COMPUTING WITH MATRIX GROUPS OVER INFINITE DOMAINS: DECIDING FINITENESS

A. S. DETINKO AND D. L. FLANNERY

1. INTRODUCTION

Computing with matrix groups over infinite fields is a relatively new and undeveloped area of computational group theory; it poses challenges that are very different to those encountered when computing with groups over finite fields. For one thing, certain basic computational problems (e.g. membership testing and the conjugacy problem) are undecidable for some classes of matrix groups over infinite fields (see [6, Section 5.2]). Serious complexity issues can also arise, such as growth of matrix entries.

Deciding finiteness is a basic computational problem in any class of potentially infinite groups. Of course, in general finiteness may not be decidable at all (see [9, 4.2, p. 51]). However, for matrix groups, the outlook is more optimistic. Note that a matrix group given by a finite set $\mathcal{S}$ of generators over a field $\mathbb{F}$ is defined over the integral domain $R \subseteq \mathbb{F}$ generated by the entries of the elements of $\mathcal{S} \cup \mathcal{S}^{-1}$. Since R is finitely generated, then, it suffices to design algorithms just for the fields $\mathbb{F} = \mathbb{P}(X_1, \ldots, X_m)$, where the X_i are independent indeterminates, $m \geq 0$, and $\mathbb{P}$ is either a number field or a finite field (cf. [13, Chapters 4 & 10]).

Several authors [1, 2] have developed algorithms (both deterministic and randomized) for deciding finiteness of matrix groups over $\mathbb{Q}$. Those algorithms, and a standard reduction obtained by representing algebraic numbers as matrices over $\mathbb{Q}$, enable finiteness testing over any number field. However, since it entails an increase in the degree of matrices, the effectiveness of this approach is limited.

Deciding finiteness for groups over functional fields is considered in [3, 8, 11]. A common theme in those papers is a reliance on computing in matrix algebras. Polynomial-time algorithms in both zero and positive characteristic were proposed, but once more their practicality is limited. Also, no implementations are publicly available.

An essentially different technique for deciding finiteness, based on changing the ground domain via congruence homomorphism, is described in [5]. This technique is universal and general: it can be applied in the same way over any domain. The technique was used in [5] for deciding finiteness of nilpotent matrix groups. Algorithms from [5] have been implemented as part of the GAP package 'Nilmat' [4]. Experimental evidence demonstrates the efficiency of the Nilmat algorithms, which successfully test finiteness of nilpotent matrix groups over $\mathbb{Q}$ in very large degrees, where other available algorithms fail.

This paper applies the technique of [5] to develop practical algorithms for deciding finiteness of matrix groups over functional fields. In fact, we tackle a broader problem: testing finiteness of a given group G; and computing $|G|$ if G is finite (computing orders is a fundamental computational problem; see e.g. [10, Section 2]). Special attention is paid to the case of zero characteristic, but we outline some ideas in positive characteristic too. Our main algorithm has been implemented in GAP, for functional fields over $\mathbb{Q}$.

2000 Mathematics Subject Classification: Primary 20H20, Secondary 68W30.

Supported in part by Science Foundation Ireland, grant RFP05/MAT0008.

We mention that [11, Section 2] contains an approach to the problem of deciding finiteness over functional fields of zero characteristic. This approach is impractical because it invokes the algorithm of [2] for deciding finiteness over $\mathbb{Q}$ in squared degree. The present paper, together with [2], gives a practical algorithm for deciding finiteness and computing orders of matrix groups over any field of zero characteristic.

