Constructive recognition of classical groups in odd characteristic

C.R. Leedham-Green and E.A. O'Brien

Abstract

Let $G = \langle X \rangle \leq \mathrm{GL}(d, F)$ be one of $\mathrm{SL}(d, F)$, $\mathrm{Sp}(d, F)$, or $\mathrm{SU}(d, F)$ where F is a finite field of odd characteristic. We present algorithms to construct standard generators for G which allow us to write an element of G as a straight-line program in X. The algorithms run in Las Vegas polynomial-time, subject to the existence of a discrete log oracle for F.

1 Introduction

The major goal of the "matrix recognition project" is the development of efficient algorithms for the investigation of subgroups of $\mathrm{GL}(d, F)$ where F is a finite field. We refer to the recent survey [28] for background related to this work. A particular aim is to identify the composition factors of $G \leq \mathrm{GL}(d, F)$. If a problem can be solved for the composition factors, then it can be frequently be solved for G.

One may intuitively think of a *straight-line program* (SLP) for $g \in G = \langle X \rangle$ as an efficiently stored group word on X that evaluates to g. For a formal definition, we refer the reader to Section 13. A critical property of an SLP is that its length is proportional to the number of multiplications and exponentiations used in constructing the corresponding group element. Babai & Szemerédi [2] prove that every element of G has an SLP in X of length at most $O(\log^2 |G|)$.

Informally, a *constructive recognition algorithm* constructs an explicit isomorphism between a group G and a "standard" (or natural) representation of G, and exploits this isomorphism to write an arbitrary element of G as an SLP in its defining generators. For a more formal definition, see [32, p. 192].

We consider subgroups of $\mathrm{GL}(d, q)$ which preserve a certain non-degenerate bilinear or sesquilinear form on V, the vector space of d-dimensional row vectors over $\mathrm{GF}(q)$ on which $\mathrm{GL}(d, q)$ acts naturally. Let $\mathrm{Sp}(d, q)$, for even d, denote the subgroup of

This work was supported in part by the Marsden Fund of New Zealand via grant UOA 412. We thank John Bray and Robert Wilson for helpful discussions on its content. 2000 *Mathematics Subject Classification*. Primary 20C20, 20C40.

$\mathrm{GL}(d,q)$ consisting of all linear transformations of V which preserve a non-degenerate alternating bilinear form. Let $\mathrm{U}(d,q)$ denote the subgroup of $\mathrm{GL}(d,q^2)$ consisting of all linear transformations of the underlying vector space of degree d over $\mathrm{GF}(q^2)$ which preserve a non-degenerate hermitian form. We use the notation $\mathrm{GX}(d,q)$ to denote $\mathrm{GL}(d,q)$, or $\mathrm{Sp}(d,q)$, or $\mathrm{U}(d,q)$ and $\mathrm{SX}(d,q)$ denotes the corresponding subgroup of matrices of determinant 1.

We present and analyse two algorithms that take as input a generating subset X of $\mathrm{SX}(d,q)$ for q odd, and return as output *standard generators* of this group as SLPs in X. Usually, these generators are defined with respect to a basis different to that for which X was defined, and a change-of-basis matrix is also returned to relate these bases.

Similar algorithms are under development for the orthogonal groups in odd characteristic. Further, characteristic 2 can also be addressed in the same style, but the resulting algorithms are more complex. We shall consider these cases in later papers.

Our principal result is the following.

Theorem 1.1 *Let $G = \langle X \rangle \leq \mathrm{GL}(d,q)$ denote $\mathrm{SL}(d,q)$, or $\mathrm{Sp}(d,q)$ for even d, or $\mathrm{SU}(d,q)$ where, in all cases, q is odd. There are Las Vegas algorithms which, given the input X, construct a new standard generating set S for G having the property that an SLP of length $O(d^2 \log q)$ can be found from S for any $g \in G$. Assuming the existence of a discrete log oracle for $\mathrm{GF}(q)$, the algorithm to construct S runs in $O(d(\xi + d^3 \log q + \chi))$ field operations, where ξ is the cost of constructing an independent (nearly) uniformly distributed random element of G, and χ is the cost of a call to a discrete log oracle for $\mathrm{GF}(q)$.*

We prove this theorem by exhibiting algorithms with the stated complexity. If we assume that a random element can be constructed in $O(d^3)$ field operations, then, for fixed q, an upper bound to the complexity of constructing the standard generators is $O(d^4)$.

Brooksbank's algorithms [7] for the natural representation of $\mathrm{Sp}(d,q)$, $\mathrm{SU}(d,q)$, and $\Omega^\epsilon(d,q)$ have complexity $O(d^5)$ for fixed q. More precisely, the complexity of his algorithm to construct standard generators for the classical group is

$$O(d^3 \log q(d + \log d \log^3 q) + \xi(d + \log\log q) + d^5 \log^2 q + \chi(\log q)).$$

The algorithm of Celler & Leedham-Green [11] for $\mathrm{SL}(d,q)$ has complexity $O(d^4 \cdot q)$.

Once we have constructed these standard generators for G, a generalised echelonisation algorithm can now be used to write a given element of G as an SLP in terms of these generators. We do not consider this task here, but refer the interested reader to the algorithm of [7, Section 5], which performs this task in $O(d^3 \log q + \log^2 q)$ field operations.

The two algorithms presented here reflect a tension between two competing tasks: the speed of construction of the standard generators, and minimising the length of the resulting SLPs for the standard generators in X. The first is designed for optimal

efficiency; the second to produce short SLPs. We consider this topic in more detail in Section 13.

We establish some notation. Let $g \in G \leq \mathrm{GL}(d,q)$, let $\bar{G}$ denote $G/G \cap Z$ where Z denotes the centre of $\mathrm{GL}(d,q)$, and let $\bar{g}$ denote the image of g in $\bar{G}$. The *projective centraliser* of $g \in G$ is the preimage in G of $C_{\bar{G}}(\bar{g})$. Further $g \in G$ is a *projective involution* if g^2 is scalar, but g is not.

A central component of both algorithms is the use of involution centralisers. In Section 2 we summarise the structure of involution centralisers for elements of classical groups in odd characteristic. In Section 3 we define standard generators for the classical groups. In Sections 4 and 5 the two algorithms are described. They rely on finding involutions whose eigenspaces have approximately the same dimension in the case of the first algorithm, and exactly the same dimension in the second. The cost of constructing such involutions is analysed in Sections 6 and Section 7. We frequently compute high powers of elements of linear groups; an algorithm for doing this efficiently is described in Section 8. In Section 9, we discuss how to construct both the derived group of a group containing a classical group and also the the factors of a direct product of two classical groups. The centraliser of an involution is constructed using an algorithm of Bray [5]; this is considered in Section 10. The base cases of the algorithms (when $d \leq 4$) are discussed in Section 11. The complexity of the algorithms and the length of the resulting SLPs for the standard generators are discussed in Section 12 and 13. Finally we report on our implementation of the algorithm, publicly available in Magma [6].

2 Centralisers of involutions in classical groups

We briefly review the structure of involution centralisers in (projective) classical groups defined over fields of odd characteristic. A detailed account can be found in [15, 4.5.1].

1. If u is an involution in $\mathrm{SL}(d,q)$, with eigenspaces E_+ and E_-, then the centraliser of u in $\mathrm{SL}(d,q)$ is $(\mathrm{GL}(E_+) \times \mathrm{GL}(E_-)) \cap \mathrm{SL}(d,q)$. The centraliser of the image of u in $\mathrm{PSL}(d,q)$ is the image of the centraliser of u in $\mathrm{SL}(d,q)$ if E_+ and E_- have different dimensions. If E_+ and E_- have the same dimension, then in $\mathrm{PSL}(d,q)$ these eigenspaces may be interchanged by the centraliser of the image of u, which is now the image of $(\mathrm{GL}(d/2,q) \wr C_2) \cap \mathrm{SL}(d,q)$ in $\mathrm{PSL}(d,q)$.

2. If u is an involution in $\mathrm{Sp}(2n,q)$, with eigenspaces E_+ and E_-, these spaces are mutually orthogonal, and the form restricted to either is non-singular. Thus the centraliser of u is $\mathrm{Sp}(E_+) \times \mathrm{Sp}(E_-)$. The centraliser of the image of u in $\mathrm{PSp}(2n,q)$ is the image of $\mathrm{Sp}(E_+) \times \mathrm{Sp}(E_-)$, except when the eigenspaces have the same dimension, when the centraliser again permutes the eigenspaces. An element of the projective centraliser permuting the eigenspaces sends (v,w) to $(w\theta, -v\theta)$, where θ is an isometry that normalises these spaces, so the image of $\mathrm{Sp}(E_+) \times \mathrm{Sp}(E_-)$ has index 2 in the projective centraliser.

3. If u is an involution in $\mathrm{SU}(d,q)$, the situation is similar. Again the eigenspaces of u are mutually orthogonal, and the form restricted to the eigenspaces is non-degenerate. The centraliser of u in $\mathrm{SU}(d,q)$ is $(\mathrm{U}(E_+) \times \mathrm{U}(E_-)) \cap \mathrm{SU}(d,q)$. The centraliser of the image of u in $\mathrm{PSU}(d,q)$ is the image of the centraliser of u in $\mathrm{SU}(d,q)$ except where the eigenspaces of u have the same dimension, when the centraliser is the image of $(\mathrm{U}(d/2,q) \wr C_2) \cap \mathrm{SU}(d,q)$ in $\mathrm{PSU}(d,q)$.

3 Standard generators for classical groups

We now describe *standard generators* for the perfect classical groups $\mathrm{SL}(d,q)$, $\mathrm{Sp}(d,q)$ and $\mathrm{SU}(d,q)$ where q is odd in all cases.

We use the notation $\mathrm{SX}(d,q)$ to denote any one of these groups, and $\mathrm{PX}(d,q)$ to denote the corresponding central quotient.

Let V be the natural module for a perfect classical group G of the above kind. The standard generators for G are defined with respect to a hyperbolic basis for V, which will be defined in terms of the given basis by a change-of-basis matrix. We define a *hyperbolic* basis for V as follows.

1. If $G = \mathrm{SL}(d,q)$ then any ordered basis, say $(e_1, \ldots, e_d)$, is hyperbolic.

2. If $G = \mathrm{Sp}(2n,q)$ then a hyperbolic basis for V is an ordered basis of the form $(e_1, f_1, \ldots, e_n, f_n)$, where, if the image of a pair of vectors (v,w) under the form is written as $v.w$, then $e_i.e_j = f_i.f_j = 0$ for all i,j (including the case $i=j$), and $e_i.f_j = 0$ for $i \neq j$, and $e_i.f_i = -f_i.e_i = 1$ for all i.

3. If $G = \mathrm{SU}(2n,q)$, then a hyperbolic basis is exactly as for $\mathrm{Sp}(2n,q)$ except that, the form being hermitian, the condition $e_i.f_i = -f_i.e_i = 1$ for all i is replaced by the condition $e_i.f_i = f_i.e_i = 1$ for all i.

4. If $G = \mathrm{SU}(2n+1,q)$, then a hyperbolic basis for V is an ordered basis of the form $(e_1, f_1, \ldots, e_n, f_n, w)$, where the above equations hold, and in addition $e_i.w = f_i.w = 0$ for all i, and $w.w = 1$.

For uniformity of exposition, we sometimes label the ordered basis for $\mathrm{SL}(2n,q)$ as $(e_1, f_1, \ldots, e_n, f_n)$ and that for $\mathrm{SL}(2n+1,q)$ as $(e_1, f_1, \ldots, e_n, f_n, w)$.

The standard generators for $\mathrm{SX}(d,q)$ are with respect to a hyperbolic basis for V; these are defined in Table 3, subject to the following conventions.

1. For the unitary groups, ω is a primitive element for $\mathrm{GF}(q^2)$ and $\alpha = \omega^{(q+1)/2}$. In all other cases ω is a primitive element for $\mathrm{GF}(q)$.

2. In all but one case, we describe v as a permutation matrix acting on the hyperbolic basis for V.

3. For $\mathrm{SU}(2n+1,q)$, the matrices x and y normalise the subspace U having ordered basis $B = (e_1, w, f_1)$ and centralise $\langle e_2, f_2, \ldots, e_n, f_n\rangle$. We list their action on U with respect to basis B.

4. All other generators normalise either a 2-dimensional or 4-dimensional subspace U having ordered basis B which is (e_1, f_1) or (e_1, f_1, e_2, f_2) and centralise the space spanned by the remaining basis vectors. We list the action of a generator on U with respect to basis B.

5. To facilitate uniform exposition, we introduce trivial generators. Observe that vx is a $2n$-cycle for $\mathrm{SL}(2n,q)$. If the dimension required to define a generator is too large, then the generator is assumed to be trivial.

It is of course a triviality to *write down* the standard generators for $G = \mathrm{SX}(d,q)$. However we must construct these elements as SLPs in the given generators of G.

