Determinant Equivalence Test over Finite Fields and over $\mathbb{Q}$

Ankit Garg
Microsoft Research India, Bangalore, India
garga@microsoft.com

Nikhil Gupta
Department of Computer Science and Automation, Indian Institute of Science, India
nikhilg@iisc.ac.in

Neeraj Kayal
Microsoft Research India, Bangalore, India
neeraka@microsoft.com

Chandan Saha
Department of Computer Science and Automation, Indian Institute of Science, India
chandan@iisc.ac.in

Abstract

The determinant polynomial $\mathsf{Det}_n(\mathbf{x})$ of degree n is the determinant of a $n \times n$ matrix of formal variables. A polynomial f is equivalent to $\mathsf{Det}_n(\mathbf{x})$ over a field $\mathbb{F}$ if there exists a $A \in GL(n^2, \mathbb{F})$ such that $f = \mathsf{Det}_n(A \cdot \mathbf{x})$. *Determinant equivalence test over* $\mathbb{F}$ is the following algorithmic task: Given black-box access to a $f \in \mathbb{F}[\mathbf{x}]$, check if f is equivalent to $\mathsf{Det}_n(\mathbf{x})$ over $\mathbb{F}$, and if so then output a transformation matrix $A \in GL(n^2, \mathbb{F})$. In (Kayal, STOC 2012), a randomized polynomial time determinant equivalence test was given over $\mathbb{F} = \mathbb{C}$. But, to our knowledge, the complexity of the problem over finite fields and over $\mathbb{Q}$ was not well understood.

In this work, we give a randomized $\text{poly}(n, \log |\mathbb{F}|)$ time determinant equivalence test over finite fields $\mathbb{F}$ (under mild restrictions on the characteristic and size of $\mathbb{F}$). Over $\mathbb{Q}$, we give an efficient randomized reduction from factoring square-free integers to determinant equivalence test for quadratic forms (i.e. the $n = 2$ case), assuming GRH. This shows that designing a polynomial-time determinant equivalence test over $\mathbb{Q}$ is a challenging task. Nevertheless, we show that determinant equivalence test over $\mathbb{Q}$ is decidable: For bounded n, there is a randomized polynomial-time determinant equivalence test over $\mathbb{Q}$ with access to an oracle for integer factoring. Moreover, for *any* n, there is a randomized polynomial-time algorithm that takes input black-box access to a $f \in \mathbb{Q}[\mathbf{x}]$ and if f is equivalent to Det_n over $\mathbb{Q}$ then it returns a $A \in GL(n^2, \text{Ł})$ such that $f = \mathsf{Det}_n(A \cdot \mathbf{x})$, where Ł is an extension field of $\mathbb{Q}$ and $[\text{Ł} : \mathbb{Q}] \leq n$.

The above algorithms over finite fields and over $\mathbb{Q}$ are obtained by giving a polynomial-time randomized reduction from determinant equivalence test to another problem, namely the *full matrix algebra isomorphism* problem. We also show a reduction in the converse direction which is efficient if n is bounded. These reductions, which hold over any $\mathbb{F}$ (under mild restrictions on the characteristic and size of $\mathbb{F}$), establish a close connection between the complexity of the two problems. This then leads to our results via applications of known results on the full algebra isomorphism problem over finite fields (Rónyai, STOC 1987 and Rónyai, J. Symb. Comput. 1990) and over $\mathbb{Q}$ (Ivanyos et al., Journal of Algebra 2012 and Babai et al., Mathematics of Computation 1990).

2012 ACM Subject Classification Theory of computation → Algebraic complexity theory

Keywords and phrases Determinant equivalence test, full matrix algebra isomorphism, Lie algebra

Digital Object Identifier 10.4230/LIPIcs.ICALP.2019.62

Category Track A: Algorithms, Complexity and Games

Related Version A full version of the paper is available at `https://eccc.weizmann.ac.il/report/2019/042/`.

46th International Colloquium on Automata, Languages, and Programming (ICALP 2019).
Editors: Christel Baier, Ioannis Chatzigiannakis, Paola Flocchini, and Stefano Leonardi;
Article No. 62; pp. 62:1–62:15
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

Acknowledgements We would like to thank Youming Qiao for pointing us to the module decomposition algorithm in [4]. NG would like to thank Vineet Nair for discussions on the structure of the Lie algebra of Det. We thank him for sharing his proof of Theorem 10. We also thank anonymous reviewers for their comments.

1 Introduction

Two m-variate polynomials $f(\mathbf{x})$ and $g(\mathbf{x})$ with coefficients from a field $\mathbb{F}$ are said to be *equivalent over* $\mathbb{F}$ if there exists a $A \in GL(m, \mathbb{F})$ such that $f = g(A \cdot \mathbf{x})$. The algorithmic task of determining if f is equivalent to g, and if so then finding a linear transformation A such that $f = g(A \cdot \mathbf{x})$, is known as the polynomial *equivalence test* problem. It is a natural problem arising in algebraic complexity theory, becoming more important with the advent of Geometric Complexity Theory (GCT) [22] – which proposes the uses of deep tools and insights from group theory, representation theory and algebraic geometry towards the study of the VP vs VNP question.

A naïve approach for equivalence test is to reduce it to solving a system of polynomial equations over $\mathbb{F}$. But, unfortunately, polynomial solvability problem is NP-hard over $\mathbb{C}$ and finite fields and not known to be decidable over $\mathbb{Q}$. Nevertheless, it does appear that the complexity of equivalence test is much lower than the complexity of solving polynomial systems. It is known that over finite fields, the polynomial equivalence problem can not be NP-hard unless PH collapses (when the polynomials are given as lists of coefficients) [29, 28].

Can we hope to solve equivalence test over $\mathbb{C}$ and over finite fields [1] in (randomized) polynomial time? Finding such an algorithm is indeed quite demanding as it was shown in [1, 2] that the graph isomorphism problem reduces in polynomial time to equivalence test for cubic forms (i.e. homogeneous degree three polynomials) over *any* field. Over $\mathbb{Q}$, it is not even known if cubic form equivalence is decidable. On the other hand, we have a fairly good understanding of the complexity of quadratic form equivalence test: Over $\mathbb{C}$ and finite fields, equivalence of two quadratic forms can be tested in polynomial time due to well-known results on classification of quadratic forms. Quadratic form equivalence over $\mathbb{Q}$ can be done in polynomial-time with access to an oracle for integer factoring (IntFact). Moreover, IntFact reduces in randomized polynomial time to quadratic form equivalence over $\mathbb{Q}$ (see [31]). Given this state of affairs, designing efficient equivalence tests for even bounded degree polynomials seems like a difficult proposition. Indeed, there is a cryptographic authentication scheme based on the presumed average-case hardness of equivalence test for constant degree polynomials (see [23]).

The work in [15] initiated the study of a kind of equivalence test in which one polynomial f is given as input and the other polynomial g belongs to a well-defined polynomial family. Some of the polynomial families that are well-studied in algebraic complexity theory, particularly in the context of arithmetic circuit lower bounds, are those defined by the power symmetric polynomial, the elementary symmetric polynomial, the permanent, the determinant and the iterated matrix multiplication polynomial. In [15], randomized polynomial time equivalence tests over $\mathbb{C}$ were given for the power symmetric polynomial and the elementary symmetric polynomial families. These equivalence tests, which also hold over finite fields and $\mathbb{Q}$, work

[1] Typically, a computation model over $\mathbb{C}$ assumes that basic arithmetic operations with complex numbers and root finding of univariate polynomials over $\mathbb{C}$ can be done efficiently. Also, we will work with finite fields that have sufficiently large size and characteristic.

even if f is given as a black-box[2]. Henceforth, let us assume that the input polynomial f is given as a black-box. Subsequently, in [16], randomized polynomial time equivalence tests over $\mathbb{C}$ were given for the permanent and the determinant polynomial families. The test for the permanent holds over finite fields and $\mathbb{Q}$, but the same is *not true* for the determinant equivalence test in [16]. In [18], an equivalence test for the iterated matrix multiplication (IMM) was given which holds over $\mathbb{C}$, finite fields and $\mathbb{Q}$ (see also [13]). The iterated matrix multiplication and the determinant families have very similar circuit complexity: Both the families are complete under p-projections for class of algebraic branching programs (ABP) (see [20, 21]). But, it was unclear if determinant admits an efficient equivalence test over finite fields and $\mathbb{Q}$, just like the iterated matrix multiplication polynomial. In this paper, we fill in this gap in our understanding.

It is worth noting that determinant equivalence test is interesting in the context of the permanent versus determinant problem [30], which conjectures that the permanent is not an affine projection of a polynomial-size determinant. Geometric Complexity Theory [22], an approach to resolving this conjecture, suggests (among other things) to look for an algorithm to determine if the (padded) permanent is in the orbit closure of a polynomial-size determinant. In this language, determinant equivalence testing is the related problem of checking if a given polynomial is in the orbit of the determinant polynomial.

