Reconstruction of Full Rank Algebraic Branching Programs

NEERAJ KAYAL, Microsoft Research, India
VINEET NAIR and CHANDAN SAHA, Indian Institute of Science, India
SÉBASTIEN TAVENAS, Microsoft Research, India and Univ. Savoie Mont Blanc, France

An algebraic branching program (ABP) A can be modelled as a product expression $X_1 \cdot X_2 \ldots X_d$, where X_1 and X_d are $1 \times w$ and $w \times 1$ matrices, respectively, and every other X_k is a $w \times w$ matrix; the entries of these matrices are linear forms in m variables over a field $\mathbb{F}$ (which we assume to be either $\mathbb{Q}$ or a field of characteristic poly(m)). The polynomial computed by A is the entry of the 1×1 matrix obtained from the product $\prod_{k=1}^{d} X_k$. We say A is a *full rank* ABP if the $w^2(d-2) + 2w$ linear forms occurring in the matrices $X_1, X_2, \ldots, X_d$ are $\mathbb{F}$-linearly independent. Our main result is a randomized reconstruction algorithm for full rank ABPs: Given blackbox access to an m-variate polynomial f of degree at most m, the algorithm outputs a full rank ABP computing f if such an ABP exists, or outputs "no full rank ABP exists" (with high probability). The running time of the algorithm is polynomial in m and β, where β is the bit length of the coefficients of f. The algorithm works even if X_k is a $w_{k-1} \times w_k$ matrix (with $w_0 = w_d = 1$), and $\mathbf{w} = (w_1, \ldots, w_{d-1})$ is *unknown*. The result is obtained by designing a randomized polynomial time equivalence test for the family of iterated matrix multiplication polynomial $\mathsf{IMM}_{\mathbf{w},d}$, the $(1,1)$-th entry of a product of d rectangular symbolic matrices whose dimensions are according to $\mathbf{w} \in \mathbb{N}^{d-1}$. At its core, the algorithm exploits a connection between the *irreducible invariant subspaces* of the Lie algebra of the group of symmetries of a polynomial f that is equivalent to $\mathsf{IMM}_{\mathbf{w},d}$ and the "layer spaces" of a full rank ABP computing f. This connection also helps determine the group of symmetries of $\mathsf{IMM}_{\mathbf{w},d}$ and show that $\mathsf{IMM}_{\mathbf{w},d}$ is characterized by its group of symmetries.

CCS Concepts: • **Theory of computation** → **Algebraic complexity theory**; *Circuit complexity*; Probabilistic computation;

Additional Key Words and Phrases: Iterated matrix multiplication, equivalence testing, group of symmetries, Lie algebra, layer spaces

ACM Reference format:
Neeraj Kayal, Vineet Nair, Chandan Saha, and Sébastien Tavenas. 2018. Reconstruction of Full Rank Algebraic Branching Programs. *ACM Trans. Comput. Theory* 11, 1, Article 2 (November 2018), 56 pages.
https://doi.org/10.1145/3282427

Authors' addresses: N. Kayal, Microsoft Research, Vigyan 9, Lavelle Road, Shanthala Nagar, Ashok Nagar, Bengaluru, Karnataka, 560001, India; email: neeraka@microsoft.com; V. Nair and C. Saha, Indian Institute of Science, Bengaluru, Karnataka, 560012, India; emails: {vineet, chandan}@iisc.ac.in; S. Tavenas, Microsoft Research, Vigyan 9, Lavelle Road, Shanthala Nagar, Ashok Nagar, Bengaluru, Karnataka, 560001, India, Univ. Savoie Mont Blanc, CNRS, LAMA, Chamb, F-73000, France; email: sebastien.tavenas@univ-smb.fr.

1942-3454/2018/11-ART2 $15.00
https://doi.org/10.1145/3282427

1 INTRODUCTION

1.1 Circuit Reconstruction

Reconstruction of arithmetic circuits is the algebraic analogue of exact learning [4] of Boolean circuits using membership and equivalence queries. A reconstruction algorithm takes input of an oracle access to an m-variate degree d polynomial f computed by a size s arithmetic circuit from some circuit class C, and it outputs an arithmetic circuit (preferably from the same class) of not too large size[1] computing f. The algorithm is allowed to make two kinds of adaptive queries to the oracle: It may ask for evaluation of f at a point $\mathbf{a} \in \mathbb{F}^m$ chosen by the algorithm (membership query). It may also form a circuit C (a hypothesis) and ask if the polynomial g, computed by C, equals f; if not, the oracle returns a point $\mathbf{b} \in \mathbb{F}^m$ such that $f(\mathbf{b}) \neq g(\mathbf{b})$ (equivalence query).[2] The desired running time of the algorithm is polynomial in m, d, s and the bit length of the coefficients of f.

Circuit reconstruction is a natural learning problem in algebraic complexity theory and is closely related to two other fundamental problems, lower bound and polynomial identity testing. Building on the ideas in References [1, 22, 29], Volkovich [43] showed that a polynomial time reconstruction algorithm for a circuit class C can be used to compute an m-variate multilinear polynomial h in $2^{O(m)}$ time such that any circuit from C computing h has size $2^{\Omega(m)}$.[3] Also, an efficient reconstruction algorithm (that uses only membership queries) for a class of circuits automatically gives an efficient blackbox[4] identity testing algorithm for the same class. In this sense, reconstruction is a "harder" problem than lower bound and identity testing.[5] However, if we allow reconstruction algorithms to be randomized (thereby giving them the power of identity testing), then we can hope to have efficient reconstructions even for some classes of circuits for which efficient blackbox identity testing algorithms are not known yet. Indeed, a randomized polynomial time reconstruction algorithm for read-once oblivious algebraic branching programs (ROABP) was given in Reference [30] much before the quasi-polynomial time hitting-set generators for the same model were designed [2, 13]. The case of read-once formulas is similar (see Reference [40]). A randomized reconstruction algorithm need not use equivalence queries as a random point $\mathbf{b}$ is a witness for $f(\mathbf{b}) \neq g(\mathbf{b})$, if $f \neq g$.[6] In this article, we will assume that reconstruction algorithms use *only* membership queries, unless we mention equivalence queries explicitly.

Another way to moderate the reconstruction setup is given by *average-case* reconstruction. Here the input polynomial f is picked according to some "natural" distribution on circuits from a class C. This relaxation led to the development of randomized polynomial time reconstruction algorithm for some powerful circuit classes [18, 20] (albeit on average). Our work is motivated by this line of research.

1.2 Previous Work on Reconstruction

We will assume that a circuit from class C computing the input polynomial f has a sum gate at the output. Otherwise, we can apply the factorization algorithm in Reference [23] to gain blackbox access to all the irreducible factors of f, thereby reducing the problem to a potentially simpler class

[1] We allow the algorithm to output sub-optimal size circuit as it is NP-hard to compute an optimal circuit for f even for restricted classes like set-multilinear depth three circuits [21].

[2] Throughout this article, we will assume that the base field $\mathbb{F}$ is sufficiently large, so if $f(\mathbf{b}) = g(\mathbf{b})$ for every $\mathbf{b} \in \mathbb{F}^m$, then $f = g$.

[3] Such an implication is not known for an h belonging to a VNP family.

[4] The algorithm has blackbox access to f, i.e., it can make only membership queries to an oracle holding f.

[5] Not much is known about the reverse direction: Do strong lower bounds or efficient blackbox identity testing for a circuit class imply efficient reconstruction for the same class? For certain interesting circuit classes, the techniques used for identity testing and lower bounds do help in efficient reconstruction (see References [18, 40]).

[6] The algorithm in Reference [30] is deterministic if we allow equivalence queries.

of circuits at the cost of making the reconstruction algorithm randomized. Thus, depth two, depth three, and depth four circuits would mean $\Sigma\Pi$, $\Sigma\Pi\Sigma$, and $\Sigma\Pi\Sigma\Pi$ circuits, respectively.

Low depth circuits: A polynomial time reconstruction algorithm for depth two circuits follows from the sparse polynomial interpolation algorithm in Reference [31]. By analysing the rank of the partial derivatives matrix, Klivans and Shpilka [30] gave a randomized reconstruction algorithm[7] for depth three circuits with fan-in of every product gate bounded by d in time polynomial in the size of the circuit and 2^d. Prior to this, a polynomial time randomized reconstruction algorithm for set-multilinear depth three circuits followed from Reference [6]. In both References [30] and [6] the output hypothesis is an ROABP. For depth three circuits with two product gates, Shpilka [38] gave a randomized reconstruction algorithm over a finite field $\mathbb{F}$ that has running time quasi-polynomial[8] in m, d, and $|\mathbb{F}|$. This algorithm was derandomized and extended to depth three circuits with constant number of product gates in Reference [24]. The output hypothesis in Reference [38] is a depth three circuit with two product gates (unless the circuit has a low *simple rank*[9]), but it works only over finite fields. Recently, Sinha [41] gave a polynomial time randomized reconstruction algorithm for depth three circuits with two product gates over rationals[10]; the output of Sinha's algorithm is also a depth three circuit with two product gates (unless the simple rank of the circuit is less than a fixed constant). For multilinear depth four circuits with two top level product gates, Reference [19] gave a randomized polynomial time reconstruction algorithm that works over both finite fields and rationals.

Restricted formulas and ABP: Recently, Minahan and Volkovich [34] gave a polynomial time reconstruction algorithm for read-once formulas by strengthening the analysis in Reference [39], the latter has a quasi-polynomial time reconstruction algorithm for the same model. Forbes and Shpilka [13] gave quasi-polynomial time reconstruction algorithms for ROABP, set-multilinear ABP and non-commutative ABP by derandomizing[11] the algorithm in Reference [30]. Prior to this, the case of non-commutative ABP reconstruction was solved in Reference [5], assuming blackbox access to the internal gates of the input ABP.

Average-case reconstruction: Few reconstruction algorithms are known under distributional assumptions on the inputs. Gupta et al. [18] gave a randomized polynomial time reconstruction algorithm for random multilinear formulas picked from a natural distribution: every sum gate computes a random linear combinations of its two children (subformulas), and at every product gate the set of variables is partitioned randomly into two equal size sets between its two children (subformulas); the subformulas are then constructed recursively. In Reference [20], a randomized polynomial time reconstruction algorithm was given for random formulas picked from the distribution of size s complete binary trees with alternating layers of sum and product gates, and the linear forms at the leaves are chosen independently and uniformly at random.

1.3 Motivation and Model

Motivation: Whether or not the techniques used to prove strong lower bounds for certain circuit classes imply efficient learning algorithms for the same classes is a fascinating direction of research. The problem bears particular relevance to natural lower bound proofs [37]. A natural lower bound proof for a circuit class gives an efficient procedure to identify some property of circuits belonging to the class. It is conceivable that such efficient procedures can be exploited to design learning

[7] The algorithm is deterministic if equivalence queries are used.

[8] The running time is polynomial in m, $|\mathbb{F}|$ if the depth three circuit is additionally multilinear.

[9] The dimension of the span of the linear forms in the two gates after removing their gcd.

[10] The result holds over characteristic zero fields. We state it for rationals as bit complexity concerns us.

[11] replacing the equivalence queries by quasi-polynomial size hitting-sets for ROABP.

algorithms for circuit classes admitting natural lower bound proofs. For Boolean circuit classes harboring $\mathrm{AC}^0[p]$, such a connection between natural proofs and PAC learning with membership queries has been established in Reference [11]. However, the situation is less clear for arithmetic circuit classes—an apparent difficulty being, unlike PAC learning, in an arithmetic circuit reconstruction problem we have to output a circuit that *exactly* computes the input polynomial function (instead of approximately, as in PAC learning). This state of the affair is also reflected by the fact that there are quite a few arithmetic circuit classes for which algebraically natural lower bound proofs are known (in the sense of References [14, 17]), but no efficient reconstruction algorithms whatsoever. To make progress and shed more light on the complexity of reconstruction, it is natural to ask if we can do efficient reconstruction for "most" circuits in a class. This takes us to the realm of average-case reconstruction, where the input circuit is picked from a class according to some natural P-samplable distribution. As mentioned above, efficient average-case randomized reconstruction algorithms for multilinear formulas and arithmetic formulas are given in References [18] and [20], respectively. Algebraic branching programs form a powerful circuit model that captures determinant computation [33] and is believably more powerful than arithmetic formulas. This motivates us to pose Problem 1 below (rather optimistically), and study a case when it can be solved.

Algebraic branching program: Algebraic branching program (ABP), an arithmetic analogue of Boolean branching program, is a well-studied model in algebraic complexity theory specially because it captures the complexity of polynomials like the iterated matrix multiplication and the symbolic determinant. Separating the computational powers of formulas and ABPs, and that of ABPs and circuits are outstanding open problems in arithmetic circuit complexity. An ABP is defined below. For the rest of this article, the base field $\mathbb{F}$ would be the field of rationals $\mathbb{Q}$.[12]

Definition 1.1 (Algebraic Branching Program). An *algebraic branching program* (ABP) of width w and length d is a product expression $X_1 \cdot X_2 \ldots X_d$, where X_1, X_d are row and column vectors of length w, respectively, and for $k \in [2, d-1]$, X_k is a $w \times w$ matrix. The entries in X_1 to X_d are affine forms in the variables $\mathbf{x} = \{x_1, x_2, \ldots, x_m\}$. The polynomial computed by the ABP is the entry of the 1×1 matrix obtained from the product $\prod_{k=1}^{d} X_k$. An ABP of width w, length d, and in m variables, and with the coefficients of the affine forms from $S \subseteq \mathbb{F}$, will be called a (w, d, m, S)-ABP.

An alternate definition: Alternatively, an ABP is defined as a layered directed acyclic graph with a source s and a sink t. A width w and length d ABP has $d + 1$ layers, where the first and the last layers contain one vertex each, labelled s and t, respectively, and every other layer has w vertices. There is an edge from every vertex in layer k to every vertex in layer $k + 1$, for all $k \in [d]$, and these edges between adjacent layers are labelled by affine forms in $\mathbf{x}$ variables. The weight of a path from s to t is the product of the edge labels in the path, and the polynomial computed by the ABP is the sum of the weights of all paths from s to t. It is easy to verify that the two definitions of ABP are equivalent. We use either of these definitions in our arguments later based on suitability.

Average-case ABP reconstruction: To study average-case complexity of the reconstruction problem for ABPs, we need to define a distribution on polynomials computed by ABPs. A seemingly natural distribution is as follows: Consider the universe of all polynomials computed by (w, d, m, S)-ABPs for some finite set $S \subseteq \mathbb{F}$ of large enough size. Pick a polynomial f uniformly at random from this universe, and give blackbox access to f as input to a reconstruction algorithm. However, a distribution is "realistic" only if there is an efficient sampling algorithm that outputs (some suitable circuit representation of) f according to the distribution. For the above distribution, it is not clear if such an efficient sampling algorithm exists. A reason being, multiple different

[12] Our results also hold over finite fields of sufficiently large (meaning, polynomial in the relevant parameters) characteristic.

ABPs may be computing the same polynomial, so picking a random ABP is not sufficient to sample from this distribution. However, picking a random ABP (as described below) gives another natural distribution for which there is a trivial efficient sampling algorithm. Let S_γ be the set of all positive and negative rational numbers with γ bits before and after the decimal.

Definition 1.2 (Random Algebraic Branching Program). Given the parameters w, d, m and γ, a *random* (w, d, m, S_γ)-ABP is a (w, d, m, S_γ)-ABP with coefficients of the affine forms chosen independently and uniformly at random from S_γ.[13]

Indeed, there is a randomized sampling algorithm that when given the parameters w, d, m, and γ outputs a random (w, d, m, S_γ)-ABP in time $(w, d, m, \gamma)^{O(1)}$. An average-case ABP reconstruction problem can then be posed as follows.

Problem 1 (Average-Case ABP Reconstruction). *Design an algorithm that when given blackbox access to a polynomial f computed by a random (w, d, m, S_γ)-ABP, outputs an ABP computing f with high probability.*[14] *The desired running time of the algorithm is $(w, d, m, \gamma)^{O(1)}$.*

Note that we allow the reconstruction algorithm to output any ABP computing f that may not be a (w, d, m, S_γ)-ABP. The main requirement is that the running time should be polynomial in w, d, m, and γ.

1.4 Our Result

We give a solution to the above problem, if the number of variables m and the size of the set S_γ are greater than $w^2 d$ and $(mwd)^2$, respectively. Observe that if the random affine forms in the matrices X_1 to X_d (as in Definition 1.2) have more than $w^2 d$ variables then these affine forms are $\mathbb{F}$-linearly independent with high probability as S_γ is also sufficiently large. This motivates us to define a *full rank* ABP. In the following discussion, by *homogeneous degree 1 part* of an affine form $a_0 + \sum_{i=1}^m a_i x_i$, we mean $\sum_{i=1}^m a_i x_i$ where $a_i \in \mathbb{F}$.

Definition 1.3 (Full Rank Algebraic Branching Program). A *full rank* ABP A of width bounded by w and length d is a product expression $X_1 \cdot X_2 \ldots X_d$, where X_1, X_d are row and column vectors of lengths w_1 and w_{d-1}, respectively, and for $k \in [2, d-1]$ X_k is a $w_{k-1} \times w_k$ matrix such that $w_k \leq w$ for all $k \in [d-1]$; the entries in X_1 to X_d are affine forms in $\mathbf{x}$ variables and moreover, the homogeneous degree 1 parts of these affine forms are $\mathbb{F}$-linearly independent. We say ABP A has *width* $\mathbf{w} = (w_1, w_2, \ldots, w_{d-1}) \in \mathbb{N}^{d-1}$.

The following is an example of a full rank ABP:

$$\begin{bmatrix} 1 + x_1 + x_2 & 2 + x_2 + x_3 & x_3 + x_4 \end{bmatrix} \begin{bmatrix} 1 + x_4 + x_5 & x_5 + x_6 \\ x_6 + x_7 & x_7 + x_8 \\ x_8 + x_9 & 4 + x_9 + x_{10} \end{bmatrix} \begin{bmatrix} 3 + x_{10} + x_{11} \\ 2 + x_{11} \end{bmatrix}.$$

A canonical example: Another example of a polynomial computed by a full rank ABP is the iterated matrix multiplication polynomial $\mathsf{IMM}_{\mathbf{w},d}$, which is the entry of the 1×1 matrix obtained from a product of d *symbolic* matrices X_1 to X_d with dimensions as in Definition 1.3. The number of variables in $\mathsf{IMM}_{\mathbf{w},d}$ is $n = w_1 + \sum_{k=2}^{d-1} w_{k-1} w_k + w_{d-1}$. See Section 2.3 for a slightly detailed definition of $\mathsf{IMM}_{\mathbf{w},d}$. Generally, in the literature, the matrices X_1 to X_d have a uniform dimension w (i.e.,

[13] More generally, S_γ can be any arbitrarily fixed set containing rational numbers of the form $\frac{p}{q}$, where p and q are γ bit integers. For concreteness of the discussion, we have fixed S_γ in a specific way.

[14] The probability is taken over the random choice of f (the polynomial computed by a random (w, d, m, S_γ)-ABP) as well as over the random bits used by the reconstruction algorithm, if it is randomized.

$w_k = w$ for every $k \in [d-1]$), and the polynomial is denoted by $\mathsf{IMM}_{w,d}$. We consider varying dimensions primarily because the algorithm in Theorem 1 below is able to handle this general setting, even if $\mathbf{w}$ is *unknown*.

Although the model full rank ABP is natural and powerful, it is nevertheless incomplete—not every polynomial f can be computed by a full rank ABP even if f is multilinear (see Observation 7.1 in Section 7.1). Our main result is an efficient randomized algorithm to reconstruct full rank ABP.

Theorem 1 (Full Rank ABP Reconstruction). *There is a randomized algorithm that takes as input a blackbox for an m variate polynomial f over $\mathbb{F}$ of degree $d \in [5, m]$, and with high probability it does the following: if f is computed by a full rank ABP, then the algorithm outputs a full rank ABP computing f, else it outputs "f does not admit a full rank ABP." The running time is poly(m, β),*[15] *where β is the bit length of the coefficients of f.*

Remarks: Theorem 1 implies an efficient average-case reconstruction algorithm for ABPs (Problem 1) when $m \geq w^2 d$ and $|S_\gamma| \geq (mwd)^2$, as a random (w, d, m, S_γ)-ABP is full rank with high probability if m and $|S_\gamma|$ are sufficiently large. The algorithm of Theorem 1 is given in Section 1.6. Following are a couple of remarks on this algorithm:

(1) *Uniqueness of full rank ABP*: Suppose f is computed by a full rank ABP of width $\mathbf{w} = (w_1, w_2, \ldots, w_{d-1})$, and assume[16] that $w_k > 1$ for every $k \in [d-1]$. Then the output of the algorithm is a full rank ABP of width $\mathbf{w}$ or $(w_{d-1}, w_{d-2}, \ldots, w_1)$, with probability at least $1 - \frac{1}{\text{poly}(w,d)}$, where $w = \max_{k \in [d-1]}\{w_k\}$. In fact, any full rank ABP computing f is "unique" up to the symmetries[17] of iterated matrix multiplication, which we study in Section 6.

(2) *No knowledge of* $\mathbf{w}$: The algorithm does not need *a priori* knowledge of the width vector $\mathbf{w}$, it only knows the number of variables m and the degree d of f. The algorithm is able to derive $\mathbf{w}$ from blackbox access to f (Section 1.6 gives a sketch of how this is done).

Observe that if f is computed by a full rank ABP of width $\mathbf{w}$ then f is an affine projection of the polynomial $\mathsf{IMM}_{\mathbf{w},d}$ via a full rank transformation (see Definition 2.8). So the above theorem is identical to the theorem below.

Theorem 1a. *Given blackbox access to an m variate polynomial $f \in \mathbb{F}[\mathbf{x}]$ of degree $d \in [5, m]$, the problem of checking if there exist a $\mathbf{w} \in \mathbb{N}^{d-1}$, a $B \in \mathbb{F}^{n \times m}$ of rank n equal to the number of variables in $\mathsf{IMM}_{\mathbf{w},d}$, and a $\mathbf{b} \in \mathbb{F}^n$ such that $f = \mathsf{IMM}_{\mathbf{w},d}(B\mathbf{x} + \mathbf{b})$,*[18] *can be solved in randomized poly(m, β) time where β is the bit length of the coefficients of f. Further, with probability at least $1 - \frac{1}{poly(n)}$, the following is true: the algorithm returns a $\mathbf{w}$, a $B \in \mathbb{F}^{n \times m}$ of rank n, and a $\mathbf{b} \in \mathbb{F}^n$ such that $f = \mathsf{IMM}_{\mathbf{w},d}(B\mathbf{x} + \mathbf{b})$ if such $\mathbf{w}$, B and $\mathbf{b}$ exist, else it outputs "f does not admit a full rank ABP."*

A full rank ABP for f can be derived readily, once we compute $\mathbf{w}$, B and $\mathbf{b}$ as above. Using known results on variable reduction and translation equivalence test (see Section 2.2) proving Theorem 1a reduces in polynomial time to giving an equivalence test (see Definition 2.9) for the $\mathsf{IMM}_{\mathbf{w},d}$ polynomial—this reduction is described in Section 1.6.

Theorem 1b (Equivalence Test for IMM). *Given blackbox access to a homogeneous n variate polynomial $f \in \mathbb{F}[\mathbf{x}]$ of degree $d \in [5, n]$, where $|\mathbf{x}| = n$, the problem of checking if there exist a*

[15] Throughout this article, poly(m) denotes a sufficiently large polynomial function in m; poly(m, β) is defined similarly.
[16] The first remark after Theorem 1b justifies this assumption.
[17] The stabilizer under the action of the general linear group.
[18] A variable set $\mathbf{x} = \{x_1, \ldots, x_m\}$ is treated as a column vector $(x_1 \ldots x_m)^T$ in the expression $B\mathbf{x} + \mathbf{b}$. The affine form entries of the column $B\mathbf{x} + \mathbf{b}$ are then plugged in place of the variables of $\mathsf{IMM}_{\mathbf{w},d}$ (following a variable ordering, like the one mentioned in Section 2.3).

$\mathbf{w} \in \mathbb{N}^{d-1}$ *and an invertible* $A \in \mathbb{F}^{n\times n}$ *such that* $f = \mathsf{IMM}_{\mathbf{w},d}(A\mathbf{x})$, *can be solved in randomized poly*(n, β) *time where* β *is the bit length of the coefficients of* f. *Further, with probability at least* $1 - \frac{1}{poly(n)}$ *the following holds: the algorithm returns a* $\mathbf{w}$, *and an invertible* $A \in \mathbb{F}^{n\times n}$ *such that* $f = \mathsf{IMM}_{\mathbf{w},d}(A\mathbf{x})$ *if such* $\mathbf{w}$ *and* A *exist, else it outputs "no such* $\mathbf{w}$ *and* A *exist."*

Remarks: Suppose $f = \mathsf{IMM}_{\mathbf{w},d}(A\mathbf{x})$, where A is an invertible matrix in $\mathbb{F}^{n\times n}$ and $\mathbf{w} = (w_1, w_2, \ldots, w_{d-1})$.

