Technische Universität Darmstadt

Fachbereich Mathematik
Schlossgartenstraße 7
64287 Darmstadt

SS 2005

Diplomarbeit

Amalgams of unitary groups in Sp(2n,q)

Max Horn

October 24, 2005

Supervised by Dr. habil. Ralf Gramlich

Contents

1. Introduction

In 1977 Kok-Wee Phan [20] published a theorem on the generation of the special unitary group $\mathrm{SU}(n+1, q^2)$ by a system of subgroups isomorphic to $\mathrm{SU}(3, q^2)$. Phan's theorem is used as an identification tool in the classification of finite simple groups (see [1]).

The revision of this classification by Gorenstein, Lyons and Solomon called for a revision of Phan's results as well. This motivated Bennett and Shpectorov [4] to provide a new proof of Phan's theorem. Their approach is based on the realization that Phan's configuration arises as the amalgam[1] of *stripped*[2] rank two parabolics in the flag-transitive action of $\mathrm{SU}(n+1, q^2)$ on the geometry of nondegenerate subspaces of the underlying unitary space. To prove the theorem, one essentially needs to classify related amalgams, and then has to study the universal completion of these amalgams.

The approach used to deal with the universal completion shows the beauty of applying geometry to this problem: Thanks to Tits' Lemma the result can be established by proving that the geometries involved are simply connected, which amounts to analyzing their simplicial complexes. Hence the original group theoretic problem has been transformed into a geometric problem.

The work described above deals with the group $\mathrm{SU}(n+1, q^2)$. However, the techniques employed in the new proof of Phan's theorem can be extended to other classical groups. In particular, a Phan-type theorem for the group $\mathrm{Sp}(2n, q)$ was presented and proved in [13] and [15].

That theorem requires $n \geq 5$ and q arbitrary, or $n = 4$ and $q \geq 3$, or $n = 3$ and $q \geq 8$. For $(n, q) = (3, 2)$, there exists a counterexample, which shows that the theorem is false in that case. So far it was not clear whether the other exceptions in the above list were true exceptions, or simply were due to shortcomings of the proof. Specifically, the proof relies on point counting arguments, and the open cases $(n, q) \in \{(3, 3), (3, 4), (3, 5), (3, 7), (4, 2)\}$ were "too small" to be tractable by those techniques. In this thesis, we show that the theorem extends to these parameters. The improved theorem can be found in Section 3.1.

Structure

The structure of this work is as follows: In Chapter 2, we provide some of the standard definitions and terminology used throughout the thesis. The reader already familiar with

[1] The reader not familiar with the notations and terminology employed in this introduction may want to read Chapter 2 first.

[2] *stripped* in the sense that the torus of $\mathrm{SU}(n+1, q^2)$ has been factored out. See Section 6.4.

them can skip over that chapter, maybe with the exception of Section 2.3, in which we describe the specific geometrical setting.

In Chapter 3 we present the main result proved in this thesis.

In Chapter 4 we analyze the universal completion of the amalgam of maximal parabolics, for all open cases, and prove part of our main result.

In Chapter 5, we refine the geometric approach from [15] such that it also covers the case $(n, q) = (3, 7)$ (yielding another proof of this case), and also correct some small errors in the original paper. We also complete the proof of our main result.

In Chapter 6 we use the results from the preceding chapters to give a generalized version of the main theorems from [15].

In Appendix A we briefly review the case $(n, q) = (3, 3)$ and compare it to some previous results for related Phan-type theorems.

In Appendix B we give finite presentations for the groups from Chapter 4, which can be used to verify the results stated there.

In Appendix C you will find the GAP [10] code used to arrive at the results in Chapter 4; in particular, it was used to find the presentations from Appendix B (the reason we print these presentations, too, is that they are semi-randomly generated, and so they differ on each run of the code).

Finally in Appendix D we list GAP code that is referred to in the proof of Lemma 5.3.3.

Acknowledgment

I would like to thank the following people:

- Ralf Gramlich, for his advice and help, for patiently answering all my questions, and for too many other things to mention here.
- Werner Nickel, for helping me with all GAP related things, providing lots of helpful feedback on this thesis, and everything else he has done for me.
- George Havas, for performing the coset enumeration on Sp(6, 7), which filled the last gap in this work.
- Max Neunhöffer, for generously spending his valuable time and computer resources on my problems.
- Henning Sudbrock and Tobias Hartnick for proofreading drafts of this thesis.
- My friends and family, who have supported me all the time.

2. Basics and Definitions

2.1. Amalgams

In this section, we introduce the notion of group amalgams. Note that we only need a special kind of amalgams; for a more general definition, see for example [24].

Definition 2.1.1. An **amalgam of groups** is a set $\mathcal{A}$ endowed with a partial multiplication and a finite family of subsets $(G_i)_{i\in I}$ such that the following holds:

(1) $\mathcal{A} = \cup_{i\in I} G_i$,

(2) the restriction of the multiplication to G_i turns G_i into a group for $i \in I$,

(3) $G_i \cap G_j$ is a subgroup both in G_i and G_j for all $i, j \in I$.

Example 2.1.2. Let G be an arbitrary group. Let $(G_i)_{i\in I}$ be a finite family of subgroups of G. Then $\mathcal{A} := \cup_{i\in I} G_i$ defines an amalgam of groups.

Definition 2.1.3. A group G is called a **completion** of an amalgam $\mathcal{A}$ if there exists a map $\pi : \mathcal{A} \to G$ (called the **completion map**) such that

(1) for all $i \in I$ the restriction of π to G_i is a homomorphism

(2) $\pi(\mathcal{A})$ generates G.

Example 2.1.4. Let G and $\mathcal{A}$ be as in Example 2.1.2. Then G together with the natural map $\pi : \mathcal{A} \to G$ (where $\pi = \mathrm{id}_{G|\mathcal{A}}$) is a completion of $\mathcal{A}$ if and only if G is generated by $\mathcal{A}$, i.e. $G = \langle \mathcal{A} \rangle$.

Example 2.1.5. The trivial group is always a completion of any amalgam. An amalgam which only permits the trivial group as a completion is called **collapsing**.

Among all completions of $\mathcal{A}$ there is a largest one which, if $\mathcal{A}$ is finite, can be defined as the group having the following finite presentation:

$$\mathcal{U}(\mathcal{A}) = \langle t_h \mid h \in \mathcal{A}, t_x t_y = t_{xy} \text{ if } xy \text{ is defined } \rangle.$$

$\mathcal{U}(\mathcal{A})$ is called the **universal completion**. Its completion map is given by

$$\psi : \mathcal{A} \to \mathcal{U}(\mathcal{A}) : g \mapsto t_g.$$

We call this completion universal since it has the universal property that for any other completion G with completion map π, there exists a (unique) group epimorphism $\widehat{\pi}$ from $\mathcal{U}(\mathcal{A})$ onto G such that the following diagram commutes:

$$\begin{array}{ccc} \mathcal{A} & \xrightarrow{\psi} & \mathcal{U}(\mathcal{A}) \\ & \searrow^{\pi} & \downarrow{\scriptstyle \widehat{\pi}} \\ & & G \end{array}$$

We define the map $\widehat{\pi}$ by first defining it on the the generators $\psi(\mathcal{A})$ of $\mathcal{U}(\mathcal{A})$ only, via $\widehat{\pi}_{|\psi(\mathcal{A})} : t_x \mapsto x$. This can be extended to a group epimorphism because

$$\widehat{\pi}(t_x t_y) = \widehat{\pi}(t_{xy}) = xy = \widehat{\pi}(t_x)\widehat{\pi}(t_y)$$

if xy (and thus t_{xy}) is defined, and otherwise define

$$\widehat{\pi}(t_x t_y) := xy = \widehat{\pi}(t_x)\widehat{\pi}(t_y).$$

We now consider the amalgam formed by subgroups of a given group G (see Example 2.1.2).

Lemma 2.1.6. *Let $(G_i)_{i\in I}$ be a finite family of subgroups of a finite group G which generates G, let $\mathcal{A} := \cup_{i\in I} G_i$ be the associated amalgam of groups, and let $\psi : \mathcal{A} \to \mathcal{U}(\mathcal{A})$ be the completion map. Then for each $i \in I$ the restriction $\psi_{|G_i} : G_i \to \mathcal{U}(\mathcal{A})$ is injective. Furthermore $\mathcal{U}(\mathcal{A}) \cong G$ if and only if any (and then all) of the G_i has the same index in G as in $\mathcal{U}(\mathcal{A})$.*

Proof. Note that G is a completion of $\mathcal{A}$, for which the completion map ι is the inclusion map. By the universal nature of $\mathcal{U}(\mathcal{A})$, there exists an epimorphism π from $\mathcal{U}(\mathcal{A})$ onto G such that the following diagram commutes:

$$\begin{array}{ccc} \mathcal{A} & \xrightarrow{\psi} & \mathcal{U}(\mathcal{A}) \\ & \searrow^{\iota} & \downarrow{\scriptstyle \pi} \\ & & G \end{array}$$

Hence $\psi_{|G_i}$ must be injective.

Let $\widetilde{G}_i := \psi(G_i)$, the subgroup of $\mathcal{U}(\mathcal{A})$ corresponding uniquely to G_i via ψ. It follow that if $\mathcal{U}(\mathcal{A}) \cong G$, the index of the G_i and $\widetilde{G}_i$ coincide, hence this direction of the claim follows immediately.

On the other hand, assume that for some i, G_i has the same index in G as $\widetilde{G}_i$ has in $\mathcal{U}(\mathcal{A})$. $\widetilde{G}_i$ intersects the kernel of π trivially, since by the above $\pi(\widetilde{G}_i) = G_i \cong \widetilde{G}_i$. But by hypothesis $[\mathcal{U}(\mathcal{A}) : \widetilde{G}_1] = [G : G_1]$, hence π is an isomorphism between $\mathcal{U}(\mathcal{A})$ and G. □

2.2. Geometries

In the following, we give a quick run-down on the basics of synthetic geometry. For a more complete introduction to the subject, refer for example to [6] or [19].

Definition 2.2.1 (Pregeometry)**.** A **pregeometry over a set** I is a triple $\mathcal{G} = (X, *, typ)$ where X is a set (its elements are called the **elements of** $\mathcal{G}$), $*$ is a symmetric and reflexive relation defined on X which is called the **incidence relation of** $\mathcal{G}$, and typ is a map from X to I (the set I is called the **type set of** $\mathcal{G}$) such that $typ(x) = typ(y)$ and $x * y$ imply $x = y$. The pregeometry $\mathcal{G}$ is called **connected** if the graph $(X, *)$ is connected.

Definition 2.2.2 (Rank, flag)**.** If $A \subseteq X$, then A is of the **type** $typ(A)$, of **rank** $|typ(A)|$, and of **corank** $|I \setminus typ(A)|$. The cardinality $|I|$ of I is called the **rank of** $\mathcal{G}$. A **flag** of $\mathcal{G}$ is a set of mutually incident elements of $\mathcal{G}$. Flags of type I are called **chambers**.

Definition 2.2.3 (Residue)**.** If F is a flag of $\mathcal{G}$, then the **residue** of F in $\mathcal{G}$ is the pregeometry $\mathcal{G}_F(X_F, *_F, typ_F)$, where X_F is the set of elements of X that are incident with but distinct from all elements of F, and $*_F, typ_F$ are the restrictions of $*$ and typ to $X_F \times X_F$ respectively X_F. The pregeometry $\mathcal{G}$ is called **residually connected** if $(X_F, *_F)$ is a connected graph for each flag F of $\mathcal{G}$ of corank greater or equal two, and non-empty for each flag F of corank one.

Definition 2.2.4 (Geometry)**.** A **geometry over** I is a pregeometry $\mathcal{G}$ over I in which every maximal flag is a chamber.

Example 2.2.5. Let V be a vector space over $\mathbb{R}$ of finite dimension n, $n \geq 3$. Denote by $\mathbb{P}(V)$ the geometry over $I := \{1, \dots, n-1\}$ consisting of the proper subspaces of V with symmetrized containment as incidence and the dimension function as the type function. The geometry $\mathbb{P}(V)$ is called the **(desarguesian) projective geometry of** V.

Definition 2.2.6. Let G be a group of automorphisms of a geometry $\mathcal{G}$ over I. We say G acts **flag-transitively** on $\mathcal{G}$ if for each $J \subseteq I$, G acts transitively on the set of flags of type J. In other words, if F_1 and F_2 are flags in $\mathcal{G}$ of equal type, then there exists $g \in G$ such that $g(F_1) = F_2$.

Definition 2.2.7. Let $\mathcal{G}$ be a geometry of rank n, let $\phi : G \to \operatorname{Aut}\mathcal{G}$ be a group homomorphism such that $\phi(G)$ acts incidence-transitively on $\mathcal{G}$. A **rank** k **parabolic** is the stabilizer of a flag of corank k from $\mathcal{G}$ with respect to the action given by $gF := \phi(g)F$. Parabolics of rank $n-1$ are called **maximal parabolics**. They are exactly the stabilizers in G of single elements of $\mathcal{G}$.

Definition 2.2.8. Let $\mathcal{G}$ be a geometry which admits points and lines as two if its types. The **collinearity graph** is an undirected graph which has as its vertices the points of $\mathcal{G}$, and in which two vertices v_1, v_2 corresponding to points p_1, p_2 are connected by an edge if and only if there exists a line l incident to both p_1 and p_2.

2.3. Geometrical setting

Let B_{2n} be the matrix

$$\left(\begin{array}{c|c} 0 & \mathrm{Id}_{n\times n} \\ \hline -\mathrm{Id}_{n\times n} & 0 \end{array}\right)$$

over $\mathbb{F}_{q^2}$. Let $(\cdot,\cdot)$ be the bilinear form defined by B_{2n} via $(x,y) := x^T B_{2n} y$. We represent $G := \mathrm{Sp}(2n,q^2)$ by the set of all invertible $(2n)\times(2n)$-matrices A over $\mathbb{F}_{q^2}$ which preserve $(\cdot,\cdot)$, that is, $A^T B_{2n} A = B_{2n}$ holds.

Let V be the vector space $\mathbb{F}_{q^2}^{2n}$ and let $\{e_1,\dots,e_n,f_1,\dots,f_n\}$ be the standard basis. We denote by $^-$ the (unique, by Prop. V.12.4 in [5]) non-trivial involutory field automorphism $x \mapsto x^q$ of $\mathbb{F}_{q^2}$. Consider the $^-$-semi-linear map $\sigma : V \to V$ defined by $e_i \mapsto f_i, f_i \mapsto -e_i$ and $\sigma(c\cdot v) = \bar{c}\sigma(v)$ for $c \in \mathbb{F}_{q^2}, v \in V$. Note that $\sigma(v) = \overline{B_{2n}v} = B_{2n}\overline{v}$. Then the centralizer $G_\sigma := \{g \in G \mid \forall v \in V : g\sigma(v) = \sigma(gv)\}$ of σ in $\mathrm{Sp}(2n,q^2)$ is isomorphic to $\mathrm{Sp}(2n,q)$ (see [15], Proposition 3.8). For our computations in the later sections, we take G_σ as our representation of $\mathrm{Sp}(2n,q)$. Note that for a matrix $A \in \mathrm{Sp}(2n,q^2)$, centralizing σ is equivalent to the condition $A^{-1} = \overline{A}^T$.

We now define the (so-called flip-flop) geometry $\mathcal{G}_C^{\mathrm{herm}}$ which we are studying in this thesis[1]. To this end, we define a $^-$-hermitian form $((\cdot,\cdot))$ by $((u,v)) := (u,\sigma(v))$. To denote orthogonality with respect to the form $(\cdot,\cdot)$, we use the symbol $\perp$. To denote orthogonality with respect to the form $((\cdot,\cdot))$, we use the symbol $\perp\!\!\!\perp$.

The objects of the geometry are all non-trivial subspaces of V which are totally isotropic with respect to $(\cdot,\cdot)$ and $((\cdot,\cdot))$-nondegenerate. The incidence relation is defined by symmetrized containment. As $\mathrm{Sp}(2n,q)$ respects both forms, $\mathrm{Sp}(2n,q)$ acts on the geometry. This action is in fact flag-transitive (see [15], Proposition 4.2). Note that the name $\mathcal{G}_C^{\mathrm{herm}}$ is inspired by the fact that the geometry is defined via a hermitian form and corresponds to the Dynkin diagram C_n (see for example [6]).

For our computations we choose the maximal flag F

$$\langle e_1\rangle \subset \langle e_1,e_2\rangle \subset \ldots \subset \langle e_1,e_2,\dots,e_n\rangle.$$

When computing stabilizers, we will refer to the stabilizers of each of these subspaces as the point stabilizer M_1, the line stabilizer M_2, the plane stabilizer M_3 and (for $n=4$) the space stabilizer M_4, respectively. The M_i are the maximal parabolics of G_σ.

[1]For an introduction to flip-flop geometries, see [3] or [12].

3. Results

3.1. Main result

By $A_{(k)}$ we denote the amalgam of rank k parabolics. Using the results from [15] and the work done here we can prove the following theorem:

Theorem 3.1.1. *G_σ is the universal completion of $\mathcal{A}_{(n-1)}$ if and only if $n \geq 3$ and $(n, q) \neq (3, 2)$.*

Proof. See Section 5.6 □

Then using the above theorem, we can refine Theorems 1 and 2 from [15] (proof identical to the one given there):

Theorem 3.1.2. *The following hold.*

(1) If $n \geq 3$ and $q \geq 3$ then G_σ is the universal completion of $\mathcal{A}_{(2)}$.

(2) If $n \geq 4$ then G_σ is the universal completion of $\mathcal{A}_{(3)}$.

3.2. Consequences

The result from Section 3.1 can be used to strengthen the Phan-type theorems for C_n as given in [11], [12], [13] and [15]. This was in fact the main motivation for the work accomplished in this thesis.