1.1 Our results

Let $n \in \mathbb{N}^{\times}$, $X = (x_{ij})_{i,j\in[n]}$ be a $n \times n$ matrix of formal variables, and $\mathbf{x} = (x_{11}\ x_{12}\ \ldots\ x_{n\,n-1}\ x_{nn})^T$ a column vector consisting of the variables in X arranged in a row-major fashion. The polynomial $\mathsf{Det}_n(\mathbf{x}) := \det(X)$; we will drop the subscript n whenever it is clear from the context. Hereafter, we will use the acronym DET for Determinant Equivalence Test.

▶ **Theorem 1** (DET over finite fields). *Let $\mathbb{F}$ be a finite field such that $|\mathbb{F}| \geq 10n^4$ and $char(\mathbb{F}) \nmid n(n-1)$. There is a randomized $\mathsf{poly}(n, \log|\mathbb{F}|)$ time algorithm that takes input black-box access to a $f \in \mathbb{F}[\mathbf{x}]$ of degree n and does the following with high probability: If f is equivalent to $\mathsf{Det}(\mathbf{x})$ over $\mathbb{F}$ then it outputs a $A \in GL(n^2, \mathbb{F})$ such that $f = \mathsf{Det}(A \cdot \mathbf{x})$; otherwise, it outputs "Fail".*

In [17], a DET over a finite field $\mathbb{F}_q$ was given that is similar to the equivalence test for the permanent in [16], but the test outputs a $A \in GL(n^2, \mathbb{F}_{q^n})$. Whereas, our algorithm (which is different and relatively more involved) outputs a $A \in GL(n^2, \mathbb{F}_q)$. One consequence of this is that the average-case ABP reconstruction algorithm in [17] holds over the base field $\mathbb{F}_q$.

▶ **Theorem 2** (DET over $\mathbb{Q}$).

(a) *There is a randomized algorithm, with oracle access to $\mathsf{IntFact}$, that takes input black-box access to a $f \in \mathbb{Q}[\mathbf{x}]$ of degree n and does the following with high probability: If f is equivalent to $\mathsf{Det}(\mathbf{x})$ over $\mathbb{Q}$ then it outputs a $A \in GL(n^2, \mathbb{Q})$ such that $f = \mathsf{Det}(A \cdot \mathbf{x})$; otherwise, it outputs "Fail". If n is bounded then the algorithm runs in time polynomial in the bit length of the coefficients of f.*

[2] An algorithm with black-box access to a m-variate polynomial f is only allowed to query the black-box for evaluations of f at points in $\mathbb{F}^m$.

(b) *There is a randomized algorithm that takes input black-box access to a $f \in \mathbb{Q}[\mathbf{x}]$ of degree n and does the following with high probability: If f is equivalent to* $\mathsf{Det}(\mathbf{x})$ *over $\mathbb{Q}$ then it outputs a $A \in GL(n^2, Ł)$ such that $f = \mathsf{Det}(A \cdot \mathbf{x})$, where Ł is an extension field of $\mathbb{Q}$ and $[Ł : \mathbb{Q}] \leq n$. The algorithm runs in time polynomial in n and the bit length of the coefficients of f.*

To our knowledge, it was not known if DET over $\mathbb{Q}$ is decidable prior to this work. It is natural to wonder if we can get rid of the IntFact oracle from part (a) of the above theorem. In this regard, we show the following.

▶ **Theorem 3** (IntFact reduces to DET for quadratic forms)**.** *Assuming GRH, we give a randomized polynomial-time reduction from factoring square-free integers to finding a $A \in M_2(\mathbb{Q})$ such that a given quadratic form $f \in \mathbb{Q}[\mathbf{x}]$ equals* $\mathsf{Det}_2(A \cdot \mathbf{x})$*, if f is equivalent to* Det_2*.*

The complexity of IntFact is the same as that of DET over $\mathbb{Q}$ for quadratic forms (modulo GRH and the use of randomization). Theorem 3 is a reduction from a result in [24].

Theorem 1 and 2 are proved by reducing DET to the *full matrix algebra isomorphism* problem. An $\mathbb{F}$-algebra $\mathcal{A}$ has two binary operations $+$ and $\cdot$ defined on its elements such that $(\mathcal{A}, +)$ is a $\mathbb{F}$-vector space, $(\mathcal{A}, +, \cdot)$ is an associative ring, and for every $a, b \in \mathbb{F}$ and $B, C \in \mathcal{A}$ it holds that $(aB)C = B(aC) = a(BC)$. For example, the set $M_n(\mathbb{F})$ of all $n \times n$ matrices over $\mathbb{F}$ is a $\mathbb{F}$-algebra with respect to the usual matrix addition and multiplication operations; it is called the full matrix algebra. Two $\mathbb{F}$-algebra $\mathcal{A}_1$ and $\mathcal{A}_2$ are isomorphic, denoted by $\mathcal{A}_1 \cong \mathcal{A}_2$, if there is a bijection ϕ from $\mathcal{A}_1$ to $\mathcal{A}_2$ such that for every $a, b \in \mathbb{F}$ and $B, C \in \mathcal{A}_1$ it holds that $\phi(aB + bC) = a\phi(B) + b\phi(C)$ and $\phi(BC) = \phi(B)\phi(C)$. Any finite dimensional $\mathbb{F}$-algebra is isomorphic to a $\mathbb{F}$-algebra $\mathcal{A}' \subseteq M_m(\mathbb{F})$, where $m = \dim_{\mathbb{F}}(\mathcal{A})$. A $\mathbb{F}$-algebra $\mathcal{A} \subseteq M_m(\mathbb{F})$ can be specified by a $\mathbb{F}$-basis $B_1, \ldots, B_r \in M_m(\mathbb{F})$.

▶ **Definition 4.** *The full matrix algebra isomorphism (FMAI) problem over $\mathbb{F}$ is the following: Given a basis of a $\mathbb{F}$-algebra $\mathcal{A} \subseteq M_m(\mathbb{F})$, check if $\mathcal{A} \cong M_n(\mathbb{F})$, where $n^2 = \dim_{\mathbb{F}}(\mathcal{A})$. If $\mathcal{A} \cong M_n(\mathbb{F})$ then output an isomorphism from $\mathcal{A}$ to $M_n(\mathbb{F})$.*

In [24, 25], a $\mathsf{poly}(m, \log|\mathbb{F}|)$ time randomized algorithm was given to solve FMAI over a finite field $\mathbb{F}$. Over $\mathbb{Q}$, the FMAI problem is more difficult. In [14, 6], a randomized algorithm (with access to a IntFact oracle) was given to solve FMAI over $\mathbb{Q}$. The algorithm runs in polynomial-time if $\dim_{\mathbb{Q}}(\mathcal{A})$ is bounded. In [3, 10], randomized polynomial time algorithms were given to compute an isomorphism from $\mathcal{A} \otimes_{\mathbb{Q}} Ł$ to $M_n(Ł)$ for some extension field $Ł \supseteq \mathbb{Q}$ satisfying $[Ł : \mathbb{Q}] \leq n$, if $\mathcal{A} \cong M_n(\mathbb{Q})$ to begin with. We give a randomized polynomial-time reduction from DET to FMAI over any sufficiently large $\mathbb{F}$ in Section 4, thereby proving Theorem 1 and 2. The reduction is obtained by giving an algorithm to decompose the Lie algebra of f into its two simple Lie subalgebras over any sufficiently large $\mathbb{F}$ (see Section 3). The same reduction also gives DET over $\mathbb{R}$ and $\mathbb{C}$ via the FMAI algorithms in [9, 26] (thereby giving alternative algorithms to the one presented in [16]). We also show a reduction from FMAI to DET (in Section 7) which is efficient if the dimension n is bounded.

The above results underscore the close connection between the DET and the FMAI problems. In order to get efficient DET over $\mathbb{Q}$ for even bounded degree polynomials, we *need to* solve FMAI efficiently for $\mathbb{Q}$-algebras of bounded dimensions. Currently, the best known algorithm for FMAI over $\mathbb{Q}$ uses an IntFact oracle [14]. This situation of the determinant is somewhat surprising as it contrasts that of IMM (the close cousin of the determinant) – IMM equivalence test over $\mathbb{Q}$ can be solved efficiently for polynomials of degree greater than four [18].