2. Deciding finiteness via congruence homomorphism

In this section we lay out some background for the main method that we use in deciding finiteness. Let Δ be an integral domain and ϱ an ideal of Δ. Denote the natural ring epimorphism $\Delta \to \Delta/\varrho$ by φ_ϱ. Recall that Δ/ϱ is an integral domain (respectively field) if and only if ϱ is prime (respectively maximal). Also, if Δ is finitely generated and ϱ is maximal, then Δ/ϱ is a finite field. We get a ring homomorphism $\mathrm{Mat}(n,\Delta) \to \mathrm{Mat}(n,\Delta/\varrho)$ by entrywise extension, and then a group homomorphism $\mathrm{GL}(n,\Delta) \to \mathrm{GL}(n,\Delta/\varrho)$ by restriction. With a slight abuse of notation, we denote all of these homomorphisms φ_ϱ. The map φ_ϱ on $\mathrm{GL}(n,\Delta)$ is a *congruence homomorphism* (with respect to ϱ). The kernel $\mathcal{G}_\varrho$ of φ_ϱ on $\mathrm{GL}(n,\Delta)$ is called a (principal) congruence subgroup. If $G \leq \mathrm{GL}(n,\Delta)$ then we denote the congruence subgroup $G \cap \mathcal{G}_\varrho$ as G_ϱ.

Let $\mathcal{S} = \{S_1,\ldots,S_r\} \subseteq \mathrm{GL}(n,\mathbb{F})$, where $\mathbb{F}$ is the field of fractions of Δ. Let $G = \langle \mathcal{S} \rangle$. Then $G \leq \mathrm{GL}(n,R)$ where the integral domain R is generated by the entries of the matrices in $\mathcal{S} \cup \mathcal{S}^{-1}$. Let μ be the least common multiple of the denominators of the generators of R. Then $R \subseteq \mu^{-1}\Delta$, the ring of fractions with denominators in the submonoid of $\Delta^\times$ generated by μ (localization of Δ at μ). We are concerned in this paper with the case $\mathrm{char}\,\mathbb{F} = 0$ and $\Delta = \mathbb{P}[X_1,\ldots,X_m]$, $m > 0$, where $\mathbb{P}$ is (isomorphic to) an algebraic number field. Note that $\mu^{-1}\Delta$ is a unique factorization domain (UFD).

If G is finite then by a result due to Mal'cev [13, 4.2, p. 51], there exists a maximal ideal ϱ of R such that φ_ϱ is an isomorphism $G \to \varphi_\varrho(G)$. To apply this result in practice, we need a way of selecting a suitable ideal ϱ i.e. such that G_ϱ is trivial. The following lemma provides criteria for a suitable ideal. (Generalizations of this lemma to positive characteristic, and to Dedekind domains, may be found in [5, Section 3].)

Lemma 2.1. ([12, Theorem 3, pp. 68-69].) *Let Δ be a UFD of characteristic zero. Suppose that λ is an irreducible element of Δ such that λ does not divide 2, and λ^2 does not divide p for any rational integer p. If $\varrho = \lambda\Delta$ then $\mathcal{G}_\varrho$ is torsion-free.*

Given elements $\lambda_1,\ldots,\lambda_t$ of a ring Δ, we denote the ideal of Δ generated by those elements as $\langle \lambda_1,\ldots,\lambda_t \rangle$.

Corollary 2.2. *Let Δ be a UFD of characteristic zero. Suppose that there exist elements $\lambda_1,\ldots,\lambda_\ell$ of Δ such that, for all i, $1 \leq i \leq \ell$,*

(i) *$\Delta/\langle \lambda_1,\ldots,\lambda_i \rangle$ is a UFD of characteristic zero*;
(ii) *λ_i is an irreducible element of $\Delta/\langle \lambda_1,\ldots,\lambda_{i-1} \rangle$ such that λ_i does not divide 2, and λ_i^2 does not divide p for any rational integer p.*

Set $\varrho = \langle \lambda_1,\ldots,\lambda_\ell \rangle$. Then $\mathcal{G}_\varrho$ is torsion-free.