We will give some required definitions, the resulting theorems as well as a sketch of their proof, in Chapter 6.

4. Determining universal completions

4.1. General approach for computing the amalgams: GAP

In order to compute the universal completion of the amalgams of parabolics which we are studying here, we do the following: First, we determine generators for each parabolic. They will be chosen such that the intersection of the parabolics is generated by the intersection of their respective generating set. Specifically, in the case $n = 3$ the maximal parabolics we consider are the point, line and plane stabilizers of our flag F, with suitably chosen generators u, v, w. These stabilizers all intersect in the flag stabilizer, and so generators of the flag stabilizer together with u, v, w generate the desired parabolics as well as their intersections (which are also parabolics). In Figure 4.1 you can see the subgroup structure they form.

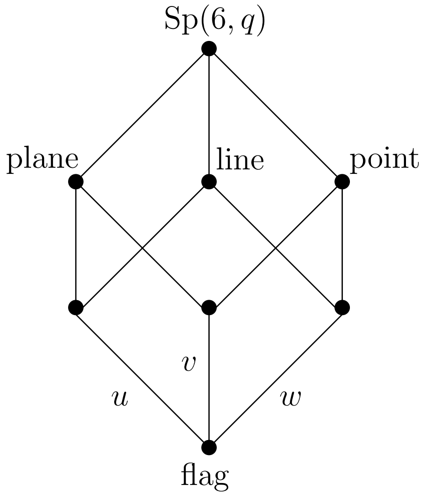


Figure 4.1.: Subgroup structure in $\mathrm{Sp}(6, q)$

To prove that the parabolics and their intersections are generated by the matrices for which we claim this, we first show that they generate a subgroup U of the desired group H; then we compute a lower bound of the size for U. If this bound equals the size of the full group H, we have thus established that $H \cong U$.

We proceed by using GAP [10] to compute finite presentations of the parabolics in terms of these generators: We first find a permutation group isomorphic to our group, then from that determine the corresponding relators (to learn more about the algorithms involved, which GAP implements, refer to [7] and [17]). Due to our choice of generators, the universal completion of the amalgam is obtained by forming the union of all the generators and relators of the parabolics.

Finally, we have to prove that this universal completion is identical to $\mathrm{Sp}(2n, q)$. For this it would be sufficient to compute the size of the group. Doing that directly via a coset enumeration over the trivial group is not feasible due to the size of this finitely presented group. Instead we compute the index of one of the parabolics inside the amalgam, which also establishes the desired isomorphism (see Lemma 2.1.6).

Before we proceed with the details of this, we present some auxiliary results which are useful for computing lower bounds on the group sizes.

4.2. Vectors with short orbit

In this section we will discuss certain vectors which have a comparatively short orbit under the group action induced by $\mathrm{Sp}(2n, q)$. These are used in constructing permutation representations of small degree, which in turn are used in computing lower bounds for the size of certain subgroups of $\mathrm{Sp}(2n, q)$.

Let V and G_σ be as defined in Section 2.3.

Definition 4.2.1. For $\lambda \in \mathbb{F}_{q^2}$, we define $V_\lambda := \{u \in V \mid \sigma(u) = \lambda u\}$.

Lemma 4.2.2. *The following hold.*

(1) V_λ is G_σ-invariant.

(2) V_λ is an $\mathbb{F}_q$-subspace of V.

(3) $V_\lambda \neq 0$ if and only if $\lambda\bar{\lambda} = -1$

(4) If $V_\lambda \neq 0$ then V_λ contains a basis of V.

Proof. Take $g \in G_\sigma$. If $u \in V_\lambda$ then

$$\sigma(g(u)) = g(\sigma(u)) = g(\lambda u) = \lambda g(u).$$

This proves statement (1). Suppose $u \in V_\lambda$. Then for $\mu \in \mathbb{F}_q$

$$\sigma(\mu u) = \bar{\mu}\sigma(u) = \mu\sigma(u) = \mu\lambda u = \lambda(\mu u).$$

This proves (2). Also, $-u = \sigma(\sigma(u)) = \bar{\lambda}\lambda u$. Thus, if $u \neq 0$ then $\lambda\bar{\lambda} = -1$. This proves the 'only if' part of (3). To prove the 'if' part, choose a basis

$$e_1, \ldots, f_n$$

for σ. Fix a $\lambda \in \mathbb{F}_{q^2}$ such that $\lambda\bar{\lambda} = -1$. Define

$$u_i := e_i - \bar{\lambda} f_i \quad \text{and} \quad v_i = \bar{\lambda} e_i + f_i$$

for $1 \leq i \leq n$.

Note that u_i and v_i are in V_λ since

$$\sigma(u_i) = \lambda e_i + f_i = \lambda(e_i - \bar{\lambda} f_i) = \lambda u_i \quad \text{and} \quad \sigma(v_i) = -e_i + \lambda f_i = \lambda(\bar{\lambda} e_i + f_i) = \lambda v_i.$$

This shows that $V_\lambda \neq 0$.

Furthermore, u_i and v_i are only proportional if $\bar{\lambda} = \lambda$, that is, $\lambda \in \mathbb{F}_q$. Thus, if $\lambda \notin \mathbb{F}_q$ then

$$\{u_1, \ldots, u_n, v_1, \ldots, v_n\}$$

is a basis of V. If $\lambda \in \mathbb{F}_q$ then consider $\lambda' = \frac{\bar{\mu}}{\mu}\lambda$, where μ is chosen so that $\frac{\bar{\mu}}{\mu} \notin \mathbb{F}_q$. By (2), $V_{\lambda'} = \mu V_\lambda$. Also, since $\lambda' \notin \mathbb{F}_q$, we have that $V_{\lambda'}$ contains a basis of V, and hence so does V_λ. □

Let $\lambda \in \mathbb{F}_{q^2}$ such that $\lambda\overline{\lambda} = -1$. Since V_λ contains a basis for V, it has the same dimension as V. Since it is a $\mathbb{F}_q$-subspace, we deduce that $|V_\lambda| = q^{2n}$. Let $v_n := \overline{\lambda} e_n + f_n \in V_\lambda$. We observe that $G_\sigma v_n \subset V_\lambda$, i.e. the orbit of v_n, is a subset of V_λ, and hence $\left|v_n^{G_\sigma}\right| < q^{2n}$.

Thus we have found a vector with an orbit that is short enough to be suitable for our purposes. For we can now use this to effectively compute lower bounds on the size of G_σ and its subgroups: All these groups induce a permutation action on the orbit $G_\sigma v_n$. Hence we can compute an homomorphic image into a permutation group. There are good algorithms (and implementations of them) for determining the size of such a permutation group.[1] Thus, we can efficiently compute a lower bound on the size of a factor group of any subgroup H of G_σ. If the action induced by the group on the orbit is faithful, then we actually obtain the exact size of the group. This is the case, but we do not show it here, as it is not necessary for our needs.

4.3. Subgroups and their sizes

The maximal parabolics of M_i, with respect to our maximal flag F are subgroups of $G_\sigma \cong \mathrm{Sp}(2n, q)$ with the following isomorphism type (see [15]):

$$M_i \cong \begin{cases} \mathrm{Sp}(2n-2i, q) \times \mathrm{GU}(i, q^2) & \text{for } 1 \leq i \leq n-1 \\ \mathrm{GU}(n, q^2) & \text{for } i = n \end{cases}.$$

Note that by $\mathrm{GU}(n, q^2)$ we denote the general unitary group of dimension n over the field $\mathbb{F}_{q^2}$ (sometimes in the literature this is referred to as $\mathrm{GU}(n, q)$, which is also the notation used by GAP).

[1] They work better the smaller the set is upon which the group acts, which is why we went to some effort to find vectors with relatively small orbit.

So we can compute the size of the M_i, since it is well known (see e.g. [25]) that

$$|\mathrm{Sp}(2n,q)| = q^{n^2}\prod_{i=1}^{n}(q^{2i}-1) \qquad \text{and} \qquad \left|\mathrm{GU}(n,q^2)\right| = q^{n(n-1)/2}\prod_{i=1}^{n}(q^i+(-1)^{i+1}).$$

These size formulas are important in the following sections, where we use them to prove that the groups generated by certain matrices are precisely the groups we are looking for.

4.4. The case $n = 3, q = 3$

In this section z denotes a primitive element in $\mathbb{F}_9$ over $\mathbb{F}_3$ with minimal polynomial $x^2 - x - 1$. We define the following matrices:

$$U := \left(\begin{array}{ccc|ccc} z^7 & z^1 & & & & \\ z^7 & z^5 & & & & \\ & & 1 & & & \\ \hline & & & z^5 & z^3 & \\ & & & z^5 & z^7 & \\ & & & & & 1 \end{array}\right) \qquad V := \left(\begin{array}{ccc|ccc} 1 & & & & & \\ & z^7 & z^1 & & & \\ & z^7 & z^5 & & & \\ \hline & & & 1 & & \\ & & & & z^5 & z^3 \\ & & & & z^5 & z^7 \end{array}\right)$$

$$W := \left(\begin{array}{ccc|ccc} 1 & & & & & \\ & 1 & & & & \\ & & z^7 & & & z^1 \\ \hline & & & 1 & & \\ & & & & 1 & \\ & & z^7 & & & z^5 \end{array}\right)$$

In addition to these elements we use diagonal matrices $D_i, 1 \leq i \leq 3$, that generate the stabilizer of the flag F, a half-split torus isomorphic to C_4^3.

Lemma 4.4.1. *Each maximal parabolic in* $\mathrm{Sp}(6,3)$ *is generated by the matrices specified in the following table together with generators of the flag stabilizer.*

stabilizer	*element*	*generators*	*isomorphism type*	*index*
M_1	$\langle e_1\rangle$	V, W	$\mathrm{Sp}(4,3)\times\mathrm{GU}(1,9)$	44226
M_2	$\langle e_1, e_2\rangle$	U, W	$\mathrm{Sp}(2,3)\times\mathrm{GU}(2,9)$	3980340
M_3	$\langle e_1, e_2, e_3\rangle$	U, V	$\mathrm{GU}(3,9)$	379080

Furthermore, the pairwise intersection of the stabilizers is generated by the intersection of their generators as given above.

Proof. The claimed generators of each M_i obviously each stabilize the corresponding element in the table. Hence they generate subgroups of the stabilizers. Also, the intersection of the generators of any two M_i forms a subgroup of the intersection of the two groups. To complete the proof, we compute, using GAP and the results from 4.2, lower bounds of the group sizes. We then verify that they are equal to the sizes of the maximal parabolics respectively the double stabilizers. □

Based on the above table, we find presentations of the maximal parabolics on the generators d_1, d_2, d_3, u, v, w. These, together with the union of the relators of said presentations give a presentation for the universal completion of the amalgam of the maximal parabolics.

Coset enumeration over the subgroup generated by d_1, d_2, d_3, v, w gives an index of 44226 which corresponds to the index of M_1 in $\mathrm{Sp}(6,3)$. By Lemma 2.1.6 this shows that $\mathrm{Sp}(6,3)$ is the universal completion of the amalgam of maximal parabolics.

4.5. The case $n = 3, q = 4$

In this section z denotes a primitive element in $\mathbb{F}_{16}$ over $\mathbb{F}_2$ with minimal polynomial $x^4 + x + 1$. We define the following matrices:

$$U := \left(\begin{array}{ccc|ccc} z^5 & z^1 & & & & \\ z^4 & z^5 & & & & \\ & & 1 & & & \\ \hline & & & z^5 & z^4 & \\ & & & z^1 & z^5 & \\ & & & & & 1 \end{array}\right) \qquad V := \left(\begin{array}{ccc|ccc} 1 & & & & & \\ & z^5 & z^1 & & & \\ & z^4 & z^5 & & & \\ \hline & & & 1 & & \\ & & & & z^5 & z^4 \\ & & & & z^1 & z^5 \end{array}\right)$$

$$W := \left(\begin{array}{ccc|ccc} 1 & & & & & \\ & 1 & & & & \\ & & z^5 & & & z^1 \\ \hline & & & 1 & & \\ & & & & 1 & \\ & & z^4 & & & z^5 \end{array}\right)$$

In addition to these elements we use diagonal matrices $D_i, 1 \leq i \leq 3$, that generate the stabilizer of the flag F, a half-split torus isomorphic to C_5^3.

Lemma 4.5.1. *Each maximal parabolic in* $\mathrm{Sp}(6,4)$ *is generated by the matrices specified in the following table together with generators of the flag stabilizer.*

stabilizer	*element*	*generators*	*isomorphism type*	*index*
M_1	$\langle e_1 \rangle$	V, W	$\mathrm{Sp}(4,4) \times \mathrm{GU}(1,16)$	838656
M_2	$\langle e_1, e_2 \rangle$	U, W	$\mathrm{Sp}(2,4) \times \mathrm{GU}(2,16)$	228114432
M_3	$\langle e_1, e_2, e_3 \rangle$	U, V	$\mathrm{GU}(3,16)$	13160448

Proof. See the proof of Lemma 4.4.1. □

Based on the above table, we find presentations of the maximal parabolics on the generators d_1, d_2, d_3, u, v, w. These, together with the union of the relators of said presentations give a presentation for the universal completion of the amalgam of the maximal parabolics.

Coset enumeration over the subgroup generated by d_1, d_2, d_3, v, w gives an index of 838656 which corresponds to the index of M_1 in $\mathrm{Sp}(6,4)$. By Lemma 2.1.6 this shows that $\mathrm{Sp}(6,4)$ is the universal completion of the amalgam of maximal parabolics.

4.6. The case $n = 3, q = 5$

In this section z denotes a primitive element in $\mathbb{F}_{25}$ over $\mathbb{F}_5$ with minimal polynomial $x^2 - x + 2$. We define the following matrices:

$$U := \left(\begin{array}{ccc|ccc} z^{18} & z^{1} & & & & \\ z^{17} & z^{18} & & & & \\ & & 1 & & & \\ \hline & & & z^{18} & z^{5} & \\ & & & z^{13} & z^{18} & \\ & & & & & 1 \end{array}\right) \qquad V := \left(\begin{array}{ccc|ccc} 1 & & & & & \\ & z^{18} & z^{1} & & & \\ & z^{17} & z^{18} & & & \\ \hline & & & 1 & & \\ & & & & z^{18} & z^{5} \\ & & & & z^{13} & z^{18} \end{array}\right)$$

$$W := \left(\begin{array}{ccc|ccc} 1 & & & & & \\ & 1 & & & & \\ & & z^{18} & & & z^{1} \\ \hline & & & 1 & & \\ & & & & 1 & \\ & & z^{17} & & & z^{18} \end{array}\right)$$

In addition to these elements we use diagonal matrices $D_i, 1 \leq i \leq 3$, that generate the stabilizer of the flag F, a half-split torus isomorphic to C_6^3.

Lemma 4.6.1. *Each maximal parabolic in* $\mathrm{Sp}(6,5)$ *is generated by the matrices specified in the following table together with generators of the flag stabilizer.*

stabilizer	*element*	*generators*	*isomorphism type*	*index*
M_1	$\langle e_1 \rangle$	V, W	$\mathrm{Sp}(4,5) \times \mathrm{GU}(1,25)$	8137500
M_2	$\langle e_1, e_2 \rangle$	U, W	$\mathrm{Sp}(2,5) \times \mathrm{GU}(2,25)$	5289375000
M_3	$\langle e_1, e_2, e_3 \rangle$	U, V	$\mathrm{GU}(3,25)$	201500000

Proof. See the proof of Lemma 4.4.1. □

Based on the above table, we find presentations of the maximal parabolics on the generators d_1, d_2, d_3, u, v, w. These, together with the union of the relators of said presentations give a presentation for the universal completion of the amalgam of the maximal parabolics.

Coset enumeration over the subgroup generated by d_1, d_2, d_3, v, w gives an index of 8137500 which corresponds to the index of M_1 in $\mathrm{Sp}(6,5)$. By Lemma 2.1.6 this shows that $\mathrm{Sp}(6,5)$ is the universal completion of the amalgam of maximal parabolics.

4.7. The case $n = 3, q = 7$

This is the biggest of the open cases, with an index of 247163742. Using our standard representation of the involved groups, that amounts to a memory requirement of about 12 GB when using ACE [9] to perform the coset enumeration. This means that one has to use a 64bit machine with sufficient memory in order to perform the enumeration.

George Havas, one of the authors of ACE, performed these computations for us on both a Sparc and an Itanium system with sufficient memory.

In this section z denotes a primitive element in $\mathbb{F}_{49}$ over $\mathbb{F}_7$ with minimal polynomial $x^2 - x + 3$. We define the following matrices:

$$U := \left(\begin{array}{ccc|ccc} z^{11} & z^{1} & & & & \\ z^{31} & z^{29} & & & & \\ & & 1 & & & \\ \hline & & & z^{29} & z^{7} & \\ & & & z^{25} & z^{11} & \\ & & & & & 1 \end{array}\right) \qquad V := \left(\begin{array}{ccc|ccc} 1 & & & & & \\ & z^{11} & z^{1} & & & \\ & z^{31} & z^{29} & & & \\ \hline & & & 1 & & \\ & & & & z^{29} & z^{7} \\ & & & & z^{25} & z^{11} \end{array}\right)$$

$$W := \left(\begin{array}{ccc|ccc} 1 & & & & & \\ & 1 & & & & \\ t & & z^{11} & & & z^{1} \\ \hline & & & 1 & & \\ & & & & 1 & \\ & & z^{31} & & & z^{29} \end{array}\right)$$

In addition to these elements we use diagonal matrices $D_i, 1 \le i \le 3$, that generate the stabilizer of the flag F, a half-split torus isomorphic to C_8^3.