(1) *Irreducibility of* $\mathsf{IMM}_{\mathbf{w},d}$: We can assume without loss of generality that $w_k > 1$ for every $k \in [d-1]$, implying $\mathsf{IMM}_{\mathbf{w},d}$ is an irreducible polynomial. If $w_k = 1$ for some $k \in [d-1]$, then $\mathsf{IMM}_{\mathbf{w},d}$ is reducible, in which case we use the factorization algorithm in [23] to get blackbox access to the irreducible factors of f and then apply Theorem 1b to each of these irreducible factors (Section 1.6 has more details on this).

(2) *Uniqueness of* $\mathbf{w}$ *and* A: Assuming $w_k > 1$ for every $k \in [d-1]$, it would follow from the proof of the theorem that $\mathbf{w}$ is unique in the following sense: if $f = \mathsf{IMM}_{\mathbf{w}',d}(A'\mathbf{x})$, where $A' \in \mathbb{F}^{n\times n}$ is invertible, then either $\mathbf{w}' = \mathbf{w}$ or $\mathbf{w}' = (w_{d-1}, w_{d-2}, \ldots, w_1)$. Since $f = X_1 \cdot X_2 \ldots X_d = X_d^T \cdot X_{d-1}^T \ldots X_1^T$, $\mathbf{w}'$ can indeed be $(w_{d-1}, w_{d-2}, \ldots, w_1)$. The invertible transformation A is also unique up to the group of symmetries (see Definition 2.10) of $\mathsf{IMM}_{\mathbf{w},d}$: if $\mathsf{IMM}_{\mathbf{w},d}(A\mathbf{x}) = \mathsf{IMM}_{\mathbf{w},d}(A'\mathbf{x})$, then AA'^{-1} is in the group of symmetries of $\mathsf{IMM}_{\mathbf{w},d}$. In Section 6, we determine this group and show that $\mathsf{IMM}_{\mathbf{w},d}$ is characterized by it.

(3) *A related result in Reference [16]*: Another useful definition of the iterated matrix multiplication polynomial is the trace of a product of d $w \times w$ symbolic matrices—let us denote this polynomial by $\mathsf{IMM}'_{w,d}$. Both the variants, $\mathsf{IMM}'_{w,d}$ and $\mathsf{IMM}_{w,d}$, are well-studied in the literature and their circuit complexities are polynomially related. However, an equivalence test for one does not immediately give an equivalence test for the other. This is partly because the group of symmetries of $\mathsf{IMM}'_{w,d}$ and $\mathsf{IMM}_{w,d}$ are not exactly the same in nature (see Section 6 for a comparison).

Let $\mathbf{x}_1, \ldots, \mathbf{x}_d$ be the sets of variables in the d matrices of $\mathsf{IMM}'_{w,d}$, respectively. A polynomial $f(\mathbf{x}_1, \ldots, \mathbf{x}_d)$ is said to be *multilinearly equivalent* to $\mathsf{IMM}'_{w,d}$ if there are invertible $w \times w$ matrices $A_1, \ldots, A_d$ such that $f = \mathsf{IMM}'_{w,d}(A_1\mathbf{x}_1, \ldots, A_d\mathbf{x}_d)$. Grochow [16] showed the following result: Given the knowledge of the variable sets $\mathbf{x}_1, \ldots, \mathbf{x}_d$, an oracle to find roots of univariate polynomials over $\mathbb{C}$ and blackbox access to a polynomial f, there is a randomized algorithm to check whether f is multilinearly equivalent to $\mathsf{IMM}'_{w,d}$ using poly(w, d) operations over $\mathbb{C}$. Due to the issue of representing complex numbers, the model of computation for this result may be assumed to be the Blum-Shub-Smale model [8]. Theorem 1b is different from the result in Reference [16] in a few ways: First, the equivalence test is for $\mathsf{IMM}_{\mathbf{w},d}$ instead of $\mathsf{IMM}'_{w,d}$. The algorithm in Theorem 1b operates without the knowledge of the variable sets $\mathbf{x}_1, \ldots, \mathbf{x}_d$ (in fact, without the knowledge of $\mathbf{w}$). It only "sees" n variables $x_1, \ldots, x_n$ that are input to the blackbox for f. Second, there is no requirement of a oracle for finding roots of univariates. The base field is $\mathbb{Q}$ or a field with sufficiently large characteristic and the model of computation is the Turing machine model. Third, Theorem 1b gives a general equivalence test whereas the algorithm in Reference [16] checks only multilinear equivalence.

1.5 Discussion

To summarize, our main contribution is a polynomial time randomized equivalence test for $\mathsf{IMM}_{\mathbf{w},d}$, even if $\mathbf{w}$ is unknown. Although, equivalence testing is an important problem in its own right, Theorem 1 does not address the average-case ABP reconstruction problem quite satisfactorily because of the restriction $m \geq w^2 d$. The more interesting and challenging scenario is when $m \ll w^2 d$ in Problem 1, and this case remains an open problem. A partial progress is made in a subsequent work [28], where a reconstruction algorithm is given over finite fields assuming $m \geq w^2$.

1.6 Algorithm and Proof Strategy

An algorithm for reconstructing full rank ABP is given in Algorithm 1. At first, we trace the steps of this algorithm to show that proving Theorem 1a reduces to proving Theorem 1b using known methods. Then, we give an equivalence test for $\mathsf{IMM}_{\mathbf{w},d}$ in Algorithm 2, which is the contribution of this work. Some relevant definitions, notations and concepts can be found in Section 2.

Reduction to Equivalence Test for IMM

We are given blackbox access to an m variate polynomial $f(\tilde{\mathbf{x}})$ in Algorithm 1 where $\tilde{\mathbf{x}} = \{x_1, \ldots, x_m\}$. Suppose $f = \mathsf{IMM}_{\mathbf{w}',d}(B'\tilde{\mathbf{x}} + \mathbf{b}')$ for some unknown $\mathbf{w}' \in \mathbb{N}^{d-1}$, $\mathbf{b}' \in \mathbb{F}^n$ and $B' \in \mathbb{F}^{n\times m}$ of rank n, where n is the number of variables in $\mathsf{IMM}_{\mathbf{w}',d}$.

(i) *Variable reduction (Step 2)*: The number of essential/redundant variables of a polynomial remains unchanged under affine projection via full rank transformation. Since $\mathsf{IMM}_{\mathbf{w}',d}$ has no redundant variables,[19] the number of essential variables of f equals n. The algorithm eliminates the $m-n$ redundant variables in f by applying Algorithm 8 and constructs a $C \in \mathsf{GL}(m)$ such that $g = f(C\tilde{\mathbf{x}})$ has only the essential variables $\mathbf{x} = \{x_1, \ldots, x_n\}$. It follows that $g = \mathsf{IMM}_{\mathbf{w}',d}(A'\mathbf{x} + \mathbf{b}')$, where $A' \in \mathsf{GL}(n)$ is the matrix $B' \cdot C$ restricted to the first n columns.

(ii) *Equivalence test (Steps 5–9)*: Since $g = \mathsf{IMM}_{\mathbf{w}',d}(A'\mathbf{x} + \mathbf{b}')$, its dth homogeneous component $g^{[d]} = \mathsf{IMM}_{\mathbf{w}',d}(A'\mathbf{x})$. In other words, $g^{[d]}$ is equivalent to $\mathsf{IMM}_{\mathbf{w}',d}$ for an unknown $\mathbf{w}' \in \mathbb{N}^{d-1}$. At this point, the algorithm calls Algorithm 2 to find a $\mathbf{w}$ and an $A \in \mathsf{GL}(n)$ such that $g^{[d]} = \mathsf{IMM}_{\mathbf{w},d}(A\mathbf{x})$, and this is achieved with high probability.

(iii) *Finding a translation (Steps 12–17)*: Since g is equal to $\mathsf{IMM}_{\mathbf{w}',d}(A' \cdot (\mathbf{x} + A'^{-1}\mathbf{b}')) = g^{[d]}(\mathbf{x} + A'^{-1}\mathbf{b}')$, g is translation equivalent to $g^{[d]}$. With high probability, Algorithm 9 finds an $\mathbf{a} \in \mathbb{F}^n$ such that $g = g^{[d]}(\mathbf{x} + \mathbf{a})$, implying $g = \mathsf{IMM}_{\mathbf{w},d}(A\mathbf{x} + A\mathbf{a})$. Thus, $\mathbf{b} = A\mathbf{a}$ is a valid translation vector.

(iv) *Final reconstruction (Steps 20–26)*: From the previous steps, we have $g = \mathsf{IMM}_{\mathbf{w},d}(A\mathbf{x} + \mathbf{b})$. Although the variables $\{x_{n+1}, \ldots, x_m\}$ are absent in g, if we pretend that g is a polynomial in all the $\tilde{\mathbf{x}}$ variables then $g = \mathsf{IMM}_{\mathbf{w},d}(A_0\tilde{\mathbf{x}} + \mathbf{b})$, where A_0 is an $n \times m$ matrix such that the $n \times n$ submatrix formed by restricting to the first n columns of A_0 equals A and the remaining $m-n$ columns of A_0 have all zero entries. Hence, $f = g(C^{-1}\tilde{\mathbf{x}}) = \mathsf{IMM}_{\mathbf{w},d}(A_0 C^{-1}\tilde{\mathbf{x}} + \mathbf{b})$, which explains the setting $B = A_0 C^{-1}$ in step 20. The identity testing in steps 21-23 takes care of the situation when, to begin with, there are no $\mathbf{w}' \in \mathbb{N}^{d-1}$, $\mathbf{b}' \in \mathbb{F}^n$ and $B' \in \mathbb{F}^{n\times m}$ of rank n such that $f = \mathsf{IMM}_{\mathbf{w}',d}(B'\tilde{\mathbf{x}} + \mathbf{b}')$.

Equivalence test for IMM

Algorithm 1 calls Algorithm 2 on a blackbox holding a homogeneous n variate polynomial $f(\mathbf{x})$ of degree $d \leq n$, and expects a $\mathbf{w} \in \mathbb{N}^{d-1}$ and an $A \in \mathsf{GL}(n)$ in return such that $f = \mathsf{IMM}_{\mathbf{w},d}(A\mathbf{x})$, if such $\mathbf{w}$ and A exist. First, we argue that f can be assumed to be an irreducible polynomial.

[19] Which follows easily from Claim 2.3.

ALGORITHM 1: Reconstructing a full rank ABP

```
INPUT: Blackbox access to an $m$ variate polynomial $f(\tilde{\mathbf{x}})$ of degree $d \leq m$.
OUTPUT: A full rank ABP computing $f$ if such an ABP exists.
/* Variable reduction */
Use Algorithm 8 to compute $n$ and $C \in \mathsf{GL}(m)$ such that $g = f(C\tilde{\mathbf{x}})$ has only the essential variables
  $\mathbf{x} = \{x_1, \ldots, x_n\}$ of $f$. If $d > n$, then output "$f$ does not admit a full rank ABP" and stop.

/* Equivalence test: Finding $\mathbf{w}$ and $A$ */
Construct a blackbox for $g^{[d]}$, the $d$th homogeneous component of $g$ (see Section 2.2).
Use Algorithm 2 to find a $\mathbf{w} \in \mathbb{N}^{d-1}$ and an $A \in \mathsf{GL}(n)$ such that $g^{[d]} = \mathsf{IMM}_{\mathbf{w},d}(A\mathbf{x})$.
if Algorithm 2 outputs "no such $\mathbf{w}$ and $A$ exist" then
    Output "$f$ does not admit a full rank ABP" and stop.
end if

/* Finding a translation $\mathbf{b}$ */
Use Algorithm 9 to find an $\mathbf{a} \in \mathbb{F}^n$ such that $g = g^{[d]}(\mathbf{x} + \mathbf{a})$.
if Algorithm 9 outputs "$g$ is not translation equivalent to $g^{[d]}$" then
    Output "$f$ does not admit a full rank ABP" and stop.
else
    Set $\mathbf{b} = A\mathbf{a}$.
end if

/* Identity testing and final reconstruction */
Let $A_0$ be the $n \times m$ matrix obtained by attaching $m - n$ "all-zero" columns to the right of $A$. Set
  $B = A_0 C^{-1}$.
Choose a point $\mathbf{a} \in S^m$ at random, where $S \subseteq \mathbb{F}$ and $|S| \geq \mathrm{poly}(n)$.
if $f(\mathbf{a}) \neq \mathsf{IMM}_{\mathbf{w},d}(B\mathbf{a} + \mathbf{b})$ then
    Output "$f$ does not admit a full rank ABP" and stop.
else
    Construct a full rank ABP $\mathsf{A}$ of width $\mathbf{w}$ from $B$ and $\mathbf{b}$. Output $\mathsf{A}$.
end if
```

(a) *Assuming irreducibility of input f in Algorithm 2*: The idea is to construct blackbox access to the irreducible factors of f using the efficient randomized polynomial factorization algorithm in Reference [23], and compute full rank ABP for each of these irreducible factors. The ABPs are then connected "in series" to form a full rank ABP for f. This process succeeds with high probability. The details are as follows: If f is not square-free (which can be easily checked using [23]), then f cannot be equivalent to $\mathsf{IMM}_{\mathbf{w},d}$ for any $\mathbf{w}$, as $\mathsf{IMM}_{\mathbf{w},d}$ is always square-free. Suppose $f = f_1 \cdots f_k$, where $f_1, \ldots, f_k$ are distinct irreducible factors of f. If there are $\mathbf{w}' \in \mathbb{N}^{d-1}$ and $A' \in \mathsf{GL}(n)$ such that $f = \mathsf{IMM}_{\mathbf{w}',d}(A'\mathbf{x})$, then the number of essential variables in f is n (as $\mathsf{IMM}_{\mathbf{w}',d}$ has no redundant variables). Also, $f_1 \cdots f_k = h_1(A'\mathbf{x}) \cdots h_k(A'\mathbf{x})$ where $h_1, \ldots, h_k$ are the irreducible factors of $\mathsf{IMM}_{\mathbf{w}',d}$. The irreducible factors of $\mathsf{IMM}_{\mathbf{w}',d}$ are "smaller IMMs" in disjoint sets of variables.[20] Hence, by uniqueness of factorization, f_ℓ is computable by a full rank ABP for every $\ell \in [k]$. Let the degree of f_ℓ be d_ℓ and n_ℓ the number of essential variables in f_ℓ. Then $n_1 + \cdots + n_k = n$. Now observe that if we invoke Algorithm 1 on input f_ℓ, it calls Algorithm 2 from within on an irreducible polynomial, as f_ℓ is homogeneous

[20] Recall, $\mathsf{IMM}_{\mathbf{w},d}$ is irreducible if $w_k > 1$ for every $k \in [d-1]$ where $\mathbf{w} = (w_1, \ldots, w_{d-1})$.

and irreducible. Algorithm 1 returns a $\mathbf{w}_\ell \in \mathbb{N}^{d_\ell - 1}$ and $B_\ell \in \mathbb{F}^{n_\ell \times n}$ of rank n_ℓ such that $f_\ell = \mathsf{IMM}_{\mathbf{w}_\ell, d_\ell}(B_\ell \mathbf{x})$ (ignoring the translation vector as f_ℓ is homogeneous). Let $\mathbf{w} \in \mathbb{N}^{d-1}$ be the vector $(\mathbf{w}_1 \; 1 \; \mathbf{w}_2 \; 1 \ldots \; 1 \; \mathbf{w}_k)$,[21] and $A \in \mathbb{F}^{n \times n}$ such that the first n_1 rows of A is B_1, next n_2 rows is B_2, and so on till last n_k rows is B_k. Then, $f = \mathsf{IMM}_{\mathbf{w},d}(A\mathbf{x})$. Clearly, A must be in $\mathsf{GL}(n)$ as the number of essential variables of f is n. Thus, it is sufficient to describe Algorithm 2 on an input f that is irreducible.

(b) *A comparison with Reference [26] and our proof strategy*: Reference [26] gave equivalence tests for the permanent and determinant polynomials by making use of their Lie algebra (see Definition 2.11). Algorithm 2 also involves Lie algebra of IMM, but there are some crucial differences in the way Lie algebra is used in Reference [26] and in here. The Lie algebra of permanent consists of diagonal matrices and hence commutative. By diagonalizing a basis of $\mathfrak{g}_f$ over $\mathbb{C}$, for an f equivalent to permanent, we can reduce the problem to the much simpler permutation and scaling (PS) equivalence problem. The Lie algebra of $n \times n$ determinant, which is isomorphic to $\mathfrak{sl}_n \oplus \mathfrak{sl}_n$, is not commutative. However, a Cartan subalgebra of $\mathfrak{sl}_n$ consists of traceless diagonal matrices. This then helps reduce the problem to PS-equivalence by diagonalizing (over $\mathbb{C}$) a basis of the centralizer of a random element in $\mathfrak{g}_f$, for an f equivalent to determinant. Both the equivalence tests involve simultaneous diagonalization of matrices over $\mathbb{C}$. It is a bit unclear how to carry through this step if the base field is $\mathbb{Q}$ and we insist on low bit complexity. The Lie algebra of IMM is not commutative. Also, we do not know if going to Cartan subalgebra helps, as we would like to avoid the simultaneous diagonalization step. Instead of Cartan subalgebras, we study invariant subspaces (Definition 2.4) of the Lie algebra $\mathfrak{g}_{\mathsf{IMM}}$. A detailed analysis of the Lie algebra (in Section 3) reveals the structure of the irreducible invariant subspaces of $\mathfrak{g}_{\mathsf{IMM}}$. It is observed that these invariant subspaces are intimately connected to the layer spaces (see Definition 2.6) of any full rank ABP computing f. At a conceptual level, this connection helps us reconstruct a full rank ABP. Once we have access to the layer spaces, we can retrieve the unknown width vector $\mathbf{w}$ whence the problem reduces to the easier problem of reconstructing an almost set-multilinear ABP (Definition 2.15).

We now give some more details on Algorithm 2. Suppose there is a $\mathbf{w} \in \mathbb{N}^{d-1}$ such that f is equivalent to $\mathsf{IMM}_{\mathbf{w},d}$. The algorithm has four main steps:

(i) *Computing irreducible invariant subspaces (Steps 2–6):* The algorithm starts by computing a basis of the Lie algebra $\mathfrak{g}_f$. It then invokes Algorithm 3 to compute bases of the d irreducible invariant subspaces of $\mathfrak{g}_f$. Algorithm 3 works by picking a random element R' in $\mathfrak{g}_f$ and factoring its characteristic polynomial $h = g_1 \cdots g_s$. By computing the closure of vectors (Definition 2.5) picked from null spaces of $g_1(R'), \ldots, g_s(R')$, the algorithm is able to find bases of the required invariant spaces.

(ii) *Computing layer spaces (Step 9)*: The direct relation between the irreducible invariant spaces of $\mathfrak{g}_{\mathsf{IMM}}$ and the layers spaces of any full rank ABP computing f (as shown in Lemma 5.2) is exploited by Algorithm 5 to compute bases of these layer spaces. This also helps establish that all the layer spaces, except two of them, are “unique” (see Lemma 5.1). The second and second-to-last layer spaces of a full rank ABP are *not* unique; however the bigger space spanned by the first two layer spaces (similarly the last two layer spaces) is unique. Algorithm 5 finds bases for these two bigger spaces along with the $d - 2$ remaining layer spaces.

[21] The notation means the entries of $\mathbf{w}_1$ are followed by 1, followed by the entries of $\mathbf{w}_2$, then a 1 again, and so on.

(iii) *Reduction to almost set-multilinear ABP (Steps 12–15)*: The layer spaces are then correctly reordered in Algorithm 6 using a randomized procedure to compute the appropriate evaluation dimensions (Definition 2.7). The reordering also yields a valid width vector $\mathbf{w}$. At this point, the problem easily reduces to reconstructing a full rank almost set-multilinear ABP by mapping the bases of the layer spaces to distinct variables. This mapping gives an $\widehat{A} \in \mathsf{GL}(n)$ such that $f(\widehat{A}\mathbf{x})$ is computable by a full rank almost set-multilinear ABP of width $\mathbf{w}$. It is "almost set-multilinear" (and not "set-multilinear") as the second and the second-to-last layer spaces are unavailable; instead, two bigger spaces are available as mentioned above.

(iv) *Reconstructing a full rank almost set-mutlilinear ABP (Steps 18–22)*: Finally, we reconstruct a full rank almost set-mutlilinear ABP computing $f(\widehat{A}\mathbf{x})$ using Algorithm 7. This algorithm is inspired by a similar algorithm for reconstructing set-multilinear ABP in Reference [30], but it is a little different from the latter as we are dealing with an "almost" set-multilinear ABP. The reconstructed ABP readily gives an $A \in \mathsf{GL}(n)$ such that $f = \mathsf{IMM}_{\mathbf{w},d}(A\mathbf{x})$.

A final identity testing (Steps 25–30) takes care of the situation when, to begin with, there is no $\mathbf{w} \in \mathbb{N}^{d-1}$ that makes f equivalent to $\mathsf{IMM}_{\mathbf{w},d}$.

2 PRELIMINARIES

2.1 Notations and Definitions

The group of invertible $n \times n$ matrices over $\mathbb{F}$ is represented by $\mathsf{GL}(n, \mathbb{F})$. Since $\mathbb{F}$ is fixed to be the field of rationals, we omit $\mathbb{F}$ and write $\mathsf{GL}(n)$. Natural numbers are denoted by $\mathbb{N} = \{1, 2, \ldots\}$. As a convention, we use $\mathbf{x}, \mathbf{y}$ and $\mathbf{z}$ to denote sets of variables, capital letters A, B, C, and so on to denote matrices, calligraphic letters like $\mathcal{U}, \mathcal{V}, \mathcal{W}$ to denote vector spaces over $\mathbb{F}$, and bold small letters like $\mathbf{u}, \mathbf{v}, \mathbf{w}$ to denote vectors in these spaces. All vectors considered in this article are column vectors, unless mentioned otherwise. An affine form in $\mathbf{x} = \{x_1, x_2, \ldots, x_n\}$ variables is $a_0 + \sum_{i=1}^{n} a_i x_i$ where for $i \in [0, d]$ $a_i \in \mathbb{F}$, and if $a_0 = 0$, then we call it a *linear form*. The first-order partial derivative of the polynomial $f(\mathbf{x})$ with respect to x_i is denoted as $\partial_{x_i}(f(\mathbf{x}))$. Below we set up some notations and terminologies.

1. Linear Algebra:

Definition 2.1 (Direct Sum). Let $\mathcal{U}, \mathcal{W}$ be subspaces of a vector space $\mathcal{V}$. Then $\mathcal{V}$ is said to be the *direct sum* of $\mathcal{U}$ and $\mathcal{W}$ denoted $\mathcal{V} = \mathcal{U} \oplus \mathcal{W}$, if $V = \mathcal{U} + \mathcal{W}$ and $\mathcal{U} \cap \mathcal{W} = \{\mathbf{0}\}$.

For $\mathcal{U}, \mathcal{W}$ subspaces of a vector space $\mathcal{V}$, $\mathcal{V} = \mathcal{U} \oplus \mathcal{W}$ if and only if for every $\mathbf{v} \in \mathcal{V}$ there exist unique $\mathbf{u} \in \mathcal{U}$ and $\mathbf{w} \in \mathcal{W}$ such that $\mathbf{v} = \mathbf{u} + \mathbf{w}$. Hence, $\dim(\mathcal{V}) = \dim(\mathcal{U}) + \dim(\mathcal{W})$.

Definition 2.2 (Null Space). *Null space* $\mathcal{N}$ of a matrix $M \in \mathbb{F}^{n \times n}$ is the space of all vectors $\mathbf{v} \in \mathbb{F}^n$, such that $M\mathbf{v} = \mathbf{0}$.

Definition 2.3 (Coordinate Subspace). Let $e_i = (0, \ldots, 1, \ldots, 0)$ be the unit vector in $\mathbb{F}^n$ with 1 at the ith position and all other coordinates zero. A *coordinate subspace* of $\mathbb{F}^n$ is a space spanned by a subset of the n unit vectors $\{e_1, e_2, \ldots, e_n\}$.

Definition 2.4 (Invariant Subspace). Let $M_1, M_2, \ldots, M_k \in \mathbb{F}^{n \times n}$. A subspace $\mathcal{U} \subseteq \mathbb{F}^n$ is called an *invariant subspace* of $\{M_1, M_2, \ldots, M_k\}$ if $M_i\,\mathcal{U} \subseteq \mathcal{U}$ for every $i \in [k]$. A nonzero invariant subspace $\mathcal{U}$ is *irreducible* if there are no invariant subspaces $\mathcal{U}_1$ and $\mathcal{U}_2$ such that $\mathcal{U} = \mathcal{U}_1 \oplus \mathcal{U}_2$, where $\mathcal{U}_1$ and $\mathcal{U}_2$ are properly contained in $\mathcal{U}$.

The following observation is immediate.

