Electronic Colloquium on Computational Complexity, Report No. 91 (2020)

# Randomized polynomial-time equivalence between determinant and trace-IMM equivalence tests

Janaky Murthy
Indian Institute of Science
janakymurthy@iisc.ac.in

Vineet Nair
Technion Israel Institute of Technology *
vineet@cs.technion.ac.il

Chandan Saha
Indian Institute of Science
chandan@iisc.ac.in

June 14, 2020

**Abstract**

Equivalence testing for a polynomial family $\{g_m\}_{m\in\mathbb{N}}$ over a field $\mathbb{F}$ is the following problem: Given black-box access to an $n$-variate polynomial $f(\mathbf{x})$, where $n$ is the number of variables in $g_m$ for some $m \in \mathbb{N}$, check if there exists an $A \in \mathrm{GL}(n,\mathbb{F})$ such that $f(\mathbf{x}) = g_m(A\mathbf{x})$. If yes, then output such an $A$. The complexity of equivalence testing has been studied for a number of important polynomial families, including the determinant (Det) and the family of iterated matrix multiplication polynomials. Two popular variants of the iterated matrix multiplication polynomial are: $\mathrm{IMM}_{w,d}$ (the $(1,1)$ entry of the product of $d$ many $w \times w$ symbolic matrices) and $\text{Tr-IMM}_{w,d}$ (the trace of the product of $d$ many $w \times w$ symbolic matrices). The families – Det, IMM and Tr-IMM – are VBP-complete under $p$-projections, and so, in this sense, they have the same complexity. But, do they have the same equivalence testing complexity? We show that the answer is "yes" for Det and Tr-IMM (modulo the use of randomness).

The above result may appear a bit surprising as the complexity of equivalence testing for IMM and that for Det are quite different over $\mathbb{Q}$: a randomized polynomial-time equivalence testing for IMM over $\mathbb{Q}$ is known [KNST19], whereas [GGKS19] showed that equivalence testing for Det over $\mathbb{Q}$ is integer factoring hard (under randomized reductions and assuming GRH). To our knowledge, the complexity of equivalence testing for Tr-IMM was not known before this work. We show that, despite the syntactic similarity between IMM and Tr-IMM, equivalence testing for Tr-IMM and that for Det are randomized polynomial-time Turing reducible to each other over any field of characteristic zero or sufficiently large. The result is obtained by connecting the two problems via another well-studied problem in computer algebra, namely the *full matrix algebra isomorphism* problem (FMAI). In particular, we prove the following:

1. Testing equivalence of polynomials to $\text{Tr-IMM}_{w,d}$, for $d \geq 3$ and $w \geq 2$, is randomized polynomial-time Turing reducible to testing equivalence of polynomials to $\mathrm{Det}_w$, the determinant of the $w \times w$ matrix of formal variables. (Here, $d$ need not be a constant.)
2. FMAI is randomized polynomial-time Turing reducible to equivalence testing (in fact, to tensor isomorphism testing) for the family of *matrix multiplication tensors* $\{\text{Tr-IMM}_{w,3}\}_{w\in\mathbb{N}}$.

These results, in conjunction with the randomized poly-time reduction (shown in [GGKS19]) from determinant equivalence testing to FMAI, imply that the four problems – FMAI, equivalence testing for Tr-IMM and for Det, and the 3-tensor isomorphism problem for the family of matrix multiplication tensors – are randomized poly-time equivalent under Turing reductions.

*A part of this work was done when the author was a graduate student at the Indian Institute of Science.

ISSN 1433-8092

# 1 Introduction

The *polynomial equivalence problem* or *equivalence testing* is the following algorithmic task: Given two $n$-variate polynomials $f$ and $g$ over a field $\mathbb{F}$ as lists of coefficients, determine if there exists an $A \in \mathrm{GL}(n, \mathbb{F})$ such that $f(\mathbf{x}) = g(A\mathbf{x})$. If yes, then $f$ is said to be *equivalent to*[1] $g$ over $\mathbb{F}$. The complexity of equivalence testing depends on the underlying field $\mathbb{F}$. Over finite fields, the problem is in $\mathsf{NP} \cap \mathsf{coAM}$ [Thi98, Sax06][2], and hence unlikely to be NP-complete. Whereas over $\mathbb{Q}$, it is not even known whether equivalence testing is decidable. The best known complexity of the problem over other fields follows from a naive reduction to solving a system of polynomial equations. However, polynomial solvability could be harder than testing polynomial equivalence.

**Connections to other problems.** A few works in the literature have related equivalence testing to other fundamental problems. For example, [AS05] showed that the special instance of cubic form equivalence is at least as hard as (but possibly harder than) graph isomorphism, irrespective of the underlying field. There is a close connection between cubic form equivalence and the algebra isomorphism problem. [AS06] gave a polynomial-time reduction from commutative algebra isomorphism to cubic form equivalence over any field. In the reverse direction, a polynomial-time reduction is known from cubic form equivalence to commutative algebra isomorphism over *almost* all fields [GQ19, AS05]. In fact, the results in [BW15], [FGS19] and [GQ19] together imply that a host of problems, which includes 3-tensor isomorphism, matrix space isometry, matrix space conjugacy, (commutative or associative) algebra isomorphism and cubic form equivalence, are polynomial-time reducible to each other. There is a cryptographic authentication scheme [Pat96] based on the presumed hardness of cubic form equivalence[3] over finite fields (or rather a generalization of it known as *Isomorphism of Polynomials with one Secret* (IP1S)[4]). It is not known whether cubic form equivalence is even decidable over $\mathbb{Q}$. In contrast, the complexity of quadratic form equivalence testing is completely resolved, primarily due to well-known classification results for quadratic forms (see [Ser73, Ara11]). The classification yields a polynomial-time quadratic form equivalence testing over finite fields. Over $\mathbb{Q}$ though, quadratic form equivalence can be solved in polynomial time only with oracle access to integer factoring. Moreover, integer factoring reduces in randomized polynomial time to quadratic form equivalence over $\mathbb{Q}$ [Wal13][5].

**Special polynomial families.** The work of [Kay11] initiated the study of a natural variant of the polynomial equivalence problem, namely equivalence testing for special families of polynomials. In this setting, we fix some important family of polynomials $\mathcal{G} = \{g_m\}_{m \in N}$ and then aim to design an equivalence testing algorithm for $\mathcal{G}$. Such an algorithm takes input black-box access[6] to a single $n$-variate polynomial $f(\mathbf{x})$ and determines whether $f$ is equivalent to $g_m$ for some $m \in \mathbb{N}$, and if

[1] Indeed, $f$ and $g$ represent the same function on $\mathbb{F}^n$ upto a change of basis.

[2] This is shown by using the classic set lower bound protocol [GS86].

[3] more generally, constant-degree form equivalence

[4] IP1S is the following problem: Given two ordered sets of $n$-variate polynomials $(f_1, f_2, \ldots, f_m)$ and $(g_1, g_2, \ldots, g_m)$, decide if there exists an $A \in \mathrm{GL}(n, \mathbb{F})$ such that $f_i(\mathbf{x}) = g_i(A\mathbf{x})$ for all $i \in [m]$. Note that even the quadratic case is non-trivial here as we are dealing with tuples of polynomials. Recently, [IQ19] gave a randomized poly-time algorithm for the quadratic IP1S problem over finite fields of odd size. In the general setting, there is an algorithm for IP1S over finite fields that is significantly better than the brute-force strategy, but it still runs in exponential time [FP06, PGC98].

[5] This reduction is to the search version of the quadratic form equivalence problem. In the search version of equivalence testing, we are required to output an invertible transformation $A$ if the input polynomials are equivalent.

[6] i.e., query access to evaluations of $f$ at chosen points from $\mathbb{F}^n$.

yes, then it also outputs an $A \in \mathrm{GL}(n, \mathbb{F})$ such that $f(\mathbf{x}) = g_m(A\mathbf{x})$.[7] [Kay12, Kay11] gave randomized polynomial-time equivalence testing algorithms for a few interesting polynomial families, viz. the determinant, the permanent, the family of elementary symmetric polynomials and the family of power symmetric polynomials. These families are quite popular in algebraic complexity theory, particularly in the context of proving arithmetic circuit lower bounds (see the surveys [SY10, CKW11, Sap15]). Except for the determinant, the algorithms in [Kay12, Kay11] work over $\mathbb{C}, \mathbb{Q}$, and finite fields[8], and for the determinant it works only over $\mathbb{C}$. Recently, [GGKS19] gave a randomized polynomial-time equivalence testing algorithm for the determinant over finite fields[9]. They also showed that determinant equivalence test over $\mathbb{Q}$ is intimately connected to integer factoring: Let $\mathrm{Det}_w(\mathbf{x})$ be the determinant of the $w \times w$ symbolic matrix. Then, deciding if a given polynomial is equivalent to $\mathrm{Det}_w$ over $\mathbb{Q}$ can be done in randomized polynomial-time with oracle access to integer factoring, provided $w$ is a constant[10]. Furthermore, assuming GRH, there is a randomized polynomial-time reduction from factoring square-free integers to finding an $A \in \mathrm{GL}(2, \mathbb{Q})$ such that a given quadratic form $f = \mathrm{Det}_2(A \cdot \mathbf{x})$, if $f$ is equivalent to $\mathrm{Det}_2$.

Determinant equivalence test is particularly interesting in the context of the permanent versus determinant problem [Val79]. An approach to solve this long-standing open problem is given by Geometric Complexity Theory (GCT) [MS01, MS08], which proposes the applications of deep tools and techniques from algebraic geometry, group theory and representation theory to achieve this goal. GCT reduces the problem to showing that the (padded) permanent polynomial is not in the *orbit closure*[11] of a polynomial-size determinant polynomial, and suggests (among other things) to develop an algorithmic approach to do the same. Equivalence testing for the determinant is the related problem of checking if a given polynomial is in the orbit of the determinant polynomial.

The determinant $\mathrm{Det} := \{\mathrm{Det}_w\}_{w \in \mathbb{N}}$ is complete (under $p$-projections) for the class VBP [12] [MV97]. Likewise, the family of *iterated matrix multiplication* polynomials is also complete for the class VBP, and has been used quite a bit in proving arithmetic circuit lower bounds. In this sense, the two families have the same complexity[13]. But, do they have similar equivalence testing complexity? Our work here, in conjunction with [GGKS19] and [KNST19], gives an answer to this question.

[7] The problem is well-posed even if $f$ is given verbosely as a list of coefficients and it is not required to output an invertible transformation $A$ in the 'yes' case. However, it turns out that for a number of popular polynomial families it is indeed possible to design efficient equivalence testing algorithms that satisfy these stronger requirements.

[8] Over $\mathbb{C}$, the computation model assumes that arithmetic with numbers in $\mathbb{C}$ and root finding of univariate polynomials over $\mathbb{C}$ can be done efficiently. Also, the finite fields are assumed to be of sufficiently large characteristic.

[9] A determinant equivalence test over finite fields was also given in [KNS19], but the algorithm there outputs an invertible transformation over a low extension of the base field.

[10] When $w$ is not a constant, [GGKS19] gave a randomized polynomial-time determinant equivalence test over $\mathbb{Q}$, but the algorithm (which works without an integer factoring oracle) outputs a transformation over a low extension of $\mathbb{Q}$.

[11] The orbit of an $n$-variate degree-$d$ polynomial $g \in \mathbb{C}[\mathbf{x}]$ is the set $\{g(A\mathbf{x}) \mid A \in \mathrm{GL}(n, \mathbb{C})\}$, and the orbit closure of $g$ is the Zariski closure of the orbit when viewed as points in $\mathbb{C}^{\binom{n+d}{d}}$.

[12] Class VBP consists of polynomial families that are computable by polynomial-size algebraic branching programs (ABP). ABP is a powerful model for computing polynomials that subsumes arithmetic formulas.

[13] Consider a class $\mathcal{C}$ of arithmetic circuits that is closed under affine projections, e.g., the class of depth three circuits. A super-polynomial lower bound for circuits in $\mathcal{C}$ computing the determinant implies a super-polynomial lower bound for circuits in $\mathcal{C}$ computing the iterated matrix multiplication polynomial (IMM) and vice versa. Thus, Det and IMM have the same complexity, and one may study the "permanent versus IMM" problem in the same vein as the permanent versus determinant problem. On the other hand, if $\mathcal{C}$ is not closed under affine projections, then there are classes (like multilinear formulas) for which a super-polynomial lower bound is known for determinant [Raz09] but not for IMM.

**Iterated matrix multiplication.** Two natural versions of the iterated matrix multiplication polynomial are: a) $\mathrm{IMM}_{w,d}$ that is defined as the $(1,1)$ entry of the product of $d$ many $w \times w$ symbolic matrices (i.e., matrices whose entries are distinct variables), and b) $\text{Tr-IMM}_{w,d}$ that is defined as the trace of the product of $d$ many $w \times w$ symbolic matrices. The $\mathrm{IMM} := \{\mathrm{IMM}_{w,d}\}_{w,d\in\mathbb{N}}$ family has been studied more from the lower bound perspective [NW97, FLMS15, KS17, KNS20, KS15, KST18, CLS19] because it naturally captures the algebraic branching program model (see Section A). On the other hand, $\text{Tr-IMM} := \{\text{Tr-IMM}_{w,d}\}_{w,d\in\mathbb{N}}$ has been studied in [Gro12, Lan15, Ges16, GIP17][14] owing to its nice structural properties (pertaining to its group of symmetries and the associated Lie algebra) that may be quite useful for studying GCT methods when applied to the "Permanent versus Tr-IMM" problem. IMM and Tr-IMM are also complete for the class VBP. Interestingly, the three polynomials – $\mathrm{Det}_w$, $\mathrm{IMM}_{w,d}$ and $\text{Tr-IMM}_{w,d}$ – are characterized by their respective groups of symmetries [Fro97, KNST19, Ges16].

**Equivalence testing for iterated matrix multiplication.** How does equivalence testing for IMM and Tr-IMM relate to that of Det? In [KNST19], a randomized polynomial-time equivalence testing algorithm was given for IMM over $\mathbb{C}, \mathbb{Q}$ and finite fields. Comparing this with the above-mentioned results on determinant equivalence test [Kay12, GGKS19], we see that the complexity of equivalence tests for Det and IMM are quite different over $\mathbb{Q}$ (unless integer factoring is easy). Is this also the case between Det and Tr-IMM? One may be tempted to say 'yes' owing to the closeness of the definitions of IMM and Tr-IMM. However, contrary to this first impression, we show that equivalence testing for Det and that for Tr-IMM are randomized polynomial-time Turing reducible to each other over $\mathbb{C}, \mathbb{Q}$ and finite fields[15] (see Corollary 1.1). Thus, viewed along this line, Det and Tr-IMM are closer to each other than to IMM.[16] For brevity, we would henceforth denote the equivalence testing problems for Det and Tr-IMM by DET and TRACE respectively.

**Connections to algebra isomorphism and 3-tensor isomorphism.** As mentioned before, cubic form equivalence, algebra isomorphism and 3-tensor isomorphism are polynomial-time equivalent. Moreover, degree-$d$ form equivalence reduces to cubic form equivalence [AS05, AS06] and $d$-tensor isomorphism reduces to 3-tensor isomorphism [GQ19] in polynomial-time, if $d$ is bounded. Det and Tr-IMM being two important polynomial families, we wonder if DET and TRACE can be linked with any natural case of algebra isomorphism. Further, do DET and TRACE reduce to any special case of cubic form equivalence or 3-tensor isomorphism? We show that the answers to these are 'yes'. The relevant problems are the *full-matrix algebra isomorphism* (FMAI) problem and the 3-tensor isomorphism problem for the family of *matrix multiplication tensors* (MMTI).

FMAI is a well-studied problem in computer algebra which is defined as follows: Given a basis

[14] Actually, [GIP17] studied a related polynomial $\text{Tr-Pow}_{w,d}$, which is the trace of the $d$-th power of a $w \times w$ symbolic matrix. They showed that a particular line of attack prescribed by GCT, namely *orbit occurrence obstructions*, cannot prove super-linear lower bound on the "Tr-Pow complexity" of the permanent. We are not aware of a similar result (or, more generally, a result that rules out the *occurrence obstructions* approach as in [BIP16, IP16]) with Tr-Pow (or Det) replaced by Tr-IMM.

[15] The reduction works over any field $\mathbb{F}$ of characteristic zero or sufficiently large. We also require that univariate polynomial factoring over $\mathbb{F}$ can be done efficiently.

[16] Talking of the difference between the 'trace model' and the '(1,1) model', a recent work [BIM+20] showed that in the non-commutative setting, the border width complexity and the width complexity of a polynomial are *not* always equal for the trace-ABP model, unlike the case for the classical $(1,1)$-ABP model [Nis91].

of a matrix algebra $\mathcal{A} \subseteq \mathcal{M}_m(\mathbb{F})$, check if $\mathcal{A}$ is isomorphic[17] to $\mathcal{M}_w(\mathbb{F})$, where $\mathcal{M}_m(\mathbb{F})$ is the algebra of $m \times m$ matrices over $\mathbb{F}$ and $\dim_{\mathbb{F}}(\mathcal{A}) = w^2$; if yes, then output an isomorphism from $\mathcal{A}$ to $\mathcal{M}_w(\mathbb{F})$. A randomized polynomial-time algorithm to solve FMAI over finite fields was given in [Rón87, Rón90], whereas over $\mathbb{Q}$ a randomized Turing reduction from FMAI to integer factoring was shown in [IRS12, CFO+15]. The reduction is polynomial-time if $\dim_{\mathbb{Q}}(\mathcal{A})$ is bounded. Also, [BR90, Ebe89] gave a randomized polynomial-time algorithm that outputs an isomorphism from $\mathcal{A} \otimes_{\mathbb{Q}} \mathbb{L}$ to $\mathcal{M}_w(\mathbb{L})$, where $\mathbb{L}$ is a degree $w$ extension field of $\mathbb{Q}$, if $\mathcal{A}$ is isomorphic to $\mathcal{M}_w(\mathbb{Q})$. The decision version of FMAI over $\mathbb{Q}$ is in NP $\cap$ coNP [Rón92]. The results for DET in [GGKS19] were obtained by giving a randomized poly-time Turing reduction from DET to FMAI. In this work, we give a randomized polynomial-time Turing reduction from TRACE to DET (Theorem 1).

A $d$-tensor is a degree-$d$ form (i.e., a degree-$d$ homogeneous polynomial) $f(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_d)$ whose every monomial has exactly one variable from each of the sets $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_d$. The $d$-tensor isomorphism problem is the following: Given two $d$-tensors $f(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_d)$ and $g(\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_d)$ decide if there exist $A_1 \in \mathrm{GL}(|\mathbf{x}_1|, \mathbb{F}), \ldots, A_d \in \mathrm{GL}(|\mathbf{x}_d|, \mathbb{F})$ such that $f = g(A_1\mathbf{x}_1, A_2\mathbf{x}_2, \ldots, A_d\mathbf{x}_d)$. The $d$-tensor isomorphism problem for a family of $d$-tensors is defined accordingly, just like equivalence testing for a family of polynomials. MMTI is the 3-tensor isomorphism problem for the family of matrix multiplication tensors $\{\text{Tr-IMM}_{w,3}\}_{w \in \mathbb{N}}$. The matrix multiplication tensor $\text{Tr-IMM}_{w,3}$ is a crucial object in the study of asymptotically fast algorithms for multiplying two $w \times w$ matrices. In this paper, we give a randomized polynomial-time Turing reduction from FMAI to MMTI (Theorem 2). Further, it follows easily from the symmetries of $\text{Tr-IMM}_{w,d}$ ( [Ges16], see Lemma 3.4) that MMTI reduces in polynomial-time to TRACE.

Thus, the above results together with the reduction in [GGKS19] show that the four problems – TRACE, DET, FMAI and MMTI – are randomized polynomial-time Turing reducible to each other. Although, the equivalence between MMTI and FMAI has the same essence as the equivalence between 3-tensor isomorphism (or cubic form equivalence) and algebra isomorphism, our proofs are quite different from the proofs in [GQ19, FGS19, AS05, AS06][18]. In particular, we do not see any easy adaptation of the arguments in [GQ19, FGS19, AS05, AS06] leading to the results mentioned above. Our proofs link MMTI with FMAI, via TRACE and DET, by exploiting the structure of the Lie algebra of $\text{Tr-IMM}_{w,d}$ (which is in the same spirit as the reduction from DET to FMAI in [GGKS19] using the Lie algebra of $\text{Det}_w$). Also, the reduction from $d$-tensor isomorphism (similarly, degree-$d$ form equivalence) to 3-tensor isomorphism (respectively, cubic form equivalence) in [GQ19, AS05, AS06] is efficient only if $d$ is a constant. Whereas, our randomized reduction from testing equivalence to $\text{Tr-IMM}_{w,d}$ to MMTI runs in time $\text{poly}(w, d)$.

### 1.1 The results (stated formally)

The polynomial $\text{Tr-IMM}_{w,d} := \text{tr}(Q_0 \cdot Q_1 \ldots Q_{d-1})$, where $Q_k$ is a $w \times w$ symbolic matrix in $\mathbf{x}_k$ variables. Throughout, we will assume that $w \geq 2$, $d \geq 3$ and $\text{char}(\mathbb{F}) = 0$ or $> (w^2 d)^5$, and univariate polynomial factoring over $\mathbb{F}$ can be done in probabilistic polynomial time. The restriction on the characteristic of $\mathbb{F}$ has not been optimized in this paper.

[17] i.e., isomorphic as algebras over $\mathbb{F}$.

[18] The reductions in these prior works are deterministic and hold for the decision versions of the problems, whereas the reductions here are randomized and for the search versions of the problems.

**Theorem 1** (TRACE to DET)**.** *There is a randomized algorithm that takes as input black-box access to an $n$-variate degree-$d$ polynomial $f$ and oracle access to* DET *over $\mathbb{F}$, and does the following with high probability: If there is a $w \in \mathbb{N}$ such that $f$ is equivalent to Tr-IMM$_{w,d}$, then it outputs an $A \in GL(n, \mathbb{F})$ such that $f =$ Tr-IMM$_{w,d}(A\mathbf{x})$; otherwise it outputs 'No such $w$ exists'. The algorithm runs in* $\text{poly}(n, \beta)$ *time, where $\beta$ is the bit length of the coefficients of $f$.*

The reduction is given in Section 4. Theorem 1 implies a randomized poly-time algorithm for TRACE over $\mathbb{C}$ and finite fields, and also over $\mathbb{Q}$ (provided the algorithm has access to integer factoring oracle and $w$ is bounded) via known results on DET [Kay12, GGKS19]. Two other remarks:

1. *No knowledge of $w$*: The algorithm requires no knowledge of $w$, if the input polynomial $f$ is equivalent to Tr-IMM$_{w,d}$ for some $w \in \mathbb{N}$ then the algorithm finds such a $w$.