Lemma 4.7.1. *Each maximal parabolic in* $\mathrm{Sp}(6,7)$ *is generated by the matrices specified in the following table together with generators of the flag stabilizer.*

stabilizer	*element*	*generators*	*isomorphism type*	*index*
M_1	$\langle e_1 \rangle$	V, W	$\mathrm{Sp}(4,7) \times \mathrm{GU}(1,49)$	247163742
M_2	$\langle e_1, e_2 \rangle$	U, W	$\mathrm{Sp}(2,7) \times \mathrm{GU}(2,49)$	605551167900
M_3	$\langle e_1, e_2, e_3 \rangle$	U, V	$\mathrm{GU}(3,49)$	12070787400

Proof. See the proof of Lemma 4.4.1. □

Based on the above table, we find presentations of the maximal parabolics on the generators d_1, d_2, d_3, u, v, w. These, together with the union of the relators of said presentations give a presentation for the universal completion of the amalgam of the maximal parabolics.

Coset enumeration over the subgroup generated by d_1, d_2, d_3, v, w gives an index of 247163742 which corresponds to the index of M_1 in $\mathrm{Sp}(6,7)$. By Lemma 2.1.6 this shows that $\mathrm{Sp}(6,7)$ is the universal completion of the amalgam of maximal parabolics.

4.8. The case $n = 4, q = 2$

In this section z denotes a primitive element in $\mathbb{F}_4$ over $\mathbb{F}_2$ with minimal polynomial $x^2 + x + 1$. We define the following matrices:

$$P_1 := \left(\begin{array}{cccc|cccc} 1 & & & & & & & \\ & 1 & & & & & & \\ & & 1 & & & & z & z^2 \\ & & & 1 & & & z^2 & 1 \\ \hline & & & & 1 & & & \\ & & & & & 1 & & \\ & & z^2 & z & & & 1 & \\ & & z & 1 & & & & 1 \end{array}\right) \qquad P_2 := \left(\begin{array}{cccc|cccc} 1 & & & & & & & \\ & & 1 & & & & & \\ & 1 & & & & & & \\ & & & 1 & & & & \\ \hline & & & & 1 & & & \\ & & & & & & 1 & \\ & & & & & 1 & & \\ & & & & & & & 1 \end{array}\right)$$

$$P_3 := \left(\begin{array}{cccc|cccc} 1 & & & & & & & \\ & z^2 & 1 & 1 & & & & \\ & 1 & z^2 & 1 & & & & \\ & 1 & 1 & z^2 & & & & \\ \hline & & & & 1 & & & \\ & & & & & z & 1 & 1 \\ & & & & & 1 & z & 1 \\ & & & & & 1 & 1 & z \end{array}\right) \qquad P_4 := \left(\begin{array}{cccc|cccc} & 1 & & & & & & \\ 1 & & & & & & & \\ & & 1 & & & & & \\ & & & 1 & & & & \\ \hline & & & & & 1 & & \\ & & & & 1 & & & \\ & & & & & & 1 & \\ & & & & & & & 1 \end{array}\right)$$

$$P_5 := \left(\begin{array}{cccc|cccc} z & 1 & 1 & & & & & \\ 1 & z & 1 & & & & & \\ 1 & 1 & z & & & & & \\ & & & 1 & & & & \\ \hline & & & & z^2 & 1 & 1 & \\ & & & & 1 & z^2 & 1 & \\ & & & & 1 & 1 & z^2 & \\ & & & & & & & 1 \end{array}\right) \qquad P_6 := \left(\begin{array}{cccc|cccc} 1 & & & & & & & \\ & 1 & & & & & & \\ & & & 1 & & & & \\ & & 1 & & & & & \\ \hline & & & & 1 & & & \\ & & & & & 1 & & \\ & & & & & & & 1 \\ & & & & & & 1 & \end{array}\right)$$

$$P_7 := \left(\begin{array}{cccc|cccc} 1 & & & & & & & \\ & 1 & & & & & & \\ & & 1 & & & & & \\ & & & & & & & 1 \\ \hline & & & & 1 & & & \\ & & & & & 1 & & \\ & & & & & & 1 & \\ & & & 1 & & & & \end{array}\right)$$

In addition to these elements we use diagonal matrices $D_i, 1 \leq i \leq 4$, that generate the stabilizer of the flag F, a half-split torus isomorphic to C_3^4.

Lemma 4.8.1. *Each maximal parabolic in* $\mathrm{Sp}(8,2)$ *is generated by the matrices specified in the following table together with generators of the flag stabilizer.*

stabilizer	*element*	*generators*	*isomorphism type*	*index*
M_1	$\langle e_1 \rangle$	P_1, P_2, P_3, P_6, P_7	$\mathrm{Sp}(6,2) \times \mathrm{GU}(1,4)$	10880
M_2	$\langle e_1, e_2 \rangle$	P_1, P_4, P_6, P_7	$\mathrm{Sp}(4,2) \times \mathrm{GU}(2,4)$	3655680
M_3	$\langle e_1, e_2, e_3 \rangle$	P_2, P_4, P_5, P_7	$\mathrm{Sp}(2,2) \times \mathrm{GU}(3,4)$	12185600
M_4	$\langle e_1, e_2, e_3, e_4 \rangle$	P_2, P_3, P_4, P_5, P_6	$\mathrm{GU}(4,4)$	609280

Proof. See the proof of Lemma 4.4.1. □

Based on the above table, we find presentations of the maximal parabolics on the generators $d_1, d_2, d_3, d_4, p_1, p_2, p_3, p_4, p_5, p_6, p_7$. These, together with the union of the relators of said presentations give a presentation for the universal completion of the amalgam of the maximal parabolics. The following presentations define each maximal parabolic.

Coset enumeration over the subgroup generated by $d_1, d_2, d_3, d_4, p_1, p_2, p_3, p_6, p_7$ gives an index of 10880 which corresponds to the index of M_1 in $\mathrm{Sp}(8,2)$. By Lemma 2.1.6 this shows that $\mathrm{Sp}(8,2)$ is the universal completion of the amalgam of maximal parabolics.

4.9. Summary

Combining the results from the preceding sections yields this proposition:

Proposition 4.9.1. *For* $(n,q) \in \{(3,3),(3,4),(3,5),(3,7),(4,2)\}$*,* G_σ *is the universal completion of* $\mathcal{A}_{(n-1)}$. □

All computations (except those for $(n,q) = (3,7)$) were performed on an Apple PowerBook G4 1.5Ghz with 1 GB RAM using GAP 4.4.5. Computation times ranged from a few seconds up to about half an hour (for $(n,q) = (3,5)$); memory requirements ranged up to 350 MB (again for $(n,q) = (3,5)$). Details regarding $(n,q) = (3,7)$ are mentioned in Section 4.7.

5. Simple connectedness of the geometry

In this chapter we will prove that for $n = 3, q \geq 7$, the geometry $\mathcal{G}_C^{\mathrm{herm}}$ is simply connected. By the following lemma, this implies that $\mathrm{Sp}(6, q)$ is the universal completion of the amalgam of its maximal parabolics, as desired. This yields another proof for the case $(3, 7)$ which we already treated in Section 4.7.

Tits' Lemma. *Suppose a group G acts flag-transitively on a geometry $\mathcal{G}$, and let $\mathcal{A}$ be the amalgam of parabolics associated with some maximal flag F of $\mathcal{G}$. Then G is the universal completion of the amalgam $\mathcal{A}$ if and only if $\mathcal{G}$ is simply connected.*

For a proof, refer for example to [8], [12], [16] or [26].

5.1. Simple connectedness

We have transformed our group theoretic problem (analyzing the universal completion of an amalgam) into a geometric one (showing that a certain geometry is simply connected). We now have to consider how to solve the latter problem. In particular, it is not immediately clear that this new problem is easier than the original problem. We need some mathematical tools and facts in order to tackle it successfully.

Being simply connected means the following for our geometry: All cycles in its incidence graph have to be null-homotopic (for details, see for example [22]).

If $q \geq 3$, every cycle in the incidence graph of $\mathcal{G}_C^{\mathrm{herm}}$ is homotopic to a cycle passing exclusively through points and lines (Lemma 5.1 in [15]). Since $\mathcal{G}_C^{\mathrm{herm}}$ is a partially linear geometry, i.e., distinct points have at most one line joining them, the points of such a cycle uniquely determine the lines of the cycle. Hence it suffices to study cycles of the collinearity graph of $\mathcal{G}_C^{\mathrm{herm}}$. Since the diameter of the collinearity graph is two (see Lemma 4.5 in [15]), every cycle of length at least six always decomposes into smaller cycles (i.e. it is the sum of these smaller cycles), and hence it suffices to study triangles, quadrangles and pentagons of the collinearity graph in order to prove simple connectedness.

5.2. Some tools

The following lemma will prove to be very useful throughout the whole section.

Recall the terminology and definitions introduced in Section 2.3. Notice that if l is a two-dimensional subspace of V of $((\cdot,\cdot))$-rank at least one, then it contains at least

$q^2 - q$ points of $\mathcal{G}_C^{\text{herm}}$. Indeed, if the $((\cdot,\cdot))$-rank of l is one then the radical is the only non-trivial isotropic subspace of l and if the $((\cdot,\cdot))$-rank of l is two then l contains $q+1$ distinct non-trivial isotropic subspaces. In particular, any point of $\mathcal{G}_C^{\text{herm}}$ is $(\cdot,\cdot)$-singular, and hence if l has $((\cdot,\cdot))$-rank one (respectively, two) it contains q^2 (respectively, q^2-q) points of $\mathcal{G}_C^{\text{herm}}$.

Lemma 5.2.1. *Let p be a point of $\mathcal{G}_C^{\text{herm}}$ and $\Pi \supset p$ be a three-dimensional subspace of V of $((\cdot,\cdot))$-rank at least two such that p is in the $(\cdot,\cdot)$-radical of Π. Then for any $((\cdot,\cdot))$-nondegenerate two-dimensional subspace l of Π, all points of $\mathcal{G}_C^{\text{herm}}$ incident with l are collinear to p, with the exception of at most $q+1$ points.*

Proof. Since p is in the $(\cdot,\cdot)$-radical of Π, all lines passing through p will be totally isotropic with respect to $(\cdot,\cdot)$ so we only need to consider $((\cdot,\cdot))$. Note that a two-dimensional subspace of Π has at least $((\cdot,\cdot))$-rank one.

Consider $l_1 = p^{\perp\!\!\perp} \cap \Pi$. Then there are at least $q^2 - q$ lines of $\mathcal{G}_C^{\text{herm}}$ incident to p that intersect l_1 in a point of $\mathcal{G}_C^{\text{herm}}$. If l is any other two-dimensional subspace of Π of $((\cdot,\cdot))$-rank at least one not containing p, then of the q^2+1 one-dimensional subspaces (including the points) it contains, q^2-q intersect one of these lines. Hence at most $q+1$ do not intersect any of the lines, from which the lemma follows. □

A direct consequence of this is that if l has $((\cdot,\cdot))$-rank one (respectively, two) it contains at least q^2-q-1 (respectively, q^2-2q-1) points collinear to p. Furthermore, we actually showed:

Lemma 5.2.2. *Let p be a point of $\mathcal{G}_C^{\text{herm}}$ and $\Pi \supset p$ be a three-dimensional subspace of V of $((\cdot,\cdot))$-rank at least two. Then any two-dimensional subspace l of Π not containing p is incident with at least q^2-q-1 (respectively, q^2-2q-1) points of $\mathcal{G}_C^{\text{herm}}$ that generate a $((\cdot,\cdot))$-nondegenerate two space with p if l has $((\cdot,\cdot))$-rank one (respectively, two).* □

5.3. Triangles

The first step is the analysis of triangles of the collinearity graph. We will call a triangle (a,b,c) a **good triangle** if a, b and c are incident to a common plane of the geometry. A triangle that is not good is called **bad**. Note that a good triangle is null-homotopic, so we only have to deal with the bad ones.

Lemma 5.3.1. *Let (a,b,c) be a bad triangle. Then we can decompose this triangle into bad triangles, in such a way that for each new triangle T_i we can find a canonical basis*

$$e_1, e_2, e_3, f_1, f_2, f_3$$

of V such that each T_i equals

$$\langle e_1\rangle, \langle e_2\rangle, \langle x_i e_1 + y_i e_2 + (k_i e_3 + f_3)\rangle$$

with $k_i\bar{k}_i = -1$ and $x_i y_i \neq 0$ and $x_i\bar{x}_i + y_i\bar{y}_i \neq 0$.

Proof. This is a consequence of the Lemmas 5.3, 5.4 and 6.1 in [15]. □

So by the preceding Lemma we only have to show for a very limited class of (bad) triangles that they can be decomposed.

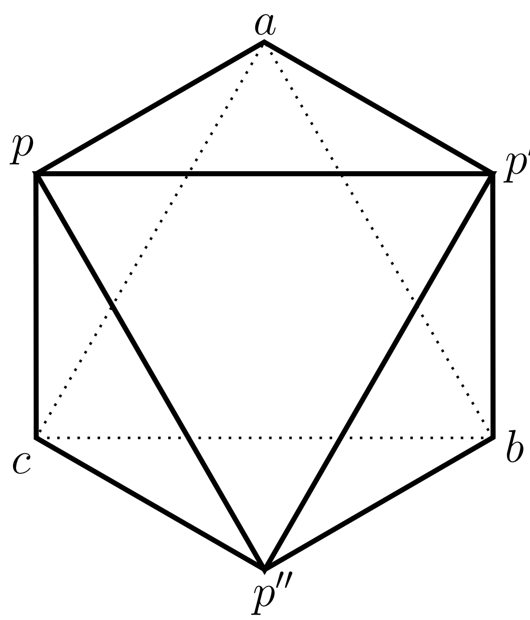


Figure 5.1.: Octahedron construction used in Lemma 5.3.2

To do this, we start with a triangle (a, b, c) and construct an octahedron with the triangle forming one face, and a suitably chosen null-homotopic triangle (p, p', p'') forming the opposite face, as depicted in Figure 5.1. With suitably chosen we mean that all triangles except for the starting triangle shall be decomposable. In the following we will prove that this is possible for $q \geq 4$.

Before we do that, we need some more tools.

Lemma 5.3.2. *Let $k, l \in \mathbb{F}_{q^2}$ such that $k\bar{k} = -1$, $l \neq 0$ Then there exists a matrix of the form*

$$A := \begin{pmatrix} 1 & & & & & \\ & 1 & & & & \\ & & x & & & -\bar{y} \\ & & & 1 & & \\ & & & & 1 & \\ & & y & & & \bar{x} \end{pmatrix} \in G_\sigma$$

such that $(ke_3 + f_3)A = (k\bar{l}e_3 + lf_3)$.

Proof. It is easy to verify that $A \in G_\sigma$ if and only if $x\bar{x} + y\bar{y} = 1$. Furthermore

$$(ke_3 + f_3)A = (kx + y)e_3 + (\bar{x} - k\bar{y})f_3 = k\overline{(\bar{x} - k\bar{y})}e_3 + (\bar{x} - k\bar{y})f_3.$$

So the claim is equivalent to showing that the following system of equations has a solution:

$$x\bar{x} + y\bar{y} = 1 \qquad \text{and} \qquad \bar{x} - k\bar{y} = l$$

Finding such a solution is easily achieved via straight forward computation: use the second equation to replace the x variable in the first equation:

$$\begin{aligned} \overline{(l + k\bar{y})}(l + k\bar{y}) + y\bar{y} &= 1 \\ \iff \qquad l\bar{l} + l\bar{k}y + \overline{(l\bar{k}y)} &= 1 \\ \iff \qquad z + \bar{z} &= 1 - l\bar{l} \in \mathbb{F}_q \end{aligned}$$

where $z := l\bar{k}y$. Now if r is a primitive root of $\mathbb{F}_{q^2}$, then $r + \bar{r} \neq 0$ and hence $z = \frac{r(1-l\bar{l})}{r+\bar{r}}$ is a solution to this last equation. Backward substitution yields the desired values for x and y. □

Lemma 5.3.3. *For $4 \leq q \leq 11$, any bad triangle can be decomposed into good triangles.*

Proof. Let a, b, c be a bad triangle. By Lemma 5.3.1, we can assume

$$(a, b, c) = (\langle e_1 \rangle, \langle e_2 \rangle, \langle xe_1 + ye_2 + (ke_3 + f_3) \rangle)$$

satisfying $k\bar{k} = -1$ and $xy \neq 0$ and $x\bar{x} + y\bar{y} \neq 0$.

Since $x \neq 0$, by Lemma 5.3.2 we can find $g \in G_\sigma$ such that

$$(a^g, b^g, c^g) = (\langle e_1 \rangle, \langle e_2 \rangle, \langle xe_1 + ye_2 + (k\bar{x}e_3 + xf_3) \rangle) = (\langle e_1 \rangle, \langle e_2 \rangle, \langle e_1 + y'e_2 + (k'e_3 + f_3) \rangle)$$

with $y' := \frac{y}{x}$ and $k' := \frac{k\bar{x}}{x}$. So every bad triangle is conjugate to such a triangle. Note that $k'\bar{k}' = \frac{k\bar{k}x\bar{x}}{x\bar{x}} = -1$, so k' can take at most $q + 1$ different values. Since $y' \neq 0$, it can take at most $q^2 - 1$ different values. Hence there are at most $(q + 1)(q^2 - 1)$ different conjugacy classes of bad triangles to consider.

It is now a simple matter of combinatorics to determine all the possible conjugacy classes of bad triangles for a given q, and then testing for each whether the triangle defined this way is decomposable. We now claim that for $4 \leq q \leq 11$ this is possible by using the octahedron construction described above, and setting

$$(p, p', p'') = (\langle f_3 \rangle, \langle se_1 + kf_1 - xf_3 \rangle, \langle te_2 + kf_1 - yf_3 \rangle)$$

where $s, t \in \mathbb{F}_{q^2} \setminus \{0\}$ are chosen suitably.

