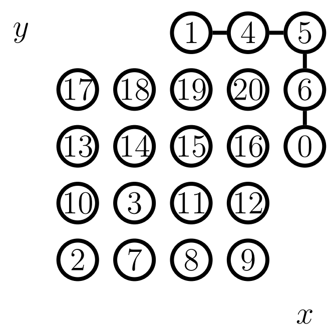
$q = 4$
$p = 2$
$e = 2$
$n = 2$
Number of points $= 21$
Number of lines $= 21$
Number of lines on a point $= 5$
Number of points on a line $= 5$

B.1 The Finite Field with 4 Elements

polynomial: $X^2 + X + 1 = 7$
$Z_i = \log_\alpha(1 + \alpha^i)$

i	γ_i	$-\gamma_i$	γ_i^{-1}	$\log_\alpha(\gamma_i)$	α^i	Z_i	$\phi(\gamma_i)$	$T(\gamma_i)$	$N(\gamma_i)$
0	$0 = 0$	0	DNE	DNE	1	DNE	0	0	0
1	$1 = 1$	1	1	3	2	2	1	0	1
2	$\alpha = \alpha$	2	3	1	3	1	3	1	1
3	$\alpha + 1 = \alpha^2$	3	2	2	1	DNE	2	1	1

B.2 The Plane



B.3 Points and Lines

PG(2, 4) has 21 points:

$P_0 = (1, 0, 0) = (1, 0, 0)$
$P_1 = (0, 1, 0) = (0, 1, 0)$
$P_2 = (0, 0, 1) = (0, 0, 1)$
$P_3 = (1, 1, 1) = (1, 1, 1)$
$P_4 = (1, 1, 0) = (1, 1, 0)$
$P_5 = (2, 1, 0) = (\alpha, 1, 0)$
$P_6 = (3, 1, 0) = (\alpha^2, 1, 0)$
$P_7 = (1, 0, 1) = (1, 0, 1)$
$P_8 = (2, 0, 1) = (\alpha, 0, 1)$
$P_9 = (3, 0, 1) = (\alpha^2, 0, 1)$
$P_{10} = (0, 1, 1) = (0, 1, 1)$
$P_{11} = (2, 1, 1) = (\alpha, 1, 1)$
$P_{12} = (3, 1, 1) = (\alpha^2, 1, 1)$
$P_{13} = (0, 2, 1) = (0, \alpha, 1)$
$P_{14} = (1, 2, 1) = (1, \alpha, 1)$
$P_{15} = (2, 2, 1) = (\alpha, \alpha, 1)$
$P_{16} = (3, 2, 1) = (\alpha^2, \alpha, 1)$
$P_{17} = (0, 3, 1) = (0, \alpha^2, 1)$
$P_{18} = (1, 3, 1) = (1, \alpha^2, 1)$
$P_{19} = (2, 3, 1) = (\alpha, \alpha^2, 1)$
$P_{20} = (3, 3, 1) = (\alpha^2, \alpha^2, 1)$

Normalized from the left:

$P_0 = (1, 0, 0)$
$P_1 = (0, 1, 0)$
$P_2 = (0, 0, 1)$
$P_3 = (1, 1, 1)$
$P_4 = (1, 1, 0)$
$P_5 = (1, 3, 0)$
$P_6 = (1, 2, 0)$
$P_7 = (1, 0, 1)$
$P_8 = (1, 0, 3)$
$P_9 = (1, 0, 2)$
$P_{10} = (0, 1, 1)$
$P_{11} = (1, 3, 3)$
$P_{12} = (1, 2, 2)$
$P_{13} = (0, 1, 3)$
$P_{14} = (1, 2, 1)$
$P_{15} = (1, 1, 3)$
$P_{16} = (1, 3, 2)$
$P_{17} = (0, 1, 2)$
$P_{18} = (1, 3, 1)$
$P_{19} = (1, 2, 3)$
$P_{20} = (1, 1, 2)$

PG(2, 4) has 21 points:

$P_{5\cdot i+j}$	0	1	2	3	4
0	(1, 0, 0)	(0, 1, 0)	(0, 0, 1)	(1, 1, 1)	(1, 1, 0)
5	(2, 1, 0)	(3, 1, 0)	(1, 0, 1)	(2, 0, 1)	(3, 0, 1)
10	(0, 1, 1)	(2, 1, 1)	(3, 1, 1)	(0, 2, 1)	(1, 2, 1)
15	(2, 2, 1)	(3, 2, 1)	(0, 3, 1)	(1, 3, 1)	(2, 3, 1)
20	(3, 3, 1)				

PG(2, 4) has 21 lines, each with 5 points:

	0	1	2	3	4
0	0	1	4	5	6
1	0	10	3	11	12
2	0	17	20	18	19
3	0	13	15	16	14
4	0	2	7	8	9
5	7	1	3	18	14
6	7	10	4	16	19
7	7	17	15	5	12
8	7	13	20	11	6
9	4	2	3	15	20
10	9	1	20	16	12
11	9	10	15	18	6
12	9	17	4	11	14
13	9	13	3	5	19
14	6	2	14	19	12
15	8	1	15	11	19
16	8	10	20	5	14
17	8	17	3	16	6
18	8	13	4	18	12
19	5	2	18	11	16
20	1	2	10	13	17

PG(2, 4) has 21 points, each with 5 lines:

	0	1	2	3	4
0	0	1	2	3	4
1	0	5	10	15	20
2	4	9	14	19	20
3	1	5	9	13	17
4	0	6	9	12	18
5	0	7	13	16	19
6	0	8	11	14	17
7	4	5	6	7	8
8	4	15	16	17	18
9	4	10	11	12	13
10	1	6	11	16	20
11	1	8	12	15	19
12	1	7	10	14	18
13	3	8	13	18	20
14	3	5	12	14	16
15	3	7	9	11	15
16	3	6	10	17	19
17	2	7	12	17	20
18	2	5	11	18	19
19	2	6	13	14	15
20	2	8	9	10	16

B.4 Subspaces of dimension 1

PG$(2, 4)$ has 21 1-subspaces:

$$L_0 = \begin{bmatrix} 100 \\ 010 \end{bmatrix} \quad L_5 = \begin{bmatrix} 101 \\ 010 \end{bmatrix} \quad L_{10} = \begin{bmatrix} 102 \\ 010 \end{bmatrix} \quad L_{15} = \begin{bmatrix} 103 \\ 010 \end{bmatrix} \quad L_{20} = \begin{bmatrix} 010 \\ 001 \end{bmatrix}$$

$$L_1 = \begin{bmatrix} 100 \\ 011 \end{bmatrix} \quad L_6 = \begin{bmatrix} 101 \\ 011 \end{bmatrix} \quad L_{11} = \begin{bmatrix} 102 \\ 011 \end{bmatrix} \quad L_{16} = \begin{bmatrix} 103 \\ 011 \end{bmatrix}$$

$$L_2 = \begin{bmatrix} 100 \\ 012 \end{bmatrix} \quad L_7 = \begin{bmatrix} 101 \\ 012 \end{bmatrix} \quad L_{12} = \begin{bmatrix} 102 \\ 012 \end{bmatrix} \quad L_{17} = \begin{bmatrix} 103 \\ 012 \end{bmatrix}$$

$$L_3 = \begin{bmatrix} 100 \\ 013 \end{bmatrix} \quad L_8 = \begin{bmatrix} 101 \\ 013 \end{bmatrix} \quad L_{13} = \begin{bmatrix} 102 \\ 013 \end{bmatrix} \quad L_{18} = \begin{bmatrix} 103 \\ 013 \end{bmatrix}$$

$$L_4 = \begin{bmatrix} 100 \\ 001 \end{bmatrix} \quad L_9 = \begin{bmatrix} 110 \\ 001 \end{bmatrix} \quad L_{14} = \begin{bmatrix} 120 \\ 001 \end{bmatrix} \quad L_{19} = \begin{bmatrix} 130 \\ 001 \end{bmatrix}$$

B.5 Line intersections

intersection of 2 lines:

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
0		0	0	0	0	1	4	5	6	4	1	6	4	5	6	1	5	6	4	5	1
1	0		0	0	0	3	10	12	11	3	12	10	11	3	12	11	10	3	12	11	10
2	0	0		0	0	18	19	17	20	20	20	18	17	19	19	19	20	17	18	18	17
3	0	0	0		0	14	16	15	13	15	16	15	14	13	14	15	14	16	13	16	13
4	0	0	0	0		7	7	7	7	2	9	9	9	9	2	8	8	8	8	2	2
5	1	3	18	14	7		7	7	7	3	1	18	14	3	14	1	14	3	18	18	1
6	4	10	19	16	7	7		7	7	4	16	10	4	19	19	19	10	16	4	16	10
7	5	12	17	15	7	7	7		7	15	12	15	17	5	12	15	5	17	12	5	17
8	6	11	20	13	7	7	7	7		20	20	6	11	13	6	11	20	6	13	11	13
9	4	3	20	15	2	3	4	15	20		20	15	4	3	2	15	20	3	4	2	2
10	1	12	20	16	9	1	16	12	20	20		9	9	9	12	1	20	16	12	16	1
11	6	10	18	15	9	18	10	15	6	15	9		9	9	6	15	10	6	18	18	10
12	4	11	17	14	9	14	4	17	11	4	9	9		9	14	11	14	17	4	11	17
13	5	3	19	13	9	3	19	5	13	3	9	9	9		19	19	5	3	13	5	13
14	6	12	19	14	2	14	19	12	6	2	12	6	14	19		19	14	6	12	2	2
15	1	11	19	15	8	1	19	15	11	15	1	15	11	19	19		8	8	8	11	1
16	5	10	20	14	8	14	10	5	20	20	20	10	14	5	14	8		8	8	5	10
17	6	3	17	16	8	3	16	17	6	3	16	6	17	3	6	8	8		8	16	17
18	4	12	18	13	8	18	4	12	13	4	12	18	4	13	12	8	8	8		18	13
19	5	11	18	16	2	18	16	5	11	2	16	18	11	5	2	11	5	16	18		2
20	1	10	17	13	2	1	10	17	13	2	1	10	17	13	2	1	10	17	13	2	

B.6 Line through point-pairs

line through 2 points:

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
0		0	4	1	0	0	0	4	4	4	1	1	1	3	3	3	3	2	2	2	2
1	0		20	5	0	0	0	5	15	10	20	15	10	20	5	15	10	20	5	15	10
2	4	20		9	9	19	14	4	4	4	20	19	14	20	14	9	19	20	19	14	9
3	1	5	9		9	13	17	5	17	13	1	1	1	13	5	9	17	17	5	13	9
4	0	0	9	9		0	0	6	18	12	6	12	18	18	12	9	6	12	18	6	9
5	0	0	19	13	0		0	7	16	13	16	19	7	13	16	7	19	7	19	13	16
6	0	0	14	17	0	0		8	17	11	11	8	14	8	14	11	17	17	11	14	8
7	4	5	4	5	6	7	8		4	4	6	8	7	8	5	7	6	7	5	6	8
8	4	15	4	17	18	16	17	4		4	16	15	18	18	16	15	17	17	18	15	16
9	4	10	4	13	12	13	11	4	4		11	12	10	13	12	11	10	12	11	13	10
10	1	20	20	1	6	16	11	6	16	11		1	1	20	16	11	6	20	11	6	16
11	1	15	19	1	12	19	8	8	15	12	1		1	8	12	15	19	12	19	15	8
12	1	10	14	1	18	7	14	7	18	10	1	1		18	14	7	10	7	18	14	10
13	3	20	20	13	18	13	8	8	18	13	20	8	18		3	3	3	20	18	13	8
14	3	5	14	5	12	16	14	5	16	12	16	12	14	3		3	3	12	5	14	16
15	3	15	9	9	9	7	11	7	15	11	11	15	7	3	3		3	7	11	15	9
16	3	10	19	17	6	19	17	6	17	10	6	19	10	3	3	3		17	19	6	10
17	2	20	20	17	12	7	17	7	17	12	20	12	7	20	12	7	17		2	2	2
18	2	5	19	5	18	19	11	5	18	11	11	19	18	18	5	11	19	2		2	2
19	2	15	14	13	6	13	14	6	15	13	6	15	14	13	14	15	6	2	2		2
20	2	10	9	9	9	16	8	8	16	10	16	8	10	8	16	9	10	2	2	2	