Proof. This follows by induction from Lemma 2.1, using the fact that the composite of congruence homomorphisms, each of which has torsion-free kernel, also has torsion-free kernel. □

Now we consider how to construct an ideal ϱ as per Corollary 2.2, in our particular situation $\Delta = \mu^{-1}\mathbb{P}[X_1,\ldots,X_m]$, where $m \geq 1$, $\mathbb{P}$ is a number field, and $\mu = \mu(X_1,\ldots,X_m)$. Let

$\alpha = (\alpha_1, \ldots, \alpha_m)$, $\alpha_i \in \mathbb{P}$. We say that α is *admissible* if $\mu(\alpha) \neq 0$. Since $\mathbb{P}$ is infinite, there exist infinitely many admissible α. Let $\alpha = (\alpha_1, \ldots, \alpha_m)$ be admissible, and define $\lambda_i = X_i - \alpha_i$, and $\varrho = \varrho(\alpha) := \langle \lambda_1, \ldots, \lambda_m \rangle$. Then $\varrho \subseteq \Delta$ and $\varphi_\varrho(\Delta) \subseteq \mathbb{P}$. By Corollary 2.2, $\mathcal{G}_\varrho$ is torsion-free. To construct $\varphi_\varrho(G)$ we just substitute α_i for X_i in entries of the S_j, $1 \leq i \leq m$, $1 \leq j \leq r$. Note that $\varphi_\varrho(G) \leq \mathrm{GL}(n, \mathbb{P})$.

If G is infinite then $\varphi_{\varrho(\alpha)}(G)$ may be finite; but as the next lemma shows, the probability of that occurrence (assuming random choice of an admissible α) is small—at least over fields with a single indeterminate.

Lemma 2.3. *Let $\mathbb{F} = \mathbb{P}(X)$. If $G = \langle \mathcal{S} \rangle$ is infinite, then there are only finitely many α such that $\varphi_{\varrho(\alpha)}(G)$ is finite.*

Proof. Since $\mathbb{P}$ is a number field, there is an upper bound $\nu = \nu(n)$ on the order of finite subgroups of $\mathrm{GL}(n, \mathbb{P})$. Since G is infinite, by Burnside's theorem [12, Theorem 1, p. 178], G has an element g of infinite order. Let $t \in \mathbb{Z}$ be greater than ν. Since $g^t \neq 1$, either there exist i, j, $i \neq j$, such that the (i,j)th entry $g_{ij}^{(t)} := g_{ij}^{(t)}(X)$ of g^t is non-zero, or there exists k such that $g_{kk}^{(t)} \neq 1$. Now there are only finitely many α such that $g_{ij}^{(t)}(\alpha) = 0$ (and only finitely many α such that $g_{kk}^{(t)}(\alpha) = 1$). Hence, for all but finitely many α, $\varphi_{\varrho(\alpha)}(g^t) \neq 1_n$. Thus, for all such α we have that $|\varphi_{\varrho(\alpha)}(G)| > \nu$ i.e. $\varphi_{\varrho(\alpha)}(G)$ is infinite. □

Remark 2.4. Arguments similar to those in the proof of Lemma 2.3 show that for more than one indeterminate, there are still infinitely many admissible α such that $\varphi_{\varrho(\alpha)}(G)$ is infinite if G is infinite.

The results of this section suggest the following strategy for deciding finiteness. First, we select an admissible α and construct $\varphi_\varrho(G) \leq \mathrm{GL}(n, \mathbb{P})$ for $\varrho = \varrho(\alpha)$. Then we test whether $\varphi_\varrho(G)$ is finite; and if so, then we test whether G_ϱ is trivial. The latter problem is dealt with in the next section.

3. An algorithm for deciding finiteness over functional fields in zero characteristic

In this section we provide a method for testing whether a congruence homomorphism φ_ϱ on a finite subgroup G of $\mathrm{GL}(n, \mathbb{F})$ is an isomorphism, where $\mathbb{F} = \mathbb{P}(X_1, \ldots, X_m)$, $\mathbb{P}$ a number field, $m > 0$. An advantage of our method is that it avoids actual construction of the congruence subgroup G_ϱ, which can be a hard computational problem.