2 Preliminaries

2.1 Notations

The set of trace zero or traceless matrices in $M_n(\mathbb{F})$ is denoted by $\mathcal{Z}_n(\mathbb{F})$; we will drop $\mathbb{F}$ from $M_n(\mathbb{F})$ and $\mathcal{Z}_n(\mathbb{F})$ when it is clear from the context. Let I_n be the $n \times n$ identity matrix. $\otimes$ denotes tensor product of two matrices. Define,

$$\mathcal{M}_{\text{col}} := I_n \otimes M_n, \quad \mathcal{M}_{\text{row}} := M_n \otimes I_n \qquad \text{and} \qquad \mathcal{L}_{\text{col}} := I_n \otimes \mathcal{Z}_n, \quad \mathcal{L}_{\text{row}} := \mathcal{Z}_n \otimes I_n.$$

Observe $\mathcal{M}_{\text{col}}$, $\mathcal{M}_{\text{row}} \subseteq M_{n^2}$ are $\mathbb{F}$-algebras isomorphic to M_n, and $\mathcal{L}_{\text{col}}$, $\mathcal{L}_{\text{row}}$ are their subspaces , respectively, of dimension $n^2 - 1$ each. Henceforth, we set $m = n^2$ and $r = n^2 - 1$.

2.2 Definitions

▶ **Definition 5** (Lie bracket). *For $A, B \in M_n$, the Lie bracket operation $[A, B] := AB - BA$.*

▶ **Definition 6** (Lie algebra of a polynomial). *The Lie algebra $\mathfrak{g}_f$ of a m-variate polynomial $f \in \mathbb{F}[\mathbf{x}]$ is the set of matrices $B = (b_{i,j})_{i,j\in[m]}$ satisfying,*

$$\sum_{i,j\in[m]} b_{i,j} \cdot x_j \cdot \frac{\partial f}{\partial x_i} = 0.$$

It is easy to verify that $[\cdot,\cdot]$ is a $\mathbb{F}$-bilinear map on M_n, and $\mathfrak{g}_f$ is an $\mathbb{F}$-vector space.[3] Let $\mathcal{V}$ be a $\mathbb{F}$-vector space, $\text{End}_{\mathbb{F}}(\mathcal{V}) := \{\varphi : \varphi \text{ is a } \mathbb{F}\text{-linear map from } \mathcal{V} \text{ to } \mathcal{V}\}$ and $\mathcal{T} \subseteq \text{End}_{\mathbb{F}}(\mathcal{V})$.

▶ **Definition 7.** *A subspace $\mathcal{U}$ of $\mathcal{V}$ is called $\mathcal{T}$-invariant if for every $\varphi \in \mathcal{T}$, $\varphi(\mathcal{U}) \subseteq \mathcal{U}$.*

If $\mathcal{T} \subseteq M_{2r}$, the terminology "invariant subspace of $\mathcal{T}$" means $\mathcal{T}$-invariant subspace of $\mathbb{F}^{2r}$.

▶ **Definition 8** (Irreducible invariant subspace). *A $\mathcal{T}$-invariant subspace $\mathcal{U}$ of $\mathcal{V}$ is irreducible if there do not exist proper $\mathcal{T}$-invariant subspaces $\mathcal{U}_1, \mathcal{U}_2$ of $\mathcal{U}$, such that $\mathcal{U} = \mathcal{U}_1 \oplus \mathcal{U}_2$.*

▶ **Definition 9** (Closure of a vector). *Let $\mathbf{w} \in \mathcal{V}$. Then, the closure of $\mathbf{w}$ with respect to $\mathcal{T}$, denoted $\text{closure}_{\mathcal{T}}(\mathbf{w})$, is the smallest $\mathcal{T}$-invariant subspace of $\mathcal{V}$ containing $\mathbf{w}$.*

2.3 Some basic results

▶ **Observation 2.1.** *For $i, j \in [n], i \neq j$, let $E_{ij} \in M_n$ be such that the (i,j)-th entry is 1 and other entries are 0, and for $\ell \in [2, n]$, let $E_\ell \in M_n$ be a diagonal matrix with the $(1,1)$-th and (ℓ,ℓ)-th entries as 1 and -1 respectively and other entries as 0. Then,*

1. *$\{I_n \otimes E_{ij},\ I_n \otimes E_\ell \ :\ i, j \in [n], i \neq j, \text{ and } \ell \in [2, n]\}$ is a basis of $\mathcal{L}_{col}$. Denote the elements of this standard basis as $S_1, \ldots, S_r$.*
2. *$\{E_{ij} \otimes I_n,\ E_\ell \otimes I_n \ :\ i, j \in [n], i \neq j, \text{ and } \ell \in [2, n]\}$ is a basis of $\mathcal{L}_{row}$. Denote the elements of this standard basis as $S_{r+1}, \ldots, S_{2r}$.*

▶ **Observation 2.2.** *For every $F \in \mathcal{M}_{row}$ and $L \in \mathcal{M}_{col}$, $[F, L] = FL - LF = 0$.*

▶ **Observation 2.3.** *For every $L_1, L_2 \in \mathcal{L}_{col}$ (or $\mathcal{L}_{row}$), $[L_1, L_2] \in \mathcal{L}_{col}$ (respectively. $\mathcal{L}_{row}$).*

[3] Over $\mathbb{C}$, $\mathfrak{g}_f$ also turns out to be a Lie algebra i.e. closed under the Lie bracket operation. However, over finite fields, it is not clear if it is closed under the bracket operation. We still stick with the terminology Lie algebra of a polynomial since in many cases, it does turn out to be closed under the bracket operation.

A proof of the following standard fact is given in Section A.1 of the Appendix of the full version [11].

▶ **Fact 1.** *Let $B \in M_n$. Then, the dimension of the space of matrices in M_n that commute with B is at least n, and the dimension of the space of matrices in $\mathcal{Z}_n$ that commute with B is at least $n - 1$.*

We would also need the following facts (see [16, 18] for their proofs).

▶ **Fact 2.** *If $g \in \mathbb{F}[\mathbf{x}]$ and $f(\mathbf{x}) = g(A \cdot \mathbf{x})$ for some $A \in GL(m, \mathbb{F})$ then $\mathfrak{g}_f = A^{-1} \cdot \mathfrak{g}_g \cdot A$.*

▶ **Fact 3.** *Suppose we have black box access to a m-variate polynomial $f \in \mathbb{F}[\mathbf{x}]$, where $|\mathbb{F}| \geq 2n^3$. Then, a basis of $\mathfrak{g}_f$ can be computed in randomized polynomial time.*

▶ **Fact 4.** *Given a basis $\{T_1, \ldots, T_s\}$ of $\mathcal{T} \subseteq M_{2r}$ and a $\mathbf{w} \in \mathbb{F}^{2r}$, a basis of $\mathrm{closure}_{\mathcal{T}}(\mathbf{w})$ can be computed in time polynomial in r and the bit length of the entries in $\mathbf{w}$ and $T_1, \ldots, T_s$.*

The following theorem on the Lie algebra of Det is well-known over $\mathbb{C}$. We give a proof over any field (with a mild condition on the characteristic) in Section A.2 of the Appendix of [11].

▶ **Theorem 10** (Lie algebra of Det)**.** *Let $n \geq 2$ and $\mathbb{F}$ be a field such that $char(\mathbb{F}) \nmid n$. Then, the Lie algebra of Det_n equals the direct sum of the spaces $\mathcal{L}_{row}$ and $\mathcal{L}_{col}$, i.e., $\mathfrak{g}_{\mathsf{Det}} = \mathcal{L}_{row} \oplus \mathcal{L}_{col}$.*

The theorem implies that $\{S_1, \ldots, S_{2r}\}$, in Observation 2.1, forms a basis of $\mathfrak{g}_{\mathsf{Det}}$. The rows and columns of every element in $\mathfrak{g}_{\mathsf{Det}}$ are indexed by the $\mathbf{x}$ variables, in order. Let $f = \mathsf{Det}(A \cdot \mathbf{x})$ for some $A \in GL(m, \mathbb{F})$. Then, Theorem 10 and Fact 2 imply that $\mathfrak{g}_f = A^{-1} \cdot \mathcal{L}_{\mathrm{row}} \cdot A \ \oplus \ A^{-1} \cdot \mathcal{L}_{\mathrm{col}} \cdot A$. We denote $A^{-1} \cdot \mathcal{L}_{\mathrm{row}} \cdot A$ and $A^{-1} \cdot \mathcal{L}_{\mathrm{col}} \cdot A$ by $\mathcal{F}_{\mathrm{row}}$ and $\mathcal{F}_{\mathrm{col}}$ respectively, and refer to $\mathcal{F}_{\mathrm{row}}$ and $\mathcal{F}_{\mathrm{col}}$ (similarly, $\mathcal{L}_{\mathrm{row}}$ and $\mathcal{L}_{\mathrm{col}}$) as the Lie subalgebras of $\mathfrak{g}_f$ (respectively, $\mathfrak{g}_{\mathsf{Det}}$) [4]. From Theorem 10, Observation 2.2 and 2.3, we get the following.

▶ **Observation 2.4.** *For every $E, F \in \mathfrak{g}_f$, $[E, F] \in \mathfrak{g}_f$.*

▶ **Observation 2.5.** *Let $\mathcal{A} \subseteq M_m$ be the $\mathbb{F}$-algebra generated by a basis of $\mathcal{F}_{col}$. Then,*

$$\mathcal{A} = A^{-1} \cdot (I_n \otimes M_n) \cdot A.$$

This can be proved easily. Finally, we record a special case of the Skolem-Noether theorem which will be used in Section 4. Its general statement can be found in [19] (page 173).