C The Group PGL(7, 11)SubgroupJanko1

The order of the group PGL(7, 11)SubgroupJanko1 is 175560
The field $\mathbb{F}_{11}$:
$Z_i = \log_\alpha(1 + \alpha^i)$

i	γ_i	$-\gamma_i$	γ_i^{-1}	$\log_\alpha(\gamma_i)$	α^i	Z_i
0	$0 = 0$	0	DNE	DNE	1	1
1	$1 = 1$	10	1	0	2	8
2	$2 = \alpha$	9	6	1	4	4
3	$3 = \alpha^8$	8	4	8	8	6
4	$4 = \alpha^2$	7	3	2	5	9
5	$5 = \alpha^4$	6	9	4	10	DNE
6	$6 = \alpha^9$	5	2	9	9	5
7	$7 = \alpha^7$	4	8	7	7	3
8	$8 = \alpha^3$	3	7	3	3	2
9	$9 = \alpha^6$	2	5	6	6	7
10	$10 = \alpha^5$	1	10	5	1	1

The group acts on a set of size 1948717
Strong generators for a group of order 175560:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 10 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 10 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 10 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 10 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 10 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 10 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 10 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 10 \end{bmatrix},$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 10 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 10 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 10 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 1 & 3 & 4 & 4 & 1 & 4 & 1 \\ 8 & 7 & 7 & 10 & 7 & 10 & 10 \\ 4 & 4 & 1 & 4 & 1 & 1 & 3 \\ 4 & 1 & 4 & 1 & 1 & 3 & 4 \\ 1 & 4 & 1 & 1 & 3 & 4 & 4 \\ 7 & 10 & 10 & 8 & 7 & 7 & 10 \\ 10 & 10 & 8 & 7 & 7 & 10 & 7 \end{bmatrix}, \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

1,0,0,0,0,0,0,0,10,0,0,0,0,0,0,0,10,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,10,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,
1,0,0,0,0,0,0,0,10,0,0,0,0,0,0,0,10,0,0,0,0,0,0,0,10,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,10,
1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,10,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,10,0,0,0,0,0,0,0,10,0,0,0,0,0,0,0,10,
1,0,0,0,0,0,0,0,0,0,0,1,0,0,0,10,0,0,0,0,0,0,0,0,0,0,10,0,0,0,1,0,0,0,0,0,0,0,0,0,0,10,0,0,0,0,1,0,0,0,
1,3,4,4,1,4,1,8,7,7,10,7,10,10,4,4,1,4,1,1,3,4,1,4,1,1,3,4,1,4,1,1,3,4,4,7,10,10,8,7,7,10,10,10,8,7,7,10,7,
0,1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,1,0,0,0,0,0,0,
Group action PGL$(7, 11)$ of degree 1948717
Group order 175560
tl=$7315, 3, 1, 1, 1, 1, 1, 8,$
Base: $(0, 1, 2, 3, 4, 5, 6, 7)$

Strong generators for a group of order 175560:

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 10 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 10 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 10 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 10 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 10 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 10 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 10 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 10 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 10 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 10 \end{bmatrix},$$

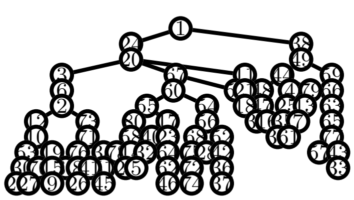
$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 10 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 10 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 10 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 1 & 3 & 4 & 4 & 1 & 4 & 1 \\ 8 & 7 & 7 & 10 & 7 & 10 & 10 \\ 4 & 4 & 1 & 4 & 1 & 1 & 3 \\ 4 & 1 & 4 & 1 & 1 & 3 & 4 \\ 1 & 4 & 1 & 1 & 3 & 4 & 4 \\ 7 & 10 & 10 & 8 & 7 & 7 & 10 \\ 10 & 10 & 8 & 7 & 7 & 10 & 7 \end{bmatrix}, \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

1,0,0,0,0,0,0,0,10,0,0,0,0,0,0,0,10,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,10,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,
1,0,0,0,0,0,0,0,10,0,0,0,0,0,0,0,10,0,0,0,0,0,0,0,10,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,10,
1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,10,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,10,0,0,0,0,0,0,0,10,0,0,0,0,0,0,0,10,
1,0,0,0,0,0,0,0,0,0,0,1,0,0,0,10,0,0,0,0,0,0,0,0,0,0,10,0,0,0,1,0,0,0,0,0,0,0,0,0,0,10,0,0,0,1,0,0,0,
1,3,4,4,1,4,1,8,7,7,10,7,10,10,4,4,1,4,1,1,3,4,1,4,1,1,3,4,1,4,1,1,3,4,4,7,10,10,8,7,7,10,10,10,8,7,7,10,7,
0,1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,1,0,0,0,0,0,0,

Stabilizer chain

Level	Base pt	Orbit length	Subgroup order
0	0	7315	175560
1	1	3	24
2	2	1	8
3	3	1	8
4	4	1	8
5	5	1	8
6	6	1	8
7	7	8	8

Basic Orbit 0



Basic Orbit 1

Basic Orbit 2

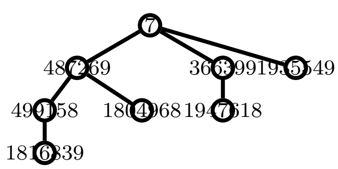
Basic Orbit 3

Basic Orbit 4

Basic Orbit 5

Basic Orbit 6

Basic Orbit 7



The base has length 8
The basic orbits are:
Basic orbit 0 is orbit of 0 of length 1948717
Basic orbit 1 is orbit of 1 of length 1948716
Basic orbit 2 is orbit of 2 of length 1948705
Basic orbit 3 is orbit of 3 of length 1948584
Basic orbit 4 is orbit of 4 of length 1947253
Basic orbit 5 is orbit of 5 of length 1932612

Basic orbit 6 is orbit of 6 of length 1771561
Basic orbit 7 is orbit of 7 of length 1000000

D The Group $\mathrm{P\Gamma L}(3,4)$

The Group $\mathrm{P\Gamma L}(3,4)$
The order of the group $\mathrm{P\Gamma L}(3,4)$ is 120960
The field $\mathbb{F}_4$:
polynomial: $X^2+X+1=7$
$Z_i = \log_\alpha(1+\alpha^i)$

i	γ_i	$-\gamma_i$	γ_i^{-1}	$\log_\alpha(\gamma_i)$	α^i	Z_i	$\phi(\gamma_i)$	$T(\gamma_i)$	$N(\gamma_i)$
0	$0=0$	0	DNE	DNE	1	DNE	0	0	0
1	$1=1$	1	1	3	2	2	1	0	1
2	$\alpha=\alpha$	2	3	1	3	1	3	1	1
3	$\alpha+1=\alpha^2$	3	2	2	1	DNE	2	1	1

The group acts on a set of size 21

i	P_i
0	$(1,0,0)$
1	$(0,1,0)$
2	$(0,0,1)$
3	$(1,1,1)$
4	$(1,1,0)$
5	$(2,1,0)$
6	$(3,1,0)$
7	$(1,0,1)$
8	$(2,0,1)$
9	$(3,0,1)$

,

i	P_i
10	$(0,1,1)$
11	$(2,1,1)$
12	$(3,1,1)$
13	$(0,2,1)$
14	$(1,2,1)$
15	$(2,2,1)$
16	$(3,2,1)$
17	$(0,3,1)$
18	$(1,3,1)$
19	$(2,3,1)$

,

i	P_i
20	$(3,3,1)$

Nice generators:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}_1, \begin{bmatrix} 1 & 0 & 0 \\ 0 & \alpha^2 & 0 \\ 0 & 0 & \alpha^2 \end{bmatrix}_0, \begin{bmatrix} 1 & 0 & 0 \\ 0 & \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}_0,$$

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}_0, \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ \alpha & 0 & 1 \end{bmatrix}_0, \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}_0,$$

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & \alpha & 1 \end{bmatrix}_0, \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}_0, \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}_0$$

Group action PΓL(3, 4) of degree 21
Group order 120960
tl=$21, 20, 16, 9, 2,$
Base: $(0, 1, 2, 3, 5)$
Strong generators for a group of order 120960:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}_1, \begin{bmatrix} 1 & 0 & 0 \\ 0 & \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}_0, \begin{bmatrix} 1 & 0 & 0 \\ 0 & \alpha^2 & 0 \\ 0 & 0 & \alpha^2 \end{bmatrix}_1,$$

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ \alpha & 0 & 1 \end{bmatrix}_0, \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & \alpha & 1 \end{bmatrix}_0, \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}_0,$$

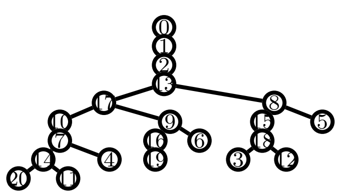
$$\begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}_0$$

1,0,0,0,1,0,0,0,1,1,
1,0,0,0,2,0,0,0,1,0,
1,0,0,0,3,0,0,0,3,1,
1,0,0,0,1,0,2,0,1,0,
1,0,0,0,1,0,0,2,1,0,
1,0,0,0,0,1,0,1,0,0,
0,1,0,1,0,0,0,0,1,0,

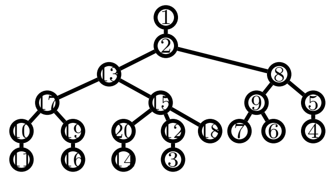
Stabilizer chain

Level	Base pt	Orbit length	Subgroup order
0	0	21	120960
1	1	20	5760
2	2	16	288
3	3	9	18
4	5	2	2

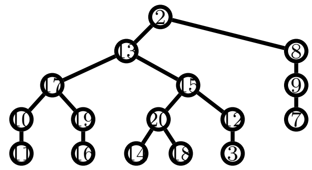
Basic Orbit 0



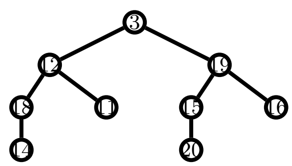
Basic Orbit 1



Basic Orbit 2



Basic Orbit 3



Basic Orbit 4

The base has length 5
The basic orbits are:
Basic orbit 0 is orbit of 0 of length 21
Basic orbit 1 is orbit of 1 of length 20
Basic orbit 2 is orbit of 2 of length 16
Basic orbit 3 is orbit of 3 of length 9
Basic orbit 4 is orbit of 5 of length 2
The 2-Sylow groups have order 2^7
The 3-Sylow groups have order 3^3
The 5-Sylow groups have order 5^1
The 7-Sylow groups have order 7^1
One 2-Sylow group has the following generators:
Strong generators for a group of order 128:

$$\left[\begin{array}{ccc} 1 & 0 & 0 \\ 1 & \alpha^2 & 1 \\ \alpha^2 & 0 & \alpha \end{array}\right]_1, \left[\begin{array}{ccc} 1 & 0 & 0 \\ 1 & 1 & 0 \\ \alpha^2 & 1 & 1 \end{array}\right]_1, \left[\begin{array}{ccc} 1 & 0 & 0 \\ 1 & 0 & \alpha^2 \\ \alpha^2 & \alpha & \alpha^2 \end{array}\right]_1,$$

$$\begin{bmatrix} 1 & 0 & 1 \\ 1 & \alpha^2 & \alpha^2 \\ \alpha^2 & 0 & 1 \end{bmatrix}_0, \begin{bmatrix} 1 & \alpha & 1 \\ 1 & \alpha & \alpha \\ \alpha^2 & \alpha^2 & 0 \end{bmatrix}_1, \begin{bmatrix} 0 & 1 & 1 \\ \alpha^2 & \alpha & \alpha^2 \\ \alpha & \alpha & \alpha^2 \end{bmatrix}_0,$$

$$\begin{bmatrix} 1 & 1 & \alpha^2 \\ 1 & 0 & \alpha \\ \alpha^2 & \alpha & \alpha \end{bmatrix}_0$$

1,0,0,1,3,1,3,0,2,1,
1,0,0,1,1,0,3,1,1,1,
1,0,0,1,0,3,3,2,3,1,
1,0,1,1,3,3,3,0,1,0,
1,2,1,1,2,2,3,3,0,1,
0,1,1,3,2,3,2,2,3,0,
1,1,3,1,0,2,3,2,2,0,
One 3-Sylow group has the following generators:
Strong generators for a group of order 27:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ \alpha & 0 & \alpha^2 \end{bmatrix}_0, \begin{bmatrix} 1 & \alpha^2 & 0 \\ 0 & \alpha & 0 \\ \alpha & \alpha^2 & \alpha^2 \end{bmatrix}_0, \begin{bmatrix} 1 & \alpha & \alpha^2 \\ \alpha & \alpha & 0 \\ 1 & 0 & \alpha^2 \end{bmatrix}_0$$

