ALGORITHMS FOR MATRIX GROUPS

E.A. O'BRIEN

Department of Mathematics, University of Auckland, Auckland, New Zealand
Email: obrien@math.auckland.ac.nz

Abstract

Existing algorithms have only limited ability to answer structural questions about subgroups G of $\mathrm{GL}(d, F)$, where F is a finite field. We discuss new and promising algorithmic approaches, both theoretical and practical, which as a first step construct a chief series for G.

1 Introduction

Research in Computational Group Theory has concentrated on four primary areas: permutation groups, finitely-presented groups, soluble groups, and matrix groups. It is now possible to study the structure of permutation groups having degrees up to about ten million; Seress [97] describes in detail the relevant algorithms. We can compute *useful* descriptions for quotients of finitely-presented groups; as one example, O'Brien & Vaughan-Lee [90] computed a power-conjugate presentation for the largest finite 2-generator group of exponent 7, showing that it has order 7^{20416}. Practical algorithms for the study of polycyclic groups are described in [59, Chapter 8].

We contrast the success in these areas with the paucity of algorithms to investigate the structure of matrix groups. Let $G = \langle X \rangle \leq \mathrm{GL}(d, F)$ where $F = \mathrm{GF}(q)$. Natural questions of interest to group-theorists include: What is the order of G? What are its composition factors? How many conjugacy classes of elements does it have? Such questions about a subgroup of S_n, the symmetric group of degree n, are answered both theoretically and practically using highly effective polynomial-time algorithms. However, for linear groups these can be answered only in certain limited contexts. As one indicator, it is difficult (using standard functions) to answer such questions about $\mathrm{GL}(8, 7)$ using either of the major computational algebra systems, GAP [46] and MAGMA [16].

A major topic of research over the past 15 years, the so-called "matrix recognition" project, has sought to address these limitations by developing effective well-understood algorithms for the study of such groups. A secondary goal is to realise the performance of these algorithms in practice, via publicly available implementations.

Thanks to Peter Brooksbank, Heiko Dietrich, Stephen Glasby, Derek Holt, Colva Roney-Dougal and Ákos Seress for their comments and corrections to the paper. This work was partially supported by the Marsden Fund of New Zealand via grant UOA721.

Cambridge University Press & Assessment
978-0-521-27904-8 — Groups St Andrews 2009 in Bath
C. M. Campbell , M. R. Quick , E. F. Robertson , C. M. Roney-Dougal , G. C. Smith , G. Traustason
Excerpt
More Information

Two approaches dominate. The *black-box approach*, discussed in Section 4, aims to construct a characteristic series $\mathcal{C}$ of subgroups for G which can be readily refined to provide a chief series; the associated algorithms are independent of the given representation. The *geometric approach*, discussed in Section 5, aims to exploit the natural linear action of G on its underlying vector space to construct a composition series for G; the associated algorithms exploit the linear representation of G. Both approaches rely on the solution of certain key tasks for simple groups which we discuss in Section 3; we survey their solutions in Sections 6–9. Presentations for the groups of Lie type on certain *standard generators* are used to ensure correctness; these are discussed in Section 10.

As we demonstrate in Section 11, the geometric approach is realised via a *composition tree*. In practice, the composition series produced from the geometric approach is readily modified to produce a chief series of G exhibiting $\mathcal{C}$. In Section 12 we consider briefly algorithms which exploit the chief series and its associated *Trivial Fitting* paradigm to answer structural questions about G. While it is not yet possible to make definitive statements about the outcome of this project, a realistic and achievable goal is to provide algorithms to answer many questions for linear groups of "small" degree, say up to degree 20 defined over moderate-sized fields.

In this paper, we aim to supplement and update the related surveys [65], [72] and [91]. Its length precludes comprehensiveness. For example, we consider neither nilpotent nor solvable linear groups. Nor do we discuss the algorithms of Detinko and Flannery and others to study finitely generated matrix groups defined over infinite fields. The excellent survey [43] addresses both omissions.