▶ **Theorem 11** (Skolem-Noether)**.** *Let $n, s \in \mathbb{N}^{\times}$ such that $n \mid s$, and $\mathcal{A} \subseteq M_s$ be a $\mathbb{F}$-algebra (containing I_s) that is isomorphic to M_n via $\phi : M_n \to \mathcal{A}$. Then there exists a $K \in GL(s, \mathbb{F})$ s.t.,*

$$\phi(C) = K^{-1} \cdot (I_{s/n} \otimes C) \cdot K, \qquad \text{for every } C \in M_n.$$

3 Decomposition of $\mathfrak{g}_f$ into its Lie subalgebras

We show how to compute bases of $\mathcal{F}_{\mathrm{row}}$ and $\mathcal{F}_{\mathrm{col}}$ from black box access to $f = \mathsf{Det}(A \cdot \mathbf{x})$.

▶ **Theorem 12** (Decomposition of $\mathfrak{g}_f$)**.** *Let $n \geq 2$, $|\mathbb{F}| \geq 10n^4$ and $\mathrm{char}(\mathbb{F}) \nmid n(n-1)$. There is a randomized algorithm, which takes input black box access to f and outputs bases of $\mathcal{F}_{row}$ and $\mathcal{F}_{col}$ with high probability. The running time is $\mathrm{poly}(n, \gamma)$, where γ is the bit length of coefficients of f.*

We first present the proof idea, and then the algorithm and its proof of correctness. The missing proofs are given in Sections B,C and D of the Appendix of [11].

[4] Observation 2.3 implies that $\mathcal{F}_{\mathrm{row}}$ and $\mathcal{F}_{\mathrm{col}}$ are closed under the Lie bracket operation and hence they are matrix Lie algebras.

3.1 Proof of Theorem 12: The idea

The algorithm relies on finding the irreducible invariant subspaces of a set of $\mathbb{F}$-linear maps on $\mathfrak{g}_f$. These linear maps (a.k.a adjoint homomorphisms of $\mathfrak{g}_f$) are defined for every $F \in \mathfrak{g}_f$,

$$\rho_F \ : \ \mathfrak{g}_f \to \mathfrak{g}_f \ \ ; \ \ E \mapsto [E, F].$$

It is easy to see that ρ_F is linear. Let $\{B_1, \ldots, B_{2r}\}$ be a basis of $\mathfrak{g}_f$ which can be computed in randomized polynomial time (by Fact 3). As ρ_F is $\mathbb{F}$-linear, we can associate a matrix $P_F \in M_{2r}$ with ρ_F, after fixing an ordering of the basis $(B_1, \ldots, B_{2r})$. Let $\mathcal{P} := \{P_F \ : \ F \in \mathfrak{g}_f\}$.

▷ Claim 13. $\mathfrak{g}_f$ and $\mathcal{P}$ are isomorphic as vector spaces via the map $F \mapsto P_F$ for every $F \in \mathfrak{g}_f$.

Its proof is given in Section B.1 of the Appendix of [11]. This implies the following.

▶ **Observation 3.1.** *The matrices $\{P_{B_1}, \ldots, P_{B_{2r}}\}$ is a basis of $\mathcal{P}$, which can be efficiently computed from $\{B_1, \ldots, B_{2r}\}$ (by considering the elements $[B_i, B_j]$, for $i, j \in [2r]$).*

We intend to study the irreducible invariant subspaces of $\mathcal{P}$ in order to compute bases of $\mathcal{F}_{\text{row}}$ and $\mathcal{F}_{\text{col}}$. The following Claim 14 would be useful in this regard.

It follows from Fact 2 that $J_i := A \cdot B_i \cdot A^{-1}$, for $i \in [2r]$, is a basis of $\mathfrak{g}_{\mathsf{Det}}$. Like ρ_F, we can associate a $\mathbb{F}$-linear map (i.e. adjoint homomorphism) χ_L with every $L \in \mathfrak{g}_{\mathsf{Det}}$ as follows:

$$\chi_L : \ \mathfrak{g}_{\mathsf{Det}} \to \mathfrak{g}_{\mathsf{Det}} \ \ ; \ \ K \mapsto [K, L].$$

Let $Q_L \in M_{2r}$ be the matrix corresponding to the linear map χ_L, with respect to the (ordered) basis $(J_1, \ldots, J_{2r})$. The following claim implies that $\mathcal{P}$ does not depend on the transformation matrix A. Thus, it is sufficient to focus on $\mathfrak{g}_{\mathsf{Det}}$ to study the invariant subspaces of $\mathcal{P}$. The proof of the claim is given in Section B.2 of the Appendix of [11].

▷ Claim 14. For every $i \in [2r]$, $Q_{J_i} = P_{B_i}$ and so the space $\mathcal{P} = \{Q_L \ : \ L \in \mathfrak{g}_{\mathsf{Det}}\}$.

Like Claim 13, $\mathfrak{g}_{\mathsf{Det}}$ and $\mathcal{P}$ are isomorphic as vector spaces via the map $L \mapsto Q_L$, for $L \in \mathfrak{g}_{\mathsf{Det}}$. The algorithm computes two invariant subspaces $\mathcal{V}_1$ and $\mathcal{V}_2$ of $\mathcal{P}$ that are defined as follows

$$\mathcal{V}_1 = \left\{ \mathbf{v} = (a_1, \ldots, a_{2r})^T \in \mathbb{F}^{2r} \ : \ \sum_{i \in [2r]} a_i \cdot J_i \in \mathcal{L}_{\text{col}} \right\},$$

$$\mathcal{V}_2 = \left\{ \mathbf{v} = (b_1, \ldots, b_{2r})^T \in \mathbb{F}^{2r} \ : \ \sum_{i \in [2r]} b_i \cdot J_i \in \mathcal{L}_{\text{row}} \right\}. \tag{1}$$

Clearly, $\dim(\mathcal{V}_1) = \dim(\mathcal{V}_2) = r$. As $B_i = A^{-1} \cdot J_i \cdot A$, for $i \in [2r]$, we get

$$\mathcal{V}_1 = \left\{ \mathbf{v} = (a_1, \ldots, a_{2r})^T \in \mathbb{F}^{2r} \ : \ \sum_{i \in [2r]} a_i \cdot B_i \in \mathcal{F}_{\text{col}} \right\},$$

$$\mathcal{V}_2 = \left\{ \mathbf{v} = (b, \ldots, b_{2r})^T \in \mathbb{F}^{2r} \ : \ \sum_{i \in [2r]} b_i \cdot B_i \in \mathcal{F}_{\text{row}} \right\}. \tag{2}$$

From bases of $\mathcal{V}_1$ and $\mathcal{V}_2$, and $(B_1, \ldots, B_{2r})$, we get bases of $\mathcal{F}_{\text{col}}$ and $\mathcal{F}_{\text{row}}$ readily. The aspects of the space $\mathcal{P}$ that help in computing $\mathcal{V}_1$ and $\mathcal{V}_2$ are the facts that these are the only two irreducible invariant subspaces of $\mathcal{P}$ and bases of these can be computed from a random element of $\mathcal{P}$. These facts are proved in the proof of correctness of Algorithm 1.

3.2 The decomposition algorithm

Algorithm 1 Computation of bases of $\mathcal{F}_{row}$ and $\mathcal{F}_{col}$.

```
Input: Black box access to $f$.
Output: Bases of spaces $\mathcal{V}_1$ and $\mathcal{V}_2$ (as in Equation (2)).

Compute a basis $B_1, \ldots, B_{2r}$ of $\mathfrak{g}_f$ (see Fact 3), and form the basis $P_{B_1}, \ldots, P_{B_{2r}}$ of $\mathcal{P}$.
Pick a random element $Q = r_1 P_{B_1} + \cdots + r_{2r} P_{B_{2r}}$ from $\mathcal{P}$, where every $r_i$ is chosen uniformly and independently at random from a fixed subset of $\mathbb{F}$ of size $10n^4$.
Compute the characteristic polynomial $h(z)$ of $Q$.
Factor $h(z)$ into irreducible factors over $\mathbb{F}$. Let $h(z) = z^{2(n-1)} \cdot h_1(z) \cdots h_k(z)$, where $z, h_1, \ldots, h_k$ are mutually coprime and irreducible. If this is not the split, output "Fail".
For every $i \in [k]$, compute a basis of the null space $\mathcal{N}_i$ of $h_i(Q)$, pick a vector $\mathbf{v}$ from the basis of $\mathcal{N}_i$ and compute a basis of $\mathcal{C}_i := \text{closure}_{\mathcal{P}}(\mathbf{v})$ (using Fact 4).
Remove repetitive spaces from the set $\{\mathcal{C}_1, \ldots, \mathcal{C}_k\}$. After this, if we are *not* left with exactly two spaces $\mathcal{U}_1$ and $\mathcal{U}_2$ then output "Fail". Else, output bases of $\mathcal{U}_1$ and $\mathcal{U}_2$.
```

