J. Symbolic Computation (2001) **11**, 1–000

Practical Algorithms for Polycyclic Matrix Groups

Gretchen Ostheimer

Mathematics Department, Tufts University, Medford, MA 02155

(Received 11 October 2001)

In this paper we describe a suite of new algorithms for studying polycyclic matrix groups — algorithms for testing membership and for uncovering the polycyclic structure of the group. We also describe an algorithm for deciding whether or not a group is solvable, which, in the important context of subgroups of $\mathrm{GL}(n, \mathbf{Z})$, is equivalent to deciding whether or not a group is polycyclic. Algorithms were developed in Baumslag *et al.* (1991) for all of these problems, but the algorithms in this paper represent a first step toward finding practical algorithms: experiments show that they are fast enough to be useful in studying some reasonably complex examples using current technology.

1. Introduction

In this paper we describe a new algorithm for deciding membership in a polycyclic-by-finite matrix group: given a finite set X of $n \times n$ invertible matrices which generate a polycyclic-by-finite matrix group G under multiplication, and given another invertible $n \times n$ matrix a, we can decide whether or not a is an element of G. In the special case of a finitely generated abelian matrix group, we can also test membership constructively and we can find a presentation for the group. The decidability of these problems was established in Baumslag *et al.* (1991), and in that paper the question was raised concerning the existence of practical algorithms. This paper is a first step toward answering that question. Other authors have made significant progress in developing fast algorithms for these and related problems in different contexts: for a summary of work regarding practical algorithms for finitely presented polycyclic groups, see Sims (1994); for finite matrix groups see, for example, Luks (1992) and Beals (1996); for infinite nilpotent-by-finite matrix groups see Dixon (1985) and Beals (1996).

The basic structure of a polycyclic-by-finite matrix group G is well-known. Let P be a polycyclic subgroup of finite index in G. Then P contains a finite-index triangularizable group T, the commutator of which is a unipotent group U:

$$\underset{}{G} \underset{\text{finite}}{\quad\triangleright\quad} P \underset{\text{finite}}{\quad\triangleright\quad} T \underset{\text{abelian}}{\quad\triangleright\quad} U.$$

In this paper, we also describe a new algorithm for finding P, T and U and for finding a special sequence of elements (called a polycyclic generating sequence) which refines the normal series $P \triangleright T \triangleright U$ to a series with cyclic quotients:

$$P_1 \triangleright P_2 \triangleright \cdots \triangleright P_k \quad\triangleright\quad T_1 \triangleright T_2 \triangleright \cdots \triangleright T_l \quad\triangleright\quad U_1 \triangleright U_2 \triangleright \cdots \triangleright U_m.$$

0747–7171/90/000000 + 00 \$03.00/0

We can also decide whether or not a given group is triangularizable (Section 2.5) or simply solvable-by-finite (Section 2.8). A subgroup of $\mathrm{GL}(n, \mathbf{Z})$ is solvable if and only if it is polycyclic, so in $\mathrm{GL}(n, \mathbf{Z})$ we can also decide whether or not a given group is polycyclic-by-finite.

Experiments testing the speed of the algorithms in this paper show that the algorithms are certainly practical enough to be useful in studying some reasonably complex triangularizable examples (Section 4).

If F is a number field such that $[F : \mathbf{Q}] = d$, then the elements of F can be represented as $d \times d$ matrices over $\mathbf{Q}$. Therefore, all of the algorithms above can also be used to study subgroups of $\mathrm{GL}(n, F)$.

The algorithms depend on two structure theorems concerning subgroups of $\mathrm{GL}(n, \mathbf{Q})$: one describes the structure of triangularizable matrix groups (Section 2.5), and the other describes the structure of polycyclic-by-finite matrix groups (Section 2.7). In both cases, we show that through a change of basis over $\mathbf{Q}$, the groups have a certain block upper triangular form. Part of this paper is based on the author's Ph. D. thesis (Ostheimer, 1996a; Ostheimer, 1996b).

1.1. Notation and definitions

Throughout this paper let $\mathbf{Z}$ denote the ring of integers, $\mathbf{Q}$ the field of rationals, and $\mathbf{C}$ the field of complex numbers. Let R denote either $\mathbf{Z}$ or $\mathbf{Q}$. If p is a prime, then the field of p-adic numbers is denoted by $\mathbf{Q}_p$, its algebraic closure by $\overline{\mathbf{Q}_p}$, and the ring of p-adic integers by $\mathbf{Z}_p$. The field with p elements is denoted by F_p. If n is a positive integer, the group of $n \times n$ matrices invertible over R is denoted by $\mathrm{GL}(n, R)$ or by $\mathrm{GL}(V)$, where V is a free R-module of rank n. The ring of $n \times n$ matrices over R is denoted by $\mathrm{M}(n, R)$. If G is a subgroup of $\mathrm{GL}(n, R)$, then the *enveloping algebra* of G, $\mathrm{Env}(G)$, is defined to be the smallest $\mathbf{Q}$-subalgebra A of the algebra of $n \times n$ matrices over $\mathbf{Q}$ such that $A \supset G$.

Let F be any field, and let E be an extension field of F. If G is a subgroup of $\mathrm{GL}(n, F)$, then G is *triangularizable over* E if there is a basis for E^n relative to which every matrix in G is upper triangular, and G is *triangularizable* if it is triangularizable over the algebraic closure of F. If G is a subgroup of $\mathrm{GL}(n, F)$ and if G is triangularizable over some extension field E of F, then there exists a finite extension L of F over which G is triangularizable. (See p. 33 of Segal (1983).) Therefore, a subgroup G of $\mathrm{GL}(n, \mathbf{Q})$ is triangularizable if and only if it is triangularizable over $\mathbf{C}$.

Let $\mathrm{Tr}_1(n, R)$ denote the group of upper triangular matrices with entries in R and 1's along the diagonal. Elements of $\mathrm{Tr}_1(n, R)$ are called *unitriangular*. An element of $\mathrm{GL}(n, R)$ is *unipotent* if all of its eigenvalues are 1. If $G < \mathrm{GL}(n, R)$ consists entirely of unipotent elements, then G is said to be *unipotent*. In this case there is a basis for R^n with respect to which G is unitriangular. (The case when $R = \mathbf{Q}$ is proved as Corollary 1.21 in Wehrfritz (1973), and the case when $R = \mathbf{Z}$ follows easily.) If G is a subgroup of $\mathrm{GL}(n, R)$, then there is a unique normal unipotent subgroup of G which is maximal among all such subgroups of G. (See 13.40 of Wehrfritz (1973).) This subgroup is called the *unipotent part* of G.

If $x \in \mathbf{Z}^k$ and $g_1, \ldots, g_k \in \mathrm{GL}(n, R)$, then g^x denotes $g_1^{x_1} \cdots g_k^{x_k}$. The identity element of $\mathrm{GL}(n, R)$ is denoted by 1. The commutator subgroup of G is denoted by G'.

1.2. Basic algorithms for polycyclic groups

Chapter 9 in Sims (1994) gives a good introduction to polycyclic groups. A group G is *polycyclic* if there is a sequence of subgroups $G_1, \ldots, G_{k+1}$ for G such that

$$G = G_1 \triangleright G_2 \triangleright \cdots \triangleright G_k \triangleright G_{k+1} = 1,$$

where for each i, G_i/G_{i+1} is cyclic. In this case, if g_iG_{i+1} is a generator for G_i/G_{i+1}, then $g_1, \ldots, g_k$ is called a *polycyclic generating sequence* for G. A group G is polycyclic if and only if G is solvable and all of the subgroups of G are finitely generated.

Polycyclic groups are finitely presented. It is often convenient to assume that a polycyclic group is given by a special presentation known as a *consistent polycyclic presentation.* Intuitively, a consistent polycyclic presentation is one from which the polycyclic structure of the group is easily gleaned. In particular, the generators in a consistent polycyclic presentation form a polycyclic generating sequence for G. (See Section 9.6 of Sims (1994) for a precise definition.)

Throughout this paper we rely on basic algorithms in Sims (1994) for working with polycyclic groups given by consistent polycyclic presentations. For example, practical algorithms exist for testing membership in a subgroup of a polycyclic group, and for finding the kernel of a homomorphism between two polycyclic groups, providing these groups are given by consistent polycyclic presentations. We also rely on the fact that a polycyclic group satisfies the ascending chain condition on subgroups. In Section 2, we will refer in some detail to two algorithms in Section 9.6 of Sims (1994) for testing membership in a subgroup H of G: POLY_SUBGROUP and POLY_MEMBER. POLY_SUBGROUP calculates a polycyclic generating sequence for H. POLY_MEMBER uses such a sequence to test membership in H. POLY_MEMBER is constructive; that is, given a polycyclic generating sequence $h_1, \ldots, h_l$ for H and $y \in H$, POLY_MEMBER can find an x in $\mathbf{Z}^l$ such that $h^x = y$.

