Open access • Proceedings Article • DOI:10.1145/1536414.1536425

Polynomial-time theory of matrix groups — Source link

László Babai, Robert Beals, Ákos Seress

Institutions: University of Chicago, Princeton University, Ohio State University

Published on: 31 May 2009 - Symposium on the Theory of Computing

Topics: Matrix group, Hidden subgroup problem, Classification of finite simple groups, Simple group and Group (mathematics)

Polynomial-time Theory of Matrix Groups

László Babai[*‡]
University of Chicago
laci@cs.uchicago.edu

Robert Beals
IDA-CCR Princeton
beals@idaccr.org

Ákos Seress[†‡]
Ohio State University
akos@math.ohio-state.edu

ABSTRACT

We consider matrix groups, specified by a list of generators, over finite fields. The two most basic questions about such groups are membership in and the order of the group. Even in the case of abelian groups it is not known how to answer these questions without solving hard number theoretic problems (factoring and discrete log); in fact, constructive membership testing in the case of 1×1 matrices is precisely the discrete log problem. So the reasonable question is whether these problems are solvable in randomized polynomial time using number theory oracles.

Building on 25 years of work, including remarkable recent developments by several groups of authors, we are now able to determine the order of a matrix group over a finite field of odd characteristic, and to perform constructive membership testing in such groups, in randomized polynomial time, using oracles for factoring and discrete log.

One of the new ingredients of this result is the following. A group is called *semisimple* if it has no abelian normal subgroups. For matrix groups over finite fields, we show that the order of the largest semisimple quotient can be determined in randomized polynomial time (no number theory oracles required and no restriction on parity).

As a by-product, we obtain a natural problem that belongs to BPP and is not known to belong either to RP or to coRP. No such problem outside the area of matrix groups appears to be known. The problem is the decision version of the above: *Given a list A of nonsingular $d \times d$ matrices over a finite field and an integer N, does the group generated by A have a semisimple quotient of order $\geq N$?*

We also make progress in the area of constructive recognition of simple groups, with the corollary that for a large class of matrix groups, our algorithms become Las Vegas.

[*]Partially supported by NSF Grant CCF-TF 0830370.
[†]Partially supported by NSF Grant CCF-TF 0830534 and NSA Grant H98230-08-1-0079.
[‡]The work reflects the views of the authors and not necessarily those of the NSF.

STOC'09, May 31–June 2, 2009, Bethesda, Maryland, USA.

Categories and Subject Descriptors

Theory of Computation [**Analysis of Algorithms and Problem Complexity**]: Nonnumerical Algorithms

General Terms

Algorithms

Keywords

computational group theory, matrix groups, discrete log

1. INTRODUCTION

The two most common explicit representations of finite groups are permutation groups and matrix groups over finite fields. We assume our groups are given by a list of generators.

The basic questions about such groups include testing membership in and computing the order of the group. A *constructive membership test* not only answers the question whether or not a given element belongs to a given group but in the case of a positive answer, it also provides a straight-line program that constructs the given element from the given generators of the group. Advanced questions ask about the structure of the group such as the names of its composition factors.

For permutation groups, Sims's classical algorithms, developed in the 1960s to serve the needs of computational group theory [49, 50], have been shown to solve the basic questions in polynomial time (cf. [36, 47]). The first analyzed version [27], motivated by the first group theoretic algorithm in graph isomorphism testing [3], appeared in 1980. Subsequently a number of advanced questions were also solved in polynomial time, including composition factors [39], Sylow subgroups [33] (cf. [41]), and the solvable radical ([38], see [42]).

In contrast, for the considerably more important class of groups of $d \times d$ matrices over the finite field of order $q = p^e$ (p a prime), even the most basic problems present great difficulties. For starters, even for $d = e = 1$, we cannot determine the order of a subgroup without factoring the integer $p - 1$, which is equivalent (under ERH) to factoring arbitrary integers [13, p. 241, Ex. 30]. Constructive membership testing in the case $d = 1$ is precisely the discrete logarithm problem. Membership for groups of 2×2 *diagonal* matrices is precisely the *decisional Diffie-Hellman problem*, a standard hard problem in cryptography. So the reasonable question is: *What can we do in (randomized) polynomial time, permitting oracles for these hard number theoretic problems?*

A complexity-theoretic study of problems of matrix groups over finite fields was initiated in the 1984 paper [12]. That paper introduced the concept of *black-box groups* and proved, among other things, that membership in black-box groups is in NP. The subsequent paper [4] put the membership problem in coAM; this problem thus served as one of the original motivations behind the concept of interactive proofs.

A *black-box* group is a finite group whose elements are encoded, not necessarily uniquely, by $(0,1)$-strings of uniform length n, with an oracle to perform group operations on the codewords, including the decision whether or not a string encodes the identity. Subgroups of a black-box group are *given* by a list of generators (i. e., strings corresponding to generators). (For the exact definition, see Def. 3.3.)

This concept turned out to be highly productive. While we cannot answer the basic questions without number theory oracles even for matrix groups, the "nonabelian structure" of the group has been mapped out in great detail in randomized polynomial time in the generality of black-box groups carrying a natural promise (see Def. 3.7 and Thm. 5.2).

Algorithmic developments started around 1990, first as two separate projects, one in the STOC/FOCS environment, another in the pure and computational group theory community. Most recently the two approaches merged and the "polynomial-time" paradigm received powerful contributions from group theorists who adopted the "black-box group" concept [2, 20, 21, 23, 29, 43].

The informal summary of the conclusions is that we were left with two layers of solvable ("nearly abelian") bottlenecks: the "outer automorphism layer" and the "solvable radical." Our main technical contribution in this paper is the removal of each of these remaining obstacles. The analysis heavily depends on recent results in statistical group theory [10, 43]. To discover the bottom layer, we build on Bray's algorithm [17], as analyzed by Parker and Wilson [43].

As a by-product, we describe a natural problem that belongs to BPP but is not known to belong to RP or coRP (Problem 2.6). As far as we can tell, no such problem outside the area of matrix groups is currently known.

The overall procedures for our results combine new elementary algorithms with a large body of previous work, algorithmic as well as group theoretic. Some of the ingredients depend on detailed knowledge of the ***classification of finite simple groups*** (CFSG). However, no such knowledge will be required for reading this paper.

2. MAIN RESULTS

The *general linear group* $\mathrm{GL}(d,q)$ consists of the $d\times d$ nonsingular matrices over the field $\mathbb{F}_q$ ($d\geq 1$, $q=p^e$ is a prime power). Subgroups $G\leq \mathrm{GL}(d,q)$ are called *matrix groups of characteristic* p. The **number theory oracles** our results refer to are the prime factorizations of the numbers q^i-1 for $i=1,\ldots,d$ (preprocessing); and Discrete Log oracles for fields of order $p^{k\ell}$ where $k\mid e$ and $\ell\leq d$ (so $p^{k\ell}\leq q^d$).

2.1 Constructive membership

Constructive membership is the problem of expressing an element in terms of the generators of the group.

Definition 2.1. Let G be a group and $S\subseteq G$. A *straight-line program* reaching some $g\in G$ from S is a sequence $(w_1,\ldots,w_m)$, $w_i\in G$, such that for each i either $w_i\in S$ or $w_i=w_j^{-1}$ for some $j<i$ or $w_i=w_jw_k$ for some $j,k<i$.

Definition 2.2. Let $G\leq H$ be groups; let G be given by a generating set S. The *constructive membership* problem for G in H is, given $g\in H$, decide whether $g\in G$, and if so find a straight-line program over S reaching g.

Our ambient group H will typically be $\mathrm{GL}(d,q)$ or S_ℓ.

2.2 Membership: odd characteristic

In this paper we prove the following main result.

Theorem 2.3. *There is a randomized polynomial-time algorithm which uses number theory oracles and, given a matrix group G of odd characteristic p,*

(*a*) *computes $|G|$ and decides membership in G;*

(*b*) *solves constructive membership in G.*

Previously similar results were known for solvable matrix groups only (Luks 1992 [40]; Luks's algorithms are deterministic). Our results give a definitive, and even recently not particularly hoped for, answer to the questions formulated a quarter century ago. The algorithms and their analysis build on a large body of prior work and most notably on the recent papers [10, 43] and Holmes et al. [29].