2 Basic concepts

We commence with a review of basic concepts.

2.1 Complexity

If f and g are real-valued functions defined on the positive integers, then $f(n) = O(g(n))$ means $|f(n)| < C|g(n)|$ for some positive constant C and all sufficiently large n.

One measure of performance is that an algorithm is *polynomial in the size of the input*. If $G = \langle X \rangle \leq \mathrm{GL}(d, q)$, then the size of the input is $|X|d^2 \log q$, since each of the d^2 entries in a matrix requires $\log q$ bits.

2.2 Black-box groups

The concept of a *black-box group* was introduced in [6]. In this model, group elements are represented by bit-strings of uniform length; the only group operations permissible are multiplication, inversion, and checking for equality with the identity element. Permutation groups and matrix groups defined over finite fields are covered by this model.

Seress [97, p. 17] defines a *black-box algorithm* as one which does not use specific features of the group representation, nor particulars of how group operations are performed; it can only use the operations listed above. However, a common assumption is that *oracles* are available to perform certain tasks – usually those not known to be solvable in polynomial time.

One such is a *discrete log oracle*: for a given non-zero $\mu \in \mathrm{GF}(q)$ and a fixed primitive element ω of $\mathrm{GF}(q)$, it returns the unique integer k in the range $1 \leq k < q$ for which $\mu = \omega^k$. The most efficient algorithms for this task run in sub-exponential time (see [98, Chapter 4]).

If the elements of a black-box group G are represented by bit-strings of uniform length n, then n is the *encoding length* of G and $|G| \leq 2^n$. If G is described by a bounded list of generators, then the size of the input to a black-box algorithm is $O(n)$. If G also has Lie rank r and is defined over a field of size q, then $|G| \geq (q-1)^r$, so both r and $\log q$ are $O(n)$.

2.3 Algorithm types and random elements

Most algorithms for linear groups are *randomised*: they rely on random selections. A *Monte Carlo* algorithm is a randomised algorithm that, with prescribed probability less than 1/2, may return an incorrect answer to a decision question. A *Las Vegas* algorithm is one that never returns an incorrect answer, but may report failure with probability less than some specified $\epsilon \in (0, 1)$. At the cost of n iterations, the probability of a correct answer can be increased to $1 - \epsilon^n$. We refer the reader to [5] for a discussion of these concepts.

Monte Carlo algorithms to construct the normal closure of a subgroup and the derived group of a black-box group are described in [97, Chapter 2].

Many algorithms use random search in a group $G \leq \mathrm{GL}(d, q)$ to find elements having prescribed property $\mathcal{P}$. Examples of $\mathcal{P}$ are having a characteristic polynomial with a factor of degree greater than $d/2$, or order divisible by a prescribed prime.

A common feature is that these algorithms depend on detailed analysis of the *proportion* of elements of finite simple groups satisfying $\mathcal{P}$. Assume we determine a lower bound, say $1/k$, for the proportion of elements in G satisfying $\mathcal{P}$. To find an element satisfying $\mathcal{P}$ by random search with probability of failure less than a given $\epsilon \in (0, 1)$, we choose a sample of uniformly distributed random elements in G of size at least $\lceil \log_e(1/\epsilon) \rceil k$.

Following [97, p. 24], an algorithm constructs an ϵ-uniformly distributed random element x of a finite group G if $(1-\epsilon)/|G| < \mathrm{Prob}(x = g) < (1+\epsilon)/|G|$ for all $g \in G$; if $\epsilon < 1/2$, then the algorithm constructs *nearly uniformly distributed* random elements of G. Babai [4] presents a black-box Monte Carlo algorithm to construct such elements in polynomial time. An alternative is the *product replacement algorithm* of Celler *et al.* [34]. That this runs in polynomial time was established by Pak [92]. Its implementations in GAP and Magma are widely used. For a discussion of both algorithms, see [97, pp. 26–30]. Another algorithm, proposed by Cooperman [39], was analysed by Dixon [44].