Once a hyperbolic basis has been chosen for V, the Weyl group of G can be defined as a section of G, namely as the group of monomial matrices in G modulo diagonal matrices, thus defining a subgroup of the symmetric group S_d. For $G = \mathrm{SL}(d,q)$, this group is S_d. For $\mathrm{Sp}(2n,q)$ the Weyl group is the subgroup of S_{2n} that preserves the system of imprimitivity with blocks $\{e_i, f_i\}$ for $1 \le i \le n$, and is thus $C_2 \wr S_n$. For each of $\mathrm{SU}(2n,q)$ and $\mathrm{SU}(2n+1,q)$, the Weyl group is also $C_2 \wr S_n$.

Lemma 3.1 *Let $G = \mathrm{SU}(d,q)$ for $d \ge 2$. Then $G = \langle s, t, \delta, u, v, x, y\rangle$.*

PROOF: If $d = 2n+1$, then a direct computation shows that

$$x^y = \begin{pmatrix} 1 & \omega^{q-2} & -\omega^{-(q+1)}/2 \\ 0 & 1 & \omega^{-2q+1} \\ 0 & 0 & 1 \end{pmatrix}.$$

Thus $S = \langle x^{y^k} : 1 \le k \le |y|\rangle$ is a non-abelian group of order q^3, having derived group and centre of order q. A similar calculation for $d = 2n$ shows that $\langle x^{y^k} : 1 \le k \le |y|\rangle$ is a group of order q^2. These correspond to the subgroups X_S^1 of [8]; the result now follows from [8, Proposition 13.6.5]. □

In other cases, the standard generators of $\mathrm{SX}(d,q)$ have the property that it is easy to construct from them any root group, and consequently they generate $\mathrm{SX}(d,q)$. The root groups are defined with respect to a maximal split torus, the group of diagonal matrices in $\mathrm{SX}(d,q)$; for a detailed description see [8].

We now summarise other important properties of our generating sets.

Theorem 3.2 *Let $G = \langle s, t, \delta, u, v, x\rangle$.*

1. *Let $G = \mathrm{SL}(d,q)$ for $d \ge 2$. Then $W := \langle s, v, x\rangle$ defines the Weyl group of G modulo its diagonal matrices. The root groups of $\mathrm{SL}(d,q)$ are obtained by conjugating $\langle t^{\delta^k} : 1 \le k \le |y|\rangle$ by elements of W. If $d = 2n$ then $Y_0 := \{s, t, \delta, u, v\}$ generates $\mathrm{SL}(2,q) \wr C_n$.*

2. *Let* G *be* $\mathrm{Sp}(2n,q)$ *for* $n \geq 2$. *Then* $W := \langle s,u,v\rangle$ *defines the Weyl group of* G *modulo its diagonal matrices. The long root groups of* G *are obtained by conjugating* $\langle t^{\delta^k} : 1 \leq k \leq |y|\rangle$ *by elements of* W*, the short root groups of* G *are obtained by conjugating* $\langle x^{\delta^k} : 1 \leq k \leq |y|\rangle$ *by elements of* W*, and* $Y_0 := \{s,t,\delta,u,v\}$ *generates* $\mathrm{SL}(2,q) \wr S_n$.

3. *Let* G *be* $\mathrm{SU}(d,q)$ *for* $d \geq 3$. *Then* $W := \langle s,u,v\rangle$ *defines the Weyl group of* G *modulo its diagonal matrices. If* $d = 2n$ *then* $Y_0 := \{s,t,\delta,u,v\}$ *generates* $\mathrm{SU}(2,q) \wr S_n$.

Group	s	t	δ	u	v	x	y
$\mathrm{SL}(2n,q)$	$\begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$	$\begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$	$\begin{pmatrix} \omega & 0 \\ 0 & \omega^{-1} \end{pmatrix}$	I_2	$(e_1,e_2,\ldots,e_n)(f_1,f_2,\ldots,f_n)$	$\begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 \end{pmatrix}$	I_4
$\mathrm{SL}(2n+1,q)$	$\begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$	$\begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$	$\begin{pmatrix} \omega & 0 \\ 0 & \omega^{-1} \end{pmatrix}$	I_2	$\begin{pmatrix} 0 & 1 \\ -I_{2n} & 0 \end{pmatrix}$	I_4	I_4
$\mathrm{Sp}(2n,q)$	$\begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$	$\begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}$	$\begin{pmatrix} \omega & 0 \\ 0 & \omega^{-1} \end{pmatrix}$	$\begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$	$(e_1,e_2,\ldots,e_n)(f_1,f_2,\ldots,f_n)$	$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{pmatrix}$	I_4
$\mathrm{SU}(2n,q)$	$\begin{pmatrix} 0 & \alpha \\ \alpha^{-q} & 0 \end{pmatrix}$	$\begin{pmatrix} 1 & \alpha \\ 0 & 1 \end{pmatrix}$	$\begin{pmatrix} \omega^{q+1} & 0 \\ 0 & \omega^{-(q+1)} \end{pmatrix}$	$\begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$	$(e_1,e_2,\ldots,e_n)(f_1,f_2,\ldots,f_n)$	$\begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} \omega & 0 & 0 & 0 \\ 0 & \omega^{-q} & 0 & 0 \\ 0 & 0 & \omega^{-1} & 0 \\ 0 & 0 & 0 & \omega^{q} \end{pmatrix}$
$\mathrm{SU}(2n+1,q)$	$\begin{pmatrix} 0 & \alpha \\ \alpha^{-q} & 0 \end{pmatrix}$	$\begin{pmatrix} 1 & \alpha \\ 0 & 1 \end{pmatrix}$	$\begin{pmatrix} \omega^{q+1} & 0 \\ 0 & \omega^{-(q+1)} \end{pmatrix}$	$\begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$	$(e_1,e_2,\ldots,e_n)(f_1,f_2,\ldots,f_n)$	$\begin{pmatrix} 1 & 1 & -1/2 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{pmatrix}$	$\begin{pmatrix} \omega & 0 & 0 \\ 0 & \omega^{q-1} & \\ 0 & 0 & \omega^{-q} \end{pmatrix}$

Table 1: Standard generators for classical groups

4 Algorithm `One`

Algorithm `One` takes as input a generating set X for $G = \mathrm{SX}(d, q)$, and returns standard generators for G as SLPs in X. The generators are in standard form when referred to a basis constructed by the algorithm. The change-of-basis matrix that expresses this basis in terms of the standard basis for the natural module is also returned.

The algorithm employs a "divide-and-conquer" strategy.

Definition 4.1 *A strong involution in* $\mathrm{SX}(d, q)$ *for* $d > 4$ *is an involution whose* -1*-eigenspace has dimensions in the range* $(d/3, 2d/3]$.

Algorithm 1: `OneEven`$(X, type)$

```
/* X is a generating set for the perfect classical group G in odd
   characteristic, of type SL or Sp or SU, in even dimension.
   Return the standard generating set Y_0 for SL(2,q) ≀ C_{d/2} if type is
   SL, otherwise for SL(2,q) ≀ S_{d/2} as subgroup of G, the SLPs for
   the elements of Y_0 and the change-of-basis matrix.                  */
```

1 **begin**
2 $d :=$ the rank of the matrices in X;
3 if $d = 2$ then return `BaseCase` (X, *type*);
4 $q :=$ the size of the field over which these matrices are defined;
5 if *type* $=$ SU then $q := q^{1/2}$;
6 Find by random search $g \in G := \langle X \rangle$ of even order such that g powers to a strong involution h;
7 Let $2k$ be the dimension of the -1-eigenspace of h;
8 Find generators for the centraliser C of h in G;
9 In the derived subgroup C' of C find generating sets X_1 and X_2 for the images in $\mathrm{SX}(2k, q)$ and $\mathrm{SX}(d - 2k, q)$ of the direct factors of C';
10 $(s_1, t_1, \delta_1, u_1, v_1) :=$ `OneEven`$(X_1, type)$;
11 $(s_2, t_2, \delta_2, u_2, v_2) :=$ `OneEven`$(X_2, type)$;
12 Let $(e_1, f_1, \ldots, e_k, f_k, e_{k+1}, f_{k+1}, \ldots, e_d, f_d)$ be the concatenation of the hyperbolic bases obtained in lines 10 and 11;
13 $a := (s_1^2)^{v_1^{-1}}(s_2^2)$;
14 Find generators for the centraliser D of a in G;
15 In the derived subgroup D' of D find a generating set X_3 for the image in $\mathrm{SX}(4, q)$ of the direct factor that acts faithfully on $\langle e_k, f_k, e_{k+1}, f_{k+1} \rangle$;
16 In $\langle X_3 \rangle$ find the permutation matrix $b = (e_k, e_{k+1})(f_k, f_{k+1})$;
17 $v := v_1 b v_2$;
18 return $(s_1, t_1, \delta_1, u_1, v)$ and the change-of-basis matrix;
19 **end**

For $\mathrm{Sp}(d, q)$ and $\mathrm{SU}(d, q)$ the eigenspaces of an involution u are mutually orthogonal, and the form restricted to either eigenspace is non-degenerate. Thus, if these spaces have dimensions e and $d - e$, then the derived subgroup of the centraliser of u in

$\mathrm{SX}(d,q)$ is $\mathrm{SX}(e,q) \times \mathrm{SX}(d-e,q)$. Note that the dimension of the -1-eigenspace of an involution in $\mathrm{SX}(d,q)$ is always even.

Algorithm `OneEven` addresses the case of even d.

If the type is SL, then the centraliser of h is $\mathrm{GL}(E_+) \times \mathrm{GL}(E_-) \cap \mathrm{SL}(d,q)$ where E_+ and E_- are the eigenspaces of h. If the type is Sp, it is $\mathrm{Sp}(E_+) \times \mathrm{Sp}(E_-)$, and if the type is SU, it is $(U(E_+) \times U(E_-)) \cap \mathrm{SU}(d,q)$. In these last two cases the restriction of the form to each of the eigenspaces is non-singular, and each eigenspace is orthogonal to the other. Thus the concatenation of a hyperbolic basis of one eigenspace with a hyperbolic basis for the other eigenspace is a hyperbolic basis for the whole space. Hence we need to *construct* a hyperbolic basis only for the base cases of the algorithm.

We make the following observations on Algorithm `OneEven`.

1. As presented the algorithm has been simplified. In lines 10 and 11 we have ignored the change-of-basis matrices that are also returned; the change-of-basis returned at line 18 is defined by the concatenation of these bases.

2. The SLPs that express the standard generators in terms of X are also returned.

3. Generators for the involution centralisers in lines 8 and 14 are constructed using the algorithm of Bray [5], see Section 10. We need only a subgroup of the centraliser that contains its derived subgroup.

4. The generators for the direct summands constructed in lines 9 and 15 are constructed by forming suitable powers of the generators of the centraliser. This step is discussed in Section 9.

5. The algorithms for the `BaseCase` call in line 3 are discussed in Section 11. In summary `BaseCase`(X, *type*) returns the standard generating set Y and the associated SLPs for the classical group $\langle X \rangle$ of the specified type.

6. The search for an element that powers to a suitable involution is discussed in Section 6.

7. The recursive calls in lines 10 and 11 are in smaller dimension. Not only are the groups of smaller Lie rank, but the matrices have degree at most $2d/3$. Hence these calls only affect the time or space complexity of the algorithm up to a constant multiple; however they contribute to the length of the SLPs produced.

8. It is easy to check that a in line 13 is an involution: its -1-eigenspace is $\langle e_k, f_k, e_{k+1}, f_{k+1} \rangle$ and its $+1$-eigenspace is $\langle e_1, f_1, \ldots, e_{k-1}, f_{k-1}, e_{k+2}, f_{k+2}, \ldots, e_d, f_d \rangle$. We discuss how to find the "glue" element b in Section 11.2.

Algorithm `OneOdd`, which considers the case of odd degree d, is similar to Algorithm `OneEven`. Our commentary on the even degree case also applies.