2. *Reduction to* TRACE-TI: The *tensor isomorphism* problem for Tr-IMM (denoted TRACE-TI) is as follows: Given blackbox access to a $d$-tensor $g(\mathbf{x}_0, \ldots, \mathbf{x}_{d-1})$, check if there are $B_0, \ldots, B_{d-1} \in \text{GL}(w^2, \mathbb{F})$ such that $g =$ Tr-IMM$_{w,d}(B_0\mathbf{x}_0, \ldots, B_{d-1}\mathbf{x}_{d-1})$, and if yes then output such $B_0, \ldots, B_{d-1}$. The algorithm in Theorem 1 first reduces TRACE to TRACE-TI (finding $w$ in this step), and then solves TRACE-TI using DET oracle over $\mathbb{F}$. The reduction from TRACE to TRACE-TI (which resembles a similar reduction used in the equivalence test for IMM [KNST19]) does not require oracle access to DET. A randomized polynomial-time algorithm for TRACE-TI *over* $\mathbb{C}$ was given in [Gro12], but the algorithm there does not reduce TRACE-TI to DET.

**Theorem 2** (FMAI to MMTI)**.** *There is a randomized algorithm that takes as input a basis of an algebra $\mathcal{A} \subseteq \mathcal{M}_m(\mathbb{F})$, and oracle access to* MMTI*, and does the following with high probability: If $\mathcal{A} \cong \mathcal{M}_w(\mathbb{F})$, where $w^2 = \dim_{\mathbb{F}}(\mathcal{A})$, then it outputs 'Yes'; otherwise it outputs 'No such $w \in \mathbb{N}$ exists'. If the algorithm outputs 'Yes', then it also outputs an algebra isomorphism from $\mathcal{A}$ to $\mathcal{M}_w(\mathbb{F})$. The algorithm runs in* $\text{poly}(m, \beta)$ *time, where $\beta$ is the bit length of the entries of the input basis matrices.*

The algorithm is given in Section 5.2. It uses a characterization of Tr-IMM$_{w,d}$ by the Lie algebra $\mathfrak{g}_{\text{Tr-IMM}}$ of its group of symmetries (Lemma 5.1) along with a nice choice of basis of $\mathfrak{g}_{\text{Tr-IMM}}$ (Section 3) to reduce FMAI to degree four TRACE-TI in *deterministic* polynomial time, which in turn reduces to MMTI in randomized polynomial time (Theorem 3). Two more remarks on Theorem 2:

1. MMTI *to* TRACE: Using oracle access to TRACE, it is easy to solve MMTI (in fact TRACE-TI) in polynomial time: Since a polynomial identity test at the end of a TRACE-TI algorithm ensures that the output of the algorithm is correct, it suffices to prove that if the input to a TRACE algorithm is a $d$-tensor $f$ that is isomorphic to Tr-IMM$_{w,d}$, then the algorithm outputs $d$ matrices $B_0, \ldots, B_{d-1}$ such that $f(\mathbf{x}) =$ Tr-IMM$_{w,d}(B_0\mathbf{x}_0, \ldots, B_{d-1}\mathbf{x}_{d-1})$. This is true as any algorithm for TRACE outputs a block-diagonal matrix $B$ such that $f(\mathbf{x}) =$ Tr-IMM$_{w,d}(B\mathbf{x})$ (from Lemma 3.4). Matrices $B_0, \ldots, B_{d-1}$ can be easily derived from $B$.

2. *A reduction from* FMAI *to* DET: A Turing reduction from FMAI to DET over $\mathbb{F}$ was given in [GGKS19] that runs in exponential time. We improve this run-time significantly: Theorems 1 and 2 imply that FMAI is in fact randomized polynomial-time Turing reducible to DET.

**Corollary 1.1.** *It follows from Theorems 1 and 2, and the randomized polynomial-time Turing reduction from* DET *to* FMAI *in [GGKS19], that the four problems –* TRACE, DET, FMAI *and* MMTI *– are randomized polynomial-time equivalent under Turing reductions (see Figure 1 below).*

As mentioned before, the next theorem (proved in Appendix E) is used in the proof of Theorem 2.

**Theorem 3** (TRACE-TI to MMTI)**.** *There is a randomized algorithm that takes as input black-box access to an $n$-variate $d$-tensor $f(\mathbf{x}_0, \ldots, \mathbf{x}_{d-1})$, and oracle access to* MMTI*, and does the following with high probability: If $f$ is isomorphic to Tr-IMM$_{w,d}$, then it outputs $B_0, B_1, \ldots, B_{d-1} \in GL(w^2, \mathbb{F})$ such that $f =$ Tr-IMM$_{w,d}(B_0\mathbf{x}_0, \ldots, B_{d-1}\mathbf{x}_{d-1})$; otherwise it outputs 'No'. The algorithm runs in* poly$(n, \beta)$ *time, where $\beta$ is the bit length of the coefficients of $f$.*

The figure below is a depiction of Corollary 1.1. An arrow from Problem A to B indicates a randomized polynomial-time Turing reduction from A to B.

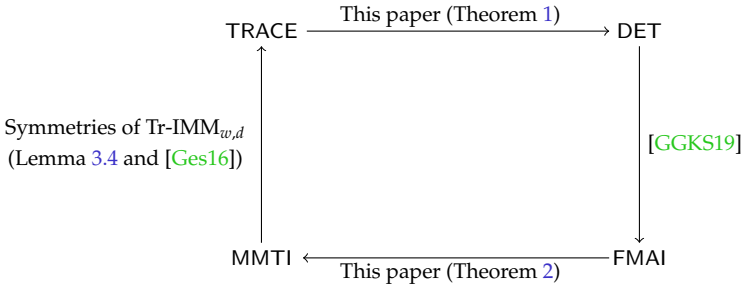


Figure 1: Reductions between TRACE, DET, FMAI, and MMTI

## 2 Notations and definitions

Recall that Tr-IMM$_{w,d} := \mathrm{tr}(Q_0 \cdot Q_1 \ldots Q_{d-1})$, where $Q_k = (x_{ij}^{(k)})_{i,j\in[w]}$. Let $\mathbf{x}_k = \{x_{ij}^{(k)}\}_{i,j\in[w]}$, $\mathbf{x} = \uplus_{k\in[0,d-1]}\mathbf{x}_k$, and $n = w^2d$. At times, we will refer to the $\mathbf{x}$ variables as $x_1, \ldots, x_n$. The $\mathbf{x}$ variables are ordered as $\mathbf{x}_0 > \mathbf{x}_1 > \ldots > \mathbf{x}_{d-1}$, and within a variable set $\mathbf{x}_k$, if $k$ is even (similarly, odd) then the variables are ordered in row-major (respectively, column-major) fashion. The rows and columns of a matrix in $\mathcal{M}_n = \mathcal{M}_n(\mathbb{F})$, and the entries of a column vector in $\mathbb{F}^n$ are indexed by $\mathbf{x}$ variables ordered as above. A matrix in $\mathcal{M}_n$ is called *block-diagonal* if the row and column of every non-zero entry of the matrix is indexed by variables from the same variable set. A few more basic definitions and terminologies about matrices, matrix products and ABP are given in Appendix A. The indices $k, \ell \in [0, d-1]$ will be treated as elements in $\mathbb{Z}/d\mathbb{Z}$, i.e., $k+1 = 0$ if $k = d-1$. Let $\mathcal{L} \subseteq \mathcal{M}_n$. A subspace $\mathcal{U} \subseteq \mathbb{F}^n$ is *$\mathcal{L}$-invariant* if for all $M \in \mathcal{L}$, $M \cdot \mathcal{U} \subseteq \mathcal{U}$.

**Definition 2.1** (Irreducible invariant subspace)**.** An $\mathcal{L}$-invariant subspace $\mathcal{U} \subseteq \mathbb{F}^n$ is irreducible if there are no proper $\mathcal{L}$-invariant subspaces $\mathcal{U}_1$ and $\mathcal{U}_2$ of $\mathcal{U}$ such that $\mathcal{U} = \mathcal{U}_1 \oplus \mathcal{U}_2$.

**Definition 2.2** (Closure of a vector)**.** The closure of a vector $\mathbf{v} \in \mathbb{F}^n$ under the action of $\mathcal{L} \subseteq \mathcal{M}_n$ is the smallest $\mathcal{L}$-invariant subspace of $\mathbb{F}^n$ containing $\mathbf{v}$.

An algorithm to compute the closure of a vector in polynomial-time is given in [KNST19]. An easy-to-work-with definition of the Lie algebra of the group of symmetries of a polynomial was given in [Kay12]. For brevity, we will call it the Lie algebra of a polynomial.[19]

[19]Geometrically speaking, the Lie algebra of an $n$-variate polynomial $f(\mathbf{x})$ is the subspace of $\mathcal{M}_n(\mathbb{F})$ obtained by translating the tangent of the algebraic set $\{A \in \mathcal{M}_n : f(A\mathbf{x}) = f(\mathbf{x})\}$ at $A = I_n$ and making it pass through origin.

**Definition 2.3** (Lie algebra $\mathfrak{g}_f$ of a polynomial $f$)**.** The Lie algebra of an $n$-variate polynomial $f(\mathbf{x})$ is denoted as $\mathfrak{g}_f$ and it consists of matrices $E = (e_{ij})_{i,j\in[n]} \in \mathcal{M}_n$ that satisfy $\sum_{i,j\in[n]} e_{ij} x_j \cdot \frac{\partial f}{\partial x_i} = 0$.

Note that $\mathfrak{g}_f$ is a vector space. It also follows that a basis of $\mathfrak{g}_f$ can be computed in randomized polynomial-time from blackbox access to $f$ by solving a linear system (see [Kay12]).

**Fact 1.** *If $f(\mathbf{x}) = g(A\mathbf{x})$ for an $A \in GL(n,\mathbb{F})$, then $\mathfrak{g}_f = A^{-1}\mathfrak{g}_g A$.*

## 3 Symmetries and Lie algebra of Tr-IMM

The symmetries and the Lie algebra $\mathfrak{g}_{\text{Tr-IMM}}$ of Tr-IMM$_{w,d}$ have been studied in [Ges16] over $\mathbb{C}$. Here, we work out the exact structure of the matrices in $\mathfrak{g}_{\text{Tr-IMM}}$ with respect to the variable ordering mentioned above, and use it to identify the $\mathfrak{g}_{\text{Tr-IMM}}$-invariant subspaces of $\mathbb{F}^n$ and the symmetries of Tr-IMM$_{w,d}$ over $\mathbb{F}$. These facts about the Lie algebra and the symmetries will be used in the proofs of Theorems 1, 2 and 3. The missing proofs of this section are given in Appendix B.

**Claim 3.1.** *If $E \in \mathfrak{g}_{Tr\text{-}IMM}$ then $E$ is block-diagonal.*

Define the spaces $\mathcal{B}_0, \ldots, \mathcal{B}_{d-1}$ of block-diagonal matrices as follows: Every matrix in $\mathcal{B}_k$ is a block-diagonal matrix whose non-zero entries are confined to the rows and columns indexed by $\mathbf{x}_k$ and $\mathbf{x}_{k+1}$ variables. For $k \in [0, d-2]$ and $B \in \mathcal{B}_k$, let $[B]_k$ be the $2w^2 \times 2w^2$ sub-matrix of $B$ whose rows and columns are indexed by $\mathbf{x}_k$ and $\mathbf{x}_{k+1}$ variables. For $B \in \mathcal{B}_{d-1}$, let $[B]_{d-1}$ be the $2w^2 \times 2w^2$ sub-matrix of $B$ whose rows and columns are indexed by $\mathbf{x}_{d-1}$ and $\mathbf{x}_0$ variables, i.e., we let the $\mathbf{x}_{d-1}$ variables index the rows and columns of $B_{d-1}$ before the $\mathbf{x}_0$ variables. If $d$ is even then

$$\begin{aligned}\mathcal{B}_k &:= \left\{ B \in \mathcal{M}_n \;:\; [B]_k = \begin{bmatrix} I_w \otimes M^T & \mathbf{0} \\ \mathbf{0} & -I_w \otimes M \end{bmatrix} \text{ for } M \in \mathcal{M}_w \right\} \text{ if } k \text{ is even,} \\ &:= \left\{ B \in \mathcal{M}_n \;:\; [B]_k = \begin{bmatrix} M^T \otimes I_w & \mathbf{0} \\ \mathbf{0} & -M \otimes I_w \end{bmatrix} \text{ for } M \in \mathcal{M}_w \right\} \text{ if } k \text{ is odd.} \end{aligned} \tag{1}$$

If $d$ is odd, then the definition of $\mathcal{B}_k$ remains the same except for $\mathcal{B}_{d-1}$ which is defined as

$$\mathcal{B}_{d-1} := \left\{ B \in \mathcal{M}_n \;:\; [B]_{d-1} = \begin{bmatrix} I_w \otimes M^T & \mathbf{0} \\ \mathbf{0} & -M \otimes I_w \end{bmatrix} \text{ for } M \in \mathcal{M}_w \right\}.$$

**Lemma 3.1.** *The space $\mathcal{B}_0 + \ldots + \mathcal{B}_{d-1}$ is contained in $\mathfrak{g}_{Tr\text{-}IMM}$.*

**Lemma 3.2.** *Suppose $B \in \mathfrak{g}_{Tr\text{-}IMM}$ and there is a $k \in [0, d-1]$ such that the non-zero entries of $B$ are confined to the rows and columns that are indexed by $\mathbf{x}_k$ and $\mathbf{x}_{k+1}$ variables. Then $B \in \mathcal{B}_k$.*

In fact $\mathfrak{g}_{\text{Tr-IMM}} = \mathcal{B}_0 + \ldots + \mathcal{B}_{d-1}$, however we do not prove this stronger statement here. Let $e_i \in \mathbb{F}^n$ be the vector with 1 in the entry indexed by $x_i \in \mathbf{x}$ and zero elsewhere. A subspace of $\mathbb{F}^n$ is a coordinate subspace if it is spanned by a set of $e_i$'s. Let $\mathcal{U}_k = \text{span}_{\mathbb{F}}\{e_i \mid x_i \in \mathbf{x}_k\}$.

**Claim 3.2.** *Any non-zero $\mathfrak{g}_{Tr\text{-}IMM}$-invariant subspace is a coordinate subspace of $\mathbb{F}^n$.*

**Lemma 3.3.** *The only irreducible $\mathfrak{g}_{Tr\text{-}IMM}$-invariant subspaces of $\mathbb{F}^n$ are $\mathcal{U}_0, \ldots, \mathcal{U}_{d-1}$.*

**Corollary 3.1.** *If $f = Tr\text{-}IMM_{w,d}(A\mathbf{x})$, where $A \in GL(n,\mathbb{F})$, then the only irreducible $\mathfrak{g}_f$-invariant subspaces of $\mathbb{F}^n$ are $A^{-1}\mathcal{U}_0, \ldots, A^{-1}\mathcal{U}_{d-1}$.*

The above lemmas help us derive the group of symmetries of Tr-IMM$_{w,d}$ over $\mathbb{F}$.

**Lemma 3.4.** *Let Tr-IMM$_{w,d} = \mathrm{tr}(Q'_0 \cdots Q'_{d-1})$, where $Q'_0 \cdots Q'_{d-1}$ is a full-rank $(w,d,n)$-matrix product in $\mathbf{x}$ variables over $\mathbb{F}$. Then there are $C_0, \ldots, C_{d-1} \in GL(w, \mathbb{F})$ and $\ell \in [0, d-1]$ such that either $Q'_k = C_k \cdot Q_{\ell+k} \cdot C_{k+1}^{-1}$ for $k \in [0, d-1]$ or $Q'_k = C_k \cdot Q_{\ell-k}^T \cdot C_{k+1}^{-1}$ for $k \in [0, d-1]$.*

# 4 Reduction from TRACE to DET: Proof of Theorem 1

The reduction is given in Algorithm 1. The algorithm proceeds by assuming that the input polynomial $f$ is equivalent to Tr-IMM$_{w,d}$ for some $w \geq 2$. A final polynomial identity test (PIT) takes care of the case when it is not. Algorithm 1 has two main steps – reduction from TRACE to TRACE-TI (Algorithm 4), and reduction from TRACE-TI to DET (Algorithm 2). Algorithm 4 is inspired by a similar reduction in [KNST19] for the IMM polynomial. Below we discuss the proof strategy of Algorithm 4, and give the details in Appendix C. Algorithm 2 is given in Section 4.1.

**Reduction from TRACE to TRACE-TI.** First, we compute bases of the irreducible $\mathfrak{g}_f$-invariant subspaces of $\mathbb{F}^n$. By Corollary 3.1, these are bases of the spaces $A^{-1}\mathcal{U}_{\sigma(0)}, \ldots, A^{-1}\mathcal{U}_{\sigma(d-1)}$, where $\sigma$ is an unknown permutation on $\{0, \ldots, d-1\}$. As $\dim_{\mathbb{F}}(\mathcal{U}_k) = w^2$, we get $w$. Now, let $V_k$ be the $n \times w^2$ matrix consisting of the basis vectors of $A^{-1}\mathcal{U}_{\sigma(k)}$. Form the $n \times n$ matrix $V = [V_0 \mid V_1 \mid \ldots \mid V_{d-1}]$. Observe that $V = A^{-1} \cdot E$, where $E$ is a "block-permuted" invertible matrix (by the definition of $\mathcal{U}_k$). Thus, $h(\mathbf{x}) := f(V\mathbf{x}) = \text{Tr-IMM}_{w,d}(E\mathbf{x})$. We now make use of the evaluation dimension measure (Definition C.1) on $h$ to essentially ensure that $E$ is a block-diagonal matrix.

**Algorithm 1** Reduction from TRACE to DET

INPUT: Blackbox access to an $n$-variate, degree $d$ polynomial $f$ and oracle access to DET.
OUTPUT: If there is an $w \in \mathbb{N}$ such that $f$ is equivalent to Tr-IMM$_{w,d}$ then output an $A \in \mathrm{GL}(n, \mathbb{F})$ such that $f(\mathbf{x}) = \text{Tr-IMM}_{w,d}(A\mathbf{x})$. Otherwise output 'No such $w$ exists'.

Reduction to TRACE-TI

1: Use Algorithm 4 with input $f$ to compute $A' \in \mathrm{GL}(n, \mathbb{F})$ and a $w \in \mathbb{N}$ such that $h(\mathbf{x}) = f(A'\mathbf{x})$ is a $d$-tensor in the variable sets $\mathbf{x}_0, \ldots, \mathbf{x}_{d-1}$ which is isomorphic to Tr-IMM$_{w,d}$. If Algorithm 4 outputs 'No', output 'No such $w$ exists'.

Reduction from TRACE-TI to DET

2: Use Algorithm 2 with input $h$, $w$ and oracle access to DET to compute matrices $B_0, \ldots, B_{d-1} \in \mathrm{GL}(w^2, \mathbb{F})$ such that $h(\mathbf{x}) = \text{Tr-IMM}_{w,d}(B_0\mathbf{x}_0, \ldots, B_{d-1}\mathbf{x}_{d-1})$. If Algorithm 2 outputs 'No' then output 'No such $w$ exists'.
3: Let $B \in \mathrm{GL}(n, \mathbb{F})$ be the block-diagonal matrix whose $k$-th block is $B_k$, and let $A = B(A')^{-1}$.

Final PIT

4: Pick a random point $\mathbf{a} \in S^n$ where $S \subseteq \mathbb{F}$ is of size $n^5$. If $f(\mathbf{a}) = \text{Tr-IMM}_{w,d}(A\mathbf{a})$ then output $w$ and $A$, else output 'No such $w$ exists'.

### 4.1 Reduction from TRACE-TI to DET

We will use a few terminologies and notations about matrices, matrix products and ABP that are defined in Appendix A. The following two claims (proved in Appendix D) help in the argument.

**Claim 4.1.** *Let $X$ be a $w \times w$ full-rank linear matrix and $Y = I_w \otimes X$. Then there does not exist non-zero matrices $T, S \in \mathcal{M}_{w^2}(\mathbb{F})$ such that $T \cdot Y = Y^T \cdot S$.*

**Claim 4.2.** *Let $X$ be a $w \times w$ full-rank linear matrix and $Y = I_w \otimes X$, and suppose $T, S \in \mathcal{M}_{w^2}(\mathbb{F})$ such that $T \cdot Y = Y \cdot S$. Then $T = S = M \otimes I_w$ for some $M \in \mathcal{M}_w(\mathbb{F})$.*

The correctness of Algorithm 2 is argued below by tracing its steps.

**Algorithm 2** Reduction from TRACE-TI to DET

INPUT: A $w \in \mathbb{N}$, blackbox access to $d$-tensor $h(\mathbf{x}_0, \ldots, \mathbf{x}_{d-1})$ that is isomorphic to Tr-IMM$_{w,d}$, and oracle access to DET.
OUTPUT: Matrices $B_0, \ldots, B_{d-1} \in \mathrm{GL}(w^2, \mathbb{F})$ such that $h(\mathbf{x}) = \text{Tr-IMM}_{w,d}(B_0\mathbf{x}_0, \ldots, B_{d-1}\mathbf{x}_{d-1})$.

1: Use the set-multilinear ABP reconstruction algorithm (which follows from [KS03]) to construct a $(w^2, d, n)$ set-multilinear ABP $Y'_0 \ldots Y'_{d-1}$ in $\mathbf{x}_0, \ldots, \mathbf{x}_{d-1}$ variables that computes $h$.
2: For $k \in [1, d-2]$, use the factorization algorithm in [KT90] to compute blackbox access to a degree-$w$ polynomial $g_k$ such that $\det(Y'_k) = \alpha_k g_k(\mathbf{x}_k)^w$, where $\alpha_k \in \mathbb{F}^\times$.
3: For $k \in [1, d-2]$, use the DET oracle on input $g_k$ to compute $X'_k$ such that $\det(X'_k) = g_k$. If DET returns $g_k$ is not equivalent to Det$_w$, then output 'No'.

4: For $k \in [1, d-2]$, let $Z_k = I_w \otimes X'_k$.
5: For $k \in [1, d-2]$, compute $T'_{k-1}$, $S'_k \in \mathrm{GL}(w^2, \mathbb{F})$ such that either $T'_{k-1} \cdot Y'_k = Z_k \cdot S'_k$ or $T'_{k-1} \cdot Y'_k = Z_k^T \cdot S'_k$. If both equalities are satisfied, output 'No' (see Observation 4.1).

6: Let $\widehat{Y}_0 = Y'_0 \cdot (T'_0)^{-1}$, $\widehat{Y}_k = (T'_{k-1}) \cdot Y'_k \cdot (T'_k)^{-1}$ for $k \in [1, d-3]$, $\widehat{Y}_{d-2} = (T'_{d-3}) \cdot Y'_{d-2} \cdot (S'_{d-2})^{-1}$, and $\widehat{Y}_{d-1} = S'_{d-2} \cdot Y'_{d-1}$.
7: Let $\widehat{X}_{d-2}$ be such that $\widehat{Y}_{d-2} = I_w \otimes \widehat{X}_{d-2}$, and for $k \in [1, d-3]$ construct $\widehat{M}_k \in \mathrm{GL}(w, \mathbb{F})$ and $\widehat{X}_k$ such that $\widehat{Y}_k = (\widehat{M}_k \otimes I_w) \cdot (I_w \otimes \widehat{X}_k)$. (See Observation 4.3.)
8: Let $\overline{Y}_{d-1} = (\prod_{k=1}^{d-3} (\widehat{M}_k \otimes I_w)) \cdot \widehat{Y}_{d-1}$. Construct $\widehat{X}_{d-1}$ such that its $(i,j)$-th entry is the $((j-1)w + i)$-th entry of $\overline{Y}_{d-1}$, and $\widehat{X}_0$ such that its $(i,j)$-th entry is the $((i-1)w + j)$-th entry of $\widehat{Y}_0$.