2.4 Some basic operations

Consider the task of multiplying two $d \times d$ matrices. Its complexity is $O(d^\omega)$ field operations, where $\omega = 3$ if we employ the traditional algorithm. Strassen's divide-and-conquer algorithm [100] reduces ω to $\log_2 7$ but at a cost: namely, the additional intricacy of an implementation and larger memory demands. Coppersmith & Winograd's result [40] that ω can be smaller than 2.376 remains of limited practical significance.

We can compute large powers m of a matrix g in at most $2\,\lfloor \log_2 m \rfloor$ multiplications by the standard doubling algorithm: $g^m = g^{m-1}g$ if m is odd and $g^m = g^{(m/2)2}$ if m is even.

Lemma 2.1

(i) *Multiplication and division operations for polynomials of degree d defined over* $\mathrm{GF}(q)$ *can be performed deterministically in $O(d \log d \log\log d)$ field operations. Using a Las Vegas algorithm, such a polynomial can be factored into its irreducible factors in $O(d^2 \log d \log\log d \log(qd))$ field operations.*

(ii) *Using Las Vegas algorithms, both the characteristic and minimal polynomial of $g \in \mathrm{GL}(d, q)$ can be computed in $O(d^3 \log d)$ field operations.*

For the cost of polynomial operations, see [101, §8.3, §9.1, Theorem 14.14]. Characteristic and minimal polynomials can be computed in the claimed time using the Las Vegas algorithms of [2, 69] and [47] respectively. Neunhöffer & Praeger [87] describe Monte Carlo and deterministic algorithms to construct the minimal polynomial; these have complexity $O(d^3)$ and $O(d^4)$ respectively and are implemented in GAP.

2.5 The pseudo-order of a matrix

To determine the order of $g \in \mathrm{GL}(d, q)$ currently requires factorisation of numbers of the form $q^i - 1$, a problem generally believed not to be solvable in polynomial time. Since $\mathrm{GL}(d, q)$ has elements of order $q^d - 1$ (namely, Singer cycles), it is not practical to compute powers of g until we obtain the identity.

Celler & Leedham-Green [35] present the following algorithm to compute the order of $g \in \mathrm{GL}(d, q)$.

- Compute a "good" multiplicative upper bound B for $|g|$.
- Factorise $B = \prod_{i=1}^{m} p_i^{\alpha_i}$ where the primes p_i are distinct.
- If $m = 1$, then calculate $g^{p_1^j}$ for $j = 1, 2, \ldots, \alpha_1 - 1$ until the identity is constructed.
- If $m > 1$ then express $B = uv$, where u, v are coprime and have approximately the same number of distinct prime factors. Now g^u has order k dividing v and g^k has order ℓ say dividing u, and the order of g is $k\ell$. Hence the algorithm proceeds by recursion on m.

They prove the following:

Theorem 2.2 *If we know a factorisation of B, then the cost of the algorithm is $O(d^4 \log q \log\log q^d)$ field operations.*

We can readily compute in polynomial time a "good" multiplicative upper bound for $|g|$. Let the factorisation over $\mathrm{GF}(q)$ of the minimal polynomial $f(x)$ of g into powers of distinct irreducible monic polynomials be given by $f(x) = \prod_{i=1}^{t} f_i(x)^{n_i}$, where $\deg(f_i) = e_i$. Then $|g|$ divides $B := \mathrm{lcm}(q^{e_1} - 1, \ldots, q^{e_t} - 1) \times p^{\beta}$, where $\beta = \lceil \log_p \max n_i \rceil$ and $\mathrm{GF}(q)$ has characteristic p.

The GAP and Magma implementations of the order algorithm are very efficient, and use databases of factorisations of numbers of the form $q^i - 1$, prepared as part of the Cunningham Project [20].

From B, 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 (by powering) its order, which divides 2^m. In particular, we can deduce if g has *even order*.