3.3 Analysis of the algorithm

Let us view the space $\mathcal{P}$ through the lens of a convenient basis of $\mathfrak{g}_{\mathsf{Det}}$, namely the standard basis $\{S_1, \ldots, S_{2r}\}$ (given in Observation 2.1). For $K \in \mathfrak{g}_{\mathsf{Det}}$, let $\mathbf{w}_K, \mathbf{v}_K \in \mathbb{F}^{2r}$ be the coordinate vectors of K with respect to the ordered bases $(S_1, \ldots, S_{2r})$ and $(J_1, \ldots, J_{2r})$ respectively. There is a basis change matrix $H \in \mathrm{GL}(2r, \mathbb{F})$, such that for every $K \in \mathfrak{g}_{\mathsf{Det}}$,

$$\mathbf{v}_K = H \cdot \mathbf{w}_K. \tag{3}$$

Recall Q_L from Claim 14. Let $R_L := H^{-1} \cdot Q_L \cdot H$, for every $L \in \mathfrak{g}_{\mathsf{Det}}$, and

$$\mathcal{R} := \{R_L \ : \ L \in \mathfrak{g}_{\mathsf{Det}}\} = H^{-1} \cdot \mathcal{P} \cdot H. \tag{4}$$

Observe that $\{R_{S_1}, \ldots, R_{S_{2r}}\}$ is a basis of $\mathcal{R}$. Also, for every $L, K \in \mathfrak{g}_{\mathsf{Det}}$.

$$R_L \cdot \mathbf{w}_K = \mathbf{w}_{[K,L]}, \tag{5}$$

▶ **Observation 3.2.** *Every $R \in \mathcal{R} \subseteq M_{2r}$ is a block diagonal matrix having two blocks of size $r \times r$ each, i.e, the non-zero entries of R are confined to the entries $\{(S_i, S_j) \ : \ i, j \in [r]\}$ and $\{(S_i, S_j) \ : \ i, j \in [r+1, 2r]\}$.*

The proof of Observation 3.2 is given in Section C.1 of the Appendix of [11]. We refer to the two blocks of R as $R^{(1)}$ and $R^{(2)}$, corresponding to $\{S_1, \ldots, S_r\}$ and $\{S_{r+1}, \ldots, S_{2r}\}$, respectively. Observation 3.3 follows directly from definition of $\mathcal{R}$.

▶ **Observation 3.3.** *$\mathcal{W}$ is an invariant subspace of $\mathcal{R}$ iff $H \cdot \mathcal{W}$ is an invariant subspace of $\mathcal{P}$.*

It allows us to switch from $\mathcal{P}$ to $\mathcal{R}$ while studying the invariant subspaces of $\mathcal{P}$. The following lemmas on the invariant subspaces of $\mathcal{R}$ are crucial in arguing the correctness of Algorithm 1. Their proofs are given in Sections C.2 and C.3 of the Appendix of [11].

▶ **Lemma 15** (Irreducible invariant subspaces). *Let $\mathbf{w}_K \in \mathbb{F}^{2r}$ for a nonzero K in $\mathcal{L}_{col}$ or in $\mathcal{L}_{row}$.*

$$\textit{Then, } \begin{aligned} \text{closure}_{\mathcal{R}}(\mathbf{w}_K) &= \{\mathbf{w}_L \ : \ L \in \mathcal{L}_{col}\} =: \mathcal{W}_1, \quad \text{if } K \in \mathcal{L}_{\text{col}}, \\ \text{closure}_{\mathcal{R}}(\mathbf{w}_K) &= \{\mathbf{w}_L \ : \ L \in \mathcal{L}_{row}\} =: \mathcal{W}_2, \quad \text{if } K \in \mathcal{L}_{\text{row}}. \end{aligned}$$

Moreover, $\mathcal{W}_1$ and $\mathcal{W}_2$ are the only two irreducible invariant subspaces of $\mathcal{R}$, and $\mathbb{F}^{2r} = \mathcal{W}_1 \oplus \mathcal{W}_2$.

▶ **Lemma 16** (Characteristic polynomial). *Let $R = \sum_{i\in[2r]} \ell_i(r_1,\ldots,r_{2r})\cdot R_{S_i}$, where $\ell_1,\ldots,\ell_{2r}$ are $\mathbb{F}$-linearly independent linear forms and $r_1,\ldots,r_{2r}$ are picked uniformly and independently at random from a fixed subset of $\mathbb{F}$ of size $10n^4$. Then, with high probability, the characteristic polynomial $h_R(z)$ of R factors as $z^{2(n-1)}\cdot h_1(z)\cdots h_k(z)$, where $z, h_1(z),\ldots,h_k(z)$ are mutually coprime irreducible polynomials over $\mathbb{F}$.*

3.3.1 Proof of correctness of Algorithm 1

In Step 2, we choose a random Q from $\mathcal{P}$. By Equation (4), there is a $R \in \mathcal{R}$, such that,

$$R = H^{-1}\cdot Q\cdot H \;=\; r_1R_{J_1}+\cdots+r_{2r}R_{J_{2r}} \;=\; \ell_1(r_1,\ldots,r_{2r})\cdot R_{S_1}+\cdots+\ell_{2r}(r_1,\ldots,r_{2r})\cdot R_{S_{2r}},$$

where $\ell_1,\ldots,\ell_{2r}$ are $\mathbb{F}$-linearly independent linear forms in $r_1,\ldots,r_{2r}$. By Lemma 16, Step 4 holds with high probability. From Observation 3.2, R is a block diagonal matrix with blocks $R^{(1)}$ and $R^{(2)}$. Let $h(z) = g_1(z)\cdot g_2(z)$, where $g_1(z)$ and $g_2(z)$ are the characteristic polynomials of $R^{(1)}$ and $R^{(2)}$, respectively. There are a couple of factors of h, say h_1 and h_2, that divide g_1 and g_2, respectively. In Step 5, we compute the null spaces $\mathcal{N}_1$ and $\mathcal{N}_2$ of $h_1(Q)$ and $h_2(Q)$ respectively. As $h_1(R) = H^{-1}\cdot h_1(Q)\cdot H$ and $h_2(R) = H^{-1}\cdot h_2(Q)\cdot H$, the null spaces of $h_1(R)$ and $h_2(R)$, denoted by $\mathcal{O}_1$ and $\mathcal{O}_2$ respectively, satisfy $\mathcal{O}_1 = H^{-1}\cdot\mathcal{N}_1$ and $\mathcal{O}_2 = H^{-1}\cdot\mathcal{N}_2$ (due to Equation (3)).

▷ Claim 17. If $\mathbf{w}_K \in \mathcal{O}_1$ (similarly, $\mathbf{w}_K \in \mathcal{O}_2$) then $K \in \mathcal{L}_{\text{col}}$ (respectively, $K \in \mathcal{L}_{\text{row}}$).

Its proof is given in Section D.1 of the Appendix of [11]. In Step 5, we also pick a vector $\mathbf{v}$ from a null space, say $\mathcal{N}_1$, and compute closure$_{\mathcal{P}}(\mathbf{v})$. Clearly, $\mathbf{v} = \mathbf{v}_K$ for some $K \in \mathfrak{g}_{\mathsf{Det}}$. So, $\mathbf{v}_K \in \mathcal{N}_1$ if and only if $\mathbf{w}_K = H^{-1}\cdot\mathbf{v}_K \in \mathcal{O}_1$. As $\mathcal{R} = H^{-1}\cdot\mathcal{P}\cdot H$, Observation 3.3 implies

$$\begin{aligned}\text{closure}_{\mathcal{P}}(\mathbf{v}_K) &= H\cdot\text{closure}_{\mathcal{R}}(\mathbf{w}_K)\\ &= H\cdot\mathcal{W}_1 \qquad (\text{ by Claim 17 and Lemma 15})\\ &= \mathcal{V}_1 \qquad\quad (\text{ by Equations (1) and (3), as } \mathcal{V}_1 = \{\mathbf{v}_L \;:\; L\in\mathcal{L}_{\text{col}}\}).\end{aligned}$$

Similarly, if we pick a $\mathbf{v}\in\mathcal{N}_2$ then closure$_{\mathcal{P}}(\mathbf{v}) = \mathcal{V}_2$. Thus, in Step 6, one of $\mathcal{U}_1$ and $\mathcal{U}_2$ is $\mathcal{V}_1$ and the other is $\mathcal{V}_2$. Finally, we can take $\mathcal{U}_1 = \mathcal{V}_1$ and $\mathcal{U}_2 = \mathcal{V}_2$ without loss of generality: Let $P \in M_m$ be the permutation matrix, such that when multiplied to $\mathbf{x}$, P maps x_{ij} to x_{ji}. Clearly, $P^{-1} = P$. The following equation holds because P is a symmetry of Det.

$$\mathsf{Det}(\mathbf{x}) = \mathsf{Det}(P\cdot\mathbf{x}) \quad\text{and hence}\quad f(\mathbf{x}) = \mathsf{Det}(A\cdot\mathbf{x}) = \mathsf{Det}(PA\cdot\mathbf{x}).$$

Observe that $\mathcal{L}_{\text{col}} = P^{-1}\cdot\mathcal{L}_{\text{row}}\cdot P$. Hence,

$$\mathcal{F}_{\text{col}} = A^{-1}P^{-1}\cdot\mathcal{L}_{\text{row}}\cdot PA \quad\text{and}\quad \mathcal{F}_{\text{row}} = A^{-1}P^{-1}\cdot\mathcal{L}_{\text{col}}\cdot PA.$$

As the transformation matrix is unknown to the algorithm, we can take it to be either A or PA.