1,0,0,0,1,0,2,0,3,0,
1,3,0,0,2,0,2,3,3,0,
1,2,3,2,2,0,1,0,3,0,
One 5-Sylow group has the following generators:
Strong generators for a group of order 5:

$$\begin{bmatrix} 1 & \alpha^2 & \alpha^2 \\ 0 & 0 & 1 \\ 1 & \alpha & \alpha^2 \end{bmatrix}_0$$

1,3,3,0,0,1,1,2,3,0,
One 7-Sylow group has the following generators:
Strong generators for a group of order 7:

$$\begin{bmatrix} 0 & 1 & \alpha \\ \alpha & 1 & 1 \\ \alpha & \alpha & \alpha^2 \end{bmatrix}_0$$

0,1,2,2,1,1,2,2,3,0,

E Conjugacy classes in $\mathrm{PGL}(3,2)$

The group order is

$$168$$

Class 0 / 6

Order of element = 1
Class size = 1
Centralizer order = 168
Normalizer order = 168
The normalizer is generated by:
Strong generators for a group of order 168:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix},$$

$$\begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

1,0,0,0,1,0,1,1,1,
1,0,0,0,1,0,0,1,1,
1,0,0,1,1,1,1,0,1,
1,0,1,1,1,0,1,0,0,

Class 1 / 6

Order of element = 2
Class size = 21
Centralizer order = 8
Normalizer order = 8
Representing element is

$$\begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$1, 0, 0, 1, 1, 0, 0, 0, 1,$
The normalizer is generated by:
Strong generators for a group of order 8:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}$$

1,0,0,0,1,0,1,0,1,
1,0,0,1,1,0,0,0,1,
1,0,0,1,1,1,0,0,1,

Class 2 / 6

Order of element = 3
Class size = 56
Centralizer order = 3
Normalizer order = 6
Representing element is

$$\begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

$1, 0, 1, 1, 1, 0, 1, 0, 0,$
The normalizer is generated by:
Strong generators for a group of order 6:

$$\begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

1,0,0,1,1,0,1,0,1,
1,0,1,1,1,0,1,0,0,

Class 3 / 6

Order of element = 4
Class size = 42
Centralizer order = 4
Normalizer order = 8
Representing element is

$$\begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix}$$

$1, 0, 0, 1, 1, 1, 1, 0, 1,$
The normalizer is generated by:
Strong generators for a group of order 8:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix}$$

1,0,0,0,1,0,1,0,1,
1,0,0,1,1,1,1,0,1,

Class 4 / 6

Order of element = 7
Class size = 24
Centralizer order = 7
Normalizer order = 21
Representing element is

$$\begin{bmatrix} 0 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

$0, 1, 1, 1, 1, 0, 1, 0, 0,$
The normalizer is generated by:
Strong generators for a group of order 21:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

1,0,0,0,1,1,1,1,0,
0,1,1,1,1,0,1,0,0,

Class 5 / 6

Order of element = 7
Class size = 24
Centralizer order = 7
Normalizer order = 21
Representing element is

$$\begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

$1, 1, 0, 1, 1, 1, 0, 1, 0,$
The normalizer is generated by:
Strong generators for a group of order 21:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 1 & 0 \end{bmatrix}, \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

1,0,0,0,1,1,1,1,0,
1,1,0,1,1,1,0,1,0,

F The Group $\mathrm{PGO}^+(6,2)$

The order of the group $\mathrm{PGO}^+(6,2)$ is 40320
The field $\mathbb{F}_2$:
$Z_i = \log_\alpha(1+\alpha^i)$

i	γ_i	$-\gamma_i$	γ_i^{-1}	$\log_\alpha(\gamma_i)$	α^i	Z_i
0	$0=0$	0	DNE	DNE	1	DNE
1	$1=1$	1	1	0	1	DNE

The group acts on a set of size 35

i	P_i
0	$(1,0,0,0,0,0)$
1	$(0,1,0,0,0,0)$
2	$(0,0,1,0,0,0)$
3	$(1,0,1,0,0,0)$
4	$(0,1,1,0,0,0)$
5	$(0,0,0,1,0,0)$
6	$(1,0,0,1,0,0)$
7	$(0,1,0,1,0,0)$
8	$(1,1,1,1,0,0)$
9	$(0,0,0,0,1,0)$

,

i	P_i
10	$(1,0,0,0,1,0)$
11	$(0,1,0,0,1,0)$
12	$(0,0,1,0,1,0)$
13	$(1,0,1,0,1,0)$
14	$(0,1,1,0,1,0)$
15	$(0,0,0,1,1,0)$
16	$(1,0,0,1,1,0)$
17	$(0,1,0,1,1,0)$
18	$(1,1,1,1,1,0)$
19	$(0,0,0,0,0,1)$

,

i	P_i
20	$(1,0,0,0,0,1)$
21	$(0,1,0,0,0,1)$
22	$(0,0,1,0,0,1)$
23	$(1,0,1,0,0,1)$
24	$(0,1,1,0,0,1)$
25	$(0,0,0,1,0,1)$
26	$(1,0,0,1,0,1)$
27	$(0,1,0,1,0,1)$
28	$(1,1,1,1,0,1)$
29	$(1,1,0,0,1,1)$

,

i	P_i
30	$(1,1,1,0,1,1)$
31	$(1,1,0,1,1,1)$
32	$(0,0,1,1,1,1)$
33	$(1,0,1,1,1,1)$
34	$(0,1,1,1,1,1)$

Strong generators for a group of order 40320:

$$\begin{bmatrix} 1&0&0&0&0&0\\ 0&1&0&0&0&0\\ 0&0&1&0&0&0\\ 0&0&0&1&0&0\\ 0&0&0&0&0&1\\ 0&0&0&0&1&0 \end{bmatrix}, \begin{bmatrix} 1&0&0&0&0&0\\ 0&1&0&0&0&0\\ 0&0&1&0&0&0\\ 0&0&1&1&1&1\\ 0&0&1&0&0&1\\ 0&0&1&0&1&0 \end{bmatrix}, \begin{bmatrix} 1&0&0&0&0&0\\ 0&1&0&0&0&0\\ 0&0&1&0&0&0\\ 0&0&0&1&1&0\\ 0&0&1&0&0&1\\ 0&0&0&0&1&0 \end{bmatrix},$$

$$\begin{bmatrix} 1&0&0&0&0&0\\ 0&1&0&0&0&0\\ 0&0&0&1&1&0\\ 0&0&0&0&0&1\\ 0&0&1&0&0&1\\ 0&0&0&1&0&0 \end{bmatrix}, \begin{bmatrix} 1&0&0&0&0&0\\ 0&1&0&0&0&0\\ 0&0&0&1&0&0\\ 0&0&1&0&1&0\\ 0&0&0&0&1&0\\ 0&0&0&1&0&1 \end{bmatrix}, \begin{bmatrix} 1&0&0&0&0&0\\ 0&1&1&1&1&1\\ 1&0&0&0&1&0\\ 1&0&0&0&0&1\\ 1&0&1&0&0&0\\ 1&0&0&1&0&0 \end{bmatrix},$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 1 & 0 \end{bmatrix},$$

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 \end{bmatrix}$$

1,0,0,0,0,0,1,0,0,0,0,0,1,0,0,0,0,0,1,0,0,0,0,0,0,1,0,0,0,0,1,0,
1,0,0,0,0,0,0,1,0,0,0,0,0,1,0,0,0,0,1,1,1,1,0,0,1,0,0,1,0,0,1,0,1,0,
1,0,0,0,0,0,0,1,0,0,0,0,0,1,0,0,0,0,0,1,1,0,0,0,1,0,0,1,0,0,0,0,1,0,
1,0,0,0,0,0,0,1,0,0,0,0,0,0,1,1,0,0,0,0,0,1,0,0,1,0,0,1,0,0,0,1,0,0,
1,0,0,0,0,0,0,1,0,0,0,0,0,0,1,0,0,0,0,1,0,1,0,0,0,0,0,1,0,0,0,0,1,0,1,
1,0,0,0,0,0,0,1,1,1,1,1,1,0,0,0,1,0,1,0,0,0,0,1,1,0,1,0,0,0,1,0,0,1,0,0,
1,0,0,0,0,0,0,1,0,1,0,0,1,0,1,0,0,1,0,0,0,1,0,0,0,0,0,1,1,0,0,0,0,0,0,1,
0,1,0,1,1,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,1,0,1,0,0,0,0,0,1,0,0,1,1,0,0,1,
0,0,1,1,1,1,1,0,1,0,0,0,0,1,1,1,1,1,1,1,1,1,0,0,0,1,0,1,0,1,0,1,0,1,1,0,
0,0,0,0,1,0,1,0,1,1,1,1,1,0,1,0,1,0,1,0,0,1,1,0,1,0,0,0,0,0,1,1,1,1,1,0,
Group action $\mathrm{PGO}^+(6,2)$ of degree 35
Group order 40320
tl=$35, 16, 9, 1, 1, 4, 2,$
Base: $(0, 1, 2, 3, 4, 5, 9)$
Strong generators for a group of order 40320:

$$\begin{bmatrix} 1\,0\,0\,0\,0\,0 \\ 0\,1\,0\,0\,0\,0 \\ 0\,0\,1\,0\,0\,0 \\ 0\,0\,0\,1\,0\,0 \\ 0\,0\,0\,0\,0\,1 \\ 0\,0\,0\,0\,1\,0 \end{bmatrix}, \begin{bmatrix} 1\,0\,0\,0\,0\,0 \\ 0\,1\,0\,0\,0\,0 \\ 0\,0\,1\,0\,0\,0 \\ 0\,0\,1\,1\,1\,1 \\ 0\,0\,1\,0\,0\,1 \\ 0\,0\,1\,0\,1\,0 \end{bmatrix}, \begin{bmatrix} 1\,0\,0\,0\,0\,0 \\ 0\,1\,0\,0\,0\,0 \\ 0\,0\,1\,0\,0\,0 \\ 0\,0\,0\,1\,1\,0 \\ 0\,0\,1\,0\,0\,1 \\ 0\,0\,0\,0\,1\,0 \end{bmatrix},$$

$$\begin{bmatrix} 1\,0\,0\,0\,0\,0 \\ 0\,1\,0\,0\,0\,0 \\ 0\,0\,0\,1\,1\,0 \\ 0\,0\,0\,0\,0\,1 \\ 0\,0\,1\,0\,0\,1 \\ 0\,0\,0\,1\,0\,0 \end{bmatrix}, \begin{bmatrix} 1\,0\,0\,0\,0\,0 \\ 0\,1\,0\,0\,0\,0 \\ 0\,0\,0\,1\,0\,0 \\ 0\,0\,1\,0\,1\,0 \\ 0\,0\,0\,0\,1\,0 \\ 0\,0\,0\,1\,0\,1 \end{bmatrix}, \begin{bmatrix} 1\,0\,0\,0\,0\,0 \\ 0\,1\,1\,1\,1\,1 \\ 1\,0\,0\,0\,1\,0 \\ 1\,0\,0\,0\,0\,1 \\ 1\,0\,1\,0\,0\,0 \\ 1\,0\,0\,1\,0\,0 \end{bmatrix},$$

$$\begin{bmatrix} 1&0&0&0&0&0 \\ 0&1&0&1&0&0 \\ 1&0&1&0&0&1 \\ 0&0&0&1&0&0 \\ 0&0&0&1&1&0 \\ 0&0&0&0&0&1 \end{bmatrix}, \begin{bmatrix} 0&1&0&1&1&0 \\ 0&0&1&0&0&0 \\ 0&1&0&0&0&0 \\ 1&0&1&0&1&0 \\ 0&0&0&0&1&0 \\ 0&1&1&0&0&1 \end{bmatrix}, \begin{bmatrix} 0&0&1&1&1&1 \\ 1&0&1&0&0&0 \\ 0&1&1&1&1&1 \\ 1&1&1&1&0&0 \\ 0&1&0&1&0&1 \\ 0&1&0&1&1&0 \end{bmatrix},$$

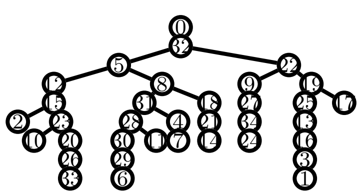
$$\begin{bmatrix} 0&0&0&0&1&0 \\ 1&0&1&1&1&1 \\ 1&0&1&0&1&0 \\ 1&0&0&1&1&0 \\ 1&0&0&0&0&0 \\ 1&1&1&1&1&0 \end{bmatrix}$$