1.3. Facts about $\mathrm{GL}(n, \mathbf{Z}_p)$

We first establish a connection between finitely generated subgroups of $\mathrm{GL}(n, \mathbf{Q})$ and subgroups of $\mathrm{GL}(n, \mathbf{Z}_p)$. If G is a finitely generated subgroup of $\mathrm{GL}(n, \mathbf{Q})$, then the subring S of $\mathbf{Q}$ generated by the entries of the elements of G is finitely generated as a ring. Therefore, there exists a prime p such that if $r \in S$, then $r = \frac{u}{v}$, where $u, v \in \mathbf{Z}$ and $(v, p) = 1$. Notice that G can be embedded in $\mathrm{GL}(n, \mathbf{Z}_p)$: the obvious embedding of $\mathbf{Q}$ into $\mathbf{Q}_p$ maps S into $\mathbf{Z}_p$, and thus there exists a group embedding of G into $\mathrm{GL}(n, \mathbf{Z}_p)$. As we saw in Section 1.1, G is triangularizable over $\overline{\mathbf{Q}_p}$ if and only if G is triangularizable over $\mathbf{C}$.

In Dixon (1985) Dixon gives a sufficient condition for a subgroup of $\mathrm{GL}(n, \mathbf{Z}_p)$ to be triangularizable over $\overline{\mathbf{Q}_p}$. Consider the ring homomorphism

$$\mathbf{Z}_p \to \mathbf{Z}_p/p\mathbf{Z}_p \cong F_p$$

and its extension to a group homomorphism $\theta : \mathrm{GL}(n, \mathbf{Z}_p) \to \mathrm{GL}(n, F_p)$. The kernel of this homomorphism is called the *principal p-congruence subgroup* of $\mathrm{GL}(n, \mathbf{Z}_p)$.

Lemma 1.1. *Let p be an odd prime, and let C be the principal p-congruence subgroup of $GL(n, \mathbf{Z}_p)$. If H is a subgroup of C, and H is solvable-by-finite, then H is triangularizable (over $\overline{\mathbf{Q}_p}$).*

PROOF. In Dixon (1985) Dixon shows that H is connected in the Zariski topology of $\mathrm{M}(n, \overline{\mathbf{Q}_p})$. By the Lie-Kolchin theorem (as stated in Lemma 8.6 of Dixon (1971)), H is triangularizable over $\overline{\mathbf{Q}_p}$. $\square$

2. Algorithms for Matrix Groups

In this section we develop a suite of algorithms for studying polycyclic-by-finite matrix groups. We will start by considering two special cases, abelian groups (Sections 2.1 and 2.4) and unitriangular groups (Sections 2.2 and 2.3). From there, we will generalize to triangularizable groups (Sections 2.5 and 2.6) and then to polycyclic-by-finite groups (Sections 2.7, 2.8 and 2.9).

2.1. Abelian groups embedded in fields

In order to develop an algorithm to find a presentation for an abelian subgroup G of $\mathrm{GL}(n, \mathbf{Q})$, we consider two special cases. Let A be the enveloping algebra of G. In the first case A is a field, and in the second case $G \leq \mathrm{Tr}_1(n, \mathbf{Q})$. In Section 2.4 we prove a structure theorem for abelian matrix groups that shows that these two cases are sufficient. The first case is discussed in this section, and the second case is discussed in Sections 2.2 and 2.3.

Let G be an abelian subgroup of $\mathrm{GL}(n, \mathbf{Q})$ given by a finite set $\{g_1, \ldots, g_k\}$ of generators. Suppose that A, the enveloping algebra of G, is a field. We want to find generators for the subgroup H of $\mathbf{Z}^k$ consisting of elements x such that $g^x = 1$. This problem was solved in Ge (1993). Ge assumes that the field is described by d^3 structure constants, where d is the dimension of A as a vector space over $\mathbf{Q}$. More precisely, in order to use his algorithm, first find a basis $b_1, \ldots, b_d$ for A over $\mathbf{Q}$, and then find constants c_{ijk} such that for all i, j,

$$b_i b_j = \sum_k c_{ijk} b_k.$$

Ge proves that the run-time of his algorithm is polynomial in the length of the input.

Once we have an algorithm to find H, we can test membership in G constructively, i.e., we can decide whether or not $a \in G$ and, if so, we can express a in terms of the given generators for G. To do so, first verify that a commutes with $g_1, \ldots, g_k$. Then rename a as g_{k+1} and find generators for $\widetilde{H}$, the subgroup of vectors x in $\mathbf{Z}^{k+1}$ such that $g^x = 1$. If there exists an element x of $\widetilde{H}$ such that $x_{k+1} = -1$, then let y be the vector consisting of the first k components of x for then $a = g^y$. Otherwise, $a \notin G$.

Ge makes no claims concerning the practicality of his algorithm. However, we independently discovered an algorithm to find a presentation in the special case when $G \leq \mathrm{GL}(n, \mathbf{Z})$. Our algorithm is similar to that of Ge, and experiments show that our algorithm is practical.

We continue the notation established at the beginning of this section, but now we assume furthermore that $G < \mathrm{GL}(n, \mathbf{Z})$. Let O_A be the ring of algebraic integers of A, and let O_A^* be the group of units of O_A. If r is the number of real embeddings of A, and $2c$ is the number of complex embeddings of A, then the usual logarithmic homomorphism ρ maps $A \setminus \{0\}$ to $\mathbf{R}^s$, where $s = r + c$. The image of O_A^* under ρ is a lattice L in a hyperplane of $\mathbf{R}^s$. (See Theorem 4.9.7 of Cohen (1995).) Let K be the subgroup of $\mathbf{Z}^k$ consisting of elements x such that $\rho(g^x) = 0$. An algorithm for finding generators for K is described in Buchmann and Pohst (1989). The algorithm requires as input a lower bound M for the length of the shortest nonzero vector in L. If A is a low degree extension of $\mathbf{Q}$, then accurate bounds on the length of a shortest nonzero vector can be obtained by finding a system of fundamental units for the group of units for O_A. (See Section 6.5 of Cohen (1995).) When this is impractical, the following lemma from Dobrowolski (1979) can be used to obtain a lower bound.

LEMMA 2.1. *Let α be a nonzero algebraic integer of degree d. Let $\{\alpha_1, \ldots, \alpha_d\}$ be the algebraic conjugates of α. If*

$$\prod_1^d max\{1, |\alpha_j|\} \leq 1 + \frac{1}{1200}\left(\frac{\log\log d}{\log d}\right)^3,$$

then α is a root of unity.

The run-time of the algorithm in Buchmann and Pohst (1989) is polynomial in $s+k$ and the size of the input data. The authors illustrate the practicality of their algorithm by finding (for real quartic fields with discriminants below 100000) a multiplicative relation between four given units.

Having found a presentation for $\rho(G)$ (that is, having found generators $k_1, \ldots, k_j$ for K), let us turn our attention back to the problem of finding a presentation for G (that is, of finding generators for H). The kernel of ρ is the torsion subgroup T of O_A^*. Let $m = |T|$, and suppose $T = \langle t \rangle$. Let ϕ be the Euler ϕ-function. The minimal polynomial f of t has degree $\phi(m)$. Therefore, $\phi(m) \leq n$, and the following lemma (suggested to the author by Jerrold Tunnell) gives a simple bound on the order of T.

LEMMA 2.2. $m \leq 2.4\phi(m)^{4/3}$.

PROOF. Let $m = p_1^{\alpha_1} \cdots p_j^{\alpha_j}$ be the prime factorization of m. If $j = 1$, then

$$\begin{aligned} \phi(m) &= m(1 - 1/p_1) \\ &\geq m/2. \end{aligned}$$

If $j > 1$, then

$$\begin{aligned} \phi(m) &= m(1 - 1/p_1) \cdots (1 - 1/p_j) \\ &= m^{3/4}(1 - 1/p_1)p_1^{\alpha_1/4} \cdots (1 - 1/p_j)p_j^{\alpha_j/4} \\ &\geq m^{3/4}(1 - 1/2)2^{1/4}(1 - 1/3)3^{1/4}. \end{aligned}$$

In either case, the result now follows. □

Therefore, T is a cyclic group of small order. A finite set of generators for $T \cap G$ is now easily found:

$$T \cap G = \langle g^{k_i} : i = 1, \ldots, j \rangle.$$

It is not hard in practice to find a cyclic generator for $T \cap G$. Since H is the kernel of the map from K to $T \cap G$ taking x to g^x, we can use the techniques of Section 9.6 of Sims (1994) to find generators for H.