For most applications, it suffices to know the *pseudo-order* of $g \in \mathrm{GL}(d, q)$, a refined version of B. Leedham-Green & O'Brien [73, Section 2] define this formally and show that it can be computed in $O(d^3 \log d + d^2 \log d \log\log d \log q)$ field operations.

2.6 Straight-line programs

One may intuitively think of a *straight-line program* (SLP) for $g \in G = \langle X \rangle$ as an efficiently stored word in X that evaluates to g; for a formal definition and discussion of their significance, see [97, p. 10]. While the length of a word in a given generating set constructed in n multiplications and inversions can increase exponentially with n, the length of the corresponding SLP is *linear* in n. Babai & Szemerédi [6] prove that every element of a finite group G has an SLP of length $O(\log^2 |G|)$ in every generating set. Both Magma and GAP use SLPs.

3 The major tasks

We identify three major problems for a (quasi)simple group $G = \langle X \rangle$. (Recall that G is *quasisimple* if G is perfect and $G/Z(G)$ is simple.)

(i) The *naming problem*: determine the name of G.

(ii) The *constructive recognition problem*: construct an isomorphism (possibly modulo scalars) between G and a "standard copy" of G.

(iii) The *constructive membership problem*: if $x \in G$, then write x as an SLP in X.

An algorithm to solve (i) may simply establish that G *contains* a named group as its unique non-abelian composition factor. Such information is useful: if we learn that G is a member of a particular family of finite simple groups, then we can apply algorithms to G which are specific to this family.

For each finite (quasi)simple group, we designate one explicit representation as its *standard copy* and designate a particular generating set as its *standard generators*.

For example, the standard copy of A_n is on n points; its standard generators are $(1, 2, 3)$ and either of $(3, \dots, n)$ or $(1, 2)(3, \dots, n)$ according to the parity of n.

To aid exposition, we focus on one common situation. Consider the classical groups, where the standard copy is the natural representation. Let $H \leq \mathrm{GL}(d, q)$ denote the natural representation of a classical group. Given as input an arbitrary permutation or projective matrix representation $G = \langle X \rangle$, a constructive recognition algorithm sets up an isomorphism between G and $H/Z(H)$.

To enable this construction, we define standard generators $\mathcal{S}$ for H. Assume we can construct the image $\bar{\mathcal{S}}$ of these standard generators in G as SLPs in X. We may now define the isomorphism $\phi : H/Z(H) \to G$. If we can solve the constructive membership problem in H, then the image in G of an arbitrary element of H can be constructed: if h has a known SLP in $\mathcal{S}$ then $\phi(h)$ is the SLP evaluated in $\bar{\mathcal{S}}$. Similarly if we can solve the constructive membership problem in G, then we can define $\tau : G \to H/Z(H)$. We say that these isomorphisms are *constructive*.

4 The black-box approach

The *black-box group approach*, initiated and pioneered by Babai and Beals (see [7] for an excellent account), focuses on the abstract structure of a finite group G. Recall, for example from [59, pp. 31–32], that G has a characteristic series of subgroups:

$$1 \leq O_\infty(G) \leq S^*(G) \leq P(G) \leq G$$

where

- $O_\infty(G)$ is the largest soluble normal subgroup of G, the *soluble radical*;
- $S^*(G)/O_\infty(G)$ is the socle of $G/O_\infty(G)$ and equals $T_1 \times \cdots \times T_k$, where each T_i is non-abelian simple;
- $\phi : G \to \mathrm{Sym}(k)$ is the representation of G induced by conjugation on $\{T_1, \dots, T_k\}$, and $P(G) = \ker \phi$;
- $P(G)/S^*(G) \leq \mathrm{Out}(T_1) \times \cdots \times \mathrm{Out}(T_k)$ and so is soluble (by the proof of the Schreier conjecture);
- $G/P(G) \leq \mathrm{Sym}(k)$ where $k \leq \log |G| / \log 60$.