1,0,0,0,0,0,0,1,0,0,0,0,0,0,1,0,0,0,0,0,0,1,0,0,0,0,0,0,1,0,0,0,0,1,0,
1,0,0,0,0,0,0,1,0,0,0,0,0,0,1,0,0,0,0,0,1,1,1,1,0,0,1,0,0,1,0,0,1,0,1,0,
1,0,0,0,0,0,0,1,0,0,0,0,0,0,1,0,0,0,0,0,0,1,1,0,0,0,1,0,0,1,0,0,0,0,1,0,
1,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,1,0,0,0,0,0,0,1,0,0,1,0,0,1,0,0,0,1,0,0,
1,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,0,1,0,1,0,0,0,0,0,1,0,0,0,0,1,0,1,
1,0,0,0,0,0,0,1,1,1,1,1,1,0,0,0,1,0,1,0,0,0,0,1,1,0,1,0,0,0,1,0,0,1,0,0,
1,0,0,0,0,0,0,1,0,1,0,0,1,0,1,0,0,1,0,0,0,1,0,0,0,0,0,1,1,0,0,0,0,0,0,1,
0,1,0,1,1,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,1,0,1,0,0,0,0,0,1,0,0,1,1,0,0,1,
0,0,1,1,1,1,1,0,1,0,0,0,0,1,1,1,1,1,1,1,1,1,0,0,0,1,0,1,0,1,0,1,0,1,1,0,
0,0,0,0,1,0,1,0,1,1,1,1,1,0,1,0,1,0,1,0,0,1,1,0,1,0,0,0,0,0,1,1,1,1,1,0,

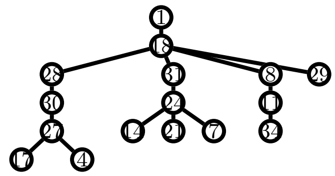
Stabilizer chain

Level	Base pt	Orbit length	Subgroup order
0	0	35	40320
1	1	16	1152
2	2	9	72
3	3	1	8
4	4	1	8
5	5	4	8
6	9	2	2

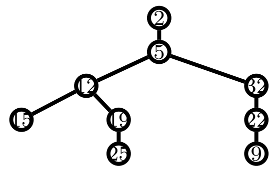
Basic Orbit 0



Basic Orbit 1



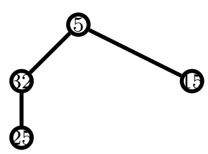
Basic Orbit 2



Basic Orbit 3

Basic Orbit 4

Basic Orbit 5



Basic Orbit 6

The base has length 7
The basic orbits are:
Basic orbit 0 is orbit of 0 of length 35
Basic orbit 1 is orbit of 1 of length 16
Basic orbit 2 is orbit of 2 of length 9
Basic orbit 3 is orbit of 3 of length 1
Basic orbit 4 is orbit of 4 of length 1

Basic orbit 5 is orbit of 5 of length 4
Basic orbit 6 is orbit of 9 of length 2

G The Group PGL(3, 9)SubgroupU_3_3(6048)

The order of the group PGL(3, 9)SubgroupU_3_3(6048) is 6048
The field $\mathbb{F}_9$:
polynomial: $X^2 + 2X + 2 = 17$
$Z_i = \log_\alpha(1 + \alpha^i)$

i	γ_i	$-\gamma_i$	γ_i^{-1}	$\log_\alpha(\gamma_i)$	α^i	Z_i	$\phi(\gamma_i)$	$T(\gamma_i)$	$N(\gamma_i)$
0	$0 = 0$	0	DNE	DNE	1	4	0	0	0
1	$1 = 1$	2	1	8	3	2	1	2	1
2	$2 = \alpha^4$	1	2	4	4	7	2	1	1
3	$\alpha = \alpha$	6	5	1	7	6	7	1	2
4	$\alpha + 1 = \alpha^2$	8	8	2	2	DNE	8	0	1
5	$\alpha + 2 = \alpha^7$	7	3	7	6	3	6	2	2
6	$2\alpha = \alpha^5$	3	7	5	8	5	5	2	2
7	$2\alpha + 1 = \alpha^3$	5	6	3	5	1	3	1	2
8	$2\alpha + 2 = \alpha^6$	4	4	6	1	4	4	0	1

The group acts on a set of size 91

i	P_i	i	P_i	i	P_i	i	P_i	i	P_i
0	$(1, 0, 0)$	10	$(7, 1, 0)$	20	$(0, 1, 1)$	30	$(2, 2, 1)$	40	$(3, 3, 1)$
1	$(0, 1, 0)$	11	$(8, 1, 0)$	21	$(2, 1, 1)$	31	$(3, 2, 1)$	41	$(4, 3, 1)$
2	$(0, 0, 1)$	12	$(1, 0, 1)$	22	$(3, 1, 1)$	32	$(4, 2, 1)$	42	$(5, 3, 1)$
3	$(1, 1, 1)$	13	$(2, 0, 1)$	23	$(4, 1, 1)$	33	$(5, 2, 1)$	43	$(6, 3, 1)$
4	$(1, 1, 0)$	14	$(3, 0, 1)$	24	$(5, 1, 1)$	34	$(6, 2, 1)$	44	$(7, 3, 1)$
5	$(2, 1, 0)$	15	$(4, 0, 1)$	25	$(6, 1, 1)$	35	$(7, 2, 1)$	45	$(8, 3, 1)$
6	$(3, 1, 0)$	16	$(5, 0, 1)$	26	$(7, 1, 1)$	36	$(8, 2, 1)$	46	$(0, 4, 1)$
7	$(4, 1, 0)$	17	$(6, 0, 1)$	27	$(8, 1, 1)$	37	$(0, 3, 1)$	47	$(1, 4, 1)$
8	$(5, 1, 0)$	18	$(7, 0, 1)$	28	$(0, 2, 1)$	38	$(1, 3, 1)$	48	$(2, 4, 1)$
9	$(6, 1, 0)$	19	$(8, 0, 1)$	29	$(1, 2, 1)$	39	$(2, 3, 1)$	49	$(3, 4, 1)$

i	P_i
50	$(4,4,1)$
51	$(5,4,1)$
52	$(6,4,1)$
53	$(7,4,1)$
54	$(8,4,1)$
55	$(0,5,1)$
56	$(1,5,1)$
57	$(2,5,1)$
58	$(3,5,1)$
59	$(4,5,1)$

,

i	P_i
60	$(5,5,1)$
61	$(6,5,1)$
62	$(7,5,1)$
63	$(8,5,1)$
64	$(0,6,1)$
65	$(1,6,1)$
66	$(2,6,1)$
67	$(3,6,1)$
68	$(4,6,1)$
69	$(5,6,1)$

,

i	P_i
70	$(6,6,1)$
71	$(7,6,1)$
72	$(8,6,1)$
73	$(0,7,1)$
74	$(1,7,1)$
75	$(2,7,1)$
76	$(3,7,1)$
77	$(4,7,1)$
78	$(5,7,1)$
79	$(6,7,1)$

,

i	P_i
80	$(7,7,1)$
81	$(8,7,1)$
82	$(0,8,1)$
83	$(1,8,1)$
84	$(2,8,1)$
85	$(3,8,1)$
86	$(4,8,1)$
87	$(5,8,1)$
88	$(6,8,1)$
89	$(7,8,1)$

,

i	P_i
90	$(8,8,1)$

Nice generators:

$$\begin{bmatrix} 1 & \alpha^5 & \alpha^2 \\ \alpha^7 & 0 & \alpha^5 \\ \alpha^6 & \alpha^7 & 1 \end{bmatrix}, \begin{bmatrix} 1 & \alpha^7 & \alpha^3 \\ \alpha^6 & \alpha & \alpha^5 \\ 0 & 1 & 1 \end{bmatrix}$$

Group action $\mathrm{PGL}(3,9)$ of degree 91
Group order 6048
tl=$63, 6, 1, 16,$
Base: $(0, 1, 2, 3)$
Strong generators for a group of order 6048:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & \alpha^4 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & \alpha^2 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 \\ 0 & \alpha^2 & 0 \\ 0 & 0 & \alpha^6 \end{bmatrix},$$

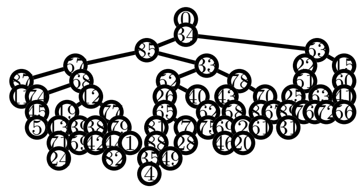
$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & \alpha^5 & \alpha^5 \\ 0 & \alpha & \alpha^5 \end{bmatrix}, \begin{bmatrix} 1 & \alpha^7 & \alpha^3 \\ \alpha^6 & \alpha & \alpha^5 \\ 0 & 1 & 1 \end{bmatrix}, \begin{bmatrix} 1 & \alpha^5 & \alpha^2 \\ \alpha^7 & 0 & \alpha^5 \\ \alpha^6 & \alpha^7 & 1 \end{bmatrix}$$

1,0,0,0,2,0,0,0,1,
1,0,0,0,1,0,0,0,4,
1,0,0,0,4,0,0,0,8,
1,0,0,0,6,6,0,3,6,
1,5,7,8,3,6,0,1,1,
1,6,4,5,0,6,8,5,1,

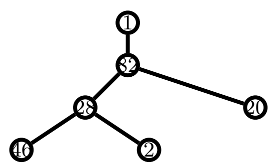
Stabilizer chain

Level	Base pt	Orbit length	Subgroup order
0	0	63	6048
1	1	6	96
2	2	1	16
3	3	16	16

Basic Orbit 0

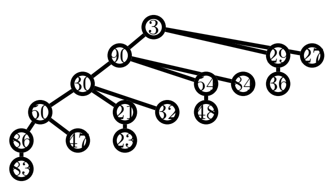


Basic Orbit 1



Basic Orbit 2

Basic Orbit 3



The base has length 4
The basic orbits are:
Basic orbit 0 is orbit of 0 of length 91
Basic orbit 1 is orbit of 1 of length 90
Basic orbit 2 is orbit of 2 of length 81
Basic orbit 3 is orbit of 3 of length 64

H Classification of linear $[8, 4, 4]$ codes over $\mathbb{F}_2$

H.1 The field of order 2

The field $\mathbb{F}_2$:
$Z_i = \log_\alpha(1 + \alpha^i)$

i	γ_i	$-\gamma_i$	γ_i^{-1}	$\log_\alpha(\gamma_i)$	α^i	Z_i
0	$0 = 0$	0	DNE	DNE	1	DNE
1	$1 = 1$	1	1	0	1	DNE

H.2 The group $\mathrm{PGL}(4, 2)$

Group action $\mathrm{PGL}(4, 2)$ of degree 15
Group order 20160
tl$=15, 14, 12, 8,$
Base: $(0, 1, 2, 3)$
Strong generators for a group of order 20160:

$$\begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 0&0&1&0\\ 1&0&0&1 \end{bmatrix}, \begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 0&0&1&0\\ 0&1&0&1 \end{bmatrix}, \begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 0&0&1&0\\ 0&0&1&1 \end{bmatrix},$$

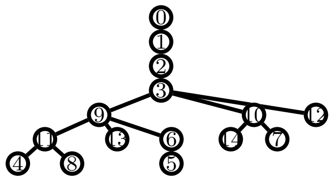
$$\begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 0&0&0&1\\ 0&0&1&0 \end{bmatrix}, \begin{bmatrix} 1&0&0&0\\ 0&0&1&0\\ 0&1&0&0\\ 0&0&0&1 \end{bmatrix}, \begin{bmatrix} 0&1&0&0\\ 1&0&0&0\\ 0&0&1&0\\ 0&0&0&1 \end{bmatrix}$$

1,0,0,0,0,1,0,0,0,0,1,0,1,0,0,1,
1,0,0,0,0,1,0,0,0,0,1,0,0,1,0,1,
1,0,0,0,0,1,0,0,0,0,1,0,0,0,1,1,
1,0,0,0,0,1,0,0,0,0,0,1,0,0,1,0,
1,0,0,0,0,0,1,0,0,1,0,0,0,0,0,1,
0,1,0,0,1,0,0,0,0,0,1,0,0,0,0,1,

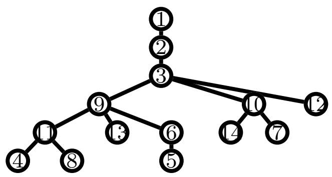
Stabilizer chain

Level	Base pt	Orbit length	Subgroup order
0	0	15	20160
1	1	14	1344
2	2	12	96
3	3	8	8