9: Obtain the transformations $B_0, \ldots, B_{d-1} \in \mathrm{GL}(w^2, \mathbb{F})$ from (the entries of) $\widehat{X}_0, \ldots, \widehat{X}_{d-1}$ respectively. Return $B_0, \ldots, B_{d-1}$.

**Steps 1–3**: Assume that $h$ is isomorphic to Tr-IMM$_{w,d}$. Hence, there is a full-rank $(w, d, n)$ set-multilinear matrix product $X_0 \ldots X_{d-1}$ in $\mathbf{x}_0, \ldots, \mathbf{x}_{d-1}$ variables such that $h = \mathrm{tr}(X_0 \ldots X_{d-1})$. From Fact 2, $h$ is computed by the $(w^2, d, n)$-set-multilinear ABP $Y_0 \ldots Y_{d-1}$, where

$$
\begin{aligned}
Y_0 &= (X_0(1,1), \ldots, X_0(1,w), X_0(2,1), \ldots, X_0(2,w), \ldots, X_0(w,1), \ldots, X_0(w,w)) \\
Y_k &= I_w \otimes X_k \quad \text{for } k \in [1, d-2] \\
Y_{d-1} &= (X_{d-1}(1,1), \ldots, X_{d-1}(w,1), X_{d-1}(1,2), \ldots, X_{d-1}(w,2), \ldots, X_{d-1}(1,w), \ldots, X_{d-1}(w,w))^T .
\end{aligned}
$$

Using the randomized polynomial-time set-multilinear ABP reconstruction algorithm in [KS03], a $(w^2, d, n)$ set-multilinear ABP $Y'_0 \dots Y'_{d-1}$ computing $h$ is constructed in Step 1. It follows from the properties of this algorithm and the ABP $Y_0 \dots Y_{d-1}$ that there are $T_0, \dots, T_{d-2} \in \mathrm{GL}(w^2, \mathbb{F})$ so that

$$Y'_0 = Y_0 \cdot T_0\,, \qquad Y'_k = T_{k-1}^{-1} \cdot Y_k \cdot T_k \quad \text{for } k \in [1, d-2], \text{ and} \qquad Y'_{d-1} = T_{d-2}^{-1} \cdot Y_{d-1}\,.^{20}$$

Hence, for all $k \in [1, d-2]$, $\det(Y'_k) = c_k(\det(X_k))^w$, where $c_k \in \mathbb{F}^\times$. As the determinant polynomial is irreducible, at Step 2, we have $g_k = \beta_k \det(X_k) = \det(\mathrm{diag}(\beta_k, 1, \dots, 1) \cdot X_k)$ for some $\beta_k \in \mathbb{F}^\times$ which implies $g_k$ is equivalent to $\mathrm{Det}_w$. At step 3, DET on input $g_k$ returns $X'_k$ such that

$$X_k = C_k \cdot X'_k \cdot D_k \quad \text{or} \quad X_k = C_k \cdot (X'_k)^T \cdot D_k \qquad \text{where } C_k, D_k \in \mathrm{GL}(w, \mathbb{F}).$$

The above follows from the group of symmetries of $\mathrm{Det}_w$ (see Fact 1 in [KNS19]).

**Steps 4–5**: At Step 4, for $k \in [1, d-2]$, the matrix $Z_k = I_w \otimes X'_k$ satisfies

$$Y_k = (I_w \otimes C_k) \cdot Z_k \cdot (I_w \otimes D_k) \quad \text{or} \quad Y_k = (I_w \otimes C_k) \cdot Z_k^T \cdot (I_w \otimes D_k).$$

Hence, at Step 5 there are $T'_{k-1} := (I_w \otimes C_k^{-1}) \cdot T_{k-1}$ and $S'_k := (I_w \otimes D_k) \cdot T_k$ in $\mathrm{GL}(w^2, \mathbb{F})$ such that

$$T'_{k-1} \cdot Y'_k = Z_k \cdot S'_k \quad \text{or} \quad T'_{k-1} \cdot Y'_k = Z_k^T \cdot S'_k.$$

Observation 4.1 uses Claim 4.1 to show that at Step 5 we can identify between the above two cases, as only one of them is true (proof in Appendix D).

**Observation 4.1.** *If $h(\mathbf{x}_0, \dots, \mathbf{x}_{d-1})$ is isomorphic to Tr-IMM$_{w,d}$ then for matrices $Y'_k$ and $Z_k$ as computed in Algorithm 2, where $k \in [1, d-2]$, there are no matrices $T'_{k-1}, S'_k \in GL(w^2, \mathbb{F})$ such that both $T'_{k-1} \cdot Y'_k = Z_k \cdot S'_k$ and $T'_{k-1} \cdot Y'_k = Z_k^T \cdot S'_k$ are simultaneously true.*

At step 5 the matrices $T'_{k-1}$ and $S'_k$ are computed by solving linear equations. Choosing a solution at random from the solution space ensures that the computed matrices $T'_{k-1}$ and $S'_k$ are invertible with high probability. Henceforth, we assume that $T'_{k-1} \cdot Y'_k = Z_k \cdot S'_k$. The proof for $T'_{k-1} \cdot Y'_k = Z_k^T \cdot S'_k$ is similar. In Observation 4.2 we show that $T'_{k-1}$ and $S'_k$ are related to $T_{k-1}$ and $T_k$ respectively for $k \in [1, d-2]$. The proof of Observation 4.2, which uses Claim 4.2, is in Appendix D.

**Observation 4.2** (Uniqueness of $T'_{k-1}$ and $S'_k$)**.** *The matrices $T'_{k-1}$ and $S'_k$ computed at Step 5 of Algorithm 2, where $k \in [1, d-2]$, satisfy the following: $(T'_{k-1})^{-1} = T_{k-1}^{-1} \cdot (I_w \otimes C_k) \cdot (M_k^{-1} \otimes I_w)$ and $S'_k = (M_k \otimes I_w) \cdot (I_w \otimes D_k) \cdot T_k$, where $M_k \in GL(w, \mathbb{F})$.*

**Steps 6–8**: Observation 4.3 proved in Appendix D describes the structure of the matrices $\widehat{Y}_0, \dots, \widehat{Y}_{d-1}$ computed at Step 6. Clearly, $\widehat{Y}_0 \dots \widehat{Y}_{d-1} = Y'_0 \dots Y'_{d-1}$ is a set-multilinear ABP computing $h$.

**Observation 4.3.** *Let $M_1, \dots, M_{d-2}$ be the matrices as defined in Observation 4.2. Then*

1. $\widehat{Y}_k = (M_k M_{k+1}^{-1} \otimes I_w) \cdot (I_w \otimes (C_k^{-1} \cdot X_k \cdot C_{k+1}))$ *for $k \in [1, d-3]$,*
2. $\widehat{Y}_{d-2} = I_w \otimes (C_{d-2}^{-1} \cdot X_{d-2} \cdot D_{d-2}^{-1})$,

[20] See Appendix A: Set-multilinear ABP reconstruction, for an explanation.

3. $\widehat{Y}_0 = Y_0 \cdot (I_w \otimes C_1) \cdot (M_1^{-1} \otimes I_w)$, *and* $\widehat{Y}_{d-1} = (M_{d-2} \otimes I_w) \cdot (I_w \otimes D_{d-2}) \cdot Y_{d-1}$.

By the above observation, at Step 7, $\widehat{X}_{d-2} = C_{d-2}^{-1} \cdot X_{d-2} \cdot D_{d-2}^{-1}$. Moreover, the structure of $\widehat{Y}_k$ (as stated in the observation) enables the algorithm to factor it in Step 7 and obtain $\widehat{X}_k, \widehat{M}_k$ such that

$$\widehat{X}_k = a_k(C_k^{-1} \cdot X_k \cdot C_{k+1}) \quad \text{and } \widehat{M}_k = a_k^{-1}(M_k \cdot M_{k+1}^{-1}) \quad \text{for some } a_k \in \mathbb{F}^\times.$$

Let $a = \prod_{k=1}^{d-3} a_k$. Then at step 8, $\overline{Y}_{d-1} = a^{-1} \cdot (M_1 \otimes I_w) \cdot (I_w \otimes D_{d-2}) \cdot Y_{d-1}$. Now, it is a simple exercise to verify that at step 8

$$\widehat{X}_0 = (M_1^T)^{-1} \cdot X_0 \cdot C_1 \quad \text{and} \quad \widehat{X}_{d-1} = a^{-1}(D_{d-2} \cdot X_{d-1} \cdot M_1^T).$$

**Step 9**: Therefore, $h = \mathrm{tr}(\widehat{X}_0 \dots \widehat{X}_{d-1})$. The transformation $B_k \in \mathrm{GL}(w^2, \mathbb{F})$ is such that its rows are the coefficient vectors of the linear forms in $\widehat{X}_k$. Hence, $h = \text{Tr-IMM}_{w,d}(B_0\mathbf{x}_0, \dots, B_{d-1}\mathbf{x}_{d-1})$.

# 5 Reduction from FMAI to MMTI : Proof of Theorem 2

## 5.1 Characterization of Tr-IMM by its Lie algebra

The following lemma gives a characterization of $\text{Tr-IMM}_{w,d}$ by its Lie algebra. The spaces $\mathcal{B}_0, \dots, \mathcal{B}_{d-1}$ are as defined in Section 3. The missing proofs are in Appendix F .

**Lemma 5.1.** *Let $f$ be a non-zero $d$-tensor in the variable sets $\mathbf{x}_0, \dots, \mathbf{x}_{d-1}$ such that for all $k \in [0, d-1]$ $\mathcal{B}_k \subseteq \mathfrak{g}_f$. Then there is an $\alpha \in \mathbb{F}^\times$ such that $f(\mathbf{x}) = \alpha \cdot$ Tr-IMM$_{w,d}(\mathbf{x})$.*

**Corollary 5.1.** *Let $B \in GL(n, \mathbb{F})$ be a block-diagonal matrix with individual blocks $B_0, \dots, B_{d-1}$ and $f$ be a non-zero $d$-tensor in the variable sets $\mathbf{x}_0, \dots, \mathbf{x}_{d-1}$ such that for all $k \in [0, d-1]$, $B^{-1} \cdot \mathcal{B}_k \cdot B \subseteq \mathfrak{g}_f$. Then there is an $\alpha \in \mathbb{F}^\times$ such that $f(\mathbf{x}) = \alpha \cdot$ Tr-IMM$_{w,d}(B_0\mathbf{x}_0, \dots, B_{d-1}\mathbf{x}_{d-1})$.*

## 5.2 Proof of Theorem 2

Algorithm 3 takes as input a basis $\{E_1, E_2, \dots, E_r\}$ of an algebra $\mathcal{A} \subseteq \mathcal{M}_m(\mathbb{F})$, and if $\mathcal{A} \cong \mathcal{M}_w$ for some $w \in \mathbb{N}$, then it computes a 4-tensor $f$ in the variable sets $\mathbf{x}_0, \mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3$ in deterministic polynomial time such that $f$ is isomorphic to $\text{Tr-IMM}_{w,4}$. It then uses Algorithm 7 in Theorem 3 (see Appendix E) to find an isomorphism from $f$ to $\text{Tr-IMM}_{w,4}$ using oracle access to MMTI in randomized polynomial time. An easy check at the end of the algorithm ensures that if the algorithm outputs an isomorphism then it is correct. Thus, we need to prove that if $\mathcal{A}$ is isomorphic to $\mathcal{M}_w$ for some $w \in \mathbb{N}$ then the algorithm outputs an isomorphism. This is argued by tracing the steps of the algorithm assuming $\mathcal{A}$ is isomorphic to $\mathcal{M}_w$ for some $w \in \mathbb{N}$.

**Steps 1–2**: At Step 2 there is a $K \in \mathrm{GL}(w^2, \mathbb{F})$ and a basis $\{C_{1,1}, \dots, C_{w,w}\}$ of $\mathcal{M}_w$ such that $L_{i,j} = K^{-1} \cdot (I_w \otimes C_{i,j}) \cdot K$ for all $i, j \in [w]$ (by the Skolem-Noether theorem, see next claim).

**Claim 5.1.** *Suppose $\mathcal{A} \cong \mathcal{M}_w$ for some $w \in \mathbb{N}$. Then there exists a $K \in GL(w^2, \mathbb{F})$ and linearly independent matrices $\{C_{1,1}, \dots, C_{w,w}\}$ in $\mathcal{M}_w$ such that $L_{i,j} = K^{-1} \cdot (I_w \otimes C_{i,j}) \cdot K$ for all $i, j \in [w]$.*

**Step 3**: The space spanned by $\{L_{1,1}^T, \dots, L_{w,w}^T\}$ is $K^T \cdot (I_w \otimes \mathcal{M}_w) \cdot (K^T)^{-1}$.

**Algorithm 3** Reduction from FMAI to MMTI

INPUT: A basis $\{E_1, E_2, \ldots, E_r\}$ of an algebra $\mathcal{A} \subseteq \mathcal{M}_m(\mathbb{F})$, and oracle access to MMTI.
OUTPUT: If $\mathcal{A} \cong \mathcal{M}_w(\mathbb{F})$ for some $w \in \mathbb{N}$ then output an algebra isomorphism $\phi : \mathcal{A} \to \mathcal{M}_w$, otherwise output 'No $w \in \mathbb{N}$ such that $\mathcal{A} \cong \mathcal{M}_w$'.

1: If $r \neq w^2$ for any $w \in \mathbb{N}$, then output 'No $w \in \mathbb{N}$ such that $\mathcal{A} \cong \mathcal{M}_w$'.

2: Rename and order the basis elements as $E_{1,1}, \ldots, E_{1,w}, \ldots, E_{w,1}, \ldots, E_{w,w}$. Compute matrices $L_{1,1}, \ldots, L_{w,w}$, whose rows and columns are indexed by the above basis elements in order, as follows: $L_{i,j}$ is the matrix corresponding to the left multiplication of $E_{i,j}$ on $E_{1,1}, \ldots E_{w,w}$. In particular, $E_{i,j} \cdot E_{i_2,j_2} = \sum_{i_1,j_1 \in [w]} L_{i,j}((i_1,j_1),(i_2,j_2))E_{i_1,j_1}$.

3: Compute a basis of the space spanned by matrices in $\mathcal{M}_{w^2}$ that commute with $\{L_{1,1}^T, \ldots, L_{w,w}^T\}$. If the dimension of this space is not $w^2$, then output 'No $w \in \mathbb{N}$ such that $\mathcal{A} \cong \mathcal{M}_w$'. Otherwise, let the computed basis be $\{N_{1,1}, \ldots, N_{w,w}\}$.

4: Compute a non-zero 4-tensor $f$ in $\mathbf{x}_0, \ldots, \mathbf{x}_3$ variables whose coefficients satisfy the following equations: a) for all $k \in [0,3]$, $k$ even, and for all $L \in \{L_{1,1}, \ldots L_{w,w}\}$

$$\sum_{i_1,j_1,i_2,j_2 \in [w^2]} L^T((i_1,j_1)(i_2,j_2))x_{i_2,j_2}^{(k)} \frac{\partial f}{x_{i_1,j_1}^{(k)}} \quad - \sum_{i_1,j_1,i_2,j_2 \in [w^2]} L((i_1,j_1)(i_2,j_2))x_{j_2,i_2}^{(k+1)} \frac{\partial f}{x_{j_1,i_1}^{(k+1)}} = \ 0. \quad (2)$$

b) for all $k \in [0,3]$, $k$ odd, and for all $N \in \{N_{1,1}, \ldots N_{w,w}\}$

$$\sum_{i_1,j_1,i_2,j_2 \in [w^2]} N^T((i_1,j_1)(i_2,j_2))x_{j_2,i_2}^{(k)} \frac{\partial f}{x_{j_1,i_1}^{(k)}} \quad - \sum_{i_1,j_1,i_2,j_2 \in [w^2]} N((i_1,j_1)(i_2,j_2))x_{i_2,j_2}^{(k+1)} \frac{\partial f}{x_{i_1,j_1}^{(k+1)}} = 0. \quad (3)$$

5: Use Algorithm 7 on input $f$ and with oracle access to MMTI. If the algorithm outputs 'No' then output 'No $w \in \mathbb{N}$ such that $\mathcal{A} \cong \mathcal{M}_w$'. Otherwise, let $B_0, B_1, B_2, B_3$ be the output of the algorithm such that $f = \text{Tr-IMM}_{w,4}(B_0\mathbf{x}_0, B_1\mathbf{x}_1, B_2\mathbf{x}_2, B_3\mathbf{x}_3)$.

6: Check if there exist matrices $F_{1,1}, \ldots, F_{w,w} \in \mathcal{M}_w$ such that $B_0 \cdot L_{i,j}^T \cdot B_0^{-1} = I_w \otimes F_{i,j}^T$ and $B_1 \cdot L_{i,j} \cdot B_1^{-1} = I_w \otimes F_{i,j}$ for all $i,j \in [w]$. If such matrices do not exist then output 'No $w \in \mathbb{N}$ such that $\mathcal{A} \cong \mathcal{M}_w$', otherwise output $\phi : \mathcal{A} \to \mathcal{M}_w$, where $\phi(E_{i,j}) = F_{i,j}$ for all $i,j \in [w]$ (extended linearly to the whole of $\mathcal{A}$) as the algebra isomorphism from $\mathcal{A}$ to $\mathcal{M}_w$.

**Observation 5.1.** *The space of matrices in $\mathcal{M}_{w^2}$ that commute with every matrix in $K^T \cdot (I_w \otimes \mathcal{M}_w) \cdot (K^T)^{-1}$ is $K^T \cdot (\mathcal{M}_w \otimes I_w) \cdot (K^T)^{-1}$. So, $\{N_{1,1}, \ldots, N_{w,w}\}$ is a basis of $K^T \cdot (\mathcal{M}_w \otimes I_w) \cdot (K^T)^{-1}$.*

**Step 4**: Let $n = 4w^2$. For $k \in [0,3]$, let $\mathcal{B}'_k$ be the following spaces: Every matrix in $\mathcal{B}'_k$ is a $n \times n$ block-diagonal matrix (with rows and columns indexed by $\mathbf{x}_0, \ldots, \mathbf{x}_3$) and its non-zero entries are confined to the rows and columns indexed by $\mathbf{x}_k$ and $\mathbf{x}_{k+1}$. For $B \in \mathcal{B}_k$, let $[B]_k$ be the $2w^2 \times 2w^2$

sub-matrix of $B$ as defined in Equation 1 (Section 3). Then

$$\mathcal{B}'_k := \left\{ B \in \mathcal{M}_n \ : \ [B]_k = \begin{bmatrix} K^T \cdot (I_w \otimes M^T)(K^T)^{-1} & \mathbf{0} \\ \mathbf{0} & K^{-1} \cdot (-I_w \otimes M) \cdot K \end{bmatrix} \text{ for } M \in \mathcal{M}_w \right\} \text{ if } k \text{ is even,}$$

$$:= \left\{ B \in \mathcal{M}_n \ : \ [B]_k = \begin{bmatrix} K^{-1} \cdot (M^T \otimes I_w) \cdot K & \mathbf{0} \\ \mathbf{0} & K^T \cdot (-M \otimes I_w) \cdot (K^T)^{-1} \end{bmatrix} \text{ for } M \in \mathcal{M}_w \right\} \text{ if } k \text{ is odd.}$$

The following observation follows from Lemma 3.1 and Fact 1.

**Observation 5.2.** *The Lie algebra of Tr-IMM$_{w,4}((K^T)^{-1}\mathbf{x}_0, K\mathbf{x}_1, (K^T)^{-1}\mathbf{x}_2, K\mathbf{x}_3)$ contains $\mathcal{B}'_0, \mathcal{B}'_1, \mathcal{B}'_2, \mathcal{B}'_3$.*

At Step 4, Algorithm 3 computes a non-zero 4-tensor $f$ such that $\mathcal{B}'_k \subseteq \mathfrak{g}_f$ for all $k \in [0,3]$. Equation 2 ensures $\mathcal{B}'_0, \mathcal{B}'_2 \in \mathfrak{g}_f$, and Equation 3 ensures $\mathcal{B}'_1, \mathcal{B}'_3 \in \mathfrak{g}_f$. That the algorithm is able to compute a non-zero $f$ (by solving a linear system) follows from Observation 5.2. Since the number of monomials in $f$ is at most $w^8$, this step runs in polynomial time.

**Step 5**: From Corollary 5.1 it follows that $f(\mathbf{x}) = \alpha \cdot \text{Tr-IMM}_{w,4}((K^T)^{-1}\mathbf{x}_0, K\mathbf{x}_1, (K^T)^{-1}\mathbf{x}_2, K\mathbf{x}_3)$ for some $\alpha \in \mathbb{F}^\times$. Hence, at step 5 with high probability Algorithm 7 outputs four matrices $B_0, B_1, B_2, B_3 \in \text{GL}(w^2, \mathbb{F})$ such that $f(\mathbf{x}) = \text{Tr-IMM}_{w,4}(B_0\mathbf{x}_0, B_1\mathbf{x}_1, B_2\mathbf{x}_2, B_3\mathbf{x}_3)$.

**Step 6**: Let $B$ be the block-diagonal matrix whose $k$-th block is $B_k$, for $k \in [0,3]$. Since $\mathcal{B}'_0 \subseteq \mathfrak{g}_f$ and $\mathfrak{g}_f = B^{-1} \cdot \mathfrak{g}_{\text{Tr-IMM}} \cdot B$ (from Fact 1), $B \cdot \mathcal{B}'_0 \cdot B^{-1} \subseteq \mathfrak{g}_{\text{Tr-IMM}}$. Observe that every matrix in $B \cdot \mathcal{B}'_0 \cdot B^{-1}$ is block-diagonal with its non-zero entries confined to the first two blocks. Hence, from Lemma 3.2, and the fact that both the spaces $B \cdot \mathcal{B}'_0 \cdot B^{-1}$ and $\mathcal{B}_0$ have dimension $w^2$, we have $B \cdot \mathcal{B}'_0 \cdot B^{-1} = \mathcal{B}_0$. In particular, for every $i, j \in [w]$ there is an $F_{i,j} \in \mathcal{M}_w$ such that $B_0 \cdot L^T_{i,j} \cdot B_0^{-1} = I_w \otimes F^T_{i,j}$ and $B_1 \cdot L_{i,j} \cdot B_1^{-1} = I_w \otimes F_{i,j}$. Finally, verify that $\phi(E_{i,j}) = F_{i,j}$ is an algebra isomorphism.