Ge uses a technique which is similar to that described above, but he uses complex logarithms instead of real logarithms. He shows that the length of the shortest nonzero vector in the resulting lattice is bounded below by 1/16.

2.2. Unitriangular subgroups of $\mathrm{GL}(n, \mathbf{Z})$

Let G be a subgroup of $\mathrm{Tr}_1(n, \mathbf{Z})$ given by a finite set $\{g_1, \ldots, g_k\}$ of generators. In order to test membership in G, we rely on algorithms which have been developed for working with polycyclic groups given by consistent polycyclic presentations. (See Section 1.2 for a description of POLY_SUBGROUP and POLY_MEMBER.) Since $\mathrm{Tr}_1(n, \mathbf{Z})$ is a finitely generated nilpotent group, it is also polycyclic. (See Example 2.1 and Proposition 3.4 in Chapter 9 of Sims (1994).) It is easy to find a consistent polycyclic presentation $\langle X | R\rangle$ for $\mathrm{Tr}_1(n, \mathbf{Z})$: Example 4.1 in Chapter 9 of Sims (1994) is easily generalized. It is also easy to express a given unitriangular matrix in terms of the generators in X. Use POLY_SUBGROUP to calculate a polycyclic generating sequence $h_1, \ldots, h_l$ for G and POLY_MEMBER to then test membership in G. In order to test our algorithm on examples, we implemented versions of POLY_SUBGROUP and POLY_MEMBER for this specific context. See Section 4 for further information.

When G is both unitriangular and abelian, we can also find a presentation for G in terms of $g_1, \ldots, g_k$, i.e., we can find generators for the subgroup H of $\mathbf{Z}^k$ consisting of elements z such that $g^z = 1$. We do so as follows. Since G is free abelian, we can use POLY_SUBGROUP to find a basis $h_1, \ldots, h_l$ for G. There is a homomorphism μ from $\mathbf{Z}^k$ to $\mathbf{Z}^l$ taking u in $\mathbf{Z}^k$ to the vector v in $\mathbf{Z}^l$ such that $h^v = g^u$. An integer matrix representing μ can be computed as follows: if e_i is the i-th vector in the usual basis for $\mathbf{Z}^k$, then POLY_MEMBER can be used to find $\mu(e_i)$, the vector in $\mathbf{Z}^l$ such that $g_i = h^{\mu(e_i)}$. The kernel H of μ is then easy to compute. As we saw in Section 2.1, if we can find a presentation for G, then it follows that our membership test for G can be made constructive.

2.3. Unitriangular subgroups of $\mathrm{GL}(n, \mathbf{Q})$

Let G be a finitely generated subgroup of $\mathrm{Tr}_1(n, \mathbf{Q})$. We reduce the problem of testing membership in G to the problem of testing membership in a subgroup of $\mathrm{Tr}_1(n, \mathbf{Z})$. In Segal (1983) (Lemma 2, Chapter 6), Segal shows that if $G = \langle g_1, \ldots, g_k\rangle$ is a finitely generated subgroup of $\mathrm{Tr}_1(n, \mathbf{Q})$, then it is easy to find an x in $\mathrm{GL}(n, \mathbf{Q})$ such that $x^{-1}Gx \leq \mathrm{Tr}_1(n, \mathbf{Z})$: find a natural number m such that mg_i is an integer matrix for all $1 \leq i \leq k$, and then let

$$x = \begin{pmatrix} 1 & 0 & \cdot & \cdot & 0 & 0 \\ 0 & m & 0 & \cdot & \cdot & 0 \\ \cdot & 0 & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & \cdot & \cdot & \cdot & \cdot & 0 \\ 0 & 0 & \cdot & \cdot & 0 & m^{n-1} \end{pmatrix}.$$

Thus, membership testing in a finitely generated subgroup of $\mathrm{Tr}_1(n, \mathbf{Q})$ is reduced to the analogous problem in $\mathrm{Tr}_1(n, \mathbf{Z})$. As with subgroups of $\mathrm{Tr}_1(n, \mathbf{Z})$, if G is also abelian, then we can make our membership test constructive, and we can find a presentation for

G. Alternative methods for working with abelian subgroups of $\mathrm{Tr}_1(n, \mathbf{Q})$ are described in Babai *et al.* (1996).

2.4. Abelian groups

The following proposition indicates how the problem of finding presentations for finitely generated abelian groups can be reduced to the corresponding problems for abelian groups embedded in fields and finitely generated abelian unitriangular groups.

Proposition 2.3. *Let G be an abelian subgroup of $GL(n, R)$. There exist a basis B for R^n and positive integers $n_1, \ldots, n_r$ such that $n_1 + \cdots + n_r = n$ and all matrices in G have the following block structure with respect to B:*

$$g = \begin{pmatrix} a_1 & * & \cdot & \cdot & * & * \\ 0 & a_2 & * & \cdot & \cdot & * \\ \cdot & 0 & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & \cdot & \cdot & \cdot & \cdot & * \\ 0 & 0 & \cdot & \cdot & 0 & a_r \end{pmatrix},$$

where for $1 \le i \le r$, a_i is an $n_i \times n_i$ matrix, and if G_i is the image of G under the map that takes g to a_i, then the enveloping algebra of G_i is a field.

Proof. Let A be the enveloping algebra of G, and suppose A is not a field. There exists a zero divisor a of A. (If $R = \mathbf{Z}$, then ensure $a \in \mathrm{M}(n, \mathbf{Z})$ by clearing denominators.) Let $V = \{v \in R^n \mid va = 0\}$. Since $0 < V < R^n$ and V is invariant under G, by induction we can choose a basis D for V relative to which $G|_V$ has the desired form. There is a complement W to V in R^n. (If $R = \mathbf{Z}$, then R^n/V is torsion-free.) Choose a basis for W and thereby a basis for R^n/V. Consider the map τ that takes an element of G to the matrix representing its action on R^n/V relative to this basis. By induction again, we can find a basis C for W such that relative to the corresponding basis for R^n/V, $\tau(G)$ has the desired form. Then C, D is a basis for R^n that satisfies the criteria of the proposition. □

The problem of computing a basis B satisfying the criteria of Proposition 2.3 is closely related to the problem of finding the primary decomposition of the radical $J(A)$ of A, where A is the enveloping algebra of G. Given a basis B satisfying the criteria of Proposition 2.3, we can compute vector space generators for M_i, the kernel of the map that takes a matrix g to a_i. Then M_i is a maximal ideal, and

$$J(A) = \bigcap_{i=1}^{r} M_i.$$

Conversely, if we are given ideal generators for the maximal ideals $M_1, \ldots, M_l$ in the primary decomposition of $J(A)$, then it follows from the uniqueness of the primary decomposition and Proposition 2.3 that there exists an i such that the nullspace V_i of M_i is a nontrivial, invariant subspace on which the enveloping algebra of G is a field.

The development of practical algorithms for finding primary decompositions is an active area of research. In Friedl and Rónyai (1985), for example, the authors show that there exists a polynomial time algorithm for computing the primary decomposition of the radical of a matrix algebra. In Eisenbud *et al.* (1992), the authors discuss the problem of finding a practical algorithm for computing the primary decomposition of an ideal in a polynomial ring. IBM's symbolic computation package AXIOM has a function for finding the primary decomposition for a zero-dimensional ideal in a polynomial ring. Further research on practical algorithms for computing the primary decomposition will shed light on how best to achieve the reduction we seek here. One method is to mimic the proof of Proposition 2.3, relying on methods in Eisenbud *et al.* (1992) or Friedl and Rónyai (1985) for deciding whether or not A is a field and, if it is not, for finding a zero divisor a in A. Note that when $R = \mathbf{Z}$, we find a basis for $\mathbf{Z}^n$ relative to which G has the desired form.