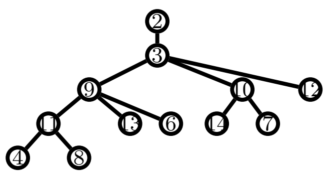
Basic Orbit 0



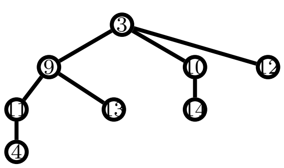
Basic Orbit 1



Basic Orbit 2



Basic Orbit 3



i	P_i
0	$(1,0,0,0)$
1	$(0,1,0,0)$
2	$(0,0,1,0)$
3	$(0,0,0,1)$
4	$(1,1,1,1)$
5	$(1,1,0,0)$
6	$(1,0,1,0)$
7	$(0,1,1,0)$
8	$(1,1,1,0)$
9	$(1,0,0,1)$

,

i	P_i
10	$(0,1,0,1)$
11	$(1,1,0,1)$
12	$(0,0,1,1)$
13	$(1,0,1,1)$
14	$(0,1,1,1)$

Poset classification up to depth 8

H.3 The orbits

H.4 Number of orbits at depth

Depth	Nb of orbits
0	1
1	1
2	1
3	1
4	2
5	2
6	1
7	1
8	1

H.5 Orbit representatives: overview

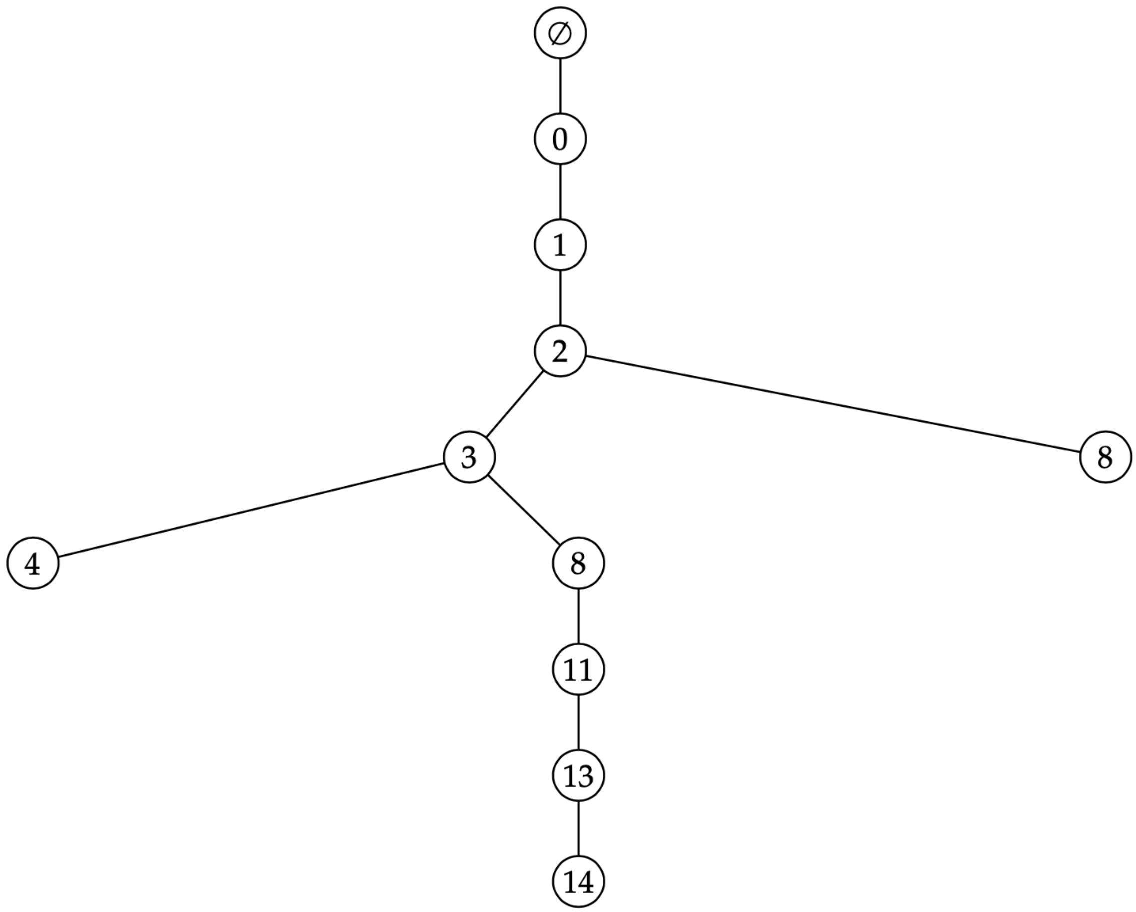
N = node
D = depth or level
O = orbit with a level
Rep = orbit representative
SO = (order of stabilizer, orbit length)
L = number of live points
F = number of flags

Gen = number of generators for the stabilizer of the orbit rep.

Table 15: Orbit Representatives

N	D	O	Rep	SO	L	F	Gen
0	0	0	{ }	(20160, 1)	15	1	6
1	1	0	{ 0 }	(1344, 15)	14	1	6
2	2	0	{ 0, 1 }	(192, 105)	12	1	6
3	3	0	{ 0, 1, 2 }	(48, 420)	9	2	6
4	4	0	{ 0, 1, 2, 3 }	(24, 840)	5	2	5
5	4	1	{ 0, 1, 2, 8 }	(192, 105)	8	0	9
6	5	0	{ 0, 1, 2, 3, 4 }	(120, 168)	0	0	7
7	5	1	{ 0, 1, 2, 3, 8 }	(24, 840)	3	1	5
8	6	0	{ 0, 1, 2, 3, 8, 11 }	(48, 420)	2	1	5
9	7	0	{ 0, 1, 2, 3, 8, 11, 13 }	(168, 120)	1	1	6
10	8	0	{ 0, 1, 2, 3, 8, 11, 13, 14 }	(1344, 15)			7

H.6 The poset of orbits



I Stabilizers and Schreier trees

I.1 Stabilizers and Schreier trees at level 0

Node 0 at Level 0 Orbit 0 / 1

$$\{\}_{20160}$$

Strong generators for a group of order 20160:

$$\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\1&0&0&1\end{bmatrix}, \begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&1&0&1\end{bmatrix}, \begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&1&1\end{bmatrix},$$

$$\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&0&1\\0&0&1&0\end{bmatrix}, \begin{bmatrix}1&0&0&0\\0&0&1&0\\0&1&0&0\\0&0&0&1\end{bmatrix}, \begin{bmatrix}0&1&0&0\\1&0&0&0\\0&0&1&0\\0&0&0&1\end{bmatrix}$$

1,0,0,0,0,1,0,0,0,0,1,0,1,0,0,1,
1,0,0,0,0,1,0,0,0,0,1,0,0,1,0,1,
1,0,0,0,0,1,0,0,0,0,1,0,0,0,1,1,
1,0,0,0,0,1,0,0,0,0,0,1,0,0,1,0,
1,0,0,0,0,0,1,0,0,1,0,0,0,0,0,1,
0,1,0,0,1,0,0,0,0,0,1,0,0,0,0,1,
There are 1 extensions
Number of generators 6
Generators for the Schreier trees:
Generators for a group of order 20160:

$$\begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 0&0&1&0\\ 1&0&0&1 \end{bmatrix} \begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 0&0&1&0\\ 0&1&0&1 \end{bmatrix} \begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 0&0&1&0\\ 0&0&1&1 \end{bmatrix} \begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 0&0&0&1\\ 0&0&1&0 \end{bmatrix}$$

$$\begin{bmatrix} 1&0&0&0\\ 0&0&1&0\\ 0&1&0&0\\ 0&0&0&1 \end{bmatrix} \begin{bmatrix} 0&1&0&0\\ 1&0&0&0\\ 0&0&1&0\\ 0&0&0&1 \end{bmatrix}$$

1,0,0,0,0,1,0,0,0,0,1,0,1,0,0,1,
1,0,0,0,0,1,0,0,0,0,1,0,0,1,0,1,
1,0,0,0,0,1,0,0,0,0,1,0,0,0,1,1,
1,0,0,0,0,1,0,0,0,0,0,1,0,0,1,0,
1,0,0,0,0,0,1,0,0,1,0,0,0,0,0,1,
0,1,0,0,1,0,0,0,0,0,1,0,0,0,0,1,
Orbit 0 / 1: Point 0 lies in an orbit of length 14 with average word length 5.28571 $H_6 = 2.40214$, $\Delta = 2.88357$

Node 0 at Level 0 Orbit 0 / 1 Tree 0 / 1

Number of generators 6

Extension number 0
Orbit representative 0
Flag orbit length 15
Flag orbit is defining new orbit 1 at level 1

I.2 Stabilizers and Schreier trees at level 1

Node 1 at Level 1 Orbit 0 / 1

$$\{0\}_{1344}$$

Strong generators for a group of order 1344:

$$\begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 0&0&1&0\\ 1&0&0&1 \end{bmatrix}, \begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 0&0&1&0\\ 1&0&1&1 \end{bmatrix}, \begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 0&0&1&0\\ 0&1&1&1 \end{bmatrix},$$

$$\begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 0&0&0&1\\ 0&0&1&0 \end{bmatrix}, \begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 1&1&1&0\\ 0&0&0&1 \end{bmatrix}, \begin{bmatrix} 1&0&0&0\\ 0&0&1&0\\ 0&1&0&0\\ 0&0&0&1 \end{bmatrix}$$

1,0,0,0,0,1,0,0,0,0,1,0,1,0,0,1,
1,0,0,0,0,1,0,0,0,0,1,0,1,0,1,1,
1,0,0,0,0,1,0,0,0,0,1,0,0,1,1,1,
1,0,0,0,0,1,0,0,0,0,0,1,0,0,1,0,
1,0,0,0,0,1,0,0,1,1,1,0,0,0,0,1,
1,0,0,0,0,0,1,0,0,1,0,0,0,0,0,1,
There are 1 extensions
Number of generators 6
Generators for the Schreier trees:
Generators for a group of order 1344:

$$\begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 0&0&1&0\\ 1&0&0&1 \end{bmatrix} \begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 0&0&1&0\\ 1&0&1&1 \end{bmatrix} \begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 0&0&1&0\\ 0&1&1&1 \end{bmatrix} \begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 0&0&0&1\\ 0&0&1&0 \end{bmatrix}$$

$$\begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 1&1&1&0\\ 0&0&0&1 \end{bmatrix} \begin{bmatrix} 1&0&0&0\\ 0&0&1&0\\ 0&1&0&0\\ 0&0&0&1 \end{bmatrix}$$

1,0,0,0,0,1,0,0,0,0,1,0,1,0,0,1,
1,0,0,0,0,1,0,0,0,0,1,0,1,0,1,1,
1,0,0,0,0,1,0,0,0,0,1,0,0,1,1,1,
1,0,0,0,0,1,0,0,0,0,0,1,0,0,1,0,
1,0,0,0,0,1,0,0,1,1,1,0,0,0,0,1,
1,0,0,0,0,0,1,0,0,1,0,0,0,0,0,1,
Orbit 0 / 1: Point 1 lies in an orbit of length 14 with average word length 4.07143 $H_6 = 2.25647$, $\Delta = 1.81496$

Node 1 at Level 1 Orbit 0 / 1 Tree 0 / 1

Number of generators 6

Extension number 0
Orbit representative 1
Flag orbit length 14
Flag orbit is defining new orbit 2 at level 2

I.3 Stabilizers and Schreier trees at level 2

Node 2 at Level 2 Orbit 0 / 1

$$\{0,1\}_{192}$$

Strong generators for a group of order 192:

$$\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\1&0&1&1\end{bmatrix}, \begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&1&1&1\end{bmatrix}, \begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&1&0&1\end{bmatrix},$$

$$\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&0&1\\0&0&1&0\end{bmatrix}, \begin{bmatrix}1&0&0&0\\0&1&0&0\\0&1&1&0\\0&0&0&1\end{bmatrix}, \begin{bmatrix}0&1&0&0\\1&0&0&0\\0&1&0&1\\1&0&1&0\end{bmatrix}$$

1,0,0,0,0,1,0,0,0,0,1,0,1,0,1,1,
1,0,0,0,0,1,0,0,0,0,1,0,0,1,1,1,
1,0,0,0,0,1,0,0,0,0,1,0,0,1,0,1,
1,0,0,0,0,1,0,0,0,0,0,1,0,0,1,0,
1,0,0,0,0,1,0,0,0,1,1,0,0,0,0,1,
0,1,0,0,1,0,0,0,0,1,0,1,1,0,1,0,
There are 1 extensions
Number of generators 6
Generators for the Schreier trees:
Generators for a group of order 192:

$$\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\1&0&1&1\end{bmatrix} \begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&1&1&1\end{bmatrix} \begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&1&0&1\end{bmatrix} \begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&0&1\\0&0&1&0\end{bmatrix}$$

$$\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&1&1&0\\0&0&0&1\end{bmatrix} \begin{bmatrix}0&1&0&0\\1&0&0&0\\0&1&0&1\\1&0&1&0\end{bmatrix}$$