Verifying that this is possible requires at most $(q+1)(q^2-1)^3$ checks. This can readily be done using a simple GAP program (see Appendix D). In particular we successfully performed these checks for $4 \leq q \leq 11$. [1] □

[1] This upper bound could easily be increased, but of course we had to stop at some point. We picked it so that it complements the previous proof presented in [15] which works without computer help for $q \geq 13$. So stopping at $q = 11$ is arbitrary, and the code should work for bigger values of q, too.

5.4. Quadrangles

Now we will shift our attention to quadrangles. By the preceding results, it is enough to decompose quadrangles into triangles, regardless whether they are good or bad. Notice that if in a quadrangle a, b, c, d we have that a and c (or b and d) are collinear then this quadrangle is immediately decomposed into two triangles.

Definition 5.4.1. We call a quadrangle a, b, c, d *half-special* if $\langle a, c\rangle$ or $\langle b, d\rangle$ is nondegenerate with respect to both forms $(\cdot,\cdot)$ and $((\cdot,\cdot))$. We call it *special* if both $\langle a, c\rangle$ and $\langle b, d\rangle$ are nondegenerate with respect to both forms.

Lemma 5.4.2. *Let $q \geq 5$. Then any quadrangle can be decomposed into triangles and half-special quadrangles.*

Proof. Consider an arbitrary quadrangle a, b, c, d. Without loss of generality we may assume that b and d are noncollinear. Pick an arbitrary point

$$s \in X = a^{\perp\!\!\perp} \cap b^{\perp} \cap d^{\perp}.$$

The point s exists because X is not totally isotropic with respect to $((\cdot,\cdot))$, being a three-dimensional space contained in the nondegenerate five-dimensional space $a^{\perp\!\!\perp}$. The line $l = \langle a, s\rangle$ has $((\cdot,\cdot))$-rank two. Using Lemma 5.2.2, the line l contains at least q^2-2q-1 points of $\mathcal{G}_C^{\mathrm{herm}}$ that are collinear with b, respectively d, and at least q^2-2q-1 points of $\mathcal{G}_C^{\mathrm{herm}}$ that generate a nondegenerate two-dimensional space with c. Since $q \geq 5$ and since l contains q^2-q points of $\mathcal{G}_C^{\mathrm{herm}}$, the space l has to contain a point p of $\mathcal{G}_C^{\mathrm{herm}}$ that generates a nondegenerate two-dimensional space with c and that is collinear to both b and d. Clearly a, b, c, d decomposes into a, b, p, d and c, b, p, d. If $(a,p) = 0$ then $\langle a, p\rangle$ is a line, implying that a, b, p, d decomposes into triangles. Otherwise, a, b, p, d is half-special w.r.t. $\langle a, p\rangle$. Similarly for c, b, p, d. □

Corollary 5.4.3. *Let $q \geq 5$. Then any quadrangle can be decomposed into triangles and special quadrangles.*

Proof. Apply Lemma 5.4.2 once to obtain triangles and half-special quadrangles. Then apply Lemma 5.4.2 again, after suitably renaming the vertices of the quadrangles, to obtain special quadrangles. □

Proposition 5.4.4. *Let $q \geq 7$. Then any quadrangle can be decomposed into triangles.*

Proof. Denote the quadrangle by

$$(a, b, c, d).$$

By the preceding lemma, we can assume w.l.o.g. that it is special, so $(a, c) \neq 0 \neq (b, d)$ and both $\langle a, c\rangle$ and $\langle b, d\rangle$ are $((\cdot,\cdot))$-nondegenerate. We try to find a point p collinear to all of a, b, c, d, which means we can decompose the quadrangle into triangles (see Figure 5.2).

Set

$$W := a^{\perp} \cap c^{\perp} \quad \text{and} \quad U_1 := W \cap b^{\perp} \quad \text{and} \quad U_2 := W \cap d^{\perp} \quad \text{and} \quad l := U_1 \cap U_2.$$

Note that $\dim W = 4$, $\dim U_1 = \dim U_2 = 3$, $\dim l = 2$. Also, W is $((\cdot,\cdot))$-nondegenerate since $\langle a, c\rangle$ is $((\cdot,\cdot))$-nondegenerate and

$$W = a^{\perp} \cap c^{\perp} = (a^{\sigma})^{\perp\!\!\perp} \cap (c^{\sigma})^{\perp\!\!\perp} = \langle a^{\sigma}, c^{\sigma}\rangle^{\perp\!\!\perp} = (\langle a, c\rangle^{\sigma})^{\perp\!\!\perp}.$$

Similar arguments hold for $a^{\perp} \cap b^{\perp}$, $b^{\perp} \cap c^{\perp}$ and so on.

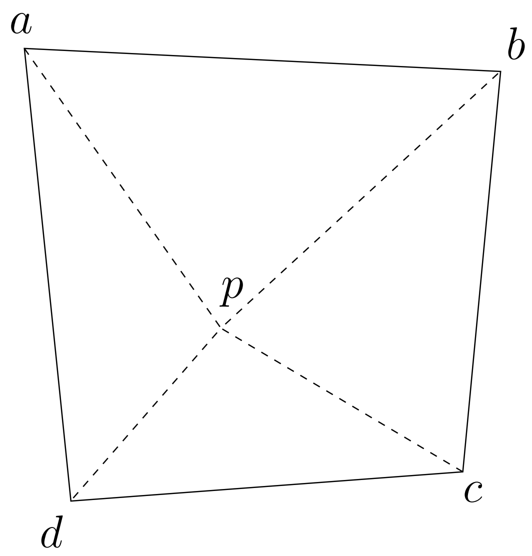


Figure 5.2.: Basic quadrangle decomposition as used in Proposition 5.4.4

We now distinguish three cases:

(1) If l is of $((\cdot,\cdot))$-rank two, then we can apply Lemma 5.2.1 to the planes $\langle a, l\rangle$, $\langle b, l\rangle$, $\langle c, l\rangle$, and $\langle d, l\rangle$ to obtain $q^2 - 5q - 4$ points of $\mathcal{G}_C^{\text{herm}}$ on l collinear to all of a, b, c, d. Notice that this is a positive number for $q \geq 7$.

(2) Suppose now that l is of $((\cdot,\cdot))$-rank one. Then the plane $\Pi := \langle b, l\rangle$ has $((\cdot,\cdot))$-rank at least one. It lies inside the four-dimensional $((\cdot,\cdot))$-nondegenerate space W. Assume Π had $((\cdot,\cdot))$-rank one. Then it has a two-dimensional $((\cdot,\cdot))$-radical R, which would be maximal totally isotropic in W, since $\dim(R) + \dim(R^{\perp\!\!\perp} \cap W) = \dim(W)$ and $R \subseteq R^{\perp\!\!\perp}$. Similarly, R can not have a polar of dimension three, which Π would be. Contradiction, thus Π has $((\cdot,\cdot))$-rank two. Similar arguments hold

for the points a, c, d instead of b. Applying Lemma 5.2.1 gives us $q^2 - 4q - 4$ points of $\mathcal{G}_C^{\text{herm}}$ collinear to all of a, b, c, d. Notice that this is a positive number for $q \geq 5$.

(3) Suppose now l is totally isotropic with respect to $((\cdot,\cdot))$. Then the planes U_1 and U_2 are $((\cdot,\cdot))$-degenerate. They must have $((\cdot,\cdot))$-rank two (this can be shown with similar arguments as used in case (2) for Π).

Let R_1 and R_2 be the one-dimensional $((\cdot,\cdot))$-radicals of U_1 and U_2. They are contained in l. For assume that $R_1 \not\subseteq l$; then $U_1 = \langle R_1, l\rangle$. But then U_1 would be totally isotropic (since l is totally isotropic, and also orthogonal to R_1, the radical of U_1), a contradiction. We argue likewise for R_2.

Furthermore, the radicals cannot coincide as otherwise we would obtain a radical for the $((\cdot,\cdot))$-nondegenerate space $a^\perp \cap c^\perp$. So we have $l = \langle R_1, R_2\rangle$. Notice that $b \not\subseteq l$, since $(b,d) \neq 0$. Hence b is different from R_1 and R_2.

Choose a line t of $\mathcal{G}_C^{\text{herm}}$ through b inside U_1. This line exists since the $((\cdot,\cdot))$-rank of U_1 is two, and b is not in the $((\cdot,\cdot))$-radical R_1 of U_1. Applying first Lemma 5.2.2 to $\langle d,t\rangle$ and then Lemma 5.2.1 to $\langle a,t\rangle$ and $\langle c,t\rangle$ yields the existence of

$$(q^2 - 2q - 1) - 2(q+1) = q^2 - 4q - 3 > 0$$

points on t collinear to a, b, c and which span a $((\cdot,\cdot))$-nondegenerate space with d. Choose one of these points not equal to b and call it b'. Then $(b',d) \neq 0$, for otherwise, $b' \in l$, contradicting that l is totally isotropic with respect to $((\cdot,\cdot))$. Hence a, b', c, d form a special quadrangle (see Figure 5.3).

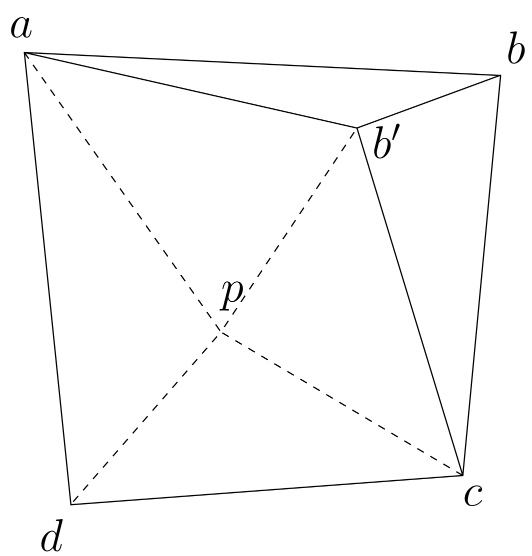


Figure 5.3.: Quadrangle decomposition in case l is degenerate.

Let $U_1' := b'^{\perp} \cap W$. We claim that U_1' intersects U_2 in a line l' that does not contain R_2, implying the $((\cdot,\cdot))$-rank of l' is two (since it is contained in U_2 which has $((\cdot,\cdot))$-rank two and doesn't intersect its radical) and so we have reduced to case (1) of this proof.

It remains to verify our last claim. Assume $R_2 \subset U_1' \cap U_2 = l'$. Then

$$R_2 \subseteq l \cap l' = (b^{\perp} \cap b'^{\perp}) \cap U_2 = (\langle b, b' \rangle^{\perp}) \cap U_2 \subset \langle b, b' \rangle^{\perp} = t^{\perp},$$

thus $t \subseteq R_2^{\perp} \cap U_1$. Notice that $R_2^{\perp} \cap U_1 = \langle b, R_2 \rangle$: Clearly $\langle b, R_2 \rangle \subseteq R_2^{\perp} \cap U_1$, since $R_2 \subset R_2^{\perp}$, $R_2 \subset l \subset U_1$, $b \subset U_1$ and $b \subset l^{\perp} \subset R_2^{\perp}$. Equality holds since R_2 is one-dimensional, and R_2 is not the $(\cdot,\cdot)$-radical of U_1 (which is b, and we already know that $b \neq R_2$), and thus both sides of the equation have the same dimension. But then also $t = \langle b, R_2 \rangle$, implying that t has $((\cdot,\cdot))$-rank one, a contradiction since t is a line of $\mathcal{G}_C^{\mathrm{herm}}$.

□

Note that the 'pyramid' construction used in the preceding proposition is **not** sufficient for $q \leq 5$, so a different approach would be needed to cover it. For a specific example, let z denote a primitive element in $\mathbb{F}_{25}$ over $\mathbb{F}_5$ with minimal polynomial $x^2 - x + 2$. Then let

$$a := \langle e_1 \rangle, \quad b := \langle e_2 \rangle, \quad c := \langle e_2 + z^{-1} e_3 + z^{-1} f_1 \rangle, \quad d := \langle e_1 + z^5 e_2 + z^6 f_2 + z^9 f_3 \rangle.$$

This is a special quadrangle, and using the definitions from Proposition 5.4.4, $l := \langle u, v \rangle$ with $u := \langle e_1 + f_3 \rangle, v := \langle e_2 + z^9 e_3 \rangle$. Now l has $((\cdot,\cdot))$-rank two, but contains no point p collinear to all of a, b, c, d.

5.5. Pentagons

Proposition 5.5.1. *Let $q \geq 5$. Then any pentagon can be decomposed into triangles and quadrangles.*

Proof. Let (a, b, c, d, e) be a pentagon. Consider the space $U := \langle a, b, d \rangle^{\perp}$ of dimension three. Its $((\cdot,\cdot))$-rank has to be at least two, as the $((\cdot,\cdot))$-rank of $\langle a, b \rangle$ is two. Choosing a $((\cdot,\cdot))$-nondegenerate two-dimensional subspace l of U and applying Lemma 5.2.1 on the planes $\langle a, l \rangle$, $\langle b, l \rangle$, $\langle d, l \rangle$, we will find

$$(q^2 - q) - 3(q + 1) = q^2 - 4q - 3 > 0$$

points on l collinear to all of a, b, d, decomposing the pentagon. □

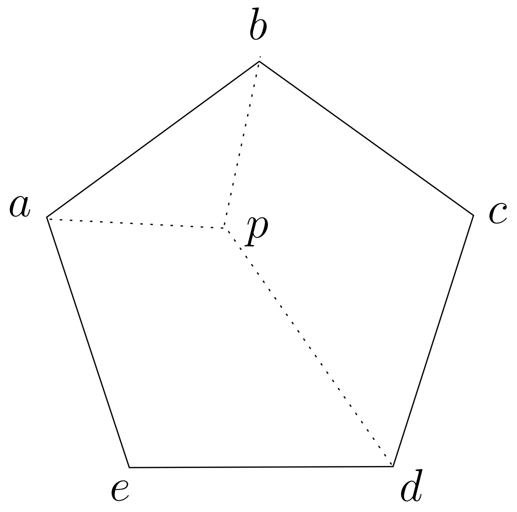


Figure 5.4.: Pentagon decomposition as used in Proposition 5.5.1

5.6. Summary

Combining the results from the preceding sections yields this proposition:

Proposition 5.6.1. *If $n = 3$ and $7 \leq q \leq 11$, the geometry $\mathcal{G}_C^{herm}$ is simply connected.*

Using the above and the work done in [15] we can now proof our main result in Section 3.1.

Proof of Theorem 3.1.1. We show that the geometry $\mathcal{G}_C^{herm}$ is simply connected if and only if $n \geq 3$ and $(n, q) \neq (3, 2)$. From this follows the claim via Tit's Lemma (page 20).

Simple connectedness for $n \geq 3$ and $(n, q) \neq (3, 2)$ is proved conjointly by Proposition 5.6.1, by combining the results from Chapter 4 with Tits' Lemma, and finally by Theorem 6.8 from [15].

If $(n, q) = (3, 2)$, then the geometry is not simply connected, as shown in [15], right after Theorem 6.8.

Finally, if $n = 2$, the simplicial complex is one dimensional, and hence only simply connected if it contains no cycles (i.e. if it is a tree). But the points $\langle e_1 \rangle$, $\langle e_2 \rangle$, $\langle f_1 \rangle$, $\langle f_2 \rangle$ form a quadrangle, and hence there exists a non-trivial cycle in the simplicial complex, thus the geometry is not simply connected. □

6. Phan-type theorems

As was already mentioned in Section 3.2, the main motivation of this thesis was to strengthen the Phan-type theorems for C_n. We will not give a proper proof of this theorem, as the work done previously in this regard remains essentially unchanged – our new bounds simply have to be plugged in. We will only sketch the steps of the proof.

We first give some required definitions in Sections 6.1 and 6.2. In Section 6.4 we outline the proof of the Phan-type theorems for C_n as given in Section 6.3.

6.1. Diagrams

Diagrams, in particular Dynkin diagrams first occured in the classification work of Cartan (for complex semisimple Lie algebras) and Coxeter (for finite reflection groups). For a proper introduction to the theory of diagrams, see for example [6] or [19]. However, to understand the (limited) use of diagrams in this thesis, the reader does not have to know the full theory behind them.

For our purposes, a diagram is a graph without loops but possibly with multiple edges, and where multiple edges are directed.

We will use Diagrams to describe subgroups of a group G: If $(G_i)_{i\in I}$ is a finite family of subgroups of G, then we form a diagram where the nodes correspond to these subgroups. The kind of edge (or lack of an edge) between two nodes i and j is used to encode how the two subgroups G_i and G_j are related.

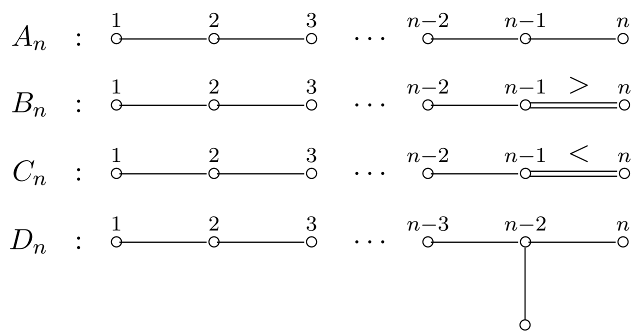
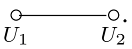


Figure 6.1.: Four fundamental Dynkin diagram series.

The infinite series of diagrams shown in Figure 6.1 play an important role in several mathematical topics, especially in classification of (finite) reflection groups and in Lie theory. In fact, the theorems presented in this thesis correspond in a certain way to the diagram C_n. See also Appendix A.