ALGORITHM 2: Equivalence test for IMM

```
INPUT: Blackbox access to a homogeneous $n$ variate degree $d$ polynomial $f$ (which can be assumed to be irreducible without any loss of generality).
OUTPUT: A $\mathbf{w} \in \mathbb{N}^{d-1}$ and an $A \in \mathsf{GL}(n)$ such that $f = \mathsf{IMM}_{\mathbf{w},d}(A\mathbf{x})$, if such $\mathbf{w}$ and $A$ exist.
/* Finding irreducible invariant subspaces */
Compute a basis of the Lie algebra $\mathfrak{g}_f$. (See Section 2.2.)
Use Algorithm 3 to compute the bases of the irreducible invariant subspaces of $\mathfrak{g}_f$.
if Algorithm 3 outputs "Fail" then
    Output "no such $\mathbf{w}$ and $A$ exist" and stop.
end if

/* Finding layer spaces from irreducible invariant subspaces */
Use Algorithm 5 to compute bases of the layer spaces of a full rank ABP computing $f$, if such an ABP exists.

/* Reduction to almost set-multilinear ABP: Finding $\mathbf{w}$ */
Use Algorithm 6 to compute a $\mathbf{w} \in \mathbb{N}^{d-1}$ and an $\widehat{A} \in \mathsf{GL}(n)$ such that $h = f(\widehat{A}\mathbf{x})$ is computable by a full rank almost set-multilinear ABP of width $\mathbf{w}$.
if Algorithm 6 outputs "Fail" then
    Output "no such $\mathbf{w}$ and $A$ exist" and stop.
end if

/* Reconstructing an almost set-multilinear ABP: Finding $A$ */
Use Algorithm 7 to reconstruct a full rank almost set-multilinear ABP $\mathsf{A}$' computing $h$.
if Algorithm 7 outputs "Fail" then
    Output "no such $\mathbf{w}$ and $A$ exist" and stop.
end if
Replace the $\mathbf{x}$ variables in $\mathsf{A}$' by $\widehat{A}^{-1}\mathbf{x}$ to obtain a full rank ABP $\mathsf{A}$. Compute $A \in \mathsf{GL}(n)$ from $\mathsf{A}$.

/* Final identity testing */
Choose a point $\mathbf{a} \in S^n$, where $S \subseteq \mathbb{F}$ and $|S| \geq \text{poly}(n)$.
if $f(\mathbf{a}) \neq \mathsf{IMM}_{\mathbf{w},d}(A\mathbf{a})$ then
    Output "no such $\mathbf{w}$ and $A$ exist" and stop.
else
    Output $\mathbf{w}$ and $A$.
end if
```

Observation 2.1. *If $\mathcal{U}$ is an invariant subspace of $\{M_1, M_2, \ldots, M_k\}$, then for every $M \in \mathcal{L} \overset{\text{def}}{=} \text{span}_{\mathbb{F}}\{M_1, M_2, \ldots, M_k\}$, $M\,\mathcal{U} \subseteq \mathcal{U}$. Hence, we say $\mathcal{U}$ is an invariant subspace of $\mathcal{L}$, a space generated by matrices.*

Definition 2.5 (Closure of a Vector). The *closure of a vector* $\mathbf{v} \in \mathbb{F}^n$ under the action of a space $\mathcal{L}$ spanned by a set of $n \times n$ matrices is the smallest invariant subspace of $\mathcal{L}$ containing $\mathbf{v}$.

Here, "smallest" is with regard to dimension of invariant subspaces. Since intersection of two invariant subspaces is also an invariant subspace of $\mathcal{L}$, the smallest invariant subspace of $\mathcal{L}$ containing $\mathbf{v}$ is unique and is contained in every invariant subspace of $\mathcal{L}$ containing $\mathbf{v}$. Algorithm 4 in Section 4.2 computes the closure of a given vector $\mathbf{v}$ under the action of $\mathcal{L}$ whose basis is given.

By identifying a linear form $\sum_{i=1}^{n} a_i x_i$ with the vector $(a_1, \ldots, a_n) \in \mathbb{F}^n$ (and vice versa), we can associate the following vector spaces with an ABP.

Definition 2.6 (Layer Spaces of an ABP). Let $X_1 \cdot X_2 \ldots X_d$ be a full rank ABP A of length d and width $\mathbf{w} = (w_1, w_2, \ldots, w_{d-1})$, where X_1 to X_d are as in Definition 1.3. Let $\mathcal{X}_i$ be the vector space in $\mathbb{F}^n$ spanned by the homogeneous degree 1 parts of the affine forms[22] in X_i for $i \in [d]$; the spaces $\mathcal{X}_1, \mathcal{X}_2, \ldots, \mathcal{X}_d$ are called the *layer spaces* of A.

2. Evaluation dimension: The rank of the partial derivative matrix of a polynomial f was introduced in Reference [35] and used subsequently in several works on lower bound, polynomial identity testing and circuit reconstruction (see Reference [40]). The following definition (which makes the notion well defined for fields of finite characteristic) appears in Reference [13].[23]

Definition 2.7 (Evaluation Dimension). The *evaluation dimension* of a polynomial $g \in \mathbb{F}[\mathbf{x}]$ with respect to a set $\mathbf{x}' \subseteq \mathbf{x}$, denoted as $\mathrm{Evaldim}_{\mathbf{x}'}(g)$, is defined as

$$\dim(\mathrm{span}_{\mathbb{F}}\{g(\mathbf{x})|_{\forall x_j \in \mathbf{x}' x_j = \alpha_j} : \alpha_j \in \mathbb{F} \text{ for every } x_j \in \mathbf{x}'\}).$$

3. Affine projection and equivalence testing: Studying polynomials by applying linear transformations (from suitable matrix groups) on the variables is at the heart of invariant theory.

Definition 2.8 (Affine Projection). An m variate polynomial f is an *affine projection* of a n variate polynomial g, if there exists a matrix $A \in \mathbb{F}^{n \times m}$ and a $\mathbf{b} \in \mathbb{F}^n$ such that $f(\mathbf{x}) = g(A\mathbf{x} + \mathbf{b})$.

In Reference [26], it was shown that given an m variate polynomial f and an n variate polynomial g, checking whether f is an affine projection of g is NP-hard, even if f and g are given in the dense representation (that is as list of coefficients of the monomials). In the above definition, we say f is an affine projection of g via a *full rank transformation*, if $m \geq n$ and A has rank n. In the affine projection via full rank transformation problem, we are given an m variate polynomial f and an n variate polynomial g in some suitable representation, and we need to determine if f is an affine projection of g via a full rank transformation. References [25, 26] studied the affine projection via full rank transformation problem for g coming from fixed families and gave polynomial time randomized algorithms to check whether a degree d polynomial f given as blackbox is an affine projection of g via a full rank transformation, where g is the elementary symmetric polynomial/permanent/determinant/power symmetric polynomial or sum-of-products polynomial. As observed in Reference [26], variable reduction and translation equivalence test (described in Section 2.2) help reduce the affine projection via full rank transformation problem to equivalence testing (see also Section 1.6).

Definition 2.9 (Equivalent Polynomials). An n variate polynomial f is *equivalent* to an n variate polynomial g, if there exists a matrix $A \in \mathsf{GL}(n)$ such that $f(\mathbf{x}) = g(A\mathbf{x})$.

The equivalence testing problem asks us to check if two n variate polynomials f and g (given in some suitable representation) are equivalent. This problem is at least as hard as the graph isomorphism problem even when f and g are cubic forms given in dense representation [3]. There is a cryptographic application [36] that assumes the problem is hard also in the *average-case* for bounded degree f and g given in dense representation. If we restrict to checking if f and g are equivalent via a permutation matrix A, then the problem is shown to be in $\mathsf{NP} \cap \mathsf{coAM}$ [42].

4. Group of symmetries and Lie algebra:

Definition 2.10 (Group of Symmetries). The *group of symmetries* of a polynomial $g \in \mathbb{F}[\mathbf{x}]$ in n variables, denoted as $\mathscr{G}_g$, is the set of all $A \in \mathsf{GL}(n)$ such that $g(A\mathbf{x}) = g(\mathbf{x})$.

[22] Identify linear forms with vectors in $\mathbb{F}^n$ as mentioned above.

[23] They attributed the definition to Ramprasad Saptharishi.

The proof of Theorem 1b involves an analysis of the Lie algebra of the group of symmetries of $\mathsf{IMM}_{\mathrm{w},d}$. We will abuse terminology slightly and say the Lie algebra of a polynomial to mean the Lie algebra of the group of symmetries of the polynomial. We will work with the following definition of Lie algebra of a polynomial (see Reference [26]).

Definition 2.11 (Lie Algebra of a Polynomial). The *Lie algebra* of a polynomial $f \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ denoted as $\mathfrak{g}_f$ is the set of all $n \times n$ matrices $E = (e_{ij})_{i,j\in[n]}$ in $\mathbb{F}^{n\times n}$ such that $\sum_{i,j\in[n]} e_{ij}x_j \cdot \frac{\partial f}{\partial x_i} = 0$.

Remark: Observe that $\mathfrak{g}_f$ is a subspace of $\mathbb{F}^{n\times n}$. It can also be shown that the space $\mathfrak{g}_f$ satisfies the *Lie bracket property*: For any $E_1, E_2 \in \mathfrak{g}_f$, $[E_1, E_2] \overset{\text{def}}{=} E_1E_2 - E_2E_1$ is also in $\mathfrak{g}_f$. We would not be needing this property, but would just use the vector space feature of $\mathfrak{g}_f$. The proof of the following well known fact is given in Reference [26], see also Section 7.2 for a proof.

Claim 2.1. *If $f(\mathbf{x}) = g(A\mathbf{x})$, where f and g are both n variate polynomials and $A \in \mathsf{GL}(n)$, then the Lie algebra of f is a conjugate of the Lie algebra of g via A, i.e., $\mathfrak{g}_f = \{A^{-1}EA : E \in \mathfrak{g}_g\} =: A^{-1}\mathfrak{g}_g A$.*

The following observation relates the invariant subspaces of the Lie algebras of two equivalent polynomials.

Observation 2.2. *Suppose $f(\mathbf{x}) = g(A\mathbf{x})$, where $\mathbf{x} = \{x_1, x_2, \ldots, x_n\}$ and $A \in \mathsf{GL}(n)$. Then $\mathcal{U} \in \mathbb{F}^n$ is an invariant subspace of $\mathfrak{g}_g$ if and only if $A^{-1}\mathcal{U}$ is an invariant subspace of $\mathfrak{g}_f$.*

Proof. $\mathcal{U}$ is an invariant subspace of $\mathfrak{g}_g$ implies, for all $E \in \mathfrak{g}_g$, $E\,\mathcal{U} \subseteq \mathcal{U}$. Consider $E' \in \mathfrak{g}_f$, using Claim 2.1, we know there exists $E \in \mathfrak{g}_g$ such that $AE'A^{-1} = E$. Since $\mathcal{U}$ is an invariant subspace of $AE'A^{-1}$, $A^{-1}\mathcal{U}$ is an invariant subspace of E'. The proof of the other direction is similar. □

2.2 Algorithmic Preliminaries

We record some of the basic algorithmic tasks on polynomials that can be performed efficiently and that we require at different places in our algorithms and proofs.

1. Computing homogeneous components of f: The ith homogeneous component (or the homogeneous degree i part) of a degree d polynomial f, denoted as $f^{[i]}$ is the sum of the degree i monomials with coefficients as in f. Clearly, $f = f^{[d]} + f^{[d-1]} + \cdots + f^{[0]}$. Given an n variate degree d polynomial f as a blackbox, there is an efficient algorithm to compute blackboxes for the d homogeneous components of f. The idea is to multiply each variable by a new formal variable t, and then interpolate the coefficients of $t^0, t^1, \ldots, t^d$; the coefficient of t^i is $f^{[i]}$.

2. Computing derivatives of f: Given a polynomial $f(x_1, x_2, \ldots, x_n)$ of degree d as a blackbox, we can efficiently construct blackboxes for the derivatives $\partial_{x_i} f$, for all $i \in [n]$. The following observation suggests that it is sufficient to construct blackboxes for certain homogeneous components.

Observation 2.3. *If $g(x_1, x_2, \ldots, x_n)$ is a homogeneous polynomial of degree d, then for all $i \in [n]$ $\partial_{x_i} g = \sum_{j=1}^{d} j \cdot x_i^{j-1}[g(x_1, x_2, \ldots, x_{i-1}, 1, x_{i+1}, \ldots, x_n)]^{[d-j]}$.*

For every $i \in [n]$, constructing a blackbox for $\partial_{x_i} f$ is immediate from the above observation as $\partial_{x_i} f = \partial_{x_i} f^{[d]} + \partial_{x_i} f^{[d-1]} + \cdots + \partial_{x_i} f^{[1]}$.

3. Space of linear dependencies of polynomials: Let $f_1, f_2, \ldots, f_m$ be n variate polynomials in $\mathbb{F}[\mathbf{x}]$ with degree bounded by d. The set $\mathcal{U} = \{(a_1\ a_2 \ldots\ a_m)^T \in \mathbb{F}^m \mid \sum_{j\in[m]} a_j f_j = 0\}$, called the space of $\mathbb{F}$-linear dependencies of $f_1, f_2, \ldots, f_m$ is a subspace of $\mathbb{F}^m$. We would like to find a basis of the space $\mathcal{U}$ given blackbox access to $f_1, f_2, \ldots, f_m$. Suppose the dimension of the $\mathbb{F}$-linear space spanned by the polynomials $f_1, f_2, \ldots, f_m$ is $m - r$ then $\dim(\mathcal{U}) = r$. An algorithm to find a basis of $\mathcal{U}$ can be derived from the following claim.

Claim 2.2. *With probability at least* $1 - \frac{1}{\text{poly}(n)}$, *the rank of the matrix* $M = (f_j(\mathbf{b}_i))_{i,j \in [m]}$ *is* $m - r$ *where* $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_m$ *are chosen independently and uniformly at random from* $S^n \subset \mathbb{F}^n$ *with* $|S| = dm \cdot poly(n)$.

The proof of the claim that involves an application of the Schwartz-Zippel lemma is given in Section 7.2. The space $\mathcal{U}$ equals the null space of M with high probability.

4. Eliminating redundant variables:

Definition 2.12 (Essential and Redundant Variables). We say an n variate polynomial f has s *essential variables* if there exists an $A \in \text{GL}(n)$ such that $f(A\mathbf{x})$ is an s variate polynomial and there exists no $A' \in \text{GL}(n)$ such that $f(A'\mathbf{x})$ is a t variate polynomial where $t < s$. An n variate polynomial has r *redundant variables* if it has $s = n - r$ essential variables.

If the number of essential variables in a polynomial $f(x_1, x_2, \ldots, x_n)$ is s, then without loss of generality, we can assume that the first s variables $x_1, x_2, \ldots, x_s$ are essential variables and the remaining variables are redundant. An algorithm to eliminate the redundant variables of a polynomial was considered in Reference [10], and it was shown that if the coefficients of a polynomial are given as input, then we can eliminate the redundant variables in polynomial time. Further, Reference [25] gave an efficient randomized algorithm to eliminate the redundant variables in a polynomial given as blackbox. For completeness, we give the algorithm in Reference [25] as part of the following claim.

Claim 2.3. *Let* r *be the number of redundant variables in an* n *variate polynomial* f *of degree* d. *Then the dimension of the space* $\mathcal{U}$ *of* $\mathbb{F}$*-linear dependencies of* $\{\partial_{x_i} f \mid i \in [n]\}$ *is* r. *Moreover, we can construct an* $A \in \text{GL}(n)$ *in randomized poly*(n, d, β) *time such that* $f(A\mathbf{x})$ *is free of the set of variables* $\{x_{n-r+1}, x_{n-r+2}, \ldots, x_n\}$, *where* β *is the bit length of the coefficients of* f.

The proof is given in Section 7.2.

5. Efficient translation equivalence test: Two n variate degree d polynomials $f, g \in \mathbb{F}[\mathbf{x}]$ are *translation equivalent* (also called shift equivalent in Reference [12]) if there exists a point $\mathbf{a} \in \mathbb{F}^n$ such that $f(\mathbf{x} + \mathbf{a}) = g(\mathbf{x})$. Translation equivalence test takes input blackbox access to two n variate polynomials f and g, and outputs an $\mathbf{a} \in \mathbb{F}^n$ such that $f(\mathbf{x} + \mathbf{a}) = g(\mathbf{x})$ if f and g are translation equivalent else outputs "f and g are not translation equivalent." As before, let β be the bit lengths of the coefficients of f and g. A randomized poly(n, d, β) time algorithm is presented in Reference [12] to test translation equivalence and find an $\mathbf{a} \in \mathbb{F}^n$ such that $f(\mathbf{x} + \mathbf{a}) = g(\mathbf{x})$, if such an $\mathbf{a}$ exists. Another randomized test was mentioned in Reference [26], which we present as proof of the following lemma in Section 7.2.

Lemma 2.13. *There is a randomized algorithm that takes input blackbox access to two* n *variate, degree* d *polynomials* f *and* g, *and with probability at least* $1 - \frac{1}{\text{poly}(n)}$ *does the following: if* f *is translation equivalent to* g, *outputs an* $\mathbf{a} \in \mathbb{F}^n$ *such that* $f(\mathbf{x} + \mathbf{a}) = g(\mathbf{x})$, *else outputs "*$f$ *and* g *are not translation equivalent." The running time of the algorithm is poly*(n, d, β), *where* β *is the bit length of the coefficients of* f *and* g.

6. Computing basis of Lie algebra: The proof of the following lemma is given in Reference [26], for completeness we include a proof in Section 7.2.

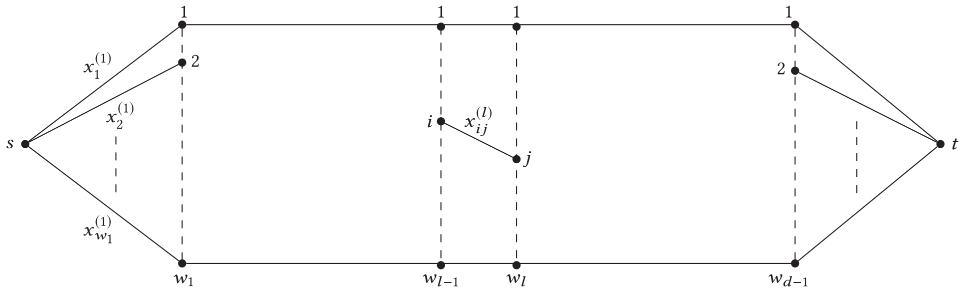

Fig. 1. Naming of variables in $\mathsf{IMM}_{\mathbf{w},d}$.

Lemma 2.14. *There is a randomized algorithm that when given blackbox access to an n variate degree d polynomial f, computes a basis of $\mathfrak{g}_f$ with probability at least $1 - \frac{1}{\text{poly}(n)}$ in time poly(n, d, β) where β is the bit length of the coefficients in f.*

2.3 Iterated Matrix Multiplication Polynomial

Let $\mathbf{w} = (w_1, w_2, \ldots, w_{d-1}) \subseteq \mathbb{N}^{d-1}$. Suppose $Q_1 = (x_1^{(1)}\ x_2^{(1)}\ \ldots\ x_{w_1}^{(1)})$, $Q_d^T = (x_1^{(d)}\ x_2^{(d)}\ \ldots\ x_{w_{d-1}}^{(d)})$ be row vectors, and for $k \in [2, d-1]$, $Q_k = (x_{ij}^{(k)})_{i \in [w_{k-1}],\, j \in [w_k]}$ be a $w_{k-1} \times w_k$ matrix, where for $i \in [w_1]$ $x_i^{(1)}$, for $i \in [w_{d-1}]$ $x_i^{(d)}$ and for $i \in [w_{k-1}], j \in [w_k]$ $x_{ij}^{(k)}$ are distinct variables. The iterated matrix multiplication polynomial $\mathsf{IMM}_{\mathbf{w},d}$ is the entry of the 1×1 matrix obtained from the product $\prod_{i=1}^{d} Q_i$. When d and $\mathbf{w}$ are clear from the context, we drop the subscripts and simply represent it by IMM. For all $k \in [d]$, we denote the set of variables in Q_k as $\mathbf{x}_k$; Figure 1 depicts an ABP computing $\mathsf{IMM}_{\mathbf{w},d}$ when the width is uniform, that is $w_1 = w_2 = \cdots = w_{d-1}$.

Ordering of variables in $\mathsf{IMM}_{\mathbf{w},d}$: From here on, we will assume that the variables $\mathbf{x}_1 \uplus \mathbf{x}_2 \uplus \cdots \uplus \mathbf{x}_d$ are ordered as follows: For $i < j$, the $\mathbf{x}_i$ variables have precedence over the $\mathbf{x}_j$ variables. Among the $\mathbf{x}_l$ variables, we follow column-major ordering, i.e., $x_{11}^{(l)} \succ \cdots \succ x_{w_{l-1}1}^{(l)} \succ \cdots \succ x_{1w_l}^{(l)} \succ \cdots \succ x_{w_{l-1}w_l}^{(l)}$. We would also refer to the variables of IMM as $\mathbf{x} = \{x_1, x_2, \ldots, x_n\}$ where x_i is the ith variable according to this ordering,[24] and $n = w_1 + \sum_{k=2}^{d-1} w_{k-1}w_k + w_{d-1}$ is the total number of variables in IMM. For $A \in \mathbb{F}^{n \times n}$, we can naturally index the rows and columns of A by the $\mathbf{x}$ variables such that the ith row or column is indexed by the ith variable.

2.4 Almost Set-multilinear ABP and a Canonical Representation

In the proof of Theorem 1b, we eventually reduce the equivalence test problem to checking whether there exists an $A \in \mathsf{GL}(n)$, such that an input polynomial $h(\mathbf{x})$ (given as blackbox) equals $\mathsf{IMM}_{\mathbf{w},d}(A\mathbf{x})$, where $\mathbf{w}$ is known, $\mathbf{x}$ is the variables of $\mathsf{IMM}_{\mathbf{w},d}$, and A satisfies the following properties:

(1) For all $k \in [d] \setminus \{2, d-1\}$, the rows indexed by $\mathbf{x}_k$ variables contain zero entries in columns indexed by variables other than $\mathbf{x}_k$.
(2) The rows indexed by $\mathbf{x}_2$ and $\mathbf{x}_{d-1}$ variables contain zero entries in columns indexed by variables other than $\mathbf{x}_1 \uplus \mathbf{x}_2$ and $\mathbf{x}_{d-1} \uplus \mathbf{x}_d$, respectively.

[24] The justification for identifying the variables $\mathbf{x}$ of f with the variables of $\mathsf{IMM}_{\mathbf{w},d}$ in this order is as follows: If f is equivalent to $\mathsf{IMM}_{\mathbf{w},d}$, then f is also equivalent to $\mathsf{IMM}_{\mathbf{w},d}(\mathbf{x})$ whose variables $\{x_1, \ldots, x_n\}$ are ordered as above. That $\mathbf{w}$ is *a priori* unknown to Algorithm 2 does not matter here.

If there exists such a block-diagonal matrix A, then we say h is computed by a *full rank almost set-multilinear* ABP as defined below.

Definition 2.15 (Full Rank Almost Set-Multilinear ABP). A *full rank almost set-multilinear* ABP of width $\mathbf{w} = (w_1, w_2, \ldots, w_{d-1})$ and length d is a product of d matrices, $X_1 \cdot X_2 \ldots X_d$, where X_k's are as in Definition 1.3 but with linear forms as entries. The linear forms in X_k are in $\mathbf{x}_k$ variables, for all $k \in [d] \setminus \{2, d-1\}$, and for X_2 and X_{d-1} the linear forms are in $\mathbf{x}_1 \uplus \mathbf{x}_2$ and $\mathbf{x}_{d-1} \uplus \mathbf{x}_d$ variables, respectively, where $\mathbf{x}_1 \uplus \mathbf{x}_2 \cdots \uplus \mathbf{x}_d$ is the set of variables in $\mathsf{IMM}_{\mathbf{w},d}$.

Conventionally, in the definition of set-multilinear ABP, the entries of X_i are linear forms in just $\mathbf{x}_i$ variables—the ABP in the above definition is almost set-multilinear as matrices X_2 and X_{d-1} violate this condition. An efficient randomized reconstruction algorithm for set-multilinear ABP follows from Reference [30]. To apply a similar reconstruction algorithm to full rank almost set-multilinear ABPs, we fix a canonical representation for the first two and the last two matrices as explained below.

Canonical form or representation: We say a full rank almost set-multilinear ABP of width $\mathbf{w}$ is in *canonical form* if the following hold:

(1a) $X_1 = (x_1^{(1)}\ x_2^{(1)} \ldots x_{w_1}^{(1)})$,

(1b) the linear forms in X_2 are such that for $l, i \in [w_1]$ and $l < i$, the variable $x_l^{(1)}$ has a zero coefficient in the (i, j)th entry (linear form) of X_2, where $j \in [w_2]$.

(2a) $X_d = (x_1^{(d)}\ x_2^{(d)} \ldots x_{w_{d-1}}^{(d)})^T$,

(2b) the linear forms in X_{d-1} are such that for $l, j \in [w_{d-1}]$ and $l < j$, the variable $x_l^{(d)}$ has a zero coefficient in the (i, j)th entry (linear form) of X_{d-1}, where $i \in [w_{d-2}]$.

The following claim states that for every full rank almost set-multilinear ABP there is another ABP in canonical form computing the same polynomial, and the latter can be computed efficiently.