In summary, the black-box approach aims to construct this characteristic series $\mathcal{C}$ for $G \leq \mathrm{GL}(d, q)$ using black-box algorithms. In 2009, as a culmination of 25 years of work, Babai, Beals & Seress [10] proved that, subject to the existence of a discrete log oracle and the ability to factorise integers of the form $q^i - 1$ for $1 \leq i \leq d$, there exist black-box polynomial-time Las Vegas algorithms to construct $\mathcal{C}$ for a large class of matrix groups. Building on results of [9], [56], [81] and [93], they solve the major tasks identified in Section 3 (and others) for groups in this class. We refer the reader to [7] and [10] for details.

In Section 12 we consider how the black-box approach underpins various practical algorithms for matrix groups.

5 Geometry following Aschbacher

By contrast, the *geometric approach* investigates whether a linear group satisfies natural and inherent geometric properties in *its action on the underlying space*. A classification of the maximal subgroups of classical groups by Aschbacher [3] underpins this approach. Let Z denote the subgroup of scalar matrices of $G \leq \mathrm{GL}(d, q)$. Then G is *almost simple modulo scalars* if there is a non-abelian simple group T such that $T \leq G/Z \leq \mathrm{Aut}(T)$, the automorphism group of T. We paraphrase Aschbacher's theorem as follows.

Theorem 5.1 *Let V be the vector space of row vectors on which $\mathrm{GL}(d, q)$ acts, and let Z be the subgroup of scalar matrices of G. If G is a subgroup of $\mathrm{GL}(d, q)$, then one of the following is true:*

C1. *G acts reducibly.*

C2. *G acts imprimitively: G preserves a decomposition of V as a direct sum $V_1 \oplus V_2 \oplus \cdots \oplus V_r$ of $r > 1$ subspaces of dimension s, which are permuted transitively by G, and so $G \leq \mathrm{GL}(s, q) \wr \mathrm{Sym}(r)$.*

C3. *G acts on V as a group of semilinear automorphisms of a (d/e)-dimensional space over the extension field $\mathrm{GF}(q^e)$ for some $e > 1$, and so G embeds in $\Gamma\mathrm{L}(d/e, q^e)$. (This includes the class of "absolutely reducible" linear groups, where G embeds in $\mathrm{GL}(d/e, q^e)$.)*

C4. *G preserves a decomposition of V as a tensor product $U \otimes W$ of spaces of dimensions $d_1, d_2 > 1$ over $\mathrm{GF}(q)$. Then G is a subgroup of the central product of $\mathrm{GL}(d_1, q)$ and $\mathrm{GL}(d_2, q)$.*

C5. *G is definable modulo scalars over a subfield: for some proper subfield $\mathrm{GF}(q')$ of $\mathrm{GF}(q)$, $G^g \leq \mathrm{GL}(d, q').Z$, for some $g \in \mathrm{GL}(d, q)$.*

C6. *For some prime r, $d = r^n$, and G is contained in the normaliser of an extraspecial group of order r^{2n+1}, or of a group of order 2^{2n+2} and symplectic-type (namely, the central product of an extraspecial group of order 2^{2n+1} with a cyclic group of order 4, amalgamating central involutions).*

C7. *G is tensor-induced: G preserves a decomposition of V as $V_1 \otimes V_2 \otimes \cdots \otimes V_m$, where each V_i has dimension $r > 1$, $d = r^m$, and the set of V_is is permuted transitively by G, and so $G/Z \leq \mathrm{PGL}(r, q) \wr \mathrm{Sym}(m)$.*

C8. *G normalises a classical group in its natural representation.*

C9. *G is almost simple modulo scalars.*

We summarise the outcome: a linear group preserves some natural linear structure in its action on the underlying space and has a normal subgroup related to this structure, or it is almost simple modulo scalars.

In broad outline, it suggests that a first step in investigating a linear group is to determine (at least one of) its categories in the Aschbacher classification. If a category is recognised, then we can investigate the group structure more completely using algorithms designed for this category. Usually, we have reduced the size and nature of the problem. For example, if $G \leq \mathrm{GL}(d, q)$ acts imprimitively, then we obtain a permutation representation of degree dividing d for G; if G preserves a tensor product, we obtain two linear groups of smaller degree. If a proper normal subgroup N exists, we investigate N and G/N recursively, ultimately obtaining a composition series for G.