6.2. Phan systems

Definition 6.2.1 (Standard pair in unitary group)**.** Subgroups U_1 and U_2 of $\mathrm{SU}(3,q^2)$ form a **standard pair** whenever each $U_i \cong \mathrm{SU}(2,q^2)$ is the stabilizer in $\mathrm{SU}(3,q^2)$ of a nonsingular vector v_i and, furthermore, v_1 and v_2 are perpendicular. Standard pairs in central quotients of $\mathrm{SU}(3,q^2)$ are defined as the images under the natural homomorphism of the standard pairs from $\mathrm{SU}(3,q^2)$. We denote a standard pair U_1, U_2 of a central quotient of $\mathrm{SU}(3,q^2)$ by

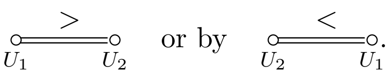


In the following we use the terminology from Section 2.3.

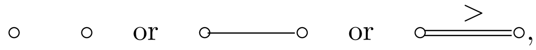
For an element U of $\mathcal{G}_C^{\mathrm{herm}}$, i.e. a $(\cdot,\cdot)$-totally singular, $((\cdot,\cdot))$-nondegenerate subspace of V, let $\mathrm{GU}(U)$ denote the subgroup of G_σ that preserves the form $((\cdot,\cdot))\ |_{U\times U}$ and acts trivially on $U^\perp \cap U^{\perp\!\!\perp}$.

For a nondegenerate σ-invariant subspace W of V denote by $\mathrm{Sp}(W)$ the subgroup of G_σ that preserves the form $(\cdot,\cdot)_{|W_\lambda \times W_\lambda}$ (see 4.2.1 for the definition of W_λ) and acts trivially on $U^\perp \cap U^{\perp\!\!\perp}$.

Definition 6.2.2 (Standard pair in symplectic group)**.** In case $n = 2$, we have $V = \langle e_1, e_1^\sigma, e_2, e_2^\sigma \rangle$, $G \cong \mathrm{Sp}(4,q^2)$, and $G_\sigma \cong \mathrm{Sp}(4,q)$. Subgroups $U_1 \cong \mathrm{Sp}(2,q)$ and $U_2 \cong \mathrm{SU}(2,q^2)$ are called a **standard pair** in G_σ if there exists a $(\cdot,\cdot)$-isotropic and $((\cdot,\cdot))$-non-isotropic vector v of V and a two-dimensional $(\cdot,\cdot)$-totally isotropic and $((\cdot,\cdot))$-nondegenerate subspace $U \ni v$ of V such that the group U_1 coincides with $\mathrm{Sp}(v^\perp \cap v^{\perp\!\!\perp})$ and the group U_2 coincides with $\mathrm{SU}(U)$. Standard pairs in central quotients of $\mathrm{Sp}(4,q)$ are defined as the images under the natural homomorphism of the standard pairs from $\mathrm{Sp}(4,q)$. We denote the standard pair U_1, U_2 of $\mathrm{Sp}(4,q)$ by



Definition 6.2.3 (Weak Phan System)**.** Let $n \geq 2$, let Δ be a Dynkin diagram with rank two subdiagrams isomorphic to



and let $I = \{1, \dots, n\}$. A group G admits a **weak Phan system of type Δ over $\mathbb{F}_{q^2}$** if G contains subgroups

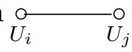
$$U_i \cong \mathrm{SL}(2, q) \cong \mathrm{Sp}(2, q) \cong \mathrm{SU}(2, q^2),$$

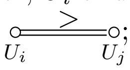
for $i \in I$, and subgroups

$$U_{ij},$$

for $i \neq j \in I$, so that the following hold:

(wP1) If (i, j) is not an edge in Δ, then U_{ij} is a central product of U_i and U_j;

(wP2) if (i, j) is an edge in Δ, then U_{ij} is isomorphic to a central quotient of $\mathrm{SU}(3, q^2)$, if (i, j) is a single edge, and isomorphic to a central quotient of $\mathrm{Sp}(4, q)$, if (i, j) is a double edge; moreover, U_i and U_j form a standard pair in U_{ij} according to the diagram 

or 

; and

(wP3) the subgroups $U_{ij}, i, j \in I$, generate G.

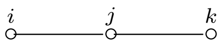
6.3. Phan-type theorems

Note that the first theorem is the interesting case; the second one is a (somewhat complicated) generalization to $q = 2$.

Main Theorem 1. *Let $q \geq 3$, let $n \geq 3$, and let G be a group that contains a weak Phan system of type C_n over $\mathbb{F}_{q^2}$. Then G is isomorphic to a central quotient of $\mathrm{Sp}(2n, q)$.*

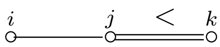
Main Theorem 2. *Let $q = 2$, let $n \geq 4$, and let G be a group that contains a weak Phan system of type C_n over $\mathbb{F}_{q^2}$. Suppose further that*

(1) for any triple i, j, k of nodes of the Dynkin diagram C_n that form a subdiagram



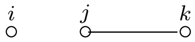
of type A_3, the subgroup $\langle U_{i,j}, U_{j,k}\rangle$ is isomorphic to a central quotient of $\mathrm{SU}(4, q^2)$;

(2) for any triple i, j, k of nodes of the Dynkin diagram C_n that form a subdiagram



of type C_3, the subgroup $\langle U_{i,j}, U_{j,k}\rangle$ is isomorphic to a central quotient of $\mathrm{Sp}(6, q)$;

(3) (i) *for any triple i, j, k of nodes of the Dynkin diagram C_n that form a subdiagram*

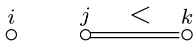


of type $A_1 \oplus A_2$, the groups U_i and $U_{j,k}$ commute elementwise; and

(ii) *for any quadruple of nodes of the Dynkin diagram C_n that form a subdiagram*

of type $A_2 \oplus A_2$, the groups $U_{i,j}$ and $U_{k,l}$ commute elementwise; and

(iii) *for any triple i, j, k of nodes of the Dynkin diagram C_n that form a subdiagram*



of type $A_1 \oplus C_2$, the groups U_i and $U_{j,k}$ commute elementwise; and

(iv) *for any quadruple of nodes of the Dynkin diagram C_n that form a subdiagram*

of type $A_2 \oplus C_2$, the groups $U_{i,j}$ and $U_{k,l}$ commute elementwise.

Then G is isomorphic to a central quotient of $\mathrm{Sp}(2n, q)$.

6.4. Sketch of proof

The goal is to arrive at a configuration of small subgroups similar to the one Phan presented, more specifically, a so-called **weak Phan system of type C_n** over $\mathbb{F}_{q^2}$ (see Section 6.2).

The configuration we are starting with, $\mathcal{A}_{(n-1)}$, is in a sense 'too big' – for a Phan system, we need a configuration of low dimensional groups, which is not the case here. But via an induction argument one can show that the amalgams of certain lower rank parabolics are already sufficient to define the group $\mathrm{Sp}(2n, q)$. This is what leads to Theorem 3.1.2.

Note now the isomorphism types of the maximal parabolics:

$$M_i \cong \mathrm{Sp}(2n-2i, q) \times \mathrm{GU}(i, q^2), \quad 1 \le i \le n$$

(see Section 4.3 for details). The intersection of all M_i is a maximal half-split torus T of $\mathrm{Sp}(2n, q)$ isomorphic to $\mathrm{GU}(1, q^2)^n$ and formed by diagonal matrices. We define the **stripped parabolics** as subgroups of M_i of the form

$$M_i^0 \cong \mathrm{Sp}(2n-2i, q) \times \mathrm{SU}(i, q^2), \quad 1 \le i \le n.$$

For an arbitrary parabolic M_J define $M_J^0 := \bigcap_{i \in J} M_i^0$, where J is a subset of the type set $I = \{1, \dots, n\}$. It can be shown that $M_J = M_J^0 T$, which explains the name 'stripped' – we have removed (stripped) the torus T from the stabilizers. We explain below why stripping is necessary.

One can show that the minimal stripped parabolics $L_i := M_{I \setminus i}^0$ are isomorphic to $\mathrm{SL}(2, q)$, corresponding to the U_i in the definition of a weak Phan system. We define $\mathcal{A}_{(s)}^0$ to be the amalgam formed by the subgroups M_J^0 for all parabolics M_J of rank s (obviously this includes the L_i). This stripped amalgam can be shown to induce a weak Phan system of type C_n over $\mathbb{F}_{q^2}$ in any nontrivial completion. It remains to classify certain classes of so-called Phan amalgams (again, see [13]). All in all, the reward for these efforts are the two Phan-type theorems we presented in Section 6.3.

6.5. Why stripping the parabolics is necessary

Before we conclude this chapter, we give a brief argument why we are using stripped parabolics instead of plain parabolics.

We already mentioned that the minimal stripped parabolics L_i are isomorphic to $\mathrm{SL}(2, q)$, corresponding to the U_i in the definition of a weak Phan system. If we had not stripped the torus, the isomorphism type would have been $\mathrm{SL}(2, q) \rtimes \mathrm{GU}(1, q^2)^{n-1}$. So the stripping ensures that we obtain a weak Phan system of type C_n over $\mathbb{F}_{q^2}$ in any non-trivial completion.

But why did we define weak Phan systems like this in the first place? Why not abandon the stripping and instead change the definition to match what we have? Because then the main theorem would not follow! Consider this example: If we form the amalgam of parabolics in $\mathrm{PSp}(2n, q)$ and compute its universal completion, we arrive again at $\mathrm{PSp}(2n, q)$. But for our Main Theorems, we really would like to get $\mathrm{Sp}(2n, q)$, showing that $\mathrm{PSp}(2n, q)$ is a central quotient of $\mathrm{Sp}(2n, q)$. Of course we already knew that, but there are other less trivial cases where the same problem arises.

However, the universal completion of the *stripped* amalgam of $\mathrm{PSp}(2n, q)$ is indeed $\mathrm{Sp}(2n, q)$, as desired. So the stripping 'removes' the difference between the amalgams of $\mathrm{Sp}(2n, q)$ and $\mathrm{PSp}(2n, q)$ (a bit more technically spoken, what happens here is that we throw away the maximal torus and reconstruct it from the amalgam). This might help convey a certain understanding as to why the stripping is important in order to arrive at a Phan-type theorem.

A. The case $(n, q) = (3, 3)$ reviewed

In this thesis, we have dealt with the groups $\mathrm{Sp}(2n, q)$ and extended a Phan-type theorem for these groups. Note that this group corresponds to the family of Dynkin diagrams C_n. This diagram series is one of four such infinite families, and to each corresponds a certain classical group, a geometry like the one described in Section 2.3, as well as a Phan-type theorem. Figure A.1 lists which diagram corresponds to which group, and cites appropriate references which deal with the corresponding Phan-type theorem.

Type	Group	References
A_n	$\mathrm{SU}(n+1, q^2)$	[4], [12], [20]
B_n	$\mathrm{Spin}(2n+1, q)$	[2]
C_n	$\mathrm{Sp}(2n, q)$	[11], [12], [13], [15]
D_n	$\mathrm{Spin}(2n, q)$	[12], [14], [18], [21]

Figure A.1.: Correspondence between Dynkin diagrams and classic groups.

We briefly want to revisit the case $(n, q) = (3, 3)$ and compare results for it in the four cases listed above.

In this thesis, we proved that for $(n, q) = (3, 3)$, the geometry corresponding to $\mathrm{Sp}(6, 3)$ and C_3 is simply connected, and the group itself is the universal completion of its standard Phan amalgam[1].

In case of A_3 or equivalently D_3, however, one can show that either a three-fold or a nine-fold cover of the geometry exists. (Richard Lyons gave a simple argument for this, see page 86 of [12]). However, nothing was known so far about the universal cover. Recently the author successfully applied the techniques used in this thesis to the case D_3 and determined the universal cover, which turned out to be nine-fold.

Finally, for B_3 nothing was known so far. The author applied the techniques of this thesis here, too. The unexpected result was that the coset enumeration (used to determine the size of the universal completion of the amalgam of parabolics) did not terminate. Based on this, we conjecture that the geometry is not simply connected, and may even admit an infinite cover.

[1] For a definition of a standard Phan amalgam of type C_n, refer to [15].

B. Group presentations

In this appendix, we give finite presentations for the groups described in Chapter 4. These can be used to reproduce the group index calculations mentioned there. With $[\cdot,\cdot]$ we will denote the standard commutator bracket, i.e. $[a,b] := aba^{-1}b^{-1}$.

B.1. $n = 3, q = 3$

We give here presentations of the maximal parabolics on the generators d_1, d_2, d_3, u, v, w. To each presentation the relators d_i^4 for $1 \le i \le 3$ and $[d_i, d_j]$ for $1 \le i < j \le 3$ need to be added.

Generators for M_1: d_1, d_2, d_3, v, w.

Relators for M_1:

$$v^3, w^3, [v,d_1], [w,d_1], [w,d_2], d_2v^{-1}d_3v^{-1}d_3^{-2}, wd_3wd_3wd_3^{-1},$$
$$vwv^{-1}wd_3^{-1}vw^{-1}v^{-1}d_3w^{-1}, d_3vwv^{-1}w^{-1}vw^{-1}v^{-1}d_3^{-1}vwv^{-1},$$
$$d_2vwv^{-1}w^{-1}v^{-1}d_3^{-1}d_2^{-1}wvwv^{-1}w^{-1}d_3w$$

Generators for M_2: d_1, d_2, d_3, u, w.

Relators for M_2:

$$u^3, w^3, [u,w], [u,d_3], [w,d_1], [w,d_2], d_1u^{-1}d_2u^{-1}d_2^{-2}, wd_3^2w^{-1}d_3^{-2}, d_3wd_3^{-1}wd_3w$$

Generators for M_3: d_1, d_2, d_3, u, v.

Relators for M_3:

$$u^3, v^3, [v,d_1], [u,d_3], d_2^2vd_2vd_3, ud_1^2d_2ud_1, u^{-1}vuv^{-1}uv$$

B.2. $n = 3, q = 4$

We give here presentations of the maximal parabolics on the generators d_1, d_2, d_3, u, v, w. To each presentation the relators d_i^5 for $1 \le i \le 3$ and $[d_i, d_j]$ for $1 \le i < j \le 3$ need to be added.

Generators for M_1: d_1, d_2, d_3, v, w.

Relators for M_1:

$$v^2, w^2, [v, d_1], [w, d_1], [w, d_2], (wd_3)^3, (vw)^4, vd_2d_3vd_2^{-1}d_3^{-1}, (vd_3d_2^{-1})^3,$$
$$(wvwvd_3^2)^4, wd_1vd_3wd_3d_2vd_3d_2^2vwd_3vd_1^{-1}d_2wvd_3wvwd_3^{-1}vd_2^{-1}d_3wd_3^{-1}wd_3,$$
$$(wd_3^{-2}wvwvd_3^{-2}wd_3^2)^2, d_3d_1d_3vwvwd_1d_3vwvd_3^{-1}vwd_1^{-2}vd_3^{-2}wd_3^{-2}w,$$
$$d_3^{-2}wvd_2^{-1}d_3^2wd_3^{-1}vwvd_3wd_3^{-1}vwvd_3wd_3^{-2}wd_3^2wvd_3wv,$$
$$d_2vd_3wvwd_3^{-1}vwd_2^{-1}d_3vwvd_3wd_3^{-1}wd_3wd_3^{-1}wd_3^2wd_3^{-2}vwvd_3^{-2}$$

Generators for M_2: d_1, d_2, d_3, u, w.

Relators for M_2:

$$u^2, w^2, [u, w], [u, d_3], [w, d_1], [w, d_2], d_3^{-1}wd_3^{-1}wd_3^{-1}w, ud_2^2ud_1^{-2}ud_1^{-1}d_2,$$
$$ud_2d_1ud_1^{-1}d_2^{-1}, ud_1ud_1^{-1}ud_1^{-1}ud_1ud_1^{-1}ud_2, wd_3^{-2}wd_3^2wd_3^{-2}wd_3^2wd_3^{-2}wd_3^2$$

Generators for M_3: d_1, d_2, d_3, u, v.

Relators for M_3:

$$u^2, v^2, [u, d_3], [v, d_1], d_2d_3^{-1}vd_3^{-2}vd_2^2v, ud_1vd_3vd_3^{-1}d_2vd_3vd_1^{-1}ud_3^{-2},$$
$$vd_3^2ud_1vd_3^2uvd_3^{-2}uvd_1^{-1}ud_3^{-2}, d_2d_3^{-1}uvd_1^{-1}ud_3^{-2}vd_1^{-2}d_2d_3^{-1}uvd_1^{-1}ud_3^{-2}d_1^{-2},$$
$$d_2d_3vd_3^{-1}d_2^{-1}v, d_2uvud_3vud_2^2d_3^{-1}uvd_1^{-1}ud_3^{-2}d_1^{-1}d_2d_3^{-1}uvd_1^{-1}ud_3^{-2}d_1^{-1},$$
$$vd_1^{-1}ud_3^{-2}d_2d_3^{-1}uvd_1^{-1}ud_3^{-2}d_1^2d_2d_3^{-1}uvd_1^{-1}ud_3^{-2}d_2^{-1}uvd_3^{-1}uvd_3^{-1},$$
$$ud_2d_1ud_2^{-1}d_1^{-1}, d_2^{-1}vd_3^{-1}vd_3^{-1}d_2^{-2}d_3^2ud_1vud_3d_2^{-1}uvd_2^{-1}vud_1ud_3^2d_1vud_3d_2^{-1}vd_1^{-1},$$
$$d_2d_3^{-1}uvd_1^{-1}uvd_2^{-2}vuvd_1^{-1}d_3^{-2}ud_2d_3^{-1}uvd_1^{-1}d_3^{-2}ud_2d_3^{-1}uvd_1^{-1}d_3^{-2}ud_2d_3^{-1}uvd_1^{-1}d_3^{-2}u$$

B.3. $n = 3, q = 5$

We give here presentations of the maximal parabolics on the generators d_1, d_2, d_3, u, v, w. To each presentation the relators d_i^6 for $1 \leq i \leq 3$ and $[d_i, d_j]$ for $1 \leq i < j \leq 3$ need to be added.

Generators for M_1: d_1, d_2, d_3, v, w.