Algorithm 2: `OneOdd`$(X, type)$

```
/* X is a generating set for the perfect classical group G in odd
   characteristic and odd dimension, of type SL or SU. Return the
   standard generating set for G, the SLPs for elements of this
   generating set, and the change-of-basis matrix.                */
```

1 **begin**
2 $d :=$ the rank of the matrices in X;
3 if $d = 3$ then return `BaseCase` (X, *type*);
4 $q :=$ the size of the field over which these matrices are defined;
5 if *type* = SU then $q := q^{1/2}$;
6 Find by random search $g \in G := \langle X \rangle$ of even order such that g powers to a strong involution h;
7 Let $2k$ be the dimension of the -1-eigenspace of h;
8 Find generators for the centraliser C of h in G;
9 In the derived subgroup C' of C find generating sets X_1 and X_2 for the images in $\mathrm{SX}(2k, q)$ and $\mathrm{SX}(d - 2k, q)$ of the direct factors of C', where X_1 centralises the -1-eigenspace of h;
10 $(s_1, t_1, \delta_1, u_1, v_1) :=$ `OneEven`$(X_1, type)$;
11 $(s_2, t_2, \delta_2, u_2, v_2, x, y) :=$ `OneOdd`$(X_2, type)$;
12 Let $(e_1, f_1, \ldots, e_k, f_k, e_{k+1}, f_{k+1}, \ldots, e_d, f_d)$ be the concatenation of the hyperbolic bases obtained in lines 10 and 11;
13 $a := (s_1^2)^{v_1^{-1}}(s_2^2)$;
14 Find generators for the centraliser D of a in G;
15 In the derived subgroup D' of D find a generating set X_3 for the image in $\mathrm{SX}(4, q)$ of the direct factor that acts faithfully on $\langle e_k, f_k, e_{k+1}, f_{k+1} \rangle$;
16 In $\langle X_3 \rangle$ find the permutation matrix $b = (e_k, e_{k+1})(f_k, f_{k+1})$;
17 $v := v_1 b v_2$;
18 return $(s_1, t_1, \delta_1, u_1, v, x, y)$ and the change-of-basis matrix;
19 **end**

We summarise the main algorithm as Algorithm `OneMain`.

The correctness and complexity of this algorithm, and the lengths of the resulting SLPs for the standard generators, are discussed in the rest of this paper.

5 Algorithm `Two`

We present a variant of the algorithms in Section 4 based on one recursive call rather than two. Again we denote the groups $\mathrm{SL}(d, q)$, $\mathrm{Sp}(d, q)$ and $\mathrm{SU}(d, q)$ by $\mathrm{SX}(d, q)$, and the corresponding projective group by $\mathrm{PX}(d, q)$.

The key idea is as follows. Suppose that d is a multiple of 4. We find an involution $h \in \mathrm{SX}(d, q)$, as in line 6 of `OneEven`, but insist that it should have both eigenspaces of dimension $d/2$.

Algorithm 3: `OneMain`$(X, type)$

end

```
/* X is a generating set for the perfect classical group G in odd
   characteristic, of type SL or Sp or SU. Return the standard
   generators for G, the SLPs for these generators, and
   change-of-basis matrix.                                    */
1 begin
2    d := the rank of the matrices in X;
3    (s, t, δ, u, v), B := OneEven(X, type);
4    Construct additional elements x and y;
5    return (s, t, δ, u, v, x, y) and the change-of-basis matrix B;
```

Let $\bar{h}$ be the image of h in $\mathrm{PX}(d,q)$. The centraliser of $\bar{h}$ in $\mathrm{PX}(d,q)$ acts on the pair of eigenspaces E_+ and E_- of h, interchanging them. We construct the projective centraliser of h by applying the algorithm of [5] to $\bar{h}$ and $\mathrm{PX}(d,q)$.

If we now find recursively the subset Y_0 of standard generators for $\mathrm{SX}(E_+)$ with respect the basis $\mathcal{B}$, then Y_0^g is a set of standard generators for $\mathrm{SX}(E_-)$ with respect to the basis $\mathcal{B}^g$. We now use these to construct standard generators for $\mathrm{SX}(d,q)$ exactly as in Algorithm `One`.

If d is an odd multiple of 2, we find an involution with one eigenspace of dimension exactly 2. The centraliser of this involution allows us to construct $\mathrm{SX}(2,q)$ and $\mathrm{SX}(d-2,q)$. The $d-2$ factor is now processed as above, since $d-2$ is a multiple of 4, and the 2 and $d-2$ factors are combined as in the first algorithm. Thus the algorithm deals with $\mathrm{SX}(d,q)$, for even values of d, in a way that is similar in outline to the familiar method of powering, that computes a^n, by recursion on n, as $(a^2)^{n/2}$ for even n and as $a(a^{n-1})$ for odd n.

Algorithms `TwoTimesFour` and `TwoTwiceOdd` describe the case of even d. Algorithm `TwoTimesFour` calls no new procedures except in line 5, where we construct an involution with eigenspaces of equal dimension. This construction is discussed in Section 7. Algorithm `TwoEven`, which summarises the even degree case, returns the generating set Y_0 defined in Section 3. We complete the construction of Y exactly as in Section 4.

If d is odd, then we find an involution whose -1-eigenspace has dimension 3, thus splitting d as $(d-3)+3$. Since $d-3$ is even, we apply the odd case *precisely once*.

The resulting `TwoOdd` is the same as `OneOdd`, except that it calls `TwoEven` rather than `OneEven`; similarly `TwoMain` calls `TwoOdd` and `TwoEven`.

The primary advantage of the second algorithm lies in its one recursive call. As we show in Section 13, this reduces the lengths of the SLPs for the standard generators.

Algorithm 4: `TwoTimesFour`$(X, type)$

/* X is a generating set for the perfect classical group G in odd characteristic, of type SL or Sp or SU, in dimension a multiple of 4. Return the standard generating set Y_0 for a copy of $\mathrm{SL}(2,q) \wr C_{d/2}$ if *type* is SL, otherwise $\mathrm{SL}(2,q) \wr S_{d/2}$ as subgroup of G, the SLPs for the elements of Y_0 and the change-of-basis matrix. */

1 **begin**

2 $d :=$ the rank of the matrices in X;

3 $q :=$ the size of the field over which these matrices are defined;

4 if *type* = SU then $q := q^{1/2}$;

5 Find by random search $g \in G := \langle X \rangle$ of even order such that g powers to an involution h with eigenspaces of dimension $2k = d/2$;

6 Find generators for the projective centraliser C of h in G and identify an element g of C that interchanges the two eigenspaces;

7 In the derived subgroup C' of C find a generating set X_1 for the image in $\mathrm{SX}(2k, q)$ one of the direct factors of C';

8 $(s_1, t_1, \delta_1, u_1, v_1) :=$ `TwoEven`$(X_1, type)$;

9 $X_2 := X_1^g$;

10 Conjugate all elements of $(s_1, t_1, \delta_1, u_1, v_1)$ by g to obtain solution $(s_2, t_2, \delta_2, u_2, v_2)$ for X_2;

11 Let $(e_1, f_1, \ldots, e_k, f_k, e_{k+1}, f_{k+1}, \ldots, e_d, f_d)$ be the concatenation of the hyperbolic basis obtained in line 9 and its image under g;

12 $a := (s_1^2)^{v_1^{-1}}(s_2^2)$;

13 Find generators for the centraliser D of a in G;

14 In the derived subgroup D' of D find a generating set X_3 for the image in $\mathrm{SX}(4, q)$ of the direct factor that acts faithfully on $\langle e_k, f_k, e_{k+1}, f_{k+1} \rangle$;

15 In $\langle X_3 \rangle$ find the permutation matrix $b = (e_k, e_{k+1})(f_k, f_{k+1})$;

16 $v := v_1 b v_2$;

17 return $(s_1, t_1, \delta_1, u_1, v)$ and the change-of-basis matrix;

18 **end**

Algorithm 5: `TwoTwiceOdd`$(X, type)$

/* X is a generating set for the perfect classical group G in odd characteristic, of type SL or Sp or SU, in dimension $d = 2(k+1)$ for even $k \geq 0$. Return the standard generating set Y_0 for a copy of $\mathrm{SL}(2,q) \wr C_{d/2}$ if *type* is SL, or $\mathrm{SL}(2,q) \wr S_{d/2}$ as subgroup of G, the SLPs for the elements of Y_0 and the change-of-basis matrix. */

1 **begin**
2 $d :=$ the rank of the matrices in X;
3 if $d = 2$ then return `BaseCase` (X, *type, false*);
4 $q :=$ the size of the field over which these matrices are defined;
5 if *type* = SU then $q := q^{1/2}$;
6 Find by random search $g \in G := \langle X \rangle$ of even order such that g powers to an involution h with eigenspaces of dimension 2 and $d-2$.
7 Find generators for the centraliser C of h in G;
8 In the derived subgroup C' of C find generating sets X_1 and X_2 for the images in $\mathrm{SX}(d-2,q)$ and $\mathrm{SX}(2,q)$ of the direct factors of C' where X_1 centralises the eigenspace of dimension 2;
9 $(s_1, t_1, \delta_1, u_1, v_1) :=$ `TwoTimesFour`$(X_1, type)$;
10 $(s_2, t_2, \delta_2, u_2, v_2) :=$ `TwoTwiceOdd`$(X_2, type)$;
11 Let $(e_1, f_1, \ldots, e_k, f_k, e_{k+1}, f_{k+1})$ be the concatenation of the hyperbolic bases obtained in lines 9 and 10;
12 $a := (s_1^2)^{v_1^{-1}}(s_2^2)$;
13 Find generators for the centraliser D of a in G;
14 In the derived subgroup D' of D find a generating set X_3 for the image in $\mathrm{SX}(4,q)$ of the direct factor that acts faithfully on $\langle e_k, f_k, e_{k+1}, f_{k+1} \rangle$;
15 In $\langle X_3 \rangle$ find the permutation matrix $b = (e_k, e_{k+1})(f_k, f_{k+1})$;
16 $v := v_1 b v_2$;
17 return $(s_1, t_1, \delta_1, u_1, v)$ and the change-of-basis matrix.
18 **end**

Algorithm 6: `TwoEven`$(X, type)$

/* X is a generating set for the perfect classical group G in odd characteristic, of type SL or Sp or SU, in even dimension. Return standard generating set Y_0 for a copy of $\mathrm{SL}(2,q) \wr S_{d/2} \leq G$, the SLPs for the elements of Y_0, and the change-of-basis matrix. */

1 **begin**
2 $d :=$ the rank of the matrices in X;
3 return `TwoTwiceOdd`$(X, type)$;
4 return `TwoTimesFour`$(X, type)$;
end

6 Finding strong involutions

In the first step of our main algorithms, as outlined in Sections 4 and 5, by random search, we obtain an element of even order that has as a power a strong involution. We now establish a lower bound to the proportion of elements of $\mathrm{SX}(d,q)$ that power up to give a strong involution.

A matrix is *separable* if its characteristic polynomial has no repeated factors. Fulman, Neumann & Praeger [13] prove the following.

Theorem 6.1 *The probability that an element of* $\mathrm{GL}(d,q)$ *or* $\mathrm{U}(d,q)$ *is separable is at least* $1-2/q$. *The probability that an element of* $\mathrm{Sp}(d,q)$ *is separable is at least* $1-3/q+O(1/q^2)$.

We use these results to assist in our analysis of the proportion of strong involutions in $\mathrm{SX}(d,q)$.

6.1 The special linear case

We commence our analysis with $\mathrm{SL}(d,q)$. We first estimate the probability that a random element of $\mathrm{GL}(d,q)$ has a power that is an involution having an eigenspace of dimension within a given range and then derive similar results for $\mathrm{SL}(d,q)$.

Lemma 6.2 *The number of irreducible monic polynomials of degree* $e>1$ *with coefficients in* $\mathrm{GF}(q)$ *is* k *where* $(q^e-1)/e > k \geq q^e(1-q^{-1})/e$.

Proof: Let k denote the number of such polynomials. We use the inclusion-exclusion principle to count the number of elements of $\mathrm{GF}(q^e)$ that do not lie in any maximal subfield containing $\mathrm{GF}(q)$, and divide this number by e, since every irreducible monic polynomial of degree e over $\mathrm{GF}(q)$ corresponds to exactly e such elements. Thus

$$k = \frac{q^e - \sum_i q^{e/p_i} + \sum_{i<j} q^{e/p_ip_j} - \cdots}{e}$$

where $p_1 < p_2 < \cdots$ are the distinct prime divisors of e. The inequality $(q^e-1)/e > k$ is obvious. If e is a prime, then $k=(q^e-q)/e \geq q^e(1-1/q)/e$, with equality if $e=2$. Now suppose that $e \geq 4$, and let ℓ denote the largest prime dividing e. Then from the above formula

$$\begin{aligned} ek &\geq q^e - q^{e/\ell} - q^{(e/\ell)-1} - \ldots - 1 \\ &= q^e - (q^{1+e/\ell}-1)/(q-1) \\ &\geq q^e - q^{1+e/2} + 1 \\ &> q^e - q^{e-1} \end{aligned}$$

as $1+e/2 \leq e-1$. □

Lemma 6.3 *Let $e > d/2$ for $d \geq 4$. The proportion of elements of $\mathrm{GL}(d,q)$ whose characteristic polynomial has an irreducible factor of degree e is $(1/e)(1+O(1/q))$. More precisely, there are universal constants c_1 and c_2 such that the proportion is always between $(1/e)(1-c_2/q)$ and $(1/e)(1-c_1/q)$.*