We retain the notation of Section 2. For $\mathcal{S} \subseteq \mathrm{GL}(n, \mathbb{F})$ and $G = \langle \mathcal{S} \rangle$, we denote the $\mathbb{P}$-enveloping algebra of G by $\langle G \rangle_\mathbb{P}$, and denote the span of $\mathcal{S}$ by $\mathrm{Span}_\mathbb{P}(\mathcal{S})$. If $\langle G \rangle_\mathbb{P}$ is finite-dimensional, then a basis of $\langle G \rangle_\mathbb{P}$ can be found by the following well-known procedure. Briefly, one constructs linearly independent subsets of $\langle G \rangle_\mathbb{P}$ recursively, at each stage testing whether products of each element of a subset with every element of $\mathcal{S} \cup \mathcal{S}^{-1}$ are contained in the span of the subset. (Note that if G is finite then we can replace $\mathcal{S} \cup \mathcal{S}^{-1}$ by $\mathcal{S}$.) We refer to that procedure as `BasisEnvAlgebra`$(\mathcal{S})$. Its output is a basis $A_1, \ldots, A_d$ of $\langle G \rangle_\mathbb{P}$ such that the A_i are finite length products of elements of $\mathcal{S} \cup \mathcal{S}^{-1}$.

Under the assumptions of Section 2, i.e. $\Delta = \mu^{-1}\mathbb{P}[X_1, \ldots, X_m]$ and $\varrho = \varrho(\alpha)$ for admissible $\alpha = (\alpha_1, \ldots, \alpha_m)$, $\alpha_i \in \mathbb{P}$, we have that φ_ϱ acts identically on the elements of $\mathbb{P}$. Hence φ_ϱ induces a homomorphism of $\mathbb{P}$-algebras $\langle G \rangle_\mathbb{P} \to \langle \varphi_\varrho(G) \rangle_\mathbb{P}$. We need the following very simple linear algebra facts.

Lemma 3.1. *Let $A_1, \dots, A_\ell$ be elements of $\mathrm{Mat}(n, \Delta)$ such that $\varphi_\varrho(A_1), \dots, \varphi_\varrho(A_\ell)$ are linearly independent over $\varphi_\varrho(\Delta)$. Then the following hold.*

(i) *$A_1, \dots, A_\ell$ are linearly independent over $\mathbb{P}$.*

(ii) *If $A \in \mathrm{Mat}(n, \Delta)$, $\varphi_\varrho(A) = \sum_{i=1}^{\ell} \beta_i \varphi_\varrho(A_i)$, $\beta_i \in \mathbb{P}$, and $A \in \mathrm{Span}_{\mathbb{P}}(A_1, \dots, A_\ell)$, then $A = \sum_{i=1}^{\ell} \beta_i A_i$.*

Lemma 3.2. *If $\langle G \rangle_{\mathbb{P}}$ is finite-dimensional then the following statements are equivalent.*

(i) *φ_ϱ is an algebra isomorphism $\langle G \rangle_{\mathbb{P}} \to \langle \varphi_\varrho(G) \rangle_{\mathbb{P}}$.*

(ii) *$\dim_{\mathbb{P}} \langle \varphi_\varrho(G) \rangle_{\mathbb{P}} = \dim_{\mathbb{P}} \langle G \rangle_{\mathbb{P}}$.*

Moreover, in the situation of (i) *or* (ii)*, $G_\varrho = 1$.*

Corollary 3.3. *If $\varphi_\varrho(G)$ is finite and $\dim_{\mathbb{P}} \langle \varphi_\varrho(G) \rangle_{\mathbb{P}} = \dim_{\mathbb{P}} \langle G \rangle_{\mathbb{P}}$ then G is finite.*