1,0,0,0,0,1,0,0,0,0,1,0,1,0,1,1,
1,0,0,0,0,1,0,0,0,0,1,0,0,1,1,1,

1,0,0,0,0,1,0,0,0,0,1,0,0,1,0,1,
1,0,0,0,0,1,0,0,0,0,0,1,0,0,1,0,
1,0,0,0,0,1,0,0,0,1,1,0,0,0,0,1,
0,1,0,0,1,0,0,0,0,1,0,1,1,0,1,0,
Orbit 0 / 1: Point 2 lies in an orbit of length 12 with average word length 2.75 $H_6 = 1.95144$, $\Delta = 0.798562$

Node 2 at Level 2 Orbit 0 / 1 Tree 0 / 1

Number of generators 6

Extension number 0
Orbit representative 2
Flag orbit length 12
Flag orbit is defining new orbit 3 at level 3

I.4 Stabilizers and Schreier trees at level 3

Node 3 at Level 3 Orbit 0 / 1

$$\{0,1,2\}_{48}$$

Strong generators for a group of order 48:

$$\begin{bmatrix} 1&0&0&0 \\ 0&1&0&0 \\ 0&0&1&0 \\ 1&1&0&1 \end{bmatrix}, \begin{bmatrix} 1&0&0&0 \\ 0&1&0&0 \\ 0&0&1&0 \\ 0&1&0&1 \end{bmatrix}, \begin{bmatrix} 1&0&0&0 \\ 0&1&0&0 \\ 0&0&1&0 \\ 1&0&1&1 \end{bmatrix},$$

$$\begin{bmatrix} 1&0&0&0 \\ 0&0&1&0 \\ 0&1&0&0 \\ 0&0&0&1 \end{bmatrix}, \begin{bmatrix} 0&1&0&0 \\ 1&0&0&0 \\ 0&0&1&0 \\ 1&0&0&1 \end{bmatrix}, \begin{bmatrix} 0&0&1&0 \\ 1&0&0&0 \\ 0&1&0&0 \\ 1&1&1&1 \end{bmatrix}$$

1,0,0,0,0,1,0,0,0,0,1,0,1,1,0,1,
1,0,0,0,0,1,0,0,0,0,1,0,0,1,0,1,
1,0,0,0,0,1,0,0,0,0,1,0,1,0,1,1,
1,0,0,0,0,0,1,0,0,1,0,0,0,0,0,1,
0,1,0,0,1,0,0,0,0,0,1,0,1,0,0,1,
0,0,1,0,1,0,0,0,0,1,0,0,1,1,1,1,
There are 2 extensions
Number of generators 6
Generators for the Schreier trees:

Generators for a group of order 48:

$$\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\1&1&0&1\end{bmatrix}\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&1&0&1\end{bmatrix}\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\1&0&1&1\end{bmatrix}\begin{bmatrix}1&0&0&0\\0&0&1&0\\0&1&0&0\\0&0&0&1\end{bmatrix}$$

$$\begin{bmatrix}0&1&0&0\\1&0&0&0\\0&0&1&0\\1&0&0&1\end{bmatrix}\begin{bmatrix}0&0&1&0\\1&0&0&0\\0&1&0&0\\1&1&1&1\end{bmatrix}$$

1,0,0,0,0,1,0,0,0,0,1,0,1,1,0,1,
1,0,0,0,0,1,0,0,0,0,1,0,0,1,0,1,
1,0,0,0,0,1,0,0,0,0,1,0,1,0,1,1,
1,0,0,0,0,0,1,0,0,1,0,0,0,0,0,1,
0,1,0,0,1,0,0,0,0,0,1,0,1,0,0,1,
0,0,1,0,1,0,0,0,0,1,0,0,1,1,1,1,
Orbit 0 / 2: Point 3 lies in an orbit of length 8 with average word length 2.125 $H_6 = 1.58125$, $\Delta = 0.543754$
Orbit 1 / 2: Point 8 lies in an orbit of length 1 with average word length 1 $H_6 = 0$, $\Delta = 1$

Node 3 at Level 3 Orbit 0 / 1 Tree 0 / 2

Number of generators 6

Extension number 0
Orbit representative 3
Flag orbit length 8
Flag orbit is defining new orbit 4 at level 4

Node 3 at Level 3 Orbit 0 / 1 Tree 1 / 2

Number of generators 6

Extension number 1
Orbit representative 8
Flag orbit length 1
Flag orbit is defining new orbit 5 at level 4

I.5 Stabilizers and Schreier trees at level 4

Node 4 at Level 4 Orbit 0 / 2

$$\{0,1,2,3\}_{24}$$

Strong generators for a group of order 24:

$$\begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 0&0&0&1\\ 0&0&1&0 \end{bmatrix}, \begin{bmatrix} 1&0&0&0\\ 0&0&1&0\\ 0&1&0&0\\ 0&0&0&1 \end{bmatrix}, \begin{bmatrix} 1&0&0&0\\ 0&0&0&1\\ 0&0&1&0\\ 0&1&0&0 \end{bmatrix},$$

$$\begin{bmatrix} 0&0&1&0\\ 1&0&0&0\\ 0&1&0&0\\ 0&0&0&1 \end{bmatrix}, \begin{bmatrix} 0&0&0&1\\ 0&1&0&0\\ 1&0&0&0\\ 0&0&1&0 \end{bmatrix}$$

1,0,0,0,0,1,0,0,0,0,0,1,0,0,1,0,
1,0,0,0,0,0,1,0,0,1,0,0,0,0,0,1,
1,0,0,0,0,0,0,1,0,0,1,0,0,1,0,0,
0,0,1,0,1,0,0,0,0,1,0,0,0,0,0,1,
0,0,0,1,0,1,0,0,1,0,0,0,0,0,1,0,
There are 2 extensions
Number of generators 5
Generators for the Schreier trees:
Generators for a group of order 24:

$$\begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 0&0&0&1\\ 0&0&1&0 \end{bmatrix} \begin{bmatrix} 1&0&0&0\\ 0&0&1&0\\ 0&1&0&0\\ 0&0&0&1 \end{bmatrix} \begin{bmatrix} 1&0&0&0\\ 0&0&0&1\\ 0&0&1&0\\ 0&1&0&0 \end{bmatrix} \begin{bmatrix} 0&0&1&0\\ 1&0&0&0\\ 0&1&0&0\\ 0&0&0&1 \end{bmatrix}$$

$$\begin{bmatrix} 0&0&0&1\\ 0&1&0&0\\ 1&0&0&0\\ 0&0&1&0 \end{bmatrix}$$

1,0,0,0,0,1,0,0,0,0,0,1,0,0,1,0,
1,0,0,0,0,0,1,0,0,1,0,0,0,0,0,1,
1,0,0,0,0,0,0,1,0,0,1,0,0,1,0,0,
0,0,1,0,1,0,0,0,0,1,0,0,0,0,0,1,
0,0,0,1,0,1,0,0,1,0,0,0,0,0,1,0,
Orbit 0 / 2: Point 4 lies in an orbit of length 1 with average word length 1 $H_5 = 0$, $\Delta = 1$
Orbit 1 / 2: Point 8 lies in an orbit of length 4 with average word length 2 $H_5 = 1.29203$, $\Delta = 0.70797$

Node 4 at Level 4 Orbit 0 / 2 Tree 0 / 2

Number of generators 5

Extension number 0
Orbit representative 4
Flag orbit length 1
Flag orbit is defining new orbit 6 at level 5

Node 4 at Level 4 Orbit 0 / 2 Tree 1 / 2

Number of generators 5

Extension number 1
Orbit representative 8
Flag orbit length 4
Flag orbit is defining new orbit 7 at level 5

Node 5 at Level 4 Orbit 1 / 2

$$\{0,1,2,8\}_{192}$$

Strong generators for a group of order 192:

$$\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\1&1&0&1\end{bmatrix},\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&1&0&1\end{bmatrix},\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\1&0&1&1\end{bmatrix},$$

$$\begin{bmatrix}1&0&0&0\\0&1&0&0\\1&1&1&0\\0&0&1&1\end{bmatrix},\begin{bmatrix}1&0&0&0\\0&0&1&0\\0&1&0&0\\0&0&0&1\end{bmatrix},\begin{bmatrix}1&0&0&0\\1&1&1&0\\0&1&0&0\\1&0&1&1\end{bmatrix},$$

$$\begin{bmatrix}0&1&0&0\\1&0&0&0\\0&0&1&0\\1&0&0&1\end{bmatrix},\begin{bmatrix}0&0&1&0\\1&0&0&0\\0&1&0&0\\1&1&1&1\end{bmatrix},\begin{bmatrix}1&1&1&0\\0&0&1&0\\0&1&0&0\\0&0&0&1\end{bmatrix}$$

1,0,0,0,0,1,0,0,0,0,1,0,1,1,0,1,
1,0,0,0,0,1,0,0,0,0,1,0,0,1,0,1,
1,0,0,0,0,1,0,0,0,0,1,0,1,0,1,1,
1,0,0,0,0,1,0,0,1,1,1,0,0,0,1,1,
1,0,0,0,0,0,1,0,0,1,0,0,0,0,0,1,
1,0,0,0,1,1,1,0,0,1,0,0,1,0,1,1,
0,1,0,0,1,0,0,0,0,0,1,0,1,0,0,1,
0,0,1,0,1,0,0,0,0,1,0,0,1,1,1,1,
1,1,1,0,0,0,1,0,0,1,0,0,0,0,0,1,
There are 0 extensions
Number of generators 9

Generators for the Schreier trees:
Generators for a group of order 192:

$$\begin{bmatrix} 1&0&0&0 \\ 0&1&0&0 \\ 0&0&1&0 \\ 1&1&0&1 \end{bmatrix} \begin{bmatrix} 1&0&0&0 \\ 0&1&0&0 \\ 0&0&1&0 \\ 0&1&0&1 \end{bmatrix} \begin{bmatrix} 1&0&0&0 \\ 0&1&0&0 \\ 0&0&1&0 \\ 1&0&1&1 \end{bmatrix} \begin{bmatrix} 1&0&0&0 \\ 0&1&0&0 \\ 1&1&1&0 \\ 0&0&1&1 \end{bmatrix}$$

$$\begin{bmatrix} 1&0&0&0 \\ 0&0&1&0 \\ 0&1&0&0 \\ 0&0&0&1 \end{bmatrix} \begin{bmatrix} 1&0&0&0 \\ 1&1&1&0 \\ 0&1&0&0 \\ 1&0&1&1 \end{bmatrix} \begin{bmatrix} 0&1&0&0 \\ 1&0&0&0 \\ 0&0&1&0 \\ 1&0&0&1 \end{bmatrix} \begin{bmatrix} 0&0&1&0 \\ 1&0&0&0 \\ 0&1&0&0 \\ 1&1&1&1 \end{bmatrix}$$

$$\begin{bmatrix} 1&1&1&0 \\ 0&0&1&0 \\ 0&1&0&0 \\ 0&0&0&1 \end{bmatrix}$$

1,0,0,0,0,1,0,0,0,0,1,0,1,1,0,1,
1,0,0,0,0,1,0,0,0,0,1,0,0,1,0,1,
1,0,0,0,0,1,0,0,0,0,1,0,1,0,1,1,
1,0,0,0,0,1,0,0,1,1,1,0,0,0,1,1,
1,0,0,0,0,0,1,0,0,1,0,0,0,0,0,1,
1,0,0,0,1,1,1,0,0,1,0,0,1,0,1,1,
0,1,0,0,1,0,0,0,0,0,1,0,1,0,0,1,
0,0,1,0,1,0,0,0,0,1,0,0,1,1,1,1,
1,1,1,0,0,0,1,0,0,1,0,0,0,0,0,1,
Orbit 0 / 1: Point 3 lies in an orbit of length 8 with average word length 2 $H_9 = 1.26186$, $\Delta = 0.73814$