PROOF: Let the characteristic polynomial of $g \in \mathrm{GL}(d,q)$ have an irreducible factor $h(x)$ of degree e. Then $\{w \in V : w.h(g) = 0\}$ is a subspace of V of dimension e. It follows that the number of elements of $\mathrm{GL}(d,q)$ of the required type is $k_1k_2k_3k_4k_5$ where k_1 is the number of subspaces of V of dimension e, k_2 is the number of irreducible monic polynomials of degree e over $\mathrm{GF}(q)$, k_3 is the number of elements of $\mathrm{GL}(e,q)$ that have a given irreducible characteristic polynomial, k_4 is the order of $\mathrm{GL}(d-e,q)$, and k_5 is the number of complements in V to a subspace of dimension e. In more detail,

$$\begin{aligned} k_1 &= \frac{(q^d-1)(q^d-q)\cdots(q^d-q^{e-1})}{(q^e-1)(q^e-q)\cdots(q^e-q^{e-1})} \\ k_3 &= \frac{(q^e-1)(q^e-q)\cdots(q^e-q^{e-1})}{(q^e-1)} \\ k_4 &= (q^{d-e}-1)(q^{d-e}-q)\cdots(q^{d-e}-q^{d-e-1}) \\ k_5 &= q^{e(d-e)}. \end{aligned}$$

The formula for k_3 arises by taking the index in $\mathrm{GL}(e,q)$ of the centraliser of an irreducible element, this centraliser being cyclic of order q^e-1. The formula for k_2 is given in Lemma 6.2. Hence $k_1k_2k_3k_4k_5 = |\mathrm{GL}(d,q)| \times k_2/(q^e-1)$. The result follows. Note that c_1 and c_2 may be taken to be positive. □

Lemma 6.4 *Let $e \in (d/3, d/2]$ for $d \geq 4$. Let S_1 denote the number of elements of $G := \mathrm{GL}(d,q)$ whose characteristic polynomial has two distinct irreducible factors of degree e; let S_2 denote the number of elements of G whose characteristic polynomial has a repeated factor of degree e; and let S_3 denote the number of elements of G whose characteristic polynomial has exactly one irreducible factor of degree e. Then $S_1 = \frac{1}{2}|\mathrm{GL}(d,q)|e^{-2}(1+O(q^{-1}))$, and $S_2 = |\mathrm{GL}(d,q)|O(q^{-1})$, and $S_3 = |\mathrm{GL}(d,q)|(e^{-1}-e^{-2})(1+O(q^{-1}))$, where the constants implied by the O notation are absolute constants, independent of d.*

PROOF: In the proof of Lemma 6.3 the fact that $e > d/2$ was only used to ensure that the characteristic polynomial of $g \in \mathrm{GL}(d,q)$ has only one irreducible factor of degree e. If the proof of Lemma 6.3 is repeated to estimate S_3, the factor k_4 must be replaced by the number of elements of $\mathrm{GL}(d-e,q)$ whose characteristic polynomials do not have an irreducible factor of degree e. A direct application of this lemma then shows that this number is the order of $\mathrm{GL}(d-e,q)$ multiplied by a factor of the form $(1-e^{-1})(1+O(1/q))$. Thus $S_3 = |\mathrm{GL}(d,q)|(e^{-1}-e^{-2})(1+O(q^{-1}))$.

The proportion of non-separable elements of $\mathrm{GL}(d,q)$ is $O(q^{-1})$ by Theorem 6.1, so $S_2 = |\mathrm{GL}(d,q)|O(q^{-1})$, and we may ignore the condition that the irreducible factors in question are distinct in our estimate of S_1. If the proof of Lemma 6.3 is repeated again to estimate S_1, the factor k_4 may be replaced by the number of elements of $\mathrm{GL}(d-e,q)$ whose characteristic polynomials have an irreducible factor of degree e, and the total must be divided by 2, since the group elements in question normalise two subspaces of dimension e. Thus $S_1 = \frac{1}{2}|\mathrm{GL}(d,q)|e^{-2}(1+O(q^{-1}))$. □

We will now show that the results of these lemmas hold if $\mathrm{GL}(d,q)$ is replaced by $\mathrm{SL}(d,q)$. First we need a preliminary result. The norm map $N : \mathrm{GF}(q^e) \to \mathrm{GF}(q)$ defines a homomorphism from $\mathrm{GF}(q^e)^\times$ onto $\mathrm{GF}(q)^\times$. Let I_a denote the intersection of the pre-image of an arbitrary $a \in \mathrm{GF}(q)^\times$ with the set of elements of $\mathrm{GF}(q^e)$ that lie in no proper subfield. Since in general the norm map will not map a proper subfield of $\mathrm{GF}(q^e)$ onto $\mathrm{GF}(q)$, the size of I_a varies with a.

Lemma 6.5 $|I_a| = c_a \frac{q^e-1}{q-1}$, *where* $c_a = 1 + O(1/q)$.

Proof: The set of elements of $\mathrm{GF}(q^e)$ mapped to $a \in \mathrm{GF}(q)^\times$ under the norm map $x \mapsto x^{1+q+q^2+\ldots+q^{e-1}}$ has size $1+q+\ldots+q^{e-1} = (q^e-1)/(q-1)$. To obtain I_a, we must remove those elements which lie in a proper subfield containing $\mathrm{GF}(q)$. Following the proof of Lemma 6.2, the number of elements of $\mathrm{GF}(q^e)^\times$ which are in a proper subfield is less than $q^{1+e/2} - 1 < q^{e-2}$ for $e \geq 6$. If e is prime, the number of elements of $\mathrm{GF}(q)$ of norm a is either 0 or $\gcd(q-1,e)$. If $e = 4$, the number of elements of $\mathrm{GF}(q^2)$ of norm a is either 0 or $2(q+1)$. In all cases the result holds. □

Lemma 6.6 *The results of Lemmas* 6.3 *and* 6.4 *hold if* $\mathrm{GL}(d,q)$ *is replaced by* $\mathrm{SL}(d,q)$.

Proof: We first prove that the proportion quoted in Lemma 6.3 is also true for $\mathrm{SL}(d,q)$. In this case k_4 must be replaced by the number of elements of $\mathrm{GL}(d-e,q)$ of a specified determinant. But the number of such elements is exactly the number of elements of $\mathrm{GL}(d-e,q)$ divided by $q-1$; so the result follows. Similarly with Lemma 6.4, if $e \neq d/2$ the proportions in the two cases are exactly equal. The point is that in these cases we consider the number of elements in $\mathrm{GL}(d-e,q)$ or in $\mathrm{GL}(d-2e,q)$, and we need to replace these numbers by the number of such elements having a given determinant, thus reducing the number by a factor of $q-1$. This also applies to S_2.

However, this argument breaks down when $e = d/2$, as in this case we cannot adjust the determinant of the element being constructed by requiring an element of $\mathrm{GL}(d-2e,q)$ to have a given determinant. Indeed, in this case the proportions are not exactly equal for $\mathrm{GL}(d,q)$ and $\mathrm{SL}(d,q)$.

Consider the norm map $N : \mathrm{GF}(q^e) \to \mathrm{GF}(q)$. Let I_a be the intersection of the pre-image of an arbitrary $a \in \mathrm{GF}(q)^\times$ with the set of elements of $\mathrm{GF}(q^e)$ that lie in no proper subfield. Lemma 6.5 implies that $|I_a| = c_a(q^e-1)/(q-1)$, where $c_a = 1+O(1/q)$.

Now S_1 is approximated, in the case of $\mathrm{SL}(d,q)$, by

$$\frac{1}{2}\sum_a |I_a||I_{a^{-1}}| \left(\frac{|\mathrm{GL}(e,q)|}{e-1}\right)^2$$

where the error in the approximation is due to the fact that we have ignored the condition that the irreducible factors of the characteristic polynomial should be distinct. The analogous estimate for S_1 in the case of $\mathrm{GL}(d,q)$ replaces $\sum_a |I_a||I_{a^{-1}}|$ by $\sum_{a,b} |I_a||I_b|$, which is approximately $q-1$ times as big, the error arising from the fact that the cardinality of I_a is not quite constant. We have seen that this error corresponds to a factor of the form $1+O(1/q)$, as does ignoring the condition that the irreducible factors in the characteristic polynomial should be different, and omitting the contribution of S_2. Note that the assumption $d \geq 4$ enforces the condition $e \geq 2$. □

We now obtain a lower bound for the proportion of $g \in \mathrm{SL}(d,q)$ such that g has even order $2n$, and g^n has an eigenspace with dimension in a given range. To perform this calculation, we consider the cyclic groups C_{q^e-1} of order q^e-1. If n is an integer, we write $v_2(n)$ for the 2-adic value of n.

Lemma 6.7 *If $v_2(m) = v_2(n)$ then $v_2(q^m-1) = v_2(q^n-1)$.*

PROOF: It suffices to consider the case where $m = kn$, and k is odd. Then $(q^m-1)/(q^n-1)$ is the sum of k powers of q^n, and so is odd. □

Lemma 6.8 *If $u < v$ then $v_2(q^{2^u}-1) < v_2(q^{2^v}-1)$, and if $u > 0$ then $v_2(q^{2^u}-1) = v_2(q^{2^{u+1}}-1) - 1$.*

PROOF: Observe that $(q^{2^{u+1}}-1)/(q^{2^u}-1) = q^{2^u}+1$ which is even. Now $v_2(q^{2^u}-1) > 1$ if $u > 0$. It then follows that $v_2(q^{2^u}+1) = 1$. □

Theorem 6.9 *For some absolute constant c, the proportion of $g \in \mathrm{SL}(d,q)$ of even order, such that a power of g is an involution with its -1-eigenspace of dimension in the range $(d/3, 2d/3]$, is at least c/d.*

PROOF: Let 2^k be the unique power of 2 in the range $(d/3, 2d/3]$. By Lemma 6.6 it suffices to prove that if $g \in \mathrm{SL}(d,q)$ has an irreducible factor of degree 2^k then the probability that g has the required property is bounded away from 0.

Let $\{W_i : i \in I\}$ be the set of composition factors of V under the action of $\langle g \rangle$. Let n_i be the order of the image of g in $\mathrm{GL}(W_i)$, and set $w_i = v_2(n_i)$, and $w = \max_i(w_i)$, and $d_i = \dim(W_i)$. If $w > 0$, then g has even order $2n$ say, and in this case the -1-eigenspace of $z := g^n$ has dimension $\sum d_i$, where the sum is over those values of i for which $w_i = w$.

Suppose now that the characteristic polynomial of g has exactly one irreducible factor of degree 2^k. By renumbering if necessary we may assume that $d_1 = 2^k$. Set $x = v_2(q^{2^k} - 1)$. The probability that $w_1 = x$ is slightly greater than $1/2$. This is because the action of g on W_1 embeds g at random in $\mathrm{GF}(q^{2^k})$, which is a cyclic group of order an odd multiple of 2^x. The distribution of possible values of g is uniform among those elements that do not lie in a proper subfield of $\mathrm{GF}(q^{2^k})$. But non-zero elements of such subfields do not have order a multiple of 2^x. If $w_1 = x$ then necessarily $w_i < w_1$ for all $i > 1$, and $w_1 = w$. It follows that g will then have even order, and that z will be an involution whose -1-eigenspace will have dimension exactly 2^k. There is a slight problem with the elements of $\mathrm{SL}(d, q)$ whose characteristic polynomials have two irreducible factors of degree 2^k, as such elements may power up to an involution whose -1-eigenspace has dimension 2^{k+1}, but the estimate of S_1 in Lemma 6.4 shows that this problem does not affect the correctness of the theorem. □

Corollary 6.10 *Such an element g in $\mathrm{SL}(d, q)$ can be found with at most $O(d(\xi + d^3 \log q))$ field operations, where ξ is the cost of constructing a random element.*

Proof: Theorem 6.9 implies that a search of length $O(d)$ will find such an element g. In $O(d^3)$ field operations the characteristic polynomial $f(t)$ of g can be computed (see [18, Section 7.2]); in $O(d^2 \log q)$ field operations it can be factorised as $f(t) = \prod_{i=1}^{k} f_i(t)$, where the $f_i(t)$ are irreducible (see [34, Theorem 14.14]).

Following the notation of the proof of Theorem 6.9, we may take W_i to be the kernel of $f_i(g)$. It remains to calculate w_i. Let m_i be the odd part of $q^{d_i} - 1$, where d_i is the degree of f_i. Now compute $s := (f_i(t)) + t^{m_i}$ in $\mathrm{GF}(q)[t]/(f_i(t))$, and iterate $s := s^2$ until s is the identity. The number of iterations determines w_i, and it is now easy to determine whether or not g satisfies the required conditions. All of the above steps may be carried out in at most $O(d^3 \log q)$ field operations. □

6.2 The symplectic and unitary groups

We first consider the symplectic groups. If $h(x) \in \mathrm{GF}(q)[x]$ is a monic polynomial with non-zero constant term, let $\tilde{h}(x) \in \mathrm{GF}(q)[x]$ be the monic polynomial whose zeros are the inverses of the zeros of $h(x)$. Hence the multiplicity of a zero of $h(x)$ is the multiplicity of its inverse in $\tilde{h}(x)$, and $h(x)\tilde{h}(x)$ is a symmetric polynomial. We start with this analogue of Lemma 6.3.