Relators for M_1:

$$[v, d_1], [w, d_2], [w, d_1], [v, d_2d_3], [w, d_3^3], v^{-3}(d_2d_3)^3, v^{-1}d_2^{-1}d_3^2v^{-1}d_3^{-2}d_2,$$
$$w^{-1}d_3^2w^{-1}d_3w^{-1}d_3^{-1}w^{-1}d_3, d_3w^{-1}d_3wd_3w^2d_3w, d_3^{-1}vd_2^{-2}d_3^{-1}v^{-1}d_3^{-1}vd_2^{-1}v^{-1},$$
$$vd_2^2v^{-1}d_3v^{-1}d_3^{-2}vd_2^{-1}, d_3vwd_2vd_2^{-1}w^{-1}vd_3^{-2}d_2w^{-1}d_2^{-1}v^{-1}wd_3,$$
$$vd_3^{-2}w^{-1}v^{-1}d_3^{-1}wd_3^{-1}w^{-1}vd_2wvd_2^{-1}d_3w^{-1}d_3^{-1}v^{-1}d_3w^{-1}v^{-1}d_3^{-1}w^{-1}d_3,$$
$$d_2v^{-1}wv^{-1}d_3d_2wv^{-1}d_3^{-1}d_2v^{-1}w^{-1}d_2^{-1}d_3^{-2}w^{-1}v^{-1}d_3^{-1}wd_3w^3d_3w,$$
$$d_3vwd_3^{-1}vd_3^{-1}v^{-1}wd_3^{-1}v^{-1}d_3vwd_3^2v^{-1}d_3^{-1}vd_2wvd_2^{-1}d_3^{-1}w^2d_3w^{-1}d_3$$

Generators for M_2: d_1, d_2, d_3, u, w.

Relators for M_2:

$$w^6, [u,w], [w,d_2], [w,d_1], [u,d_3], d_3^3w^3, [u,d_1d_2], (u^{-1}d_1^3)^2, d_3^2wd_3^{-1}w^{-1}d_3^{-1}wd_3^{-1}w^{-1},$$
$$d_1^3u^2d_2^{-3}u^{-1}, wd_3wd_3^{-1}wd_3^{-2}wd_3^{-1}, ud_2^{-1}u^{-1}d_1^{-1}u^{-2}d_1u^{-1}d_2, d_1^2ud_1^{-1}ud_1d_2^{-1}u^{-1}d_1^{-1}u^{-1}$$

Generators for M_3: d_1, d_2, d_3, u, v.

Relators for M_3:

$$[v,d_1], [u,d_3], [v,d_2d_3], [u,d_1d_2], (uv)^{-3}, (v^{-1}d_2^3)^2, (d_1^3u)^{-2}, d_1^3u^2d_2^{-3}u^{-1}, d_2^3v^2d_3^{-3}v^{-1},$$
$$ud_2^{-1}u^{-1}d_1^{-1}u^{-2}d_1u^{-1}d_2, vd_3^{-1}v^{-1}d_2^{-1}v^{-2}d_2v^{-1}d_3, d_2^2vd_2^{-1}vd_2d_3^{-1}v^{-1}d_2^{-1}v^{-1},$$
$$ud_1^{-1}ud_2ud_2^{-2}u^{-1}d_1^2, vuv^{-1}d_3^{-1}vu^{-1}v^{-1}ud_1u^{-1}$$

B.4. $n = 3, q = 7$

We give here presentations of the maximal parabolics on the generators d_1, d_2, d_3, u, v, w. To each presentation the relators d_i^8 for $1 \leq i \leq 3$ and $[d_i, d_j]$ for $1 \leq i < j \leq 3$ need to be added.

Generators for M_1: d_1, d_2, d_3, v, w.

Relators for M_1:

$$[v,d_1], [w,d_1], [w,d_2], [v,d_2d_2], v^3d_2v^{-1}d_3^{-1}, w^2d_3^{-1}wd_3wd_3wd_3^{-1}, d_2^2d_3d_2d_3^3vd_3v,$$
$$d_3wd_3^{-1}w^{-1}d_3^{-1}w^{-1}d_3^{-1}wd_3, v^{-1}d_3wvd_3vw^{-1}d_3^{-1}v^{-1}d_3^{-1}, d_3w^2d_3^{-2}w^{-1}d_3^{-2}w^{-2},$$
$$vd_3w^{-1}vw^{-1}v^{-1}w^3d_3^{-1}wd_3^{-2}w^{-1}d_3v^2d_3wv^{-2}w^{-1}d_3^{-1}w^{-1},$$
$$wv^{-1}d_3^{-1}v^{-1}d_2^{-1}d_3^2d_2^{-1}w^2d_3d_2^{-1}d_1^{-1}v^{-1}d_3^{-1}d_1v^{-1}d_3^2d_2^{-2}w^2d_3^2d_2^{-1},$$
$$d_2v^{-1}d_3v^{-1}w^{-1}d_1v^{-1}d_3v^{-1}d_3^{-2}w^{-1}vd_1d_2d_3^{-1}w^{-2}d_3^{-2}d_2^2vd_1^{-2}d_3^{-3}wd_3,$$
$$d_1^2v^{-1}d_2^{-2}d_3^2w^2d_3d_2^{-1}d_1^{-1}v^{-1}d_1^{-1}d_3wd_3d_1vd_1d_2d_3^{-1}w^{-2}d_3^{-2}d_2^2vd_1^{-2}w^{-1}d_3^{-2},$$
$$d_1vd_1d_2d_3^{-1}w^{-2}d_3^{-2}d_2d_3^{-1}v^{-2}d_3d_2^{-2}d_3^2w^2d_3d_2^{-1}v^{-2}d_2^{-2}d_3^2w^2d_3d_2^{-1}d_1^{-1}v^{-1}d_1^{-1}wd_3,$$
$$d_1v^2d_3wv^{-2}w^{-1}d_1v^{-1}d_2^{-2}d_3^2w^2d_3d_2^{-1}d_1^{-1}v^{-1}w^{-1}d_1v^{-1}d_2^{-2}d_3^2w^2d_3d_2^{-1}d_1^{-1}v^{-1}d_1^{-1}d_3^{-2}w^{-1}d_3$$

Generators for M_2: d_1, d_2, d_3, u, w.

Relators for M_2:

$$[u,w], [u,d_3], [w,d_1], [w,d_2], ud_1d_2u^{-1}d_2^{-1}d_1^{-1}, d_2^{-1}u^3d_1u^{-1}, ud_2u^5d_1^{-1},$$
$$wd_3^2wd_3^{-1}w^{-1}d_3^{-1}w^{-1}d_3^{-1}, d_3^{-1}w^2d_3^{-1}wd_3wd_3w, d_1d_2d_1u^{-1}d_2^2d_1^3u^{-1},$$
$$w^2d_3^{-4}w^2, d_2^{-1}d_1^{-1}u^{-1}d_1^{-1}u^{-1}d_1^{-2}u^{-1}d_1^{-1}u^{-1}d_1^{-2}, ud_2^{-1}d_1ud_1^{-1}u^{-1}d_2^{-1}ud_2u^{-1}d_1^2d_2^{-1}$$

Generators for M_3: d_1, d_2, d_3, u, v.

Relators for M_3:

$$[v,d_1],[u,d_3],vd_2d_3v^{-1}d_3^{-1}d_2^{-1},ud_2d_1u^{-1}d_1^{-1}d_2^{-1},d_3^{-1}v^3d_2v^{-1},$$
$$u^3d_1u^{-1}d_2^{-1},vd_3v^5d_2^{-1},d_2ud_1^{-1}ud_1^{-1}ud_2u,d_2d_1^2d_2d_1d_2u^{-1}d_1u^{-1}d_1,$$
$$d_2^2d_3v^{-1}d_3^2d_2^3v^{-1},v^{-1}u^{-1}v^{-1}d_3v^{-1}u^{-1}v^{-1}u^{-1}d_1^{-1}u^{-1},$$
$$d_3^{-1}d_2^{-1}v^{-1}d_2^{-1}v^{-1}d_2^{-2}v^{-1}d_2^{-1}v^{-1}d_2^{-2},ud_2^{-1}ud_2^{-1}d_1^{-1}d_2^{-2}ud_2^{-1}ud_2^{-2},$$
$$ud_1d_2^{-1}ud_2^2u^{-2}d_1^{-1}u^{-1}d_1d_2^{-2},v^{-2}u^{-1}v^{-1}u^{-1}v^{-2}u^2d_1ud_2^{-2}d_1$$

B.5. $n = 4, q = 2$

We give here presentations of the maximal parabolics on the generators d_1 till d_4 and p_1 till p_7. To each presentation the relators d_i^3 for $1 \leq i \leq 4$ and $[d_i, d_j]$ for $1 \leq i < j \leq 4$ need to be added.

Generators for M_1: $d_1, d_2, d_3, d_4, p_1, p_2, p_3, p_6, p_7$.

Relators for M_1:

$$p_1^2,p_2^2,p_3^3,p_6^2,p_7^2,[p_1,d_1],[p_1,d_2],[p_2,d_1],[p_2,d_4],[p_2,p_3],[p_2,p_7],[p_3,d_1],[p_3,p_6],$$
$$[p_6,d_1],[p_6,d_2],[p_7,d_1],[p_7,d_2],[p_7,d_3],[p_7,d_4],(p_7p_1)^4,p_2d_2^{-1}p_2d_3,p_6d_3p_6d_4^{-1},$$
$$p_3^{-1}p_2d_4p_3^{-1}d_2^{-1}d_3^{-1},p_6p_3^{-1}d_2p_3^{-1}d_4^{-1}d_3^{-1},p_1p_7p_1p_6d_4p_7p_6,d_4^{-1}p_1d_3p_1d_3^{-1}p_6p_1p_7,$$
$$d_4p_6p_1d_4p_6d_4^{-1}p_1d_3^{-1},d_4p_1p_7p_1d_4^{-1}p_1p_7p_1,p_1p_6d_4p_1d_3^{-1}d_4^{-1}p_1p_6d_4,$$
$$d_4p_3p_7p_3^{-1}p_7d_4^{-1}p_3p_7p_3^{-1}d_4^{-1}p_7,p_2p_1d_3^{-1}p_3^{-1}p_1p_7p_6p_7p_6p_3^{-1}d_3p_1d_3,$$
$$p_2p_6p_1p_2d_2^{-1}d_4p_1d_4p_3p_7d_2^{-1}p_3d_4p_6p_1,p_3p_7d_2^{-1}p_3d_3p_3^{-1}d_2p_7p_3d_2p_7p_6p_3^{-1}p_7d_3^{-1},$$
$$p_3p_7d_2^{-1}p_3p_1p_3^{-1}d_2p_7p_3^{-1}p_7d_4p_6p_1d_4^{-1}p_6p_7p_1d_3^{-1}$$

Generators for M_2: $d_1, d_2, d_3, d_4, p_1, p_4, p_6, p_7$.

Relators for M_2:

$$p_1^2,p_4^2,p_6^2,p_7^2,[p_1,d_1],[p_1,d_2],[p_1,p_4],[p_4,d_3],[p_4,d_4],[p_4,p_7],[p_6,d_2],[p_7,d_1],$$
$$[p_7,d_2],[p_7,d_3],[p_7,d_4],(p_1p_7)^4,p_4d_1^{-1}p_4d_2,p_6d_4^{-1}p_6d_3,p_1p_7p_1d_3p_6p_7p_6,$$
$$p_1p_6d_4^{-1}p_1p_7d_4^{-1}p_1d_3,d_4p_1p_6p_1d_3d_4^{-1}p_7p_1d_3^{-1}p_7p_6$$

Generators for M_3: $d_1, d_2, d_3, d_4, p_2, p_4, p_5, p_7$.

Relators for M_3:

$$p_2^2,p_4^2,p_5^3,p_7^2,[p_2,d_1],[p_2,d_4],[p_2,p_5],[p_2,p_7],[p_4,d_3],[p_4,d_4],[p_4,p_5],[p_4,p_7],$$
$$[p_5,d_4],[p_5,p_7],[p_7,d_1],[p_7,d_2],[p_7,d_3],[p_7,d_4],d_3p_2d_2^{-1}p_2,p_4d_1^{-1}p_4d_2,$$
$$p_4p_5d_3p_5d_1^{-1}d_2^{-1},p_5^{-1}d_1^{-1}p_5^{-1}d_2d_3p_2$$

Generators for M_4: $d_1, d_2, d_3, d_4, p_2, p_3, p_4, p_5, p_6$.

Relators for M_4:

$$p_2^2, p_4^2, p_6^2, [p_2, d_1], [p_2, d_4], [p_3, p_6], [p_4, d_3], [p_4, d_4], [p_4, p_5], [p_4, p_6], [p_6, d_1], [p_6, d_2],$$
$$(p_5p_3)^3, p_6d_3^{-1}p_6d_4, d_2p_4d_1^{-1}p_4, d_3p_2d_2^{-1}p_2, p_3p_5d_1p_5p_3d_4^{-1}, p_2p_5p_3p_2p_3^{-1}p_5^{-1},$$
$$(p_5p_3d_4d_1^{-1})^2, d_3p_6p_3d_3d_1^{-1}d_2^{-1}d_3^{-1}p_3d_1d_4, d_1^{-1}d_4^{-1}p_3p_5p_3^{-1}p_5^{-1}d_1^{-1}p_3^{-1}p_5^{-1}p_2,$$
$$p_4d_1p_5^{-1}d_2p_5p_3d_1d_4^{-1}p_3p_5, p_5p_3^2p_5d_1d_4d_2d_3p_3^{-1}, d_4d_1d_4^{-1}p_5p_3d_1d_4^{-1}d_1^{-1}p_3p_5,$$
$$p_5d_2d_3d_4p_5p_3d_4p_5p_3, d_2d_1p_3p_4p_3^{-1}d_4^{-1}p_5p_6p_5^{-1}d_2^{-1}d_1^{-1}d_4p_5p_3,$$
$$p_5^{-1}p_3p_4p_3^{-1}d_1^{-1}d_2^{-1}p_5p_3p_5^{-1}p_3^{-1}d_2d_1p_5p_6,$$
$$p_3p_4p_3^{-1}d_1^{-1}d_2^{-1}p_5p_3d_4p_5^{-1}p_3^{-1}d_1d_2p_5p_6p_5^{-1}d_1^{-1},$$
$$p_3^{-1}p_5^{-1}p_2d_2d_1p_3p_4p_3^{-1}d_2^{-1}p_2d_2p_5p_6p_5^{-1}d_2^{-1}d_1^{-1}p_3p_5,$$
$$p_3^{-1}p_5^{-1}d_2p_5p_6p_5^{-1}d_1^{-1}d_2^{-1}d_4p_3^{-1}p_5^{-1}d_2p_5p_6p_5^{-1}d_4d_2^{-1}d_1^{-1}$$

C. GAP code: Amalgams

C.1. sp63.gap

Main code file for $n = 3$; this can be used to verify simple connectedness of $\mathrm{Sp}(6, q)$.

```
# $Id: sp63.gap,v 1.58 2005/07/09 21:11:34 maxhorn Exp $

Read("mylib.gap");

# The parameters of our group
n := 3;
q := 3;

Read("common.gap");
Read("build_sp6_gens.gap");
Read("build_rels.gap");
Read("check_amalgam.gap");
```