2.3 Unconditional results

The ***solvable radical*** $\mathrm{Rad}(G)$ is the largest solvable normal subgroup of G. Importantly, in a black-box group G, the radical is recognizable (Thm. 4.3, cf. Def. 3.4) and therefore, if G is a black-box group then $G/\mathrm{Rad}(G)$ can be treated as a black-box group. Note that $G/\mathrm{Rad}(G)$ is the largest quotient of G without abelian normal subgroups. Thus, by cutting out $\mathrm{Rad}(G)$ we remove the "abelian bottom" of the group. Our second main result says this is sufficient for a genuine BPP algorithm.

Theorem 2.4. *There is a randomized polynomial-time algorithm which, given a matrix group $G\leq \mathrm{GL}(d,p)$, computes $|G/\mathrm{Rad}(G)|$.*

Note that no oracles are required for this result and the case $p=2$ is not excluded. In fact, the result works even in the generality of "black-box groups of characteristic p" (Theorem 5.2, cf. Def. 3.7).

Our third main result finds the radical, the most elusive structural component of the group.

Theorem 2.5. *There is a randomized polynomial-time algorithm which, given a matrix group $G\leq \mathrm{GL}(d,p)$ with p odd, computes* $\mathrm{Rad}(G)$.

Note that no oracles are required for this result.

The limitation to odd characteristic is related to our current inability to efficiently find an element of even order in black-box simple groups of Lie type of characteristic 2; a major open problem (cf Section 3.2). (Random sampling has an exponentially small chance of finding such elements.) Specifically, Theorems 2.5 and 2.3(a) build on [43]; Theorem 2.3(b), in addition, also uses a technique from [29]. These techniques do not work in the case $p=2$.

Nevertheless, with different methods, we make progress in the even characteristic case as well (Section 2.7).

2.4 A problem in BPP

Groups without abelian normal subgroups are called ***semisimple***; so $G/\mathrm{Rad}(G)$ is the largest semisimple quotient of G. Consider the following decision problem:

Problem 2.6. Given a prime p, integers N and d, and a list A of nonsingular $d \times d$ matrices over $\mathbb{F}_p$, does the group generated by A have a semisimple quotient of order $\geq N$?

This problem is in BPP by Theorem 2.4 but currently we are not able to place it in either RP or coRP.

2.5 Composition series

Theorem 2.7. *There is a randomized polynomial-time algorithm which, given a matrix group $G \leq \mathrm{GL}(d,p)$,*

(a) *finds a composition series of $G/\operatorname{Rad}(G)$, and finds black-box representations and the standard names of the composition factors of $G/\operatorname{Rad}(G)$;*

(b) *finds a composition series of G, and lists the orders of the abelian composition factors, using number theory oracles, assuming $p \neq 2$.*

Note that part (a) does not use number theory oracles and does not exclude the case $p = 2$. Note also that the composition factors discovered (unconditionally) in part (a) include all nonabelian composition factors of G.

Theorem 2.3(a) is an immediate consequence of Theorem 2.7(b); and Theorem 2.4 is an immediate consequence of Theorem 2.7(a).

2.6 Constructive recognition

Constructive recognition of a simple group L is the ability to compute, in both directions, an isomorphism between the given representation of L and the *natural representation* of L (the explicit representation used in their textbook definitions). With each finite simple group L one can associate an explicit "standard" set of $O(\log|L|)$ generators with the property that given $h \in L$, one can express h as a word of length $O(\log|L|)$ in these generators, and such a word can be found in polynomial time (by generalized Gaussian elimination).

Definition 2.8. Let L be a finite simple group, given in its natural representation, along with its standard set T of generators. By *constructive recognition* of L within a class $\mathfrak{B}$ of black-box groups we mean the following promise problem: Given a black-box group $G = \langle S\rangle \in \mathfrak{B}$, known to be isomorphic to L, find a set S^* of generators of G and a bijection $S^* \to T$ which extends to an isomorphism $\lambda\colon G \to L$, and set up a data structure which permits the computation of $\lambda(g)$ for any $g \in G$.

We note that computing λ^{-1} reduces to the postulated computations: for $h \in L$, represent h as a word $w(T)$; then $\lambda^{-1}(h) = w(S^*)$.

We say that a constructive recognition algorithm works within a certain resource bound (such as "Las Vegas polynomial time with number theory oracles") if S^* is found, and for any $g \in G$, $\lambda(g)$ is found, within the given resource bound. The class $\mathfrak{B}$ we shall use consists of "quotients of matrix groups by recognizable normal subgroups" (cf. Def. 3.4). We refer to this class as "matrix group quotients."

We call a class $\mathcal{N}$ of finite simple groups "nice" if the members of $\mathcal{N}$ admit constructive recognition in randomized polynomial time with number theory oracles within the class of matrix group quotients.

The conjecture is that all finite simple groups form a nice class. Note that a subclass of a nice class is nice, and the union of a finite number of nice classes is nice.

Combining our machinery with Conder et al [23] and with [34] and the work by Brooksbank and Kantor [18, 19, 20, 21], we obtain the following result.

Theorem 2.9. *All finite simple groups, with the possible exception of those of exceptional Lie type, form a nice class.*

The result is obvious for sporadic groups. For cyclic groups of prime order, it follows from [6, 40] and our algorithms in Section 5; for the alternating groups, from [15] (cf. [6, 16]). The main content of Theorem 2.9 is that the *classical* simple groups are nice. The bulk of the proof can be found in [34] and in a series of papers by Brooksbank and Kantor [18, 19, 20, 21]; the latter prove constructive recognizability of the classical simple black-box groups assuming, in addition to the number theory oracles, an oracle for black-box constructive recognition of the groups $\mathrm{PSL}(2,q)$ (see Sec. 3.1).

In our context (the groups are given as matrix group quotients), the Brooksbank-Kantor reduction requires constructive recognition of $\mathrm{PSL}(2,q)$ given as a matrix group quotient. Our contribution is that, using pioneering work by Conder et al. [23], we provide such a recognition algorithm.

Lemma 2.10. *Let $G \leq \mathrm{GL}(n,p)$ be given. Let N be a recognizable normal subgroup of G such that $G/N \cong \mathrm{PSL}(2,q)$. Then we can constructively recognize the $\mathrm{PSL}(2,q)$ quotient in Las Vegas polynomial time using number theory oracles.*

Conder et al. [23] solve this problem for the case when $N = 1$, i. e., $G \cong \mathrm{PSL}(2,q)$. They erroneously remark that with this result they provide the oracle needed by Brooksbank and Kantor; this is incorrect even if the simple group L in question is given as a matrix group (rather than as a matrix group quotient).

2.7 Membership: all characteristics

To handle the case $p = 2$ we take an approach which works in all characteristics so in this subsection we make no assumption on the parity of p.

Let us fix a "nice" class $\mathcal{N}$ of finite simple groups (Section 2.6). We say that a group G is *p-nice* if all composition factors of G that are of Lie type of characteristic p belong to $\mathcal{N}$. (Note that under the conjecture mentioned, all groups are p-nice for all p.)

Theorem 2.11. *Theorem 2.3 holds for all groups $G \leq \mathrm{GL}(d,p)$ that are p-nice.*

We sketch the proof of this result and the corollaries below in Section 8. We should mention that essentially this result was erroneously claimed as [35, Thm. 6.1]. Two errors invalidate that claim: (1) [35] adopted the error of [23] mentioned after Lemma 2.10 ; and (2) [35] overlooked the unresolved status of the "outer automorphism layer" (see our Theorem 5.1).