Lemma 6.11 *Let $m > n/2$ where $n \geq 2$. The proportion of elements of $\mathrm{Sp}(2n, q)$ whose characteristic polynomial has a factor $h(x)$ where $h(x)$ is irreducible of degree m and $h(x) \neq \tilde{h}(x)$ is $(1/2m)(1 + O(1/q))$, where the constants implied by the O notation are absolute constants, independent of n.*

PROOF: Let $g \in \mathrm{Sp}(2n,q)$ act on the natural module V, and let $h(x)$ be an irreducible factor of degree m of the characteristic polynomial $f(x)$ of g. Let V_0 be the kernel of $h(g)$. Since $h(x) \neq \tilde{h}(x)$ it follows that V_0 is totally isotropic. Also $\tilde{h}(x)$ is a factor of $f(x)$, and if V_1 is the kernel of $\tilde{h}(x)$ then V_1 is totally isotropic. Since $h(x)$ and $\tilde{h}(x)$ divide $f(x)$ with multiplicity 1, V_0 and V_1 are uniquely determined, and the form restricted to $V_0 \oplus V_1$ is non-singular. Now let $e_1, \ldots, e_m$ be a basis for V_0. A basis $f_1, \ldots, f_m$ for V_1 is then determined by the conditions $B(e_i, f_j) = 0$ for $i \neq j$, and $B(e_i, f_i) = 1$ for all i, where $B(-,-)$ is the symplectic form that is preserved. The matrix for g restricted to V_0 now determines the matrix of g restricted to V_1, since g preserves the form.

Thus the number of possibilities for g is the product $k_1k_2k_3k_4k_5/2$, where k_1 is the number of choices for V_0, and k_2 is the number of choices for V_1 given V_0, and k_3 is the number of irreducible monic polynomials $h(x)$ of degree m over $\mathrm{GF}(q)$ such that $h(x) \neq \tilde{h}(x)$, and k_4 is the number of elements of $\mathrm{GL}(m,q)$ with a given irreducible characteristic polynomial, and k_5 is the order of $\mathrm{Sp}(2n-2m,q)$. The factor $1/2$ in the above expression arises from the fact that every such element g is counted twice, because of the symmetry between $h(x)$ and $\tilde{h}(x)$. In more detail

$$\begin{aligned}
k_1 &= \frac{(q^{2n}-1)(q^{2n-1}-q)(q^{2n-2}-q^2)\cdots(q^{2n-m+1}-q^{m-1})}{(q^m-1)(q^m-q)(q^m-q^2)\cdots(q^m-q^{m-1})} \\
k_2 &= q^{(2n-m)+(2n-m-1)+(2n-m-2)+\cdots+(2n-2m+1)} \\
k_3 &\sim q^m/m \\
k_4 &= \frac{(q^m-1)(q^m-q)(q^m-q^2)\cdots(q^m-q^{m-1})}{q^m-1} \\
k_5 &= q^{(n-m)^2}\prod_{i=1}^{n-m}(q^{2i}-1).
\end{aligned}$$

These results are obtained as follows. For k_1, we count the number of sequences of linearly independent elements $(e_1, e_2, \ldots)$ such that each is orthogonal to its predecessors, and divide by the order of $\mathrm{GL}(m,q)$. For k_2, we observe that there is a 1-1 correspondence between the set of candidate subspaces for V_1 and the set of sequences $(f_1, f_2, \ldots, f_m)$ of elements of V such that each f_j satisfies m linearly independent conditions $B(e_i, f_j) = 0$ for $i \neq j$, and $B(e_j, f_j) = 1$, and $B(f_k, f_j) = 0$ for $k < j$. We observe that k_3 is the number of orbits of the Galois group of $\mathrm{GF}(q^m)$ over $\mathrm{GF}(q)$ acting on those $a \in \mathrm{GF}(q^m)$ that do not lie in a proper subfield containing $\mathrm{GF}(q)$, and have the property that the orbit of a does not contain a^{-1}. This last condition is equivalent to the statement that $h(x) \neq \tilde{h}(x)$. Note that $h(x) = \tilde{h}(x)$ if and only if m is even, and $a^{-1} = a^{q^{m/2}}$. A precise formula for k_3 would be rather complex, so we obtain instead the following estimate. If we ignore this last condition, then Lemma 6.2 estimates k_3. Now it is clear that if $a \in \mathrm{GF}(q^m)$ satisfies the above equation then the norm of a is 1. In other words, the constant term of $h(x)$ is 1. But this is exactly the problem solved in the proof of Lemma 6.6, so we find that, for some absolute constants

c_1 and c_2, k_3 lies between $(1-c_2/q)q^m/m$ and $(1-c_1/q)q^m/m$. The product of the k_i is $k_3|\mathrm{Sp}(2n,q)|/(q^m-1)$ and the result follows. □

Lemma 6.12 *Let $m \in (n/3, n/2]$. Let S_1 denote the number of elements of $G := \mathrm{Sp}(2n,q)$ whose characteristic polynomial has four distinct irreducible factors of degree m, of the form $h(x)$, $\tilde{h}(x)$, $k(x)$, and $\tilde{k}(x)$; let S_2 denote the number of elements of G whose characteristic polynomial has two distinct repeated factors of degree m, of the form $h(x)$ and $\tilde{h}(x)$; and let S_3 denote the number of elements of G whose characteristic polynomial has exactly two distinct irreducible factors of degree m, of the form $h(x)$ and $\tilde{h}(x)$. Then $S_1 = \frac{1}{8}|\mathrm{Sp}(2n,q)|m^{-2}(1+O(q^{-1}))$, and $S_2 = |\mathrm{Sp}(2n,q)|O(q^{-1})$, and $S_3 = \frac{1}{2}|\mathrm{Sp}(2n,q)|(m^{-1}-\frac{1}{2}m^{-2})(1+O(q^{-1}))$, where the constants implied by the O notation are absolute constants, independent of n.*

PROOF: The proof is similar to that of Lemma 6.4. Care has to be taken with counting the number of times that elements of the analogue of S_1 are counted. The characteristic polynomial of such an element g now has four distinct irreducible factors $h(x)$, $\tilde{h}(x)$, $k(x)$, and $\tilde{k}(x)$ of degree m. This leads to such elements being counted eight times. □

We now obtain the analogue of Theorem 6.9.

Theorem 6.13 *For some absolute constant $c > 0$, the proportion of $g \in \mathrm{Sp}(2n,q)$ of even order, such that a power of g is an involution with its -1-eigenspace of dimension in the range $(2n/3, 4n/3]$, is at least c/n.*

PROOF: Given Lemmas 6.11 and 6.12, the proof is essentially the same as that of Theorem 6.9. We adopt the notation of that proof. One must consider the contribution of W_i to the eigenspaces of z when the characteristic polynomial of g restricted to W_i is an irreducible polynomial $h(x)$ such that $h(x) = \tilde{h}(x)$. But in this case if α is a zero of $h(x)$ then so is α^{-1}, and $\alpha^{q^m} = \alpha^{-1}$, where W_i has dimension $2m$, and the order of g divides q^m+1. It is easy to see that if i is even then $v_2(q^i+1) = 1$, and if i is odd then $v_2(q^i+1) = v_2(q+1)$. Thus W_i will contribute nothing to the dimension of the -1-eigenspace of z. The result follows. □

We finally turn to the unitary groups.

Theorem 6.14 *For some absolute constant $c > 0$, the proportion of $g \in \mathrm{SU}(d,q)$ that have even order, such that a power of g is an involution with its -1-eigenspace of dimension in the range $(d/3, 2d/3]$, is at least c/d.*

PROOF: The analysis in this case is almost exactly the same as for the symplectic groups. The only difference comes from the analysis of the restriction of g to W_i where now we require $h(x)$ to be the image of $\tilde{h}(x)$ under the Frobenius map $a \mapsto a^q$. This now requires W_i to have odd dimension $2t+1$, say, and then the order of g will divide

$q^{2t+1}+1$. □

In summary, Theorems 6.9, 6.13 and 6.14 provide an estimate of the complexity of finding a strong involution of the type required as $O(d(\xi + d^3 \log q))$ field operations.

In Algorithm `TwoMain`, we search in $\mathrm{SX}(d,q)$ for an element that powers to an involution with eigenspaces of dimension 2 and $d-2$ if d is twice odd, and 3 and dimension $d-3$ if d is odd. Lemma 6.3 and its analogues for the classical groups show that we can find such elements in $O(d)$ trials.

7 Involutions with eigenspaces of equal dimension

Our next objective is to describe and analyse an algorithm to construct an involution in $\mathrm{SX}(d,q)$ with eigenspaces of equal dimension. This necessarily presupposes that d is a multiple of 4. We use such an element in Algorithm `TwoEven`.

We describe a recursive procedure to construct an involution in $G = \mathrm{SL}(d,q)$ whose -1-eigenspace has a specified even dimension e.

1. Search randomly for $g \in G$ of even order that powers to an involution h_1 satisfying the conditions of Theorem 6.9.

2. Let r and s denote the ranks of the -1- and $+1$-eigenspaces of h_1.

3. If $r = e$ then h_1 is the desired involution.

4. Consider the case where $s \leq e < r$. Construct the centraliser in G of h_1 and obtain generators for the special linear group S_- on the -1-space, where S_- acts as the identity on the $+1$-eigenspace of h_1. By recursion on d, an involution can be found in S_- whose -1-eigenspace has dimension e.

5. Consider the case where $e \leq \min(r,s)$. If $r \leq s$ then construct the centraliser in G of h_1; using the method of Section 9, obtain generators for the special linear group S_- on this -1-eigenspace, where S_- acts as the identity on the $+1$-eigenspace. By recursion on d, an involution can be found in S_- whose -1-eigenspace has dimension e. Similarly, if $s < r$ then construct S_+, and search in S_+ for an involution whose -1-eigenspace has dimension e.

6. Consider the cases where $s \geq e > r$ or $e \geq \max(r,s)$. Construct the centraliser in G of h_1, and obtain generators for the special linear group S_+ on the $+1$-eigenspace of h_1, where S_+ acts as the identity on the -1-eigenspace. Now an involution h_2 is found recursively in S_+ whose -1-eigenspace has dimension $e-r$. Then $h_1 h_2$ is an involution of the required type.

The recursion is founded trivially with the case $d = 4$.

Theorem 7.1 *Using this algorithm, an involution in* $\mathrm{SL}(d,q)$ *can be constructed with* $O(d(\xi + d^3 \log q))$ *field operations that has its* -1*-eigenspace of any even dimension in* $[0,d]$.

PROOF: Corollary 6.10 implies that h_1 can be constructed with at most $O(d(\xi + d^3 \log q))$ field operations. We shall see in Sections 9 and 10 that generators for S_- and S_+ can be constructed in at most $O(d\xi + d^3 \log q)$ field operations. Thus the above algorithm requires $O(d(\xi + d^3 \log q))$ field operations, plus the number of field operations required in the recursive call. Since the dimension of the matrices in a recursive call is at most $2d/3$, the total complexity is as stated. □

Similar results can be obtained for the other classical groups.

8 Exponentiation

A frequent step in our algorithms is computing the power g^n for some $g \in \mathrm{GL}(d,q)$ and integer n. Sometimes we raise g to a high power in order to construct an involution, and we may write down this involution without performing the calculation. However, if, for example, we want to construct elements of one direct factor of a direct product of two groups by exponentiation, then we must explicitly compute the required power.

The value of n may be as large as $O(q^d)$. We could construct g^n with $O(\log(n))$ multiplications using the familiar black-box squaring technique. Instead, we describe the following faster algorithm to perform this task.

1. Construct the Frobenius normal form of g and record the change-of-basis matrix.

2. From the Frobenius normal form, we read off the minimal polynomial $h(x)$ of g, and factorise $h(x)$ as a product of irreducible polynomials.

3. This form determines a multiplicative upper bound to the order of g. If $\{f_i(x) : i \in I\}$ is the set of distinct irreducible factors of $h(x)$, and if d_i is the degree of $f_i(x)$, then the order of the semi-simple part of g divides $\prod_i q^{d_i} - 1$, and the order of the idempotent part of g can be read off directly. The product of these two factors gives the required upper bound m.

4. If $n > m$ we replace n by $n \bmod m$. By repeated squaring we calculate $x^n \bmod h(x)$ as a polynomial of degree d.

5. This polynomial is evaluated in g to give g^n.

6. Conjugate g^n by the inverse of the change-of-basis matrix to return to the original basis.

We now consider the complexity of this algorithm.