Claim 2.4. *Let h be an n variate, degree d polynomial computable by a full rank almost set-multilinear ABP of width $\mathbf{w} = (w_1, w_2, \ldots, w_{d-1})$ and length d. There is a randomized algorithm that takes input blackbox access to h and the width vector $\mathbf{w}$, and outputs a full rank almost set-multilinear ABP of width $\mathbf{w}$ in canonical form computing h, with probability at least $1 - \frac{1}{\text{poly}(n)}$. The running time of the algorithm is poly(n, β), where β is the bit length of the coefficients of h.*

We prove the claim in Section 5.3. The algorithm is similar to reconstruction of set-multilinear ABP in Reference [30], except that the latter needs to be adapted suitably as we are dealing with almost set-multilinear ABP.

3 LIE ALGEBRA OF IMM

Dropping the subscripts $\mathbf{w}$ and d, we refer to $\mathsf{IMM}_{\mathbf{w},d}$ as IMM. We show that the Lie algebra, $\mathfrak{g}_{\mathsf{IMM}}$ consists of well-structured subspaces, and by analysing these subspaces, we are able to identify all the irreducible invariant subspaces of $\mathfrak{g}_{\mathsf{IMM}}$.

3.1 Structure of the Lie Algebra $\mathfrak{g}_{\mathsf{IMM}}$

Recall that $\mathbf{x} = \mathbf{x}_1 \uplus \mathbf{x}_2 \uplus \cdots \uplus \mathbf{x}_d$ are the variables of IMM that are also referred to as $\{x_1, x_2, \ldots, x_n\}$[25] for notational convenience.

Lemma 3.1. *Let $\mathcal{W}_1, \mathcal{W}_2, \mathcal{W}_3$ be the following sets (spaces) of matrices:*

[25] Following the ordering mentioned in Section 2.3.

(1) *$\mathcal{W}_1$ consists of all matrices $D = (d_{ij})_{i,j\in[n]}$ such that D is diagonal and*

$$\sum_{i=1}^{n} d_{ii}x_i \cdot \frac{\partial \mathsf{IMM}}{\partial x_i} = 0.$$

(2) *$\mathcal{W}_2$ consists of all matrices $B = (b_{ij})_{i,j\in[n]}$ such that*

$$\sum_{i,j\in[n]} b_{ij}x_j \cdot \frac{\partial \mathsf{IMM}}{\partial x_i} = 0,$$

where in every summand $b_{ij} \neq 0$ only if $x_i \neq x_j$ and $x_i, x_j \in \mathbf{x}_l$ for some $l \in [d]$.

(3) *$\mathcal{W}_3$ consists of all matrices $C = (c_{ij})_{i,j\in[n]}$ such that*

$$\sum_{i,j\in[n]} c_{ij}x_j \cdot \frac{\partial \mathsf{IMM}}{\partial x_i} = 0,$$

where in every summand $c_{ij} \neq 0$ only if either $x_i \in \mathbf{x}_2$, $x_j \in \mathbf{x}_1$ or $x_i \in \mathbf{x}_{d-1}$, $x_j \in \mathbf{x}_d$.

Then $\mathfrak{g}_{\mathsf{IMM}} = \mathcal{W}_1 \oplus \mathcal{W}_2 \oplus \mathcal{W}_3$.

The proof of Lemma 3.1 is given in Section 7.3.

Elaboration on Lemma 3.1: An element $E = (e_{ij})_{i,j\in[n]}$ of $\mathfrak{g}_{\mathsf{IMM}}$ is an $n \times n$ matrix with rows and columns indexed by variables of IMM following the ordering mentioned in Section 2.3. Since $\sum_{i,j\in[n]} e_{ij}x_j \cdot \frac{\partial \mathsf{IMM}}{\partial x_i} = 0$, E appears as shown in Figure 2, where the row indices correspond to derivatives and column indices correspond to *shifts*.[26] The proof will show that E is a sum of three matrices $D \in \mathcal{W}_1$, $B \in \mathcal{W}_2$ and $C \in \mathcal{W}_3$ such that

(1) D contributes to the diagonal entries.
(2) B contributes to the block-diagonal entries of E corresponding to the locations:
   - $(x_i^{(1)}, x_j^{(1)})$ where $i, j \in [w_1]$ and $i \neq j$,
   - $(x_i^{(d)}, x_j^{(d)})$ where $i, j \in [w_{d-1}]$ and $i \neq j$,
   - $(x_{ij}^{(l)}, x_{pq}^{(l)})$ where $i, p \in [w_{l-1}]$ and $j, q \in [w_l]$ for $l \in [2, d-1]$, and $(i, j) \neq (p, q)$.

(3) C contributes to the two corner rectangular blocks corresponding to:
   - rows labelled by $\mathbf{x}_2$ variables and columns labelled by $\mathbf{x}_1$ variables,
   - rows labelled by $\mathbf{x}_{d-1}$ variables and columns labelled by $\mathbf{x}_d$ variables.

To get a finer understanding of $\mathfrak{g}_{\mathsf{IMM}}$ and its dimension, we look at the spaces $\mathcal{W}_1, \mathcal{W}_2$, and $\mathcal{W}_3$ closely, and henceforth call them the *diagonal space*, the *block-diagonal space* and the *corner space*, respectively.

Corner space $\mathcal{W}_3$:

Lemma 3.2 (Corner Space). *The space $\mathcal{W}_3 = \mathcal{W}_3^{(a)} \oplus \mathcal{W}_3^{(b)}$ where $\mathcal{W}_3^{(a)} = \mathcal{A}_1 \oplus \mathcal{A}_2 \oplus \cdots \oplus \mathcal{A}_{w_2}$ and $\mathcal{W}_3^{(b)} = \mathcal{A'}_1 \oplus \mathcal{A'}_2 \oplus \cdots \oplus \mathcal{A'}_{w_{d-2}}$ such that for every $i \in [w_2]$ $\mathcal{A}_i$ is isomorphic to the space of $w_1 \times w_1$ anti-symmetric matrices over $\mathbb{F}$, and for every $j \in [w_{d-2}]$ $\mathcal{A'}_j$ is isomorphic to the space of $w_{d-1} \times w_{d-1}$ anti-symmetric matrices over $\mathbb{F}$. Hence,* $\dim(\mathcal{W}_3) = \frac{1}{2}[w_1w_2(w_1 - 1) + w_{d-1}w_{d-2}(w_{d-1} - 1)]$.

The proof is in Section 7.3. We briefly elaborate on the statement here.

[26] Borrowing terminology from the *shifted partial derivatives* measure [27].

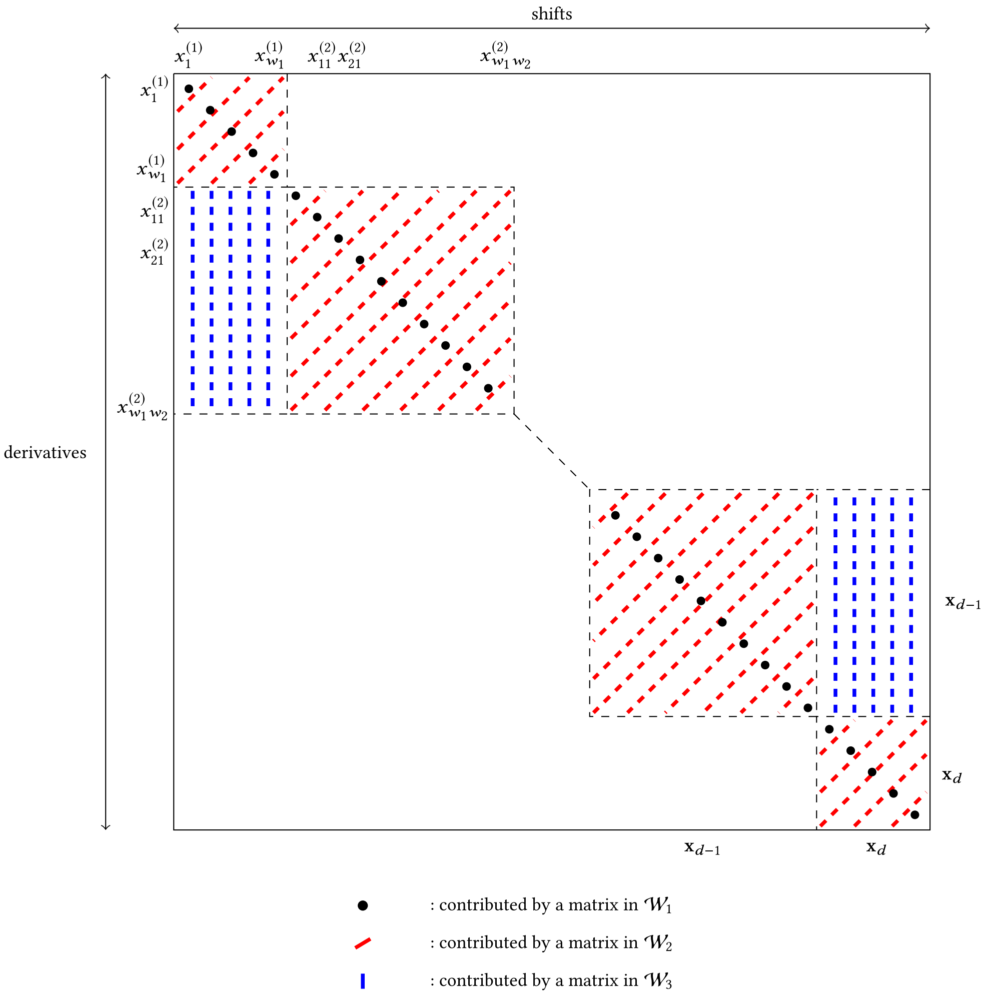
shifts
$x_1^{(1)}$
$x_{w_1}^{(1)}$
$x_{11}^{(2)}$
$x_{21}^{(2)}$
$x_{w_1 w_2}^{(2)}$
derivatives
$\mathbf{x}_{d-1}$
$\mathbf{x}_d$
● : contributed by a matrix in $\mathcal{W}_1$
: contributed by a matrix in $\mathcal{W}_2$
: contributed by a matrix in $\mathcal{W}_3$

Fig. 2. A matrix E in $\mathfrak{g}_{\text{IMM}}$.

Elaboration on Lemma 3.2: Every element $C \in \mathcal{W}_3$ can be expressed as a sum of two $n \times n$ matrices $C^{(a)} \in \mathcal{W}_3^{(a)}$ and $C^{(b)} \in \mathcal{W}_3^{(b)}$. $C^{(a)}$ looks as shown in Figure 3, where for every $i \in [w_2]$ $C_i^{(a)}$ is an anti-symmetric matrix. The structure of $C^{(b)}$ is similar[27] to that of $C^{(a)}$ with nonzero entries restricted to the rows indexed by $\mathbf{x}_{d-1}$ variables and columns indexed by $\mathbf{x}_d$ variables.

Block-diagonal space $\mathcal{W}_2$: In the following lemma, $\mathcal{Z}_{w_k}$ denotes the space of $w_k \times w_k$ matrices with diagonal entries zero for $k \in [d-1]$. Also, for notational convenience, we assume that $w_0 = w_d = 1$. We will also use the tensor product of matrices: if $A = (a_{i,j}) \in \mathbb{F}^{r \times s}$ and $B \in \mathbb{F}^{t \times u}$, then $A \otimes B$ is the $(rt) \times (su)$ matrix given by

[27] Once we rearrange the rows in $C^{(b)}$ indexed by variables in $\mathbf{x}_{d-1}$ according to row major ordering (instead of column major ordering) of variables in $\mathbf{x}_{d-1}$.

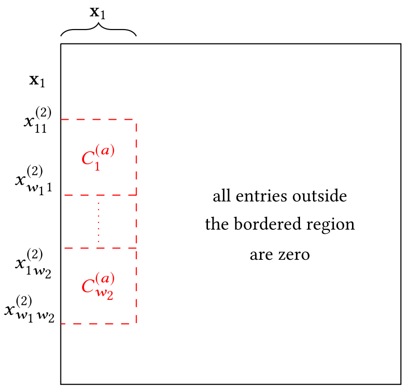

Fig. 3. A matrix $C^{(a)}$ in $\mathcal{W}_3^{(a)}$.

$$A \otimes B = \begin{bmatrix} a_{1,1}B & \cdots & a_{1,s}B \\ \vdots & \vdots & \vdots \\ a_{r,1}B & \cdots & a_{r,s}B \end{bmatrix}.$$

Lemma 3.3 (Block-Diagonal Space). *The space $\mathcal{W}_2 = \mathcal{B}_1 \oplus \mathcal{B}_2 \oplus \cdots \oplus \mathcal{B}_{d-1}$ such that for every $k \in [d-1]$, $\mathcal{B}_k$ is isomorphic to the $\mathbb{F}$-linear space spanned by $t_k \times t_k$ matrices of the form*

$$\begin{bmatrix} -Z^T \otimes I_{w_{k-1}} & 0 \\ 0 & I_{w_{k+1}} \otimes Z \end{bmatrix}_{t_k \times t_k} \quad \text{where } Z \in \mathcal{Z}_{w_k} \text{ and } t_k = w_k(w_{k-1} + w_{k+1}). \tag{1}$$

Hence, $\dim(\mathcal{W}_2) = \sum_{k=1}^{d-1}(w_k^2 - w_k)$.

The proof is in Section 7.3.

Elaboration on Lemma 3.3: An element $B \in \mathcal{W}_2$ is a sum of $d-1$, $n \times n$ matrices $B_1, B_2, \ldots, B_{d-1}$ such that for every $k \in [d-1]$, $B_k \in \mathcal{B}_k$ and the nonzero entries of B_k are restricted to the rows and columns indexed by $\mathbf{x}_k \uplus \mathbf{x}_{k+1}$ variables. The submatrix in B_k corresponding to these rows and columns looks as shown in Equation (1).

Diagonal space $\mathcal{W}_1$: In the next lemma, $\mathcal{Y}_{w_k}$ denotes the space of $w_k \times w_k$ diagonal matrices for $k \in [d-1]$. As before, we assume $w_0 = w_d = 1$.

Lemma 3.4 (Diagonal Space). *The space $\mathcal{W}_1$ contains the space $\mathcal{D}_1 \oplus \mathcal{D}_2 \oplus \cdots \oplus \mathcal{D}_{d-1}$ such that for every $k \in [d-1]$, $\mathcal{D}_k$ is isomorphic to the $\mathbb{F}$-linear space spanned by $t_k \times t_k$ matrices of the form*

$$\begin{bmatrix} -Y \otimes I_{w_{k-1}} & 0 \\ 0 & I_{w_{k+1}} \otimes Y \end{bmatrix}_{t_k \times t_k}, \quad \text{where } Y \in \mathcal{Y}_{w_k} \text{ and } t_k = w_k(w_{k-1} + w_{k+1}). \tag{2}$$

Hence, $\dim(\mathcal{W}_1) \geq \sum_{k=1}^{d-1} w_k$.

The proof (still given in Section 7.3) is similar to that of Lemma 3.3.

Elaboration on Lemma 3.4: An element $D \in \mathcal{D}_1 \oplus \mathcal{D}_2 \oplus \cdots \oplus \mathcal{D}_{d-1}$ is a sum of $d-1$, $n \times n$ matrices $D_1, D_2, \ldots, D_{d-1}$ such that for every $k \in [d-1]$, $D_k \in \mathcal{D}_k$ and the nonzero entries of D_k are restricted to the rows and columns indexed by $\mathbf{x}_k \uplus \mathbf{x}_{k+1}$ variables. The submatrix in D_k corresponding to these rows and columns looks as shown in Equation (2).

3.2 Random Elements of $\mathfrak{g}_{\mathsf{IMM}}$

The algorithm in Theorem 1b involves picking a random matrix R' in $\mathfrak{g}_f$ and computing its characteristic polynomial $h(x)$. To ensure the correctness of the algorithm, $h(x)$ will have to be square free over $\mathbb{F}$. In Lemma 3.5, we show that the characteristic polynomial of a random matrix R in $\mathfrak{g}_{\mathsf{IMM}}$ is square free with high probability. From Claim 2.1 this implies that if f is equivalent to IMM then the characteristic polynomial of R' is also square free with high probability.

Claim 3.1. *There is a diagonal matrix $D \in \mathfrak{g}_{\mathsf{IMM}}$ with all entries distinct.*

Proof. From Lemma 3.4, we know that for $k \in [d-1]$ the submatrix of $D_k \in \mathcal{D}_k$ defined by the rows and columns indexed by the variables in $\mathbf{x}_k \uplus \mathbf{x}_{k+1}$ is

$$\begin{bmatrix} -Y_k \otimes I_{w_{k-1}} & 0 \\ 0 & I_{w_{k+1}} \otimes Y_k \end{bmatrix},$$

where $Y_k \in \mathcal{Y}_k$. Let the (i,i)th entry of Y_k be $y_i^{(k)}$ and pretend that these entries are distinct formal variables, say $\mathbf{y}$ variables. Consider the matrix $D = \sum_{i=1}^{d-1} D_i$ and observe the following:

a. For $k \in [2, d-1]$, the $(x_{ij}^{(k)}, x_{ij}^{(k)})$th entry of D is $y_i^{(k-1)} - y_j^{(k)}$ where $i \in [w_{k-1}]$ and $j \in [w_k]$.
b. The $(x_i^{(1)}, x_i^{(1)})$th and $(x_j^{(d)}, x_j^{(d)})$th entry of D are $-y_i^{(1)}$ and $y_j^{(d-1)}$, respectively, where $i \in [w_1]$ and $j \in [w_{d-1}]$.

In particular, all the diagonal entries of D are distinct linear forms in the $\mathbf{y}$ variables. Hence, if we assign values to the $\mathbf{y}$ variables uniformly at random from a set $S \subseteq \mathbb{F}$ such that $|S| \geq n^2$ then with non zero probability D has all diagonal entries distinct after the random assignment. □

Lemma 3.5. *If $\{L_1, L_2, \ldots, L_m\}$ is a basis of the Lie algebra $\mathfrak{g}_{\mathsf{IMM}}$ then the characteristic polynomial of an element $L = \sum_{i=1}^{m} r_i L_i$, where $r_i \in_R \mathbb{F}$ is picked independently and uniformly at random from $[2n^3]$, is square free with probability at least $1 - \frac{1}{\mathrm{poly}(n)}$.*

Proof. Pretend that the r_i's are formal variables. The characteristic polynomial $h_{\mathbf{r}}(x)$ of L is a polynomial in x with coefficients that are polynomial of degree at most n in $\mathbf{r} = \{r_1, r_2, \ldots, r_m\}$ variables.

Observation 3.1. *The discriminant of $h_{\mathbf{r}}(x)$, $\mathrm{disc}(h_{\mathbf{r}}(x)) := \mathrm{res}_x(h_{\mathbf{r}}, \frac{\partial h_{\mathbf{r}}}{\partial x})$, is a nonzero polynomial in $\mathbf{r}$ variables of degree at most*[28] *$2n^2$, where $\mathrm{res}_x(h_{\mathbf{r}}, \frac{\partial h_{\mathbf{r}}}{\partial x})$ is the resultant of $h_{\mathbf{r}}$ and $\frac{\partial h_{\mathbf{r}}}{\partial x}$ when treated as univariates in x.*

Observation 3.1 is proved at the end of the section. Since $\mathrm{disc}(h_{\mathbf{r}}(x))$ is not an identically zero polynomial in the $\mathbf{r}$ variables and has degree less than $2n^2$, if we set every $\mathbf{r}$ variable uniformly and independently at random to a value in $[2n^3]$ then using Schwartz-Zippel lemma with probability at least $1 - \frac{1}{\mathrm{poly}(n)}$, $\gcd(h_{\mathbf{r}}, \frac{\partial h_{\mathbf{r}}}{\partial x}) = 1$. This implies with probability at least $1 - \frac{1}{\mathrm{poly}(n)}$, $h_{\mathbf{r}}(x)$ is square free.

Proof of Observation 3.1. $h_{\mathbf{r}}$ is a monic polynomial in x of degree n and $\frac{\partial h_{\mathbf{r}}}{\partial x}$ is a polynomial in x of degree $(n-1)$. Also the coefficient of x^{n-1} in $\frac{\partial h_{\mathbf{r}}}{\partial x}$ is $\mathbf{r}$ variable free. The Sylvester matrix of $h_{\mathbf{r}}$ and $\frac{\partial h_{\mathbf{r}}}{\partial x}$ with respect to variable x is a $(2n-1) \times (2n-1)$ matrix. Thus, $\mathrm{res}_x(h_{\mathbf{r}}, \frac{\partial h_{\mathbf{r}}}{\partial x})$ is a polynomial in the $\mathbf{r}$-variables of degree less than $2n^2$. If $\mathrm{res}_x(h_{\mathbf{r}}, \frac{\partial h_{\mathbf{r}}}{\partial x})$ is identically zero as a polynomial in $\mathbf{r}$ then for every setting of $\mathbf{r}$ to field elements $\gcd(h_{\mathbf{r}}, \frac{\partial h_{\mathbf{r}}}{\partial x}) \neq 1$ implying $h_{\mathbf{r}}$ is not square free. This would contradict Claim 3.1, as we can set the $\mathbf{r}$ variables appropriately such that L is a diagonal matrix with distinct diagonal entries, and $h_{\mathbf{r}}$ for such a setting of the $\mathbf{r}$ variables is square free. □

[28] A careful analysis could show that the degree is in fact $n(n-1)$, but we do not need such a precision here.

3.3 Invariant Subspaces of $\mathfrak{g}_{\mathsf{IMM}}$

The ordering of the variables in IMM allows us to identify them naturally with the unit vectors $e_1, e_2, \ldots, e_n$ in $\mathbb{F}^n$—the vector e_i corresponds to the ith variable in the ordering. We will write e_x to refer to the unit vector corresponding to the variable x. Let $\mathcal{U}_{1,2}$ represent the coordinate subspace spanned by the unit vectors corresponding to the variables in $\mathbf{x}_1 \uplus \mathbf{x}_2$. Similarly, $\mathcal{U}_k$ represents the coordinate subspace spanned by the unit vectors corresponding to the variables in $\mathbf{x}_k$ for $k \in [2, d-1]$, and $\mathcal{U}_{d-1,d}$ represents the coordinate subspace spanned by the unit vectors corresponding to the variables in $\mathbf{x}_{d-1} \uplus \mathbf{x}_d$. In Lemma 3.6, we establish that $\mathcal{U}_{1,2}, \mathcal{U}_2, \ldots, \mathcal{U}_{d-1}, \mathcal{U}_{d-1,d}$ are the only irreducible invariant subspaces of $\mathfrak{g}_{\mathsf{IMM}}$.

Claim 3.2. *Let $\mathcal{U}$ be a nonzero invariant subspace of $\mathfrak{g}_{\mathsf{IMM}}$. If $\mathbf{u} = (u_1, u_2, \ldots, u_n)^T \in \mathcal{U}$ and $u_j \neq 0$ then $e_j \in \mathcal{U}$, implying $\mathcal{U}$ is a coordinate subspace.*

Proof. Claim 3.1 states that there is a diagonal matrix $D \in \mathfrak{g}_{\mathsf{IMM}}$ with distinct diagonal entries $\lambda_1, \lambda_2, \ldots, \lambda_n$. Since $\mathcal{U}$ is invariant for D, if $\mathbf{u} = (u_1, u_2, \ldots, u_n)^T \in \mathcal{U}$ then $(\lambda_1^i u_1, \lambda_2^i u_2, \ldots, \lambda_n^i u_n) \in \mathcal{U}$ for every $i \in \mathbb{N}$. Let $S_{\mathbf{u}} := \{j \in [n] | u_j \neq 0\}$ be the support of $\mathbf{u} \neq 0$. As $\lambda_1, \lambda_2, \ldots, \lambda_n$ are distinct, the vectors $(\lambda_1^i u_1, \lambda_2^i u_2, \ldots, \lambda_n^i u_n)$ are $\mathbb{F}$-linearly independent for $0 \leq i < |S_{\mathbf{u}}|$. Hence, the unit vector $e_j \in \mathcal{U}$ for every $j \in S_{\mathbf{u}}$. It follows that $\mathcal{U}$ is the coordinate subspace spanned by those e_j for which $j \in S_{\mathbf{u}}$ for some $\mathbf{u} \in \mathcal{U}$. □

Lemma 3.6. *The only irreducible invariant subspaces of $\mathfrak{g}_{\mathsf{IMM}}$ are $\mathcal{U}_{1,2}, \mathcal{U}_2, \ldots, \mathcal{U}_{d-1}, \mathcal{U}_{d-1,d}$.*

Proof. It follows from Lemma 3.1 and Figure 2 that $\mathcal{U}_{1,2}, \mathcal{U}_2, \ldots, \mathcal{U}_{d-1}, \mathcal{U}_{d-1,d}$ are invariant subspaces. We show in the next two claims that the spaces $\mathcal{U}_{1,2}, \mathcal{U}_2, \ldots, \mathcal{U}_{d-1}, \mathcal{U}_{d-1,d}$ are irreducible. The proofs are given in Section 7.3. □

Claim 3.3. *No invariant subspace of $\mathfrak{g}_{\mathsf{IMM}}$ is properly contained in $\mathcal{U}_k$ for $k \in [2, d-1]$.*

Claim 3.4. *The invariant subspaces $\mathcal{U}_{1,2}$ and $\mathcal{U}_{d-1,d}$ are irreducible, and the only invariant subspace properly contained in $\mathcal{U}_{1,2}$ (respectively, $\mathcal{U}_{d-1,d}$) is $\mathcal{U}_2$ (respectively, $\mathcal{U}_{d-1}$).*

We in fact show in the proof of Claim 3.3 that the closure of e_x under the action of $\mathfrak{g}_{\mathsf{IMM}}$ is $\mathcal{U}_k$ for any $x \in \mathbf{x}_k$, where $k \in [2, d-1]$. Similarly, in the proof of Claim 3.4, we show that the closure of e_x under the action of $\mathfrak{g}_{\mathsf{IMM}}$ is $\mathcal{U}_{1,2}$ (respectively, $\mathcal{U}_{d-1,d}$) for any $x \in \mathbf{x}_1$ (respectively, $x \in \mathbf{x}_d$). This observation helps infer that the spaces $\mathcal{U}_{1,2}, \mathcal{U}_2, \ldots, \mathcal{U}_{d-1}, \mathcal{U}_{d-1,d}$ are the only irreducible invariant subspaces of $\mathfrak{g}_{\mathsf{IMM}}$: Suppose $\mathcal{V}$ is an irreducible invariant subspace. If $e_x \in \mathcal{V}$ for some $x \in \mathbf{x}_k$ where $k \in [2, d-1]$, then $\mathcal{U}_k \subseteq \mathcal{V}$ as $\mathcal{U}_k$ is the closure of e_x. If $e_x \in \mathcal{V}$ for some $x \in \mathbf{x}_1$ (respectively, $x \in \mathbf{x}_d$), then $\mathcal{U}_{1,2} \subseteq \mathcal{V}$ (respectively, $\mathcal{U}_{d-1,d} \subseteq \mathcal{V}$) as $\mathcal{U}_{1,2}$ (respectively, $\mathcal{U}_{d-1,d}$) is the closure of e_x. Therefore, $\mathcal{V}$ is a direct sum of some of the irreducible invariant subspaces $\mathcal{U}_{1,2}, \mathcal{U}_2, \ldots, \mathcal{U}_{d-1}, \mathcal{U}_{d-1,d}$. Since $\mathcal{V}$ is irreducible, it is equal to one of these irreducible invariant subspaces.