A comparison with [8], [4]: In [8, 7], a polynomial time algorithm was given to decompose a semisimple Lie algebra over $\mathbb{Q}$ (more generally, a characteristic 0 field) into a direct sum of simple Lie subalgebras. The Lie algebra $\mathfrak{g}_{\mathsf{Det}}$ is semisimple and $\mathcal{L}_{\text{col}}$ and $\mathcal{L}_{\text{row}}$ are its two simple Lie subalgebras. So, our decomposition problem is a special case of the problem studied in [8]. However, our algorithm works over any sufficiently large field $\mathbb{F}$ (in particular, finite fields), if char$(F) \nmid n(n-1)$. It is not quite clear to us if the algorithm in [8] (which is somewhat different from our algorithm) can be easily adapted to achieve the same result in this special case. Lemma 15 shows that the decomposition of $\mathbb{F}^{2r}$ into irreducible invariant subspaces of $\mathcal{R}$ is unique. Using this information, it is possible to use the module decomposition algorithm in [4] to compute bases of $\mathcal{F}_{\text{col}}$ and $\mathcal{F}_{\text{row}}$ in randomized polynomial time over finite fields. However, the module decomposition algorithm in [4] does not work in general over $\mathbb{Q}$ without moving to an extension field.

A comparison with [12, 13]: In [12] and Section 4.9 of [13], a DET over $\mathbb{C}$ was given by reducing it to the Lie algebra conjugacy problem. It was also suggested there that the approach can be made to work over finite field by reduction to the problem of finding and diagonalizing split Cartan subalgebra and then applying Ryba's algorithm [27]. However, it is not quite clear to us how to carry out this approach in full details. Despite the similarities between these two approaches originating from the use of Lie algebra, our approach of reducing DET to FMAI does appear somewhat different from the approach suggested in [12, 13].

4 Reduction of DET to FMAI

We give a randomized polynomial time reduction from DET to the FMAI problem. Recall the FMAI problem from Definition 4: An algorithm for FMAI takes input an ordered basis $(L_1, \ldots, L_m)$ of a $\mathbb{F}$-algebra $\mathcal{A} \subseteq M_s$ such that $\mathcal{A} \cong M_n$, and outputs a $\mathbb{F}$-algebra isomorphism $\phi : \mathcal{A} \to M_n$ in the form of an ordered basis $(C_1, \ldots, C_m)$ of M_n, where $C_i = \phi(L_i)$ for $i \in [m]$.

▶ **Lemma 18** (Reduction of DET to FMAI). *Let $n \geq 2$, $|\mathbb{F}| > 10n^4$ and $\text{char}(\mathbb{F}) \nmid n(n-1)$. Then, there exists a randomized algorithm, with oracle access to FMAI, that takes input black-box access to a $f \in \mathbb{F}[\mathbf{x}]$ of degree n and solves DET for f over $\mathbb{F}$ with high probability. The running time of the algorithm is polynomial in n and the bit length of the coefficients of f.*

The proof of this lemma follows from the proof of correctness of the following algorithm.

4.1 The algorithm

Algorithm 2 Reduction of DET to FMAI.

```
Input: Black-box access to $f \in \mathbb{F}[\mathbf{x}]$ of degree $n$, and oracle access to an algo for FMAI.
Output: $B \in \text{GL}(m, \mathbb{F})$ such that $f = \text{Det}(B \cdot \mathbf{x})$, if such a $B$ exists. Else, output "Fail".

Invoke Algorithm 1. Let $\{U_1, \ldots, U_r\}$ be the basis of the space $\mathcal{U}_1$ returned by Algorithm 1, where $\mathcal{U}_1 = \mathcal{F}_{\text{col}}$.
Generate a basis $\{L_1, \ldots, L_k\}$ of the algebra $\mathcal{A} := \mathbb{F}[U_1, \ldots, U_r]$. If $k \neq m$, output "Fail".
Invoke the FMAI oracle on $(L_1, \ldots, L_m)$ which returns a basis $(C_1, \ldots, C_m)$ of $M_n$.
Pick a random $M \in M_m$ satisfying $L_i \cdot M = M \cdot (I_n \otimes C_i)$ for every $i \in [m]$.
Let $b$ be the evaluation of $f(M \cdot \mathbf{x})$ at $x_{11} = \ldots = x_{nn} = 1$ and remaining $x_{ij}$ set to 0.
If $M \notin \text{GL}(m, \mathbb{F})$ or $b = 0$, output "Fail". Else, set $D = \text{diag}(b, 1, \ldots, 1) \in M_n$. Output $(I_n \otimes D) \cdot M^{-1}$.
```

4.2 Proof of correctness of Algorithm 2

If f is not equivalent to Det then it can be detected with high probability by checking if $f(\mathbf{a}) = b \cdot \mathsf{Det}(M^{-1}\mathbf{a})$ at a random point $\mathbf{a} \in_r S^m$, where $S \subseteq \mathbb{F}$ is sufficiently large. So, assume that $f = \mathsf{Det}(A \cdot \mathbf{x})$ for some $A \in \mathrm{GL}(m, \mathbb{F})$. The correctness of Algorithm 1 ensure that $\mathcal{U}_1 = \mathcal{F}_{\mathrm{col}}$ without loss of generality. Step 2 can be executed efficiently by checking if $U_iU_j \in \mathrm{span}_{\mathbb{F}}\{U_1, \dots, U_r\}$ for $i, j \in [r]$. Observation 2.5 implies that $\mathcal{A} \cong M_n$, i.e., $L_i = A^{-1} \cdot (I_n \otimes B_i) \cdot A$ for every $i \in [m]$, where $\{B_1, \dots, B_m\}$ is a basis of M_n. In Step 3, the FMAI oracle returns a $\mathbb{F}$-algebra isomorphism $\phi : \mathcal{A} \to M_n$ such that $\{C_i = \phi(L_i) : i \in [m]\}$ is a basis of M_n. The following claim ensures the existence of a matrix M, computed in Step 4. Its proof is given in Section E.1 of the Appendix of [11].

▷ Claim 19. There exists a $S \in \mathrm{GL}(n, \mathbb{F})$ such that $B_i = S^{-1} \cdot C_i \cdot S$ for every $i \in [m]$.

Consider the linear system defined by the equation $L_i \cdot M = M \cdot (I_n \otimes C_i)$, where the entries of M are taken as variables. Step 4 is executed by picking the free variables of the solution space of the system from a sufficiently large subset of $\mathbb{F}$. Finally, the correctness of Step 6 is argued in the proof of the following claim which is given in Section E.2 of the Appendix of [11].

▷ Claim 20. Suppose $f = \mathsf{Det}(A \cdot \mathbf{x})$, where $A \in \mathrm{GL}(m, \mathbb{F})$. Then, $f = \mathsf{Det}((I_n \otimes D) \cdot M^{-1} \cdot \mathbf{x})$ with high probability.

5 DET over finite fields and over $\mathbb{Q}$

The proofs of Theorem 1 and 2 are completed by replacing the FMAI oracle in Step 3 of Algorithm 2 by the following known algorithms for FMAI over finite fields and $\mathbb{Q}$.

▶ **Theorem 21** (Theorem 5.1 of [25]). *Let $\mathbb{F}$ be a finite field. Given a basis of a $\mathbb{F}$-algebra $\mathcal{A} \subseteq M_m$ such that $\mathcal{A} \cong M_n$, an isomorphism $\phi : \mathcal{A} \to M_n$ can be constructed in randomized* $(m, \log |\mathbb{F}|)$ *time.*

▶ **Theorem 22** (Theorem 1 of [14]). *There is a randomized algorithm with oracle access to* IntFact *that takes input a basis of a $\mathbb{Q}$-algebra $\mathcal{A} \subseteq M_m$ such that $\mathcal{A} \cong M_n$, and outputs an isomorphism $\phi : \mathcal{A} \to M_n$ with high probability. The algorithm runs in time polynomial in the bit length of the input, if n is bounded.*

▶ **Theorem 23** (Lemma 2.5 of [3]). *There is a randomized algorithm that takes input a basis of a $\mathbb{Q}$-algebra $\mathcal{A} \subseteq M_m$ such that $\mathcal{A} \cong M_n$, and outputs an isomorphism $\phi : \mathcal{A} \otimes_{\mathbb{Q}} Ł \to M_n(Ł)$ with high probability, where Ł is an extension field of $\mathbb{Q}$ satisfying $[Ł : \mathbb{Q}] \leq n$. The algorithm runs in time polynomial in the bit length of the input.*

6 Factoring hardness of DET over $\mathbb{Q}$

This section is devoted to proving Theorem 3. We show that DET in the 2×2 setting over $\mathbb{Q}$ is at least as hard as factoring square-free integers. We will need the following theorem.