Lemma 8.1 *Let $g \in \mathrm{GL}(d,q)$ and let $0 \leq n < q^d$. Then g^n can be computed using the above algorithm with $O(d^3 + d^2 \log d \log\log d \log q)$ field operations.*

PROOF: The Frobenius normal form of g can be computed with $O(d^3)$ field operations [33] and provides the minimal polynomial. The minimal polynomial can be factored in $O(d^2 \log q)$ field operations [34, Theorem 14.14]. Calculating $x^n \bmod h(x)$ requires $O(\log(n))$ multiplications in $\mathrm{GF}(q)[x]/(h(x))$, at most $O(d^2 \log d \log\log d \log q)$ field operations [34]. Evaluating the resultant polynomial in g requires $O(d)$ matrix multiplications; but multiplying by g only costs $O(d^2)$ field operations, since g is sparse in Frobenius normal form. Finally, conjugating g by the inverse of the change-of-basis matrix costs a further $O(d^3)$ field operations. □

One should consider the cost of dividing m by n, even though this does not contribute to the number of field operations. However, for our applications, the exponent n is always less than q^d, so reducing m modulo n is unnecessary.

There is no need to prefer one normal form for g to another, provided that the normal form can be computed in at most $O(d^3)$ field operations, the form is sparse, and the minimum polynomial and multiplicative upper bound for the order of g can be determined readily from the normal form.

This algorithm is similar to that of [10] to determine the order of an element of $\mathrm{GL}(d,q)$.

9 Derived subgroups and direct products

In this section, we solve two problems which have closely related solutions.

1. Given a generating set for H where $\mathrm{SX}(e,q) \leq H \leq \mathrm{GX}(e,q)$, construct a generating set for the derived subgroup of H.

2. Given a generating set of $\mathrm{SX}(e,q) \times \mathrm{SX}(d-e,q)$, construct a generating set for one of the direct factors.

Our solutions and their analysis rely on the work of Niemeyer & Praeger [25, 27]. We assume throughout that the characteristic is odd.

9.1 Constructing the derived group

The one-sided Monte Carlo recognition algorithm of [25] takes as input a subset Y of $\mathrm{GX}(d,q)$ and endeavours to prove that $G := \langle Y \rangle$ contains $\mathrm{SX}(d,q)$, given that G is an irreducible subgroup of $\mathrm{GX}(d,q)$ that does not preserve any bilinear or quadratic form not preserved by $\mathrm{GX}(d,q)$.

We say that P is a set of *test primes* if, whenever $S \subseteq G$ and S has an element of order a multiple of p for all $p \in P$, then $\langle S \rangle$ either contains $\mathrm{SX}(d,q)$, or is reducible, or preserves a form not preserved by $\mathrm{SX}(d,q)$. Now S is a set of *test elements* for G

if for every $p \in P$, there is $g \in S$ of order a multiple of p. To find a suitable set S of test elements, the expected number of random elements to be examined is at most $O(\log \log d)$; see [25, Proposition 7.5].

A *primitive prime divisor* of $q^e - 1$ is a prime divisor of $q^e - 1$ that does not divide $q^i - 1$ for any positive integer $i < e$. If r is a primitive prime divisor of $q^e - 1$ then $r \equiv 1 \bmod e$, and so $r \geq e + 1$. Further, r is a *large* primitive prime divisor of $q^e - 1$ if r is a primitive prime divisor of $q^e - 1$, and either $r > e + 1$ or r^2 divides $q^e - 1$. Finally, r is defined to be a *basic* primitive prime divisor of $q^e - 1$ if r is a primitive prime divisor of $p^{ae} - 1$ where p is prime and $q = p^a$. A *ppd*$(d, q; e)$ element of G is one whose order is a multiple of a primitive prime divisor of $q^e - 1$.

Omitting the orthogonal groups, Theorem 5.7 of [25] proves the following.

Theorem 9.1 *Let* $\mathrm{SX}(d, q) \leq G \leq \mathrm{GX}(d, q)$. *The proportion ppd*$(G, e)$ *of ppd*$(d, q; e)$ *elements of* G *satisfies* $1/(e+1) \leq \mathrm{ppd}(G, e) \leq 1/e$, *except for odd* e *in the symplectic case and even* e *in the unitary case when it is 0.*

Consider first the general case where G is a *generic subgroup* of $\mathrm{GL}(d, q)$ [25, Definition 3.2]. Now, in all but one exceptional case, P has the property that each prime in P does not divide $q - 1$ when $G = \mathrm{SL}(d, q)$ and does not divide $q^2 - 1$ when $G = \mathrm{SU}(d, q)$. Hence the elements of S *remain* test elements when raised to the power n, where $n = q - 1$ in the case of $\mathrm{GL}(d, q)$, and is $q + 1$ in the case of $\mathrm{U}(d, q)$. (Recall that $\mathrm{Sp}(d, q)$ is perfect.) Thus the n-th powers of the test elements also generate either $\mathrm{SX}(d, q)$, or a reducible group, or a group that preserves some form not preserved by $\mathrm{GX}(d, q)$.

In more detail, the set S of test elements for G contains two elements, g_1 and g_2: one is a basic and the other a large $\mathrm{ppd}(d, q; e_i)$ elements where $e_1 \neq e_2$, and $e_i > d/2$ for $i = 1, 2$. Hence our putative generating set for G' contains two elements h_1 and h_2, powers of g_1 and g_2, that are $\mathrm{ppd}(d, q; e_i)$ elements. Thus they act irreducibly on subspaces W_1 and W_2 of V of dimensions e_1 and e_2 respectively. It is now easy to deduce that the probability that $\langle h_1, h_2 \rangle$ acts irreducibly on the underlying space V is at least $\prod_{i=1}^{\infty}(1 - 1/2^i) > 0.28$. Hence in general the subspaces will span V; if not, then h_1 may be replaced by a suitable G-conjugate of h_1. Thus we may assume that H acts irreducibly on V.

It remains to decide whether H preserves some form not preserved by $\mathrm{SX}(d, q)$. To prove that $g \in \mathrm{SL}(d, q)$ does not preserve a non-degenerate symplectic or symmetric bilinear form, it suffices to prove that the order of g is a multiple of a primitive prime divisor of $q^e - 1$ for some odd $e > d/2$. Since $d > 2$ such an e exists, and as e is odd such a primitive prime divisor exists. Theorem 5.7 of [25] proves that the proportion of such elements is at least $1/(e + 1)$; and so the proportion quantifying over all odd $e > d/2$ is at least $1/6$. Thus we expect to find such an element in a constant number of trials.

To prove that $g \in \mathrm{SL}(d, q)$ does not preserve a hermitian form, where $q = q_0^2$, it suffices to prove that the order of g is a multiple of a primitive prime divisor of $q_0^e - 1$ for some odd $e > d/2$. Note that a necessary condition for $g \in \mathrm{SL}(d, q)$ to be of order

a multiple of a primitive prime divisor of $q_0^e - 1$ or of $q^e - 1$ is that g should have an irreducible factor of degree e in its characteristic polynomial, since e is odd, and thus define a conjugacy class of e elements of $\mathrm{GF}(q^e)^\times$. We have seen in the proof of Lemma 6.3 that if g is uniformly distributed in $\mathrm{SL}(d,q)$ then the conjugacy classes of $\mathrm{GF}(q^e)^\times$ that arise are uniformly distributed amongst those classes not contained in $\mathrm{GF}(q^k)^\times$ for any proper subfield $\mathrm{GF}(q^k)$ of $\mathrm{GF}(q^e)$. As we are concerned with elements of order a multiple of a primitive prime divisor of $q_0^e - 1$, the conjugacy class of $\mathrm{GF}(q^e)^\times$ in question cannot fall into any proper subfield of $\mathrm{GF}(q^e)$ containing $\mathrm{GF}(q)$. Since the proportion of elements of $\mathrm{GF}(q^e)^\times$ whose orders are a multiple of a given primitive prime divisor ℓ of $q_0^e - 1$ is $1 - 1/\ell$, and the same holds for primitive prime divisors of $q^e - 1$, the ratio of the number of elements of $\mathrm{SL}(d,q)$ whose orders are a multiple of a primitive prime divisor of $q_0^e - 1$ to those whose orders are a multiple of a primitive prime divisor of $q^e - 1$ is greater than $2/3$. It follows again that examining a constant number of elements of $\mathrm{SL}(d,q)$ will find an appropriate ppd-element.

Consider now the exceptional generic case. In [25, Case 1, p. 159], the authors discuss how to distinguish subgroups of $Z \times \mathrm{PGL}(2,7)$ from $G = \mathrm{SL}(3,q)$ when $q = 3.2^s - 1$ for some $s \geq 2$. This they do by observing that $\mathrm{SL}(3,q)$ contains many elements of order a multiple of 8, and $Z \times \mathrm{PGL}(2,7)$ contains none. In this case G has a reasonable proportion of elements g of order a multiple of 16, and, since $q - 1$ is an odd multiple of 2, g^{q-1} has order a multiple of 8. This fails when $s = 2$ and $q = 11$. But $1/40$ of the elements of $\mathrm{GL}(3,11)$ have order a multiple of 8 and determinant 1, and so we search for such elements and do not power them.

If G is not generic, then the set of test elements is more elaborate. For example, in [27, Table 8, p. 248], the set of test elements for $\mathrm{SU}(3,3)$ and $\mathrm{SU}(3,5)$ include one of even order, and elements of the type required cannot be obtained from the corresponding general unitary groups by raising to the power 4 (in the first case) or 6 (in the second). These cases can be treated individually; alternatively, since $d \leq 6$, we can use the black-box algorithm of [32, Theorem 2.4.8] to construct the derived group in time $O(\log^2 q)$.

Niemeyer & Praeger [25] exclude the case $d = 2$. In this case, we can find by random search an element that powers up to an element g_1 of determinant 1 and order $q + 1$, take a conjugate g_2 of this element that does not commute with g_1. With high probability, these elements will generate $\mathrm{SL}(2,q)$, and can be found by considering at most $O(\log \log q)$ random elements. If $q = 5$, we take an additional element of order 5 to exclude the possibility that this pair of elements generates $2.A_4$.

We now consider the overall cost of the algorithm to construct the derived group.

Theorem 9.2 *Let G be a generic subgroup of $\mathrm{GX}(d,q)$. A generating set for the derived group of G can be constructed in Las Vegas time $O(\xi \log \log d + d^3 \log^2 d \log q)$ field operations.*

Proof: The algorithm of [25] has running time approximately $O(d^3 \log^2 d \log q)$ field operations. For a more precise statement of its complexity, see [30].

We need up to four exponentiations, the exponent being $q-1$ or $q+1$; its cost is estimated in Theorem 8.1.

At a cost of $O(d^3)$ field operations [18, Section 7.2], we compute a constant number of characteristic polynomials to exclude the possibility that we have constructed a group that preserves a form.

We must also construct the spaces W_1 and W_2 fixed by two elements, h_1 and h_2. Let h_1 have characteristic polynomial $f(x)$. Then $f(x) = u(x)w(x)$, where u is irreducible of degree e for some $e > d/2$, and W_1 is the image of $w(g)$. Let $w(x) = a_0 + a_1x + \cdots + x^{d-e}$, and compute $y := v.w(g)$ in $O(d^3)$ field operations. Then $y \in W_1$, and the probability that y is zero is $1 : q^{d-e}$. If y is non-zero, we "spin" y under h_1 to obtain a basis for W_1. The total Las Vegas time for computing W_1 is $O(d^3)$ [17]. The same applies to computing W_2. To determine whether or not $W_1 + W_2 = V$ costs $O(d^3)$ field operations. □

9.2 Decomposing a direct product

Assume we have a generating set for $G = \mathrm{SX}(e,q) \times \mathrm{SX}(d-e,q)$, and wish to construct generating sets for the direct factors. We assume that we have performed a base change on G and so can readily read off the projection of an element of G onto each factor.

We use essentially the same algorithm as that for constructing the derived group of $\mathrm{GX}(d,q)$. Namely, we construct an element of $\mathrm{SX}(e,q)$ by taking a random element (g_1,g_2) of G, and raise this element to the power n, where now n is the order of g_2. In general, a test element has an order that is a multiple of some prime, and we need to assess the probability that the order of g_2 will not be a multiple of this prime.

It follows from Theorem 5.7 of [25] that the probability of the relevant ppd-property of g_2 being destroyed by powering is less than $2/d$. This remains the case if we raise (g_1,g_2) to the power given by the *pseudo-order* of g_2, thus avoiding problems with integer factorisation. (See Section 10 for definition.)

In the non-generic case, the value of d is at most 6, and so we can use the following result of [4] in these cases.

Theorem 9.3 *Let C be a finite simple classical group, with natural module of dimension d. For a prime p, the proportion of p-regular elements of C is greater than $1/2d$.*

10 Constructing an involution centraliser

The centraliser of an involution in a black-box group having an order oracle can be constructed using an algorithm of Bray [5]. Elements of the centraliser are constructed using the following result.

Theorem 10.1 *If u is an involution in a group G, and g is an arbitrary element of G, then $[u,g]$ either has odd order $2k+1$, in which case $g[u,g]^k$ commutes with u, or has even order $2k$, in which case both $[u,g]^k$ and $[u,g^{-1}]^k$ commute with u.*

That these elements centralise u follows from elementary properties of dihedral groups.