Before proceeding to the statement of our algorithm, we make some observations relating to the last step in the algorithm. Suppose that φ_ϱ is one-to-one on $\mathcal{S} = \{S_1, \dots, S_r\}$. Let $\bar{S}_i = \varphi_\varrho(S_i)$. Then given any element $\bar{A} = \bar{S}_{k_1} \cdots \bar{S}_{k_t}$ of $\langle \varphi_\varrho(G) \rangle_{\mathbb{P}}$, we are able to define a canonical pre-image $A = S_{k_1} \cdots S_{k_t}$ in $\langle G \rangle_{\mathbb{P}}$. Thus, if we have a basis $\bar{\mathcal{A}} = \{\bar{A}_1, \dots, \bar{A}_d\}$ of $\langle \varphi_\varrho(G) \rangle_{\mathbb{P}}$ computed via `BasisEnvAlgebra`, then we readily gain $\mathcal{A} = \{A_1, \dots, A_d\} \subseteq \langle G \rangle_{\mathbb{P}}$. By Lemma 3.1, $\mathcal{A}$ is linearly independent.

We now summarize all of the above in the form of an algorithm.

`IsFiniteMatGroupFuncRN`($\mathcal{S}$)

Input: $\mathcal{S} = \{S_1, \dots, S_r\}$, $S_i \in \mathrm{GL}(n, \mathbb{F})$, $\mathbb{F} = \mathbb{P}(X_1, \dots, X_m)$, $\mathbb{P}$ a number field.

Output: a message 'true' meaning that $G = \langle \mathcal{S} \rangle$ is finite, or a message 'false' otherwise.

(1) Find α admissible for $\mathcal{S} \cup \mathcal{S}^{-1}$ and compute $\bar{\mathcal{S}} := \{\bar{S}_i := \varphi_\varrho(S_i) \mid 1 \le i \le r\}$, $\varrho = \varrho(\alpha)$.

(2) If $\bar{S}_i = \bar{S}_j$ for some $i \ne j$ then return 'false'.

(3) If $\varphi_\varrho(G) = \langle \bar{\mathcal{S}} \rangle \le \mathrm{GL}(n, \mathbb{P})$ is infinite then return 'false'.

(4) Construct $\bar{\mathcal{A}} := \mathtt{BasisEnvAlgebra}(\bar{\mathcal{S}}) := \{\bar{A}_1, \dots, \bar{A}_d\}$, and find $\mathcal{A} := \{A_1, \dots, A_d\}$. For $\bar{A}_i \in \bar{\mathcal{A}}$, $\bar{S}_i \in \bar{\mathcal{S}}$, find $\alpha_k \in \mathbb{P}$ such that $\bar{A}_i \bar{S}_i = \sum_{k=1}^{d} \alpha_k \bar{A}_k$.
If for some i, j we have $A_i S_j \ne \sum_{k=1}^{d} \alpha_k A_k$ then return 'false'; else return 'true'.

3.1. **Analysis of algorithm.** A significant computational advantage of `IsFiniteMatGroupFuncRN` is that most of its operations are performed over $\mathbb{P}$, rather than over $\mathbb{P}(X_1, \dots, X_m)$.

Step (3) requires deciding finiteness over $\mathbb{P}$. For that purpose one may employ the algorithm of [2]; then the efficiency of `IsFiniteMatGroupFuncRN` would depend on the efficiency of the algorithm from [2]. Apart from this, the most time-consuming part of `IsFiniteMatGroupFuncRN` is Step (4), i.e. computing a basis of $\langle \varphi_\varrho(G) \rangle_{\mathbb{P}}$. If the input group G is finite then Step (4) is unavoidable. On the other hand, if G is infinite then Lemma 2.3 (and Remark 2.4) indicate that Step (4) likely will not be reached; that is, infiniteness of the input will be detected at an earlier stage of the algorithm. So we expect `IsFiniteMatGroupFuncRN` to complete more quickly for infinite rather than finite input.