Now let us assume that G is an abelian subgroup of $\mathrm{GL}(n, R)$ given by a finite set $\{g_1, \ldots, g_k\}$ of generators, and that we have found a basis for R^n satisfying the criteria of Proposition 2.3. We now show that we can obtain a presentation for G, i.e., we can find generators for the subgroup H of $\mathbf{Z}^k$ consisting of elements z such that $g^z = 1$. Let ψ be the group homomorphism given below.

$$g = \begin{pmatrix} a_1 & * & \cdot & \cdot & * & * \\ 0 & a_2 & * & \cdot & \cdot & * \\ \cdot & 0 & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & \cdot & \cdot & \cdot & \cdot & * \\ 0 & 0 & \cdot & \cdot & 0 & a_r \end{pmatrix} \longmapsto \begin{pmatrix} a_1 & 0 & \cdot & \cdot & 0 & 0 \\ 0 & a_2 & 0 & \cdot & \cdot & 0 \\ \cdot & 0 & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & \cdot & \cdot & \cdot & \cdot & 0 \\ 0 & 0 & \cdot & \cdot & 0 & a_r \end{pmatrix}$$

Let π_i be the group homomorphism that takes g to a_i. As we saw in Section 2.1, we can find generators for the subgroup K_i of $\mathbf{Z}^k$ consisting of elements z such that $\pi_i(g^z) = 1$, and hence we can find a basis $k_1, \ldots, k_s$ for

$$K = \bigcap_{i=1}^{r} K_i = \{x \in \mathbf{Z}^k \mid \psi(g^x) = 1\}.$$

Let S be the $s \times k$ matrix whose i-th row is k_i. The image of K (under the map that takes x to g^x) is an abelian subgroup of $\mathrm{Tr}_1(n, \mathbf{Z})$ and hence is free. As described in Section 2.2, we can use POLY_SUBGROUP to find a basis $h_1, \ldots, h_l$ for the image of K. Let μ be the homomorphism from $\mathbf{Z}^s$ to $\mathbf{Z}^l$ that takes the standard basis vector $e_i \in \mathbf{Z}^s$ to the vector $v \in \mathbf{Z}^l$ such that $g^{k_i} = h^v$. An integer matrix representing μ can be found by using POLY_MEMBER to compute $\mu(e_i)$. Use Hermite normal form calculations to find a matrix T whose rows form a basis for the kernel of μ. The rows of TS then form a generating set for H. As described in Section 2.1, it follows that we can also test membership in G constructively.

2.5. A structure theorem for triangularizable groups

Recall that an abelian subgroup G of $\mathrm{GL}(n, R)$ is triangularizable. (See p. 29 of Segal (1983).) We can use this fact to construct other examples of triangularizable subgroups. Let $n = n_1 + \cdots + n_r$, and let G_i be an abelian subgroup of $\mathrm{GL}(n_i, R)$. Choose as our generators for G matrices that have the following block structure:

$$\begin{pmatrix} a_1 & * & \cdot & \cdot & * & * \\ 0 & a_2 & * & \cdot & \cdot & * \\ \cdot & 0 & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & \cdot & \cdot & \cdot & \cdot & * \\ 0 & 0 & \cdot & \cdot & 0 & a_r \end{pmatrix},$$

where $a_i \in G_i$ for all i. Clearly, G is triangularizable. In this section we show that given a finite set of generators for a triangularizable subgroup G of $\mathrm{GL}(n, R)$, we can find a basis for R^n relative to which G has this form. Furthermore, we describe an algorithm for deciding whether or not a subgroup of $\mathrm{GL}(n, R)$ given by a finite set of generators is triangularizable.

Clearly, a triangularizable subgroup is unipotent-by-abelian: after a change of basis over $\mathbf{C}$ the group is upper triangular, and the commutator of two upper triangular matrices is an upper triangular matrix with 1's on the diagonal. We will now show that all unipotent-by-abelian subgroups have the desired form.

For any positive integer n and for any subgroup L of $\mathrm{GL}(n, R)$, let

$$V(L) = \{v \in R^n \mid vl_1l_2 = vl_2l_1 \, \forall \, l_1, \, l_2 \in L\}.$$

The following lemma establishes some obvious properties of $V(L)$.

Lemma 2.4. *Let L be a subgroup of $GL(n, R)$. Then $V(L) \leq R^n$ is an R-submodule of R^n invariant under L, the image of L under the map to $GL(V(L))$ is abelian, and $V(L)$ has a complement in R^n.*

Proof. $V(L)$ is invariant under the action of L since if $v \in V(L)$ and $x, g, h \in L$, then

$$vx(gh - hg) = v((xg)h - h(xg)) + (v(hx - xh))g = 0.$$

If $R = \mathbf{Z}$, then $\mathbf{Z}^n/V(L)$ is torsion-free. Hence for $R = \mathbf{Z}$ or $\mathbf{Q}$, $V(L)$ has a complement in R^n. The rest follows easily. □

Let $V_0(L) = 0$. Inductively, define a chain

$$0 = V_0(L) \leq V_1(L) \leq V_2(L) \leq \cdots$$

of L-invariant R-submodules of R^n as follows. Let $\overline{L}$ be the image of the map taking L to $\mathrm{GL}(R^n/V_i(L))$. Let $V_{i+1}(L)$ be the pullback of $V(\overline{L})$ under the map $R^n \to R^n/V_i(L)$. Then $V_{i+1}(L)$ is L-invariant. To see this, let $x \in V_{i+1}(L)$ and $g \in L$. Then $x + V_i(L) \in V(\overline{L})$. By Lemma 2.4 $xg + V_i(L) \in V(\overline{L})$ and hence $xg \in V_{i+1}(L)$.

For unipotent-by-abelian matrix groups, more can be said.

Lemma 2.5. *Let G be a unipotent-by-abelian subgroup of $GL(n, R)$. Then $V(G) \neq 0$.*

Proof. Recall (from Section 1.1) that since G' is unipotent, there is a basis for R^n with respect to which G' is unitriangular. Therefore, there exists a nonzero element x of R^n such that $xg = x$ for all $g \in G'$. For all g, h in G,

$$x(gh - hg) = x(ghg^{-1}h^{-1} - 1)hg = 0.$$

□

LEMMA 2.6. *Let G be a unipotent-by-abelian subgroup of $GL(n,R)$. There exists a positive integer r such that $V_r(G) = R^n$. For all $0 \le i \le r-1$, $V_i(G) < V_{i+1}(G)$, $V_i(G)$ has a complement in R^n, and the image of G under the map to $GL(V_{i+1}(G)/V_i(G))$ is abelian.*

PROOF. Assume that $V_i(G) \neq R^n$. Let $\overline{G}$ be the image of the map taking G to $\mathrm{GL}(R^n/V_i(G))$. Then $\overline{G}$ is unipotent-by-abelian since $\overline{G'} = \overline{G}'$ is unipotent. Since $V(\overline{G}) \neq 0$, $V_i(G) < V_{i+1}(G)$. Therefore, there exists a positive integer r such that $V_r(G) = R^n$. The quotient $R^n/V_{i+1}(G)$ is isomorphic to $(R^n/V_i(G))/V(\overline{G})$, and this latter quotient is free since $V(\overline{G})$ has a complement in $R^n/V_i(G)$. Therefore $V_{i+1}(G)$ has a complement in R^n. It follows from the definition of $V_{i+1}(G)$ that the image of G under the map to $\mathrm{GL}(V_{i+1}(G)/V_i(G))$ is abelian. □

Our structure theorem for unipotent-by-abelian matrix groups now follows immediately.

PROPOSITION 2.7. *Let G be a unipotent-by-abelian subgroup of $GL(n,R)$. There exist a basis B for R^n and positive integers $n_1, \ldots, n_r$ such that $n_1 + \cdots + n_r = n$ and all matrices in G have the following block structure with respect to B:*

$$g = \begin{pmatrix} a_1 & * & \cdot & \cdot & * & * \\ 0 & a_2 & * & \cdot & \cdot & * \\ \cdot & 0 & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & \cdot & \cdot & \cdot & \cdot & * \\ 0 & 0 & \cdot & \cdot & 0 & a_r \end{pmatrix},$$

where for $1 \le i \le r$, a_i is an $n_i \times n_i$ matrix, and the image of G under the map that takes g to a_i is abelian.

Note that when $R = \mathbf{Z}$, we find a basis for $\mathbf{Z}^n$ relative to which G has the desired form.

The following corollary is an immediate consequence.