Bray [5] also proves that if g is uniformly distributed among the elements of G for which $[u, g]$ has odd order, then $g[u, g]^k$ is uniformly distributed among the elements of the centraliser of u. If $[u, g]$ has even even, then the elements returned are involutions; but if just one of these is selected, then it is independently and uniformly distributed within that class of involutions.

Let $u \in \mathrm{SL}(d, q)$ and let E_+ and E_- denote the eigenspaces of u. We apply the Bray algorithm in the following contexts.

1. Construct a generating set for a subgroup of the centraliser of u that contains $\mathrm{SL}(E_+) \times \mathrm{SL}(E_-)$.

2. The eigenspaces, E_+ and E_-, have the same dimension. Construct the projective centraliser of u. As we observed in Section 2, its preimage in G contains an element which interchanges the eigenspaces.

The other contexts are similar, but with $\mathrm{SL}(d, q)$ replaced by the other classical groups.

Parker & Wilson [31] prove the following:

Theorem 10.2 *There is an absolute constant c such that if G is a finite quasisimple classical group, with natural module of dimension d over a field of odd order, and u is an involution in G, then $[u, g]$ has odd order for at least a proportion c/d of the elements g of G.*

Hence, by a random search of length at most $O(d)$, we construct random elements of the centraliser of the involution. Liebeck & Shalev [21] prove that if $H_0 \leq H \leq \mathrm{Aut}(H_0)$, where H_0 is a finite simple group, then the probability that two random elements of G generate a group containing H_0 tends to 1 as $|H_0|$ tends to infinity. A similar result clearly holds for a direct product of two simple groups.

In its black-box application, this algorithm assumes the existence of an order oracle. We do not require such an oracle for a linear group. If a multiplicative upper-bound B for the order of $g \in G$ is available, then we can learn in polynomial time the *exact* power of 2 (or of any specified prime) which divides $|g|$. By repeated division by 2, we write $B = 2^m b$ where b is odd. Now we compute $h = g^b$, and determine its order which divides 2^m by repeated squaring. If $g \in \mathrm{GL}(d, q)$, then a multiplicative upper bound of magnitude $O(q^d)$ can be obtained for $|g|$ using the algorithms of [10] and [33] in at most $O(d^3 \log q)$ field operations. We call this upper bound the *pseudo-order* of g. Further, as discussed in [19], the construction of the centraliser of an involution requires only knowledge of the pseudo-order.

In summary, [19, Theorem 7] implies the following.

Theorem 10.3 *The Bray algorithm to construct the centraliser of an involution in $\mathrm{SX}(d, q)$ has complexity $O(d(\xi + d^3 \log q))$ field operations.*

This algorithm can be readily adapted (using projective rather than linear pseudo-orders) to compute the preimage in $\mathrm{SX}(d,q)$ of the centraliser of an involution in $\mathrm{PX}(d,q)$.

Once we construct a subgroup of the centraliser containing its derived group, we can apply the algorithms of Section 9 to obtain its projection onto each factor and then obtain the derived group of each projection. Now we can use the algorithm of [25] to deduce that the projection contains a perfect classical group in its natural representation. If the factors have the same dimension, there is a small possibility that the given elements generate a group that contains a diagonal embedding of $\mathrm{SX}(d/2,q)$ in $\mathrm{SX}(d/2,q) \times \mathrm{SX}(d/2,q)$ but does not contain the full direct product. This case is easily detected. We can also readily detect when an element of the centraliser interchanges the eigenspaces.

We summarise the preceding discussion.

Theorem 10.4 *Let h be an involution in $\langle X \rangle = G$, where $\mathrm{SX}(d,q) \leq G\Gamma\mathrm{X}(d,q)$. Assume that the -1-eigenspace of h has dimension e in the range $(d/3, 2d/3]$. Generating sets for the images in $\mathrm{SX}(e,q)$ and $\mathrm{SX}(d-e,q)$ that centralise the eigenspaces can be found in $O(d(\xi + d^3 \log q))$ field operations. If $e = d/2$ so that $d \equiv 0 \bmod 4$, then we can similarly find an element in $\mathrm{SX}(d,q) \wr C_2$ which interchanges the two copies of $\mathrm{SX}(d,q)$.*

11 The base cases

We now consider the base cases for Algorithms `One` and `Two`. Recall that, for $d = 2n$, $Y_0 = \{s, t, \delta, u, v\}$ generates $\mathrm{SX}(2,q) \wr C_n$ or $\mathrm{SX}(2,q) \wr S_n$. This observation influences the organisation of our algorithms and particularly impacts on our handling of the base cases. As the first and major part of each algorithm, we construct Y_0. Then, as a final step, we construct the additional elements x, y. Clearly the elements of Y_0 could be constructed by constructively recognising $\mathrm{SX}(4,q)$; however, both u and v can be obtained by constructively recognising $\mathrm{SX}(2,q) \wr C_2$, a computation practically easier than that for $\mathrm{SX}(4,q)$. Hence we designate the following as base cases: $\mathrm{SX}(2,q)$, $\mathrm{SX}(2,q) \wr C_2$, $\mathrm{SX}(3,q)$ and $\mathrm{SX}(4,q)$. The last two arise *at most once* during an application of Algorithm `One` or `Two`.

The construction of a hyperbolic basis for a vector space with a given symplectic or hermitian form, can be carried out in $O(d^3)$ field operations [16, Chapter 2]. This calculation is performed for base cases only.

In the remainder of this section, we outline the specialised algorithms for the base cases. We first summarise their cost.

Theorem 11.1 *Subject to the availability of a discrete log oracle for $\mathrm{GF}(q)$, SLPs for standard generators and other elements of $\mathrm{SX}(d,q)$ for $d \leq 4$ can be constructed in $O(\xi \log\log q + \log q)$ field operations.*

11.1 $\mathrm{SL}(2,q)$

The base case encountered most frequently is $\mathrm{SL}(2,q)$ in its natural representation. An algorithm to construct an element of $\mathrm{SL}(2,q)$ as an SLP in an arbitrary generating set is described in [12]. This algorithm requires $O(\log q)$ field operations, and the availability of discrete logarithms in $\mathrm{GF}(q)$. Observe that $\mathrm{SU}(2,q)$ is isomorphic to $\mathrm{SL}(2,q)$ and can be written over $\mathrm{GF}(q)$ using the algorithm of [14] in $O(\log q)$ operations.

11.2 $\mathrm{SL}(2,q) \wr C_2$

In executing Algorithms `OneEven` or `OneOdd`, or `TwoTimesFour` or `TwiceTwiceOdd`, each pair of recursive calls generates an instance of the following problem.

Problem 11.2 *Let V be the natural module of $G = \mathrm{SX}(4,q)$, and let (e_1, f_1, e_2, f_2) be a hyperbolic basis for V. Given a generating set for X, and the involution u, where u maps e_1 to $-e_1$ and f_1 to $-f_1$, and centralises the other basis elements, construct the involution b that permutes the basis elements, interchanging e_1 with e_2, and f_1 with f_2.*

Consider the procedure `OneEven`. Observe that in l. 15 we construct $\mathrm{SX}(4,q)$. Now b is the permutation matrix used in l. 16 to "glue" v_1 and v_2 together to form v, the long cycle. We could use the algorithm of Section 11.3 to find b directly in $\mathrm{SX}(4,q)$. Instead, for reasons of practical efficiency, we use the following algorithm to find b inside the projective centraliser of $u \in \mathrm{SX}(4,q)$.

1. Construct the projective centraliser H of u in $\mathrm{SX}(4,q)$, using the Bray algorithm.
2. Since $\mathrm{SL}(2,q) \wr C_2 \leq H \leq \mathrm{GL}(2,q) \wr C_2$, we find $h \in \mathrm{SL}(2,q) \wr C_2$ that interchanges the spaces $\langle e_1, f_1 \rangle$ and $\langle e_2, f_2 \rangle$.
3. Now bh lies in $\mathrm{SL}(2,q) \times \mathrm{SL}(2,q)$. Using the algorithms described in Section 9, we construct the two direct factors, solve in each direct factor for the projection of jh and so construct bh as an SLP. We can now solve for b.

This algorithm requires $O(\log q)$ field operations. Observe that we can conjugate, using h, the solution from one copy of $\mathrm{SL}(2,q)$ to the other, thus requiring just one constructive recognition of $\mathrm{SL}(2,q)$.

11.3 $\mathrm{SX}(3,q)$ and $\mathrm{SX}(4,q)$

We use the involution-centraliser algorithm of [19] to construct standard generators for $\mathrm{SX}(3,q)$, and the additional elements $x, y \in \mathrm{SX}(4,q)$.

We briefly summarise this algorithm. Assume $G = \langle X \rangle$ is a black-box group with order oracle. We are given $g \in G$ to be expressed as an SLP in X. In our description, if we "find" an element g of G, then we obtain its SLP in X. First find by random search $h \in G$ such that gh has even order 2ℓ, and $z := (gh)^\ell$ is a non-central involution.

Now find, by random search and powering, an involution $x \in G$ such that xz has even order $2m$, and $y := (xz)^m$ is a non-central involution. Note that an SLP is known for x, but, at this stage, not for either of y nor z. Observe that x, y and z are non-central involutions. We construct their centralisers using the Bray algorithm. We assume that we can solve the explicit membership problem in these centralisers. In particular, we find y as an element of the centraliser in G of x, and z as an element of the centraliser in G of y, and gh as an element of the centraliser in G of z. Now that we know an SLP for gh and h, we can write down an SLP for g.

In summary, this algorithm reduces the constructive membership test for G to three constructive membership tests in involution centralisers in G. But this is an imperfect recursion, since the algorithm may not be applicable to these centralisers. We do not rely on the recursion; instead we construct explicitly the desired elements of the centralisers, since their derived groups are (direct products of) $\mathrm{SL}(2, q)$ and we can use the algorithm of [12]. In this context, the complexity of the involution-centraliser algorithm is that stated in Theorem 11.1.

As presented, this is a black-box algorithm requiring an order oracle. If G is a linear group, the algorithm does not require an order oracle, exploiting instead the multiplicative bound for the order of an element which can be obtained in polynomial time as described in Section 10.

Since the practical performance of this algorithm is rather slow for large fields, we organised Algorithm `One` and `Two` to ensure that they each need *at most one application*. If the dimension d of the input group is odd, then we invoke this algorithm once to construct standard generators for $\mathrm{SX}(3, q)$. If d is even, then as a final step, we construct the additional generators x and y using this algorithm. Let $h \in G = \mathrm{SX}(d, q)$ be the involution whose -1-eigenspace is $\langle e_1, f_1, e_2, f_2 \rangle$. Observe that h can be readily constructed from Y_0, and that both x and y are elements of $C_G(h)$.

12 Complexity of the algorithms

We now analyse the principal algorithms, and in the next section estimate the length of the SLPs that express the canonical generators as words in the given generators. The time analysis is based on counting the number of field operations, and the number of calls to the discrete logarithm oracles. Use of discrete logarithms in a given field requires first the setting up of certain tables, and these tables are consulted for each application. The time spent in the discrete logarithm algorithm, and the space that it requires, are not proportional to the number of applications in a given field.

Babai [1] presented a Monte Carlo algorithm to construct in polynomial time nearly uniformly distributed random elements of a finite group. An alternative is the *product replacement algorithm* of Celler *et al.* [9]. That this is also polynomial time was established by Pak [29]. For a discussion of both algorithms, we refer the reader to [32, pp. 26-30].

We now complete our analysis of the main algorithms.

Theorem 12.1 *The number of field operations carried out in Algorithm* `OneEven` *is* $O(d(\xi + d^3 \log q)$.

PROOF: The proportion of elements of G with the required property in line 6 is at least k/d for some absolute constant k, as proved in Section 6.

The number of field operations required in lines 8 and 14 is $O(d(\xi + d^3 \log q))$, as proved in Section 10.

The recursive calls in lines 10 and 11 are to cases of dimension at most $2d/3$, and hence they increase by only a constant factor the number of field operations.

The number of field operations required in lines 9 and 13 is at most $O(d^3 \log q)$, as proved in Section 9.

The result follows. □

We estimate the number of calls to the $\mathrm{SL}(2, q)$ constructive recognition algorithm and the associated discrete logarithm oracle.

Theorem 12.2 *If* $d > 2$, *then Algorithms* `OneEven` *and* `TwoTimesFour` *generate at most* $2d - 3$ *and* $3 \log d$ *calls to the discrete logarithm oracle for* $\mathrm{GF}(q)$ *respectively.*

PROOF: Each call to the constructive recognition oracle for $\mathrm{SL}(2, q)$ generates three calls to the discrete logarithm oracle for $\mathrm{GF}(q)$ [12]. Each solution to Problem 11.2 requires 3 calls to the oracle.

Let $f(d)$ be the number of calls generated by applying `OneEven` to $\mathrm{SX}(d, q)$. Then $f(2) = f(4) = 3$ and $f(d) = f(e) + f(d - e) + 3$ for $d > 4$ and some $e \in (d/3, 2d/3]$. It follows that $f(d) \leq 2d - 3$ for $d > 2$.