C.2. sp82.gap

Main code file for $\mathrm{Sp}(8, 2)$. Since for $q = 2$ we need additional generators, this is separate from the code for $\mathrm{Sp}(6, q)$ since we need more subgroup generators, and we also find those generators in a different way.

```
# $Id: sp82.gap,v 1.42 2005/08/21 16:59:54 maxhorn Exp $

Read("mylib.gap");

# The parameters of our group
n := 4;
q := 2;

Read("common.gap");

##########################################
# Determining suitable generators.
##########################################
Print("Determining suitable generators...\n");
```

```
# We list a hard coded set of generators here.
p :=
[ [ [ 1,   0,   0,   0,   0,   0,   0,   0 ],
    [ 0,   1,   0,   0,   0,   0,   0,   0 ],
    [ 0,   0,   1,   0,   0,   0, z^1, z^2 ],
    [ 0,   0,   0,   1,   0,   0, z^2,   1 ],
    [ 0,   0,   0,   0,   1,   0,   0,   0 ],
    [ 0,   0,   0,   0,   0,   1,   0,   0 ],
    [ 0,   0, z^2, z^1,   0,   0,   1,   0 ],
    [ 0,   0, z^1,   1,   0,   0,   0,   1 ] ],
  [ [ 1, 0, 0, 0, 0, 0, 0, 0 ],
    [ 0, 0, 1, 0, 0, 0, 0, 0 ],
    [ 0, 1, 0, 0, 0, 0, 0, 0 ],
    [ 0, 0, 0, 1, 0, 0, 0, 0 ],
    [ 0, 0, 0, 0, 1, 0, 0, 0 ],
    [ 0, 0, 0, 0, 0, 0, 1, 0 ],
    [ 0, 0, 0, 0, 0, 1, 0, 0 ],
    [ 0, 0, 0, 0, 0, 0, 0, 1 ] ],
  [ [ 1, 0, 0, 0, 0, 0, 0, 0 ],
    [ 0, z^2, 1, 1, 0, 0, 0, 0 ],
    [ 0, 1, z^2, 1, 0, 0, 0, 0 ],
    [ 0, 1, 1, z^2, 0, 0, 0, 0 ],
    [ 0, 0, 0, 0, 1, 0, 0, 0 ],
    [ 0, 0, 0, 0, 0, z, 1, 1 ],
    [ 0, 0, 0, 0, 0, 1, z, 1 ],
    [ 0, 0, 0, 0, 0, 1, 1, z ] ],
  [ [ 0, 1, 0, 0, 0, 0, 0, 0 ],
    [ 1, 0, 0, 0, 0, 0, 0, 0 ],
    [ 0, 0, 1, 0, 0, 0, 0, 0 ],
    [ 0, 0, 0, 1, 0, 0, 0, 0 ],
    [ 0, 0, 0, 0, 0, 1, 0, 0 ],
    [ 0, 0, 0, 0, 1, 0, 0, 0 ],
    [ 0, 0, 0, 0, 0, 0, 1, 0 ],
    [ 0, 0, 0, 0, 0, 0, 0, 1 ] ],
  [ [ z, 1, 1, 0, 0, 0, 0, 0 ],
    [ 1, z, 1, 0, 0, 0, 0, 0 ],
    [ 1, 1, z, 0, 0, 0, 0, 0 ],
    [ 0, 0, 0, 1, 0, 0, 0, 0 ],
    [ 0, 0, 0, 0, z^2, 1, 1, 0 ],
    [ 0, 0, 0, 0, 1, z^2, 1, 0 ],
    [ 0, 0, 0, 0, 1, 1, z^2, 0 ],
    [ 0, 0, 0, 0, 0, 0, 0, 1 ] ],
  [ [ 1, 0, 0, 0, 0, 0, 0, 0 ],
    [ 0, 1, 0, 0, 0, 0, 0, 0 ],
    [ 0, 0, 0, 1, 0, 0, 0, 0 ],
    [ 0, 0, 1, 0, 0, 0, 0, 0 ],
    [ 0, 0, 0, 0, 1, 0, 0, 0 ],
    [ 0, 0, 0, 0, 0, 1, 0, 0 ],
    [ 0, 0, 0, 0, 0, 0, 0, 1 ],
```

```
    [ 0, 0, 0, 0, 0, 0, 1, 0 ] ],
  [ [ 1, 0, 0, 0, 0, 0, 0, 0 ],
    [ 0, 1, 0, 0, 0, 0, 0, 0 ],
    [ 0, 0, 1, 0, 0, 0, 0, 0 ],
    [ 0, 0, 0, 0, 0, 0, 0, 1 ],
    [ 0, 0, 0, 0, 1, 0, 0, 0 ],
    [ 0, 0, 0, 0, 0, 1, 0, 0 ],
    [ 0, 0, 0, 0, 0, 0, 1, 0 ],
    [ 0, 0, 0, 1, 0, 0, 0, 0 ] ] ] * z^0;

# Verify that the elements stabilize the subspaces as demanded.
Assert(0, OnSubspacesByCanonicalBasis(u_space[1],p[1]) = u_space[1]);
Assert(0, OnSubspacesByCanonicalBasis(u_space[2],p[1]) = u_space[2]);

Assert(0, OnSubspacesByCanonicalBasis(u_space[1],p[2]) = u_space[1]);
Assert(0, OnSubspacesByCanonicalBasis(u_space[3],p[2]) = u_space[3]);
Assert(0, OnSubspacesByCanonicalBasis(u_space[4],p[2]) = u_space[4]);

Assert(0, OnSubspacesByCanonicalBasis(u_space[1],p[3]) = u_space[1]);
Assert(0, OnSubspacesByCanonicalBasis(u_space[4],p[3]) = u_space[4]);

Assert(0, OnSubspacesByCanonicalBasis(u_space[2],p[4]) = u_space[2]);
Assert(0, OnSubspacesByCanonicalBasis(u_space[3],p[4]) = u_space[3]);
Assert(0, OnSubspacesByCanonicalBasis(u_space[4],p[4]) = u_space[4]);

Assert(0, OnSubspacesByCanonicalBasis(u_space[3],p[5]) = u_space[3]);
Assert(0, OnSubspacesByCanonicalBasis(u_space[4],p[5]) = u_space[4]);

Assert(0, OnSubspacesByCanonicalBasis(u_space[1],p[6]) = u_space[1]);
Assert(0, OnSubspacesByCanonicalBasis(u_space[2],p[6]) = u_space[2]);
Assert(0, OnSubspacesByCanonicalBasis(u_space[4],p[6]) = u_space[4]);

Assert(0, OnSubspacesByCanonicalBasis(u_space[1],p[7]) = u_space[1]);
Assert(0, OnSubspacesByCanonicalBasis(u_space[2],p[7]) = u_space[2]);
Assert(0, OnSubspacesByCanonicalBasis(u_space[3],p[7]) = u_space[3]);

# On request, verify that the elements, together with the flag stabilizer,
# really generate the maximal stabilizers.
Assert(0, ClosureGroup(flag_stab, [p[1],p[2],p[3]]) = u_stab[1]);
Assert(0, ClosureGroup(flag_stab, [p[1],p[4],p[6]]) = u_stab[2]);
Assert(0, ClosureGroup(flag_stab, [p[2],p[4],p[5],p[7]]) = u_stab[3]);
Assert(0, ClosureGroup(flag_stab, [p[3],p[6],p[5]]) = u_stab[4]);

###########################################
# Build and check the amalgam
###########################################
gens := Concatenation(flag_gens, p);;
Read("build_rels.gap");
Read("check_amalgam.gap");
```

C.3. mylib.gap

This file contains various useful functions used repeatedly by the other code.

```
# $Id: mylib.gap,v 1.30 2005/08/03 17:34:10 maxhorn Exp $

#
# Apply the Frobenius automorphismus to a matrix.
#
FrobeniusToMatrix := function ( M, F )
    local f, deg;
    Assert(0, IsMatrix(M));
    deg := DegreeOverPrimeField(F);
    f := FrobeniusAutomorphism(F) ^ (deg/2);
    return List(M, row -> OnTuples(row,f));
end;;

#
# Compute a flip-flop matrix as described in Section 1.2.
#
FlipFlobBilinearForm := function ( n, F )
    local I;
    I := IdentityMat(n/2, F);
    return BlockMatrix([[1,2,I],[2,1,-I]],2,2);
end;;

#
# Compute the value of the first form used in the paper: (u,v).
# This is a direct implementation using a for loop; it's
# simply much faster than the 'naive' implementation using
# matrix/vector multiplications.
#
ProdA := function(u, v)
    local n, i, sum;
    n := Length(u)/2;
    sum := 0;
    for i in [1..n] do
        sum := sum + u[i] * v[i+n];
        sum := sum - u[i+n] * v[i];
    od;
    return sum;
    #return Sum([1..n], i -> u[i] * v[i+n]) - Sum([1..n], i -> u[i+n] * v[i]);
    #return u*B*v;
end;;

#
# Compute the value of the second form used in the paper: ((u,v)).
# This is a direct implementation using a for loop; it's
# simply much faster than the 'naive' implementation using
```

```
# matrix/vector multiplications.
#
ProdB := function(u, v)
    local n, i, sum;
    n := Length(u);
    sum := 0;
    for i in [1..n] do
        sum := sum + u[i] * v[i]^q;
    od;
    return sum;
    #return Sum([1..Length(u)], i -> u[i] * v[i]^q);
    #return u*List(v, x -> x^q);
end;;

#
# Auxillary function, used by ExtendMatrixGroup and in build_sp6_gens.gap.
# It takes a (2n)x(2n) matrix M, and 'enlarges it' by inserting two
# additional rows/columns. The matrix
#   / A B \
#   \ C D /
# for off = 0 becomes
#   / A   B   \
#   |   1     |
#   | C   D   |
#   \       1 /
# and for off = 1 becomes
#   / 1       \
#   |   A   B |
#   |     1   |
#   \   C   D /
#
ExtendMatrix := function ( M, off )
    local ExtM, n, k, i, j, xoff, yoff;
    n := Length(M);
    ExtM := IdentityMat(n+2, DefaultFieldOfMatrix(M));

    k := (n - (n mod 2))/2;

    xoff := off;
    for i in [1..n] do
        yoff := off;
        for j in [1..n] do
            ExtM[i+xoff][j+yoff] := M[i][j];
            if j = k or (j = 2*k and j+yoff < n+1)  then yoff := yoff + 1; fi;
        od;
        if i = k or (i = 2*k and i+xoff < n+1) then xoff := xoff + 1; fi;
    od;
    return ExtM;
end;;
```

```
#
# Extend a (2n)x(2n) matrix group to a new (2n+2)x(2n+2) matrix group,
# by extending all its generators (via ExtendMatrix).
#
ExtendMatrixGroup := function ( G )
    local gens;
    Assert(0, IsMatrixGroup(G));

    gens := GeneratorsOfGroup(G);
    gens := ShallowCopy(Concatenation(
            List(gens, g -> ExtendMatrix(g,0)),
            List(gens, g -> ExtendMatrix(g,1))));
    Sort(gens);
    return GroupWithGenerators(Unique(gens));
end;;

#
# We can efficiently compute a (lower bound on) the size of a matrix group G
# by finding a vector v with an orbit on which G acts (faithfully). The
# following method does just that. The caller is responsible for supplying
# a suitable vector v (the shorter the orbit, the faster the computations).
#
SizeViaOrbit := function (G, v)
    local orbit, phi, size;

    orbit := Orbit(G, v);;
    orbit := ShallowCopy(orbit);; Sort(orbit);
    phi := ActionHomomorphism(G, orbit);
    size := Size(Image(phi));
    return size;
end;;

#
# Find a finite presentation of the given group, by first converting it to a
# perumtation presentation on the orbit of the given vector v_orb.
#
GroupRelatorsViaOrbit := function( G, v_orb, FG, fg_subset, gens )
    local orb, phi, H, psi, Fgens, Frels;

    Print("  Computing orbit...\n");
    orb := Orbit( G, v_orb );;
    orb := ShallowCopy(orb);;
    Sort(orb);
    #
    Print("  Computing permutation group ...\n");
    phi := ActionHomomorphism( G, orb, "surjective" );;
    H := Image( phi );;
    #
```

```
    Print("  Verifying that perm group is isomorphic image of the stabilizer...\n");
    Assert(0, IsSurjective(phi) and Size(H) = Size(G));
    #
    Print("  Determining finite presentation...\n");
    psi := IsomorphismFpGroupByGenerators( H, GeneratorsOfGroup(H) );;
    #
    Fgens := FreeGeneratorsOfFpGroup(Image(psi));
    Frels := RelatorsOfFpGroup(Image(psi));
    Assert(0, ForAll( Frels, r->MappedWord( r, Fgens, gens{fg_subset} ) = gens[1]^0 ));
    return List( Frels, r->MappedWord( r, Fgens, GeneratorsOfGroup(FG){fg_subset} ) );
end;;

#
# Check whether the given group is a (subgroup of) Sp(n,q), i.e. whether
# its generators fulfill all required properties.
#
AssertIsSPqSubgroup := function ( G )
    local F, B, n, gens;
    Assert(0, IsMatrixGroup(G));
    F := DefaultFieldOfMatrixGroup(G);
    n := DimensionOfMatrixGroup(G);
    B := FlipFlobBilinearForm(n, F);
    gens := GeneratorsOfGroup(G);
    Assert(0, ForAll(gens, g -> g*B*TransposedMat(g) = B));
    Assert(0, ForAll(gens, g -> FrobeniusToMatrix(g, F) = TransposedMat(g)^-1));
end;;

#
# Check whether the given group is of type Sp(n,q).
#
AssertIsSPq := function ( G )
    local F, z, n, v, q;

    # Verify that our 'candidate' is a subgroup of a Sp(n,q).
    AssertIsSPqSubgroup(G);

    F := DefaultFieldOfMatrixGroup(G);
    z := PrimitiveRoot(F);
    n := DimensionOfMatrixGroup(G);
    q := Characteristic(F)^(DegreeOverPrimeField(F) / 2);

    # Now compute a lower bound for its size. If it matches the
    # size of Sp(2*k,q), they are isomorphic.
    v := List([1..n], x -> 0*z^0);
    v[n/2] := z^0;
    v[n] := First(F, g -> g^(q+1) = -1*z^0);
    Assert(0, Size(Sp(n,q)) = SizeViaOrbit(G, v));
end;;
```

```
#
# Determine GU(n,q), but with the correct sesquilinear form.
#
MyGU := function (n, q)
    local F, G, A, I, T, z, i, a, b, m;

    F := GF(q^2);
    z := PrimitiveRoot(F);
    G := GU(n,q);

    # The form is irrelevant if n = 1.
    if n = 1 then
        return G;
    fi;

    A := InvariantSesquilinearForm(G).matrix;
    I := IdentityMat(n, F);

    # The sesqulinear form A which GAP uses by default for GU is not, as
    # desired by us, the identity matrix. Hence we have to conjugate GU
    # before using it. To this end we compute a matrix T for which
    #       TransposedMat(T) * A * FrobeniusToMatrix(T)) = I
    # We then conjugate GU by that matrix.

    # HACK: We (ab)use the fact that we know the precise structure of A.
    # In theory, this could change in new versions of GAP, so we add a
    # quick check.
    Assert(0, A = Reversed(I));

    # We now compute T, starting from a null matrix and filling in entries.
    T := NullMat(n, n, F);

    # Find a such that a^q + a = 1.
    a := First(F, g -> g^q+g=z^0);

    # Find b such that b^q * b = 1.
    b := First(F, g -> g^(q+1) = -1*z^0);

    # Compute n/2, rounded up.
    m := (n + (n mod 2)) / 2;

    for i in [1..m] do
        T[i][i] := a;
        T[n-i+1][i] := z^0;
    od;
    for i in [(m+1)..n] do
        T[n-i+1][i] := a^q * b;
        T[i][i] := -z^0 * b;
    od;
```

```
    # Verify the transformation matrix T works as expected.
    Assert(0, TransposedMat(T) * A * FrobeniusToMatrix(T, F) = I);
    G := G^T;
    Assert(0, ForAll(GeneratorsOfGroup(G), g -> TransposedMat(g) * FrobeniusToMatrix(g, F) = I));
    return G;
end;;

#
# Compute our prefered matrix presentation of Sp(n,q).
#
MySp := function (n, q)
    local F, z, d, spq, g, S, k;

    Assert(0, IsEvenInt(n));

    F := GF(q^2);
    z := PrimitiveRoot(F);

    if n = 0 then
        return TrivialGroup();
    fi;

    spq := [];
    d := First(F, g -> g^(q+1)+z^(q+1)=z^0);

    spq[2] := GroupWithGenerators(
        [ [ [   z^(q-1), 0*z ],
            [ 0*z, z^(1-q) ] ],
          [ [  d^q, z ],
            [ -z^q, d ] ] ]);

    if n = 2 then
        return spq[2];
    fi;

    spq[4] := ExtendMatrixGroup(spq[2]);
    for g in GeneratorsOfGroup(spq[2]) do
        S := BlockMatrix([[1,1,g],[2,2,FrobeniusToMatrix(g, F)]],2,2);
        spq[4] := ClosureGroupAddElm(spq[4], MatrixByBlockMatrix(S));
    od;
    if q = 2 then
        spq[4] := ClosureGroupAddElm(spq[4],
                    [[1,1,0,1],
                     [1,1,1,0],
                     [0,1,1,1],
                     [1,0,1,1]] * z^0);
    fi;
    SetSize(spq[4], Size(Sp(4,q)));
```

```
    if n = 4 then
        return spq[4];
    fi;

    for k in [3..n/2] do
        spq[2*k] := ExtendMatrixGroup(spq[2*k-2]);
        SetSize(spq[2*k], Size(Sp(2*k,q)));
    od;

    return spq[n];
end;;

#
# This function determines generators for the maximal stabilizer S_d
# in Sp(2*n,q), which is of the form Sp(2n-2d,q) x GU(d, q^2).
#
SubspaceStabGens := function (n, q, d)
    local g, G, h, F, z, i, gens;

    F := GF(q^2);
    z := PrimitiveRoot(F);
    gens := [];

    # Determine generators for the symplectic part.
    G := MySp(2*n-2*d, q);
    for g in GeneratorsOfGroup(G) do
        for i in [1..d] do
            g := ExtendMatrix(g, 1);
        od;
        Add(gens, g);
    od;

    # Determine generators for the unitary part.
    G := MyGU(d, q);
    for g in GeneratorsOfGroup(G) do
        h := BlockMatrix([[1,1,g],[2,2,FrobeniusToMatrix(g,F)]],2,2) * z^0;
        for i in [1..n-d] do
            h := ExtendMatrix(h, 0);
        od;
        Add(gens, MatrixByBlockMatrix(h));
    od;

    return gens;
end;;
```

C.4. common.gap

```
# $Id: common.gap,v 1.39 2005/07/25 18:56:16 maxhorn Exp $

# Set VERIFY_CORRECTNESS to false to turn off some of the most
# expensive  correctness checks in the source. All cheap checks
# will stay on. Turning it off can improve perfomance coniderably.
VERIFY_CORRECTNESS := true;

F := GF(q*q);
z := PrimitiveRoot(F);

##########################################
#
# Compute the elemnts of our maximal flag, as
# well as the sizes of their stabilizers.
#
u_space := [];
u_size := [];
for i in [1..n] do
    u_space[i] := IdentityMat(2*n, F){[1..i]};
    u_size[i] := Size(GU(i, q));
    if n > i then
        u_size[i] := u_size[i] * Size(Sp(2*(n-i), q));
    fi;
od;

##########################################
#
# Compute the flag stabilizer (diagonal matrices).
#
flag_gens := [];
for i in [1..n] do
    tmp := List( [1..2*n], j -> z^0 );
    tmp[i] := z^(q-1);
    tmp[i+n] := z^(1-q);
    Add(flag_gens, DiagonalMat( tmp ));
od;
flag_stab := GroupWithGenerators(flag_gens);
Assert(0, Size(flag_stab) = (q+1)^n);
AssertIsSPqSubgroup(flag_stab);
for i in [1..n] do
    Assert(0, ForAll(flag_gens, g -> OnSubspacesByCanonicalBasis(u_space[i],g) = u_space[i]));
od;

##########################################
#
# Find a value lambda such that lambda^(q+1) = -1 in F_{q^2}.
# This is later used to define vectors with relatively short
```

```
# orbit, typically of the form (1,0,0,v,0,0) and (1,0,1,v,0,v).
#
lambda := First(F, g -> g^(q+1) = -1*z^0);

##########################################
#
# If requested, verify that MySp(n,q) returns
# the correct group.
#
if VERIFY_CORRECTNESS then
    for k in [1..n] do
        Print("Verifying Sp(", 2*k, ",q)...\n");
        AssertIsSPq(MySp(2*k,q));
    od;
fi;

##########################################
#
# Determine the stabilizers.
#
v_orb := List([1..2*n], x -> 0*z^0);
v_orb[1] := z^0;
v_orb[n] := z^0;
v_orb[n+1] := lambda;
v_orb[2*n] := lambda;

u_stab := [];
for i in [1..n] do
    Print("  -> determining stabilizer ", i, "\n");
    gens := SubspaceStabGens(n, q, i);
    Assert(0, ForAll(gens, g -> OnSubspacesByCanonicalBasis(u_space[i],g) = u_space[i]));
    u_stab[i] := GroupWithGenerators(gens);

    if VERIFY_CORRECTNESS then
        Assert(0, SizeViaOrbit(u_stab[i], v_orb) = u_size[i]);
        AssertIsSPqSubgroup(u_stab[i]);
    fi;
    SetSize(u_stab[i], u_size[i]);
od;
```