COROLLARY 2.8. *Let G be a subgroup of $GL(n,R)$. Then G is triangularizable if and only if G is unipotent-by-abelian.*

Let A be the enveloping algebra of G, where G is a triangularizable subgroup of $\mathrm{GL}(n,R)$. Just as in the case of abelian matrix groups (Section 2.4), the problem of computing a basis for R^n satisfying the criteria of Proposition 2.7 is closely related to the problem of expressing the radical $J(A)$ as the intersection of maximal ideals. Let π_i be the map that takes g to a_i. As we saw in Section 2.4, we may assume that for each i, the enveloping algebra of $\pi_i(G)$ is a field. In this case, the kernel M_i of the algebra homomorphism extending π_i is a maximal ideal, and

$$J(A) = \bigcap_{i=1}^{r} M_i.$$

Further research into practical algorithms for finding primary decompositions will shed light on how best to compute a desired basis for R^n. One possibility is to use an algorithm based on Lemma 2.6 both to decide whether or not a subgroup G of $\mathrm{GL}(n,R)$ (given by a finite set of generators) is triangularizable, and, if so, to find a desired basis for

R^n. We can do so as follows. Find a basis $\{b_1, \ldots, b_d\}$ for A as a vector space over $\mathbf{Q}$. (If $R = \mathbf{Z}$, then choose the b_i's in $\mathrm{GL}(n, \mathbf{Z})$.) Then $V(G) = \bigcap_{i,j} W_{i,j}$, where $W_{i,j} = \{v \in R^n \mid v(b_i b_j - b_j b_i) = 0\}$. If $V(G) = 0$, then G is not triangularizable. Otherwise, find a complement W to $V(G)$ in R^n. Now proceed inductively to decide whether or not $\overline{G}$, the image of the map taking G to $\mathrm{GL}(R^n/V(G))$, is triangularizable, and, if so, to find a basis $v_1, \ldots, v_j$ for W such that $\overline{G}$ has the desired form with respect to $v_1 + V(G), \ldots, v_j + V(G)$.

2.6. Testing membership in triangularizable groups

Let G be a triangularizable, polycyclic subgroup of $\mathrm{GL}(n, R)$ given by a finite generating set, and let $a \in \mathrm{GL}(n, R)$. We will now reduce the problem of deciding whether or not $a \in G$ to two cases considered previously, namely, when G is finitely generated and abelian, or when G is finitely generated and unitriangular. Note that a triangularizable subgroup of $\mathrm{GL}(n, \mathbf{Z})$ is always polycyclic, but this is not the case for triangularizable subgroups of $\mathrm{GL}(n, \mathbf{Q})$. For example, let G be the subgroup of $\mathrm{GL}(2, \mathbf{Q})$ generated by

$$x = \begin{pmatrix} 2 & 0 \\ 0 & 1 \end{pmatrix} \text{ and } k = \begin{pmatrix} 1 & 1 \\ 0 & 1 \end{pmatrix}.$$

Then G' is abelian. Furthermore,

$$G' = \left\{ \begin{pmatrix} 1 & r \\ 0 & 1 \end{pmatrix} \mid r \in \mathbf{Z}[1/2] \right\}.$$

Therefore, G' is not finitely generated.

As we saw in the previous section, we may assume that we have found a basis for R^n satisfying the criteria of Proposition 2.7. We may also assume that a has a block structure like that of G with respect to this basis. Let ψ be the map from $\langle G, a \rangle$ to $\mathrm{GL}(n, R)$ given below.

$$\begin{pmatrix} a_1 & * & \cdot & \cdot & * & * \\ 0 & a_2 & * & \cdot & \cdot & * \\ \cdot & 0 & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & \cdot & \cdot & \cdot & \cdot & * \\ 0 & 0 & \cdot & \cdot & 0 & a_r \end{pmatrix} \longmapsto \begin{pmatrix} a_1 & 0 & \cdot & \cdot & 0 & 0 \\ 0 & a_2 & 0 & \cdot & \cdot & 0 \\ \cdot & 0 & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & \cdot & \cdot & \cdot & \cdot & 0 \\ 0 & 0 & \cdot & \cdot & 0 & a_r \end{pmatrix}$$

As we saw in Section 2.4, we can test membership in $\psi(G)$ constructively, i.e., we can decide whether or not $\psi(a) \in \psi(G)$, and, if so, we can find $g \in G$ such that $\psi(a) = \psi(g)$. Thus we are left with the problem of deciding whether or not ag^{-1} is in the intersection H of G and the kernel of ψ.

Notice that H is precisely the unipotent part U of G. Clearly, H is a normal, unipotent subgroup of G, and hence $H \leq U$. As before, let G_i be the image of G under the map π_i that takes a matrix to the i-th block along the block diagonal, and let n_i be the size of the matrices in G_i (so that $G_i \leq \mathrm{GL}(n_i, R)$). Suppose g is an element of U. Let $a_i = \pi_i(g)$. Since g is unipotent, $(a_i - 1)^{n_i} = 0$. But A_i, the enveloping algebra of G_i, is a field, and hence $a_i - 1 = 0$. Therefore, $g \in H$.

Find a finite set of generators for H as follows. Recall from Section 2.4 that we can find a presentation for $\psi(G)$ in terms of our given generators, i.e., we can find generators $k_1, \ldots, k_s$ for the subgroup K of vectors v in $\mathbf{Z}^k$ such that $\psi(g^v) = 1$. Let X be a finite set of normal generators for G', and let $\widetilde{H} = \langle g^{k_1}, \ldots, g^{k_s}, X\rangle$. Then H is the normal closure of $\widetilde{H}$ in G. By conjugating elements in $\widetilde{H}$ by $g_1, \ldots, g_k$, keep building up $\widetilde{H}$ until $\widetilde{H}$ is normal in G, in which case $\widetilde{H} = H$. (Since H satisfies the ascending chain condition on subgroups, we do not need to conjugate by $g_1^{-1}, \ldots, g_k^{-1}$.) This process is guaranteed to terminate since G is assumed to be polycyclic. Now we have generators for H, so we can use the methods of Section 2.3 to decide whether or not $ag^{-1} \in H$.

2.7. A structure theorem for polycyclic-by-finite groups

Let G be a polycyclic-by-finite subgroup of $\mathrm{GL}(n, R)$. In this section we prove the following proposition concerning the structure of G.

PROPOSITION 2.9. *Let G be a polycyclic-by-finite subgroup of $GL(n, R)$. There exist a basis B for R^n and positive integers $n_1, \ldots, n_r$ such that $n_1 + \cdots + n_r = n$ and all matrices in G have the following block structure with respect to B*

$$g = \begin{pmatrix} a_1 & * & \cdot & \cdot & * & * \\ 0 & a_2 & * & \cdot & \cdot & * \\ \cdot & 0 & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & \cdot & \cdot & \cdot & \cdot & * \\ 0 & 0 & \cdot & \cdot & 0 & a_r \end{pmatrix},$$

where for $1 \le i \le r$, a_i is an $n_i \times n_i$ matrix and the image of G under the map that takes g to a_i is abelian-by-finite.

PROOF. There exists a subgroup $H \lhd G$ such that $|G : H| < \infty$ and H is triangularizable. (See Theorem 3, Chapter 2 of Segal (1983) as well as Lemma 1.1 of this paper.) Let $V_0(H), \ldots, V_r(H)$ be defined as in Section 2.5. By Lemma 2.6, it suffices to show that each $V_i(H)$ is invariant under G. Since $V_i(H)$ is invariant under H, and $H \lhd G$, it follows easily that for all $g \in G$, $V_i(H)g$ is invariant under H.

Next we show that for all i, H is abelian on $V_i(H)g/V_{i-1}(H)g$. Let $x \in V_i(H)$ and $h_1, h_2 \in H$. Then there exist $h_1', h_2' \in H$ such that $gh_1 = h_1'g$ and $gh_2 = h_2'g$. Therefore

$$xgh_1h_2 - xgh_2h_1 = xh_1'gh_2 - xh_2'gh_1 = xh_1'h_2'g - xh_2'h_1'g \in V_{i-1}(H)g.$$

We now show by induction on i that $V_i(H)g = V_i(H)$. This is trivial in the case when $i = 0$. By induction we may assume that $V_{i-1}(H)g = V_{i-1}(H)$. Then H is abelian on $V_i(H)g/V_{i-1}(H)$. Therefore $V_i(H)g \le V_i(H)$ (by our definition of $V_i(H)$), and hence $V_i(H)g = V_i(H)$. The proposition now follows. □

2.8. Deciding if a matrix group is solvable-by-finite

Let G be a subgroup of $\mathrm{GL}(n, R)$ given by a finite set $\{g_1, \ldots, g_k\}$ of generators. Let p be an odd prime which does not divide the denominator of any of the entries of $g_1, \ldots, g_k, g_1^{-1}, \ldots, g_k^{-1}$. (For example, if $G \le \mathrm{GL}(n, \mathbf{Z})$, then let $p = 3$.) In Section 1.3 we saw that G can be embedded in $\mathrm{GL}(n, \mathbf{Z}_p)$, and we defined a homomorphism $\theta : \mathrm{GL}(n, \mathbf{Z}_p) \to \mathrm{GL}(n, F_p)$. Let K be the intersection of G with the kernel of θ. By Lemma 1.1, G is solvable-by-finite if and only if K is triangularizable. This allows us to reduce problems concerning solvable-by-finite groups to corresponding problems concerning triangularizable groups. We will now elaborate on how this reduction can be achieved.