One more computational obstacle that may occur at Step (4) is large size of entries of the $\bar{S}_i$. In that event, to compute a basis of $\langle \bar{G} \rangle_{\mathbb{P}}$ we can reduce entries of the $\bar{S}_i$ modulo a maximal ideal $\bar{\varrho}$ of $\bar{\Delta} := \Delta/\varrho$. That is, we apply another congruence homomorphism, in order to transfer the computation to the setting of a group over a finite field. A detailed discussion of this idea is beyond the scope of this paper.

3.2. **Related algorithms.**

(a) *Computing orders.* In addition to its main function, `IsFiniteMatGroupFuncRN` provides a solution of one more important computational problem: computing the order of a finite subgroup of $\mathrm{GL}(n,\mathbb{F})$. If G is finite (as recognized by `IsFiniteMatGroupFuncRN`) then $|G| = |\varphi_\varrho(G)|$. Since $\varphi_\varrho(G) \leq \mathrm{GL}(n, \Delta/\varrho)$ is a finitely generated Dedekind domain, analogously to Lemma 2.1 we may choose a maximal ideal σ of Δ/ϱ such that $\varphi_\varrho(G)$ has trivial congruence subgroup with respect to φ_σ (see [5, Section 3]). Thus we calculate the order of G by calculating the order of an isomorphic copy of G in some $\mathrm{GL}(n,q)$. In particular, if G is cyclic, then $|G|$ may be calculated by the algorithm of Celler and Leedham-Green (see [10, Section 2]).

(b) *Deciding finiteness over* $\mathbb{R}$ *and* $\mathbb{C}$. Our methods are valid for an input finitely generated matrix group G over any field of characteristic zero. That is, `IsFiniteMatGroupFuncRN` together with [2] may be used to decide finiteness and compute orders for $G \leq \mathrm{GL}(n,\mathbb{R})$ or $G \leq \mathrm{GL}(n,\mathbb{C})$. In this approach matrix entries are handled symbolically, dispensing with the need for floating point representation of numbers.

3.3. **Implementation and experimental results.** Our algorithms for deciding finiteness of matrix groups over functional fields have been implemented in GAP [7]. In this subsection we present experimental results that characterize the efficiency of `IsFiniteMatGroupFuncRN`, depending on the main input parameters.

As noted previously, when the image $\bar{G}$ of G under a congruence homomorphism is finite, the algorithm will proceed to the most computationally intensive stage. In turn, the time for completion of that stage will depend on whether the input group G is really finite or not; the former case is the most complicated. To address these issues, we performed experiments for groups which have extremal properties. Specifically, we tested groups G such that (a) both G and $\bar{G}$ are absolutely irreducible, so give the largest dimension n^2 of their enveloping algebras; and (b) $\bar{G}$ has order $2^n n!$, which is an upper bound on the order of finite subgroups of $\mathrm{GL}(n,\mathbb{Q})$ for $n \geq 10$. Some results, for $\mathbb{F} = \mathbb{Q}(X)$, are displayed in the table below. The experiments were carried out on a Pentium 4 running 1.73 GHz under Windows. CPU time is in the format minutes : seconds : milliseconds.

G	n	r	$\lvert\bar{G}\rvert$	Runtime($\bar{G}$)	Runtime(G)
G_{11}	10	3	$2^{10}10!$	0 : 00 : 02.438	0 : 00 : 05.547
G_{12}	10	3	$2^{10}10!$	”	0 : 01 : 31.781
G_{21}	20	3	$2^{20}20!$	0 : 00 : 03.063	0 : 17 : 40.703
G_{22}	20	3	$2^{20}20!$	”	0 : 19 : 06.547
G_{31}	36	12	648	0 : 00 : 03.172	0 : 02 : 02.469
G_{32}	36	12	648	”	0 : 16 : 59.078