C.5. build_sp6_gens.gap

```
# $Id: build_sp6_gens.gap,v 1.7 2005/08/21 16:59:54 maxhorn Exp $

##########################################
#
# Determine elements u, v, w, which stabilize the
```

```
# subspaces as follows:
#  u - u_space[2], u_space[3]
#  v - u_space[1], u_space[3]
#  w - u_space[1], u_space[2]
#
# Moreover, we want that that they generate, together
# the generators of the flag stabilizer, the maximal
# stabilizers as follows:
#
#  u, v -> stabilizer of u_space[3]
#  u, w -> stabilizer of u_space[2]
#  v, w -> stabilizer of u_space[1]
#
##########################################
Print("Determining suitable generators...\n");

# Hack: We rely on our knowledge about the generators of MySp(2,q).
# If that ever changes, we have to compute g differently.
g := MySp(2,q).2;
u := ExtendMatrix(BlockMatrix([[1,1,g],[2,2,FrobeniusToMatrix(g, F)]],2,2),0);
v := ExtendMatrix(BlockMatrix([[1,1,g],[2,2,FrobeniusToMatrix(g, F)]],2,2),1);
w := ExtendMatrix(ExtendMatrix(g,1),1);

# Verify that u,v,w stabilize the subspaces as demanded.
Assert(0, OnSubspacesByCanonicalBasis(u_space[2],u) = u_space[2]);
Assert(0, OnSubspacesByCanonicalBasis(u_space[3],u) = u_space[3]);
Assert(0, OnSubspacesByCanonicalBasis(u_space[1],v) = u_space[1]);
Assert(0, OnSubspacesByCanonicalBasis(u_space[3],v) = u_space[3]);
Assert(0, OnSubspacesByCanonicalBasis(u_space[1],w) = u_space[1]);
Assert(0, OnSubspacesByCanonicalBasis(u_space[2],w) = u_space[2]);

# On request, verify that u,v,w, together with the flag stabilizer,
# really generate the maximal stabilizers.
if VERIFY_CORRECTNESS then
    tmp := ClosureGroup(flag_stab, [u,w]);
    Assert(0, u_size[2] = SizeViaOrbit(tmp, v_orb));
    AssertIsSPqSubgroup(tmp);

    tmp := ClosureGroup(flag_stab, [u,v]);
    Assert(0, u_size[3] = SizeViaOrbit(tmp, v_orb));
    AssertIsSPqSubgroup(tmp);

    tmp := ClosureGroup(flag_stab, [v,w]);
    Assert(0, u_size[1] = SizeViaOrbit(tmp, v_orb));
    AssertIsSPqSubgroup(tmp);
fi;

gens := Concatenation(flag_gens, [u,v,w]);;
```

C.6. build_rels.gap

```
# $Id: build_rels.gap,v 1.18 2005/08/21 16:59:54 maxhorn Exp $

Print("Constructing the amalgam...\n");

SetInfoLevel(InfoFpGroup, 3);

FG := FreeGroup(Length(gens));

# Select the generators for each stabilizer
stab_gens := [];
for i in [1..Size(u_space)] do
    stab_gens[i] := [];
    for j in [1..Size(gens)] do
        if OnSubspacesByCanonicalBasis(u_space[i],gens[j] ) = u_space[i] then
            Add(stab_gens[i], j);
        fi;
    od;
od;

u_rels := [];
rels := [];
stabs := List( stab_gens, t->Group( gens{t} ) );

# We now determine finite presentations for each stabilizer.
# To do this, we first find a permutation group isomorphic
# to that stabilizer, then use IsomorphismFpGroupByGenerators
# to get the finite presentation.
#
# Finally, we form the union of the relators of the stabilizers.
#
for i in [1..Length(stab_gens)] do
    # Verify and set the size of the group.
    # We assume here that v_orb has been set to a vector with short orbit!
    Assert(0, SizeViaOrbit(stabs[i], v_orb) = u_size[i]);
    SetSize(stabs[i], u_size[i]);
    #
    Print("Generating relators for ", stab_gens[i], "\n");
    u_rels[i] := GroupRelatorsViaOrbit(stabs[i], v_orb, FG, stab_gens[i], gens);
    Append(rels, u_rels[i]);
od;

rels := Set(rels);

Assert(0, ForAll( rels, r->MappedWord( r, GeneratorsOfGroup(FG), gens ) = gens[1]^0 ));
```

C.7. check_amalgam.gap

```
# $Id: check_amalgam.gap,v 1.4 2005/07/25 18:56:16 maxhorn Exp $

#
# The universal completion of the amalgam:
#
A := FG / rels;

#
# The point stabilizer insider the universal completion:
#
U := Subgroup(A, GeneratorsOfGroup(A){stab_gens[1]});

#
# We now determine the index of U in A, using ACE.
#
# To this end, we estimate the required amount of memory, as follows:
#
# ACE needs at least 2*n words of memory for every coset class, where n is
# the number of generators (one word for each generator, another for its
# inverse).
#
# We already know the number of expected coset classes; but of course, ACE
# usually will generate more intermediate coset classes than there
# actually are, so we have to increase this value by a certain "safety
# margin" to arrive at the final estimate.
#

# The expected index = expected number of coset classes.
target_index := Size(Sp(2*n,q)) / u_size[1];
expected_mem := target_index * (2*Size(GeneratorsOfGroup(FG))+1);

# Load ACE.
LoadPackage( "ace" );
SetInfoACELevel( 2 );
TCENUM := ACETCENUM;

# Perform the enumeration. We empirically determined that the
# felsch-0 strategy works best for our purposes.
real_index := Index(A, U : max := 0, workspace := expected_mem, messages := 10000, felsch:=0);

# Check whether the computed index matches our expectations or not.
Print("Sp(",2*n,",",q,")");
if real_index = target_index then
    Print(" is universal completion of our amalgam!\n");
else
    Print(" is NOT the universal completion of our amalgam!\n");
fi;
```

D. GAP code: combinatorics with triangles

The following code was used to check all interesting triangle conjugacy classes, as described in the proof of Lemma 5.3.3. It uses `mylib.gap`, see Section C.3.

```
# $Id: fast_tri2.gap,v 1.12 2005/08/04 17:06:31 maxhorn Exp $

Read("mylib.gap");

# Specify the prime power q, we'll consider the groups Sp(6, q^2).
# Works for any prime power q>=4.
q := 11;

#
# Various useful bits and pieces
#
F := GF(q*q);
z := PrimitiveRoot(F);

NormB := function(u)
    return ProdB(u,u);
end;;

# The set of all values k which fullfil k * k^q = -1
k_set := Set(Filtered(F, g -> g^(q+1) = -1*z^0));

# The set of units in F_{q^2}
FStar := Filtered(F, g -> g <> 0*z^0);

# Computer the outer product of 'space' with itself using the two-argument
# function 'func'. We use this to compute the gram matrices of a given
# vector span.
Outer := function (space, func)
    local mat, n, x, y;
    n := Length(space);
    mat := NullMat(n, n);
    for x in [1..n] do
        for y in [1..n] do
            mat[x][y] := func(space[x], space[y]);
        od;
    od;
    return mat;
end;;
```

```
# Determine whether the vectors in 'space' span a geometric subspace.
# Condition for that is that the subspace is non-degenerate w.r.t. the
# first form (ProdA) and totally degenerate w.r.t. the second form (ProdB).
IsGeometric := function (space)
    local n;
    n := Length(space);
    if n <> RankMat(space) then
        return false;
    fi;
    if (n > 1) and (Outer(space, ProdA) <> NullMat(n, n, F)) then
        return false;
    fi;
    return 0*z^0 <> Determinant(Outer(space, ProdB));
end;;


# We consider triangles of this form:
# p1 = <e1>, p2 = <e2>, p3 = <x*e1 + y*e2 + k*e3 + f3>
# where k^(q+1) = -1, x and y are non-zero, and x^(q+1) + y^(q+1) <> 0.
# Thanks to Lemma 3.3.2, we can limit ourselves to the case x=1.
#
# We  search for triples of points a,b,c with the property that each of them
# is geometric, the lines connecting them are geometric, and the triangle
# <a,b,c> is geometric as well.
#
# Whenever we find such a triple, we check whether the octahedron we get by
# combining <p1,p2,p3> and <a,b,c> in a certain way (see InducesGoodOctahedron
# and proposition 3.3.3) is decomposable.
#
# Whenever we find such a triangle, we have thus shown that the original
# bad triangle <p1,p2,p3> is null homotopic (and hence not that bad after all).
#
InducesGoodOctahedron := function (xBad, yBad, k, a, b, c)
    local p1, p2, p3;
    p1 := [1, 0, 0, 0, 0, 0] * z^0;
    p2 := [0, 1, 0, 0, 0, 0] * z^0;
    p3 := [xBad, yBad, k, 0, 0, 1] * z^0;
    return IsGeometric([a, b, c])
        and IsGeometric([a])
        and IsGeometric([b])
        and IsGeometric([c])
        and IsGeometric([a, b])
        and IsGeometric([a, c])
        and IsGeometric([b, c])
        #
        and IsGeometric([a, p1])
        and IsGeometric([a, p2])
        and IsGeometric([b, p2])
```

```
        and IsGeometric([b, p3])
        and IsGeometric([c, p1])
        and IsGeometric([c, p3])
        #
        and IsGeometric([a, b, p2])
        and IsGeometric([a, p1, p2])
        and IsGeometric([b, p2, p3])
        and IsGeometric([a, c, p1])
        and IsGeometric([b, c, p3])
        and IsGeometric([c, p1, p3]);
end;;


#
# Iterate over certain special points <a>, <b>, <c>; we check whether
# they form a triangle; if they do, we then check whether the cylinder
# formed by it, and thus our starting triangle, is null homotopic.
# If so, we return true.
#
IsDecomposable := function (xBad, yBad, k)
    local a, b, c, b1, c2, nA, nB, nAB, p1, p2, p3;
    a := [0, 0, 0, 0, 0, 1] * z^0;
    b := [1, 0, 0, k, 0, -xBad] * z^0;
    c := [0, 1, 0, 0, k, -yBad] * z^0;
    p1 := [1, 0, 0, 0, 0, 0] * z^0;
    p2 := [0, 1, 0, 0, 0, 0] * z^0;
    p3 := [xBad, yBad, k, 0, 0, 1] * z^0;
    nA := NormB(a);
    for b1 in FStar do
        b[1] := b1;
        nB := NormB(b);
        nAB := ProdB(a,b);
        # Perform a precheck to see whether this value of b will lead
        # to a suitable octahedron. If we encounter already here any
        # non-geometric components, we can short circuit the search
        # and immediately proceed to the next possible value for b.
        if nB <> 0*z^0 and nA*nB <> nAB^(q+1) and IsGeometric([b, p2])
          and IsGeometric([b, p3]) and IsGeometric([b, p2, p3])
          and IsGeometric([a, b, p2]) then
            for c2 in FStar do
                c[2] := c2;
                if InducesGoodOctahedron(xBad, yBad, k, a, b, c) then
                    return true;
                fi;
            od;
        fi;
    od;
    return false;
end;;
```

```
# Finally, we iterate over all normalized (i.e. x = 1) bad triangles
# and invoke IsDecomposable for each of them, to test whether
# the triangle we are looking at is null homotopic.
cGood:=0;
cBad:=0;
xBad := z^0;
for yBad in FStar do
    if xBad^(q+1) + yBad^(q+1) <> 0*z^0 then
        for k in k_set do
            if IsDecomposable(xBad, yBad, k) then
                cGood := cGood + 1;
            else
                cBad := cBad + 1;
            fi;
            if (cGood+cBad) mod 10 = 0 then
                Print("Bad: ", cBad, ", good: ", cGood, "\n");
            fi;
        od;
    fi;
od;
Print("There are ",cGood+cBad," conjugacy classes of bad triangles.\n");
Print("We were able to decompose ",cGood," triangles and failed for ",cBad,"\n");
```

Bibliography

[1] Michael Aschbacher. A characterization of Chevalley groups over fields of odd order, Parts I, II. *Ann. of Math.*, 106:353–468, 1977.

[2] Curtis D. Bennett, Ralf Gramlich, Corneliu Hoffman, and Sergey Shpectorov. Odd-dimensional orthogonal groups as amalgams of unitary groups. preprint.

[3] Curtis D. Bennett, Ralf Gramlich, Corneliu Hoffman, and Sergey Shpectorov. Curtis-Phan-Tits theory. In Alexander A. Ivanov, Martin W. Liebeck, and Jan Saxl, editors, *Groups, Combinatorics and Geometry: Durham 2001*, pages 13–29, New Jersey, 2003. World Scientific.

[4] Curtis D. Bennett and Sergey Shpectorov. A new proof for Phan's theorem. *J. Group Theory*, 7:287–310, 2004.

[5] Nicolas Bourbaki. *Elements of Mathematics, Algebra II, Chapters 4–7.* Springer, Berlin, 1990.

[6] Francis Buekenhout and Arjeh M. Cohen. *Diagram geometry.* Springer, Berlin, in preparation, preprints of the Chapters 1 – 19 can be found at `http://www.win.tue.nl/∼amc/buek`.

[7] John J. Cannon. Construction of defining relators for finite groups. *Discrete Math.*, 5:105–129, 1973.

[8] Jon R. Dunlap. *Uniqueness of Curtis-Phan-Tits amalgams.* PhD thesis, Bowling Green State University, 2005.

[9] Greg Gamble, George Havas, Alexander Hulpke, and Colin Ramsay. *ACE: a GAP 4 interface to the Advanced Coset Enumerator of G. Havas and C. Ramsay.*, 2000. Available from: `http://www.math.rwth-aachen.de/~Greg.Gamble/ACE/`.

[10] The GAP Group. *GAP – Groups, Algorithms, and Programming, Version 4.4*, 2005. Available from: `http://www.gap-system.org`.

[11] Ralf Gramlich. *On Graphs, geometries, and groups of Lie type.* PhD thesis, TU Eindhoven, 2002. Available from: `http://www.mathematik.tu-darmstadt.de/~gramlich/docs/thesis.ps`.

[12] Ralf Gramlich. *Phan Theory.* Habilitationsschrift, TU Darmstadt, 2004. Available from: `http://www.mathematik.tu-darmstadt.de/~gramlich/docs/habil.ps`.

[13] Ralf Gramlich. Weak Phan systems of type C_n. *J. Algebra*, 280:1–19, 2004.

[14] Ralf Gramlich, Corneliu Hoffman, Werner Nickel, and Sergey Shpectorov. Even-dimensional orthogonal groups as amalgams of unitary groups. *J. Algebra*, 284:141–173, 2005.

[15] Ralf Gramlich, Corneliu Hoffman, and Sergey Shpectorov. A Phan-type theorem for $Sp(2n, q)$. *J. Algebra*, 264:358–384, 2003.

[16] Alexander A. Ivanov and Sergey Shpectorov. *Geometry of sporadic groups II: representations and amalgams*, volume 91 of *Encyclopedia of Mathematics*. Cambridge University Press, 2002.

[17] Joachim Neubüser. An elementary introduction to coset table methods in computational group theory. In Colin M. Campbell and Edmund F. Robertson, editors, *Groups-St.Andrews 1981, Proceedings of a conference, St.Andrews 1981*, volume 71 of *London Math. Soc. Lecture Note Series*, pages 1–45. Cambridge University Press, 1982.

[18] Werner Nickel. Amalgams for two special orthogonal groups. preprint.

[19] Antonio Pasini. *Diagram Geometries*, volume 1125 of *Oxford Science Publications*. Oxford University Press, Oxford, 1994.

[20] Kok-wee Phan. On groups generated by three-dimensional special unitary groups, I. *J. Austral. Math. Soc. Ser. A*, 23:67–77, 1977.

[21] Kok-wee Phan. On groups generated by three-dimensional special unitary groups, II. *J. Austral. Math. Soc. Ser. A*, 23:129–146, 1977.

[22] Herbert Seifert and William Threlfall. *Lehrbuch der Topologie*. Chelsea Publishing Company, New York, 1934.

[23] Jean Pierre Serre. *Arbres, amalgames,* SL_2, volume 46 of *Astérisque*. Soc. Math. France, Paris, 1977.

[24] Jean Pierre Serre. *Trees*. Springer, Berlin, corrected second printing edition, 2003.

[25] Donald E. Taylor. *The geometry of the classical groups*, volume 9 of *Sigma Series in Pure Mathematics*. Heldermann Verlag, Berlin, 1992.

[26] Jacques Tits. Ordonnés, immeubles et sommes amalgamées. *Bull. Soc. Math. Belg. Sér. A*, 38:367–387, 1986.