Corollary 3.7 (Uniqueness of Decomposition). *The decomposition,*

$$\mathbb{F}^n = \mathcal{U}_{1,2} \oplus \mathcal{U}_3 \oplus \cdots \oplus \mathcal{U}_{d-2} \oplus \mathcal{U}_{d-1,d},$$

is unique in the following sense; if $\mathbb{F}^n = \mathcal{V}_1 \oplus \mathcal{V}_2 \oplus \cdots \oplus \mathcal{V}_s$, where $\mathcal{V}_i'$s are irreducible invariant subspaces of $\mathfrak{g}_{\mathsf{IMM}}$, then $s = d-2$ and for every $i \in [s]$, $\mathcal{V}_i$ is equal to $\mathcal{U}_{1,2}$ or $\mathcal{U}_{d-1,d}$, or some $\mathcal{U}_k$ for $k \in [3, d-2]$.

Proof. Since $\mathcal{V}_i$'s are irreducible invariant subspaces, from Lemma 3.6 it follows that for every $i \in [s]$ $\mathcal{V}_i$ equals one among $\mathcal{U}_{1,2}, \mathcal{U}_2, \ldots, \mathcal{U}_{d-1}, \mathcal{U}_{d-1,d}$. Since $\mathcal{V}_1, \mathcal{V}_2, \ldots, \mathcal{V}_s$ span the entire $\mathbb{F}^n$, the only possible decomposition is $\mathbb{F}^n = \mathcal{U}_{1,2} \oplus \mathcal{U}_3 \oplus \cdots \oplus \mathcal{U}_{d-2} \oplus \mathcal{U}_{d-1,d}$. □

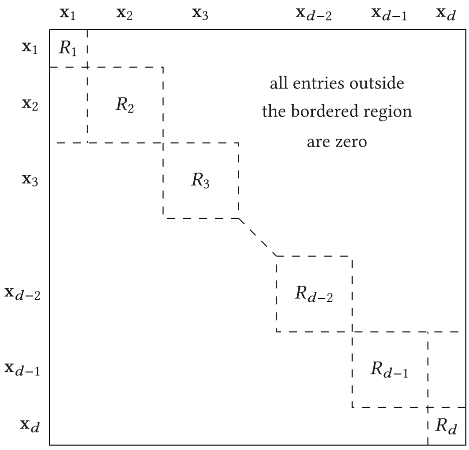

Fig. 4. Random element R in $\mathfrak{g}_{\mathsf{IMM}}$.

4 LIE ALGEBRA OF f EQUIVALENT TO IMM

Let f be an n variate polynomial such that $f = \mathsf{IMM}_{\mathbf{w},d}(A\mathbf{x})$, where $\mathbf{w} = (w_1, w_2, \ldots, w_{d-1}) \in \mathbb{N}^{d-1}$ and $A \in \mathsf{GL}(n)$. It follows, $n = w_1 + \sum_{i=2}^{d-1} w_{i-1}w_i + w_{d-1}$. From Section 2.2 and Lemma 3.6, we know $A^{-1}\mathcal{U}_{1,2}, A^{-1}\mathcal{U}_2, \ldots, A^{-1}\mathcal{U}_{d-1}, A^{-1}\mathcal{U}_{d-1,d}$ are the only irreducible invariant subspaces of $\mathfrak{g}_f$, and $A^{-1}\mathcal{U}_2$ (respectively, $A^{-1}\mathcal{U}_{d-1}$) is the only invariant subspace properly contained in $A^{-1}\mathcal{U}_{1,2}$ (respectively, $A^{-1}\mathcal{U}_{d-1,d}$). Also from Corollary 3.7 it follows that $\mathbb{F}^n = A^{-1}\mathcal{U}_{1,2} \oplus A^{-1}\mathcal{U}_3 \oplus \cdots \oplus A^{-1}\mathcal{U}_{d-2} \oplus A^{-1}\mathcal{U}_{d-1,d}$. In this section, we give an efficient randomized algorithm to compute a basis of each of the spaces $A^{-1}\mathcal{U}_{1,2}, A^{-1}\mathcal{U}_2, \ldots, A^{-1}\mathcal{U}_{d-1}, A^{-1}\mathcal{U}_{d-1,d}$ given only blackbox access to f (but no knowledge of $\mathbf{w}$ or A).

4.1 Computing Invariant Subspaces of the Lie Algebra $\mathfrak{g}_f$

First, we efficiently compute a basis $\{L_1', L_2', \ldots, L_m'\}$ of $\mathfrak{g}_f$ using the algorithm stated in Lemma 2.14. By Claim 2.1, $L_1 = AL_1'A^{-1}, L_2 = AL_2'A^{-1}, \ldots, L_m = AL_m'A^{-1}$ form a basis of $\mathfrak{g}_{\mathsf{IMM}}$. Suppose $R' = \sum_{i=1}^m r_iL_i'$ is a random element of $\mathfrak{g}_f$, chosen by picking the r_i's independently and uniformly at random from $[2n^3]$. Then $R = AR'A^{-1} = \sum_{i=1}^m r_iL_i$ is a random element of $\mathfrak{g}_{\mathsf{IMM}}$ and it follows from Lemma 3.5 that the characteristic polynomial of R is square free with probability at least $1 - \frac{1}{\mathrm{poly}(n)}$. So assume henceforth that the characteristic polynomial of R (and hence also of R') is square free.

Moreover, from Figure 2 it follows that R has the structure as shown in Figure 4. Let $h(x) = \prod_{i=1}^d h_i(x)$ be the characteristic polynomial of R and R', where $h_i(x)$ is the characteristic polynomial of R_i, and $g_1(x), g_2(x), \ldots, g_s(x)$ be the distinct irreducible factors of $h(x)$ over $\mathbb{F}$. Suppose $\mathcal{N}_i'$ is the null space of $g_i(R')$. Thus, $\mathcal{N}_i$, the null space of $g_i(R)$ (equal to $A \cdot g_i(R') \cdot A^{-1}$), is $A\mathcal{N}_i'$ for $i \in [s]$. We study the null spaces $\mathcal{N}_1, \mathcal{N}_2, \ldots, \mathcal{N}_s$ in the next two claims and show how to extract out the irreducible invariant subspaces of $\mathfrak{g}_f$ from $\mathcal{N}_1', \mathcal{N}_2', \ldots, \mathcal{N}_s'$ (as specified in Algorithm 3). The proofs of these claims (using simple linear algebra) can be found in Section 7.4.

Claim 4.1. *For all $i \in [s]$, let $\mathcal{N}_i$ and $\mathcal{N}_i'$ be the null spaces of $g_i(R)$ and $g_i(R')$. Then*

(1) $\mathbb{F}^n = \mathcal{N}_1 \oplus \mathcal{N}_2 \oplus \cdots \oplus \mathcal{N}_s = \mathcal{N}_1' \oplus \mathcal{N}_2' \oplus \cdots \oplus \mathcal{N}_s'$.
(2) For all $i \in [s]$, $dim(\mathcal{N}_i) = dim(\mathcal{N}_i') = deg_x(g_i)$.

Claim 4.2. *Suppose $g_i(x)$ is an irreducible factor of the characteristic polynomial $h_k(x)$ of R_k (depicted in Figure 4) for some $k \in [d]$. Then the following holds:*

(1) If $k \in [2, d-1]$ then $\mathcal{N}_i \subseteq \mathcal{U}_k$ (equivalently $\mathcal{N}'_i \subseteq A^{-1}\mathcal{U}_k$).

(2) If $k = 1$ then $\mathcal{N}_i \subseteq \mathcal{U}_{1,2}$ (equivalently $\mathcal{N}'_i \subseteq A^{-1}\mathcal{U}_{1,2}$), and if $k = d$ then $\mathcal{N}_i \subseteq \mathcal{U}_{d-1,d}$ (equivalently $\mathcal{N}'_i \subseteq A^{-1}\mathcal{U}_{d-1,d}$).

Claim 4.3.

(1) If $g_{l_1}(x), g_{l_2}(x), \ldots, g_{l_r}(x)$ are all the irreducible factors of $h_k(x)$ for $k \in [2, d-1]$ then $A^{-1}\mathcal{U}_k = \mathcal{N}'_{l_1} \oplus \mathcal{N}'_{l_2} \oplus \cdots \oplus \mathcal{N}'_{l_r}$.

(2) If $g_{l_1}(x), g_{l_2}(x), \ldots, g_{l_r}(x)$ are all the irreducible factors of $h_1(x)h_2(x)$ (respectively, $h_{d-1}(x)h_d(x)$) then $A^{-1}\mathcal{U}_{1,2} = \mathcal{N}'_{l_1} \oplus \mathcal{N}'_{l_2} \oplus \cdots \oplus \mathcal{N}'_{l_r}$ (respectively, $A^{-1}\mathcal{U}_{d-1,d} = \mathcal{N}'_{l_1} \oplus \mathcal{N}'_{l_2} \oplus \cdots \oplus \mathcal{N}'_{l_r}$).

Proof. If $k \in [2, d-1]$ then $\mathcal{N}'_{l_1} + \mathcal{N}'_{l_2} + \cdots + \mathcal{N}'_{l_r}$ is a direct sum and

$$\dim(A^{-1}\mathcal{U}_k) = \deg_x(h_k) = \sum_{j=1}^{r} \deg_x(g_{l_j}) = \sum_{j=1}^{r} \dim(\mathcal{N}'_{l_j}), \text{ which follow from Claim 4.1.}$$

Hence, from Claim 4.2, $A^{-1}\mathcal{U}_k = \mathcal{N}'_{l_1} \oplus \mathcal{N}'_{l_2} \oplus \cdots \oplus \mathcal{N}'_{l_r}$. The proof for the second part is similar. □

Lemma 4.1. *Given as input bases of the null spaces $\mathcal{N}'_1, \mathcal{N}'_2, \ldots, \mathcal{N}'_s$, we can compute bases of the spaces $A^{-1}\mathcal{U}_{1,2}, A^{-1}\mathcal{U}_2, \ldots, A^{-1}\mathcal{U}_{d-1}, A^{-1}\mathcal{U}_{d-1,d}$ in deterministic polynomial time.*

Proof. Recall $\mathcal{N}'_i$ is the null space of $g_i(R')$, where $g_i(x)$ is an irreducible factor of $h_k(x)$ for some $k \in [d]$.

Case A: $k \in [2, d-1]$; from Claim 4.2 it follows that $\mathcal{N}'_i \subseteq A^{-1}\mathcal{U}_k$. Pick a basis vector $\mathbf{v}$ in $\mathcal{N}'_i$ and compute the closure of $\mathbf{v}$ under the action of $\mathfrak{g}_f$ using Algorithm 4 given in Section 4.2. Since the closure of $\mathbf{v}$ is the smallest invariant subspace of $\mathfrak{g}_f$ containing $\mathbf{v}$, by Claim 3.3 the closure of $\mathbf{v}$ equals $A^{-1}\mathcal{U}_k$.

Case B: $k = 1$ or $k = d$; the arguments for $k = 1$ and $k = d$ are similar. We prove it for $k = 1$. From Claim 4.2, we have $\mathcal{N}'_i \subseteq A^{-1}\mathcal{U}_{1,2}$. Pick a basis vector $\mathbf{v}$ of $\mathcal{N}'_i$ and compute its closure under the action of $\mathfrak{g}_f$ using Algorithm 4. Similar to case A, this gives us an invariant subspace of $\mathfrak{g}_f$ contained in $A^{-1}\mathcal{U}_{1,2}$ and by Claim 3.4 this invariant subspace is either $A^{-1}\mathcal{U}_2$ or $A^{-1}\mathcal{U}_{1,2}$. However, $\mathcal{N}'_i \cap A^{-1}\mathcal{U}_2$ (by Claim 4.3) is empty, as $g_i(x)$ is an irreducible factor of $h_1(x)$ (not $h_2(x)$). Hence, $\mathbf{v} \notin A^{-1}\mathcal{U}_2$ and the closure of $\mathbf{v}$ must be $A^{-1}\mathcal{U}_{1,2}$. □

To summarize, first we pick a random element R' in $\mathfrak{g}_f$, find its characteristic polynomial $h(x)$ and factorize $h(x)$ to get the irreducible factors $g_1(x), g_2(x), \ldots, g_s(x)$. Then, we compute the null spaces $\mathcal{N}'_1, \mathcal{N}'_2, \ldots, \mathcal{N}'_s$ of $g_1(R'), g_2(R'), \ldots, g_s(R')$, respectively. By applying Lemma 4.1, we find the invariant subspaces of $\mathfrak{g}_f$, $A^{-1}\mathcal{U}_{1,2}$, $A^{-1}\mathcal{U}_2$, …, $A^{-1}\mathcal{U}_{d-1}$, $A^{-1}\mathcal{U}_{d-1,d}$ from these null spaces. We present this formally in Algorithm 3.

Comments on Algorithm 3:

a. Observe that in step 6 of the algorithm, we need $\mathbb{F}$ to be $\mathbb{Q}$ (as assumed) or a finite field, because univariate factorization can be done effectively over such fields [7, 9, 32].

b. When Algorithm 3 is invoked in Algorithm 2 for an n variate degree d polynomial f, there may not exists a $\mathbf{w} \in \mathbb{N}^{d-1}$ and an $A \in \mathsf{GL}(n)$ such that $f = \mathsf{IMM}_{\mathbf{w},d}(A\mathbf{x})$. We point out a few additional checks that need to be added to the above algorithm to handle this case. In step 9, if the pruned list (after removing repetitions) has size other than d then output “Fail.” Also from Claim 3.4, exactly two subspaces in the pruned list $\{\mathcal{V}_1, \mathcal{V}_2, \ldots, \mathcal{V}_d\}$, say, $\mathcal{V}_2$ and $\mathcal{V}_{d-1}$, should be subspaces of other vector spaces, say $\mathcal{V}_1$ and $\mathcal{V}_d$, respectively. We can find these two spaces by doing a pairwise check among the d vector spaces. If such subspaces

ALGORITHM 3: Computing irreducible invariant subspaces of $\mathfrak{g}_f$

```
INPUT: A basis $\{L'_1, L'_2, \ldots, L'_m\}$ of $\mathfrak{g}_f$.
OUTPUT: Bases of the irreducible invariant subspaces of $\mathfrak{g}_f$.
Pick a random element $R' = \sum_{j=1}^{m} r_j L'_j$ in $\mathfrak{g}_f$, where $r_j \in_R [2n^3]$.
Compute the characteristic polynomial $h(x)$ of $R'$.
if $h(x)$ is not square free then
    Output "Fail" and stop.
end if
Factor $h(x) = g_1(x) \cdot g_2(x) \ldots g_s(x)$ into irreducible factors over $\mathbb{F}$.
Find bases of the null spaces $\mathcal{N}'_1, \mathcal{N}'_2, \ldots, \mathcal{N}'_s$ of $g_1(R'), g_2(R'), \ldots, g_s(R')$, respectively.
For every $\mathcal{N}'_i$, pick a vector $\mathbf{v}$ in the basis of $\mathcal{N}'_i$ and compute the closure of $\mathbf{v}$ with respect to $\mathfrak{g}_f$ using
    Algorithm 4 given in Section 4.2.
Let $\{\mathcal{V}_1, \mathcal{V}_2, \ldots, \mathcal{V}_s\}$ be the list of the closure spaces; check for all $i \neq j$ and $i, j \in [s]$, whether $\mathcal{V}_i = \mathcal{V}_j$
    to remove repetitions from the above list and get the pruned list $\{\mathcal{V}_1, \mathcal{V}_2, \ldots, \mathcal{V}_d\}$.[29]
Output the set $\{\mathcal{V}_1, \mathcal{V}_2, \ldots, \mathcal{V}_d\}$.
```

do not exist among $\mathcal{V}_1, \mathcal{V}_2, \ldots, \mathcal{V}_d$ then output "Fail." Further, if $\mathbb{F}^n \neq \mathcal{V}_1 \oplus \mathcal{V}_3 \oplus \cdots \oplus \mathcal{V}_{d-2} \oplus \mathcal{V}_d$ (assuming $\mathcal{V}_2 \subseteq \mathcal{V}_1$ and $\mathcal{V}_{d-1} \subseteq \mathcal{V}_d$) then output "Fail."

c. It follows from the above discussion, if $f = \mathsf{IMM}_{\mathbf{w},d}(A\mathbf{x})$, then we can assume $\mathcal{V}_3, \mathcal{V}_4, \ldots, \mathcal{V}_{d-2}$ are the spaces $A^{-1}\mathcal{U}_3, A^{-1}\mathcal{U}_4, \ldots, A^{-1}\mathcal{U}_{d-2}$ in some *unknown* order. The spaces $\mathcal{V}_1, \mathcal{V}_2$ and $\mathcal{V}_d, \mathcal{V}_{d-1}$ are either the spaces $A^{-1}\mathcal{U}_{1,2}, A^{-1}\mathcal{U}_2$ and $A^{-1}\mathcal{U}_{d-1,d}, A^{-1}\mathcal{U}_{d-1}$, respectively, or the spaces $A^{-1}\mathcal{U}_{d-1,d}$, $A^{-1}\mathcal{U}_{d-1}$ and $A^{-1}\mathcal{U}_{1,2}, A^{-1}\mathcal{U}_2$, respectively.

4.2 Closure of a Vector Under the Action of $\mathfrak{g}_f$

Algorithm 4 computes the closure of $\mathbf{v} \in \mathbb{F}^n$ under the action of a space $\mathcal{L}$ spanned by $n \times n$ matrices. Let $\{M_1, M_2, \ldots, M_m\}$ be a basis of $\mathcal{L}$ where $M_i \in \mathbb{F}^{n \times n}$. For a set of vectors $T = \{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_q\} \subseteq \mathbb{F}^n$, let $\mathcal{L} \cdot T$ denote the set $\{M_a \mathbf{v}_b | a \in [m] \text{ and } b \in [q]\}$.

ALGORITHM 4: Computing the closure of $\mathbf{v}$ under the action of $\mathcal{L}$

```
INPUT: $\mathbf{v} \in \mathbb{F}^n$ and a basis $\{M_1, M_2, \ldots, M_m\}$ of $\mathcal{L}$.
OUTPUT: Basis of the closure of $\mathbf{v}$ under the action of $\mathcal{L}$.
Let $\mathcal{V}^{(0)} = \{\mathbf{v}\}$ and $\mathcal{V}^{(1)} = \text{span}_{\mathbb{F}}\{\mathbf{v}, M_1\mathbf{v}, \ldots, M_m\mathbf{v}\}$.
Set $i = 1$.
Compute a basis of $\mathcal{V}^{(1)}$ and let $T_1 = \{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_{q_1}\}$ be this basis.
while $\mathcal{V}^{(i-1)} \neq \mathcal{V}^{(i)}$ do
    Set $i = i + 1$.
    Compute a basis for $\mathcal{V}^{(i)} = \text{span}_{\mathbb{F}}\{T_{i-1} \cup \mathcal{L} \cdot T_{i-1}\}$ and let $T_i = \{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_{q_i}\}$ be this basis.
end while
Output $T_i$.
```

Claim 4.4. *Algorithm 4 computes the closure of* $\mathbf{v} \in \mathbb{F}^n$ *under the action of* $\mathcal{L}$ *in time polynomial in n and the bit length of the entries of* $\mathbf{v}$ *and* $M_1, M_2, \ldots, M_m$.

Proof. The closure of $\mathbf{v}$ under the action of $\mathcal{L}$ is the $\mathbb{F}$-linear span of all vectors of the form $\mu.\mathbf{v}$, where μ is a non-commutative monomial in $M_1, M_2, \ldots, M_m$ (including unity). Algorithm 4

[29] Reusing symbols.

computes exactly this set and hence the closure of $\mathbf{v}$. Moreover, $\dim(\mathcal{V}^{(i)}) \leq n$ and in every iteration of the while loop $\dim(\mathcal{V}^{(i)}) > \dim(\mathcal{V}^{(i-1)})$, until $\mathcal{V}^{(i)} = \mathcal{V}^{(i-1)}$. Hence, Algorithm 4 runs in time polynomial in n and the bit length of the entries of $\mathbf{v}$ and $M_1, M_2, \ldots, M_m$. □

5 RECONSTRUCTION OF FULL RANK ABP FOR *F*

Let f be a polynomial equivalent to $\mathsf{IMM}_{\mathbf{w},d}$ for some (unknown) $\mathbf{w} \in \mathbb{N}^{d-1}$. In this section, we show that the invariant subspaces of $\mathfrak{g}_f$ let us compute a $\mathbf{w} \in \mathbb{N}^{d-1}$ and an $A \in \mathsf{GL}(n)$ such that $f = \mathsf{IMM}_{\mathbf{w},d}(A\mathbf{x})$. Since f is equivalent to $\mathsf{IMM}_{\mathbf{w},d}$, it is computable by a full rank ABP $X_1 \cdot X_2 \ldots X_{d-1} \cdot X_d$ of width $\mathbf{w}$ and length d with linear form entries in the matrices. We call this full rank ABP A, which, as explained below, is not the only full rank ABP computing f.

Many full rank ABPs for *f*: The full rank ABP $X_1' \cdot X_2' \cdots X_d'$ resulting from *each* of the following three transformations on A still computes f,

(1) *Transposition*: Set $X_k' = X_{d+1-k}^T$ for $k \in [d]$.
(2) *Left-right multiplications*: Let $A_1, \ldots, A_{d-1}$ be matrices such that $A_k \in \mathsf{GL}(w_k)$ for every $k \in [d-1]$. Set $X_1' = X_1 \cdot A_1$, $X_d' = A_{d-1}^{-1} \cdot X_d$, and $X_k' = A_{k-1}^{-1} \cdot X_k \cdot A_k$ for $k \in [2, d-1]$.
(3) *Corner translations*: Suppose $\{C_{11}, C_{12}, \ldots, C_{1w_2}\}$ and $\{C_{d1}, C_{d2}, \ldots, C_{dw_{d-2}}\}$ are two sets containing anti-symmetric matrices in $\mathbb{F}^{w_1 \times w_1}$ and $\mathbb{F}^{w_{d-1} \times w_{d-1}}$, respectively. Let $Y_2 \in \mathbb{F}[\mathbf{x}]^{w_1 \times w_2}$ (respectively, $Y_{d-1} \in \mathbb{F}[\mathbf{x}]^{w_{d-2} \times w_{d-1}}$) be a matrix with its ith column (respectively, ith row) equal to $C_{1i} \cdot X_1^T$ (respectively, $X_d^T \cdot C_{di}$). Set $X_2' = X_2 + Y_2$, $X_{d-1}' = X_{d-1} + Y_{d-1}$, and $X_k' = X_k$ for $k \in [d] \setminus \{2, d-1\}$.

In each of the above three cases $f = X_1' \cdot X_2' \cdots X_d'$; this is easy to verify for cases 1 and 2, in case 3 observe that $X_1 \cdot C_{1i} \cdot X_1^T = X_d^T \cdot C_{di} \cdot X_d = 0$. It turns out that the full rank ABPs obtained by (repeatedly) applying the above three transformations on A are the only full rank ABPs computing f. This would follow from the discussion in Section 6. Although there are multiple full rank ABPs for f, the layer spaces of these ABPs are unique (Lemma 5.1). This uniqueness of the layer spaces essentially facilitates the recovery of a full rank ABP for f. Let us denote the span of the linear forms[30] in X_1 and X_2 (respectively, X_{d-1} and X_d) by $\mathcal{X}_{1,2}$ (respectively, $\mathcal{X}_{d-1,d}$).