The groups G_{i1} are infinite, and the G_{i2} are finite. For each i, the image groups $\bar{G}_{i1}, \bar{G}_{i2}$ are conjugate subgroups of $\mathrm{GL}(n,\mathbb{Q})$. For $i = 1, 2$, $\bar{G}_{i1}$ and $\bar{G}_{i2}$ are full monomial subgroups of $\mathrm{GL}(n,\mathbb{Q})$. The groups $\bar{G}_{3i}$ are finite nilpotent, and were constructed using the function `MonomialNilpotentMatGroup` of [4]. The runtime of Step (3) of `IsFiniteMatGroupFuncRN` (that is, deciding finiteness of $\bar{G}_{ij} \leq \mathrm{GL}(n,\mathbb{Q})$) is shown in column 5 of the table. This may be compared with the total runtime, given in the last column. Finiteness of $\bar{G}_{3i}$ in Step (3) was tested using the function `IsNilpotentFinite` of [4], whereas finiteness of $\bar{G}_{1i}$ and $\bar{G}_{2i}$ was tested using the GAP function `IsFinite`.

One other parameter that affects the speed of `IsFiniteMatGroupFuncRN` is the size of entries of input matrices. To monitor this, we took the entries of G_{12} to be integral polynomials of degree up to 30, with coefficients up to 2,000,000. Other groups in the table have matrix entries of much more moderate size.

4. The case of positive characteristic

In this section we discuss how the methods used in this paper may be applied in the case of positive characteristic. Most of the computation is transferred to finite fields; and, again, we can compute orders.

Let $G \leq \mathrm{GL}(n, \mathbb{F})$ be finitely generated, where $\mathbb{F} = \mathbb{F}_q(X_1, \ldots, X_m)$, $\mathbb{F}_q$ the field of size q. As usual, $G \leq \mathrm{GL}(n, \Delta)$ where $\Delta = \mu^{-1}\mathbb{F}_q[X_1, \ldots, X_m]$ for some $\mu = \mu(X_1, \ldots, X_m) \in \mathbb{F}_q[X_1, \ldots, X_m]$. There are no theoretical barriers to carrying through the above machinery for deciding finiteness of G. However, new computational difficulties may appear in positive characteristic. Firstly, the congruence subgroup need not be torsion-free, i.e. it could contain non-trivial p-elements, where $p = \mathrm{char}\,\mathbb{F}$ (see [5, Section 3]). Furthermore, we may have to extend the coefficient field to select admissible $\alpha = (\alpha_1, \ldots \alpha_m)$ for the requisite congruence homomorphism. That is, we may have to extend $\mathbb{F}_q$ to find α_i in the algebraic closure $\overline{\mathbb{F}}_q$ such that $\mu(\alpha_1, \ldots \alpha_m) \neq 0$, and $\varphi_{\varrho(\alpha)}(G) \cong G$ for a finite input G. Fortunately, it can be shown that there are infinitely many α with $\alpha_i \in \overline{\mathbb{F}}_q$ such that $\varphi_{\varrho(\alpha)}(G) \cong G$. In the case $m = 1$, $\varphi_{\varrho(\alpha)}(G) \cong G$ for all but finitely many α. Hence, if $m = 1$, G is finite and α is randomly chosen, then with a high probability we will find that $G_{\varrho(\alpha)}$ is trivial. Moreover, there exists a positive integer $\gamma = \gamma(n, r)$ such that if the α_i are all chosen in $\overline{\mathbb{F}}_q \setminus \mathbb{F}_{q^\gamma}$, then $G_{\varrho(\alpha)}$ is trivial. Taking these facts into account, we have developed an algorithm `IsFiniteMatGroupFuncFF` for deciding finiteness over $\mathbb{F}_q(X)$, analogous to `IsFiniteMatGroupFuncRN`. The two algorithms are different at Step (1), where in `IsFiniteMatGroupFuncFF` we select $\alpha_i \in \overline{\mathbb{F}}_q \setminus \mathbb{F}_{q^\gamma}$; and there is of course no need to test finiteness of $\varphi_{\varrho(\alpha)}(G)$ (cf. Step (3) of `IsFiniteMatGroupFuncRN`).