*Comparison with [GGKS19]*: In [GGKS19], FMAI is reduced to DET by using the fact that $\text{Det}_w$ is characterized by its Lie algebra (see Lemma 7.1 in [GGKS19]). If the input algebra $\mathcal{A}$ is isomorphic to $\mathcal{M}_w$ then the algorithm in [GGKS19] computes a *degree-w* polynomial $f$ in $w^2$ variables such that $\mathfrak{g}_f$ contains the Lie algebra of a polynomial equivalent to $\text{Det}_w$. Hence, the time complexity of their algorithm is $w^{O(w)}$. Algorithm 3 follows the same approach, but computes a *degree four* polynomial $f$ such that $\mathfrak{g}_f$ contains the Lie algebra of a polynomial equivalent to Tr-IMM$_{w,4}$. So, the complexity of this algorithm is $w^{O(1)}$.

## Acknowledgments

We are thankful to Avi Wigderson for his suggestion on designing an equivalence testing algorithm for Tr-IMM at the end of VN's presentation at CCC 2017. We would also like to thank Christian Ikenmeyer for his question on equivalence testing for Tr-IMM which encouraged us to work on this problem. Thanks also to Neeraj Kayal and Ankit Garg for helpful discussions, and particularly to Neeraj for pointing us to [GQ19]. VN is thankful to be funded by the European Union's Horizon 2020 research and innovation programme under grant agreement No 682203 -ERC-[ Inf-Speed-Tradeoff].

## References


[Ara11] Manuel Araújo. Classification of Quadratic Forms. https://www.math.tecnico.ulisboa.pt/~ggranja/manuel.pdf, 2011.

[AS05] Manindra Agrawal and Nitin Saxena. Automorphisms of finite rings and applications to complexity of problems. In *23rd Annual Symposium on Theoretical Aspects of Computer Science, STACS 2005*, pages 1–17, 2005.

[AS06] Manindra Agrawal and Nitin Saxena. Equivalence of f-algebras and cubic forms. In *23rd Annual Symposium on Theoretical Aspects of Computer Science, STACS 2006*, pages 115–126, 2006.

[BIM+20] Markus Bläser, Christian Ikenmeyer, Meena Mahajan, Anurag Pandey, and Nitin Saurabh. Algebraic branching programs, border complexity, and tangent spaces. *Electronic Colloquium on Computational Complexity (ECCC)*, 27:31, 2020.

[BIP16] Peter Bürgisser, Christian Ikenmeyer, and Greta Panova. No occurrence obstructions in geometric complexity theory. In *57th Symposium on Foundations of Computer Science, FOCS*, pages 386–395, 2016.

[BR90] László Babai and Lajos Rónyai. Computing irreducible representations of finite groups. *Mathematics of Computation*, 55(192):705–722, 1990.

[BW15] Peter A Brooksbank and James B Wilson. The module isomorphism problem reconsidered. *Journal of Algebra*, 421:541–559, 2015.

[CFO+15] 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.

[CKW11] Xi Chen, Neeraj Kayal, and Avi Wigderson. Partial derivatives in arithmetic complexity and beyond. *Foundations and Trends in Theoretical Computer Science*, 6(1-2):1–138, 2011.

[CLS19] Suryajith Chillara, Nutan Limaye, and Srikanth Srinivasan. Small-depth multilinear formula lower bounds for iterated matrix multiplication with applications. *SIAM J. Comput.*, 48(1):70–92, 2019. Conference version appeared in the proceedings of STACS 2018.

[Ebe89] W. M. Eberly. *Computations for algebras and group representations*. PhD thesis, Department of Computer Science, University of Toronto, 1989.

[FGS19] Vyacheslav Futorny, Joshua A. Grochow, and Vladimir V. Sergeichuk. Wildness for tensors. *Linear Algebra and its Applications*, 566:212–244, 2019.

[FLMS15] Hervé Fournier, Nutan Limaye, Guillaume Malod, and Srikanth Srinivasan. Lower bounds for depth-4 formulas computing iterated matrix multiplication. *SIAM J. Comput.*, 44(5):1173–1201, 2015. Conference version appeared in the proceedings of STOC 2014.

[FP06] Jean-Charles Faugère and Ludovic Perret. Polynomial Equivalence Problems: Algorithmic and Theoretical Aspects. In Serge Vaudenay, editor, *International Conference on the Theory and Applications of Cryptographic Techniques, Advances in Cryptology - EUROCRYPT*, pages 30–47, 2006.

[Fro97] Georg Frobenius. Ueber die darstellung der endlichen gruppen durch linearc substitutionen. *Sitzungber. der Berliner Akademie*, 7:994–1015, 1897.

[FS13] Michael A. Forbes and Amir Shpilka. Quasipolynomial-time identity testing of noncommutative and read-once oblivious algebraic branching programs. In *54th Symposium on Foundations of Computer Science, FOCS 2013*, pages 243–252, 2013.

[Ges16] Fulvio Gesmundo. Geometric aspects of iterated matrix multiplication. *Journal of Algebra*, 461:42–64, 2016.

[GGKS19] Ankit Garg, Nikhil Gupta, Neeraj Kayal, and Chandan Saha. Determinant equivalence test over finite fields and over Q. In *46th International Colloquium on Automata, Languages, and Programming, ICALP 2019*, pages 62:1–62:15, 2019.

[GIP17] Fulvio Gesmundo, Christian Ikenmeyer, and Greta Panova. Geometric complexity theory and matrix powering. *Differential Geometry and its Applications*, 55:106–127, 2017.

[GQ19] Joshua A. Grochow and Youming Qiao. Isomorphism problems for tensors, groups, and cubic forms: completeness and reductions. *CoRR*, abs/1907.00309, 2019.

[Gro12] Joshua A. Grochow. *Symmetry and equivalence relations in classical and geometric complexity theory*. PhD thesis, The University of Chicago, 2012. Available from https://www.cs.colorado.edu/~jgrochow/grochow-thesis.pdf.

[GS86] Shafi Goldwasser and Michael Sipser. Private coins versus public coins in interactive proof systems. In *Proceedings of the 18th Symposium on the Theory of Computing, STOC 1986*, pages 59–68, 1986.

[IP16] Christian Ikenmeyer and Greta Panova. Rectangular kronecker coefficients and plethysms in geometric complexity theory. In *57th Symposium on Foundations of Computer Science, FOCS*, pages 396–405, 2016.

[IQ19] Gábor Ivanyos and Youming Qiao. Algorithms Based on *-Algebras, and Their Applications to Isomorphism of Polynomials with One Secret, Group Isomorphism, and Polynomial Identity Testing. *SIAM J. Comput.*, 48(3):926–963, 2019. Conference version appeared in the proceedings of SODA 2018.

[IRS12] Gábor Ivanyos, Lajos Rónyai, and Joseph Schicho. Splitting full matrix algebras over algebraic number fields. *Jounral of Algebra*, 354:211–223, 2012.

[Kay11] Neeraj Kayal. Efficient algorithms for some special cases of the polynomial equivalence problem. In *Proceedings of the 22nd Symposium on Discrete Algorithms, SODA 2011*, pages 1409–1421, 2011.

[Kay12] Neeraj Kayal. Affine projections of polynomials: extended abstract. In *Proceedings of the 44th Symposium on Theory of Computing, STOC 2012*, pages 643–662, 2012. Full text available from https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/Projection.pdf.

[KNS19] Neeraj Kayal, Vineet Nair, and Chandan Saha. Average-case linear matrix factorization and reconstruction of low width algebraic branching programs. *Computational Complexity*, 28(4):749–828, 2019.

[KNS20] Neeraj Kayal, Vineet Nair, and Chandan Saha. Separation between read-once oblivious algebraic branching programs (roabps) and multilinear depth-three circuits. *ACM Trans. Comput. Theory*, 12(1), 2020. Conference version appeared in the proceedings of STACS 2016.

[KNST19] Neeraj Kayal, Vineet Nair, Chandan Saha, and Sébastien Tavenas. Reconstruction of full rank algebraic branching programs. *TOCT*, 11(1):2:1–2:56, 2019. Conference version appeared in the proceedings of CCC 2017.

[KS03] Adam Klivans and Amir Shpilka. Learning arithmetic circuits via partial derivatives. In *Proceedings of the 16th Conference on Learning Theory, COLT 2003*, pages 463–476, 2003.

[KS15] Neeraj Kayal and Chandan Saha. Lower bounds for sums of products of low arity polynomials. *Electronic Colloquium on Computational Complexity (ECCC)*, 22:73, 2015.

[KS17] Mrinal Kumar and Shubhangi Saraf. On the Power of Homogeneous Depth 4 Arithmetic Circuits. *SIAM J. Comput.*, 46(1):336–387, 2017. Conference version appeared in the proceedings of FOCS 2014.

[KST18] Neeraj Kayal, Chandan Saha, and Sébastien Tavenas. On the size of homogeneous and of depth-four formulas with low individual degree. *Theory of Computing*, 14(1):1–46, 2018. Conference version appeared in the proceedings of STOC 2016.

[KT90] Erich Kaltofen and Barry M. Trager. Computing with Polynomials Given By Black Boxes for Their Evaluations: Greatest Common Divisors, Factorization, Separation of Numerators and Denominators. *J. Symb. Comput.*, 9(3):301–320, 1990. Conference version appeared in the proceedings of FOCS 1998.

[Lan15] J. M Landsberg. Geometric complexity theory: an introduction for geometers. *ANNALI DELL'UNIVERSITA' DI FERRARA*, 61(1):65–117, 2015.

[Lor08] Falko Lorenz. *Algebra Volumne 2: Fields with structures*. Algebras and advanced topics. Springer, 2008.

[MS01] 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.

[MS08] Ketan Mulmuley and Milind A. Sohoni. Geometric complexity theory II: towards explicit obstructions for embeddings among class varieties. *SIAM J. Comput.*, 38(3):1175–1206, 2008.

[MV97] Meena Mahajan and V. Vinay. Determinant: Combinatorics, algorithms, and complexity. *Chicago J. Theor. Comput. Sci.*, 1997, 1997.

[Nis91] Noam Nisan. Lower bounds for non-commutative computation (extended abstract). In *Proceedings of the 23rd Symposium on Theory of Computing, STOC 1991*, pages 410–418, 1991.

[NW97] Noam Nisan and Avi Wigderson. Lower Bounds on Arithmetic Circuits Via Partial Derivatives. *Computational Complexity*, 6(3):217–234, 1997.

[Pat96] Jacques Patarin. Hidden fields equations (HFE) and isomorphisms of polynomials (IP): two new families of asymmetric algorithms. In *International Conference on the Theory and Application of Cryptographic Techniques, Advances in Cryptology - EUROCRYPT*, pages 33–48, 1996.

[PGC98] Jacques Patarin, Louis Goubin, and Nicolas Courtois. Improved algorithms for isomorphisms of polynomials. In *International Conference on the Theory and Application of Cryptographic Techniques, Advances in Cryptology - EUROCRYPT*, pages 184–200, 1998.

[Raz09] Ran Raz. Multi-linear formulas for permanent and determinant are of super-polynomial size. *J. ACM*, 56(2):8:1–8:17, 2009. Conference version appeared in the proceedings of STOC 2004.

[Rón87] Lajos Rónyai. Simple algebras are difficult. In *Proceedings of the 19th Symposium on Theory of Computing, STOC 1987*, pages 398–408, 1987.

[Rón90] Lajos Rónyai. Computing the structure of finite algebras. *J. Symb. Comput.*, 9(3):355–373, 1990.

[Rón92] Lajos Rónyai. Algorithmic properties of maximal orders in simple algebras over Q. *Computational Complexity*, 2:225–243, 1992.

[Sap15] Ramprasad Saptharishi. A survey of lower bounds in arithmetic circuit complexity. *Github survey*, 2015.

[Sax06] Nitin Saxena. *Morphisms of rings and applications to complexity*. PhD thesis, Indian Institute of Technology, Kanpur, 2006.

[Ser73] Jean-Pierre Serre. *A Course in Arithmetic*. Springer-Verlag New York, 1973.

[SY10] Amir Shpilka and Amir Yehudayoff. Arithmetic circuits: A survey of recent results and open questions. *Foundations and Trends in Theoretical Computer Science*, 5(3-4):207–388, 2010.

[Thi98] Thomas Thierauf. The isomorphism problem for read-once branching programs and arithmetic circuits. *Chicago J. Theor. Comput. Sci.*, 1998, 1998.

[Val79] Leslie G. Valiant. Completeness classes in algebra. In *Proceedings of the 11th Symposium on Theory of Computing, STOC 1979*, pages 249–261, 1979.

[Wal13] Lars Ambrosius Wallenborn. Computing the hilbert symbol, quadratic form equivalence and integer factoring. Diploma thesis, 2013.

# A Preliminaries on algebraic branching programs and matrix products

**Set-multilinear polynomial**: A *set-multilinear monomial* in $\mathbf{x}_0, \ldots, \mathbf{x}_{d-1}$ variables has exactly one variable from $\mathbf{x}_k$, for all $k \in [0, d-1]$. The coefficient of a non set-multilinear monomial is zero in a *set-multilinear polynomial* in $\mathbf{x}_0, \ldots, \mathbf{x}_{d-1}$ variables.

The following definition is motivated from the the fact that monomials in Tr-IMM$_{w,d}$ correspond to a path in the $d$-layer graph capturing the matrix product $Q_0 \ldots Q_{d-1}$.

**Definition A.1** (Path monomial)**.** A set-multilinear monomial in $\mathbf{x}_0, \ldots, \mathbf{x}_{d-1}$ variables is called as a *path monomial* if it has a non-zero coefficient in the Tr-IMM$_{w,d}$ polynomial, and a set-multilinear monomial that is not a path monomial is called a *non-path monomial*.

**Linear matrices**: A matrix with entries as linear forms in $\mathbf{x}$ variables over $\mathbb{F}$ is called a linear matrix in $\mathbf{x}$ variables over $\mathbb{F}$. If $\mathbf{x}$, $\mathbb{F}$ are clear from the context, then it is simply called a linear matrix. If the linear forms in a linear matrix are linearly independent, then we say it is a *full-rank* linear matrix.

**Algebraic branching program (ABP)**: A $(w, d, n)$-ABP is a matrix product $Y_0 \cdot Y_1 \ldots Y_{d-1}$, where $Y_0$ and $Y_{d-1}$ are row and column linear matrices of size $w$, and $Y_k$ is a $w \times w$ linear matrix in $\mathbf{x}$ variables for $k \in [1, d-2]$. The polynomial computed by the ABP is the entry in the resulting $1 \times 1$ matrix. Note that in the general definition of an ABP the intermediate widths of matrices can vary, but throughout this article we work with uniform width ABPs unless stated otherwise. A *full-rank* ABP is a $(w, d, n)$-ABP where the $w^2(d-2) + 2w$ linear forms in its matrices are linearly independent. A *set-multilinear* ABP in $\mathbf{x}_0, \ldots, \mathbf{x}_{d-1}$ variables is a $(w, d, n)$-ABP where the linear forms in $Y_k$ are in $\mathbf{x}_k$ variables. The following fact is easily inferred.

**Fact 2.** *The Tr-IMM$_{w,d}$ polynomial is computed by a $(w^2, d, n)$-set-multilinear ABP $Y_0 \ldots Y_{d-1}$ in $\mathbf{x}_0, \ldots, \mathbf{x}_{d-1}$ variables, where $Y_0 = (Q_0(1,1), Q_0(1,2), \ldots Q_0(1,w), Q_0(2,1), \ldots, Q_0(w,w))$, $Y_k = I_w \otimes Q_k$ for $k \in [1, d-2]$, and $Y_{d-1} = (Q_{d-1}(1,1), Q_{d-1}(2,1), \ldots Q_{d-1}(w,1), Q_{d-1}(1,2), \ldots, Q_{d-1}(w,w))^T$.*

**Matrix Product**: A matrix product $X_0 \ldots X_{d-1}$, where $X_0, \ldots, X_{d-1}$ are $w \times w$ linear matrices is denoted as a $(w, d, n)$-matrix product. If the $w^2 d$ linear forms in the matrices of a $(w, d, n)$-matrix product are linearly independent then we say it is a *full-rank* $(w, d, n)$-matrix product. Additionally, if $X_k$ has linear forms in only $\mathbf{x}_k$ variables for $k \in [0, d-1]$ then we call it a $(w, d, n)$ set-multilinear matrix product in $\mathbf{x}_0, \ldots, \mathbf{x}_{d-1}$ variables.

**Set-multilinear ABP reconstruction**: Here, we note the main properties of the set-multilinear ABP reconstruction algorithm in [KS03] for set-multilinear ABPs with varying intermediate widths. A set-multilinear ABP $Y_0 \ldots Y_{d-1}$ in $\mathbf{x}_0, \ldots, \mathbf{x}_{d-1}$ variables has width-sequence $(w_0, \ldots, w_{d-2})$, if $Y_0$ is a row linear matrix of size $w_0$ in $\mathbf{x}_0$ variables, $Y_k$ is a $w_{k-1} \times w_k$ linear matrix in $\mathbf{x}_k$ variables for $k \in [1, d-2]$, and $Y_{d-1}$ is a column linear matrix of size $w_{d-2}$ in $\mathbf{x}_{d-1}$ variables. The next observation is proved using evaluation dimension (see Definition C.1), its proof is omitted here.

**Observation A.1.** *Suppose $f$ is a set-multilinear polynomial in $\mathbf{x}_0, \ldots, \mathbf{x}_{d-1}$ variables. Then, there is a set-multilinear ABP in $\mathbf{x}_0, \ldots, \mathbf{x}_{d-1}$ variables of width sequence $(w_0, \ldots, w_{d-2})$ computing $f$, such that any other set-multilinear ABP in $\mathbf{x}_0, \ldots, \mathbf{x}_{d-1}$ variables of width-sequence $(w'_0, \ldots, w'_{d-2})$ computing $f$ satisfies $w_k \leq w'_k$ for $k \in [0, d-2]$. Such a set-multilinear ABP in $\mathbf{x}_0, \ldots, \mathbf{x}_{d-1}$ variables of width-sequence $(w_0, \ldots, w_{d-2})$ computing $f$ is called a min-width set-multilinear ABP for $f$.*

Given blackbox access to a set-multilinear polynomial $f(\mathbf{x}_0, \ldots, \mathbf{x}_{d-1})$, the set-multilinear ABP reconstruction algorithm in [KS03] reconstructs a min-width set-multilinear ABP in $\mathbf{x}_0, \ldots, \mathbf{x}_{d-1}$ variables computing $f$ in randomized polynomial-time. Finally, the following observation regarding the relation between two min-width ABPs computing $f$ is easy to prove and its proof is omitted.

**Observation A.2.** *Suppose $f$ is a set-multilinear polynomial in $\mathbf{x}_0, \ldots, \mathbf{x}_{d-1}$ variables, and $Y_0 \ldots Y_{d-1}$ and $Y'_0 \ldots Y'_{d-1}$ are two min-width set-multilinear ABPs in $\mathbf{x}_0, \ldots, \mathbf{x}_{d-1}$ variables of width-sequence $(w_0, \ldots, w_{d-2})$ computing $f$. Then there are matrices $T_k \in GL(w_k, \mathbb{F})$ $k \in [0, d-2]$, such that $Y'_0 = Y_0 \cdot T_0$, $Y'_k = T_{k-1}^{-1} \cdot Y_k \cdot T_k$ for $k \in [1, d-2]$, and $Y'_{d-1} = T_{d-2}^{-1} \cdot Y_{d-1}$.*

# B Proofs from Section 3

*Claim 3.1 (restated): If $E \in \mathfrak{g}_{Tr\text{-}IMM}$ then $E$ is block-diagonal.*

*Proof.* Since $E \in \mathfrak{g}_{\text{Tr-IMM}}$, the entries of $E = (e_{i,j})_{i,j\in[n]}$ satisfy the following equation,

$$\sum_{i,j\in[n]} e_{ij} \cdot x_j \cdot \frac{\partial \text{Tr-IMM}}{\partial x_i} = 0 \ . \tag{4}$$

Equation 4 can be rewritten as follows

$$\underbrace{\sum_{\substack{x_i, x_j \in \mathbf{x}_k \\ k\in[0,d-1]}} e_{ij} \cdot x_j \cdot \frac{\partial \text{Tr-IMM}}{\partial x_i}}_{(a)} + \underbrace{\sum_{\substack{x_i \in \mathbf{x}_\ell \ x_j \in \mathbf{x}_k \\ \ell,k\in[0,d-1], \ell\neq k}} e_{ij} \cdot x_j \cdot \frac{\partial \text{Tr-IMM}}{\partial x_i}}_{(b)} = 0. \tag{5}$$

In Equation 5, term (a) corresponds to the *block-diagonal* entries of $E$ and term (b) corresponds to the *non block-diagonal* entries of $E$. Observe that the terms are monomial disjoint: monomials in term (a) have variables from each variable set $\mathbf{x}_0, \ldots, \mathbf{x}_{d-1}$, whereas monomials in term (b) have two variables from $\mathbf{x}_k$ and no variable from $\mathbf{x}_\ell$ for $\ell, k \in [0, d-1]$ and $\ell \neq k$. This implies terms (a) and (b) are individually equal to zero,

$$\sum_{\substack{x_i, x_j \in \mathbf{x}_k \\ k\in[0,d-1]}} e_{ij} \cdot x_j \cdot \frac{\partial \text{Tr-IMM}}{\partial x_i} = 0 \tag{6}$$

$$\sum_{\substack{x_i \in \mathbf{x}_\ell \ x_j \in \mathbf{x}_k \\ \ell,k\in[0,d-1], \ell\neq k}} e_{ij} \cdot x_j \cdot \frac{\partial \text{Tr-IMM}}{\partial x_i} = 0. \tag{7}$$

Additionally in Equation 7, for $x_i \in \mathbf{x}_\ell, x_j \in \mathbf{x}_k$, $x_{i'} \in \mathbf{x}_{\ell'}, x_{j'} \in \mathbf{x}_{k'}$ and $(\ell, k) \neq (\ell', k')$ the terms $x_j \frac{\partial \text{Tr-IMM}}{\partial x_i}$ and $x_{j'} \frac{\partial \text{Tr-IMM}}{\partial x_{i'}}$ are monomial disjoint. Thus for every pair $(\ell, k)$ such that $\ell \neq k$

$$\sum_{x_i \in \mathbf{x}_\ell \ x_j \in \mathbf{x}_k} e_{ij} \cdot x_j \cdot \frac{\partial \text{Tr-IMM}}{\partial x_i} = 0. \tag{8}$$

In Equation 8, group the coefficients of the term $\frac{\partial \text{Tr-IMM}}{\partial x_i}$ together and rewrite it as

$$\sum_{x_i \in \mathbf{x}_\ell} L_{x_i}^{(\ell,k)} \frac{\partial \text{Tr-IMM}}{\partial x_i} = 0, \tag{9}$$

where $L_{x_i}^{(\ell,k)}$ is a linear form in the $\mathbf{x}_k$ variables. Now we show that $L_{x_i}^{(\ell,k)} = 0$.