Lemma 5.1 (Uniqueness of the Layer Spaces of Full Rank ABP for f). *Suppose $X_1 \cdot X_2 \cdots X_d$ and $X_1' \cdot X_2' \cdots X_d'$ are two full rank ABPs of widths $\mathbf{w} = (w_1, w_2, \ldots, w_{d-1})$ and $\mathbf{w}' = (w_1', w_2', \ldots, w_{d-1}')$, respectively, computing the same polynomial f. Then one of the following two cases is true:*

a. *$w_k' = w_k$ for $k \in [d-1]$, and the spaces $\mathcal{X}_1', \mathcal{X}_{1,2}', \mathcal{X}_3', \ldots, \mathcal{X}_{d-1,d}', \mathcal{X}_d'$ are the spaces $\mathcal{X}_1, \mathcal{X}_{1,2}, \mathcal{X}_3, \ldots, \mathcal{X}_{d-1,d}, \mathcal{X}_d$, respectively.*
b. *$w_k' = w_{d-k}$ for $k \in [d-1]$, and the spaces $\mathcal{X}_1', \mathcal{X}_{1,2}', \mathcal{X}_3', \ldots, \mathcal{X}_{d-1,d}', \mathcal{X}_d'$ are the spaces $\mathcal{X}_d, \mathcal{X}_{d-1,d}, \mathcal{X}_{d-2}, \ldots, \mathcal{X}_{1,2}, \mathcal{X}_1$, respectively.*

The lemma would help characterize the group of symmetries of IMM in Section 6; the proof would follow readily from Claim 5.1 in Section 5.2. With an eye on Section 6 and for better clarity in the reduction to almost set-multilinear ABP in Section 5.2, we take a slight detour and show next how to compute these "unique" layer spaces of A.

[30] Identify linear forms with vectors in $\mathbb{F}^n$ as mentioned in Definition 2.6.

5.1 Computing Layer Spaces from Invariant Subspaces of $\mathfrak{g}_f$

Algorithm 3 outputs bases of the irreducible invariant subspaces $\{\mathcal{V}_i | i \in [d]\}$ of $\mathfrak{g}_f$. Recall, we assumed without loss of generality that $\mathcal{V}_2$ and $\mathcal{V}_{d-1}$ are subspaces of $\mathcal{V}_1$ and $\mathcal{V}_d$, respectively. The spaces $\mathcal{V}_1, \mathcal{V}_2$ and $\mathcal{V}_d, \mathcal{V}_{d-1}$ are either the spaces $A^{-1}\mathcal{U}_{1,2}, A^{-1}\mathcal{U}_2$ and $A^{-1}\mathcal{U}_{d-1,d}, A^{-1}\mathcal{U}_{d-1}$, respectively, or the spaces $A^{-1}\mathcal{U}_{d-1,d}, A^{-1}\mathcal{U}_{d-1}$ and $A^{-1}\mathcal{U}_{1,2}, A^{-1}\mathcal{U}_2$, respectively. Every other $\mathcal{V}_k$ is equal to $A^{-1}\mathcal{U}_{\sigma(k)}$ for some permutation σ on $[3, d-2]$ (σ is not known at the end of Algorithm 3). Hence,

$$\mathbb{F}^n = \mathcal{V}_1 \oplus \mathcal{V}_3 \oplus \cdots \oplus \mathcal{V}_{d-2} \oplus \mathcal{V}_d. \tag{3}$$

Since $\mathcal{V}_2 \subseteq \mathcal{V}_1$, we can start with a basis of $\mathcal{V}_2$ and fill in more elements from the basis of $\mathcal{V}_1$ to get a new basis of $\mathcal{V}_1$. Thus, we can assume the basis of $\mathcal{V}_2$ is contained in the basis of $\mathcal{V}_1$. Likewise, the basis of $\mathcal{V}_{d-1}$ is contained in the basis of $\mathcal{V}_d$.

Order the basis vectors of $\mathcal{V}_1$ such that the basis vectors of $\mathcal{V}_2$ are at the end and order the basis vectors of $\mathcal{V}_d$ such that the basis vectors of $\mathcal{V}_{d-1}$ are at the beginning. For $k \in [3, d-2]$, the basis vectors of $\mathcal{V}_k$ are ordered in an arbitrary way. Let u_k denote the dimension of $\mathcal{V}_k$ for $k \in [d]$. We identify the space $\mathcal{V}_k$ with an $n \times u_k$ matrix V_k, where the ith column in V_k is the ith basis vector of $\mathcal{V}_k$ in the above specified order. Algorithm 5 computes the layer spaces of A using V_1 to V_d. Let $t_2 = u_1$ and $t_k = u_k + t_{k-1}$ for $k \in [3, d-2]$.

ALGORITHM 5: Computing the layer spaces of A

```
INPUT: Bases of the irreducible invariant subspaces of $\mathfrak{g}_f$.
OUTPUT: Bases of the layer spaces of $\mathsf{A}$.
1: Form an $n \times n$ matrix $V$ by concatenating the columns of the matrices $V_1, V_3, \ldots, V_{d-2}, V_d$ in order, that
   is $V = [V_1|V_3|\ldots|V_{d-2}|V_d]$.
2: Compute $V^{-1}$. Number the rows of $V^{-1}$ by 1 to $n$.
3: Let $\mathcal{Y}_1$ be the space spanned by the first $u_1 - u_2$ rows of $V^{-1}$, and $\mathcal{Y}_{1,2}$ be the space spanned by the first
   $u_1$ rows of $V^{-1}$. Let $\mathcal{Y}_{d-1,d}$ be the space spanned by the last $u_d$ rows of $V^{-1}$ and $\mathcal{Y}_d$ be the space spanned
   by the last $u_d - u_{d-1}$ rows of $V^{-1}$. Finally, for every $k \in [3, d-2]$, let $\mathcal{Y}_k$ be the space spanned by the
   rows of $V^{-1}$ that are numbered by $t_{k-1} + 1$ to $t_{k-1} + u_k$. Output the bases of the spaces
   $\mathcal{Y}_1, \mathcal{Y}_{1,2}, \mathcal{Y}_3, \ldots, \mathcal{Y}_{d-2}, \mathcal{Y}_{d-1,d}, \mathcal{Y}_d$ in order.
```

Comments on Algorithm 5: Algorithm 2 invokes Algorithm 5 only after Algorithm 3, which returns "Fail" if $\mathbb{F}^n \neq \mathcal{V}_1 \oplus \mathcal{V}_3 \oplus \cdots \oplus \mathcal{V}_{d-2} \oplus \mathcal{V}_d$ (see comments after Algorithm 3). This ensures Equation (3) is satisfied and so V^{-1} exists in step 2 of the above algorithm, even if there are no $\mathbf{w} \in \mathbb{N}^{d-1}$ and $A \in \mathsf{GL}(n)$ such that $f = \mathsf{IMM}_{\mathbf{w},d}(A\mathbf{x})$.

Lemma 5.2. *If $f = X_1 \cdot X_2 \cdots X_d$ and $\mathcal{Y}_1, \mathcal{Y}_{1,2}, \mathcal{Y}_3, \ldots, \mathcal{Y}_{d-2}, \mathcal{Y}_{d-1,d}, \mathcal{Y}_d$ is the output of Algorithm 5 then there is a permutation σ on $[3, d-2]$ such that the following hold:*

(1) For every $k \in [3, d-2]$, $\mathcal{Y}_k = \mathcal{X}_{\sigma(k)}$.

(2) Either $\mathcal{Y}_1, \mathcal{Y}_{1,2}$ and $\mathcal{Y}_d, \mathcal{Y}_{d-1,d}$ are $\mathcal{X}_1, \mathcal{X}_{1,2}$ and $\mathcal{X}_d, \mathcal{X}_{d-1,d}$, respectively, or $\mathcal{Y}_1, \mathcal{Y}_{1,2}$ and $\mathcal{Y}_d, \mathcal{Y}_{d-1,d}$ are $\mathcal{X}_d, \mathcal{X}_{d-1,d}$ and $\mathcal{X}_1, \mathcal{X}_{1,2}$, respectively.

The proof is given in Section 7.5.

5.2 Reduction to Almost Set-Multilinear ABP

The outline: Once the invariant spaces of $\mathfrak{g}_f$ are computed, the reduction proceeds like this: As observed in the proof of Lemma 5.2, the matrix V in Algorithm 5 equals $A^{-1}E$ where E looks as shown in Figure 14. If $f = \mathsf{IMM}_{\mathbf{w},d}(A\mathbf{x})$ then $f(V\mathbf{x}) = \mathsf{IMM}_{\mathbf{w},d}(E\mathbf{x})$. Owing to the structure of

E, $f(V\mathbf{x})$ is computed by a full rank almost set-multilinear ABP, except that the ordering of the groups of variables occurring in the different layers of the ABP is unknown as σ is unknown. The "correct" ordering along with a width vector can be retrieved by applying evaluation dimension, thereby completing the reduction. For a slightly neater presentation of the details (and with the intent of proving Lemma 5.1), we deviate from this strategy a little bit and make use of the layer spaces that have already been computed by Algorithm 5.

The details: Algorithm 5 computes the spaces $\mathcal{Y}_1, \mathcal{Y}_{1,2}, \mathcal{Y}_3, \ldots, \mathcal{Y}_{d-2}, \mathcal{Y}_{d-1,d}, \mathcal{Y}_d$, which (according to Lemma 5.2) are either the spaces $\mathcal{X}_1, \mathcal{X}_{1,2}, \mathcal{X}_{\sigma(3)}, \ldots, \mathcal{X}_{\sigma(d-2)}, \mathcal{X}_{d-1,d}, \mathcal{X}_d$, respectively, or the spaces $\mathcal{X}_d, \mathcal{X}_{d-1,d}, \mathcal{X}_{\sigma(3)}, \ldots, \mathcal{X}_{\sigma(d-2)}, \mathcal{X}_{1,2}, \mathcal{X}_1$, respectively, for some unknown permutation σ on $[3, d-2]$. The claim below (proved in Section 7.5) shows how to correctly reorder these layer spaces.

Claim 5.1. *There is a randomized polynomial time algorithm that takes input the bases of the layer spaces* $\mathcal{Y}_1, \mathcal{Y}_{1,2}, \mathcal{Y}_3, \ldots, \mathcal{Y}_{d-2}, \mathcal{Y}_{d-1,d}, \mathcal{Y}_d$ *and with probability at least* $1 - \frac{1}{\text{poly}(n)}$ *reorders these layer spaces and outputs a width vector* $\mathbf{w}'$ *such that the reordered sequence of spaces and* $\mathbf{w}'$ *are:*

(1) *either* $\mathcal{X}_1, \mathcal{X}_{1,2}, \mathcal{X}_3, \ldots, \mathcal{X}_{d-2}, \mathcal{X}_{d-1,d}, \mathcal{X}_d$ *and* $(w_1, w_2, \ldots, w_{d-1})$, *respectively,*
(2) *or* $\mathcal{X}_d, \mathcal{X}_{d-1,d}, \mathcal{X}_{d-2}, \ldots, \mathcal{X}_3, \mathcal{X}_{1,2}, \mathcal{X}_1$ *and* $(w_d, w_{d-1}, \ldots, w_1)$, *respectively.*

Note: Until the algorithm in the claim is applied to reorder the spaces, Algorithm 2 is totally oblivious of the width vector $\mathbf{w}$ (it has been used only in the analysis thus far). So, due to the legitimacy of the transposition transformation mentioned at the beginning of this section, we may as well assume that the $\mathbf{w}'$ in the above claim is in fact our $\mathbf{w}$, and the output ordered sequence of spaces is $\mathcal{X}_1, \mathcal{X}_{1,2}, \mathcal{X}_3, \ldots, \mathcal{X}_{d-2}, \mathcal{X}_{d-1,d}, \mathcal{X}_d$.

Claim 5.2. *Given bases of the spaces* $\mathcal{X}_1, \mathcal{X}_{1,2}, \mathcal{X}_3, \ldots, \mathcal{X}_{d-2}, \mathcal{X}_{d-1,d}, \mathcal{X}_d$ *and* $\mathbf{w}$, *we can find an* $\widehat{A} \in \mathsf{GL}(n)$ *in polynomial time such that* $f(\widehat{A}\mathbf{x})$ *is computable by a full rank almost set-multilinear ABP of width* $\mathbf{w}$.

Proof. Identify the variables $x_1, \ldots, x_n$ with the variables $\mathbf{x}_1 \uplus \ldots \uplus \mathbf{x}_d$ of $\mathsf{IMM}_{\mathbf{w},d}$ following the ordering prescribed in Section 2.3. The map $\mathbf{x} \mapsto \widehat{A}\mathbf{x}$ should satisfy the following conditions:

(a) For every $k \in [3, d-2]$, the linear forms corresponding[31] to the basis vectors of $\mathcal{X}_k$ map to distinct variables in $\mathbf{x}_k$.
(b) The linear forms corresponding to the basis vectors in $\mathcal{X}_1$ (similarly, $\mathcal{X}_d$) map to distinct variables in $\mathbf{x}_1$ (similarly, $\mathbf{x}_d$).
(c) The linear forms corresponding to the basis vectors in $\mathcal{X}_{1,2}$ (similarly, $\mathcal{X}_{d-1,d}$) map to distinct variables in $\mathbf{x}_1 \uplus \mathbf{x}_2$ (similarly, $\mathbf{x}_{d-1} \uplus \mathbf{x}_d$).

Conditions (b) and (c) can be simultaneously satisfied as the basis of $\mathcal{X}_1$ (similarly, $\mathcal{X}_d$) is contained in the basis of $\mathcal{X}_{1,2}$ (similarly, $\mathcal{X}_{d-1,d}$) by construction. Such an $\widehat{A}$ can be easily obtained. □

We summarize the discussion in Algorithm 6.

Comments on Algorithm 6: The proof of Claim 5.1 includes Observation 7.5, which helps Algorithm 6 in step 1 to reorder the layer spaces. If f is not equivalent to $\mathsf{IMM}_{\mathbf{w},d}$ for some $\mathbf{w}$ then Algorithm 6 may fail in step 1, as at some stage it may not be able to find a variable set $\mathbf{z}_k$ such that $\text{Evaldim}_{\mathbf{y}_j \uplus \mathbf{z}_k}(h) < |\mathbf{z}_k|$ (see proof of Observation 7.5). When Algorithm 2 invokes Algorithm 6, if step 1 fails then the latter outputs "Fail" and stops.

[31] Recall, linear forms in $\mathbf{x}$ variables and vectors in $\mathbb{F}^n$ are naturally identified with each other.

ALGORITHM 6: Reduction to full rank almost set-multilinear ABP

```
INPUT: Bases of the layer spaces $\mathcal{Y}_1, \mathcal{Y}_{1,2}, \mathcal{Y}_3, \ldots, \mathcal{Y}_{d-2}, \mathcal{Y}_{d-1,d}, \mathcal{Y}_d$ from Algorithm 5.
OUTPUT: A $\mathbf{w} \in \mathbb{N}^{d-1}$ and an $\widehat{A} \in \mathrm{GL}(n)$ such that $f(\widehat{A}\mathbf{x})$ is computable by a full rank almost
set-multilinear ABP of width $\mathbf{w}$.
1: Reorder the layer spaces to $\mathcal{X}_1, \mathcal{X}_{1,2}, \mathcal{X}_3, \ldots, \mathcal{X}_{d-2}, \mathcal{X}_{d-1,d}, \mathcal{X}_d$ and obtain $\mathbf{w}$ (using Claim 5.1). /* This
   step succeeds with high probability if $f$ is equivalent to $\mathsf{IMM}_{\mathbf{w},d}$ for some $\mathbf{w}$. */
2: Find $\widehat{A} \in \mathrm{GL}(n)$ from the reordered spaces and $\mathbf{w}$ (using Claim 5.2).
```

5.3 Reconstructing Almost Set-Multilinear ABP

We prove Claim 2.4 in this section. Let $h = f(\widehat{A}\mathbf{x})$; identify $\mathbf{x}$ with the variables $\mathbf{x}_1 \uplus \ldots \uplus \mathbf{x}_d$ of $\mathsf{IMM}_{\mathbf{w},d}$ as before. From Claim 5.2, h is computable by a full rank almost set-multilinear ABP of width $\mathbf{w}$. Algorithm 2 uses Algorithm 7 to reconstruct a full rank almost set-multilinear ABP for h and then it replaces $\mathbf{x}$ by $\widehat{A}^{-1}\mathbf{x}$ to output a full rank ABP for f. The correctness of Algorithm 7 is presented as part of the proof of Claim 2.4. We begin with the following two observations the proofs of which appear in Section 7.5.

Observation 5.1. *If h is computable by a full rank almost set-multilinear ABP of width $\mathbf{w}$ then there is a full rank almost set-multilinear ABP of width $\mathbf{w}$ in canonical form computing h.*

Observation 5.2. *Let $X_1 \cdot X_2 \cdots X_d$ be a full rank almost set-multilinear ABP, and $C_k = X_k \cdots X_d$ for $k \in [2, d]$. Let the lth entry of C_k be h_{kl} for $l \in [w_{k-1}]$. Then the polynomials $\{h_{k1}, h_{k2}, \ldots, h_{kw_{k-1}}\}$ are $\mathbb{F}$-linearly independent.*

Notations for Algorithm 7: For $k \in [d-1]$, let $t_k = |\mathbf{x}_1 \uplus \mathbf{x}_2 \uplus \cdots \uplus \mathbf{x}_k|$ and $m_k = |\mathbf{x}_{k+1} \uplus \mathbf{x}_{k+2} \uplus \cdots \uplus \mathbf{x}_d|$. The (i, j)th entry of a matrix X is denoted by $X(i, j)$, and $e_{w_k,i}$ denotes a vector in $\mathbb{F}^{w_k}$ with the ith entry 1 and other entries 0. Let $\mathbf{y}_i$ denote the following partial assignment to the $\mathbf{x}_1$ variables: $x_i^{(1)}, \ldots, x_{w_1}^{(1)}$ are kept intact, while the remaining variables are set to zero. Similarly, $\mathbf{z}_j$ denotes the following partial assignment to the $\mathbf{x}_d$ variables: $x_j^{(d)}, \ldots, x_{w_{d-1}}^{(d)}$ are kept intact, while the remaining variables are set to zero. The notation $h(\mathbf{a}_i, \mathbf{x}_k, \mathbf{b}_j)$ means the variables $\mathbf{x}_1 \uplus \ldots \uplus \mathbf{x}_{k-1}$ are given the assignment $\mathbf{a}_i \in \mathbb{F}^{t_{k-1}}$ and the variables $\mathbf{x}_{k+1} \uplus \ldots \uplus \mathbf{x}_d$ are given the assignment $\mathbf{b}_j \in \mathbb{F}^{m_k}$. The connotations for $h(\mathbf{y}_i, \mathbf{x}_2, \mathbf{b}_j)$ and $h(\mathbf{a}_i, \mathbf{x}_{d-1}, \mathbf{z}_j)$ are similar. The function $\mathrm{poly}(n)$ is a suitably large polynomial function in n, say n^7.

Proof of Claim 2.4. By Observation 5.1, there is a full rank ABP $X_1' \cdot X_2' \cdots X_d'$ in canonical form computing h. Hence, $X_1 = X_1' = (x_1^{(1)}\ x_2^{(1)}\ \ldots\ x_{w_1}^{(1)})$ and $X_d = X_d' = (x_1^{(d)}\ x_2^{(d)}\ \ldots\ x_{w_{d-1}}^{(d)})$. We show next that with probability at least $1 - \frac{1}{\mathrm{poly}(n)}$, Algorithm 7 constructs $X_2, X_3, \ldots, X_{d-1}$ such that $X_2 = X_2' \cdot T_2, X_{d-1} = T_{d-2}^{-1} \cdot X_{d-1}'$ and $X_k = T_{k-1}^{-1} \cdot X_k' \cdot T_k$ for every $k \in [3, d-2]$, where $T_i \in \mathrm{GL}(w_i)$ for $i \in [2, d-2]$.

Steps 3–13: The matrix X_2 is formed in these steps. By Claim 5.2, the polynomials $h_{31}, \ldots, h_{3w_2}$ are $\mathbb{F}$-linearly independent. Since $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_{w_2}$ are randomly chosen in step 3, the matrix T_2 with (r, c)th entry $h_{3r}(\mathbf{b}_c)$ is in $\mathrm{GL}(w_2)$ with high probability. Let $X_2'T_2(i, j)$ be the (i, j)th entry of $X_2'T_2$. Observe that

$$h(\mathbf{y}_i, \mathbf{x}_2, \mathbf{b}_j) = X_2'T_2(i, j) \cdot x_i^{(1)} + \cdots + X_2'T_2(w_1, j) \cdot x_{w_1}^{(1)}.$$

As $h(\mathbf{y}_i, \mathbf{x}_2, \mathbf{b}_j)$ is a quadratic polynomial, we can compute it from blackbox access using the sparse polynomial interpolation algorithm in Reference [31]. By induction on the rows, $X_2(p, j) = X_2'T_2(p, j)$ for every $p \in [i+1, w_1]$ and $j \in [w_2]$. So in step 8, $g_j = X_2'T_2(i, j) \cdot x_i^{(1)}$, leading to $X_2(i, j) = X_2'T_2(i, j)$ in step 9.

ALGORITHM 7: Reconstruction of full rank almost set-multlinear ABP

```
INPUT: Blackbox access to an $n$ variate polynomial $h$ and the width vector $\mathbf{w}$.
OUTPUT: A full rank almost set-multilinear ABP of width $\mathbf{w}$ in canonical form computing $h$.
1:  Set $X_1 = (x_1^{(1)} x_2^{(1)} \dots x_{w_1}^{(1)})$ and $X_d = (x_1^{(d)} x_2^{(d)} \dots x_{w_{d-1}}^{(d)})^T$.
2:
3:  Choose $w_2$ random points $\{\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_{w_2}\}$ from $S^{m_2}$ such that $S \subset \mathbb{F}$ and $|S| = \mathrm{poly}(n)$.
4:  Set $i = w_1$.
5:  while $i \geq 1$ do
6:      for every $j \in [w_2]$ do
7:          Interpolate the quadratic $h(\mathbf{y}_i, \mathbf{x}_2, \mathbf{b}_j)$.
8:          Set $g_j = h(\mathbf{y}_i, \mathbf{x}_2, \mathbf{b}_j) - \sum_{p=i+1}^{w_1} X_2(p, j) \cdot x_p^{(1)}$.
9:          If $g_j$ is not divisible by $x_i^{(1)}$, output "Fail". Else, set $X_2(i, j) = g_j / x_i^{(1)}$.
10:     end for
11:     Set $i = i - 1$.
12: end while
13: If the linear forms in $X_2$ are not $\mathbb{F}$-linearly independent, output "Fail".
14:
15: Set $k = 3$.
16: while $k \leq d - 2$ do
17:     Find $w_{k-1}$ evaluations, $\{\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_{w_{k-1}}\} \subset \mathbb{F}^{t_{k-1}}$, of $\mathbf{x}_1 \uplus \mathbf{x}_2 \uplus \dots \uplus \mathbf{x}_{k-1}$ variables such that
        $X_1 \cdot X_2 \cdots X_{k-1}$ evaluated at $\mathbf{a}_i$ equals $e_{w_{k-1}, i}$.
18:     Choose $w_k$ random points $\{\mathbf{b}_1, \mathbf{b}_2, \dots, \mathbf{b}_{w_k}\}$ from $S^{m_k}$ such that $S \subset \mathbb{F}$ and $|S| = \mathrm{poly}(n)$.
19:     Interpolate the linear forms $h(\mathbf{a}_i, \mathbf{x}_k, \mathbf{b}_j)$ for $i \in [w_{k-1}], j \in [w_k]$.
20:     Set $X_k(i, j) = h(\mathbf{a}_i, \mathbf{x}_k, \mathbf{b}_j)$ for $i \in [w_{k-1}], j \in [w_k]$.
21:     If the linear forms in $X_k$ are not $\mathbb{F}$-linearly independent, output "Fail".
22:     Set $k = k + 1$.
23: end while
24:
25: Find $w_{d-2}$ evaluations, $\{\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_{w_{d-2}}\} \subset \mathbb{F}^{t_{d-2}}$, of $\mathbf{x}_1 \uplus \mathbf{x}_2 \uplus \dots \uplus \mathbf{x}_{d-2}$ variables such that
    $X_1 \cdot X_2 \cdots X_{d-2}$ evaluated at $\mathbf{a}_i$ equals $e_{w_{d-2}, i}$ .
26: Set $j = w_{d-1}$.
27: while $j \geq 1$ do
28:     for every $i \in [w_{d-2}]$ do
29:         Interpolate the quadratic $h(\mathbf{a}_i, \mathbf{x}_{d-1}, \mathbf{z}_j)$.
30:         Set $g_i = h(\mathbf{a}_i, \mathbf{x}_{d-1}, \mathbf{z}_j) - \sum_{q=j+1}^{w_{d-1}} X_{d-1}(i, q) \cdot x_q^{(d)}$.
31:         If $g_i$ is not divisible by $x_j^{(d)}$, output "Fail". Else, set $X_{d-1}(i, j) = g_i / x_j^{(d)}$.
32:     end for
33:     Set $j = j - 1$.
34: end while
35: If the linear forms in $X_{d-1}$ are not $\mathbb{F}$-linearly independent, output "Fail".
36:
37: Output $X_1 \cdot X_2 \cdots X_{d-1} \cdot X_d$ as the full rank almost set-multilinear ABP for $h$.
```

Steps 15–23: The matrices $X_3, \dots, X_{d-2}$ are formed in these steps. By the time the algorithm reaches step 17, it has already computed $X_2, \dots, X_{k-1}$ such that $X_2 = X_2' T_2$ and $X_q = T_{q-1}^{-1} X_q' T_q$ for $q \in [3, k-1]$, where $T_q \in \mathrm{GL}(w_q)$. So, $X_1' \dots X_{k-1}' = X_1 \dots X_{k-1} T_{k-1}^{-1}$. As the linear forms in $X_1, \dots, X_{k-1}$ are $\mathbb{F}$-linearly independent (otherwise the algorithm would have terminated in step 13 or 21), we can easily compute points $\{\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_{w_{k-1}}\}$ satisfying the required condition in step 17. By Claim 5.2, the polynomials $h_{(k+1)1}, \dots, h_{(k+1)w_k}$ are $\mathbb{F}$-linearly independent. Since

$\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_{w_k}$ are randomly chosen in step 18, the matrix T_k with (r, c)th entry $h_{(k+1)r}(\mathbf{b}_c)$ is in $\mathrm{GL}(w_k)$ with high probability. Now observe that $h(\mathbf{a}_i, \mathbf{x}_k, \mathbf{b}_j)$ is the (i, j)th entry of $T_{k-1}^{-1} X_k' T_k$, which implies $X_k = T_{k-1}^{-1} X_k' T_k$ from step 20.