The *base cases* for the geometric approach are groups in C8 and C9: classical groups in their natural representation, and other groups which are almost simple modulo scalars. Liebeck [74] proved that "most" maximal subgroups of $\mathrm{GL}(d, q)$ have order at most q^{3d}, small by contrast with $|\mathrm{GL}(d, q)|$; the exceptions are known. Further, the absolutely irreducible representations of degree at most 250 of all quasisimple finite groups are now explicitly known: see Hiss & Malle [55] and Lübeck [78].

Landazuri & Seitz [71] and Seitz & Zalesskii [96] provide lower bounds for degrees of non-linear irreducible projective representations of finite Chevalley groups. They show that a faithful projective representation in cross characteristic has degree that is polynomial in the defining characteristic. Hence our principal focus is on matrix representations in *defining characteristic*.

5.1 Deciding membership of an Aschbacher category

In [91] we reported in detail on the algorithms developed to decide if $G = \langle X \rangle \leq \mathrm{GL}(d, q)$, acting on the underlying vector space V, lies in one of the first seven Aschbacher categories. Consequently we only update that report. In Section 6.1 we report on a Monte Carlo algorithm which decides if G is in C8.

5.1.1 Reducible groups

The MeatAxe algorithm of Holt & Rees [57] is Las Vegas and has complexity $O(d^3(d \log d + \log q))$. A key component is a search in the $\mathrm{GF}(q)$-algebra generated by X for an element whose characteristic polynomial has an irreducible factor of multiplicity one. The analysis of [57], completed in [64], shows that the proportion of such elements is at least 0.08.

A matrix A over $\mathrm{GF}(q)$ for which the underlying vector space, considered as a $\mathrm{GF}(q)[A]$-module, has at least one cyclic primary component is *f-cyclic*. Glasby & Praeger [49] present and analyse a test for the irreducibility of G using the set of f-cyclic matrices in G, which contains as a proper subset those considered in [57].

5.1.2 C3 and C5

Holt *et al.* [58] present the Smash algorithm: effectively an algorithmic realisation of Clifford's theorem [36] about decompositions of V preserved by a non-scalar normal subgroup of G.

If G acts absolutely irreducibly, then we apply Smash to a normal generating set for its derived group G' to decide if G acts semilinearly. The polynomial-time algorithm of [48] to decide membership in C5 requires that G' acts absolutely irreducibly on V. Implementations of both are available in Magma.

Carlson, Neunhöffer & Roney-Dougal [33] present a polynomial-time Las Vegas algorithm to find a non-trivial "reduction" of an irreducible group G that either lies in C3 or C5, or whose derived group does not act absolutely irreducibly on V. In particular, they deduce that G is in one of C2, C3, C4, or C5; or obtain a homomorphism from G to $\mathrm{GF}(q)^\times$. An implementation is available in GAP.

5.1.3 Normalisers of p-groups

If G is in C6, then it normalises a group R of order either r^{2n+1} (extraspecial) or 2^{2n+2} (symplectic-type).

Brooksbank, Niemeyer & Seress [25] present an algorithm to produce a non-trivial homomorphism from G to either $\mathrm{GL}(2m, r)$ or $\mathrm{Sym}(r^m)$ where $1 \leqslant m \leqslant n$. They prove that this algorithm runs in polynomial time when G is either the full normaliser in $\mathrm{GL}(d, q)$ of R, or $d = r^2$. The special case where $d = r$ was solved by Niemeyer [89]. Implementations are available in GAP and Magma.

5.1.4 Towards polynomial time?

A major theoretical challenge is the following: decide membership of a given group $G \leq \mathrm{GL}(d, q)$ in a *specific* Aschbacher category in polynomial time. This we can always do for C1 and C8, and sometimes for C3, C5 and C6.