First we must obtain a description of K. One way to do so (as in Dixon (1985)) is to get a set T of coset representatives for G/K and to then use the Schreier construction to get a set of $1+|T|(k-1)$ generators for K. (See Proposition 3.4 in Chapter 4 of Sims (1994).) If we have a reasonable number of generators for K, then we can decide whether or not G is solvable-by-finite since we can decide whether or not K is triangularizable. (See Section 2.7.)

The Schreier construction will be impractical in many cases since even if p is small, $|T|$ might be large: for example, $|\mathrm{GL}(10, F_3)| \sim 3 \times 10^{47}$. When $|T|$ is large, we can use techniques for working with matrix groups over finite fields to obtain a presentation for G/K and thereby obtain normal generators for K. One way to achieve this is to view G/K as a permutation group on the vectors in F_p^n and to use algorithms for finding presentations of permutation groups. If G is solvable, then we can instead use special methods for finding presentations for solvable subgroups of $\mathrm{GL}(n, F_p)$. (See Luks (1992).)

Suppose now that K is given by normal generators, i.e., suppose we have generators for $\widetilde{K}$ such that $\widetilde{K}^G = K$. We can decide whether or not $\widetilde{K}$ is triangularizable, and if it is, then we can test membership in $\widetilde{K}$. By conjugating elements in $\widetilde{K}$ by $g_1, \ldots, g_k, g_1^{-1}, \ldots, g_k^{-1}$, keep building up $\widetilde{K}$, stopping when either it is not triangularizable (in which case G is not solvable-by-finite) or $\widetilde{K}$ is both triangularizable and normal in G (in which case $\widetilde{K} = K$ and G is solvable-by-finite). If we let K_i be the approximation for K obtained after i iterations, then

$$K_1 \leq K_2 \leq \cdots,$$

and $\bigcup K_i = K$. Since K is finitely generated, there exists an i such that $K_i = K$, so the algorithm is guaranteed to terminate.

2.9. Testing membership in a polycyclic-by-finite group

We now discuss the problem of deciding whether or not a given matrix a in $\mathrm{GL}(n, \mathbf{Q})$ is an element of G, where G is a polycyclic-by-finite matrix group given by a finite set $\{g_1, \ldots, g_k\}$ of generators. We describe two algorithms for this problem. (We have not yet compared the efficiency of these algorithms.)

We carry over our notation from Section 2.8: S is the subring of $\mathbf{Q}$ generated by the entries of $g_1, \ldots, g_k, g_1^{-1}, \ldots, g_k^{-1}$; p is an odd prime which does not divide the denominator of any of the entries of $g_1, \ldots, g_k, g_1^{-1}, \ldots, g_k^{-1}$; θ is the map from $\mathrm{GL}(n, \mathbf{Z}_p)$ to $\mathrm{GL}(n, F_p)$ defined in Section 1.3; K is the intersection of G with the kernel of θ; T is a set of coset representatives for G/K. We may assume that the entries of a are in S, and as we saw in the Section 2.8, we may assume that we have generators for K.

The first algorithm for deciding whether or not $a \in G$ relies on the algorithm in Section 2.6 for testing membership in a triangularizable matrix group. If $|G/K|$ is small, then find a set T of coset representatives for G/K and simply try each $t \in T$, testing whether or not $at^{-1} \in K$. If $|G/K|$ is large, then use techniques for matrices over finite fields to decide whether or not $\theta(a) \in \theta(G)$ and, if so, to find g in G such that $\theta(a) = \theta(g)$. Then $a \in G$ if and only if $ag^{-1} \in K$. In the case when G is polycyclic, this approach would allow us to use algorithms developed for working with solvable matrix groups over finite fields. (See Luks (1992).)

The second algorithm for deciding whether or not $a \in G$ relies on the algorithms in Beals (1996) for working with abelian-by-finite matrix groups as well as the algorithm in Section 2.3 for testing membership in a unitriangular matrix group. Recall that $K \triangleleft G$, $|G : K| < \infty$, and K is triangularizable. Find a basis for R^n relative to which K has the block structure described in Proposition 2.7. With respect to this basis for R^n, G has the block structure described in Proposition 2.9. (This follows from the proof of Proposition 2.9.) Assume that a has the same block structure with respect to this basis. Let ψ be the map from $\langle G, a\rangle$ to $\mathrm{GL}(n, R)$ given below.

$$\begin{pmatrix} a_1 & * & \cdot & \cdot & * & * \\ 0 & a_2 & * & \cdot & \cdot & * \\ \cdot & 0 & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & \cdot & \cdot & \cdot & \cdot & * \\ 0 & 0 & \cdot & \cdot & 0 & a_r \end{pmatrix} \longmapsto \begin{pmatrix} a_1 & 0 & \cdot & \cdot & 0 & 0 \\ 0 & a_2 & 0 & \cdot & \cdot & 0 \\ \cdot & 0 & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ 0 & \cdot & \cdot & \cdot & \cdot & 0 \\ 0 & 0 & \cdot & \cdot & 0 & a_r \end{pmatrix}$$

We can obtain generators for the intersection H of G with the kernel of ψ much as we found the unipotent part of a triangularizable matrix group in Section 2.6. Use the algorithm in Beals (1996) to find a presentation for $\psi(G)$, and from that obtain a set $\{h_1, \ldots, h_l\}$ of normal generators for H. Let $\widetilde{H} = \langle h_1, \ldots, h_l\rangle$. Then H is the normal closure of $\widetilde{H}$ in G. By conjugating elements in $\widetilde{H}$ by $g_1, \ldots, g_k$, keep building up $\widetilde{H}$ until $\widetilde{H}$ is normal in G, in which case $\widetilde{H} = H$. (Since H satisfies the ascending chain condition on subgroups, we do not need to conjugate by $g_1^{-1}, \ldots, g_k^{-1}$.) This process is guaranteed to terminate since G is assumed to be polycyclic-by-finite. Use the algorithm in Beals (1996) to decide whether or not $\psi(a) \in \psi(G)$, and, if so, to find a $g \in G$ such that $\phi(a) = \psi(g)$. Then $a \in G$ if and only if $ag^{-1} \in H$.

3. Related Work

In Baumslag *et al.* (1991) many fundamental and useful problems concerning polycyclic matrix groups, including all of the problems discussed in this paper, are proved to be decidable. Many of the algorithms in Baumslag *et al.* (1991) are not suitable for computer implementation as that was not the goal of that paper. On the other hand, as will be shown in the Section 4, the algorithms in this paper are suitable for computer implementation. We will not describe the algorithms in Baumslag *et al.* (1991) completely here, but we will highlight several particularly impractical steps in order to contrast them with the algorithms in Section 2.

Suppose that G is a polycyclic-by-finite matrix group. In Baumslag *et al.* (1991) an algorithm is given to decide whether or not a given matrix a is an element of G. If m is a positive integer, let θ_m be the map from $\mathrm{GL}(n, \mathbf{Z})$ to $\mathrm{GL}(n, \mathbf{Z}/m\mathbf{Z})$. If a is not an element of G, then there exists an m such that $\theta_m(a)$ is not an element of $\theta_m(G)$ (Theorem 5 in Chapter 5 of Segal (1983)). In Baumslag *et al.* (1991), the authors verify that a is not an element of G by simply trying all positive integers m, testing whether or not $\theta_m(a)$ is an element of $\theta_m(G)$, and stopping if it is not. They verify that a is an element of G by enumerating the elements of G, stopping if a is found. These two procedures are then run in parallel. Eventually one of them stops, and then it is known whether or not $a \in G$. The algorithm in Baumslag *et al.* (1991) to decide whether or not a given matrix group is polycyclic-by-finite relies on this algorithm for testing membership in such a group.

In Baumslag *et al.* (1991) an algorithm is given to verify that G is triangularizable.