Our approach depends very much on asymptotic bounds for γ, and may necessitate working in large degree extensions of $\mathbb{F}_q$. At the same time, α_i such that $\varphi_{\varrho(\alpha)}(G) \cong G$ may exist in a smaller field, perhaps even in $\mathbb{F}_q$ itself. So we developed a modification of `IsFiniteMatGroupFuncFF` that constructs a finite chain of ideals $\varrho(\alpha^{(1)}), \ldots, \varrho(\alpha^{(\ell)})$ such that the torsion part of the congruence subgroup $G_{\varrho(\alpha^{(i)})}$ is smaller than that of $G_{\varrho(\alpha^{(i-1)})}$. Additionally, if G is finite then $G_{\varrho(\alpha^{(\ell)})}$ is torsion-free i.e. trivial.

In conclusion, notice that both `IsFiniteMatGroupFuncFF` and its modification construct an isomorphic copy of a finite subgroup G of $\mathrm{GL}(n, \mathbb{F})$ in some $\mathrm{GL}(n, q^k)$, $k \geq 1$. Thus we can compute orders if the input G is found to be finite, as before.

References

1. L. Babai, *Deciding finiteness of matrix groups in Las Vegas polynomial time*, Proceedings of the Third Annual ACM-SIAM Symposium on Discrete Algorithms (Orlando, FL, 1992) (New York), ACM, 1992, pp. 33–40.
2. L. Babai, R. Beals, and D. N. Rockmore, *Deciding finiteness of matrix groups in deterministic polynomial time*, Proc. of International Symposium on Symbolic and Algebraic Computation ISSAC '93 (ACM Press), 1993, pp. 117–126.
3. A. S. Detinko, *On deciding finiteness for matrix groups over fields of positive characteristic*, LMS J. Comput. Math. **4** (2001), 64–72 (electronic).
4. A. S. Detinko, B. Eick, and D. L. Flannery, Nilmat—Computing with nilpotent matrix groups. A refereed GAP 4 package; see http://www.gap-system.org/Packages/nilmat.html (2007).

5. A. S. Detinko and D. L. Flannery, *Algorithms for computing with nilpotent matrix groups over infinite domains*, J. Symbolic Comput. **43** (2008), 8–26.
6. B. Eick, *Computational group theory*, Jahresbericht der DMV 107, Heft 3 (2005), 155–170.
7. The GAP group, GAP - *Groups, Algorithms, and Programming*, Version 4.4.10, http://www.gap-system.org.
8. G. Ivanyos, *Deciding finiteness for matrix semigroups over function fields over finite fields*, Israel J. Math. **124** (2001), 185–188.
9. R. C. Lyndon and P. E. Schupp, *Combinatorial group theory*, Springer, 2001.
10. E. A. O'Brien, *Towards effective algorithms for linear groups*, Finite geometries, groups, and computation, Walter de Gruyter, Berlin, 2006, pp. 163–190.
11. D. N. Rockmore, K.-S. Tan, and R. Beals, *Deciding finiteness for matrix groups over function fields*, Israel J. Math. **109** (1999), 93–116.
12. D. A. Suprunenko, *Matrix groups*, Transl. Math. Monogr., vol. 45, American Mathematical Society, Providence, RI, 1976.
13. B. A. F. Wehrfritz, *Infinite linear groups*, Springer-Verlag, 1973.

DEPARTMENT OF MATHEMATICS, NATIONAL UNIVERSITY OF IRELAND, GALWAY, IRELAND

E-mail: alla.detinko@nuigalway.ie
dane.flannery@nuigalway.ie