2.8 Las Vegas upgrade

The "Short Presentation Conjecture" (SPC) [12] states that all finite simple groups G have presentations of bit-lengths polylog$|G|$. The constructive version (CSPC) requires these presentations to be explicit. It was shown in [12] that while membership in black box groups is in NP, under CSPC it is also in coNP. There does not seem an alternative to short presentations to certify nonmembership; CSPC, therefore, remains a cornerstone of any attempt to upgrade membership tests to Las Vegas status. CSPC has

been verified for all finite simple groups except for the "Ree groups of rank 1," a class of exceptional simple groups, denoted ${}^2G_2(q)$ where $q = 3^{2k+1}$, $k \geq 1$ [8, 31, 51].

Corollary 2.12. *If $G \leq \mathrm{GL}(n,p)$ is p-nice and in case $p = 3$ none of the composition factors of G is of type ${}^2G_2(q)$ then a presentation for G can be computed in Las Vegas polynomial time using number theory oracles.*

This upgrades our main results to Las Vegas for such groups:

Corollary 2.13. *Under the conditions of Corollary 2.12, the order of G, as well as constructive membership in G, are computable in Las Vegas polynomial time using number theory oracles.*

Terminology. In the rest of this paper, ***"efficient"*** will mean "randomized polynomial time" (no number theory oracles permitted unless expressly stated otherwise). We do not claim practical efficiency.

3. PRELIMINARIES

3.1 Group theory review

For the basics of group theory we refer to Rotman [45]. Here we briefly review notation and some concepts.

Let G be a group. $H \leq G$ denotes a subgroup, $N \lhd G$ normal subgroup ($G = N$ permitted), G/N the *quotient group*. G is ***simple*** if $|G| \geq 2$ and $N \lhd G \Rightarrow N = 1$ or $N = G$. The subgroup generated by the subset $A \subseteq G$ is denoted by $\langle A \rangle$. The ***normal closure*** of $A \subseteq G$ is the smallest normal subgroup containing A, denoted $\langle A^G \rangle$. We call $A \subseteq G$ a set of ***normal generators*** of G if $\langle A^G \rangle = G$. The commutator of a and b is $[a,b] = a^{-1}b^{-1}ab$. For $A, B \subseteq G$ we set $[A,B] = \langle [a,b] : a \in A, b \in B \rangle \leq G$. The *center* of G is $Z(G) = \{z \in G : [z,G] = 1\}$. The *central quotient* of G is $G/Z(G)$. The *derived subgroup* is $G' = [G,G]$. We say that G is ***perfect*** if $G' = G$. The *commutator chain* of G is the chain $G \geq G' \geq G'' \geq \ldots$. The *stable commutator* is the unique perfect member of this chain. The group G is *solvable* if its stable commutator is the identity. Every group has a unique maximal solvable normal subgroup, the ***solvable radical*** $\mathrm{Rad}(G)$.

$\mathrm{Aut}(G)$ denotes the automorphism group of G. For $g \in G$, the conjugation map $\gamma_g : a \mapsto g^{-1}ag$ is an *inner automorphism* of G. The group $\mathrm{Inn}(G) = \{\gamma_g : g \in G\}$ is normal in $\mathrm{Aut}(G)$; the quotient $\mathrm{Out}(G) = \mathrm{Aut}(G)/\mathrm{Inn}(G)$ is the ***outer automorphism group*** of G.

A p-group is a group of order a power of p. All p-groups are solvable. An ***elementary abelian p-group*** is a direct product of cyclic groups of order p.

The ***socle*** $\mathrm{Soc}(G)$ is the product of the minimal normal subgroups of G. It is a direct product of simple groups.

Fact 3.1. *If $T_1, \ldots, T_m$ are nonabelian simple groups then the only min. normal subgroups of $T_1 \times \cdots \times T_m$ are the T_i.*

A ***presentation*** $G = \langle A \,|\, \mathcal{R} \rangle$ of a group G is a description of G in terms of a set A of generators and a set $\mathcal{R}$ of relators, i. e., words in the free group F over the alphabet A. The group G is then the quotient $F/\langle \mathcal{R}^F \rangle$.

The most important examples of finite groups are permutation groups (subgroups of the symmetric groups S_n) and matrix groups over finite fields. The *even permutations* form the *alternating group* A_n which has index 2 in S_n. A *permutation representation* of G is a homomorphism $\varphi : G \to S_n$. We say that φ is *faithful* if it is injective ($\ker(\varphi) = 1$).

The *general linear group* $\mathrm{GL}(d,q)$ consists of the $d \times d$ nonsingular matrices over $\mathbb{F}_q$. If $q = p^e$ then $\mathrm{GL}(d,q) \leq \mathrm{GL}(de,p)$. The subgroups of the groups $\mathrm{GL}(d,q)$ ($q = p^e$) are the *matrix groups of characteristic p*. The *special linear group* $\mathrm{SL}(d,q)$ consists of the matrices $g \in \mathrm{GL}(d,q)$ with $\det(g) = 1$. The *projective special linear group* is the central quotient $\mathrm{PSL}(d,q) = \mathrm{SL}(d,q)/Z(\mathrm{SL}(d,q))$.

3.2 Classification of finite simple groups

For information about the Classification of Finite Simple Groups (CFSG) we refer to [24]. In brief: the finite simple groups are the cyclic groups of prime order, the alternating groups of degree ≥ 5, several families of "simple groups of Lie type," and a finite number of "sporadic" simple groups. Two consequences of the CFSG regarding the outer automorphism group are most pertinent.

Fact 3.2. *Let G be a finite simple group. Then $\mathrm{Out}(G)$ is solvable in three steps: $\mathrm{Out}(G)''' = 1$; and $|\mathrm{Out}(G)| = O(\log |G|)$.*

The *simple groups of Lie type* are central quotients of certain matrix groups over finite fields. The simplest examples are the groups $\mathrm{PSL}(d,q)$ (Sec. 3.1). The Lie-type simple groups come in two brands, "classical" and "exceptional." There are six classes of *classical simple groups*, each parametrized by a pair (d,q) where d refers to the dimension (of the matrices in the natural representation of these groups, i. e., in the matrix representation used in their definition, before taking the central quotient) and q is the order of the field. The names of the classes of classical groups are projective linear, symplectic, unitary, and three kinds of orthogonal groups. (The adjective "projective" applies to all and indicates that we factor out the center.) There are ten classes of *exceptional groups*, each parametrized by a prime power; each class is represented by matrices of a fixed dimension, so all exceptional groups are represented by matrix groups of bounded dimension (in their natural representation). Each Lie type simple group has an associated positive integer r called its *Lie rank*. We note that $r = \Theta(d)$ where d the is dimension mentioned above.

3.3 Black-box groups

Definition 3.3. Let G be a finite group. A *black-box representation* of G with code-length n is a surjection $f : S \to G$ for some subset $S \subseteq \{0,1\}^n$ of "valid strings," along with an oracle that performs the group operations: given two valid strings x, y, the oracle produces valid strings z, u such that $f(x)f(y) = f(z)$ and $f(x)^{-1} = f(u)$, and also answers the question whether or not $f(x) = 1$. We say that G is "given" as a black-box group if in addition a list of valid strings $x_1, \ldots, x_k$ is given such that $\langle f(x_1), \ldots, f(x_k) \rangle = G$.

Note that $|G| \leq 2^n$ where n is the code-length. The complexity of black-box group algorithms is always relative to the input length, which is $|A|n$ if G is given as $G = \langle A \rangle$.

Definition 3.4. A subgroup $H \leq G$ of a b.b. group is *known* if a set of generators for H is known. To *compute* H means to compute generators for H. A subgroup $H \leq G$ is *recognizable* if for any $g \in G$ we can efficiently test membership $g \in H$.

A subgroup can be recognizable without being efficiently computable. Recognizable subgroups include the center and the solvable radical of G, neither of which can we compute efficiently for b.b. groups. (In this paper we show how to compute the radical efficiently for matrix groups.) In the context of permutation groups, the automorphism group of a graph is recognizable (within the symmetric group on the vertices) but we cannot (currently) compute it efficiently.

The converse to this question, namely, whether a known subgroup is necessarily recognizable, has been the main question of the area for 25 years; it is this question that we resolve for matrix groups of odd characteristic, using number theory oracles.