This algorithm enumerates the algebraic number fields F and enumerates for each F the $n \times n$ nonsingular matrices p over F, testing whether or not $p^{-1}Gp$ is upper-triangular, stopping if this is the case. If G is polycyclic, a triangularizable subgroup H of finite index in G is found by enumerating all the normal subgroups H of G, testing each H to see if it is triangularizable and stopping when such an H is found. The unipotent part of G is then found by triangulating the matrices in H over F. Note that one might have to consider number fields F whose degree over $\mathbf{Q}$ is as much as $n!$. It is clear that these algorithms in Baumslag *et al.* (1991) are not practical, even for simple examples.

Algorithms for testing membership in an abelian subgroup of $\mathrm{GL}(n, R)$, as well as algorithms to find a presentation for such a group, were discovered independently by the author and Babai *et al.* (1996). The algorithm in Babai *et al.* (1996) is proved to have a run-time which is polynomial in the length of the input. While that algorithm is in some respects similar to the one presented here, the method for testing membership in a matrix group which is both unipotent and abelian is quite different since the algorithm described in this paper works for all finitely generated unipotent groups.

In Beals (1996) an algorithm is given for deciding whether or not a given finitely generated subgroup of $\mathrm{GL}(n, R)$ is solvable-by-finite. The author proves that the run-time of his algorithm is polynomial in the length of input; the practicality of the algorithm has not been tested. We described in Section 2.8 a practical alternative to the algorithm in Beals (1996).

As we saw in Sections 2.8 and 2.9, our algorithms for solvable-by-finite matrix groups rely on the observation in Dixon (1985) that solvable-by-finite subgroups of the p-congruence subgroup of $\mathrm{GL}(n, \mathbf{Z}_p)$ are triangularizable for p an odd prime. We also saw that it is an easy consequence of this result that problems concerning solvable-by-finite matrix groups can be reduced to problems about triangularizable matrix groups. Dixon illustrates this himself by using such a reduction to solve the stabilizer and orbit problems for a nilpotent-by-finite matrix group G: given u in R^n, find the stabilizer of u in G, and given u, v in R^n, decide whether or not u and v are in the same orbit under the action of G.

In Beals (1996) the work in Babai *et al.* (1996) concerning abelian matrix groups is extended to the case of abelian-by-finite matrix groups. In this paper, we described two algorithms for testing membership in a polycyclic-by-finite subgroup of $\mathrm{GL}(n, R)$, one of which relies on these abelian-by-finite results and the other of which does not. Experimentation is needed to compare the efficiency of all of these algorithms in practice.

4. Experiments

In this section I summarize the results of 14 experiments in which I traced through the algorithms for working with triangularizable subgroups of $\mathrm{GL}(n, \mathbf{Z})$. Some of the algorithms have been implemented (in Maple). The computational number theory algorithms described in Section 2.1 have not been fully implemented. Therefore, I stepped through parts of those algorithms by hand. These experiments were run on a SPARCstation-20 with 256 megabytes of RAM and 4 processors, each running at 100 MHz. Examples ranged from 4×4 matrices with 1 digit entries to 9×9 matrices with 4 digit entries. The experiments all ran quickly (with no noticeable wait) except in one part of the algorithm—the part which calculates a polycyclic generating sequence for a unipotent group (as described in Sections 2.2 and 2.3). Because of this bottleneck, the 9×9 matrix group with 4 digit entries still had not terminated after 21 cpu hours. Were it not for this bottleneck, it seems very likely that I could have handled much larger examples.

Only triangularizable groups were considered in these experiments. In Section 2.5, we saw that all triangularizable matrix groups can be conjugated into a certain block form, where each of the groups arising along the block diagonal is abelian with a field as its enveloping algebra. The algorithms for finding the desired block form, finding the unipotent part, finding a polycyclic generating sequence for the group, and testing membership all appear to be practical for triangularizable groups. The algorithm to decide whether a given subgroup of $\mathrm{GL}(n, \mathbf{Z})$ is triangularizable is an easy extension of the algorithm to find the desired block form for a triangularizable subgroup; therefore, it is reasonable to expect that it, too, is practical. For abelian groups, the algorithms for testing membership and finding a presentation appear to be practical.

In order to perform experiments on polycyclic-by-finite groups which are not triangularizable, one would need implementations of algorithms for working with matrix groups over finite fields. One could then implement the algorithms in Sections 2.8 and 2.9.

The results presented here are preliminary. A full implementation of these algorithms, including the algorithms for triangularizable groups, would be a significant and worthwhile project. Ideally, an implementation platform would include some primitives for working with finitely generated abelian groups, number fields and matrix groups over finite fields.

4.1. Creating examples

Proposition 2.7, the structure theorem for triangularizable groups, shows that canonical examples can be created from abelian matrix groups which are embedded in fields. To create examples of such abelian groups, I used the tables in Cohen (1995) describing over 200 number fields of degree two or three. For a given field F, the table gives an irreducible polynomial f over $\mathbf{Z}$ such that $F = \mathbf{Q}[x]/(f)$, and a system of fundamental units as linear combinations of powers of α, where α is a root of f. Let m be the companion matrix for f. By taking suitable linear combinations of powers of m, I computed the fundamental units as matrices. I then formed three words in these units (products of the units and their inverses), and thus I got three generators for an abelian matrix group with a field as its enveloping algebra. (Notice that an abelian group created in this way is torsion-free, and its matrices have dimension 2 or 3.) I restricted my attention to examples with integral entries with between 1 and 4 digits. I did this for seven different fields, and thereby created seven different abelian groups, each given by three generators. To build a triangularizable group, I strung two or three of the abelian groups along a diagonal, thereby obtaining three generators for an abelian group in block diagonal form. For each of these three generators I then chose random integers (in a certain range) for each of the entries above the block diagonal. In this way, I created seven different triangularizable subgroups of $\mathrm{GL}(n, \mathbf{Z})$, where n equaled 4, 6, or 9; each of these groups was given by three generators.

For each of the seven triangularizable groups, I constructed four elements to test for membership. The first was a word of length six in the generators, and the second was a word of length 11 in the generators. The third was identical to the first apart from a random change in one of the off-diagonal elements. The fourth was identical to the first in the off-diagonal elements, but the matrices along the diagonal were each chosen independently, thus reducing the chances of ending up with an element of the triangularizable group.

In order to test the algorithm to find the desired block form, I created seven triangularizable groups which were not in this form by performing a change of basis on the original seven block upper triangular groups created above. The change of basis matrix was obtained by doing n elementary row operations (over $\mathbf{Z}$) to the identity matrix, where n (4, 6, or 9) was the dimension of the matrix group in question.

4.2. Methods

In this section, I provide some details about how the algorithms were implemented (and how I worked around the fact that some of them were not fully implemented).

Let G be a triangularizable group which is not in the desired block form. The algorithm for finding this form as described in Section 2.5 calls for calculating a basis for $\mathrm{Env}(G)$ over $\mathbf{Q}$ and then considering the Lie bracket of each pair of basis elements. In my experiments, I simply considered the Lie bracket of each pair of group generators. Using simple linear algebra, I found the intersection V of the nullspaces of these Lie brackets, and in all cases, I found that V was in fact a nontrivial invariant subgroup of $\mathbf{Z}^n$. Using linear algebra again, I found a complement W to V, and proceeded inductively to find a suitable change of basis for W. In this way, I obtained a basis with respect to which G was in block upper triangular form, with abelian groups along the diagonal. I then checked to see whether or not these abelian groups were embedded in fields as follows. Let H be one of the abelian groups along the block diagonal of G. I picked one of the elements h of H appearing in the block diagonal of one of the generators for G. I decided whether or not h was a primitive element for $\mathrm{Env}(H)$ by checking whether or not the other group generators for H were linear combinations of powers of h; in every case h was a primitive element. I then computed the minimal polynomial f for h; in every case f was irreducible, from which I could deduce that $\mathrm{Env}(H)$ was a field. This algorithm was partially implemented as a Maple program; running the full algorithm required some user interaction.

Now let G be a triangularizable group which is in the desired block form, and let $a_1, \ldots, a_4$ be the elements which I was testing for membership. Let $H = \langle G, a_1, \ldots, a_4 \rangle$, and let ψ be the map from H to the block diagonal abelian group as described in Section 2.6.