Let $x_i = x_{p,q}^{(\ell)}$ be the $(p,q)$-th entry of $Q_\ell$, where $p, q \in [w]$. Also let $Q'_\ell$ be a $w \times w$ matrix whose $(p,q)$-th entry is the linear form $L_{x_i}^{(\ell,k)}$. Then from Equation 9,

$$\sum_{x_i \in \mathbf{x}_\ell} L_{x_i}^{(\ell,k)} \frac{\partial \text{Tr-IMM}}{\partial x_i} \;=\; \text{tr}(Q_0 \dots Q'_\ell \dots Q_{d-1}) \;=\; 0\,. \tag{10}$$

Now suppose for contradiction $L_{x_i}^{(\ell,k)} \neq 0$. Then there is a $x_{u,v}^{(k)} \in \mathbf{x}_k$ such that the coefficient of $x_{u,v}^{(k)}$ in $L_{x_i}^{(\ell,k)}$ is non-zero. We argue for the cases $k \notin \{\ell-1, \ell+1\}$ and $k \in \{\ell-1, \ell+1\}$ separately. If $k \notin \{\ell-1, \ell+1\}$ then a path monomial $\mu$ can be chosen such that $\mu$ contains the variable $x_i = x_{p,q}^{(\ell)}$ and $x_{u,v}^{(k)}$. In Equation 10 set all the variables to zero except the variables appearing in $\mu$. Under this assignment the polynomial computed by $\text{tr}(Q_0 \dots Q'_\ell \dots Q_{d-1})$ is non-zero as the linear form $L_{x_i}^{(\ell,k)} \neq 0$, which is a contradiction. Now suppose $k = \ell - 1$. Then choose a path monomial $\mu$ containing the variables $x_i = x_{p,q}^{(\ell)}$ and $x_{u',p}^{(k)}$ where $u \neq u'$, and in Equation 10 set all the variables to zero except the variables appearing in $\mu$ and the variable $x_{u,v}^{(k)}$. Again under this assignment the polynomial computed by $\text{tr}(Q_0 \dots Q'_\ell \dots Q_{d-1})$ is non-zero as the linear form $L_{x_i}^{(\ell,k)} \neq 0$, which is a contradiction. For $k = \ell + 1$, choosing a path monomial $\mu$ containing the variables $x_i = x_{p,q}^{(\ell)}$ and $x_{q,v'}^{(k)}$ where $v \neq v'$ suffices. □

*Lemma 3.1 (restated): The space $\mathcal{B}_0 + \ldots + \mathcal{B}_{d-1}$ is contained in $\mathfrak{g}_{\text{Tr-IMM}}$.*

*Proof.* It is sufficient to prove that for every $k \in [0, d-1]$, $\mathcal{B}_k \subseteq \mathfrak{g}_{\text{Tr-IMM}}$. Let $k \in [0, d-2]$, $k$ even, and $B \in \mathcal{B}_k$. Then there is an $M \in \mathcal{M}_w$ such that

$$[B]_k = \begin{bmatrix} I_w \otimes M^T & \mathbf{0} \\ \mathbf{0} & -I_w \otimes M \end{bmatrix}\,.$$

Let $M = (m_{i,j})_{i,j \in [w]}$, and $\ell_{i,j}^{(k)} = \sum_{v \in [w]} m_{v,j} x_{i,v}^{(k)}$ and $\ell_{i,j}^{(k+1)} = \sum_{v \in [w]} -m_{i,v} x_{v,j}^{(k+1)}$ for all $i, j \in [w]$. Further, let $Q'_k = (\ell_{i,j}^{(k)})_{i,j \in [k]}$, and $Q'_{k+1} = (\ell_{i,j}^{(k+1)})_{i,j \in [w]}$.

**Observation B.1.** *The matrix $B \in \mathfrak{g}_{\text{Tr-IMM}}$ if and only if the following holds:*

$$\begin{aligned} \sum_{i,j \in [w]} \ell_{i,j}^{(k)} \frac{\partial \text{Tr-IMM}}{x_{i,j}^{(k)}} + \sum_{i,j \in [w]} \ell_{i,j}^{(k+1)} \frac{\partial \text{Tr-IMM}}{x_{i,j}^{(k+1)}} &= \text{tr}(Q_0 \cdots Q_{k-1}(Q'_k \cdot Q_{k+1} + Q_k \cdot Q'_{k+1}) Q_{k+2} \cdots Q_{d-1}) \\ &= 0\,. \end{aligned}$$

**Observation B.2.** *The matrices $Q'_k$ and $Q'_{k+1}$ are such that $Q'_k = Q_k \cdot M$ and $Q'_{k+1} = -M \cdot Q_{k+1}$.*

Thus, $Q'_k \cdot Q_{k+1} = -Q_k \cdot Q'_{k+1}$, and hence $B \in \mathfrak{g}_{\text{Tr-IMM}}$. The proofs for the remaining two cases: a) $k \in [0, d-1]$, $d$ even and $k$ odd, and b) $k = d-1$ and $d$ odd are similar. □

*Lemma 3.2 (restated): Suppose $B \in \mathfrak{g}_{Tr\text{-}IMM}$ and there is a $k \in [0, d-1]$ such that the non-zero entries of $B$ are confined to the rows and columns that are indexed by $\mathbf{x}_k$ and $\mathbf{x}_{k+1}$ variables. Then $B \in \mathcal{B}_k$.*

*Proof.* Let $\ell_{i,j}^{(k)}$ and $\ell_{i,j}^{(k+1)}$ be the linear forms whose coefficients are given by the row vectors indexed by $x_{i,j}^{(k)}$ and $x_{i,j}^{(k+1)}$ variables in $B$ respectively. From the structure of $B$ it follows that $\mathbf{x}_k$ and $\mathbf{x}_{k+1}$ are the only variables with non-zero coefficients in $\ell_{i,j}^{(k)}$ and $\ell_{i,j}^{(k+1)}$ respectively. Let $Q'_k = (\ell_{i,j}^{(k)})_{i,j\in[w]}$, and $Q'_{k+1} = (\ell_{i,j}^{(k+1)})_{i,j\in[w]}$. Since $B \in \mathfrak{g}_{\text{Tr-IMM}}$,

$$\text{tr}(Q_0 \dots Q'_k \cdot Q_{k+1} \dots Q_{d-1} + Q_0 \dots Q_k \cdot Q'_{k+1} \dots Q_{d-1}) = 0 \ . \tag{11}$$

**Observation B.3.** *Equation 11 implies $Q'_k \cdot Q_{k+1} + Q_k \cdot Q'_{k+1} = 0$*

*Proof.* The third line in the following sequence of equations follows from the fact that trace remains invariant under rotations.

$$\begin{aligned} &\text{tr}(Q_0 \dots Q'_k \cdot Q_{k+1} \dots Q_{d-1} + Q_0 \dots Q_k \cdot Q'_{k+1} \dots Q_{d-1}) \\ &= \text{tr}(Q_0 \dots Q_{k-1}(Q'_k \cdot Q_{k+1} + Q_k \cdot Q'_{k+1})Q_{k+2} \dots Q_{d-1}) \\ &= \text{tr}((Q'_k \cdot Q_{k+1} + Q_k \cdot Q'_{k+1})Q_{k+2} \dots Q_{d-1} \cdot Q_0 \dots Q_{k-1}) = 0 \,. \end{aligned}$$

Assign $0/1$ values to the variables in $\mathbf{x}_{k+3}, \dots, \mathbf{x}_{d-1}, \mathbf{x}_0, \dots, \mathbf{x}_{k-1}$ such that $Q_{k+3}, \dots, Q_{d-1}, Q_0, \dots, Q_{k-1}$ become identity matrices under this assignment. As the entries of $Q_{k+2}$ are distinct $\mathbf{x}_{k+2}$ variables, we have $Q'_k \cdot Q_{k+1} + Q_k \cdot Q'_{k+1} = 0$. □

By setting the $\mathbf{x}_{k+1}$ variables to $0/1$ so that $Q_{k+1}$ becomes identity in the equation $Q'_k \cdot Q_{k+1} + Q_k \cdot Q'_{k+1} = 0$, we get $Q'_k = Q_1 M$ for some $M \in \mathcal{M}_w$. Similarly, $Q'_{k+1} = NQ_{k+1}$. Thus, $Q_k M Q_{k+1} + Q_k N Q_{k+1} = 0$ which implies $N = -M$. At this point, the structure of $B$ can be determined using $M$ and then it is easily observed that $B \in \mathcal{B}_k$. □

**Claim B.1.** *There is a diagonal matrix in $\mathfrak{g}_{Tr\text{-}IMM}$ with distinct diagonal entries.*

*Proof.* For $k \in [0, d-1]$, let $D_k$ be a $w \times w$ diagonal matrix whose $i$-th diagonal entry is denoted as $d_i^{(k)}$. For $k \in [0, d-1]$ let $B_k \in \mathcal{B}_k$ be the diagonal matrix whose $2w^2 \times 2w^2$ sub-matrix indexed by $\mathbf{x}_k \uplus \mathbf{x}_{k+1}$ variables, denoted $[B_k]_k$, looks as follows: if $d$ is even then

$$\begin{aligned} [B_k]_k &= \begin{bmatrix} I_w \otimes D_k & \mathbf{0} \\ \mathbf{0} & -I_w \otimes D_k \end{bmatrix} \text{ if } k \text{ is even,} \\ &= \begin{bmatrix} D_k \otimes I_w & \mathbf{0} \\ \mathbf{0} & -D_k \otimes I_w \end{bmatrix} \text{ if } k \text{ is odd.} \end{aligned} \tag{12}$$

If $d$ is odd, then $B_0, \dots, B_{d-2}$ remain the same, and only $B_{d-1}$ is defined differently and in this case

$$[B_{d-1}]_{d-1} = \begin{bmatrix} I_w \otimes D_{d-1} & \mathbf{0} \\ \mathbf{0} & -D_{d-1} \otimes I_w \end{bmatrix} .$$

Suppose $B = \sum_{k=0}^{d-1} B_k$. Then $B$ is a diagonal matrix in $\mathfrak{g}_{\text{Tr-IMM}}$ (from Lemma 3.1) whose diagonal entry indexed by the variable $x_{i,j}^{(k)}$ is equal to $d_j^{(k)} - d_i^{(k-1)}$. If we pretend the entries of $D_0, \ldots, D_{d-1}$ to be formal variables, say $\mathbf{d}$ variables, then the $n$ diagonal entries of $B$ are $n$ distinct linear forms in $\mathbf{d}$ variables. Hence, if we assign values to the $\mathbf{d}$ variables uniformly at random from a set $S \subseteq \mathbb{F}$ such that $|S| \geq n^3$ then with non-zero probability $B$ has all diagonal entries distinct after the random assignment. □

**Lemma B.1.** *Let $E_1, \ldots, E_a$ be a basis of $\mathfrak{g}_{Tr\text{-}IMM}$ and $E = \sum_{i=1}^{a} r_i E_i$ where $r_i \in_r S \subset \mathbb{F}, |S| \geq n^4$. Then the characteristic polynomial of $E$ is square-free with probability $1 - o(1)$.*

*Proof.* If we treat $\mathbf{r} = \{r_1, \ldots, r_a\}$ as formal variables then the characteristic polynomial $h_r(x)$ of $E$ is a polynomial in $x$ with coefficients that are polynomial of degree at most $n$ in $\mathbf{r}$ variables. Observe that the discriminant of $h_r(x)$, denoted $\text{disc}(h_r(x))$ is a non-zero polynomial in the $\mathbf{r}$ variables of degree at most $2n^2$. This is because if $\text{disc}(h_r(x))$ is identically zero as polynomial in $\mathbf{r}$ variables then for every evaluation of $\mathbf{r}$ variables to field elements, $h_r(x)$ is not a square-free polynomial. This contradicts Claim B.1, as we can set the $\mathbf{r}$ variables appropriately such that $E$ is a diagonal matrix with distinct entries and $h_r(x)$ for such a setting is square-free. Since $\text{disc}(h_r(x))$ is not an identically zero polynomial in $\mathbf{r}$ variables and has degree less than $2n^2$, if we set the $\mathbf{r}$ variables independently and uniformly at random from $S \subseteq \mathbb{F}$, $|S| \geq 2n^3$ then with probability $1 - o(1)$ $\text{disc}(h_r(x)) \neq 0$, i.e., $h_r(x)$ is square-free. □

*Claim 3.2 (restated): Any non-zero $\mathfrak{g}_{Tr\text{-}IMM}$-invariant subspace is a coordinate subspace of $\mathbb{F}^n$.*

*Proof.* Let $\mathbf{u} = (u_1, \ldots, u_n) \in \mathcal{U}$, $S_{\mathbf{u}}$ be the set of non-zero coordinates of $\mathbf{u}$, that is $S_{\mathbf{u}} := \{j : u_j \neq 0 \text{ and } j \in [n]\}$, and $D$ be a diagonal matrix as in Claim B.1 with distinct diagonal entries $\lambda_1, \ldots, \lambda_n$. Then the vectors $\{(\lambda_1^i u_1, \ldots, \lambda_n^i u_n) \in \mathcal{U} \mid i \in [0, |S_{\mathbf{u}}| - 1]\}$ are $\mathbb{F}$-linearly independent. Hence for all $j \in S_{\mathbf{u}}$, $e_j \in \mathcal{U}$ .This implies $\mathcal{U}$ is a coordinate subspace. □

*Lemma 3.3 (restated): The only irreducible $\mathfrak{g}_{Tr\text{-}IMM}$-invariant subspaces of $\mathbb{F}^n$ are $\mathcal{U}_0, \ldots, \mathcal{U}_{d-1}$.*

*Proof.* It follows from Claim 3.1 that $\mathcal{U}_0, \ldots, \mathcal{U}_{d-1}$ are $\mathfrak{g}_{\text{Tr-IMM}}$-invariant subspaces. We show that $\mathcal{U}_k$ is irreducible for $k \in [0, d-1]$. Suppose $\mathcal{U}$ is a non-zero $\mathfrak{g}_{\text{Tr-IMM}}$-invariant subspace and $\mathcal{U} \subset \mathcal{U}_k$ for some $k \in [0, d-1]$. Then $\mathcal{U}$ is a coordinate subspace of $\mathbb{F}^n$ (from Claim 3.2). Let $e_x \in \mathbb{F}^n$ be the coordinate vector with one in the entry indexed by the variable $x \in \mathbf{x}$. Then to prove that $\mathcal{U} = \mathcal{U}_k$, it is sufficient to show that $e_x \in \mathcal{U}$ for all $x \in \mathbf{x}_k$. We show this when $d$ is even. (For $d$ odd the matrices $B_k$ and $B_{k-1}$ defined below need to be appropriately redefined for $k = d-1$ so that $B_{d-1} \in \mathcal{B}_{d-1}$ and $B_{d-2} \in \mathcal{B}_{d-2}$.) Let $1_w$ be the all ones $w \times w$ matrix. Define the matrices $B_k \in \mathcal{B}_k$ and $B_{k-1} \in \mathcal{B}_{k-1}$ as follows: If $k$ is odd then

$$[B_k]_k = \begin{bmatrix} 1_w \otimes I_w & 0 \\ 0 & -1_w \otimes I_w \end{bmatrix} \quad \text{and} \quad [B_{k-1}]_{k-1} = \begin{bmatrix} -I_w \otimes 1_w & 0 \\ 0 & I_w \otimes 1_w \end{bmatrix}.$$

If $k$ is even then

$$[B_k]_k = \begin{bmatrix} -I_w \otimes 1_w & 0 \\ 0 & I_w \otimes 1_w \end{bmatrix} \quad \text{and} \quad [B_{k-1}]_{k-1} = \begin{bmatrix} 1_w \otimes I_w & 0 \\ 0 & -1_w \otimes I_w \end{bmatrix}.$$

Consider the matrix $E = B_{k-1} + B_k$ in $\mathfrak{g}_{\text{Tr-IMM}}$. Since $\mathcal{U}$ is a coordinate subspace, there is a $y = x_{i,j}^{(k)} \in \mathbf{x}_k$ such that $e_y \in \mathcal{U}$. Observation B.4 follows from the structure of $E$ and Claim 3.2.

**Observation B.4.** *The entries of the vector $Ee_y$ indexed by the variables in $\{x_{i,1}^{(k)}, x_{i,2}^{(k)}, \ldots, x_{i,w}^{(k)}\}$ and $\{x_{1,j}^{(k)}, x_{2,j}^{(k)}, \ldots, x_{w,j}^{(k)}\}$ are one and hence the coordinate vectors corresponding to these variables are in $\mathcal{U}$.*

Applying Observation B.4 repeatedly we have that $e_x \in \mathcal{U}$ for all $x \in \mathbf{x}_k$. Hence, $\mathcal{U} = \mathcal{U}_k$ implying $\mathcal{U}_k$ is irreducible. Finally, we argue that $\mathcal{U}_0, \ldots, \mathcal{U}_{d-1}$ are the only irreducible $\mathfrak{g}_{\text{Tr-IMM}}$-invariant subspaces. Let $\mathcal{U}$ be an irreducible $\mathfrak{g}_{\text{Tr-IMM}}$-invariant subspace and hence a coordinate subspace of $\mathbb{F}^n$. Suppose $e_y \in \mathcal{U}$, where $y \in \mathbf{x}_k$ for some $k \in [0, d-1]$. Applying Observation B.4 repeatedly we have that $e_x \in \mathcal{U}$ for all $x \in \mathbf{x}_k$. Hence, $\mathcal{U}_k \subseteq \mathcal{U}$. Since $\mathcal{U}$ is irreducible, $\mathcal{U} = \mathcal{U}_k$. □

*Corollary 3.1 (restated): If $f = \text{Tr-IMM}_{w,d}(A\mathbf{x})$, where $A \in GL(n, \mathbb{F})$, then the only irreducible $\mathfrak{g}_f$-invariant subspaces of $\mathbb{F}^n$ are $A^{-1}\mathcal{U}_0, \ldots, A^{-1}\mathcal{U}_{d-1}$.*

*Proof.* This follows by observing that $\mathcal{U}$ is an irreducible $\mathfrak{g}_{\text{Tr-IMM}}$-invariant subspace if and only if $A^{-1}\mathcal{U}$ is an irreducible $\mathfrak{g}_f$-invariant subspace Since $\mathcal{U}_0, \ldots, \mathcal{U}_{d-1}$ are the only irreducible $\mathfrak{g}_{\text{Tr-IMM}}$-invariant subspaces, $A^{-1}\mathcal{U}_0, \ldots, A^{-1}\mathcal{U}_{d-1}$ are the only irreducible $\mathfrak{g}_f$-invariant subspaces. □

*Lemma 3.4 (restated): Let $\text{Tr-IMM}_{w,d} = \text{tr}(Q_0' \cdots Q_{d-1}')$, where $Q_0' \cdots Q_{d-1}'$ is a full-rank $(w, d, n)$-matrix product in $\mathbf{x}$ variables over $\mathbb{F}$. Then there are $C_0, \ldots, C_{d-1} \in GL(w, \mathbb{F})$ and $\ell \in [0, d-1]$ such that either $Q_k' = C_k \cdot Q_{\ell+k} \cdot C_{k+1}^{-1}$ for $k \in [0, d-1]$ or $Q_k' = C_k \cdot Q_{\ell-k}^T \cdot C_{k+1}^{-1}$ for $k \in [0, d-1]$.*

*Proof.* The proof of Lemma 3.4 uses the following observation, which is on the evaluation dimension (Definition C.1) of a polynomial expressed as the trace of a full-rank set-multilinear matrix product. The proof of Observation B.5 is similar to the proof of Observation C.1.

**Observation B.5.** *Let $f = tr(X_0 \ldots X_{d-1})$, where $X_0 \ldots X_{d-1}$ is a full-rank $(w, d, n)$ set-multilinear matrix product in $\mathbf{x}_0, \ldots, \mathbf{x}_{d-1}$ variables. Then a) for $k \in [0, d-1]$ and $k' \in \{k-1, k+1\}$ $\text{Evaldim}_{\mathbf{x}_k \uplus \mathbf{x}_{k'}}(f) = w^2$, and b) for $k \in [0, d-1]$ and $k' \in [0, d-1] \setminus \{k-1, k, k+1\}$ $\text{Evaldim}_{\mathbf{x}_k \uplus \mathbf{x}_{k'}}(f) = w^4$.*

Let $A \in \text{GL}(n, \mathbb{F})$ be such that the row of $A$ indexed by the $x_{i,j}^{(k)}$ variable determine the coefficients of the linear form in the $(i, j)$-th entry of $Q_k'$ for $i, j \in [w]$ and $k \in [0, d-1]$. Then $\text{Tr-IMM}_{w,d} = \text{Tr-IMM}_{w,d}(A\mathbf{x})$. Observation B.6 proves that $A$ is a block-diagonal matrix up to a rotation.

**Observation B.6.** *There is a permutation $\sigma$ of $[0, d-1]$ such that the non-zero entries of the rows of $A$ indexed by the $\mathbf{x}_k$ variables are confined to the columns of $A$ indexed by $\mathbf{x}_{\sigma(k)}$ variables. Further, there is an $\ell \in [0, d-1]$ such that either $\sigma(k) = \ell + k$ for $k \in [0, d-1]$ or $\sigma(k) = \ell - k$ for $k \in [0, d-1]$.*

*Proof.* By Lemma 3.1, the irreducible invariant subspaces of the Lie algebra of $\text{Tr-IMM}_{w,d}(A\mathbf{x})$ are $A^{-1}\mathcal{U}_0, \ldots, A^{-1}\mathcal{U}_{d-1}$. But the irreducible $\mathfrak{g}_{\text{Tr-IMM}}$-invariant subspaces are $\mathcal{U}_0, \ldots, \mathcal{U}_{d-1}$ (Lemma 3.3). Hence, there is a permutation $\sigma$ of $[0, d-1]$ such that $A^{-1}\mathcal{U}_k = \mathcal{U}_{\sigma(k)}$ for $k \in [0, d-1]$. Since $\mathcal{U}_k$ is the subspace spanned by the vectors whose non-zero entries are indexed by $\mathbf{x}_k$ variables, the non-zero entries of the columns of $A^{-1}$ indexed by the $\mathbf{x}_k$ variables are confined to the rows of $A^{-1}$ indexed by $\mathbf{x}_{\sigma(k)}$ variables. Consequently, the non-zero entries of the rows of $A$ indexed by the $\mathbf{x}_k$ variables are confined to the columns of $A$ indexed by $\mathbf{x}_{\sigma(k)}$ variables. Hence, $Q_0' \ldots Q_{d-1}'$ is a full-rank $(w, d, n)$ set-multilinear matrix product in $\mathbf{x}_{\sigma(0)}, \ldots, \mathbf{x}_{\sigma(d-1)}$ variables.
For $k, k' \in [0, d-1]$, if $k' \in \{k-1, k+1\}$ then $\text{Evaldim}_{\mathbf{x}_k \uplus \mathbf{x}_{k'}}(\text{Tr-IMM}_{w,d}) = w^2$, and if $k' \in [0, d-1] \setminus \{k-1, k, k+1\}$ then $\text{Evaldim}_{\mathbf{x}_k \uplus \mathbf{x}_{k'}}(f) = w^4$ (from Observation B.5). Let $\sigma(0) = \ell$. Then again using Observation B.5 either $\sigma(k) = \ell + k$ for $k \in [0, d-1]$, or $\sigma(k) = \ell - k$ for $k \in [0, d-1]$. □

Let $\sigma$ be as in Observation B.6. We assume that there is an $\ell \in [0, d-1]$ such that $\sigma(k) = \ell + k$ for $k \in [0, d-1]$ and prove that there are matrices $C_0, \ldots, C_{d-1}, D_0, \ldots, D_{d-1} \in \mathrm{GL}(w, \mathbb{F})$ and non-zero $\alpha_0, \ldots, \alpha_{d-1} \in \mathbb{F}$ such that $Q'_k = C_k \cdot Q_{\ell+k} \cdot D_k$ for $k \in [0, d-1]$, $D_{d-1}C_0 = \alpha_0 I_w$, $D_k \cdot C_{k+1} = \alpha_{k+1} I_w$ for $k \in [0, d-2]$, and $\prod_{k\in[0,d-1]} \alpha_k = 1$. Using a similar argument it can be shown that if $\sigma(k) = \ell - k$ for $k \in [0, d-1]$ then there are matrices $C_0, \ldots, C_{d-1}, D_0, \ldots, D_{d-1} \in \mathrm{GL}(w, \mathbb{F})$ and non-zero $\alpha_0, \ldots, \alpha_{d-1} \in \mathbb{F}$ such that $Q'_k = C_k \cdot Q^T_{\ell-k} \cdot D_k$ for $k \in [0, d-1]$, $D_{d-1}C_0 = \alpha_0 I_w$, $D_k \cdot C_{k+1} = \alpha_{k+1} I_w$ for $k \in [0, d-2]$, and $\prod_{k\in[0,d-1]} \alpha_k = 1$. For ease of exposition, we also assume that $\ell = 0$, and it can be easily verified that the arguments continue to hold for an arbitrary $\ell$. Notice that if $\ell = 0$ then $A$ is a block-diagonal matrix. Denote the block of $A$ indexed by $\mathbf{x}_k$ variables as $A_k$. The proof of the lemma is now almost complete using Observation B.7.