Recently Neunhöffer [86] has further developed and analysed variations of the Smash algorithm, and has also reformulated the Aschbacher categories to facilitate easier membership problems. This work and the "reduction algorithms" of [25] and [33] suggest that, subject to the availability of discrete log and integer factorisation oracles, it may be possible using matrix group algorithms to construct in polynomial time the composition factors of G. We contrast this with the results obtained in the black-box context [10].

6 Naming algorithms

Let b and e be positive integers with $b > 1$. A prime r dividing $b^e - 1$ is a *primitive prime divisor* of $b^e - 1$ if $r | (b^e - 1)$ but $r \not| (b^i - 1)$ for $1 \leq i < e$. Zsigmondy [107] proved that $b^e - 1$ has a primitive prime divisor unless $(b, e) = (2, 6)$, or $e = 2$ and $b + 1$ is a power of 2. Recall that

$$|\mathrm{GL}(d, q)| = q^{\binom{d}{2}} \prod_{i=1}^{d} (q^i - 1).$$

Hence primitive prime divisors of $q^e - 1$ for various $e \leq d$ divide both the orders of $\mathrm{GL}(d, q)$ and of the other classical groups. We say that $g \in \mathrm{GL}(d, q)$ is a *ppd-element* if its order is divisible by some primitive prime divisor of $q^e - 1$ for some $e \in \{1, \ldots, d\}$.

6.1 Classical groups in natural representation

Much of the recent activity on algorithms for linear groups was stimulated by Neumann & Praeger [84], who presented a Monte Carlo algorithm to decide whether or not a subgroup of $\mathrm{GL}(d, q)$ contains $\mathrm{SL}(d, q)$.

Niemeyer & Praeger [88] answer the equivalent question for an arbitrary classical group. This they do by refining a classification by Guralnick *et al.* [51] of the subgroups of $\mathrm{GL}(d, q)$ which contain ppd-elements for $e > d/2$. The resulting Monte Carlo algorithms have complexity $O(\log \log d(\xi + d^{\omega}(\log q)^2))$, where ξ is the cost of selecting a random element and d^{ω} is the cost of matrix multiplication. For an excellent account, see [94]. Their implementation is available in Magma.

6.2 Black-box groups of Lie type

Babai *et al.* [8] present a black-box algorithm to name a group G of Lie type in known defining characteristic p. The algorithm selects a sample $\mathcal{L}$ of random elements in G, and determines the three largest integers $v_1 > v_2 > v_3$ such that at least one member of $\mathcal{L}$ has order divisible by a primitive prime divisor of $p^v - 1$ for $v = v_1, v_2$, or v_3. Usually $\{v_1, v_2, v_3\}$ determines $|G|$ and so names G. The algorithm of Altseimer & Borovik [1] distinguishes between $\mathrm{P}\Omega(2m + 1, q)$ and $\mathrm{PSp}(2m, q)$ for odd q. The central result of [8] is the following.

Theorem 6.1 *Given a black-box group G isomorphic to a simple group of Lie type of known characteristic, the standard name of G can be computed using a polynomial-time Monte Carlo algorithm.*

An implementation developed by Malle and O'Brien is distributed with GAP and Magma. It includes naming procedures for the other quasisimple groups: if the non-abelian composition factor is alternating or sporadic, then we identify it by considering the orders of random elements.

6.3 Determining the defining characteristic

Theorem 6.1 assumes that the defining characteristic of the input group of Lie type is *known*.

Problem 6.2 Let G be a group of Lie type in *unknown* defining characteristic r. Determine r.

Liebeck & O'Brien [76] present a Monte Carlo polynomial-time black-box algorithm which proceeds recursively through centralisers of involutions of G to find $\mathrm{SL}(2, F)$, where F is a field in characteristic r. It is now easy to read off the value of r.

Kantor & Seress [67] prove that the three largest element orders determine the characteristic of Lie-type simple groups of odd characteristic, and use this result to underpin an alternative algorithm.

The former is distributed in Magma, the latter in GAP.