Remark 3.5. As pointed out in [12], if G is a b.b. group and N is a *recognizable normal subgroup* then the quotient group G/N can be treated as a b.b. group, using the composition $f : S \to G \to G/N$. This transition gives particular power to the b.b. group model and is used extensively in the present paper.

Remark 3.6. Black-box groups have appeared in the quantum computing literature, typically with the restriction that f be one-to-one (the codewords are unique). This is a severe limitation since it removes the flexibility indicated in the preceding remark. Note, however, that our main result implies (via Shor [48]) that membership in and order of matrix groups of odd characteristic (or characteristic 2 with "nice" composition factors) belongs to the class BQP. Previously this was only known for solvable matrix groups [40].

The quantum complexity of black-box group membership has also been studied for its possible impact on the separation of the complexity classes QMA and QCMA [1].

Definition 3.7. We say that a black-box group G of encoding length n is a ***black-box group of characteristic*** p if G is isomorphic to a quotient of a subgroup of $\mathrm{GL}(m, p)$ where $m = \lceil n/\log p \rceil$.

4. PREREQUISITES

Of the three subsections below, only the second depends on detailed knowledge of CFSG.

4.1 General theory

First we make a general observation which follows by combining [7] and well-known facts about presentations such as [40, Lemma 4.1].

Proposition 4.1. *If G is a black-box group, N a recognizable normal subgroup, and we have constructive membership testing in and a presentation of G/N then we can efficiently compute N.*

As mentioned in the Introduction, for permutation groups, the basic problems, including order, constructive membership, and normal closure can be solved in polynomial time. Moreover, the same algorithms also construct a *presentation* of the group. We shall also need the fact that for permutation groups, a composition chain can be found in polynomial time [39].

For matrix groups, Luks found deterministic algorithms for the following problems.

Theorem 4.2 (Luks [40])**.** *Let $G \leq \mathrm{GL}(n, p)$. Solvability of G can be decided in deterministic polynomial time. Moreover, if G is solvable then constructive membership testing in G can be performed and the order of G and a presentation of G can be found in deterministic polynomial time using number theory oracles.*

For all other efficient algorithms we are aware of in matrix groups over finite fields, randomization is necessary. Randomization comes on two levels. The elementary method of "random subproducts" suffices for the following.

Theorem 4.3 ([7])**.** *Given a black-box group $G = \langle A \rangle$ with code-length n, one can efficiently find (i) a set of $O(n)$ generators for G; (ii) the normal closure of any subset $B \subseteq G$; (iii) the commutator chain and the stable commutator of G; (iv) decide solvability of G.*

All the remaining algorithms require access to ***nearly uniformly distributed random elements of*** G. This can be done in polynomial time for all black-box groups [5]. (A variation of the [5] algorithm, proposed by G. Cooperman and recently analyzed by Dixon [25], shows greatly improved time bound. For practical purposes, the "product replacement" heuristic due to Leedham-Green and Soicher [22] is preferred (cf. [28])).

4.2 Solvable-by-simple groups

A group G is *solvable-by-simple* if $G/\operatorname{Rad}(G)$ is simple. We review algorithms for such G.

An algorithm from [15] handles the case when $G/\operatorname{Rad}(G)$ has a low degree permutation representation.

Theorem 4.4 ("Blind descent" [15])**.** *Given a black-box group G and an integer m, if G has a nontrivial permutation representation of degree $\leq m$ then one can find, in randomized $O((m+n)^c)$ group operations, a faithful permutation representation of G of degree $\leq m^c$, or a nontrivial element from a proper normal subgroup of G. (Here c is an absolute constant.)*

This result builds on, and can be powerfully combined with, the following theorem.

Theorem 4.5 ([37, 46, 26])**.** *If $G \leq \mathrm{GL}(d, p)$ has a composition factor T which is a simple group of Lie type of characteristic $r \neq p$ then T has a faithful permutation representation of degree $\leq d^c$ for some absolute constant c.*

So the case not covered by Theorem 4.4 is when $G/\operatorname{Rad}(G)$ is of Lie type of characteristic p. There has been much recent progress on this difficult case.

The first major result we require is the statistical recognition of black-box simple groups of characteristic p.

Theorem 4.6 ([9])**.** *Let G be a black-box group with the additional promise that G is a simple group of Lie type of characteristic p. Then the order of G can be found efficiently, based on a small sample of the orders of its elements.*

Actually, the statistics are based not on the orders of elements (which require a factorization of $|\mathrm{GL}(m, p)|$ for some m), but on polynomial-time computable properties of the orders.

This in fact means that we can recognize the standard name of the simple group G, except for an infinite sequence of pairs of finite simple groups of equal order. These pairs have been efficiently separated by Altseimer and Borovik [2], building on an algorithm due to Bray [17] for centralizers of involutions.

Theorem 4.7 ([17, 2, 43])**.** *Let G be a black-box group isomorphic to a simple group of Lie type in odd characteristic p. Then involution centralizers can be computed in randomized polynomial time.*

Definition 4.8. We say that the group G is an *irreducible affine extension of characteristic p* of the simple group T if G has a minimal normal p-subgroup N on which G acts nontrivially by conjugation and $G/N \cong T$. (Note that in this case, $N = \mathrm{Rad}(G)$ is elementary abelian.)

A further application of Bray's method will be central to our main result:

Theorem 4.9 ("affine descent," Parker-Wilson[43])**.** *Let p be an odd prime and G a black-box group with the additional promise that G is an irreducible affine extension of characteristic p of a simple group T of Lie-type of characteristic p. Then a nontrivial element of $\mathrm{Rad}(G)$ can be found efficiently.*

Finally, we need the following algorithmic result.

Theorem 4.10 ([11, Thm. 4.15])**.** *Let G be a black-box group of characteristic p. Assume $G/Z(G)$ is nonabelian simple. Then we can efficiently find $Z(G)$.*

The main tool in the analysis of our algorithms is the following result in statistical group theory:

Theorem 4.11 ([10, Cor.1.3 & Thm.1.4])**.** *For a nonabelian finite simple group G and a prime r, let $\rho_r(G)$ denote the proportion of elements in G of order relatively prime to r. (a) For all r and G, $\rho_r(G) \geq c/\sqrt{\log |G|}$, where $c > 0$ is an absolute constant. (b) If G is a quotient of a subgroup of $\mathrm{GL}(d, q)$ for some d, q then $\rho_r(G) \geq \min\{1/31, 1/(2d)\}$.*

Note that for a black-box group of code-length n this means a proportion of $\Omega(1/\sqrt{n})$, so random sampling will find elements of order relatively prime to r with fair frequency.

4.3 Overall framework

The structural frame of the overall algorithm is the following normal chain, defined in the programmatic paper [6]; much of the development during the past decade, outlined above, was directly or indirectly in response to that paper.

$$1 \leq \mathrm{Rad}(G) \leq \mathrm{Soc}^*(G) \leq \mathrm{Pker}(G) \leq G. \qquad (1)$$

We define the terms in this chain. Let $H = G/\mathrm{Rad}(G)$ and let $\varphi : G \to H$ be the natural surjection. Then $\mathrm{Soc}^*(G) = \varphi^{-1}(\mathrm{Soc}(H))$. Let $\mathrm{Soc}(H) = T_1 \times \cdots \times T_m$ where the T_i are nonabelian simple groups. Then by Fact 3.1, conjugation by G permutes the set $\{T_1, \ldots, T_m\}$, thus we obtain a permutation representation $G \to S_m$. We define $\mathrm{Pker}(G)$ as the kernel of this representation.

To find the order of G, we only need to find the order of each of the four "layers" in this chain. We note that the top layer, $G/\mathrm{Pker}(G)$ is a permutation group ($\leq S_m$); the second layer, $\mathrm{Pker}(G)/\mathrm{Soc}^*(G)$ is a subgroup of $\mathrm{Out}(T_1) \times \cdots \times \mathrm{Out}(T_m)$ and is therefore solvable; the third layer, $\mathrm{Soc}^*(G)/\mathrm{Rad}(G)$ is a product of simple groups; and the most elusive fourth layer, $\mathrm{Rad}(G)$, is solvable.

We summarize the main results of [6].