**Observation B.7.** *For $k \in [0, d-1]$, there are matrices $P_k, S_k \in GL(w, \mathbb{F})$ such that $A_k = (I_w \otimes P_k)(S_k \otimes I_w)$.*

*Proof.* Fix a $k \in [0, d-1]$ such that $k$ is even. We will show that there are matrices $P_k, S_k \in \mathrm{GL}(w, \mathbb{F})$ such that $A_k = (I_w \otimes P_k)(S_k \otimes I_w)$, and a similar argument shows that there are matrices $P_{k+1}, S_{k+1} \in \mathrm{GL}(w, \mathbb{F})$ such that $A_{k+1} = (I_w \otimes P_{k+1})(S_{k+1} \otimes I_w)$. Since $A$ is block-diagonal, $A^{-1}\mathcal{B}_k A = \mathcal{B}_k$ for all $k \in [0, d-1]$, from Lemma 3.2, and Fact 1. Hence, for every $M \in \mathcal{M}_w$ there is a unique $N \in \mathcal{M}_w$ such that

$$(I_w \otimes M)A_k = A_k(I_w \otimes N)\,.$$

Call the $w \times w$ sub-matrix of $A_k$ whose rows are indexed by $x^{(k)}_{i,1}, \ldots, x^{(k)}_{i,w}$, variables, and the columns are indexed by $x^{(k)}_{j,1}, , \ldots, x^{(k)}_{j,w}$, variables as $A_k(i, j)$. Note that for all $i, j \in [w]$ the following holds: $M \cdot A_k(i, j) = A_k(i, j) \cdot N$. Since this holds for any $M \in \mathcal{M}_w$, either $A_k(i, j)$ is invertible or $A_k(i, j)$ is the zero matrix for $i, j \in [w]$. Choose an $i, j \in [w]$ such that $A_k(i, j)$ is invertible, and let $P_k = A_k(i, j)$. Since $A_k$ is invertible there exists such an $i, j \in [w]$. Let $u, v \in [w]$ be such that $A_k(u, v)$ is invertible. Then for any $M \in \mathcal{M}_w$,

$$P_k^{-1} \cdot M \cdot P_k = A_k(u, v)^{-1} \cdot M \cdot A_k(u, v)\,.$$

Since the above holds for any $M \in \mathcal{M}_w$, there is a non-zero $s_{u,v} \in \mathbb{F}$ such that $A_k(u, v) = s_{u,v}P_k$. Let $S_k = (s_{u,v})_{u,v\in[w]}$, where $s_{i,j} = 1$, and for any $u, v \in [w]$ if $A_k(u, v)$ is zero then $s_{u,v} = 0$. It is easily observed that $A_k = (I_w \otimes P_k)(S_k \otimes I_w)$, and as $A_k$ is invertible $S_k$ is invertible. □

From Observation B.7, for $k \in [0, d-1]$ the following is true: if $k$ is even then $Q'_k = S_k \cdot Q_k \cdot P_k^T$, and if $k$ is odd then $Q'_k = P_k \cdot Q_k \cdot S_k^T$. For ease of notation, if $k$ is even then rename $P_k^T$ as $D_k$ and $S_k$ as $C_k$, and if $k$ is odd then rename $P_k$ as $C_k$ and $S_k^T$ as $D_k$. Hence for $k \in [0, d-1]$, $Q'_k = C_k \cdot Q_k \cdot D_k$. Now, observe the following

$$\begin{aligned}\mathrm{tr}(Q_0 \ldots Q_{d-1}) &= \mathrm{tr}(C_0 \cdot Q_0 \cdot D_0 \ldots C_{d-1} \cdot Q_{d-1} \cdot D_{d-1}) \\ &= \mathrm{tr}(D_{d-1} \cdot C_0 \cdot Q_0 \cdot D_0 \ldots C_{d-1} \cdot Q_{d-1})\end{aligned}$$

The last line line in the above equation follows from the fact that trace of a matrix product remains invariant under rotations. Since the entries of $Q_{d-1}$ are distinct variables disjoint from the variables in $Q_0, \ldots, Q_{d-2}$

$$Q_0 \ldots Q_{d-2} = D_{d-1} \cdot C_0 \cdot Q_0 \cdot D_0 \ldots Q_{d-2} \cdot D_{d-2} \cdot C_{d-1}\,.$$

Substitute $Q_k = (D_k \cdot C_{k+1})^{-1}$ for $k \in [2, d-2]$, and $Q_1 = (D_0 \cdot C_1)^{-1}(D_1 \cdot C_2)^{-1}$ in the above equation, and let $M = \prod_{k\in[0,d-2]}(D_k \cdot C_{k+1})^{-1}$. Then

$$Q_0 \cdot M = D_{d-1} \cdot C_0 \cdot Q_0 \,.$$

Since the entries of $Q_0$ are distinct variables, there is a non-zero $\alpha_0 \in \mathbb{F}$ such that $D_{d-1} \cdot C_0 = M = \alpha_0 I_w$. Similarly, it can be shown that there is a non-zero $\alpha_{k+1} \in \mathbb{F}$ such that $D_k \cdot C_{k+1} = \alpha_{k+1} I_w$ for $k \in [0, d-2]$. Moreover, as

$$\mathrm{tr}(Q_0 \ldots Q_{d-1}) = \mathrm{tr}(C_0 \cdot Q_0 \cdot D_0 \ldots C_{d-1} \cdot Q_{d-1} \cdot D_{d-1})$$

it follows that $\prod_{k\in[0,d-1]} \alpha_k = 1$. Finally, observe the following

$$Q'_k = \left( (\prod_{\ell\in[k+1,d-1]} \alpha_\ell) C_k \right) \cdot Q_k \cdot \left( (\prod_{\ell\in[k+1,d-1]} \alpha_\ell^{-1}) D_k \right) \quad \text{for } k \in [0, d-2].$$

Reusing symbols for ease of notation, rename $C_k$ as $(\prod_{\ell\in[k+1,d-1]} \alpha_\ell) C_k$, and $D_k$ as $(\prod_{\ell\in[k+1,d-1]} \alpha_\ell^{-1}) D_k$, and notice that $D_k = C_{k+1}^{-1}$ for $k \in [0, d-2]$, and $D_{d-1} = C_0^{-1}$. □

# C Reduction from TRACE to TRACE-TI

**Algorithm 4** Reduction from TRACE to TRACE-TI

INPUT: Blackbox access to an $n$-variate degree-$d$ polynomial $f$.
OUTPUT: $A' \in \mathrm{GL}(n, \mathbb{F})$ and $w \in \mathbb{N}$ such that $h(\mathbf{x}) = f(A'\mathbf{x})$ is a $d$-tensor in the variable sets $\mathbf{x}_0, \ldots, \mathbf{x}_{d-1}$ which is isomorphic to Tr-IMM$_{w,d}$.

Compute the irreducible $\mathfrak{g}_f$-invariant subspaces

1: Call Algorithm 5 on input $f$. Let $\{\mathcal{V}_0, \ldots, \mathcal{V}_{d-1}\}$ be the spaces returned by Algorithm 5. If Algorithm 5 returns 'No' then output 'No'.

Reduction to TRACE-TI

2: Call Algorithm 6 on input $\{\mathcal{V}_0, \ldots, \mathcal{V}_{d-1}\}$, and let $A' \in \mathrm{GL}(n, \mathbb{F})$ and $w \in \mathbb{N}$ be the output of Algorithm 6. If Algorithm 6 returns 'No' then output 'No'. Otherwise, return $A'$ and $w$.

Algorithm 4 is analysed by assuming that there is an $A \in \mathrm{GL}(n, \mathbb{F})$ satisfying $f = \text{Tr-IMM}_{w,d}(A\mathbf{x})$. The final PIT at the end of Algorithm 1 handles the case when $f$ is not equivalent to Tr-IMM$_{w,d}$. In Step 1, Algorithm 5 computes a set of bases of the irreducible $\mathfrak{g}_f$-invariant subspaces. Algorithm 6 in Step 2 uses the bases to compute an $A' \in \mathrm{GL}(n, \mathbb{F})$ and the $w \in \mathbb{N}$ such that $h(\mathbf{x}) = f(A'\mathbf{x})$ is a $d$-tensor in the variable sets $\mathbf{x}_0, \ldots, \mathbf{x}_{d-1}$ which is isomorphic to Tr-IMM$_{w,d}$.

## C.1 Computing the irreducible $\mathfrak{g}_f$-invariant subspaces

Algorithm 5 is similar to Algorithm 3 in [KNST19] which computes the irreducible invariant subspaces of the Lie algebra of a polynomial equivalent to IMM$_{w,d}$.

**Steps 1–4**: A basis of $\mathfrak{g}_f$ is computed using Lemma 2.2 in [KNST19] (also see [Kay12]). At Step 2, let $R \in \mathfrak{g}_{\text{Tr-IMM}}$ such that $R = A \cdot R' \cdot A^{-1}$. Since the matrices in $\mathfrak{g}_{\text{Tr-IMM}}$ are block-diagonal

**Algorithm 5** Computing the irreducible $\mathfrak{g}_f$-invariant subspaces

INPUT: Blackbox access to an $n$-variate degree-$d$ polynomial $f$.
OUTPUT: A set of bases of the irreducible $\mathfrak{g}_f$-invariant subspaces.

1: Compute a basis $\{F_1, \ldots, F_a\}$ of $\mathfrak{g}_f$ using Lemma 2.2 in [KNST19].
2: Pick a random element $R' = \sum_{i=1}^{a} r_i F_i \in \mathfrak{g}_f$, where $r_i \in_r S$ is chosen independently and uniformly at random from $S \subseteq \mathbb{F}$ for every $i \in [n-1]$, and $|S| = 2n^3$ .
3: Compute the characteristic polynomial $q(x)$ of $R'$.
4: If $q(x)$ is not square-free then output 'No'. Otherwise compute the irreducible factors of $q(x)$ over $\mathbb{F}$. Call the irreducible factors $p_1(x), \ldots, p_s(x)$.
5: Compute bases of the null spaces $\mathcal{N}'_1, \ldots, \mathcal{N}'_s$ of $p_1(R'), \ldots, p_s(R')$ respectively.
6: For every $i \in [s]$, pick a non-zero vector $\mathbf{v} \in \mathcal{N}'_i$ and compute a basis of the closure of $\mathbf{v}$ under the action of $\mathfrak{g}_f$ using Algorithm 4 in [KNST19].
7: Let $\mathcal{V}_1, \ldots, \mathcal{V}_s$ be the list of the closure spaces (here, we are identifying spaces with their bases). Remove duplicates from the list by comparing every pair of spaces and get the pruned list $\mathcal{V}_0, \ldots, \mathcal{V}_{d-1}$. If the number of distinct closure spaces is not equal to $d$, or the dimension of all the closure spaces are not the same then output 'No'. Else, output the list $\{\mathcal{V}_0, \ldots, \mathcal{V}_{d-1}\}$.

(Claim 3.1), $R$ is a block-diagonal matrix with individual blocks $R_0, \ldots, R_{d-1}$ as shown in Figure 2. The characteristic polynomial $q(x)$ computed at Step 3 is square-free with high probability (Lemma B.1). Note that $q(x) = \prod_{k\in[0,d-1]} q_k(x)$, where $q_k(x)$ is the characteristic polynomial of $R_k$. At Step 4, the algorithm invokes a univariate polynomial factorization algorithm over $\mathbb{F}$. Observe that every irreducible factor $p_i(x)$ of $q(x)$ is a factor of $q_k(x)$ for some $k \in [0, d-1]$.

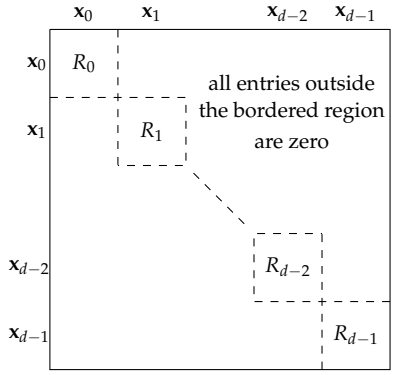


Figure 2: A random matrix $R \in \mathfrak{g}_{\text{Tr-IMM}}$

**Step 5–7**: Let $\mathcal{N}_i$ and $\mathcal{N}'_i$ be the null spaces of $p_i(R)$ and $p_i(R')$ respectively. Then $\mathcal{N}_i = A\mathcal{N}'_i$.

**Lemma C.1.** *Let $p_i(x)$ be an irreducible factor of $q_k(x)$, and $\mathbf{v} \in \mathcal{N}'_i$ be a non-zero vector. Then, the closure of $\mathbf{v}$ under the action of $\mathfrak{g}_f$ is the irreducible $\mathfrak{g}_f$-invariant subspace $A^{-1}\mathcal{U}_k$. Thus, at the end of Step 7 there is a permutation $\sigma$ of $[0, d-1]$ such that $\mathcal{V}_k = A^{-1}\mathcal{U}_{\sigma(k)}$ for all $k \in [0, d-1]$.*

*Proof.* Consider the following claim.

**Claim C.1.** $\mathcal{N}_i' \subseteq A^{-1}\mathcal{U}_k$.

The proofs of Lemma 3.3 and Corollary 3.1 in fact show that no $\mathfrak{g}_f$-invariant subspace is properly contained in $A^{-1}\mathcal{U}_k$. Observe that the closure of a vector under the action of $\mathfrak{g}_f$ is a $\mathfrak{g}_f$-invariant subspace by definition. Hence, by the above claim, the closure of $\mathbf{v}$ under the action of $\mathfrak{g}_f$ is $A^{-1}\mathcal{U}_k$.

*Proof of Claim C.1.* It is sufficient to show that $\mathcal{N}_i \subseteq \mathcal{U}_k$. Let $\mathbf{u} \in \mathcal{N}_i$. Let $\mathbf{u}_\ell \in \mathbb{F}^{w^2}$ be the vector obtained by restricting $\mathbf{u}$ to the entries that are indexed by $\mathbf{x}_\ell$ variables for $\ell \in [0, d-1]$. The matrix $q_k(R)$ is block-diagonal with blocks $q_k(R_0), \ldots, q_k(R_{d-1})$. Since $\mathbf{u} \in \mathcal{N}_i$, $p_i(R) \cdot \mathbf{u} = 0$ and so $q_k(R) \cdot \mathbf{u} = 0$. Hence,

$$q_k(R_\ell) \cdot \mathbf{u}_\ell = 0 \quad \text{for all } \ell \in [0, d-1]. \tag{13}$$

Further,

$$q_\ell(R_\ell) \cdot \mathbf{u}_\ell = 0 \quad \text{for all } \ell \in [0, d-1], \tag{14}$$

as $q_\ell(R_\ell) = 0$ (the characteristic polynomial of $R_\ell$ being $q_\ell(x)$). Since $q_k(x)$ and $q_\ell(x)$ are co-prime for $k \neq \ell$, there are polynomials $s(x)$ and $t(x)$ such that $s(x)q_k(x) + t(x)q_\ell(x) = 1$. This implies $s(R_\ell)q_k(R_\ell) + t(R_\ell)q_\ell(R_\ell) = I_{w^2}$. Hence, $s(R_\ell)q_k(R_\ell)\mathbf{u}_\ell + t(R_\ell)q_\ell(R_\ell)\mathbf{u}_\ell = \mathbf{u}_\ell$. From Equations 13 and 14, $\mathbf{u}_\ell = \mathbf{0}$ for all $k \neq \ell$. □

## C.2 Reduction to TRACE-TI

**Algorithm 6** Reduction to TRACE-TI

INPUT: The irreducible $\mathfrak{g}_f$-invariant subspaces $\mathcal{V}_0, \ldots, \mathcal{V}_{d-1}$.
OUTPUT: $A' \in \mathrm{GL}(n, \mathbb{F})$ and $w \in \mathbb{N}$ such that $h(\mathbf{x}) = f(A'\mathbf{x})$ is a $d$-tensor in the variable sets $\mathbf{x}_0, \ldots, \mathbf{x}_{d-1}$ which is isomorphic to Tr-IMM$_{w,d}$.

1: Determine $w$ such that $w^2$ is the dimension of each of the spaces $\mathcal{V}_0, \ldots, \mathcal{V}_{d-1}$. If there does not exist such a $w$ then output 'No'.
2: Construct the $n \times n$ matrix $V$ such that the $kw^2 + 1, \ldots, (k+1)w^2$ columns of $V$ are the basis vectors of $\mathcal{V}_k$ for $k \in [0, d-1]$.
3: Compute a permutation $\tau$ of $[0, d-1]$ that is equal to $\sigma^{-1}$ (up to a "rotation"), where $\sigma$ is the permutation in Lemma C.1.
4: Compute a block-permuted permutation matrix $B$ that maps the variables in $\mathbf{x}_{\tau(k)}$ to the variables in $\mathbf{x}_k$ for all $k \in [0, d-1]$, i.e., $B \cdot (\mathbf{x}_{\tau(0)}\ \mathbf{x}_{\tau(1)} \ldots \mathbf{x}_{\tau(d-1)})^T = (\mathbf{x}_0\ \mathbf{x}_1 \ldots \mathbf{x}_{d-1})^T$.
5: Return $A' = V \cdot B$ and $w$.

**Definition C.1** (Evaluation dimension [FS13, Nis91])**.** Let $g(\mathbf{x})$ be an $n$-variate polynomial and $\mathbf{x}' \subseteq \mathbf{x}$. Let $g(\mathbf{x})_{\mathbf{x}'=\alpha}$ denote the partial evaluation of $g$ at $\mathbf{x}' = \alpha \in \mathbb{F}^{|\mathbf{x}'|}$. The evaluation dimension of $g$ with respect to $\mathbf{x}'$ is defined as $\mathrm{Evaldim}_{\mathbf{x}'}(g) := \dim(\mathrm{span}_{\mathbb{F}}(\{g(\mathbf{x})_{\mathbf{x}'=\alpha} : \alpha \in \mathbb{F}^{|\mathbf{x}'|}\}))$ .

We use the above definition to analyse Algorithm 6.

**Steps 1–2**: The correctness of Step 1 follows from Corollary 3.1. Let $V_k$ be the $n \times w^2$ matrix whose columns are the basis vectors of the $\mathfrak{g}_f$-invariant subspace $\mathcal{V}_k$. Then the matrix $V$ constructed at Step 2 is obtained by concatenating the matrices $V_0, \ldots, V_{d-1}$ in this order, denoted

$V_0|V_1|\ldots|V_{d-2}|V_{d-1}$. From Lemma C.1, there is a permutation $\sigma$ of $[0,d-1]$ such that $\mathcal{V}_k = A^{-1}\mathcal{U}_{\sigma(k)}$. Hence, there is a matrix $E_k \in \mathbb{F}^{n\times w^2}$ such that $V_k = A^{-1}E_k$ and the non-zero entries of $E_k$ are confined to the rows indexed by $\mathbf{x}_{\sigma(k)}$ variables. Let $E = E_0|\ldots|E_{d-1}$. Then $V = A^{-1}\cdot E$. Observe that $E$ is a block-permuted matrix, i.e., the columns indexed by $\mathbf{x}_k$ variables have non-zero entries confined to the rows indexed by $\mathbf{x}_{\sigma(k)}$ variables. Thus, $g(\mathbf{x}) := f(V\mathbf{x}) = \text{Tr-IMM}_{w,d}(E\mathbf{x})$.

**Steps 3–5**: Step 3 uses the algorithm in next claim to determine $\tau$.

**Claim C.2.** *There is a randomized polynomial-time algorithm that takes input blackbox access to $g$ and with probability $1-o(1)$ outputs a permutation $\tau$ of $[0,d-1]$ such that there is an $\ell \in [0,d-1]$ satisfying either a) $\tau(k) = \sigma^{-1}(\ell+k)$ for all $k\in[0,d-1]$, or b) $\tau(k) = \sigma^{-1}(\ell-k)$ for all $k\in[0,d-1]$.*

The claim is proved below after completing the analysis of Algorithm 6. Assume that $\tau(k) = \sigma^{-1}(\ell+k)$ for all $k\in[0,d-1]$; the analysis for $\tau(k) = \sigma^{-1}(\ell-k)$ for all $k\in[0,d-1]$ is similar. Since $g = \text{Tr-IMM}_{w,d}(E\mathbf{x})$, there is a full-rank $(w,d,n)$-set-multilinear matrix product $X_0\cdots X_{d-1}$ in the variable sets $\mathbf{x}_{\sigma^{-1}(0)},\ldots,\mathbf{x}_{\sigma^{-1}(d-1)}$ respectively such that $g(\mathbf{x}) = \text{tr}(X_0\cdots X_{d-1}) = \text{tr}(X_\ell\cdot X_{\ell+1}\ldots X_{d-1}\cdot X_0\ldots X_{\ell-1})$. Renaming $X_{\ell+k}$ as $X_k$ for all $k\in[0,d-1]$ and reusing symbols, it is inferred that there is a full-rank $(w,d,n)$-set-multilinear matrix product $X_0\cdots X_{d-1}$ in the variable sets $\mathbf{x}_{\tau(0)},\ldots,\mathbf{x}_{\tau(d-1)}$ respectively such that $g = \text{tr}(X_0\cdots X_{d-1})$. Hence, at Steps 4 and 5, it is readily seen that $g(B\mathbf{x}) = f(VB\mathbf{x})$ is computed by a full-rank $(w,d,n)$-set-multilinear matrix product $X'_0\cdots X'_{d-1}$ in the variable sets $\mathbf{x}_0,\ldots,\mathbf{x}_{d-1}$ respectively, i.e., $f(VB\mathbf{x}) = \text{tr}(X'_0\cdots X'_{d-1})$.