▶ **Theorem 24** ([24]). *Assuming GRH, there is a randomized polynomial time reduction from the problem of factoring square-free integers to the following problem: Given non-zero $a, b \in \mathbb{Q}$, find rational numbers x, y, z (not all zero) such that $x^2 - ay^2 - bz^2 = 0$, if there exists such a solution.*

We will also need the following proposition, cited in [24], to prove the next theorem. We give a proof from [5] in Section F.1 of [11], for completeness.

▶ **Proposition 25.** *Let $a, b \in \mathbb{Q}^\times$. Then the equation $x^2 - ay^2 - bz^2 = 0$ has a non-zero rational solution if and only if the equation $x^2 - ay^2 - bz^2 + abw^2 = 0$ has a non-zero rational solution.*

We are now ready to prove integer factoring hardness of DET in the next theorem.

▶ **Theorem 26.** *Consider the polynomial $f_{a,b}(\mathbf{x}) = x_{1,1}^2 - ax_{1,2}^2 - bx_{2,1}^2 + abx_{2,2}^2$, where $a, b \in \mathbb{Q}$ are non-zero. Then $f_{a,b}(\mathbf{x}) = \mathsf{Det}_2(A \cdot \mathbf{x})$ for some $A \in \mathrm{GL}(4, \mathbb{Q})$ if and only if the equation $x^2 - ay^2 - bz^2 = 0$ has a non-zero rational solution (moreover, such a rational solution can be efficiently computed from A).*

Its proof is given in Section F.2 of [11]. Combining Theorems 24 and 26, we obtain Theorem 3.

▶ Remark 27. We want to explain how we got to the above reduction. Ronyai [24] proved that the FMAI problem over $\mathbb{Q}$ is factoring hard even for $n = 2$ via quaternion algebras. If one takes a specific quaternion algebra and tries to constructs a polynomial f whose Lie algebra is the traceless part of the quaternion algebra, then it turns out the polynomial $f_{a,b}(\mathbf{x})$ is the unique homogeneous degree 2 polynomial that comes out. But in any case, in hindsight, the polynomial $f_{a,b}(\mathbf{x})$ seems like a natural candidate to use.

7 Characterization of the determinant by its Lie algebra

In this section, we reduce FMAI to DET under mild restrictions on $\mathbb{F}$. We start with the claim that the Lie algebra of the determinant characterizes the determinant. This is well known over $\mathbb{C}$, but we give a proof in Section G.1 of [11] that works under mild restrictions on $\mathbb{F}$.

▶ **Lemma 28.** *Let $f \in \mathbb{F}[\mathbf{x}]$ be any homogeneous polynomial of degree n such that $\mathcal{L}_{col} \subseteq \mathfrak{g}_f$ (recall $\mathcal{L}_{col}$ from Section 2). Also suppose $\mathrm{char}(\mathbb{F}) \nmid n$. Then $f(\mathbf{x}) = \alpha \cdot \mathsf{Det}_n(\mathbf{x})$ for some $\alpha \in \mathbb{F}$.*

▶ Remark 29. Note that without the $\mathrm{char}(\mathbb{F}) \nmid n$ condition, Lemma 28 is not true. For example, $f(\mathbf{x}) = x_{1,1}^n + \mathsf{Det}_n(\mathbf{x})$ will have the same Lie algebra as $\mathsf{Det}_n(\mathbf{x})$ if $\mathrm{char}(\mathbb{F}) \mid n$.

▶ **Corollary 30.** *Let $f \in \mathbb{F}[\mathbf{x}]$ be a degree n homogeneous polynomial. Suppose that $A^{-1} \cdot \mathcal{L}_{col} \cdot A \subseteq \mathfrak{g}_f$ for some $A \in \mathrm{GL}(n^2, \mathbb{F})$ and $\mathrm{char}(\mathbb{F}) \nmid n$. Then $f(\mathbf{x}) = \alpha \cdot \mathsf{Det}_n(A \cdot \mathbf{x})$ for some $\alpha \in \mathbb{F}$.*

Proof. Consider $f'(\mathbf{x}) = f(A^{-1} \cdot \mathbf{x})$. By Fact 2, $\mathfrak{g}_{f'} = A \cdot \mathfrak{g}_f \cdot A^{-1}$, so $\mathcal{L}_{\mathrm{col}} \subseteq \mathfrak{g}_{f'}$. By Lemma 28, we get that $f'(\mathbf{x}) = \alpha \cdot \mathsf{Det}_n(\mathbf{x})$ for some $\alpha \in \mathbb{F}$ and hence $f(\mathbf{x}) = \alpha \cdot \mathsf{Det}_n(A \cdot \mathbf{x})$. ◀

Corollary 30 allows us to reduce FMAI to DET when n is constant (see Algorithm 3).

7.1 Proof of correctness of Algorithm 3 when $\mathrm{char}(\mathbb{F}) \nmid n$

The proof of correctness will follow from the following proposition, proved in Section G.2 of [11]. The matrices $B_{i,j}$ and $L_{i,j}$ are as defined in Step 2 of the algorithm.

▶ **Proposition 31.** *Suppose the algebra $\mathcal{A}$ spanned by $B_{1,1}, \ldots, B_{n,n}$ is isomorphic to M_n. Then there exist $K \in \mathrm{GL}(n^2, \mathbb{F})$ and $C_{1,1}, \ldots, C_{n,n} \in M_n$, s.t. $L_{i,j} = K^{-1}(I_n \otimes C_{i,j}) K$ for all $i, j \in [n]$.*

Now let us proceed to the proof of correctness of Algorithm 3. First of all, it is easy to ensure that whenever the algorithm outputs an isomorphism, it is actually an isomorphism. So what we need to prove is the converse. Suppose the algebra $\mathcal{A}$ is isomorphic to M_n. Then by Proposition 31, the space spanned by $\tilde{L}_1, \ldots, \tilde{L}_{n^2-1}$ is $K^{-1} \cdot \mathcal{L}_{\text{col}} \cdot K$. Then by Corollary 30, there is a unique solution to the equations in Step 4 given by $f(\mathbf{x}) = \alpha \cdot \mathsf{Det}_n(K \cdot \mathbf{x})$, for some $\alpha \in \mathbb{F}$, and so f is equivalent to the determinant. Hence, in Step 5, we will get an $A \in \mathrm{GL}(n^2, \mathbb{F})$ s.t. $f(\mathbf{x}) = \mathsf{Det}_n(A \cdot \mathbf{x})$. Since $\tilde{L}_1, \ldots, \tilde{L}_{n^2-1}$ span a Lie algebra of dimension n^2-1 and since they lie inside the Lie algebra of $\mathsf{Det}_n(A \cdot \mathbf{x})$, we must have that $\tilde{L}_1, \ldots, \tilde{L}_{n^2-1}$ span either $A^{-1} \cdot \mathcal{L}_{\text{col}} \cdot A$ or $A^{-1} \cdot \mathcal{L}_{\text{row}} \cdot A$. From this, we get that one of the following conditions should be true:

- There exist matrices $F_{1,1}, \ldots, F_{n,n} \in M_n$ such that $A \cdot L_{i,j} \cdot A^{-1} = I_n \otimes F_{i,j}$ for all $i, j \in [n]$.
- There exist matrices $F_{1,1}, \ldots, F_{n,n} \in M_n$ such that $A \cdot L_{i,j} \cdot A^{-1} = F_{i,j} \otimes I_n$ for all $i, j \in [n]$.

The implies that the algorithm will output 1 and an isomorphism into M_n. The complexity of the reduction is dominated by Step 4 which takes $n^{O(n)}$ field operations.