Let $g(d)$ be the number of calls generated by applying `TwoTimesFour` to $\mathrm{SX}(d, q)$, where $d = 4n$. Again $g(2) = g(4) = 3$ and $g(4n) = g(2n) + 3$ for $n > 2$. Hence $g(d) \leq 3 \log d$. □

Similar results hold for the other algorithms. If we use the involution-centraliser algorithm [19] to construct either standard generators for $\mathrm{SX}(3, q)$, or additional generators $x, y \in \mathrm{SX}(4, q)$, then the number of calls to the oracle in each case is 9.

13 Straight-line programs

We now consider the length of the SLPs for the standard generators for $\mathrm{SX}(d, q)$ constructed by our algorithms.

In its simplest form, an SLP on a subset X of a group G is a string, each of whose entries is either a pointer to an element of X, or a pointer to a previous entry of the string, or an ordered pair of pointers to not necessarily distinct previous entries. Every entry of the string defines an element of G. An entry that points to an element of X defines that element. An entry that points to a previous entry defines the inverse of the element defined by that entry. An entry that points to two previous entries defines the product, in that order, of the elements defined by those entries.

Such a simple SLP defines an element of G, namely the element defined by the last entry, and it can be obtained by computing in turn the elements for successive entries. The SLP is primarily used by replacing the elements X of G by the elements Y of some group H, where X and Y are in one-to-one correspondence, and then evaluating the element of H that the SLP then defines.

We now identify other desirable features of SLPs.

1. We need to replace the second type of node, that defines the inverse of a previously defined element, by a type of node with two fields, one pointing to a previous entry, and one containing a possibly negative integer. The element defined is then the element defined by the entry to which the former field points, raised to the power defined by the latter field. This reflects the fact that we raise group elements to very large powers, and have an efficient algorithm described in Section 8 for performing this.

2. An SLP may define a number of elements of G, and not just one element, so a sequence of nodes may be specified as giving rise to elements of G. Thus we wish to return a single SLP that defines all of the standard generators of $\mathrm{SX}(d, q)$, rather than an SLP for each generator. This avoids duplication when two or more of the standard generators rely on common calculations.

3. A critical concern is how the number of trials in a random search for a group element affects the length of an SLP that defines that element. Any discussion of this requires consideration of the algorithm used to generate random elements. We make two reasonable assumptions:

   (a) the associated random process is a stochastic process taking place in a graph whose vertices are defined by a *seed*;

   (b) a random number generator now determines which edge adjoining the current vertex in the graph will be followed in the stochastic process.

   By default, the length of the SLP will then increase by a constant amount for *every trial, successful or unsuccessful*. Should its length reflect only those trials which are successful? One additional assumption which allows us to explore this question is the following: *When embarking on a search that is expected to require d trials, we record the value of the seed, and repeatedly carry out a random search, using our random process, but returning, after every $\ell(d)$ steps, for some function ℓ of d, to the stored value of the seed, until we succeed.* We hypothesise that values for $\ell(d)$ range from $\log d$ to d and analyse the lengths of the SLPs for these values below.

Theorem 13.1 *If the* SLPs *constructed satisfy properties* $1-3$ *above, then their lengths are the following.*

$\ell(d)$	`OneMain`	`TwoMain`
$\log d$	$O(d)$	$O(\log^3 d)$
d	$O(d \log d)$	$O(d \log d)$

PROOF: We wish to find functions $f(d)$ and $g(d)$ such that the lengths of the SLPs returned by Algorithms `One` and `Two` are bounded above by these functions respectively.

It suffices for f to satisfy $f(d) \geq f(e) + f(d-e) + c\ell(d)$ whenever $d \geq 5$ for some constant c, if $f(d)$ is large enough for small values of d. Since of necessity $f(d) > f(e) + f(d-e)$ it follows that $f(d)$ is at least linear in d.

Consider, for example, the case where $\ell(d) = d$. Suppose that we take a constant k such that $k > c/\log(3/2)$, taking all logarithms to base 2. Suppose now that $f(n) < kn\log(n)$ for all $n < d$ for some $d > 4$, and let $e \in (d/3, 2d/3]$. Then

$$\begin{aligned} f(e) + f(d-e) + cd &< ke\log(e) + k(d-e)\log(d-e) + cd \\ &< kd\log(d) - kd\log(3/2) + cd \\ &< kd\log(d), \end{aligned}$$

as required.

Algorithm `Two` recurses either from the case $d = 4n$ to the case $d = 2n$ in one step, or from the case $d = 4n+2$ to the case $d = 4n$ and then to the case $d = 2n$. It is easy to see that the effect on the length of the SLP in the latter situation is dominated by the second step. If d is initially odd, then the contribution of the reduction to the even case, which is carried out once, may also be ignored here. The main contribution to the length of the SLP in passing from $d = 4n$ to $d = 2n$ arises from constructing an involution whose eigenspaces have dimension $2n$. This involution is constructed recursively, where the length of the recursion is $O(\log d)$. Thus the contribution to the length of the SLP in constructing this involution is $O(\log(d)\ell(d))$. Hence $g(4n) \leq g(2n) + c\log(n)\ell(n)$ and $g(4n+2) \leq g(2n) + c\log(n)\ell(n)$. If $\ell(d) = O(\log d)$, then the inequality $g(n) \leq g(\lceil n/2 \rceil) + c\log^2(n)$ is satisfied by $g(n) = k\log^3(n)$ for large enough k.

Similar calculations can be carried for the other two cases, yielding the stated results. □

14 An implementation

Our implementation of these algorithms is publicly available in MAGMA. It uses:

- the product replacement algorithm [9] to generate random elements;
- our implementations of Bray's algorithm [5] and the involution-centraliser algorithm [19].
- our implementations of the algorithm of [12] and [22].

The computations reported in Table 2 were carried out using MAGMA V2.13 on a Pentium IV 2.8 GHz processor. The input to the algorithm is $\mathrm{SX}(d, q)$. In the column entitled "Time", we list the CPU time in seconds taken to construct the standard generators.

Table 2: Performance of implementation for a sample of groups

Input	Time
$\mathrm{SL}(6, 5^8)$	2.2
$\mathrm{SL}(40, 5^8)$	22.5
$\mathrm{SL}(80, 5^8)$	130.8
$\mathrm{Sp}(10, 5^{10})$	19.5
$\mathrm{Sp}(40, 5^{10})$	280.4
$\mathrm{SU}(8, 3^{16})$	22.6
$\mathrm{SU}(20, 5^{12})$	47.6
$\mathrm{SU}(70, 5^2)$	191.3

References

[1] László Babai, Local expansion of vertex-transitive graphs and random generation in finite groups, *Theory of Computing*, (Los Angeles, 1991), pp. 164–174. Association for Computing Machinery, New York, 1991.

[2] László Babai and Endre Szemerédi. On the complexity of matrix group problems, I. In *Proc. 25th IEEE Sympos. Foundations Comp. Sci.*, pages 229–240, 1984.

[3] László Babai and Robert Beals, A polynomial-time theory of black-box groups. I. In *Groups St. Andrews 1997 in Bath, I*, volume 260 of *London Math. Soc. Lecture Note Ser.*, pages 30–64, Cambridge, 1999. Cambridge Univ. Press.

[4] L. Babai, P. Pálfy and J. Saxl, On the number of p-regular elements in simple groups, preprint.

[5] J.N. Bray, An improved method of finding the centralizer of an involution, *Arch. Math. (Basel)* **74** (2000), 241–245.

[6] Wieb Bosma, John Cannon, and Catherine Playoust, The MAGMA algebra system I: The user language, *J. Symbolic Comput.*, **24**, 235–265, 1997.

[7] P.A. Brooksbank, Constructive recognition of classical groups in their natural representation. *J. Symbolic Comput.* **35** (2003), 195–239.

[8] Roger Carter. Simple groups of Lie Type. Wiley-Interscience, 1989.

[9] Frank Celler, Charles R. Leedham-Green, Scott H. Murray, Alice C. Niemeyer and E.A. O'Brien, Generating random elements of a finite group, *Comm. Algebra*, **23** (1995), 4931–4948.

[10] Frank Celler and C.R. Leedham-Green, Calculating the order of an invertible matrix, In *Groups and Computation II*, volume 28 of *Amer. Math. Soc. DIMACS Series*, pages 55–60. (DIMACS, 1995), 1997.

[11] F. Celler and C.R. Leedham-Green. A constructive recognition algorithm for the special linear group. In *The atlas of finite groups: ten years on (Birmingham, 1995)*, volume 249 of *London Math. Soc. Lecture Note Ser.*, pages 11–26, Cambridge, 1998. Cambridge Univ. Press.

[12] M.D.E. Conder, C.R. Leedham-Green, and E.A. O'Brien. Constructive recognition of PSL$(2, q)$. *Trans. Amer. Math. Soc.* **358**, 1203–1221, 2006.

[13] Jason Fulman, Peter M. Neumann, and Cheryl E. Praeger. A Generating Function Approach to the Enumeration of Matrices in Classical Groups over Finite Fields. Mem. Amer. Math. Soc. **176**, no. 830, 2005.

[14] S.P. Glasby, C.R. Leedham-Green, and E.A. O'Brien. Writing projective representations over subfields. *J. Algebra*, 295, 51-61, 2006.

[15] Daniel Gorenstein, Richard Lyons, and Ronald Solomon. The classification of the finite simple groups. Number 3. Part I, American Mathematical Society, Providence, RI, 1998.

[16] Larry C. Grove. Classical Groups and Geometric Algebra. AMS Graduate Studies in Math. **39**.

[17] D.F. Holt and S. Rees. Testing modules for irreducibility. *J. Austral. Math. Soc. Ser. A* **57** (1994), 1–16.

[18] Derek F. Holt, Bettina Eick, and Eamonn A. O'Brien. *Handbook of computational group theory*. Chapman and Hall/CRC, London, 2005.

[19] P.E. Holmes, S.A. Linton, E.A. O'Brien, A.J.E. Ryba and R.A. Wilson, Constructive membership in black-box groups, preprint.

[20] William M. Kantor and Ákos Seress. Black box classical groups. *Mem. Amer. Math. Soc.*, **149**, 2001.

[21] M.W. Liebeck and A. Shalev. The probability of generating a finite simple group, *Geom. Ded.* **56** (1995), 103–113.

[22] F. Lübeck, K. Magaard, and E.A. O'Brien. Constructive recognition of $\mathrm{SL}_3(q)$. Preprint 2005.

[23] B. Huppert. *Endliche Gruppen I*, volume 134 of *Grundlehren Math. Wiss.* Springer-Verlag, Berlin, Heidelberg, New York, 1967.

[24] Peter M. Neumann and Cheryl E. Praeger. A recognition algorithm for special linear groups. *Proc. London Math. Soc.* (3), 65:555–603, 1992.

[25] A.C. Niemeyer and C.E. Praeger. A recognition algorithm for classical groups over finite fields, *Proc. London Math. Soc.* **77** (1998), 117–169.

[26] Alice C. Niemeyer and Cheryl E. Praeger. Implementing a recognition algorithm for classical groups. In *Groups and Computation, II (New Brunswick, NJ, 1995)*, volume 28 of *DIMACS Ser. Discrete Math. Theoret. Comput. Sci.*, pages 273–296, Providence, RI, 1997. Amer. Math. Soc.

[27] Alice C.Niemeyer and Cheryl E.Praeger. A recognition algorithm for non-generic classical groups over finite fields. J. Austral. Math. Soc. Ser. A **67** (1999), no. 2, 223–253.

[28] E.A. O'Brien. Towards effective algorithms for linear groups. *Finite Geometries, Groups and Computation*, (Colorado), pp. 163-190. De Gruyter, Berlin, 2006.

[29] Igor Pak. The product replacement algorithm is polynomial. In *41st Annual Symposium on Foundations of Computer Science (Redondo Beach, CA, 2000)*, 476–485, IEEE Comput. Soc. Press, Los Alamitos, CA, 2000.

[30] Cheryl E. Praeger. Primitive prime divisor elements in finite classical groups. In *Groups St. Andrews 1997 in Bath, II*, 605–623, Cambridge Univ. Press, Cambridge, 1999.

[31] C.W. Parker and R.A. Wilson. Recognising simplicity in black-box groups. Preprint 2005.

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

[33] Arne Storjohann. An $O(n^3)$ algorithm for the Frobenius normal form. In *Proceedings of the 1998 International Symposium on Symbolic and Algebraic Computation* (Rostock), 101–104, ACM, New York, 1998.

[34] Joachim von zur Gathen and Jürgen Gerhard, *Modern Computer Algebra*, Cambridge University Press, 2002.

School of Mathematical Sciences
Queen Mary, University of London
London E1 4NS, United Kingdom
United Kingdom
C.R.Leedham-Green@qmul.ac.uk

Department of Mathematics
Private Bag 92019, Auckland
University of Auckland
New Zealand
obrien@math.auckland.ac.nz

Last revised December 8, 2007