The first step was to test membership in $\psi(G)$ and to find a presentation for $\psi(G)$, both of which were accomplished by finding a presentation for $\psi(H)$. In my examples $\psi(H)$ was the direct product of several torsion-free abelian groups, and the enveloping algebra of $\psi(H)$ was the direct sum $F_1 \oplus \cdots \oplus F_r$ of several fields. Therefore, I was able to simplify the algorithms of Section 2.4 as follows. Using the logarithmic homomorphism for each of the fields $F_1, \ldots, F_r$, I constructed a map from ψ from H to $\mathbf{R}^s$ for some s; in my examples ψ was actually an embedding (since my abelian groups were torsion-free). Let L be the image of this map, a lattice in a hyperplane of $\mathbf{R}^s$. The algorithm in Buchmann and Pohst (1989) was used to find a presentation for $\psi(H)$ and hence for H; the algorithm requires as input a lower bound for the length of a nonzero vector in L. In these experiments, the fields in question were of low degree; therefore, I computed the lower bound using the fundamental units for each of the fields $F_1, \ldots, F_r$. I then followed the Buchmann and Pohst (1989) algorithm closely.

In each of the seven experiments, the Buchmann and Pohst (1989) algorithm showed that a_4 was not an element of the group and it produced three unitriangular elements b_1, b_2, b_3 such that $a_i \in G$ if and only if $b_i \in U$, where U was the unipotent part of G. From the presentation for $\psi(G)$ and normal generators for G', I obtained normal generators for U.

I implemented in Maple a function NORMAL_CLOSURE to compute the normal closure of a subgroup of $\mathrm{Tr}_1(n, \mathbf{Z})$ (given by a finite set of generators) in a subgroup G of $\mathrm{GL}(n, \mathbf{Z})$ (also given by a finite set of generators). Subroutines included versions of POLY_SUBGROUP and POLY_MEMBER (from Section 9.6 of Sims (1994)). My algorithms were slightly different from those in Sims (1994): there was no need to rewrite matrices as words in the standard generators for $\mathrm{Tr}_1(n, \mathbf{Z})$.

For $n = 6$, the routine to compute the power of a matrix was optimized by Eddie H. Lo (Lo, 1996). Lo's idea is based on an observation by Hall that collection in a finitely presented nilpotent group can be performed using polynomials (Sims, 1994). Choose any ordering on the 15 off-diagonal entries in a given matrix in $\mathrm{Tr}_1(6, \mathbf{Z})$. Compute 15 polynomials in $\mathbf{Q}[x, a_1, \ldots, a_{15}]$ with the following property: if a is a matrix in $\mathrm{Tr}_1(6, \mathbf{Z})$ with off-diagonal elements $a_1, \ldots, a_{15}$, then the entries of a^x are given by the polynomials. When raising matrices to very large exponents, this optimization resulted in a substantial improvement in run-time.

4.3. Results

I give each of the seven triangularizable groups in the desired block form a name of the form $G.b.s.d$, where b is the number of blocks along the diagonal, s is the size of each block, and d is the maximum number of digits in each entry. For example, $G.3.2.4$ has 3 2×2 blocks along the diagonal, it has random entries above the block diagonal, and the maximum number of digits in an entry is 4. The seven groups obtained from these via a change of basis are referred to as $XG.b.s.d$. Note that in each case, the entries in the resulting matrix are larger than d. For example, $XG.3.2.4$ is derived from $G.3.2.4$, and the maximum number of digits in an entry of $XG.3.2.4$ is 7.

The algorithm for finding the desired block form completed successfully (with no noticeable wait) for all seven of the groups $XG.2.2.1$, $XG.2.2.4$, $XG.2.3.1$, $XG.2.3.4$, $XG.3.2.1$, $XG.3.2.4$, and $XG.3.3.4$. There was never any growth in the number of digits in the entries in the desired block form, and in some cases the number of digits decreased.

The algorithm for testing membership, finding the unipotent part, and finding a polycyclic generating sequence for a triangularizable group completed successfully for the following groups: $G.2.2.1$, $G.2.2.4$, $G.2.3.1$, $G.2.3.4$, $G.3.2.1$, and $G.3.2.4$. It was aborted after 21 cpu hours for $G.3.3.4$.

In the six successful cases, the algorithm found that the first three test matrices were in the group and that the fourth was not. The reason that the third test matrix was in the group is that in each of the experiments, the unipotent part of the group was large in the following sense. Let G be a given triangularizable group in the desired block form. Let H be the group consisting of all matrices with the same block form and with identity matrices along the diagonal. The unipotent part of G was of small finite index in H.

It appears that POLY_SUBGROUP is the source of the serious bottleneck that occurs in NORMAL_CLOSURE: in the six completed experiments, the subroutine POLY_SUBGROUP was only called from NORMAL_CLOSURE between 2 and 20 times, but each call to POLY_SUBGROUP took an average of between .3 and 30 cpu minutes, depending on the dimension of the matrix group and the size of the original entries. In the 9×9 example with 4 digit entries, NORMAL_CLOSURE still had not terminated after 21 cpu hours, and the average time for a call to POLY_SUBGROUP up to that time was 3.5 cpu hours. One reason that POLY_SUBGROUP appears to be so slow is that in the process of calculating a polycyclic generating sequence it often raises matrices to huge powers: a 16 digit exponent arose in the smallest example, and a 4200 digit exponent arose in the largest example. Despite the alarmingly large exponents, in the experiments that run successfully, the generators for the unipotent part always had single digit entries.

5. Acknowledgments

I would like to thank my Ph. D. thesis advisor, Charles Sims, for many helpful conversations over the years; part of this paper is based on my thesis. I would like to thank Eddie H. Lo for implementing the optimization discussed in Section 4, Jerry Tunnell, for suggesting Lemma 2.2, Emmi Schatz, for advice concerning the experiments, and DIMACS, for providing office space and computer facilities on several visits to New Jersey. Finally, I would like to thank the reviewers for their thoughtful comments.

References

Babai, L., Beals, R., yi Cai, J., Ivanyos, G., Luks, E. M. (1996). Multiplicative equations over commuting matrices. In *Proc. 7th ACM-SIAM SODA (Symp. on Discrete Algorithms)*, pages 498–507, New York. Association for Computing Machinery.

Baumslag, G., Cannonito, F. B., Robinson, D. J. S., Segal, D. (1991). The algorithmic theory of polycyclic-by-finite groups. *J. Algebra*, 142:118–149.

Beals, R. (1996). Towards polynomial time algorithms for matrix groups. In Finkelstein, L., Kantor, W. M., editors, *Groups and Computation II*, volume 28 of *DIMACS Series in Discrete Mathematics and Computer Science*, pages 31–54, Providence, RI. American Mathematical Society.

Buchmann, J., Pohst, M. (1989). Computing a lattice basis from a system of generating vectors. In *EUROCAL '87, European Conference on Computer Algebra*, volume 378 of *Lecture notes in computer science*, pages 54–63. Springer-Verlag.

Cohen, H. (1995). *A Course in Computational Algebraic Number Theory*, volume 138 of *Graduate texts in mathematics*. Springer-Verlag, New York.

Dixon, J. D. (1971). *The Structure of Linear Groups*. Van Nostrand Reinhold, New York.

Dixon, J. D. (1985). The orbit-stabilizer problem for linear groups. *Can. J. Math.*, 37:238–259.

Dobrowolski, E. (1979). On a question of Lehmer and the number of irreducible factors of a polynomial. *Acta Arith.*, 34:391–401.

Eisenbud, D., Huneke, C., Vasconcelos, W. (1992). Direct methods for primary decomposition. *Invent. Math.*, 110:207–235.

Friedl, K., Rónyai, L. (1985). Polynomial time solutions of some problems in abstract algebra. In *Proc. 17th ACM STOC*, pages 153–162, New York. Association for Computing Machinery.

Ge, G. (1993). *Algorithms Related to Multiplicative Representations of Algebraic Numbers*. PhD thesis, U. C. Berkeley, Math Dept.

Lo, E. H. (1996). personal communication.

Luks, E. M. (1992). Computing in solvable matrix groups. In *Proc. 33rd IEEE FOCS*, pages 111–120, Los Alamitos, CA. IEEE Computer Society Press.

Ostheimer, G. (1996a). *Algorithms for Polycyclic-by-Finite Groups*. PhD thesis, Rutgers University, Math Dept.

Ostheimer, G. (1996b). Algorithms for polycyclic-by-finite matrix groups. In Finkelstein, L., Kantor, W. M., editors, *Groups and Computation II*, volume 28 of *DIMACS Series in Discrete Mathematics and Computer Science*, pages 297–307, Providence, RI. American Mathematical Society.

Segal, D. (1983). *Polycyclic Groups*, volume 82 of *Cambridge tracts in mathematics*. Cambridge University Press, New York.

Sims, C. C. (1994). *Computation with Finitely Presented Groups*, volume 48 of *Encyclopedia of mathematics and its applications*. Cambridge University Press, New York.

Wehrfritz, B. A. F. (1973). *Infinite Linear Groups*. Springer-Verlag, Berlin.