Theorem 4.12 ([6])**.** *Given a black-box group G of characteristic p, the following can be computed efficiently: the permutation representation $\varphi : G \to S_m$; its kernel $\mathrm{Pker}(G)$; for each $i \leq m$, a perfect subgroup $T_i^* \leq G$ such that $T_i^* \mathrm{Rad}(G)/\mathrm{Rad}(G) = T_i$; a black-box representation of characteristic p for each T_i; and a permutation representation for each T_i that is not Lie-type of characteristic p.*

Corollary 4.13. *Given a black-box group G of characteristic p, we can efficiently find the orders of layers 1 and 3.*

Indeed, layer 1 is an explicit permutation group; and for layer 3, if T_i is not Lie type of characteristic p then we again have a permutation representation; in the remaining (hard) case, we can find $|T_i|$ by Theorem 4.6. In this paper we find the orders of the solvable layers 2 and 4.

5. OUTER AUTOMORPHISM LAYER

The following result maps out layer 2 (the "outer automorphism layer").

Theorem 5.1. *Let G be a black-box group of characteristic p. Then a faithful permutation representation of $\mathrm{Pker}(G)/\mathrm{Soc}^*(G)$ can be computed efficiently. Consequently the order of this group as well as generators for $\mathrm{Soc}^*(G)$ can be computed efficiently.*

Combined with Corollary 4.13 this will yield the following result, which includes Theorem 2.7(a) and Theorem 2.4.

Theorem 5.2. *For a black-box group G of a given finite characteristic, one can efficiently determine the order of the quotient group $G/\mathrm{Rad}(G)$.*

Lemma 5.3. *Let G be a black-box group of characteristic p with the promise that there exist nonabelian simple groups $T_1, \ldots, T_m$ such that $T_1 \times \cdots \times T_m \leq G \leq \mathrm{Aut}(T_1) \times \cdots \times \mathrm{Aut}(T_m)$. Let $\varphi_i : G \to \mathrm{Aut}(T_i)$ be the i-th projection. Given $g \in G$ and $i \leq m$ we can efficiently decide whether or not $\varphi_i(g) \in T_i$. (Note that we are not assuming that φ_i is "given.")*

Proof. For simplicity, we give the proof under the additional assumption that a superset $\mathcal{P}$ of the primes dividing the order of G is given. However, this assumption can be dispensed with, using an explicit set of "pretend-primes" to find appropriate "pseudo-orders" of elements (see [6, 10]).

Note that given $\mathcal{P}$, we can compute the order of any $g \in G$. For integers r and z, by the r'-part of z we mean the largest divisor of z relatively prime to r. For $h \in G$, let $e_r(h)$ denote the r'-part of the order of h.

Algorithm 1
split $G''' = T_1 \times \cdots \times T_m$ into its factors T_j
for $r \,|\, |g|$ **do**
  $g_r := g^{e_r(g)}$
  set "$\varphi_i(g_r) \in T_i$" = FALSE
  repeat $O(\sqrt{n} \log |\mathcal{P}|)$ times
    $x :=$ random element of T_i
    $y := (g_r x)^{e_r(g_r x)}$
    if $[y, T_i] = 1$ **then** set "$\varphi_i(g_r) \in T_i$" = TRUE
  end(repeat)
  if "$\varphi_i(g_r) \in T_i$" = FALSE **then**
    return "$\varphi_i(g) \notin T_i$", **exit**
end(for)
return "$\varphi_i(g) \in T_i$"

Proof of correctness. We have $\mathrm{Rad}(G) = 1$ and $\mathrm{Soc}(G) = G''' = T_1 \times \cdots \times T_m$. According to Theorem 4.12, we can compute each T_j.

Let now $h \in G$ be an r-element, $x \in T_i$, and $y = (hx)^{e_r(gx)}$. Note that if $\varphi_i(g) \notin T_i$ then for any $x \in T_i$, we have $\varphi_i(y) \notin T_i$ and therefore $[y, T_i] \neq 1$. On the other hand, if $\varphi_i(h) \in T_i$ then for a random $x \in T_i$, the element $\varphi_i(h)x = \varphi_i(hx)$ is also a random element of T_i and therefore, by Theorem 4.11, it has an $\Omega(1/\sqrt{n})$ chance of having order relatively prime to r, in which case $\varphi_i(y) = 1$ and therefore $[y, T_i] = 1$. So repeating the test $[y, T_i] = 1$ for $O(\sqrt{n} \log |\mathcal{P}|)$ random choices of x will, for any constant c, with probability $> 1 - |\mathcal{P}|^{-c}$, tell whether or not $\varphi_i(h) \in T_i$. By choosing c large enough, we ensure that with high probability, there will be no error for any r.

For $g \in G$, the Chinese Remainder Theorem gives $g \in \langle g^{e_r(g)} \mid r \in \mathcal{P} \rangle$, so $\varphi_i(g) \in T_i$ exactly if $\varphi(g^{e_r(g)}) \in T_i$ for all primes $r \,\big|\, |g|$. □

Now to prove Theorem 5.1, we may assume $\mathrm{Rad}(G) = 1$ (viewing $G/\mathrm{Rad}(G)$ as a black-box group, because $\mathrm{Rad}(G)$ is recognizable according to Theorem 4.3 (ii) and (iv)). According to Theorem 4.12 we have generators for $\mathrm{Pker}(G)$, so we may assume $G = \mathrm{Pker}(G)$. In other words, we are exactly in the situation of Lemma 5.3. Using that lemma, we are able to construct the Cayley table (multiplication table) of $\varphi_i(G)/T_i$ efficiently because, by Fact 3.2, $|\varphi_i(G)/T_i| \leq |\mathrm{Out}(T_i)| = O(\log |T_i|)$ is small. We can then turn the Cayley table into a faithful (regular) permutation representation of $\varphi_i(G)/T_i$.

Finally, the map $g \mapsto (\varphi_1(g)T_1, \ldots, \varphi_m(g)T_m)$ now gives a permutation representation of $G = \mathrm{Pker}(G)$ on the small domain $\bigcup_i \varphi_i(G)/T_i$ with kernel $\mathrm{Soc}(G) = \mathrm{Soc}^*(G)$. □

6. THE SOLVABLE RADICAL

Above we have seen three algorithms, in Thms 4.4, 4.9, and 4.10, which produce elements of the radical in certain very special cases. We describe a general "adaptation principle" which allows us to essentially apply the algorithms of Theorems 4.9, and 4.10 to quotients of G by normal subgroups which we haven't constructed and cannot recognize.

Definition 6.1. We say that a black-box group algorithm *distinguishes* groups G and H if: (1.) The output of the algorithm, for any input group, is "yes" or "no". (2.) The probability of output "yes" on input G differs from the probability of output "yes" on input H by at least some constant $c > 0$.

Note that the gap c can be amplified to $1 - \epsilon$ by repetition and threshold vote.

The algorithms of Theorems 4.9 and 4.10 can be used to distinguish G from H if $\mathrm{Rad}(H)$ is trivial but $\mathrm{Rad}(G)$ is of the type that would be found by the algorithm (simply have the algorithm output "yes" if it finds a nontrivial element of the radical).

Lemma 6.2 (Adaptation Principle)**.** *Let G be a black-box group, with normal subgroups A and B, such that $B \leq A$ and A is recognizable. Suppose we have a black-box algorithm which distinguishes G/B from G/A. Then the algorithm can be modified to compute a list U, of elements of A, containing with high probability at least one element of $A \setminus B$.*

Proof. We run the algorithm on G/A; that is, we use the membership test for A when the algorithm performs an identity test. We let U be the list of elements tested which were found to lie in A. Since the algorithm distinguishes G/B from G/A, it must be the case that, with high probability, some identity test has distinct answers for G/B and G/A. That is, some element of U lies in $A \setminus B$. □

A more detailed explanation of a similar idea can be found in [14, p. 41].

The following lemma is the key step in the proof of Theorem 2.5.