Node 5 at Level 4 Orbit 1 / 2 Tree 0 / 1

Number of generators 9

Cannot find an extension for point 3

I.6 Stabilizers and Schreier trees at level 5

Node 6 at Level 5 Orbit 0 / 2

$$\{0, 1, 2, 3, 4\}_{120}$$

Strong generators for a group of order 120:

$$\begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 0&0&1&0\\ 1&1&1&1 \end{bmatrix}, \begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 0&0&0&1\\ 0&0&1&0 \end{bmatrix}, \begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 1&1&1&1\\ 0&0&1&0 \end{bmatrix},$$

$$\begin{bmatrix} 1&0&0&0\\ 0&0&1&0\\ 0&1&0&0\\ 0&0&0&1 \end{bmatrix}, \begin{bmatrix} 1&0&0&0\\ 0&0&0&1\\ 0&0&1&0\\ 0&1&0&0 \end{bmatrix}, \begin{bmatrix} 0&0&1&0\\ 1&0&0&0\\ 0&1&0&0\\ 0&0&0&1 \end{bmatrix},$$

$$\begin{bmatrix} 1&1&1&1\\ 0&1&0&0\\ 0&0&0&1\\ 0&0&1&0 \end{bmatrix}$$

1,0,0,0,0,1,0,0,0,0,1,0,1,1,1,1,
1,0,0,0,0,1,0,0,0,0,0,1,0,0,1,0,
1,0,0,0,0,1,0,0,1,1,1,1,0,0,1,0,
1,0,0,0,0,0,1,0,0,1,0,0,0,0,0,1,
1,0,0,0,0,0,0,1,0,0,1,0,0,1,0,0,
0,0,1,0,1,0,0,0,0,1,0,0,0,0,0,1,
1,1,1,1,0,1,0,0,0,0,0,1,0,0,1,0,
There are 0 extensions
Number of generators 7
Generators for the Schreier trees:
Generators for a group of order 120:

$$\begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 0&0&1&0\\ 1&1&1&1 \end{bmatrix} \begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 0&0&0&1\\ 0&0&1&0 \end{bmatrix} \begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 1&1&1&1\\ 0&0&1&0 \end{bmatrix} \begin{bmatrix} 1&0&0&0\\ 0&0&1&0\\ 0&1&0&0\\ 0&0&0&1 \end{bmatrix}$$

$$\begin{bmatrix} 1&0&0&0\\ 0&0&0&1\\ 0&0&1&0\\ 0&1&0&0 \end{bmatrix} \begin{bmatrix} 0&0&1&0\\ 1&0&0&0\\ 0&1&0&0\\ 0&0&0&1 \end{bmatrix} \begin{bmatrix} 1&1&1&1\\ 0&1&0&0\\ 0&0&0&1\\ 0&0&1&0 \end{bmatrix}$$

1,0,0,0,0,1,0,0,0,0,1,0,1,1,1,1,
1,0,0,0,0,1,0,0,0,0,0,1,0,0,1,0,
1,0,0,0,0,1,0,0,1,1,1,1,0,0,1,0,
1,0,0,0,0,0,1,0,0,1,0,0,0,0,0,1,
1,0,0,0,0,0,0,1,0,0,1,0,0,1,0,0,
0,0,1,0,1,0,0,0,0,1,0,0,0,0,0,1,
1,1,1,1,0,1,0,0,0,0,0,1,0,0,1,0,

Node 7 at Level 5 Orbit 1 / 2

$$\{0, 1, 2, 3, 8\}_{24}$$

Strong generators for a group of order 24:

$$\begin{bmatrix} 1&0&0&0 \\ 0&1&0&0 \\ 1&1&1&0 \\ 0&0&0&1 \end{bmatrix}, \begin{bmatrix} 1&0&0&0 \\ 0&0&1&0 \\ 0&1&0&0 \\ 0&0&0&1 \end{bmatrix}, \begin{bmatrix} 1&0&0&0 \\ 1&1&1&0 \\ 0&1&0&0 \\ 0&0&0&1 \end{bmatrix},$$

$$\begin{bmatrix} 0&0&1&0 \\ 1&0&0&0 \\ 0&1&0&0 \\ 0&0&0&1 \end{bmatrix}, \begin{bmatrix} 1&1&1&0 \\ 1&0&0&0 \\ 0&1&0&0 \\ 0&0&0&1 \end{bmatrix}$$

1,0,0,0,0,1,0,0,1,1,1,0,0,0,0,1,
1,0,0,0,0,0,1,0,0,1,0,0,0,0,0,1,
1,0,0,0,1,1,1,0,0,1,0,0,0,0,0,1,
0,0,1,0,1,0,0,0,0,1,0,0,0,0,0,1,
1,1,1,0,1,0,0,0,0,1,0,0,0,0,0,1,
There are 1 extensions
Number of generators 5
Generators for the Schreier trees:
Generators for a group of order 24:

$$\begin{bmatrix} 1&0&0&0 \\ 0&1&0&0 \\ 1&1&1&0 \\ 0&0&0&1 \end{bmatrix} \begin{bmatrix} 1&0&0&0 \\ 0&0&1&0 \\ 0&1&0&0 \\ 0&0&0&1 \end{bmatrix} \begin{bmatrix} 1&0&0&0 \\ 1&1&1&0 \\ 0&1&0&0 \\ 0&0&0&1 \end{bmatrix} \begin{bmatrix} 0&0&1&0 \\ 1&0&0&0 \\ 0&1&0&0 \\ 0&0&0&1 \end{bmatrix}$$

$$\begin{bmatrix} 1&1&1&0 \\ 1&0&0&0 \\ 0&1&0&0 \\ 0&0&0&1 \end{bmatrix}$$

1,0,0,0,0,1,0,0,1,1,1,0,0,0,0,1,
1,0,0,0,0,0,1,0,0,1,0,0,0,0,0,1,
1,0,0,0,1,1,1,0,0,1,0,0,0,0,0,1,
0,0,1,0,1,0,0,0,0,1,0,0,0,0,0,1,
1,1,1,0,1,0,0,0,0,1,0,0,0,0,0,1,
Orbit 0 / 1: Point 11 lies in an orbit of length 3 with average word length 1.66667 $H_5 = 1$, $\Delta = 0.666667$

Node 7 at Level 5 Orbit 1 / 2 Tree 0 / 1

Number of generators 5

Extension number 0
Orbit representative 11
Flag orbit length 3
Flag orbit is defining new orbit 8 at level 6

I.7 Stabilizers and Schreier trees at level 6

Node 8 at Level 6 Orbit 0 / 1

$$\{0,1,2,3,8,11\}_{48}$$

Strong generators for a group of order 48:

$$\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\1&1&0&1\end{bmatrix},\begin{bmatrix}1&0&0&0\\0&1&0&0\\1&1&1&0\\0&0&0&1\end{bmatrix},\begin{bmatrix}1&0&0&0\\0&1&0&0\\1&1&0&1\\0&0&1&0\end{bmatrix},$$

$$\begin{bmatrix}1&1&1&0\\0&0&1&0\\1&0&0&0\\0&0&0&1\end{bmatrix},\begin{bmatrix}1&1&0&1\\0&0&0&1\\1&0&0&0\\0&0&1&0\end{bmatrix}$$

1,0,0,0,0,1,0,0,0,0,1,0,1,1,0,1,
1,0,0,0,0,1,0,0,1,1,1,0,0,0,0,1,
1,0,0,0,0,1,0,0,1,1,0,1,0,0,1,0,
1,1,1,0,0,0,1,0,1,0,0,0,0,0,0,1,
1,1,0,1,0,0,0,1,1,0,0,0,0,0,1,0,
There are 1 extensions
Number of generators 5
Generators for the Schreier trees:
Generators for a group of order 48:

$$\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\1&1&0&1\end{bmatrix}\begin{bmatrix}1&0&0&0\\0&1&0&0\\1&1&1&0\\0&0&0&1\end{bmatrix}\begin{bmatrix}1&0&0&0\\0&1&0&0\\1&1&0&1\\0&0&1&0\end{bmatrix}\begin{bmatrix}1&1&1&0\\0&0&1&0\\1&0&0&0\\0&0&0&1\end{bmatrix}$$

$$\begin{bmatrix}1&1&0&1\\0&0&0&1\\1&0&0&0\\0&0&1&0\end{bmatrix}$$

1,0,0,0,0,1,0,0,0,0,1,0,1,1,0,1,
1,0,0,0,0,1,0,0,1,1,1,0,0,0,0,1,
1,0,0,0,0,1,0,0,1,1,0,1,0,0,1,0,

1,1,1,0,0,0,1,0,1,0,0,0,0,0,0,1,
1,1,0,1,0,0,0,1,1,0,0,0,0,0,1,0,
Orbit 0 / 1: Point 13 lies in an orbit of length 2 with average word length 1.5 $H_5 = 0.682606$, $\Delta = 0.817394$

Node 8 at Level 6 Orbit 0 / 1 Tree 0 / 1

Number of generators 5

Extension number 0
Orbit representative 13
Flag orbit length 2
Flag orbit is defining new orbit 9 at level 7

I.8 Stabilizers and Schreier trees at level 7

Node 9 at Level 7 Orbit 0 / 1

$$\{0, 1, 2, 3, 8, 11, 13\}_{168}$$

Strong generators for a group of order 168:

$$\begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 0&0&0&1\\ 0&0&1&0 \end{bmatrix}, \begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 1&1&1&0\\ 1&1&0&1 \end{bmatrix}, \begin{bmatrix} 1&0&0&0\\ 0&0&1&0\\ 1&0&1&1\\ 1&1&1&0 \end{bmatrix},$$

$$\begin{bmatrix} 0&0&0&1\\ 1&1&0&1\\ 1&0&0&0\\ 0&0&1&0 \end{bmatrix}, \begin{bmatrix} 1&1&0&1\\ 0&0&0&1\\ 1&0&0&0\\ 1&1&1&0 \end{bmatrix}, \begin{bmatrix} 1&0&1&1\\ 1&1&1&0\\ 1&1&0&1\\ 0&1&0&0 \end{bmatrix}$$

1,0,0,0,0,1,0,0,0,0,0,1,0,0,1,0,
1,0,0,0,0,1,0,0,1,1,1,0,1,1,0,1,
1,0,0,0,0,0,1,0,1,0,1,1,1,1,1,0,
0,0,0,1,1,1,0,1,1,0,0,0,0,0,1,0,
1,1,0,1,0,0,0,1,1,0,0,0,1,1,1,0,
1,0,1,1,1,1,1,0,1,1,0,1,0,1,0,0,
There are 1 extensions
Number of generators 6
Generators for the Schreier trees:
Generators for a group of order 168:

$$\begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 0&0&0&1\\ 0&0&1&0 \end{bmatrix} \begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 1&1&1&0\\ 1&1&0&1 \end{bmatrix} \begin{bmatrix} 1&0&0&0\\ 0&0&1&0\\ 1&0&1&1\\ 1&1&1&0 \end{bmatrix} \begin{bmatrix} 0&0&0&1\\ 1&1&0&1\\ 1&0&0&0\\ 0&0&1&0 \end{bmatrix}$$

$$\begin{bmatrix} 1&1&0&1\\ 0&0&0&1\\ 1&0&0&0\\ 1&1&1&0 \end{bmatrix} \begin{bmatrix} 1&0&1&1\\ 1&1&1&0\\ 1&1&0&1\\ 0&1&0&0 \end{bmatrix}$$

1,0,0,0,0,1,0,0,0,0,0,1,0,0,1,0,
1,0,0,0,0,1,0,0,1,1,1,0,1,1,0,1,
1,0,0,0,0,0,1,0,1,0,1,1,1,1,1,0,
0,0,0,1,1,1,0,1,1,0,0,0,0,0,1,0,
1,1,0,1,0,0,0,1,1,0,0,0,1,1,1,0,
1,0,1,1,1,1,1,0,1,1,0,1,0,1,0,0,
Orbit 0 / 1: Point 14 lies in an orbit of length 1 with average word length 1 $H_6 = 0$, $\Delta = 1$

Node 9 at Level 7 Orbit 0 / 1 Tree 0 / 1

Number of generators 6

Extension number 0
Orbit representative 14
Flag orbit length 1
Flag orbit is defining new orbit 10 at level 8

I.9 Stabilizers and Schreier trees at level 8

Node 10 at Level 8 Orbit 0 / 1

$$\{0, 1, 2, 3, 8, 11, 13, 14\}_{1344}$$

Strong generators for a group of order 1344:

$$\begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 0&0&1&0\\ 1&1&0&1 \end{bmatrix}, \begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 0&0&1&0\\ 0&1&1&1 \end{bmatrix}, \begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 0&0&0&1\\ 0&0&1&0 \end{bmatrix},$$

$$\begin{bmatrix} 1&0&0&0\\ 0&1&0&0\\ 1&1&1&0\\ 1&1&0&1 \end{bmatrix}, \begin{bmatrix} 1&0&0&0\\ 0&0&1&0\\ 1&0&1&1\\ 1&1&1&0 \end{bmatrix}, \begin{bmatrix} 0&0&0&1\\ 1&1&0&1\\ 1&0&0&0\\ 0&0&1&0 \end{bmatrix},$$

$$\begin{bmatrix} 0&1&1&1\\ 1&1&0&1\\ 1&1&1&0\\ 0&0&0&1 \end{bmatrix}$$

1,0,0,0,0,1,0,0,0,0,1,0,1,1,0,1,
1,0,0,0,0,1,0,0,0,0,1,0,0,1,1,1,
1,0,0,0,0,1,0,0,0,0,0,1,0,0,1,0,
1,0,0,0,0,1,0,0,1,1,1,0,1,1,0,1,
1,0,0,0,0,0,1,0,1,0,1,1,1,1,1,0,
0,0,0,1,1,1,0,1,1,0,0,0,0,0,1,0,
0,1,1,1,1,1,0,1,1,1,1,0,0,0,0,1,
There are 0 extensions
Number of generators 7