*Proof of Claim C.2.* The following observation is the key to computing $\tau$.

**Observation C.1.** *Let $\ell\in[0,d-1]$ and $r = \sigma^{-1}(\ell)$. Then a) for $r'\in\{\sigma^{-1}(\ell-1),\sigma^{-1}(\ell+1)\}$, $\text{Evaldim}_{\mathbf{x}_r\uplus\mathbf{x}_{r'}}(g) = w^2$, and b) for $r'\in[0,d-1]\setminus\{\sigma^{-1}(\ell),\sigma^{-1}(\ell+1),\sigma^{-1}(\ell-1)\}$, $\text{Evaldim}_{\mathbf{x}_r\uplus\mathbf{x}_{r'}}(g) = w^4$. There is a randomized polynomial-time algorithm to compute $\text{Evaldim}_{\mathbf{x}_r\uplus\mathbf{x}_{r'}}(g)$ for all $r,r'\in[0,d-1]$.*

The observation is proved after completing the proof of the claim. Observation C.1 is used $\binom{d}{2}$ times to determine $S_r = \{\sigma^{-1}(\ell-1),\sigma^{-1}(\ell+1)\}$ where $r=\sigma^{-1}(\ell)$, for every $r\in[0,d-1]$. Using the knowledge of $S_0,\ldots,S_{d-1}$, $\tau$ is determined which is equal to $\sigma^{-1}$ up to a rotation as follows. Choose an arbitrary element $r\in[0,d-1]$ and set $\tau(0) = r$, and let $\ell\in[0,d-1]$ be such that $\sigma^{-1}(\ell) = r$. We can construct $\tau$ by choosing either of the elements in $S_r = \{\sigma^{-1}(\ell-1),\sigma^{-1}(\ell+1)\}$: if $\sigma^{-1}(\ell-1)$ is chosen then $\tau$ constructed will be such that $\tau(k) = \sigma^{-1}(\ell-k)$ for $k\in[0,d-1]$, and if $\sigma^{-1}(\ell+1)$ is chosen then $\tau$ constructed will be such that $\tau(k) = \sigma^{-1}(\ell+k)$ for $k\in[0,d-1]$. Without loss of generality assume $\sigma^{-1}(\ell+1)$ is chosen. Set $\tau(1) = \sigma^{-1}(\ell+1)$. The remaining part of $\tau$ is determined sequentially as follows. Suppose, for some $k\in[0,d-2]$, $\tau(i) = \sigma^{-1}(\ell+i)$ for all $i\in[0,k]$. Then $S_{\sigma^{-1}(\ell+k)} = \{\sigma^{-1}(\ell+k-1),\sigma^{-1}(\ell+k+1)\}$ and $\tau(k-1) = \sigma^{-1}(\ell+k-1)$. Choose the other element in $S_{\sigma^{-1}(\ell+k)}$ and set $\tau(k+1) = \sigma^{-1}(\ell+k+1)$. □

*Proof of Observation C.1.* There is a full-rank $(w,d,n)$-set-multilinear matrix product $X_0\cdots X_{d-1}$ in the variable sets $\mathbf{x}_{\sigma^{-1}(0)},\ldots,\mathbf{x}_{\sigma^{-1}(d-1)}$ respectively such that $g(\mathbf{x}) = \text{tr}(X_0\cdots X_{d-1})$.
*Case a*: Suppose $r' = \sigma^{-1}(\ell+1)$; the proof for $r' = \sigma^{-1}(\ell-1)$ is similar. Let $\mathcal{A} = \text{span}_{\mathbb{F}}\{g(\mathbf{x})|_{\mathbf{x}_r\uplus\mathbf{x}_{r'}=\alpha}, \alpha\in\mathbb{F}^{2w^2}\}$. Observe that $g(\mathbf{x}) = \text{tr}(X_0\ldots X_{d-1}) = \text{tr}(X_\ell\cdot X_{\ell+1}\ldots X_{d-1}\cdot X_0\ldots X_{\ell-1})$. Let $Y_1$ be the row vector of size $w^2$ whose $((i-1)\cdot w+j)$-th entry is the $(i,j)$-th entry of $X_\ell\cdot X_{\ell+1}$, for $i,j\in[w]$. Similarly, let $Y_2$ be the column vector of size $w^2$ whose $((j-1)\cdot w+i)$-th entry is the $(i,j)$-th entry of $X_{\ell+2}\ldots X_{d-1}\cdot X_0\ldots X_{\ell-1}$, for $i,j\in[w]$. From the construction of $Y_1$ and $Y_2$, $g(\mathbf{x}) = Y_1\cdot Y_2$.

Since $X_\ell, X_{\ell+1}$ are full-rank linear matrices in disjoint variable sets, there is a point $\alpha_i \in \mathbb{F}^{2w^2}$ such that the $i$-th entry of $Y_1$ evaluated at this point is equal to one and the remaining entries are zero, for every $i \in [w^2]$. Hence, every entry of $Y_2$ is in $\mathcal{A}$, and further as $X_{\ell+1} \dots X_{d-1} \cdot X_0 \dots X_{\ell-1}$ is a full-rank set-multilinear matrix product, the $w^2$ entries of $Y_2$ are $\mathbb{F}$-linearly independent. Thus the entries of $Y_2$ form a basis of $\mathcal{A}$, and $\text{Evaldim}_{\mathbf{x}_r \uplus \mathbf{x}_{r'}}(g) = \dim(\mathcal{A}) = w^2$.

*Case b*: Suppose $r' = \sigma^{-1}(\ell')$ and $\ell' \notin \{\ell-1, \ell, \ell+1\}$. Let $\mathcal{A} = \text{span}_{\mathbb{F}}\{g(\mathbf{x})|_{\mathbf{x}_r \uplus \mathbf{x}_{r'} = \alpha}, \alpha \in \mathbb{F}^{2w^2}\}$. Again $g(\mathbf{x}) = \text{tr}(X_\ell \cdot X_{\ell+1} \dots X_{\ell'} \dots X_{\ell-1})$. Let $P = X_{\ell+1} \dots X_{\ell'-1} = (p_{i,j})_{i,j \in [w]}$ and $T = X_{\ell'+1} \dots X_{\ell-1} = (t_{i,j})_{i,j \in [w]}$. Since $X_0 \dots X_{d-1}$ is a full-rank set-multilinear matrix product, the $w^4$ polynomials $\{p_{i_1,j_1} \cdot t_{i_2,j_2} \mid i_1, j_1, i_2, j_2 \in [w]\}$ are linearly independent over $\mathbb{F}$. Moreover, $\mathbf{x}_r \uplus \mathbf{x}_{r'}$ can be substituted appropriately such that these $w^4$ polynomials are in $\mathcal{A}$. Since $\mathcal{A} = \text{span}_{\mathbb{F}}\{p_{i_1,j_1} \cdot t_{i_2,j_2} \mid i_1, j_1, i_2, j_2 \in [w]\}$, the $w^4$ polynomials $\{p_{i_1,j_1} \cdot t_{i_2,j_2} \mid i_1, j_1, i_2, j_2 \in [w]\}$ form a basis of $\mathcal{A}$. This implies $\text{Evaldim}_{\mathbf{x}_r \uplus \mathbf{x}_{r'}}(g) = \dim(\mathcal{A}) = w^4$.

*A polynomial-time randomized procedure to compute* $\text{Evaldim}_{\mathbf{x}_r \uplus \mathbf{x}_{r'}}(g)$: Let $S \subset \mathbb{F}$ such that $|S| = n^4$. Choose points $\mathbf{a}_1, \dots, \mathbf{a}_{w^4} \in_r S^{2w^2}$ independently and uniformly at random and output the dimension of the $\mathbb{F}$-linear space spanned by the polynomials $g(\mathbf{x} \setminus \{\mathbf{x}_r, \mathbf{x}_{r'}\}, \mathbf{a}_1), \dots, g(\mathbf{x} \setminus \{\mathbf{x}_r, \mathbf{x}_{r'}\}, \mathbf{a}_{w^4})$ using Claim 2.2 in [KNST19]. The proof of correctness of this procedure is similar to the proof of correctness of the randomized procedure in Observation E.1 in [KNST19]. □

# D Proofs from Section 4

*Claim 4.1 (restated): Let $X$ be a $w \times w$ full-rank linear matrix and $Y = I_w \otimes X$. Then there does not exist non-zero matrices $T, S \in \mathcal{M}_{w^2}(\mathbb{F})$ such that $T \cdot Y = Y^T \cdot S$.*

*Proof.* Since $X$ is a full-rank linear matrix, by applying an invertible transformation we may assume without loss of generality that the entries of $X$ are distinct $w^2$ variables. Hence, it is sufficient to prove the claim when $X$ is symbolic matrix with entries being distinct variables. Suppose for contradiction, there are non-zero matrices $T$ and $S$ such that $T, S \in \mathcal{M}_{w^2}(\mathbb{F})$ and $T \cdot Y = Y^T \cdot S$. Let $T_{i,j}$ (respectively $S_{i,j}$) denote the $(i,j)$-th $w \times w$ sub-matrix of $T$ (respectively $S$) corresponding to the rows numbered from $(w(i-1)+1)$ to $(wi)$, and columns numbered from $(w(j-1)+1)$ to $(wj)$ of $T$ (respectively $S$), for $i, j \in [w]$. Then $T_{i,j} \cdot X = X^T \cdot S_{i,j}$, for every $i, j \in [w]$. For $u \in [2, w]$, observe that the $(1,u)$ entries of $T_{i,j} \cdot X$ and $X^T \cdot S_{i,j}$ are variable disjoint implying that all the columns except the first column of of $S_{i,j}$ are zero columns for every $i, j \in [w]$. Similarly comparing the $(2,1)$ entries of $T_{i,j} \cdot X$ and $X^T \cdot S_{i,j}$, it is observed that even the first column of $S_{i,j}$ is a zero column for every $i, j \in [w]$. This implies $S$ is a zero matrix, and hence $T$ is a zero matrix. □

*Claim 4.2 (restated): Let $X$ be a $w \times w$ full-rank linear matrix and $Y = I_w \otimes X$, and suppose $T, S \in \mathcal{M}_{w^2}(\mathbb{F})$ such that $T \cdot Y = Y \cdot S$. Then $T = S = M \otimes I_w$ for some $M \in \mathcal{M}_w(\mathbb{F})$.*

*Proof.* Similar to the proof of Claim 4.1, it is sufficient to prove Claim 4.2 for the case when $X$ is a $w \times w$ symbolic matrix with entries being distinct variables. Let $T, S \in \mathcal{M}_{w^2}(\mathbb{F})$ be such that $T \cdot Y = Y \cdot S$. Also let $\mathbf{a} \in \mathbb{F}^{w^2}$ be such that $X$ evaluated at $\mathbf{a}$ is equal to $I_w$. Now evaluating the expression $T \cdot Y = Y \cdot S$ at $\mathbf{a}$, it is inferred that $T = S$. Let $T_{i,j}$ denote the $(i,j)$-th $w \times w$ sub-matrix of $T$ corresponding to the rows numbered from $(w(i-1)+1)$ to $(wi)$, and columns numbered from $(w(j-1)+1)$ to $(wj)$ of $T$, for $i, j \in [w]$. Then $T_{i,j} \cdot X = X \cdot T_{i,j}$, for every $i, j \in [w]$. Since

the entries of $X$ are distinct variables, $T_{i,j} = m_{i,j} I_w$, where $m_{i,j} \in \mathbb{F}$. Hence $T = M \otimes I_w$, where $M = (m_{i,j})_{i,j \in [w]}$. □

*Observation 4.1 (restated): If $h(\mathbf{x}_0, \ldots, \mathbf{x}_{d-1})$ is isomorphic to Tr-IMM$_{w,d}$ then for matrices $Y'_k$ and $Z_k$ as computed in Algorithm 2, where $k \in [1, d-2]$, there are no matrices $T'_{k-1}, S'_k \in GL(w^2, \mathbb{F})$ such that both $T'_{k-1} \cdot Y'_k = Z_k \cdot S'_k$ and $T'_{k-1} \cdot Y'_k = Z_k^T \cdot S'_k$ are simultaneously true.*

*Proof.* Since $h(\mathbf{x})$ is multilinearly equivalent to Tr-IMM$_{w,d}$, as argued in Section 4.1 for all $k \in [1, d-2]$, $Z_k = I_w \otimes X'_k$, and $Y'_k = T_{k-1}^{-1} \cdot Y_k \cdot T_k$. Moreover, either

$$Y_k = (I_w \otimes C_k) \cdot Z_k \cdot (I_w \otimes D_k)] \quad \text{or} \quad Y_k = (I_w \otimes C_k) \cdot Z_k^T \cdot (I_w \otimes D_k).$$

We prove the observation when $Y_k = (I_w \otimes C_k) \cdot Z_k \cdot (I_w \otimes D_k)$, and the proof for $Y_k = (I_w \otimes C_k) \cdot Z_k^T \cdot (I_w \otimes D_k)$ is similar. Suppose there are matrices $T'_{k-1}, S'_k \in \mathrm{GL}(w^2, \mathbb{F})$ such that both $T'_{k-1} \cdot Y'_k = Z_k \cdot S'_k$ and $T'_{k-1} \cdot Y'_k = Z_k^T \cdot S'_k$ are simultaneously true. Then Equations 15 and 16 are simultaneously true.

$$\begin{aligned} T'_{k-1} \cdot (T_{k-1}^{-1} \cdot Y_k \cdot T_k) &= Z_k \cdot S'_k \\ (T'_{k-1} T_{k-1}^{-1}) \cdot (I_w \otimes C_k) \cdot Z_k \cdot (I_w \otimes D_k) \cdot T_k &= Z_k \cdot S'_k \\ (T'_{k-1} T_{k-1}^{-1}) \cdot (I_w \otimes C_k) \cdot Z_k &= Z_k \cdot (S'_k T_k^{-1}) \cdot (I_w \otimes D_k^{-1}) \end{aligned} \tag{15}$$

$$\begin{aligned} T'_{k-1} \cdot (T_{k-1}^{-1} \cdot Y_k \cdot T_k) &= Z_k^T \cdot S'_k \\ (T'_{k-1} T_{k-1}^{-1}) \cdot (I_w \otimes C_k) \cdot Z_k \cdot (I_w \otimes D_k) \cdot T_k &= Z_k^T \cdot S'_k \\ (T'_{k-1} T_{k-1}^{-1}) \cdot (I_w \otimes C_k) \cdot Z_k &= Z_k^T \cdot (S'_k T_k^{-1}) \cdot (I_w \otimes D_k^{-1}) \end{aligned} \tag{16}$$

Since $X'_k$ is a full-rank linear matrix in $\mathbf{x}_k$ variables and $Z_k = I_w \otimes X'_k$, this contradicts Claim 4.1. □

*Observation 4.2 (restated): The matrices $T'_{k-1}$ and $S'_k$ computed at Step 5 of Algorithm 2, where $k \in [1, d-2]$, satisfy the following: $(T'_{k-1})^{-1} = T_{k-1}^{-1} \cdot (I_w \otimes C_k) \cdot (M_k^{-1} \otimes I_w)$ and $S'_k = (M_k \otimes I_w) \cdot (I_w \otimes D_k) \cdot T_k$, where $M_k \in GL(w, \mathbb{F})$.*

*Proof.* Substitute $Y'_k = T_{k-1}^{-1} \cdot (I_w \otimes C_k) \cdot Z_k \cdot (I_w \otimes D_k) \cdot T_k$ in $T'_{k-1} \cdot Y'_k = Z_k \cdot S'_k$. Then

$$T'_{k-1} \cdot T_{k-1}^{-1} \cdot (I_w \otimes C_k) \cdot Z_k = Z_k \cdot S'_k \cdot T_k^{-1} \cdot (I_w \otimes D_k^{-1}).$$

Hence, from Claim 4.2, there is a matrix $M_k \in \mathrm{GL}(w, \mathbb{F})$ such that

$$(T'_{k-1} T_{k-1}^{-1}) \cdot (I_w \otimes C_k) = (S'_k T_k^{-1}) \cdot (I_w \otimes D_k^{-1}) = M_k \otimes I_w \,.$$

This implies $(T'_{k-1})^{-1} = T_{k-1}^{-1} \cdot (I_w \otimes C_k) \cdot (M_k^{-1} \otimes I_w)$ and $S'_k = (M_k \otimes I_w) \cdot (I_w \otimes D_k) \cdot T_k$. □

*Observation 4.3 (restated): Let $M_1, \ldots, M_{d-2}$ be the matrices as defined in Observation 4.2. Then*

1. $\widehat{Y}_k = (M_k M_{k+1}^{-1} \otimes I_w) \cdot (I_w \otimes (C_k^{-1} \cdot X_k \cdot C_{k+1}))$ *for* $k \in [1, d-3]$,
2. $\widehat{Y}_{d-2} = I_w \otimes (C_{d-2}^{-1} \cdot X_{d-2} \cdot D_{d-2}^{-1})$,

3. $\widehat{Y}_0 = Y_0 \cdot (I_w \otimes C_1) \cdot (M_1^{-1} \otimes I_w)$, *and* $\widehat{Y}_{d-1} = (M_{d-2} \otimes I_w) \cdot (I_w \otimes D_{d-2}) \cdot Y_{d-1}$.

*Proof.* Let $T'_k, S'_k, T_k$ for $k \in [0, d-2]$, and $Y'_k, Y_k, X'_k, X_k$ for $k \in [0, d-1$ be as in Section 4.1.

*a)* For $k \in [1, d-3]$, $\widehat{Y}_k = T'_{k-1} \cdot Y'_k \cdot (T'_k)^{-1}$, $Y'_k = T_{k-1}^{-1} \cdot Y_k \cdot T_k$, and $Y_k = I_w \otimes X_k$. From Observation 4.2,

$$T'_{k-1} = (M_k \otimes I_w) \cdot (I_w \otimes C_k^{-1}) \cdot T_{k-1} \quad \text{and} \quad (T'_k)^{-1} = T_k^{-1} \cdot (I_w \otimes C_{k+1}) \cdot (M_{k+1}^{-1} \otimes I_w)\,,$$

and hence for $k \in [1, d-3]$

$$\widehat{Y}_k = T'_{k-1} \cdot Y'_k \cdot (T'_k)^{-1} = (M_k \otimes I_w) \cdot (I_w \otimes C_k^{-1}) \cdot Y_k \cdot (I_w \otimes C_{k+1}) \cdot (M_{k+1}^{-1} \otimes I_w)\,.$$

Since $(I_w \otimes (C_k^{-1} \cdot X_k \cdot C_{k+1})) \cdot (M_{k+1}^{-1} \otimes I_w) = (M_{k+1}^{-1} \otimes I_w) \cdot (I_w \otimes (C_k^{-1} \cdot X_k \cdot C_{k+1}))$, for $k \in [1, d-3]$

$$\widehat{Y}_k = (M_k M_{k+1}^{-1} \otimes I_w) \cdot (I_w \otimes (C_k^{-1} \cdot X_k \cdot C_{k+1}))\,.$$

*b)* Recall that $\widehat{Y}_{d-2} = T'_{d-3} \cdot Y'_{d-2} \cdot (S'_{d-2})^{-1}$, $Y'_{d-2} = T_{d-3}^{-1} \cdot Y_{d-2} \cdot T_{d-2}$, and $Y_{d-2} = I_w \otimes X_{d-2}$. From Observation 4.2,

$$T'_{d-3} = (M_{d-2} \otimes I_w) \cdot (I_w \otimes C_{d-2}^{-1}) \cdot T_{d-3}, \quad \text{and} \quad (S'_{d-2})^{-1} = T_{d-2}^{-1} \cdot (I_w \otimes D_{d-2}^{-1}) \cdot (M_{d-2}^{-1} \otimes I_w).$$

Hence,

$$\widehat{Y}_{d-2} = T'_{d-3} \cdot Y'_{d-2} \cdot (S'_{d-2})^{-1} = (M_{d-2} \otimes I_w) \cdot (I_w \otimes C_{d-2}^{-1}) \cdot Y_{d-2} \cdot (I_w \otimes D_{d-2}^{-1}) \cdot (M_{d-2}^{-1} \otimes I_w)\,.$$

Since $(I_w \otimes (C_{d-2}^{-1} \cdot X_{d-2} \cdot D_{d-2}^{-1})) \cdot (M_{d-2}^{-1} \otimes I_w) = (M_{d-2}^{-1} \otimes I_w) \cdot (I_w \otimes (C_{d-2}^{-1} \cdot X_{d-2} \cdot D_{d-2}^{-1}))$,

$$\widehat{Y}_{d-2} = I_w \otimes (C_{d-2}^{-1} \cdot X_{d-2} \cdot D_{d-2}^{-1})\,.$$

*c)* Recall $\widehat{Y}_0 = Y'_0 \cdot (T'_0)^{-1}$, $\widehat{Y}_{d-1} = S'_{d-2} \cdot Y'_d$, and $Y'_0 = Y_0 \cdot T_0$, $Y'd-1 = T_{d-2}^{-1} \cdot Y_{d-1}$. From Observation 4.2, $(T'_0)^{-1} = T_0^{-1} \cdot (I_w \otimes C_1) \cdot (M_1^{-1} \otimes I_w)$ and $S'_{d-2} = (M_{d-2} \otimes I_w) \cdot (I_w \otimes D_{d-2}) \cdot T_{d-2}$. Hence,

$$\widehat{Y}_0 = Y'_0 \cdot (T'_0)^{-1} = Y_0 \cdot (I_w \otimes C_1) \cdot (M_1^{-1} \otimes I_w) \text{ and}$$
$$Y'_{d-1} = T_{d-2}^{-1} \cdot Y_{d-1} = (M_{d-2} \otimes I_w) \cdot (I_w \otimes D_{d-2}) \cdot Y_{d-1}\,.$$

□

# E Reduction from TRACE-TI to MMTI: Proof of Theorem 3

The input to Algorithm 7 is blackbox access to a $d$-tensor $f$ in the variable sets $\mathbf{x}_0, \ldots, \mathbf{x}_{d-1}$, and oracle access to MMTI. With high probability the algorithm does the following: If $f$ is isomorphic to Tr-IMM$_{w,d}$ then it outputs $B_0, \ldots, B_{d-1} \in \mathrm{GL}(w^2, \mathbb{F})$ such that $f =$ Tr-IMM$_{w,d}(B_0\mathbf{x}_0, \ldots, B_{d-1}\mathbf{x}_{d-1})$, otherwise it outputs 'No'. Since a PIT at the end of the algorithm ensures that the output of the algorithm is correct with high probability, we can assume that $f$ is isomorphic to Tr-IMM$_{w,d}$.