Lemma 6.3. *Let G be a black box group of odd characteristic p. Assume G is perfect and that $G/\mathrm{Rad}(G)$ is simple. Then $\mathrm{Rad}(G)$ can be constructed efficiently.*

For the proof, we first use Theorem 4.4 on $G/\mathrm{Rad}(G)$ (so we require only the simple group case of the [15] algorithm). If a permutation representation is constructed then we compute its kernel, which must be $\mathrm{Rad}(G)$. Otherwise, we are in the case that G has no permutation representation of small degree.

In this case we shall use adaptations, via Lemma 6.2, of the algorithms of Theorems 4.9, and 4.10. The recognizable subgroup A is $\mathrm{Rad}(G)$. Note that to run the adapted algorithms we need only specify A. We concatenate the lists U returned by several iterations of the adapted algorithms; we will show that the resulting list is likely to be a set of normal generators for $\mathrm{Rad}(G)$.

Algorithm 2a (: The input G is a black box group of odd characteristic p. G is perfect, and $G/\mathrm{Rad}(G)$ is simple. We construct $\mathrm{Rad}(G)$.:)

run "blind descent" (Theorem 4.4) on $G/\mathrm{Rad}(G)$.
if a representation $\varphi : G \to S_k$ is found **then**
  return $\ker(\varphi)$, **exit**.
$U := \emptyset$.
repeat $O(n)$ times:
  (A) $U := U \cup \mathrm{AdaptedCenter}(G)$
  (B) $U := U \cup \mathrm{AdaptedAffineDescent}(G)$
end(repeat)
return $\langle U^G \rangle$

Proof of correctness. If $G/\mathrm{Rad}(G)$ has a low degree permutation representation, then the first part of the algorithm will find such a representation, which, when viewed as a representation of G, has kernel exactly $\mathrm{Rad}(G)$, and we are done.

Otherwise: we observe that "$U \subseteq \mathrm{Rad}(G)$" is indeed a loop-invariant. Suppose that before a particular iteration of the "repeat" loop, $\langle U^G \rangle \neq \mathrm{Rad}(G)$. We claim that then with high probability, $\langle U^G \rangle$ increases during this iteration.

Let $A = \mathrm{Rad}(G)$. Let us now choose a normal subgroup $B \lhd G$ such that $U \subseteq B < A$, with B be maximal under these constraints. Then A/B is elementary abelian for some prime r. There are two possibilities:

(a) $A/B \leq Z(G/B)$.
(b) G/B acts nontrivially and irreducibly on A/B.

If G belongs to case (a) then method (A) produces elements of $A \setminus B$ with high probability. If G belongs to case (b) then our initial run of the algorithm of Theorem 4.4

forces $r = p$, and method (B) produces elements of $A \setminus B$ with high probability.

Since $\langle U^G \rangle$ can only increase $\log |G| \leq n$ times, $O(n)$ iterations will guarantee an exponentially high $(1 - C^{-n})$ probability of success ($\langle U^G \rangle = \mathrm{Rad}(G)$) by a Chernoff argument because each round succeeds with constant probability in increasing $\langle U^G \rangle$ while $\langle U^G \rangle < \mathrm{Rad}(G)$. □

Lemma 6.4. *Let G be a black-box group. Suppose we are given a superset $\mathcal{P}$ of the prime divisors of $|G|$. Suppose further that $G = H \times R$ where $H = T_1 \times \cdots \times T_m$ and the T_i are nonabelian simple groups. Assume each T_i is known. Then any given g in G can be efficiently expressed as the product $g_R g_1 \cdots g_m$, with $g_R \in R$ and $g_i \in T_i$.*

Proof. Let $h \in G$. As before, let $e_r(h)$ denote the r'-part of the order of h. We say that the *support* of h is the set $\mathrm{supp}(h) = \{i \in [m] : [h, T_i] \neq 1\}$.

We are claiming we can compute the projection maps from G to R and to the T_i. Note that any element of G can effectively be expressed, via the Chinese Remainder Theorem, in terms of the $g^{e_r(g)}$, which have prime power order. Since the projections are homomorphisms, it suffices to show that we can compute them in the case that $|g|$ is a power of some prime r.

Algorithm 2b
$I := \{1, \ldots, m\}$
while $I \neq \emptyset$ **do**
  for $i \in I$ select $t_i \in T_i$ at random **end(for)**
  $a := g t_1 \cdots t_m$
  $h := a^{e_r(a)}$
  $I := \mathrm{supp}(h)$
end(while)
$a := g t_1 \cdots t_m$
compute N : $N \equiv 1 \pmod{|g|}$, $N \equiv 0 \pmod{e_r(a)}$
$g_R := a^N$
for $i = 1, \ldots, m$ **do** $g_i := {g_R}^{-1} ({a t_i}^{-1})^N$ **end(for)**

Proof of correctness. Note that h commutes with T_i iff the T_i component of h is trivial. This happens iff the T_i component of a has order prime to r, which depends only on g and t_i, and not on t_j for $j \neq i$. So as soon as we find a good t_i, so that $i \notin \mathrm{supp}(h)$, we have $i \notin I$ for subsequent iterations, and t_i remains unchanged.

By Theorem 4.11, $|\mathrm{supp}(h)|$ is expected to be reduced by a factor of $(1 - 1/s)$ in each round of the **while** loop, where $s = O(\max_i \sqrt{\log |T_i|}) = O(n)$; so with high probability we reach $\mathrm{supp}(h) = \emptyset$ in $O(n \log m)$ rounds of the **while** loop.

By choice of N, we have $g^N = g$, but the T_i component of $(g t_i)^N$ is trivial. So the computed g_R value is the R component of g. Also, the computed g_i value is the T_i component of g, as it agrees with g in the T_i component and has all other components trivial. □

Now we proceed to the proof of Theorem 2.5. By Theorem 5.1 we can efficiently construct $\mathrm{Soc}^*(G)$, so we may assume $G = \mathrm{Soc}^*(G)$. Let the T_i^* be as in Theorem 4.12. Then the T_i^* satisfy the conditions of Lemma 6.3 and therefore we can compute $S_i := \mathrm{Rad}(T_i^*) = T_i^* \cap \mathrm{Rad}(G)$. Let S be the normal closure of $\bigcup_{i=1}^m S_i$.

Now, for each i, $T_i \lhd G/S$, therefore $G/S = T_1 \times \cdots \times T_m \times R$ where $R = \mathrm{Rad}(G/S)$. To generate $\mathrm{Rad}(G)/S$, we run, $O(n)$ times, the adaptation of the algorithm of Lemma 6.4, as in Lemma 6.2. The subgroup A is $\mathrm{Rad}\, G$, and the subgroup B is the normal closure of $\bigcup_{i=1}^m S_i$, and we view the algorithm of Lemma 6.4 as distinguishing $R = 1$ from $R \neq 1$. □

Remark 6.5. In our application of Lemma 6.2 to Lemma 6.4, there are two kinds of identity tests: those involved in computing $\mathrm{supp}(h)$, and testing if g_R is the identity. Note for the tests of the first type, we get the same answer mod $\mathrm{Rad}(G)$ as we would mod S: an element which centralizes T_i mod $\mathrm{Rad}(G)$ necessarily centralizes T_i mod S. Therefore, the elements g_R which we construct, which are of course trivial mod $\mathrm{Rad}(G)$, are actually the projections into R of the g when considered mod S. In particular, to obtain a generating set for R, it suffices to project a generating set of G consisting of elements of prime power order.

7. CONSTRUCTIVE MEMBERSHIP

In this section we combine our framework with a recent method of Holmes et al. [29] to obtain constructive membership testing for all matrix groups of odd characteristic (Theorem 2.3).

Centralizers of involutions are a key ingredient; we require Bray's algorithm 4.7, as analyzed in [2] and [43]. Holmes et al. [29] give a reduction of constructive membership in a black-box group G to three instances of constructive membership in involution centralizers in G; they show their reduction runs in polynomial time in the case G is a Lie type simple group of odd characteristic. Their timing does not include the time for the subproblems, and as the subproblems involve groups that are not simple, additional work is needed before one can recursively apply their algorithm. They state ([29, p.729]) that "If the obstructions to a fully recursive algorithm could be overcome, then [29, Thm.2] could be used to bound ... the number of recursive calls to a polynomial in" (the Lie rank of G). For matrix groups, we complete this project below.