I.10 Summary of Orbits

	Rep	#	Ago	Objects
0	0	1	6	0

Ago distribution:

$$6$$

Group order 6 appears for the following 1 classes: {0}

Orbit 0 / 1

Orbit 0 / 1 stored at 0 is represented by input object 0 and appears 1 times:
set of points of size 18: (1, 67, 78, 57, 90, 16, 60, 93, 28, 127, 62, 95, 30, 129, 43, 120, 33, 132)
Group order 6
Stabilizer: Strong generators for a group of order 6:

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 10 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 1 & 8 \\ 5 & 9 & 5 \\ 8 & 1 & 1 \end{bmatrix}$$

1,0,0,0,10,0,0,0,1,
1,1,8,5,9,5,8,1,1,
This isomorphism type appears 1 times, namely for the following 1 input objects: {0}

$\rightarrow$	133_1	1_2
134_0	12	0

$\downarrow$	133_1	1_2
134_0	12	19

$\rightarrow$	133_1	1_2
18_0	12	1
115_4	12	0
1_3	0	1

$\downarrow$	46_1	15_6	48_5	24_7	1_2
18_0	3	2	1	0	18
115_4	9	10	11	12	0
1_3	0	0	0	0	1

$\rightarrow$	46_1	15_6	48_5	24_7	1_2
12_0	8	1	3	0	1
6_8	7	3	2	0	1
9_4	5	0	3	4	0
12_{10}	4	2	2	4	0
36_{11}	4	1	4	3	0
12_{12}	4	0	6	2	0
27_9	3	2	5	2	0
12_{15}	3	1	7	1	0
1_{16}	3	0	9	0	0
3_{14}	2	2	8	0	0
3_{13}	1	6	3	2	0
1_3	0	0	0	0	1

J Orbiter directory structure

Here is a schematic diagram of the Orbiter directory structure:

```
orbiter
|-- ORBITER
|   |-- doc
|   |-- examples
|   |   |-- groups
|   |       |-- orthogonal
|   |-- src
|       |-- apps
|       |   |-- algebra
|       |   |-- arcs
|       |   |-- blt
|       |   |-- codes
|       |   |-- combinatorics
|       |   |-- graph_classify
|       |   |-- graph_theory
|       |   |-- groups
|       |   |-- linear_spaces
|       |   |-- main
```

```
|       |   |-- ovoid
|       |   |-- packing
|       |   |-- projective_space
|       |   |-- regular_ls
|       |   |-- semifield
|       |   |-- solver
|       |   |-- spread
|       |   |-- subspace_orbits
|       |   |-- surfaces
|       |   |-- test
|       |   |-- tools
|       |-- contrib
|       |-- lib
|           |-- DISCRETA
|           |-- classification
|           |   |-- classify
|           |   |-- poset_classification
|           |   |-- set_stabilizer
|           |-- foundations
|           |   |-- BitSet
|           |   |-- CUDA
|           |   |   |-- Linalg
|           |   |-- algebra_and_number_theory
|           |   |-- coding_theory
|           |   |-- combinatorics
|           |   |-- data_structures
|           |   |-- geometry
|           |   |   |-- DATA
|           |   |-- globals
|           |   |-- graph_theory
|           |   |   |-- Clique
|           |   |-- graph_theory_nauty
|           |   |-- graphics
|           |   |-- io_and_os
|           |   |-- solvers
|           |   |-- statistics
|           |-- group_actions
|           |   |-- actions
|           |   |-- data_structures
|           |   |-- groups
|           |   |-- induced_actions
|           |-- top_level
|               |-- algebra_and_number_theory
|               |-- combinatorics
|               |-- geometry
|               |-- isomorph
|               |-- orbits
```

```
66      |                 |-- solver
67      |-- bin
68
```

The bin directory on line 67 contains the executables. The directories on lines 9-29 contain the main source files of all Orbiter applications. The directories on lines 31-66 contain the Orbiter library. The application source files rely on the library. The directory contrib on line 30 contains makefiles which show how Orbiter applications can be used. The directory doc on line 3 contains this user's guide. The directory DATA on line 46 contains geometric data that is compiled into the Orbiter applications. The directory DISCRETA on line 32 contains legacy code from the DISCRETA project [3]. The directory grapth_theory_nauty on line 50 contains Brendan McKay's Nauty [16].

K The Orbiter executables

At present, Orbiter comes with the following 159 executables in the bin subdirectory.

```
BN_pair.out
a5_in_PSL.out
action_on_set_partitions.out
all_cliques.out (Section 8)
all_cycles.out (Section 8)
all_k_subsets.out
all_rainbow_cliques.out (Section 8)
analyze_projective_code.out (Section 10)
andre.out (Section 10)
arc_lifting_main.out
arcs_main.out
arcs_orderly.out
awss.out
bent.out
blt_main.out
borel.out
burnside.out
canonical_form.out (Section 10)
cayley.out (Section 8)
cayley_sym_n.out (Section 8)
cc2widor.out
cheat_sheet_GF.out (Section 2)
cheat_sheet_PG.out (Section 3)
classify_cubic_curves.out (Section 10)
code_cosets.out
codes.out (Section 9)
collect.out
colored_graph.out (Section 8)
concatenate_files.out
conjugacy_classes_sym_n.out
costas.out
create_BLT_set_main.out
create_element.out
create_element_of_order.out
create_file.out
create_graph.out (Section 8)
create_layered_graph_file.out (Section 8)
create_object.out
create_surface_main.out (Section 11)
deep_search.out
delandtsheer_doyen_main.out
desarguesian_spread.out (Section 10)
design.out
design_create_main.out
determine_conic.out (Section 10)
determine_cubic.out (Section 10)
determine_quadric.out (Section 10)
dio.out
distribution.out
dlx.out
draw_colored_graph.out (Section 8)
draw_graph.out (Section 8)
eigenstuff.out
exceptional_isomorphism_O4_main.out
factor_cyclotomic.out
ferdinand.out
field_plot.out
find_element.out
flag.out
get_poly.out
gl_classes.out
graph.out (Section 8)
```

group_ring.out
hadamard.out
hall_system_main.out
hermitian_points.out (Section 10)
hermitian_spreads_main.out (Section 10)
intersection.out
isomorph_testing.out
johnson_table.out (Section 8)
join_sets.out
k_arc_generator_main.out
k_arc_lifting.out
kramer_mesner.out
latex_table.out
layered_graph_main.out (Section 8)
linear_group.out (Section 6)
linear_set_main.out
long_orbit.out
loop.out
make_design.out
make_poster.out
matrix_rank.out
maxfit.out
memory_usage.out
missing_files.out
nauty.out (Section 8)
orthogonal.out
orthogonal_group.out (Section 6)
orthogonal_points.out (Section 10)
ovoid.out
packing.out
packing_main.out
packing_was_main.out
parameters.out
pentomino_5x5.out
plot_decomposition_matrix.out
plot_stats_on_graphs.out
plot_xy.out
points.out (Section 10)
polar.out (Section 10)
polynomial_orbits.out
poset_of_subsets.out
prepare_frames.out
process.out (Section 10)
puzzle.out
rainbow_cliques.out (Section 8)
random_permutation.out
rank_anything.out
rank_subsets_lex.out
read_orbiter_file.out
read_solutions.out
read_types.out
read_vector_and_extract_set.out
reflection.out
regular_ls.out
run_blt.out
run_lifting.out
scheduler.out
semifield_classify_main.out
semifield_main.out
simeon.out
solve_diophant.out
split.out
split_spreadsheet.out
spread_classify.out
spread_create.out
srg.out (Section 8)
study_surface.out
subprimitive.out
subspace_orbits_main.out (Section 7)
surface_classify.out (Section 11)
surfaces_arc_lifting_main.out
tao.out
tdo_print.out
tdo_refine.out
tdo_start.out
test_arc.out
test_hyperoval.out
test_longinteger.out
transpose.out
treedraw.out (Section 8)
unrank.out
widor.out
wreath_product.out

References

[1] Johannes André. Über nicht-Desarguessche Ebenen mit transitiver Translationsgruppe. *Math. Z.*, 60:156–186, 1954.

[2] Anton Betten, Michael Braun, Harald Fripertinger, Adalbert Kerber, Axel Kohnert, and Alfred Wassermann. *Error-correcting linear codes*, volume 18 of *Algorithms and Computation in Mathematics*. Springer-Verlag, Berlin, 2006. Classification by isometry and applications, With 1 CD-ROM (Windows and Linux).

[3] Anton Betten, Evi Haberberger, Reinhard Laue, and Alfred Wassermann. *DISCRETA – A program system for the construction t-designs*. Lehrstuhl II für Mathematik, Universität Bayreuth, 1999. `http://www.mathe2.uni-bayreuth.de/~discreta`.

[4] Anton Betten and Fatma Karaoğlu. Cubic surfaces over small finite fields. *Des. Codes Cryptogr.*, 87(4):931–953, 2019.

[5] Wieb Bosma, John Cannon, and Catherine Playoust. The Magma algebra system. I. The user language. *J. Symbolic Comput.*, 24(3-4):235–265, 1997. Computational algebra and number theory (London, 1993).

[6] Amie Bray. Algebraic Sets in Affine Space over Finite Fields, Term Paper, Combinatorics, Colorado State University, 2019.

[7] R. H. Bruck and R. C. Bose. The construction of translation planes from projective spaces. *J. Algebra*, 1:85–102, 1964.

[8] G. Butler. *Fundamental algorithms for permutation groups*, volume 559 of *Lecture Notes in Computer Science*. Springer-Verlag, Berlin, 1991.

[9] The GAP Group. *GAP – Groups, Algorithms, and Programming, Version 4.8.7*, 2017.

[10] D. G. Glynn and J. W. P. Hirschfeld. On the classification of geometric codes by polynomial functions. *Des. Codes Cryptogr.*, 6(3):189–204, 1995.

[11] Derek F. Holt, Bettina Eick, and Eamonn A. O'Brien. *Handbook of computational group theory*. Discrete Mathematics and its Applications (Boca Raton). Chapman & Hall/CRC, Boca Raton, FL, 2005.

[12] Fatma Karaoğlu. The cubic surfaces with twenty-seven lines over finite fields, Ph.D. thesis, University of Sussex, 2018.

[13] Wen-Ch'ing Winnie Li. Character sums and abelian Ramanujan graphs. *J. Number Theory*, 41(2):199–217, 1992. With an appendix by Ke Qin Feng and the author.

[14] A. Lubotzky, R. Phillips, and P. Sarnak. Ramanujan graphs. *Combinatorica*, 8(3):261–277, 1988.

[15] Maple 18. Maplesoft, a division of Waterloo Maple Inc., Waterloo, Ontario.

[16] Nauty User's Guide (Version 2.6), Brendan McKay, Australian National University, 2016.

[17] L. Schläfli. An attempt to determine the twenty-seven lines upon a surface of the third order and to divide such surfaces into species in reference to the reality of the lines upon the surface, *Quart. J. Math.* **2** (1858), 55–110.

[18] Ákos Seress. *Permutation group algorithms*, volume 152 of *Cambridge Tracts in Mathematics*. Cambridge University Press, Cambridge, 2003.

[19] Ibrahim A. I. Suleiman and Robert A. Wilson. The 2-modular characters of Conway's third group Co_3. *J. Symbolic Comput.*, 24(3-4):493–506, 1997.

[20] Magma system. Magma Calculator `http://magma.maths.usyd.edu.au/calc/`, accessed 11/24/2019.

[21] Robert Wilson, Peter Walsh, Jonathan Tripp, Ibrahim Suleiman, Richard Parker, Simon Norton, Simon Nickerson, Steve Linton, John Bray, and Rachel Abbott. ATLAS of Finite Group Representations - Version 3, `http://brauer.maths.qmul.ac.uk/Atlas/v3/`, accessed 11/24/2019.