**Steps 1–2**: Let $n = w^2 d$. Since $f$ is isomorphic to Tr-IMM$_{w,d}$, there is a full-rank $(w, d, n)$-set-multilinear matrix product $X_0 \ldots X_{d-1}$ in $\mathbf{x}_0, \ldots, \mathbf{x}_{d-1}$ variables such that $f = \mathrm{tr}(X_0 \ldots X_{d-1})$.

**Algorithm 7** Reduction from TRACE-TI to MMTI

INPUT: Blackbox access to a $d$-tensor $f(\mathbf{x}_0, \ldots, \mathbf{x}_{d-1})$, where $d \geq 3$, and oracle access to MMTI.
OUTPUT: Matrices $B_0, \ldots, B_{d-1} \in \mathrm{GL}(w^2, \mathbb{F})$ such that $f(\mathbf{x}) = \text{Tr-IMM}_{w,d}(B_0\mathbf{x}_0, \ldots, B_{d-1}\mathbf{x}_{d-1})$.

Computing the first three matrices

1: Choose $d-3$ random points $\mathbf{a}_3, \ldots \mathbf{a}_{d-1} \in S^{w^2}$, where $S \subseteq \mathbb{F}$ and $|S| \geq n^5$. Let $\mathbf{y} = \uplus_{k\in[0,2]}\mathbf{x}_k$, and $h(\mathbf{y}) = f(\mathbf{y}, \mathbf{a}_3, \ldots, \mathbf{a}_{d-1})$.

2: Query MMTI on input $h(\mathbf{y})$. If MMTI outputs 'No' then output 'No'. Otherwise, let $B_0, B_1, B_2 \in \mathrm{GL}(w^2, \mathbb{F})$ be the output of MMTI. Using $B_0, B_1, B_2$ compute $w \times w$ linear matrices $X'_0, X'_1, X'_2$ respectively such that $h(\mathbf{y}) = \mathrm{tr}(X'_0 \cdot X'_1 \cdot X'_2)$.

The $d = 4$ case

3: For $i, j \in [w]$, and $k \in [0, 2]$, compute $\mathbf{b}^{(k)}_{i,j} \in \mathbb{F}^{w^2}$ such that $X'_k(\mathbf{b}^{(k)}_{i,j})$ has one in the $(i, j)$-th entry and zero elsewhere. Here $X'_k(\mathbf{b}^{(k)}_{i,j})$ is the matrix $X'_k$ with its entries evaluated at $\mathbf{b}^{(k)}_{i,j}$.

4: If $d = 4$ then construct $X'_3$ such that its $(i, j)$-th entry is the linear form $f(\mathbf{b}^{(0)}_{j,1}, \mathbf{b}^{(1)}_{1,1}, \mathbf{b}^{(2)}_{1,i}, \mathbf{x}_3)$. Let $B_3$ be the transformation matrix on $\mathbf{x}_3$ which is derived from $X'_3$. Go to Step 10.

The $d \geq 5$ case

5: Let $g(\mathbf{x} \setminus \mathbf{y}) = f(\mathbf{b}^{(0)}_{1,1}, \mathbf{b}^{(1)}_{1,1}, \mathbf{b}^{(2)}_{1,1}, \mathbf{x} \setminus \mathbf{y})$. Use the set-multilinear ABP reconstruction algorithm in [KS03] (also see Claim 2.4 in [KNST19]) to construct a full-rank $(w, d, n)$-set-multilinear ABP $Y'_3 \cdots Y'_{d-1}$ in $\mathbf{x}_3, \ldots, \mathbf{x}_{d-1}$ variables that computes the polynomial $g$.

6: For $j \in [w]$, compute $\mathbf{b}^{(d-1)}_j \in \mathbb{F}^{w^2}$ such that the $j$-th entry of $Y'_{d-1}(\mathbf{b}^{(d-1)}_j) \in \mathbb{F}^w$ is one and other entries are zero. For $k \in [4, d-2]$ let $X'_k = Y'_k$, and compute $\mathbf{b}^{(k)}_{i,j} \in \mathbb{F}^{w^2}$ such that $X'_k(\mathbf{b}^{(k)}_{i,j}) \in \mathbb{F}^{w\times w}$ has one in the $(i, j)$-th entry and other entries are zero, where $i, j \in [w]$.

7: Construct $X'_3$ whose $(i, j)$-th entry is $f(\mathbf{b}^{(0)}_{1,1}, \mathbf{b}^{(1)}_{1,1}, \mathbf{b}^{(0)}_{1,i}, \mathbf{x}_3, \mathbf{b}^{(4)}_{j,j}, \mathbf{b}^{(5)}_{j,j}, \ldots, \mathbf{b}^{(d-1)}_j)$. For $i, j \in [w]$, compute $\mathbf{b}^{(3)}_{i,j} \in \mathbb{F}^{w^2}$ such that the $(i, j)$-th entry of $X'_3(\mathbf{b}^{(3)}_{i,j})$ is one and other entries zero.

8: Construct $X'_{d-1}$ such that its $(i, j)$ entry is $f(\mathbf{b}^{(0)}_{j,1}, \mathbf{b}^{(1)}_{1,1}, \ldots, \mathbf{b}^{(d-2)}_{1,i}, \mathbf{x}_{d-1})$, for $i, j \in [w]$. Finally, $B_k$ be the transformation matrix on $\mathbf{x}_k$ which is derived from $X'_k$ for $k \in [3, d-1]$.

Final PIT

9: Pick random points $\mathbf{c}_0, \ldots, \mathbf{c}_{d-1} \in S^{w^2}$, where $S \subseteq \mathbb{F}$ and $|S| \geq n^5$. If $f(\mathbf{c}_0, \ldots, \mathbf{c}_{d-1}) = \text{Tr-IMM}_{w,d}(B_0\mathbf{c}_0, \ldots, B_{d-1}\mathbf{c}_{d-1})$ then output $B_0, \ldots, B_{d-1}$, otherwise output 'No'.

Hence, $h(\mathbf{y}) = \mathrm{tr}(X_0 \cdot X_1 \cdot X_2 \cdot X_3(\mathbf{a}_3) \ldots X_{d-1}(\mathbf{a}_{d-1}))$, where $X_k(\mathbf{a}_k)$ is $X_k$ with its entries evaluated at $\mathbf{a}_k$. Since $X_k$ is a full-rank linear matrix, with high probability, $X_k(\mathbf{a}_k) \in \mathrm{GL}(w, \mathbb{F})$. Then $M = X_3(\mathbf{a}_3) \ldots X_{d-1}(\mathbf{a}_{d-1}) \in \mathrm{GL}(w, \mathbb{F})$, and $h(\mathbf{y}) = \mathrm{tr}(X_0 \cdot X_1 \cdot (X_2 M))$ is isomorphic to Tr-IMM$_{w,3}$. At Step 2, MMTI returns $B_0, B_1, B_2 \in \mathrm{GL}(w^2, \mathbb{F})$ such that $h(\mathbf{y}) = \text{Tr-IMM}_{w,d}(B_0\mathbf{x}_0, B_1\mathbf{x}_1, B_2\mathbf{x}_2)$. It now follows from Lemma 3.4 that corresponding to $X'_0, X'_1, X'_2$ there are matrices $C_0, C_1, C_2 \in \mathrm{GL}(w, \mathbb{F})$ such that $X'_0 = C_0^{-1} \cdot X_0 \cdot C_1$, $X'_1 = C_1^{-1} \cdot X_1 \cdot C_2$, and $X'_2 = C_2^{-1} \cdot (X_2 M) \cdot C_0$.

**Steps 3–4**: The $d = 4$ case arises in Algorithm 3. We present this case separately as it is easier

to handle. Since $X'_0, X'_1, X'_2$ are full-rank linear matrices, at Step 3 a point $\mathbf{b}_{i,j}^{(k)}$ exists such that the $(i,j)$-th entry of $X_k(\mathbf{b}_{i,j}^{(k)})$ is one and other entries are zero. The point $\mathbf{b}_{i,j}^{(k)}$ can be computed by solving a system of linear equations. If $d = 4$ then $f = \mathrm{tr}(X'_0 \cdot X'_1 \cdot X'_2 \cdot (C_0^{-1}M^{-1}) \cdot X_3 \cdot C_0))$. Verify that $f(\mathbf{b}_{j,1}^{(0)}, \mathbf{b}_{1,1}^{(1)}, \mathbf{b}_{1,i}^{(2)}, \mathbf{x}_3)$ is the $(i,j)$-th entry of $(C_0^{-1}M^{-1}) \cdot X_3 \cdot C_0$.

**Steps 5–8**: Let $Y = (C_0^{-1}M^{-1}) \cdot X_3 \ldots X_{d-1}C_0$. Observe that $f(\mathbf{x}) = \mathrm{tr}(X'_0 \cdot X'_1 \cdot X'_2 \cdot Y)$. At Step 5, $g(\mathbf{x} \setminus \mathbf{y})$ is equals the $(1,1)$ entry of $Y$. Let $Y_3 = (C_0^{-1}M^{-1}) \cdot X_3$, $Y_k = X_k$ for $k \in [4, d-2]$, and $Y_{d-1} = X_{d-1} \cdot C_0$, and $Y_3(i,*)$ and $Y_{d-1}(*,j)$ denote the $i$-th row of $Y_3$ and the $j$-th column of $Y_{d-1}$ respectively. Then $g(\mathbf{x} \setminus \mathbf{y})$ is computed by the full-rank $(w,d,n)$-set-multilinear ABP $Y_3(1,*) \cdot Y_4 \ldots Y_{d-2} \cdot Y_{d-1}(*,1)$ in $\mathbf{x}_3, \ldots, \mathbf{x}_{d-1}$ variables. Using the algorithm in [KS03], a full-rank $(w,d,n)$-set-multilinear ABP $Y'_3 \cdot Y'_4 \cdot \cdots Y'_{d-2} \cdot Y'_{d-1}$ in $\mathbf{x}_3, \ldots, \mathbf{x}_{d-1}$ variables is constructed that computes $g$. The ABP constructed is such that there are matrices $C_3, \ldots, C_{d-2} \in \mathrm{GL}(w, \mathbb{F})$ such that $Y'_3 = Y_3(1,*) \cdot C_3$, $Y'_k = C_{k-1}^{-1} \cdot Y_k \cdot C_k$ for $k \in [4, d-2]$, and $Y'_{d-1} = C_{d-2}^{-1} \cdot Y_{d-1}(*,1)$. At Step 6, the points $\mathbf{b}_j^{(d-1)}$ and $\mathbf{b}_{i,j}^{(k)}$ are computed by solving systems of linear equations. Verify that $f = \mathrm{Trace}(X'_0 \cdot X'_1 \cdot X'_2 \cdot (Y_3C_3) \cdot X'_4 \cdots X'_{d-2} \cdot (C_{d-2}^{-1}Y_{d-1}))$. This implies that $f(\mathbf{b}_{1,1}^{(0)}, \mathbf{b}_{1,1}^{(1)}, \mathbf{b}_{1,i}^{(0)}, \mathbf{x}_3, \mathbf{b}_{j,j}^{(4)}, \mathbf{b}_{j,j}^{(5)}, \ldots, \mathbf{b}_j^{(d-1)})$ is the $(i,j)$-th entry of $Y_3C_3$ at Step 7. Further, $f(\mathbf{b}_{j,1}^{(0)}, \mathbf{b}_{1,1}^{(1)}, \ldots, \mathbf{b}_{1,i}^{(d-2)}, \mathbf{x}_{d-1})$ is the $(i,j)$-th entry of $C_{d-2}^{-1}Y_{d-1}$ at Step 8. Hence, $X'_3 = Y_3C_3$ and $X'_{d-1} = C_{d-2}^{-1}Y_{d-1}$. In particular, $f = \mathrm{tr}(X'_0 \ldots X'_{d-1}) = \text{Tr-IMM}_{w,d}(B_0\mathbf{x}_0, \ldots, B_{d-1}\mathbf{x}_{d-1})$.

# F Proofs from Section 5

*Lemma 5.1 (restated): Let $f$ be a non-zero $d$-tensor in the variable sets $\mathbf{x}_0, \ldots, \mathbf{x}_{d-1}$ such that for all $k \in [0, d-1]$ $\mathcal{B}_k \subseteq \mathfrak{g}_f$. Then there is an $\alpha \in \mathbb{F}^\times$ such that $f(\mathbf{x}) = \alpha \cdot$ Tr-IMM$_{w,d}(\mathbf{x})$.*

*Proof.* A path monomial (see Definition A.1) looks like $\mu = x_{i_0,i_1}^{(0)} \cdot x_{i_1,i_2}^{(1)} \ldots x_{i_{d-1},i_0}^{(d-1)}$. The next claim shows that the coefficient of a non-path monomial in $f$ is zero. In the claim, $f_{(u,v),(p,q)}^{(k,k+1)}$ denotes the coefficient of $x_{u,v}^{(k)}x_{p,q}^{(k+1)}$ in $f$ over $\mathbb{F}[\mathbf{x} \setminus \{\mathbf{x}_k, \mathbf{x}_{k+1}\}]$, where $u, v, p, q \in [w]$.

**Claim F.1.** *Let $\mu$ be a non-path monomial. Then the coefficient of $\mu$ in $f$ is zero.*

*Proof.* Let $\mu = x_{i_0,j_0}^{(0)} \cdot x_{i_1,j_1}^{(1)} \ldots x_{i_{d-1},j_{d-1}}^{(d-1)}$ be a non-path monomial. Hence there is a $k \in [0, d-1]$ such that $j_k \neq i_{k+1}$. Suppose $k \in [0, d-2]$, and $k$ is even. Let $D \in \mathcal{M}_w$ be a diagonal matrix such that its $(j_k, j_k)$ entry is one and all the other entries are zero. Let $B \in \mathcal{B}_k$ be a block-diagonal matrix whose $2w^2 \times 2w^2$ sub-matrix indexed by $\mathbf{x}_k \uplus \mathbf{x}_{k+1}$ looks like

$$\begin{bmatrix} I_w \otimes D & \mathbf{0} \\ \mathbf{0} & -I_w \otimes D \end{bmatrix}.$$

Since $B \in \mathfrak{g}_f$, we have (by the variable ordering in $\mathbf{x}_k$ and $\mathbf{x}_{k+1}$),

$$\sum_{u \in [w]} x_{u,j_k}^{(k)} \left( \sum_{p,q \in [w]} x_{p,q}^{(k+1)} f_{(u,j_k),(p,q)}^{(k,k+1)} \right) = \sum_{u \in [w]} x_{j_k,u}^{(k+1)} \left( \sum_{p,q \in [w]} x_{p,q}^{(k)} f_{(p,q),(j_k,u)}^{(k,k+1)} \right). \tag{17}$$

From Equation 17 we conclude that for all $u, q \in [w]$, $f^{(k,k+1)}_{(u,j_k),(p,q)} = 0$ if $p \neq j_k$. Now suppose for contradiction the coefficient of $\mu$ in $f$ is non-zero. Then the coefficient of $x^{(k)}_{i_k,j_k} x^{(k+1)}_{i_{k+1},j_{k+1}}$ (i.e., $f^{(k,k+1)}_{(i_k,j_k),(i_{k+1},j_{k+1})}$) is non-zero. Since $j_k \neq i_{k+1}$, this is a contradiction. If $k \in [0, d-1]$ and $k$ is odd then the proof is similar and the only thing to note in this case is that $B \in \mathcal{B}_k$ is such that its $2w^2 \times 2w^2$ sub-matrix indexed by the $\mathbf{x}_k \uplus \mathbf{x}_{k+1}$ looks like

$$\begin{bmatrix} D \otimes I_w & \mathbf{0} \\ \mathbf{0} & -D \otimes I_w \end{bmatrix}.$$

Finally, if $k = d-1$ and $d$ is odd then again the proof is similar but $B \in \mathcal{B}_{d-1}$ in this case is such that the $w^2 \times w^2$ sub-matrix of $B$ indexed by $\mathbf{x}_{d-1}$ variables is equal to $I_w \otimes D$, and the $w^2 \times w^2$ sub-matrix of $B$ indexed by $\mathbf{x}_0$ variables is equal to $-D \otimes I_w$. □

**Claim F.2.** *Let $\mu_1 = x^{(0)}_{i_0,i_1} \cdot x^{(1)}_{i_1,i_2} \dots x^{(k)}_{i_k,i_{k+1}} \cdot x^{(k+1)}_{i_{k+1},i_{k+2}} \dots x^{(d-1)}_{i_{d-1},i_0}$, and $\mu_2 = x^{(0)}_{i_0,i_1} \cdot x^{(1)}_{i_1,i_2} \dots x^{(k)}_{i_k,i'_{k+1}} \cdot x^{(k+1)}_{i'_{k+1},i_{k+2}} \dots x^{(d-1)}_{i_{d-1},i_0}$ be two path monomials. Then the coefficients of $\mu_1$ and $\mu_2$ in $f$ are equal.*

*Proof.* Suppose $k \in [0, d-2]$ and $k$ is even. Let $M \in \mathcal{M}_w$ be such that its $(i'_{k+1}, i_{k+1})$ is one and all its other entries are zero, and $B \in \mathcal{B}_k$ be a block-diagonal matrix such that $B$ restricted to the $2w^2 \times 2w^2$ sub-matrix indexed by $\mathbf{x}_k \uplus \mathbf{x}_{k+1}$ variables is as shown below

$$\begin{bmatrix} I_w \otimes M^T & \mathbf{0} \\ \mathbf{0} & -I_w \otimes M \end{bmatrix}.$$

Let the coefficients of $\mu_1$ and $\mu_2$ in $f$ be equal to $\alpha_1$ and $\alpha_2$. Since $B \in \mathfrak{g}_f$, we have

$$\sum_{u \in [w]} x^{(k)}_{u,i'_{k+1}} \frac{\partial f}{\partial x^{(k)}_{u,i_{k+1}}} - \sum_{u \in [w]} x^{(k+1)}_{i_{k+1},u} \frac{\partial f}{\partial x^{(k+1)}_{i'_{k+1},u}} = 0. \tag{18}$$

The coefficient of $\mu = x^{(0)}_{i_0,i_1} \cdot x^{(1)}_{i_1,i_2} \dots x^{(k)}_{i_k,i'_{k+1}} \cdot x^{(k+1)}_{i_{k+1},i_{k+2}} \dots x^{(d-1)}_{i_{d-1},i_0}$ in Equation 18 is equal to $\alpha_1 - \alpha_2 = 0$. Hence $\alpha_1 = \alpha_2$. The proof for the two remaining cases: a) $k \in [0, d-1]$ and $k$ odd, and b) $k = d-1$ and $d$ odd, follow similarly by constructing appropriate $B$ matrices. □

We now use above the claim to show the following.

**Claim F.3.** *Let $\mu_1 = x^{(0)}_{i_0,i_1} \cdot x^{(1)}_{i_1,i_2} \dots x^{(d-1)}_{i_{d-1},i_0}$, and $\mu_2 = x^{(0)}_{j_0,j_1} \cdot x^{(1)}_{j_1,j_2} \dots x^{(d-1)}_{j_{d-1},j_0}$ be two path monomials. Then the coefficient of $\mu_1$ and $\mu_2$ in $f$ are equal.*

*Proof.* For $k \in [1, d-2]$, let $\nu_k = x^{(0)}_{i_0,j_1} \cdot x^{(1)}_{j_1,j_2} \dots x^{(k-1)}_{j_{k-1},j_k} \cdot x^{(k)}_{j_k,i_{k+1}} \cdot x^{(k+1)}_{i_{k+1},i_{k+2}} \dots x^{(d-1)}_{i_{d-1},i_0}$, and $\nu_{d-1} = x^{(0)}_{i_0,j_1} \cdot x^{(1)}_{j_1,j_2} \dots x^{(d-2)}_{j_{d-2},j_{d-1}} \cdot x^{(d-1)}_{j_{d-1},i_0}$. From Claim F.2, the coefficients of $\mu_1$ and $\nu_1$ in $f$ are equal, the coefficients of $\nu_k$ and $\nu_{k+1}$ in $f$ are equal for $k \in [1, d-2]$, and the coefficients of $\nu_{d-1}$ and $\mu_2$ in $f$ are equal. Hence the coefficients of $\mu_1$ and $\mu_2$ in $f$ are equal. □

It follows immediately that there is an $\alpha \in \mathbb{F}^\times$ such that $f = \alpha \cdot \text{Tr-IMM}_{w,d}(\mathbf{x})$. □

*Corollary 5.1 (restated): Let $B \in \mathsf{GL}(n, \mathbb{F})$ be a block-diagonal matrix with individual blocks $B_0, \dots, B_{d-1}$ and $f$ be a non-zero $d$-tensor in the variable sets $\mathbf{x}_0, \dots, \mathbf{x}_{d-1}$ such that for all $k \in [0, d-1]$, $B^{-1} \cdot \mathcal{B}_k \cdot B \subseteq \mathfrak{g}_f$. Then there is an $\alpha \in \mathbb{F}^\times$ such that $f(\mathbf{x}) = \alpha \cdot Tr\text{-}IMM_{w,d}(B_0\mathbf{x}_0, \dots, B_{d-1}\mathbf{x}_{d-1})$.*

*Proof.* Let $g(\mathbf{x}) = f(B_0^{-1}\mathbf{x}_0, \ldots, B_{d-1}^{-1}\mathbf{x}_{d-1})$. Since $B^{-1} \cdot \mathcal{B}_k \cdot B \subseteq \mathfrak{g}_f$, $\mathcal{B}_k \subseteq \mathfrak{g}_g$ for $k \in [0, d-1]$ (from Fact 1). Hence, from Lemma 5.1 there is an $\alpha \in \mathbb{F}^\times$ such that $g = \alpha \cdot \text{Tr-IMM}_{w,d}$. Thus $f(\mathbf{x}) = \alpha \cdot \text{Tr-IMM}_{w,d}(B_0\mathbf{x}_0, \ldots, B_{d-1}\mathbf{x}_{d-1})$. □

*Claim 5.1 (restated): Suppose $\mathcal{A} \cong \mathcal{M}_w$ for some $w \in \mathbb{N}$. Then there exists a $K \in GL(w^2, \mathbb{F})$ and linearly independent matrices $\{C_{1,1}, \ldots, C_{w,w}\}$ in $\mathcal{M}_w$ such that $L_{i,j} = K^{-1} \cdot (I_w \otimes C_{i,j}) \cdot K$ for all $i, j \in [w]$.*

*Proof.* Let $\mathcal{L}$ be the algebra generated by the matrices $\{L_{1,1}, \ldots, L_{w,w}\}$. It is easy to see that $\mathcal{A} \cong \mathcal{L} \cong \mathcal{M}_w$ and $\mathcal{L}$ contains $I_{w^2}$. From Skolem-Noether theorem (see Theorem 5 in [GGKS19], and [Lor08]) we have that there is a $K \in \mathsf{GL}(w^2, \mathbb{F})$ and linearly independent matrices $\{C_{1,1}, \ldots, C_{w,w}\}$ in $\mathcal{M}_w$ such that $L_{i,j} = K^{-1} \cdot (I_w \otimes C_{i,j}) \cdot K$ for all $i, j \in [w]$. □



ECCC ISSN 1433-8092
https://eccc.weizmann.ac.il