Steps 25–35: In these steps, matrix X_{d-1} is formed. The argument showing $X_{d-1} = T_{d-2}^{-1} X_{d-1}'$ is similar to the argument used for steps 3–13, except that now we induct on columns instead of rows.

The output ABP $X_1 \ldots X_d$ is in canonical form as $X_1' \ldots X_d'$ is also in canonical form. It is clear that the total running time of the algorithm is $\mathrm{poly}(n, \beta)$, where β is the bit length of the coefficients of h, which influences the bit length of the values returned by the blackbox. □

6 SYMMETRIES OF IMM

Recall from Section 2.3, $\mathsf{IMM}_{\mathbf{w},d}$ (for brevity IMM) is the n variate polynomial computed by the full rank ABP $Q_1 \cdot Q_2 \cdots Q_d$, where the set of variables in Q_k is $\mathbf{x}_k$ for every $k \in [d]$. In this section, we determine the group of symmetries of IMM (denoted by $\mathcal{G}_{\mathsf{IMM}}$) and show that IMM is characterized by its symmetries. We make a note of a few notations and terminologies below.

Notations:

- Calligraphic letters $\mathcal{H}, \mathcal{C}, \mathcal{M}$ and $\mathcal{T}$ denote subgroups of $\mathcal{G}_{\mathsf{IMM}}$. Let $\mathcal{C}$ and $\mathcal{H}$ be subgroups of $\mathcal{G}_{\mathsf{IMM}}$ such that $\mathcal{C} \cap \mathcal{H} = I_n$ and for every $H \in \mathcal{H}$ and $C \in \mathcal{C}$, $H \cdot C \cdot H^{-1} \in \mathcal{C}$. Then $\mathcal{C} \rtimes \mathcal{H}$ denotes the *semidirect product* of $\mathcal{C}$ and $\mathcal{H}$.[32]
- For every $A \in \mathcal{G}_{\mathsf{IMM}}$ the full rank ABP obtained by replacing $\mathbf{x}$ by $A\mathbf{x}$ in $Q_1 \cdot Q_2 \cdots Q_d$ is termed as the full rank ABP *determined by A*. This full rank ABP also computes IMM.
- Let X be a matrix with entries as linear forms in $\mathbf{y} \uplus \mathbf{z}$ variables. We break X into two parts $X(\mathbf{y})$ and $X(\mathbf{z})$ such that $X = X(\mathbf{y}) + X(\mathbf{z})$. The (i, j)th linear form in $X(\mathbf{y})$ (respectively, $X(\mathbf{z})$) is the part of the (i, j)th linear form of X in $\mathbf{y}$ (respectively, $\mathbf{z}$) variables.

6.1 The Group $\mathcal{G}_{\mathsf{IMM}}$

Three subgroups of $\mathcal{G}_{\mathsf{IMM}}$: As before, let $\mathbf{w} = (w_1, w_2, \ldots, w_{d-1})$ and $w_k > 1$ for every $k \in [d-1]$. In Theorem 2 below, we show that $\mathcal{G}_{\mathsf{IMM}}$ is generated by three special subgroups.

(1) *Transposition subgroup $\mathcal{T}$*: If $w_k \neq w_{d-k}$ for any $k \in [d-1]$ then $\mathcal{T}$ is the trivial group containing only I_n. Otherwise, if $w_k = w_{d-k}$ for every $k \in [d-1]$ then $\mathcal{T}$ is the group consisting of two elements I_n and T. The matrix T is such that the full rank ABP determined by T is $Q_d^T \cdot Q_{d-1}^T \cdots Q_1^T$. Clearly, T is a permutation matrix and $T^2 = I_n$.

(2) *Left-right multiplications subgroup $\mathcal{M}$*: An $M \in \mathrm{GL}(n)$ is in $\mathcal{M}$ if and only if the full rank ABP $X_1 \cdot X_2 \cdots X_d$ determined by M has the following structure: There are matrices $A_1, \ldots, A_{d-1}$ with $A_k \in \mathrm{GL}(w_k)$ for every $k \in [d-1]$, such that $X_1 = Q_1 \cdot A_1$, $X_d = A_{d-1}^{-1} \cdot Q_d$, and $X_k = A_{k-1}^{-1} \cdot Q_k \cdot A_k$ for $k \in [2, d-1]$. It is easy to verify that $\mathcal{M}$ is a subgroup of $\mathcal{G}_{\mathsf{IMM}}$ and is isomorphic to the direct product $\mathrm{GL}(w_1) \times \ldots \times \mathrm{GL}(w_{d-1})$.

(3) *Corner translations subgroup $\mathcal{C}$*: A matrix $C \in \mathrm{GL}(n)$ is in $\mathcal{C}$ if and only if the full rank ABP $X_1 \cdot X_2 \cdots X_d$ determined by C has the following structure: There are two sets $\{C_{11}, C_{12}, \ldots, C_{1w_2}\}$ and $\{C_{d1}, C_{d2}, \ldots, C_{dw_{d-2}}\}$ containing anti-symmetric matrices in $\mathbb{F}^{w_1 \times w_1}$ and $\mathbb{F}^{w_{d-1} \times w_{d-1}}$, respectively, such that $X_2 = Q_2 + Y_2$ and $X_{d-1} = Q_{d-1} + Y_{d-1}$, where $Y_2 \in \mathbb{F}[\mathbf{x}_1]^{w_1 \times w_2}$ (respectively, $Y_{d-1} \in \mathbb{F}[\mathbf{x}_d]^{w_{d-2} \times w_{d-1}}$) is a matrix with its ith column (respectively ith row) equal to $C_{1i} \cdot Q_1^T$ (respectively, $Q_d^T \cdot C_{di}$). For every other

[32] $\mathcal{C} \rtimes \mathcal{H}$ is the set $\mathcal{C}\mathcal{H}$, which can be easily shown to be a subgroup of $\mathcal{G}_{\mathsf{IMM}}$, and it also follows that $\mathcal{C}$ is a normal subgroup of $\mathcal{C} \rtimes \mathcal{H}$.

$k \in [d] \setminus \{2, d-1\}$, $X_k = Q_k$. Observe that $Q_1 \cdot C_{1i} \cdot Q_1^T = Q_d^T \cdot C_{di} \cdot Q_d = 0$. It can also be verified that $\mathcal{C}$ is an abelian subgroup of $\mathcal{G}_{\mathsf{IMM}}$ and is isomorphic to the direct product $\mathcal{A}_{w_1}^{w_2} \times \mathcal{A}_{w_{d-1}}^{w_{d-2}}$, where $\mathcal{A}_w$ is the group of $w \times w$ anti-symmetric matrices under matrix addition and $\mathcal{A}_w^k$ is the k times direct product of this group.

Theorem 2 (Symmetries of IMM). *$\mathcal{G}_{\mathsf{IMM}} = \mathcal{C} \rtimes \mathcal{H}$, where $\mathcal{H} = \mathcal{M} \rtimes \mathcal{T}$.*

We prove Theorem 2 below. Following are a couple of remarks on it.

Remarks:

(a) *Characterization*: Let f be an n variate degree d polynomial satisfying the following: For any n variate degree d polynomial g, $\mathcal{G}_f = \mathcal{G}_g$ if and only if $f = \alpha \cdot g$ for some nonzero $\alpha \in \mathbb{F}$. Then f is said to be characterized by $\mathcal{G}_f$. We prove IMM is characterized by $\mathcal{G}_{\mathsf{IMM}}$ in Lemma 6.1. The groups $\mathcal{M}$ and $\mathcal{C}$ generate the "continuous symmetries" of IMM.

(b) *Comparison with a related work*: In [15] a different choice of the IMM polynomial is considered, namely the trace of a product of d square symbolic matrices—let us call this polynomial IMM′.[33] The group of symmetries of IMM′ is determined in Reference [15] and it is shown that IMM′ is characterized by $\mathcal{G}_{\mathsf{IMM}'}$. The group of symmetries of IMM′, like IMM, is generated by the transposition subgroup, the left-right multiplication subgroup, and (instead of the corner translations subgroup) the *circular transformations subgroup*—an element in this subgroup cyclically rotates the order of the matrices and hence does not change the trace of the product.

Proof of Theorem 2

We begin with the following observation, which is immediate from Lemma 5.1.

Observation 6.1. *If $X_1 \cdot X_2 \cdots X_d$ is a width $\mathbf{w}' = (w_1', w_2', \ldots, w_{d-1}')$ full rank ABP computing $\mathsf{IMM}_{\mathbf{w},d}$ then either*

(1) $w_k' = w_k$ for $k \in [d-1]$, and the spaces $\mathcal{X}_1, \mathcal{X}_{1,2}, \mathcal{X}_3, \ldots, \mathcal{X}_{d-1,d}, \mathcal{X}_d$ are the spaces $\mathcal{Q}_1, \mathcal{Q}_{1,2}, \mathcal{Q}_3, \ldots, \mathcal{Q}_{d-1,d}, \mathcal{Q}_d$, respectively, or

(2) $w_k' = w_{d-k}$ for $k \in [d-1]$, and the spaces $\mathcal{X}_1, \mathcal{X}_{1,2}, \mathcal{X}_3, \ldots, \mathcal{X}_{d-1,d}, \mathcal{X}_d$ are the spaces $\mathcal{Q}_d, \mathcal{Q}_{d-1,d}, \mathcal{Q}_{d-2}, \ldots, \mathcal{Q}_{1,2}, \mathcal{Q}_1$, respectively.

From the definitions of $\mathcal{T}$, $\mathcal{M}$, and $\mathcal{C}$ it follows that $\mathcal{C} \cap \mathcal{M} = \mathcal{C} \cap \mathcal{T} = \mathcal{M} \cap \mathcal{T} = I_n$. The claim below shows $\mathcal{G}_{\mathsf{IMM}}$ is generated by $\mathcal{C}$, $\mathcal{M}$, and $\mathcal{T}$.

Claim 6.1. *For every $A \in \mathcal{G}_{\mathsf{IMM}}$, there exist $C \in \mathcal{C}$, $M \in \mathcal{M}$ and $\tilde{T} \in \mathcal{T}$ such that $A = C \cdot M \cdot \tilde{T}$.*

Proof. Let $X_1 \cdot X_2 \cdots X_d$ be the full rank ABP A of width $\mathbf{w}$ determined by A. If $w_k = w_{d-k}$ for $k \in [d-1]$ then the spaces $\mathcal{X}_1, \mathcal{X}_{1,2}, \mathcal{X}_3, \ldots, \mathcal{X}_{d-1,d}, \mathcal{X}_d$ are either equal to $\mathcal{Q}_1, \mathcal{Q}_{1,2}, \mathcal{Q}_3, \ldots, \mathcal{Q}_{d-1,d}, \mathcal{Q}_d$, respectively, or $\mathcal{Q}_d, \mathcal{Q}_{d-1,d}, \mathcal{Q}_{d-2}, \ldots, \mathcal{Q}_{1,2}, \mathcal{Q}_1$, respectively (from Observation 6.1). Otherwise, if $w_k \neq w_{d-k}$ for any $k \in [d-1]$ then the spaces $\mathcal{X}_1, \mathcal{X}_{1,2}, \mathcal{X}_3, \ldots, \mathcal{X}_{d-1,d}, \mathcal{X}_d$ have only one choice and are equal to $\mathcal{Q}_1, \mathcal{Q}_{1,2}, \mathcal{Q}_3, \ldots, \mathcal{Q}_{d-1,d}, \mathcal{Q}_d$, respectively. We deal with these two choices of layer spaces separately.

Case A: Suppose $\mathcal{X}_1, \mathcal{X}_{1,2}, \mathcal{X}_3, \ldots, \mathcal{X}_{d-1,d}, \mathcal{X}_d$ are equal to $\mathcal{Q}_1, \mathcal{Q}_{1,2}, \mathcal{Q}_3, \ldots, \mathcal{Q}_{d-1,d}, \mathcal{Q}_d$, respectively. Hence, A looks as shown in Figure 5. The linear forms in X_2, X_{d-1} are in variables

[33] The complexities of IMM and IMM′ are polynomially related to each other, in particular, both are complete for algebraic branching programs under p-projections. But their groups of symmetries are slightly different.

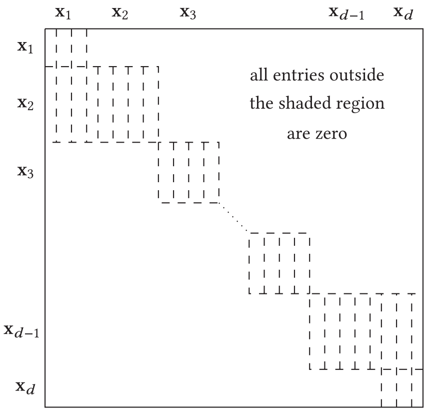

Fig. 5. Matrix A in $\mathcal{G}_{\mathsf{IMM}}$.

$\mathbf{x}_1 \uplus \mathbf{x}_2, \mathbf{x}_{d-1} \uplus \mathbf{x}_d$, respectively. Further,

$$\prod_{k=1}^{d} X_k = X_1 \cdot (X_2(\mathbf{x}_1) + X_2(\mathbf{x}_2)) \cdot \left(\prod_{k=3}^{d-2} X_k\right) \cdot (X_{d-1}(\mathbf{x}_{d-1}) + X_{d-1}(\mathbf{x}_d)) \cdot X_d = \mathsf{IMM}.^{34}$$

Since A is a full rank ABP and each monomial in IMM contains one variable from each set $\mathbf{x}_k$,

$$X_1 \cdot X_2(\mathbf{x}_2) \cdot \left(\prod_{k=3}^{d-2} X_k\right) \cdot X_{d-1}(\mathbf{x}_{d-1}) \cdot X_d = \mathsf{IMM}, \quad \text{and also}$$

$X_1 \cdot X_2(\mathbf{x}_1) \cdot \prod_{k=3}^{d-2} X_k \cdot X_{d-1}(\mathbf{x}_{d-1}) \cdot X_d = 0$ and $X_1 \cdot X_2(\mathbf{x}_2) \cdot \prod_{k=3}^{d-2} X_k \cdot X_{d-1}(\mathbf{x}_d) \cdot X_d = 0$, implying

$$X_1 \cdot X_2(\mathbf{x}_1) = 0_{w_2}^T \quad \text{and} \quad X_{d-1}(\mathbf{x}_d) \cdot X_d = 0_{w_{d-2}}, \tag{4}$$

where 0_w is a zero (column) vector in $\mathbb{F}^w$. Observation 6.2, the proof of which is in Section 7.6, proves the existence of a matrix $M \in \mathcal{M}$ such that the full rank ABP determined by M is $X_1 \cdot X_2(\mathbf{x}_2) \cdot X_3 \cdots X_{d-2} \cdot X_{d-1}(\mathbf{x}_{d-1}) \cdot X_d$. □

Observation 6.2. *There are matrices $A_1, \ldots, A_{d-1}$ with $A_k \in \mathsf{GL}(w_k)$ for every $k \in [d-1]$, such that $X_1 = Q_1 \cdot A_1$, $X_2(\mathbf{x}_2) = A_1^{-1} \cdot Q_2 \cdot A_2$, $X_{d-1}(\mathbf{x}_{d-1}) = A_{d-2}^{-1} \cdot Q_{d-1} \cdot A_{d-1}$, $X_d = A_{d-1}^{-1} \cdot Q_d$, and $X_k = A_{k-1}^{-1} \cdot Q_k \cdot A_k$ for $k \in [3, d-2]$.*

We now show the existence of a $C \in \mathcal{C}$ such that the full rank ABP determined by $C \cdot M$ is $X_1 \cdot X_2 \cdots X_d$, from which the claim follows by letting $\tilde{T} = I_n$. Since the linear forms in X_1 are $\mathbb{F}$-linearly independent, there are $w_1 \times w_1$ matrices $\{C_{11}, C_{12}, \ldots, C_{1w_2}\}$ such that the ith column of $X_2(\mathbf{x}_1)$ is $C_{1i} X_1^T$. So from Equation (4), $X_1 \cdot C_{1i} \cdot X_1^T = 0$ (equivalently $Q_1 \cdot C_{1i} \cdot Q_1^T = 0$) implying C_{1i} is an anti-symmetric matrix for every $i \in [w_2]$. Similarly, there are $w_{d-1} \times w_{d-1}$ anti-symmetric matrices $\{C_{d1}, C_{d2}, \ldots, C_{dw_{d-2}}\}$ such that the ith row of $X_{d-1}(\mathbf{x}_d)$ is $X_d^T C_{di}$. Let $C \in \mathsf{GL}(n)$ be such that the ABP determined by it is $Q_1 Q_2' Q_3 \cdots Q_{d-2} Q_{d-1}' Q_d$ where $Q_2' = Q_2 + Y_2$ and $Q_{d-1}' = Q_{d-1} + Y_{d-1}$, the ith column (respectively, ith row) of Y_2 (respectively, Y_{d-1}) is $C_{1i} Q_1^T$ (respectively, $Q_{d-1}^T C_{di}$). By construction, $C \in \mathcal{C}$ and the ABP determined by $C \cdot M$ is $X_1 \cdot X_2 \cdots X_d$.

Case B: Suppose $\mathcal{X}_1, \mathcal{X}_{1,2}, \mathcal{X}_3, \ldots, \mathcal{X}_{d-1,d}, \mathcal{X}_d$ are the spaces $\mathcal{Q}_d, \mathcal{Q}_{d-1,d}, \mathcal{Q}_{d-2}, \ldots, \mathcal{Q}_{1,2}, \mathcal{Q}_1$, respectively. This implies $w_k = w_{d-k}$ for $k \in [d-1]$, and hence the full rank ABP determined by T is

[34] We abuse notation slightly and write the 1×1 matrix $[\mathsf{IMM}]_{1\times 1}$ as IMM.

$Q_d^T \cdot Q_{d-1}^T \cdots Q_1^T$. From here the existence of $M \in \mathcal{M}$ and $C \in \mathcal{C}$ such that the full rank ABP determined by $M \cdot C \cdot T$ is $X_1 \cdot X_2 \cdots X_d$ follows just as in Case A. This completes the proof of the claim.

Observe that if $T \in \mathcal{T}$ then for every $M \in \mathcal{M}$, $T \cdot M \cdot T^{-1} \in \mathcal{M}$. Let $\mathcal{H} = \mathcal{M} \rtimes \mathcal{T}$. Clearly, $\mathcal{C} \cap \mathcal{H} = I_n$. The following claim along with Claim 6.1 then conclude the proof of Theorem 2.

Claim 6.2. *For every $C \in \mathcal{C}$ and $H \in \mathcal{H}$, $H \cdot C \cdot H^{-1} \in \mathcal{C}$.*

Proof. Let $H = M \cdot T$ where $M \in \mathcal{M}$ and $T \in \mathcal{T}$, and $A = MT \cdot C \cdot T^{-1}M^{-1}$. Suppose $X_1 \cdot X_2 \cdots X_{d-1} \cdot X_d$ is the ABP determined by A. The matrices T and T^{-1} in A together ensure that the spaces $\mathcal{X}_1, \mathcal{X}_{1,2}, \mathcal{X}_3, \ldots, \mathcal{X}_{d-1,d}, \mathcal{X}_d$ are equal to $\mathcal{Q}_1, \mathcal{Q}_{1,2}, \mathcal{Q}_3, \ldots, \mathcal{Q}_{d-1,d}, \mathcal{Q}_d$, respectively. Also the matrices M and M^{-1} together ensure that $X_i = Q_i$ for $i \in [d] \setminus \{2, d-1\}$, $X_2(\mathbf{x}_2) = Q_2$ and $X_{d-1}(\mathbf{x}_{d-1}) = Q_{d-1}$. Arguing as in Claim 6.1, we can infer that $A \in \mathcal{C}$. □

6.2 Characterization of IMM by $\mathcal{G}_{\mathsf{IMM}}$

For every $f = \alpha \cdot \mathsf{IMM}$, where $\alpha \in \mathbb{F}$ and $\alpha \neq 0$, it is easily observed that $\mathcal{G}_f = \mathcal{G}_{\mathsf{IMM}}$. We prove the converse in the following lemma for any homogeneous degree d polynomial in the $\mathbf{x}$ variables.

Lemma 6.1. *Let f be a homogeneous degree d polynomial in n variables $\mathbf{x} = \mathbf{x}_1 \uplus \ldots \uplus \mathbf{x}_d$. If $|\mathbb{F}| > d+1$ and the left-right multiplications subgroup $\mathcal{M}$ of $\mathcal{G}_{\mathsf{IMM}}$ is contained in $\mathcal{G}_f$ then $f = \alpha \cdot \mathsf{IMM}$ for some nonzero $\alpha \in \mathbb{F}$.*

Proof. First, we show that such an f is set-multilinear in the sets $\mathbf{x}_1, \ldots, \mathbf{x}_d$: Every monomial in f has exactly one variable from each of the sets $\mathbf{x}_1, \ldots, \mathbf{x}_d$. As $|\mathbb{F}| > d+1$, there is a nonzero $\rho \in \mathbb{F}$ that is not an eth root of unity for any $e \leq d$. Every element in $\mathcal{M}$ is defined by $d-1$ matrices $A_1, \ldots, A_{d-1}$ such that $A_k \in \mathsf{GL}(w_k)$ for every $k \in [d-1]$. For a $k \in [d-1]$, consider the element $M \in \mathcal{M}$ that is defined by $A_k = \rho \cdot I_{w_k}$ and $A_l = I_{w_l}$ for $l \in [d-1]$ and $l \neq k$. Then, $f(M \cdot \mathbf{x}) = f(\mathbf{x}_1, \ldots, \rho\mathbf{x}_k, \rho^{-1}\mathbf{x}_{k+1}, \ldots, \mathbf{x}_d)$, which by assumption is f. Comparing the coefficients of the monomials of $f(M \cdot \mathbf{x})$ and f, we observe that in every monomial of f the number of variables from $\mathbf{x}_k$ and $\mathbf{x}_{k+1}$ must be the same as ρ is not an eth root of unity for any $e \leq d$. Since k is chosen arbitrarily and f is homogeneous of degree d, f must be set-multilinear in the sets $\mathbf{x}_1, \ldots, \mathbf{x}_d$.

The proof is by induction on the degree of f. For $i \in [w_1]$, let $\mathbf{x}_{2i}$ be the set of variables in the ith row of Q_2, and Q_{2i} be the $1 \times w_2$ matrix containing the ith row of Q_2. Let IMM_i be the degree $d-1$ iterated matrix multiplication polynomial computed by the ABP $Q_{2i} \cdot Q_3 \cdots Q_d$. As f is set-multilinear, it can be expressed as

$$f = g_1 \cdot x_1^{(1)} + \cdots + g_{w_1} \cdot x_{w_1}^{(1)}, \tag{5}$$

where $g_1, \ldots, g_{w_1}$ are set-multilinear polynomials in the sets $\mathbf{x}_2, \ldots, \mathbf{x}_d$. Moreover, we can argue that g_i is set-multilinear in $\mathbf{x}_{2i}, \mathbf{x}_3, \ldots, \mathbf{x}_d$ as follows: Consider an $N \in \mathcal{M}$ that is defined by a diagonal matrix $A_1 \in \mathsf{GL}(w_1)$ whose (i,i)th entry is ρ and all other diagonal entries are 1; every other $A_l = I_{w_l}$ for $l \in [2, d-1]$. The transformation N scales the variable $x_i^{(1)}$ by ρ and the variables in $\mathbf{x}_{2i}$ by ρ^{-1}. By comparing the coefficients of the monomials of $f(N \cdot \mathbf{x})$ and f, we can conclude that g_i is set-multilinear in $\mathbf{x}_{2i}, \mathbf{x}_3, \ldots, \mathbf{x}_d$ for every $i \in [w_1]$.