First we extract what we need from the main results of [29]. For a prime p, let $\mathcal{L}_p$ denote the class of Lie-type simple groups of characteristic p. For $G \in \mathcal{L}_p$, let $r_p(G)$ denote the Lie-rank of G if G is classical and the Lie-rank plus 1 if G is exceptional. For an arbitrary group G, let $r_p(G)$ denote the maximum Lie-rank of those composition factors of G which belong to $\mathcal{L}_p$, and 0 if G has no such composition factor.

Theorem 7.1 (Holmes et al. [29])**.** *There is an absolute constant C such that the following holds. Let $G \in \mathcal{L}_p$ be a black-box group. Then constructive membership in G reduces, in randomized polynomial time, to constructive membership in three subgroups $H_i \leq G$ $(i = 1, 2, 3)$ such that (i) $r_p(H_i) < r_p(G)$; (ii) $r_p(G) \geq C \Rightarrow (\forall i)(r_p(H_i) < 3 r_p(G)/4)$.* □

Proof of Theorem 2.3(b): Let $G = \langle S \rangle \leq \mathrm{GL}(d, p)$, p odd. We construct an SLP from S to $g \in \mathrm{GL}(d, p)$; if any step fails, we abort with "$g \notin G$."

Let $\varphi : G \to H$ be a homomorphism. We adopt the convention that an SLP routine applied to (G, g, φ) computes an SLP in H which, lifted to G, reaches $x \in G$ with $\varphi(x) = \varphi(g)$, replaces g with gx^{-1}, and replaces G with $\ker(\varphi)$. We use the following subroutines: SLP-Perm(G, g, φ) [49, 50, 27, 36] for φ a permutation representation; SLP-Sol(G, g) [40] for G a solvable matrix group; and SLP-Lie(G, g, φ) [29] for $H = \mathrm{Im}(\varphi) \in \mathcal{L}_p$. The first two of these are self-contained,

the last recursively calls our Algorithm 3 below on three subproblems. - We identify some important homomorphisms which we can compute:

1. $\varphi_{\text{top}} : G \to G/\operatorname{Pker}(G)$ (by Theorem 4.12)
2. $\varphi_{\text{out}} : \operatorname{Pker}(G) \to \operatorname{Pker}(G)/\operatorname{Soc}^*(G)$ (Thm. 5.1)
3. $\varphi_{\text{small}} : T \to S_\ell$, $T \notin \mathcal{L}_p$ nonab. simple (Th. 4.4)
4. $\varphi_i : \operatorname{Soc}^*(G) \to T_i$ (Lem. 6.4 proj. mod $\operatorname{Rad}(G)$)

Algorithm 3
1. find $T_1^*, \ldots, T_m^*$ as in Theorem 4.12
2. SLP-Perm$(G, g, \varphi_{\text{top}})$ (:reduced to Pker:)
3. SLP-Perm$(G, g, \varphi_{\text{out}})$ (:reduced to Soc*:)
4. **for** $i = 1, \ldots m$:
5. **if** $\varphi_i(G) \in \mathcal{L}_p$ **then** SLP-Lie(G, g, φ_i)
6. **else** SLP-Perm$(G, g, \varphi_{\text{small}} \circ \varphi_i)$ **end(if)**
7. **end(for)** (:reduced to Rad:)
8. SLP-Sol(G, g)

Line 5 calls the procedure for 3 subproblems. We estimate the number $f(G)$ of recursive calls. We note that $r_p(G) = O((\log_p |G|)^{1/2})$.

Let $G_1, \ldots, G_t$ be the groups on which Algorithm 3 is called recursively by line 5 within the **for** loop of line 4. So $\sum_i \log_p |G_i| \le 3 \log_p |G|$. Let $\Phi_j = \sum \log_p |H|$ where the summation extends over all groups occurring at level j of the recursion tree. It follows that $\Phi_j \le 3\Phi_{j-1}$. For $r_p(G) \ge C$ we also have $r_p(G_i) \le (3/4) r_p(G)$; therefore the height of the recursion tree is $h < C + \log_{4/3} r_p(G)$. Therefore the number of nodes of the tree is $f(G) \le \sum_j \Phi_j = O(3^h \Phi_0) = O(r_p(G)^{\log_{4/3} 3} \log_p |G|) = O((\log_p |G|)^{2.91})$. □

8. ARBITRARY CHARACTERISTIC

In this section we no longer require the characteristic to be odd. Instead, we place additional restrictions on the composition factors ("nice" groups, cf. Sec. 2.6) and make more extensive use of discrete log oracles through the constructive recognition algorithms of [23, 18, 19, 20, 21].

Before proving Lemma 2.10, we state a structural result.

Definition 8.1. Let $T \lhd G$ and $S = G/T$. Let V be a group with a G-action; let L be the kernel of this action. We say that this action is S-trivial if $LT = G$.

Proposition 8.2. *Let G be perfect, $R = \operatorname{Rad}(G)$, $S = G/R$ simple. Let p be a prime. Assume G acts S-trivially on all p'-chief-factors of R. Then R contains a p-subgroup K such that $K \lhd G$ and $R/K \le Z(G/K)$.*

We omit the proof.

Proof of Lemma 2.10: Let $G \le \mathrm{GL}(d, p)$ and suppose that G has a recognizable normal subgroup N with quotient group $H \cong \mathrm{PSL}(2, q)$ for some prime power q. We show how to explicitly map G to H and perform constructive recognition in the image.

By Theorem 4.6, we know q. If q is tiny, we use the algorithm of [34] on the black box group G/N. We assume henceforth that q is not tiny.

We apply the algorithm of Theorem 4.12 to compute a homomorphism $\varphi : G \to S_m$ with kernel $\operatorname{Pker}(G)$ and subgroups T_i^* for each composition factor T_i of $\operatorname{Soc}^*(G)/\operatorname{Rad}(G)$. We have by Theorem 4.5 that q is a power of p, $H = T_i$ for some i, and the conditions of Proposition 8.2 are met by the subgroup T_i^*. Also, modulo the radical, G is the direct product of T_i^* and N; so we can use Lemma 6.4 to project G onto T_i^*.

It now suffices to explicitly map T_i^* to H and perform constructive recognition in the image. We compute a composition series of $\mathbb{F}_p^d$, considered as a T_i^* module [44, 30, 32]. It follows from Proposition 8.2 that on one of the composition factors T_i^* acts as a quasisimple matrix group (cf. [35, Sec. 6]) and so the algorithm of [23] can be applied for constructive recognition. □

Recall (Section 2.6) that Lemma 2.10 completes the proof of Theorem 2.9, establishing a large nice family $\mathcal{N}$ of simple groups.

Lemma 8.3. *Let $G \le \mathrm{GL}(d, p)$. Assume G is perfect and that $G/\operatorname{Rad}(G) \in \mathcal{N}$ is a Lie-type simple group. Then $\operatorname{Rad}(G)$ can be constructed efficiently using number theory oracles.*

Proof. Apply Prop. 4.1, using constructive recognition of $G/\operatorname{Rad}(G)$. □

Proof of Theorem 2.11(b): The only place where we have to modify the algorithm we gave for odd characteristic is that we need to find a new solution to the constructive membership problem in groups $H \le \mathrm{GL}(d, p)$, where $H/\operatorname{Rad}(H)$ is a simple group in $\mathcal{N}$. Since $H/\operatorname{Rad}(H) \in \mathcal{N}$, the constructive recognition algorithm for $H/\operatorname{Rad}(H)$ yields an SLP that reduces the constructive membership problem to $\operatorname{Rad}(H)$; the latter is solved via [40]. □

Proof of Corollaries 2.12 and 2.13. If the composition factors of a group G are constructively recognizable then standard techniques (cf. [47, Section 8.4], [40, Lemma 4.1]) allow us to write a polynomial-length presentation for G. Evaluating this presentation we can check the correctness of our calculations, thereby turning our algorithms into Las Vegas. □

9. REFERENCES

[1] S. Aaronson and G. Kuperberg. Quantum versus classical proofs and advice. *Theory of Computing*, 3:129–157, 2007.