Algorithm 3 Reduction of FMAI to DET.

```
Input: Basis $\{B_1, \ldots, B_r\}$ of a $\mathbb{F}$-algebra $\mathcal{A} \subseteq M_m$, and access to an algorithm for DET.
Output: if $\mathcal{A} \cong M_n$ for some $n \in \mathbb{N}$, then output an isomorphism, 0 otherwise.

1: If $r = \dim_{\mathbb{F}} \mathcal{A} \neq n^2$ for any $n \in \mathbb{N}$, output 0 and halt.
2: Index the basis elements by $[n] \times [n]$, i.e., rename them as $B_{1,1}, \ldots, B_{n,n}$. Compute $n^2 \times n^2$ matrices $L_{1,1}, \ldots, L_{n,n}$ as follows: $L_{i,j}$ is the matrix corresponding to the left-multiplication action of $B_{i,j}$ on $B_{1,1}, \ldots, B_{n,n}$. That is $B_{i,j} \cdot B_{i_2,j_2} = \sum_{i_1,j_1} L_{i,j}((i_1,j_1),(i_2,j_2)) \cdot B_{i_1,j_1}$.
3: Compute a basis for the traceless parts of the matrices $L_{i,j}$. That is, compute a basis $\tilde{L}_1, \ldots, \tilde{L}_s$ of the space spanned by $L_{1,1} - \frac{\mathrm{tr}(L_{1,1})}{n^2} I_{n^2}, \ldots, L_{n,n} - \frac{\mathrm{tr}(L_{n,n})}{n^2} I_{n^2}$. If $s \neq n^2 - 1$, output 0 and halt.
4: Find a non-zero homogeneous polynomial of degree $n$, $f(\mathbf{x})$, satisfying the equations
       $$\sum_{i_1,j_1,i_2,j_2} M((i_1,j_1),(i_2,j_2)) \cdot x_{i_2,j_2} \cdot \frac{\partial f}{\partial x_{i_1,j_1}} = 0$$
   for every $M \in \{\tilde{L}_1, \ldots, \tilde{L}_{n^2-1}\}$ (these give linear equations in the coefficients of $f$). If no such non-zero polynomial exists then output 0 and halt.
5: Run DET on $f$. If the output is "Fail" then output 0 and halt. If $f(\mathbf{x}) = \mathsf{Det}_n(A \cdot \mathbf{x})$ then check if there exist matrices $F_{1,1}, \ldots, F_{n,n} \in M_n$ such $A \cdot L_{i,j} \cdot A^{-1} = I_n \otimes F_{i,j}$ for all $i, j$. If yes, output 1 and the isomorphism $\phi(B_{i,j}) = F_{i,j}$ (extended linearly to whole of $\mathcal{A}$). If no, check if there exist matrices $F_{1,1}, \ldots, F_{n,n} \in M_n$ such that $A \cdot L_{i,j} \cdot A^{-1} = F_{i,j} \otimes I_n$ for all $i, j$. If yes, output 1 and the isomorphism $\phi(B_{i,j}) = F_{i,j}$ (extended linearly to whole of $\mathcal{A}$). If no, output 0.
```

References

1 Manindra Agrawal and Nitin Saxena. Automorphisms of Finite Rings and Applications to Complexity of Problems. In *STACS 2005, 22nd Annual Symposium on Theoretical Aspects of Computer Science, Stuttgart, Germany, February 24-26, 2005, Proceedings*, pages 1–17, 2005.

2 Manindra Agrawal and Nitin Saxena. Equivalence of F-Algebras and Cubic Forms. In *STACS 2006, 23rd Annual Symposium on Theoretical Aspects of Computer Science, Marseille, France, February 23-25, 2006, Proceedings*, pages 115–126, 2006.

3 László Babai and Lajos Rónyai. Computing irreducible representations of finite groups. *Mathematics of Computation*, 55(192):705–722, 1990.

4 Alexander L. Chistov, Gábor Ivanyos, and Marek Karpinski. Polynomial Time Algorithms for Modules over Finite Dimensional Algebras. In *Proceedings of the 1997 International Symposium on Symbolic and Algebraic Computation, ISSAC '97, Maui, Hawaii, USA, July 21-23, 1997*, pages 68–74, 1997.

5 Keith Conrad. Quaternion algebras, 2016.

6 J. E. Cremona, T. A. Fisher, C. O'Neil, D. Simon, and M. Stoll. Explicit n-descent on elliptic curves III. Algorithms. *Math. Comput.*, 84(292):895–922, 2015. `arXiv:1107.3516`.

7 W.A. de Graaf. *Algorithms for Finite-Dimensional Lie Algebras.* PhD thesis, Technical University of Eindhoven, 1997.

8 W.A. de Graaf. Calculating the structure of a semisimple Lie algebra. *Journal of Pure and Applied Algebra*, 117-118:319–329, 1997.

9 Wayne Eberly. Decompositions of algebras over R and C. *computational complexity*, 1(3):211–234, September 1991.

10 W.M. Eberly. *Computations for algebras and group representations.* PhD thesis, Department of Computer Science, University of Toronto, 1989.

11 Ankit Garg, Nikhil Gupta, Neeraj Kayal, and Chandan Saha. Determinant equivalence test over finite fields and over $\mathbf{Q}$. *Electronic Colloquium on Computational Complexity (ECCC)*, 26:42, 2019. URL: `https://eccc.weizmann.ac.il/report/2019/042`.

12 Joshua A. Grochow. Matrix Lie algebra isomorphism. In *IEEE Conference on Computational Complexity (CCC12)*, pages 203–213, 2012.

13 Joshua Abraham Grochow. *Symmetry and equivalence relations in classical and geometric complexity theory.* PhD thesis, Department of Computer Science, The University of Chicago, Chicago, Illinois, 2012.

14 Gábor Ivanyos, Lajos Rónyai, and Josef Schicho. Splitting full matrix algebras over algebraic number fields. *Journal of Algebra*, 354:211–223, 2012. `arXiv:1106.6191`.

15 Neeraj Kayal. Efficient algorithms for some special cases of the polynomial equivalence problem. In *Proceedings of the Twenty-Second Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2011, San Francisco, California, USA, January 23-25, 2011*, pages 1409–1421, 2011.

16 Neeraj Kayal. Affine projections of polynomials: extended abstract. In *Proceedings of the 44th Symposium on Theory of Computing Conference, STOC 2012, New York, NY, USA, May 19 - 22, 2012*, pages 643–662, 2012.

17 Neeraj Kayal, Vineet Nair, and Chandan Saha. Average-case linear matrix factorization and reconstruction of low width Algebraic Branching Programs. *Electronic Colloquium on Computational Complexity (ECCC)*, 25:29, 2018.

18 Neeraj Kayal, Vineet Nair, Chandan Saha, and Sébastien Tavenas. Reconstruction of Full Rank Algebraic Branching Programs. In *32nd Computational Complexity Conference, CCC 2017, July 6-9, 2017, Riga, Latvia*, pages 21:1–21:61, 2017.

19 Falko Lorenz. *Algebra Volumne 2: Fields with structures, Algebras and advanced topics.* Springer, 2008.

20 Meena Mahajan and V. Vinay. A Combinatorial Algorithm for the Determinant. In *Proceedings of the Eighth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), 5-7 January 1997, New Orleans, Louisiana, USA.*, pages 730–738, 1997.

21 Meena Mahajan and V. Vinay. Determinant: Combinatorics, Algorithms, and Complexity. *Chicago J. Theor. Comput. Sci.*, 1997, 1997.

22 Ketan Mulmuley and Milind A. Sohoni. Geometric Complexity Theory I: An Approach to the P vs. NP and Related Problems. *SIAM J. Comput.*, 31(2):496–526, 2001.

23 Jacques Patarin. Hidden Fields Equations (HFE) and Isomorphisms of Polynomials (IP): Two New Families of Asymmetric Algorithms. In *Advances in Cryptology - EUROCRYPT '96, International Conference on the Theory and Application of Cryptographic Techniques, Saragossa, Spain, May 12-16, 1996, Proceeding*, pages 33–48, 1996.

24 Lajos Rónyai. Simple Algebras Are Difficult. In *Proceedings of the 19th Annual ACM Symposium on Theory of Computing (STOC), 1987, New York, New York, USA*, pages 398–408, 1987.

25 Lajos Rónyai. Computing the Structure of Finite Algebras. *J. Symb. Comput.*, 9(3):355–373, 1990.

26 Lajos Rónyai. A Deterministic Method for Computing Splitting Elements in Simple Algebras over Q. *Journal of Algorithms*, 16:24–32, 1994.

27 Alexander J.E. Ryba. Computer construction of split Cartan subalgebras. *J. Algebra*, 309:455–483, 2007.

28 Nitin Saxena. *Morphisms of rings and applications to complexity.* PhD thesis, Indian Institute of Technology Kanpur, 2006.

29 Thomas Thierauf. The Isomorphism Problem for Read-Once Branching Programs and Arithmetic Circuits. *Chicago J. Theor. Comput. Sci.*, 1998, 1998.

30 Leslie G. Valiant. Completeness Classes in Algebra. In *Proceedings of the 11h Annual ACM Symposium on Theory of Computing, April 30 - May 2, 1979, Atlanta, Georgia, USA*, pages 249–261, 1979.

31 Lars Ambrosius Wallenborn. Computing the Hilbert symbol, quadratic form equivalence and integer factoring. Diploma thesis, University of Bonn, 2013.