[2] C. Altseimer and A. V. Borovik. Probabilistic recognition of orthogonal and symplectic groups. In *Groups and Computation III*, pp 1–20. deGruyter, 2001.

[3] L. Babai. Monte Carlo algorithms in graph isomorphism testing. *Université de Montréal Tech. Rep.*, DMS 79-10:42 pages, 1979.

[4] L. Babai. Trading group theory for randomness. In *Proc. 17th STOC*, pp. 421–429. ACM, 1985.

[5] L. Babai. Local expansion of vertex-transitive graphs and random generation in finite groups. In *Proc. 23rd STOC*, pp. 164–174. ACM, 1991.

[6] L. Babai and R. Beals. A polynomial-time theory of black-box groups I. In *Groups St Andrews 1997 in Bath, I*, pp. 30–64. London Math. Soc. Lect. Notes 260, 1999.

[7] L. Babai, G. Cooperman, L. Finkelstein, E. M. Luks, and Á. Seress. Fast Monte Carlo algorithms for permutation groups. *J. Computer and System Sci.*, 50:296–308, 1995. (Prelim. 23rd STOC, 1991)

[8] L. Babai, A. J. Goodman, W. M. Kantor, E. M. Luks, and P. P. Pálfy. Short presentations for finite groups. *J. Algebra*, 194:79–112, 1997.

[9] L. Babai, W. M. Kantor, P. P. Pálfy, and Á. Seress. Black-box recognition of finite simple groups of Lie type by statistics of element orders. *J. Group Theory*, 5:383–401, 2002.

[10] L. Babai, P. P. Pálfy, and J. Saxl. On the number of p-regular elements in simple groups. *LMS J. Computation and Math.*, 2009, to appear.

[11] L. Babai and A. Shalev. Recognizing simplicity of black-box groups and the frequency of p-singular elements in affine groups. In *Groups and Comp. III*, pp. 39–62. deGruyter, 2001.

[12] L. Babai and E. Szemerédi. On the complexity of matrix group problems I. In *Proc. 25th FOCS*, pp. 229–240. IEEE Comp. Soc., 1984.

[13] E. Bach and J. O. Shallit. *Algorithmic Number Theory I: Efficient Algorithms*. MIT Press, 1996.

[14] R. Beals. Towards polynomial time algorithms for matrix groups. In *Groups and Computation II*, pp. 31–54. DIMACS, 1997.

[15] R. Beals and L. Babai. Las Vegas algorithms for matrix groups. In *Proc. 34th FOCS*, pp. 427–436. IEEE Comp. Soc., 1993.

[16] R. Beals, C. R. Leedham-Green, A. C. Niemeyer, C. E. Praeger, and Á. Seress. A black-box group algorithm for recognizing finite symmetric and alternating groups. *Trans. Amer. Math. Soc*, 355:2097–2113, 2003.

[17] J. Bray. An improved method for generating the centralizer of an involution. *Arch. Math.*, 74:241–245, 2000.

[18] P. A. Brooksbank. Fast constructive recognition of black-box unitary groups. *LMS J. Comput. Math.*, 6:162–197, 2003.

[19] P. A. Brooksbank. Fast constructive recognition of black-box symplectic groups. *J. Algebra*, 320:885–909, 2008.

[20] P. A. Brooksbank and W. M. Kantor. On constructive recognition of a black box $\mathrm{PSL}(d, q)$. In *Groups and Computation III*, pp. 95–111. deGruyter, 2001.

[21] P. A. Brooksbank and W. M. Kantor. Fast constructive recognition of black box orthogonal groups. *J. Algebra*, 300:256–288, 2006.

[22] F. Celler, C. R. Leedham-Green, S. Murray, A. C. Niemeyer, and E. A. O'Brien. Generating random elements of a finite group. *Comm. Alg.*, 23:4931–4948, 1995.

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

[24] J. H. Conway, R. T. Curtis, S. P. Norton, R. A. Parker, R. A. Wilson: *ATLAS of Finite Groups.* Clarendon Press, Oxford, 1985.

[25] J. D. Dixon. Generating random elements in finite groups. *Electronic J. Combinatorics*, 15/1:R94, 2008.

[26] W. Feit and J. Tits. Projective representations of minimum degree of group extensions. *Canad. J. Math.*, 30:1092–1102, 1978.

[27] M. L. Furst, J. Hopcroft, and E. M. Luks. Polynomial-time algorithms for permutation groups. In *Proc. 21st FOCS*, pp. 36–41, IEEE C.S., 1980.

[28] A. Gamburd and I. Pak. Expansion of product replacement graphs. *Combinatorica*, 26:411–429, 2006.

[29] P. E. Holmes, S. A. Linton, E. A. O'Brien, A. J. E. A. Ryba, and R. A. Wilson. Constructive membership in black-box groups. *J. Group Theory*, 11:747–763, 2008.

[30] D. F. Holt and S. Rees. Testing modules for irreducibility. *J. Austral. Math. Soc. Series A*, 57:1–16, 1994.

[31] A. Hulpke and Á. Seress. Short presentations for three-dimensional unitary groups. *J. Algebra*, 245:719–729, 2001.

[32] G. Ivanyos and K. Lux. Treating the exceptional cases of the MeatAxe. *Experimental Mathematics*, 9:373–381, 2000.

[33] W. M. Kantor. Sylow's theorem in polynomial time. *J. Computer Sys. Sci.*, 30:359–394, 1985.

[34] W. M. Kantor and Á. Seress. Black box classical groups. *Mem. AMS*, 149, Nr. 708:viii+168 pp., 2001.

[35] W. M. Kantor and Á. Seress. Computing with matrix groups. In *Groups, Combinatorics, and Geometry (Durham 2001)*, pp. 123–137. World Scientific, 2003.

[36] D. E. Knuth. Notes on efficient representation of perm groups. *Combinatorica*, 11:33–43, 1991.

[37] V. Landazuri and G. M. Seitz. On the minimal degrees of projective representations of the finite Chevalley groups. *J. Algebra*, 32:418–443, 1974.

[38] E. M. Luks. Lectures on polynomial-time computation in groups. Northeastern U. TR NU-CCS-90-16, 1990.

[39] E. M. Luks. Computing the composition factors of a permutation group in polynomial time. *Combinatorica*, 7:87–99, 1987.

[40] E. M. Luks. Computing in solvable matrix groups. In *Proc. 33rd FOCS*, pp. 111–120. IEEE C.S., 1992.

[41] E. M. Luks. Permutation groups and polynomial-time computation. In *Groups and Computation*, pp. 139–175. DIMACS, 1993.

[42] E. M. Luks and Á. Seress. Computing the Fitting subgroup and solvable radical of small-base permutation groups in nearly linear time. In *Groups and Computation II*, pp. 169–181. DIMACS, 1997.

[43] C. W. Parker and R. A. Wilson. Recognising simplicity of black-box groups. *Manuscript*, 2004.

[44] L. Rónyai. Computing the Structure of Finite Algebras. *J. Symb. Comp.*, 9:355–373, 1990.

[45] J. Rotman. *An Introduction to the Theory of Groups.* Springer, 1994.

[46] G. Seitz and A. E. Zalesskii. On the minimal degrees of projective representations of the finite Chevalley groups, II. *J. Algebra*, 158:233–243, 1993.

[47] Á. Seress. *Permutation Group Algorithms.* Cambridge University Press, 2003.

[48] P. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM J. Computing*, 26:1484–1509, 1977.

[49] C. C. Sims. Computational methods in the study of permutation groups. In *Computational problems in abstract algebra (Oxford, 1967)*, pp. 169–183. Pergamon Press, 1970.

[50] C. C. Sims. Computation with permutation groups. In *Proc. Second ACM Symp. on Symbolic and Algebraic Manipulation*, pp. 23–28. ACM, 1971.

[51] M. Suzuki. On a class of doubly transitive groups. *Ann. Math.*, 75:105–145, 1962.