Let $\mathcal{M}'$ be the subgroup of $\mathcal{M}$ containing those $M \in \mathcal{M}$ for which $A_1 = I_{w_1}$. From Equation (5), we can infer that $g_i(M \cdot \mathbf{x}) = g_i$ for $M \in \mathcal{M}'$, and hence the left-right multiplications subgroup of $\mathcal{G}_{\mathsf{IMM}_i}$ is contained in the group of symmetries of g_i. As degree of g_i is $d-1$, by induction[35]

[35] The base case $d = 1$ is trivial to show.

$g_i = \alpha_i \cdot \mathsf{IMM}_i$ for some nonzero $\alpha_i \in \mathbb{F}$ and

$$f = \alpha_1 \cdot \mathsf{IMM}_1 \cdot x_1^{(1)} + \cdots + \alpha_{w_1} \cdot \mathsf{IMM}_{w_1} \cdot x_{w_1}^{(1)}. \tag{6}$$

Next, we show that $\alpha_1 = \ldots = \alpha_{w_1}$, thereby completing the proof.

For an $i \in [2, w_1]$, let $A_1 \in \mathrm{GL}(w_1)$ be the upper triangular matrix whose diagonal entries are 1, the $(1, i)$th entry is 1 and remaining entries are zero. Let U be the matrix in $\mathcal{M}$ defined by A_1 and $A_l = I_{w_l}$ for $l \in [2, d-1]$. The transformation U has the following effect on the variables:

$$\begin{aligned} x_i^{(1)} &\mapsto x_1^{(1)} + x_i^{(1)} \text{ and} \\ x_{1j}^{(2)} &\mapsto x_{1j}^{(2)} - x_{ij}^{(2)} \text{ for every } j \in [w_2], \end{aligned}$$

every other $\mathbf{x}$ variable maps to itself. Applying U to f in Equation (6), we get

$$\begin{aligned} f &= f(U \cdot \mathbf{x}) \\ &= \alpha_1 \cdot (\mathsf{IMM}_1 - \mathsf{IMM}_i) \cdot x_1^{(1)} + \cdots + \alpha_i \cdot \mathsf{IMM}_i \cdot (x_1^{(1)} + x_i^{(1)}) + \cdots + \alpha_{w_1} \cdot \mathsf{IMM}_{w_1} \cdot x_{w_1}^{(1)} \\ &= f + (\alpha_i - \alpha_1) \cdot \mathsf{IMM}_i \cdot x_1^{(1)}, \\ &\Rightarrow \alpha_i - \alpha_1 = 0. \end{aligned}$$

Since this is true for any $i \in [2, w_1]$, we have $\alpha_1 = \ldots = \alpha_{w_1}$. □

7 PROOF OF TECHNICAL LEMMAS, CLAIMS, AND OBSERVATION

7.1 Incompleteness of Full Rank ABP

Observation 7.1. *For every sufficiently large $m \in \mathbb{N}$ there is an m variate multilinear polynomial that is not computable by full rank ABP.*

Proof. A full rank ABP computing an m variate polynomial f has both its width and length bounded by m, so f can also be computed by an ABP (not full rank) of width and length exactly m. Hence, it is sufficient to show that there is an m variate multilinear polynomial that is not computable by the latter kind of ABP. The number of edges in an ABP of width m and length m is $n = m^2(m-2) + 2m$. Let these n edges be $e_1, e_2, \ldots, e_n$ and suppose the edge e_i is labelled by the affine form $l_i = \sum_{j=1}^{m} c_{ij} x_j + c_{i0}$. Treat c_{ij}'s as formal variables. Then each of the $\binom{2n}{n}$ coefficients of the polynomial f computed by such an ABP is a polynomial in these $n(m+1)$ formal variables. Since $n(m+1) < 2^m$ for sufficiently large m, the coefficients of f restricted to just the multilinear monomials $\mathfrak{m}_1, \mathfrak{m}_2, \ldots, \mathfrak{m}_{2^m}$ are algebraically dependent. Let $h \neq 0$ be an annihilating polynomial of these coefficients. Since h is nonzero, there is a point $\mathbf{a} = (a_1, \ldots, a_{2^m}) \in \mathbb{F}^{2^m}$ such that $h(\mathbf{a}) \neq 0$. It follows that the multilinear polynomial $g \overset{\text{def}}{=} \sum_{i=1}^{2^m} a_i \mathfrak{m}_i$ is not computable by an ABP of width m and length m, which means g is not computable by a full rank ABP. □

7.2 Proof of Lemmas and Claims in Section 2

Claim 2.1 (Restated). *If $f(\mathbf{x}) = g(A\mathbf{x})$, where f and g are both n variate polynomials and $A \in \mathrm{GL}(n)$, then the Lie algebra of f is a conjugate of the Lie algebra of g via A, i.e., $\mathfrak{g}_f = \{A^{-1}EA : E \in \mathfrak{g}_g\} =: A^{-1}\mathfrak{g}_g A$.*

Proof. Let $Q = (q_{i,j})_{i,j \in [n]} \in \mathfrak{g}_f$. Hence,

$$\sum_{i,j \in [n]} q_{ij} x_j \cdot \frac{\partial f}{\partial x_i} = 0 \ \Rightarrow \ \sum_{i,j \in [n]} q_{ij} x_j \cdot \frac{\partial g(A\mathbf{x})}{\partial x_i} = 0. \tag{7}$$

Let $A = (a_{ki})_{k,i \in [n]}$. Using chain rule of derivatives,

$$\frac{\partial g(A\mathbf{x})}{\partial x_i} = \sum_{k \in [n]} \frac{\partial g}{\partial x_k}(A\mathbf{x}) \cdot a_{ki}.$$

Let $A^{-1} = (b_{jl})_{j,l \in [n]}$ and $(A\mathbf{x})_l$ be the lth entry of $A\mathbf{x}$. Then $x_j = \sum_{l \in [n]} b_{jl}(A\mathbf{x})_l$. From Equation (7),

$$\begin{aligned} &\sum_{i,j \in [n]} q_{ij} \cdot \left(\sum_{l \in [n]} b_{jl}(A\mathbf{x})_l \right) \cdot \left(\sum_{k \in [n]} \frac{\partial g}{\partial x_k}(A\mathbf{x}) \cdot a_{ki} \right) = 0, \\ \Rightarrow &\sum_{k,l \in [n]} (A\mathbf{x})_l \cdot \frac{\partial g}{\partial x_k}(A\mathbf{x}) \cdot \left(\sum_{i,j \in [n]} a_{ki} q_{ij} b_{jl} \right) = 0, \\ \Rightarrow &\sum_{k,l \in [n]} x_l \cdot \frac{\partial g}{\partial x_k} \cdot \left(\sum_{i,j \in [n]} a_{ki} q_{ij} b_{jl} \right) = 0 \quad \text{(Substituting } \mathbf{x} \text{ by } A^{-1}\mathbf{x}). \end{aligned}$$

Observe that $\sum_{i,j \in [n]} a_{ki} q_{ij} b_{jl}$ is the (k,l)th entry of AQA^{-1}. Hence, $AQA^{-1} \in \mathfrak{g}_g$ implying $\mathfrak{g}_f \subseteq A^{-1}\mathfrak{g}_g A$. Similarly, $\mathfrak{g}_g \subseteq A\mathfrak{g}_f A^{-1}$ as $g = f(A^{-1}\mathbf{x})$, implying $\mathfrak{g}_f = A^{-1}\mathfrak{g}_g A$. □

Claim 2.2 (Restated). *With probability at least $1 - \frac{1}{poly(n)}$, the rank of the matrix $M = (f_j(\mathbf{b}_i))_{i,j \in [m]}$ is $m - r$ where $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_m$ are chosen independently and uniformly at random from $S^n \subset \mathbb{F}^n$ with $|S| = dm \cdot poly(n)$.*

Proof. Recall, we assumed that the dimension of the $\mathbb{F}$-linear space spanned by the n variate polynomials $f_1, f_2, \ldots, f_m$ is $m - r$. Without loss of generality assume $f_1, f_2, \ldots, f_{m-r}$ form a basis of this linear space. Clearly, the rank of $M = (f_j(\mathbf{b}_i))_{i,j \in [m]}$ is less than or equal to $m - r$. Let $M_{m-r} = (f_j(\mathbf{b}_i))_{i,j \in [m-r]}$. That $\mathrm{Det}(M_{m-r}) \neq 0$ with probability at least $1 - \frac{1}{\mathrm{poly}(n)}$ over the random choices of $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_m$ can be argued as follows: Let $\mathbf{y}_i = \{y_1^{(i)}, y_2^{(i)}, \ldots, y_n^{(i)}\}$ for $i \in [m - r]$ be disjoint sets of variables. Rename the $\mathbf{x} = \{x_1, x_2, \ldots, x_n\}$ variables in $f_j(\mathbf{x})$ to $\mathbf{y}_i$ and call these new polynomials $f_j(\mathbf{y}_i)$ for $i, j \in [m - r]$. Let Y be an $(m - r) \times (m - r)$ matrix whose (i, j)th entry is $(f_j(\mathbf{y}_i))_{i \in [m-r]}$. Since $f_1, f_2, \ldots, f_{m-r}$ are $\mathbb{F}$-linearly independent, $\mathrm{Det}(Y) \neq 0$—this can be argued easily using induction. As $\deg(\mathrm{Det}(Y)) = d(m - r) \leq dm$, by Schwartz-Zippel lemma, $\mathrm{Det}(M_{m-r}) \neq 0$ with probability at least $1 - \frac{1}{\mathrm{poly}(n)}$. □

Claim 2.3 (Restated). *Let r be the number of redundant variables in an n variate polynomial f of degree d. Then the dimension of the space $\mathcal{U}$ of $\mathbb{F}$-linear dependencies of $\{\partial_{x_i} f | i \in [n]\}$ is r. Moreover, we can construct an $A \in \mathrm{GL}(n)$ in randomized $poly(n, d, \beta)$ time such that $f(A\mathbf{x})$ is free of the set of variables $\{x_{n-r+1}, x_{n-r+2}, \ldots, x_n\}$ with high probability, where β is the bit length of the coefficients of f.*

Proof. Let $B = (b_{ij})_{i,j \in [n]} \in \mathrm{GL}(n)$ such that $f(B\mathbf{x})$ is a polynomial in $x_1, x_2, \ldots, x_s$, where $s = n - r$. For $n - r + 1 \leq j \leq n$

$$\begin{aligned} \frac{\partial f(B\mathbf{x})}{\partial x_j} &= 0, \\ \Rightarrow \sum_{i=1}^{n} b_{ij} \cdot \frac{\partial f}{\partial x_i}(B\mathbf{x}) &= 0 \quad \text{(by chain rule)}, \\ \Rightarrow \sum_{i=1}^{n} b_{ij} \cdot \frac{\partial f}{\partial x_i} &= 0 \quad \text{(substituting } \mathbf{x} \text{ by } B^{-1}\mathbf{x}). \end{aligned}$$

Since $B \in \mathrm{GL}(n)$, we conclude $\dim(\mathcal{U}) \geq r$. Let $\{(a_{1j}\ a_{2j}\ \ldots\ a_{nj})^T : (n - \dim(\mathcal{U}) + 1) \leq j \leq n\}$ be a basis of $\mathcal{U}$. Then,

$$\sum_{i=1}^{n} a_{ij} \cdot \frac{\partial f}{\partial x_i} = 0.$$

Let $A \in \mathrm{GL}(n)$ such that for $(n - \dim(\mathcal{U}) + 1) \leq j \leq n$, the jth column of A is $(a_{1j}\ a_{2j}\ \ldots\ a_{nj})^T$ and the remaining columns of A are arbitrary vectors that make A a full rank matrix. Then,

$$\sum_{i=1}^{n} a_{ij} \cdot \frac{\partial f}{\partial x_i} = 0 \quad \Rightarrow \quad \sum_{i=1}^{n} a_{ij} \cdot \frac{\partial f}{\partial x_i}(A\mathbf{x}) = 0 \quad \Rightarrow \quad \frac{\partial f(A\mathbf{x})}{\partial x_j} = 0.$$

This implies $f(A\mathbf{x})$ is a polynomial free of x_j variable for $(n - \dim(\mathcal{U}) + 1) \leq j \leq n$. Hence, $\dim(\mathcal{U}) \leq r$.

Blackbox for polynomials $\partial_{x_1} f, \partial_{x_2} f, \ldots, \partial_{x_n} f$ can be constructed in $\mathrm{poly}(n, d, \beta)$ time from blackbox access to f and a basis for the space $\mathcal{U}$ of $\mathbb{F}$-linear dependencies of polynomials $\partial_{x_1} f, \partial_{x_2} f, \ldots, \partial_{x_n} f$ can also be constructed in randomized $\mathrm{poly}(n, d, \beta)$ time (see Section 2.2). Thus, we can construct an $A \in \mathrm{GL}(n)$ (similar to the construction shown above) from a blackbox access to f in randomized $\mathrm{poly}(n, d, \beta)$ time such that $f(A\mathbf{x})$ is free of the set of variables $\{x_{n-r+1}, x_{n-r+2}, \ldots, x_n\}$. We summarize this in Algorithm 8. □

ALGORITHM 8: Eliminating redundant variables

```
INPUT: Blackbox access to an $n$ variate polynomial $f(\mathbf{x})$.
OUTPUT: An $r$ and an $A \in \mathrm{GL}(n)$ such that $r$ is the number of redundant variables in $f$ and $f(A\mathbf{x})$ is free
of the variables $x_{n-r+1}, x_{n-r+2}, \ldots, x_n$.
Compute blackbox access to $\partial_{x_1} f, \partial_{x_2} f, \ldots, \partial_{x_n} f$ (see Section 2.2).
Compute a basis $\{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_r\}$ of the space of $\mathbb{F}$-linear dependencies of $\partial_{x_1} f, \partial_{x_2} f, \ldots, \partial_{x_n} f$ (using
  the random substitution idea in Claim 2.2). /* This step succeeds in computing the required basis with
  high probability. */
Construct an $A \in \mathrm{GL}(n)$ such that the last $r$ columns of $A$ are $\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_r$ and the remaining columns
  of $A$ are chosen arbitrarily to make $A$ a full rank matrix.
Return $r$ and $A$.
```

Lemma 2.13 (Restated). *There is a randomized algorithm that takes input blackbox access to two n variate, degree d polynomials f and g, and with probability at least $1 - \frac{1}{poly(n)}$ does the following: if f is translation equivalent to g, outputs an $\mathbf{a} \in \mathbb{F}^n$ such that $f(\mathbf{x} + \mathbf{a}) = g(\mathbf{x})$, else outputs "$f$ and g are not translation equivalent." The running time of the algorithm is $poly(n, d, \beta)$, where β is the bit length of the coefficients of f and g.*

Proof. We present the algorithm formally in Algorithm 9. If it succeeds in computing a point $\mathbf{a} \in \mathbb{F}^n$ in the end (in step 20), it performs a randomized blackbox polynomial identity test (PIT) to check whether $f(\mathbf{x} + \mathbf{a}) = g(\mathbf{x})$ (in step 22). If f and g are not translation equivalent, this final PIT finds it with probability at least $1 - \frac{1}{\mathrm{poly}(n)}$. So, for the analysis of the algorithm, we can assume there is an $\mathbf{a} = (a_1\ a_2\ \ldots\ a_n)^T \in \mathbb{F}^n$ such that $f(\mathbf{x} + \mathbf{a}) = g(\mathbf{x})$. The strategy outlined below helps to argue the correctness of Algorithm 9.

Strategy: Suppose $f(\mathbf{x} + \mathbf{a}) = g(\mathbf{x})$. By equating the degree d and degree $d - 1$ homogeneous components of f and g, we get the following equations:

$$f^{[d]} = g^{[d]} \text{ and}$$

$$f^{[d-1]} + \sum_{i=1}^{n} a_i \cdot \frac{\partial f^{[d]}}{\partial x_i} = g^{[d-1]} \Rightarrow \sum_{i=1}^{n} a_i \cdot \frac{\partial f^{[d]}}{\partial x_i} = g^{[d-1]} - f^{[d-1]}. \tag{8}$$

Let $f_i = \frac{\partial f^{[d]}}{\partial x_i}$ for $i \in [n]$. Blackbox access to the homogeneous components of f: $f^{[0]}, f^{[1]}, \ldots, f^{[d]}$, the homogeneous components of g: $g^{[0]}, g^{[1]}, \ldots, g^{[d]}$ and $f_1, f_2, \ldots f_n$ can be constructed from blackbox access to f and g in poly(n, d, β) time (see Section 2.2). If $f_1, f_2, \ldots, f_n$ are $\mathbb{F}$-linearly independent then with high probability over the random choices of $\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_n \in \mathbb{F}^n$ the matrix $(f_j(\mathbf{b}_i))_{i,j\in[n]}$ has full rank (from Claim 2.2). Hence, we can solve for $a_1, a_2, \ldots, a_n$ uniquely from Equation (8). In the general case (when $f_1, f_2, \ldots, f_n$ may be $\mathbb{F}$-linearly dependent), the algorithm repeatedly applies *variable reduction* and *degree reduction* (as described below) to compute $\mathbf{a}$.

Variable reduction. We construct a transformation $A \in \mathrm{GL}(n)$ such that $f^{[d]}(A\mathbf{x})$ has only the essential variables $x_1, \ldots, x_m$ (see Claim 2.3). Let $\tilde{f} = f(A\mathbf{x})$, $\tilde{g} = g(A\mathbf{x})$. It is sufficient to compute a point $\mathbf{b} = (b_1\ b_2\ \ldots\ b_n)^T \in \mathbb{F}^n$ such that $\tilde{f}(\mathbf{x} + \mathbf{b}) = \tilde{g}(\mathbf{x})$ as

$$\tilde{f}(\mathbf{x} + \mathbf{b}) = \tilde{g}(\mathbf{x}) \Rightarrow f(A\mathbf{x} + A\mathbf{b}) = g(A\mathbf{x}) \Rightarrow f(\mathbf{x} + A\mathbf{b}) = g(\mathbf{x}).$$

So, we can choose $\mathbf{a} = A\mathbf{b}$. As in Equation (8),

$$\tilde{f}^{[d]} = \tilde{g}^{[d]} \quad \text{and} \quad \sum_{i=1}^{m} b_i \cdot \frac{\partial \tilde{f}^{[d]}}{\partial x_i} = \tilde{g}^{[d-1]} - \tilde{f}^{[d-1]}. \tag{9}$$

The derivatives $\partial_{x_i}\tilde{f}^{[d]}$ for $i > m$ are zero as $\tilde{f}^{[d]} = f^{[d]}(A\mathbf{x})$ has only the essential variables $x_1, x_2, \ldots, x_m$. Also the polynomials $\{\partial_{x_i}\tilde{f}^{[d]} : i \in [m]\}$ are $\mathbb{F}$-linearly independent (by Claim 2.3). Hence, we can solve for unique $b_1, b_2, \ldots, b_m$ satisfying Equation (9) as before.

Degree reduction. To compute $b_{m+1}, b_{m+2}, \ldots, b_n$, we reduce the problem to finding a point that asserts translation equivalence of two degree $d-1$ polynomials. Let $\mathbf{b}' = (b_1\ b_2\ \ldots\ b_m\ 0\ \ldots\ 0)^T$, $\widehat{f} = \tilde{f}(\mathbf{x} + \mathbf{b}')$. Further, let $\mathbf{e} \in \mathbb{F}^n$ such that $\widehat{f}(\mathbf{x} + \mathbf{e}) = \tilde{g}(\mathbf{x})$. Then the first m coordinates of $\mathbf{e}$ must be zero,[36] and we can choose $\mathbf{b} = \mathbf{b}' + \mathbf{e}$. We have the following equations:

$$\begin{aligned} &\widehat{f}^{[d]}(\mathbf{x} + \mathbf{e}) + (\widehat{f} - \widehat{f}^{[d]})(\mathbf{x} + \mathbf{e}) = \tilde{g}^{[d]}(\mathbf{x}) + (\tilde{g} - \tilde{g}^{[d]})(\mathbf{x}) \\ \Leftrightarrow\ &\tilde{f}^{[d]}(\mathbf{x} + \mathbf{e}) + (\widehat{f} - \widehat{f}^{[d]})(\mathbf{x} + \mathbf{e}) = \tilde{g}^{[d]}(\mathbf{x}) + (\tilde{g} - \tilde{g}^{[d]})(\mathbf{x}) \quad (\text{as } \widehat{f}^{[d]} = \tilde{f}^{[d]}). \end{aligned}$$

Since $\tilde{f}^{[d]}$ has only $x_1, x_2, \ldots, x_m$ variables and the first m coordinates of $\mathbf{e}$ are zero, the above statement is equivalent to

$$\begin{aligned} &\tilde{f}^{[d]}(\mathbf{x}) + (\widehat{f} - \widehat{f}^{[d]})(\mathbf{x} + \mathbf{e}) = \tilde{g}^{[d]}(\mathbf{x}) + (\tilde{g} - \tilde{g}^{[d]})(\mathbf{x}) \\ \Leftrightarrow\ &(\widehat{f} - \widehat{f}^{[d]})(\mathbf{x} + \mathbf{e}) = (\tilde{g} - \tilde{g}^{[d]})(\mathbf{x}) \ \ (\text{from Equation (9)}). \end{aligned}$$

The polynomials $\widehat{f} - \widehat{f}^{[d]}$ and $\tilde{g} - \tilde{g}^{[d]}$ have degree at most $d-1$ and blackboxes for these polynomials can be constructed in poly(n, d, β) time. Therefore the problem reduces to computing a point $\mathbf{e} \in \mathbb{F}^n$ that asserts translation equivalence of two degree $(d-1)$ polynomials.

Correctness of Algorithm 9: In steps 4–11, the algorithm carries out variable reduction and computes a part of the translation $\mathbf{b}$ that we call $\mathbf{b}'$ in the above argument. The remaining part of $\mathbf{b}$ (which is the vector $\mathbf{e}$ above) is computed by carrying out degree reduction in step 12 and then inducting on lower degree polynomials. These parts are then added appropriately in step 17, and finally an $\mathbf{a}$ is recovered in step 20. □

[36] As $b_1, b_2, \ldots, b_m$ can be solved uniquely.

ALGORITHM 9: Translation equivalence test

```
   INPUT: Blackbox access to two $n$ variate, degree $d$ polynomials $f$ and $g$.
   OUTPUT: A point $\mathbf{a} \in \mathbb{F}^n$ such that $f(\mathbf{x} + \mathbf{a}) = g(\mathbf{x})$, if such an $\mathbf{a}$ exists.
 1: Set $\ell = d$, $p = f$ and $q = g$.
 2:
 3: while $\ell > 0$ do
 4:     Using Algorithm 8 find an $m$ and an $A_\ell \in \mathrm{GL}(n)$ such that the variables $x_{m+1}, x_{m+2}, \ldots, x_n$ do not
        appear in $p^{[\ell]}(A_\ell \mathbf{x})$. /* With high probability $m$ is the number of essential variables in $p^{[\ell]}$. */
 5:     Let $\tilde{p} = p(A_\ell \mathbf{x})$ and $\tilde{q} = q(A_\ell \mathbf{x})$. Construct blackbox access to $\tilde{p}^{[\ell]}, \tilde{p}^{[\ell-1]}, \tilde{q}^{[\ell]}, \tilde{q}^{[\ell-1]}$ and $\partial_{x_i} \tilde{p}^{[\ell]}$ for
        $i \in [m]$.
 6:     Check if $\tilde{p}^{[\ell]} = \tilde{q}^{[\ell]}$. If not, output "$f$ and $g$ are not translation equivalent" and stop. /* The check
        succeeds with high probability. */
 7:     Solve for unique $b_1, b_2, \ldots, b_m$ satisfying
            $\sum_{i=1}^{m} b_i \cdot \frac{\partial \tilde{p}^{[\ell]}}{\partial x_i} = \tilde{q}^{[\ell-1]} - \tilde{p}^{[\ell-1]}$   (using the random substitution idea in Claim2.2).
        If the solving fails, output "$f$ and $g$ are not translation equivalent." /* This step succeeds with high
        probability if $m$ is the number of essential variables in $p^{[\ell]}$ in step 4. */
 8:     if $m = n$ then
 9:         Set $\mathbf{b}_\ell = (b_1 \; b_2 \; \ldots \; b_n)$ and exit while loop.
10:     else
11:         Set $\mathbf{b}_\ell = (b_1 \; b_2 \; \ldots b_m \; 0 \; \ldots \; 0) \in \mathbb{F}^n$.
12:         Construct blackbox access to $(\tilde{p} - \tilde{p}^{[\ell]})(\mathbf{x} + \mathbf{b}_\ell)$ and $(\tilde{q} - \tilde{q}^{[\ell]})(\mathbf{x})$. Set $p = (\tilde{p} - \tilde{p}^{[\ell]})(\mathbf{x} + \mathbf{b}_\ell)$,
            $q = (\tilde{q} - \tilde{q}^{[\ell]})(\mathbf{x})$ and $\ell = \ell - 1$.
13:     end if
14: end while
15:
16: while $\ell < d$ do
17:     Set $\mathbf{b}_{\ell+1} = \mathbf{b}_{\ell+1} + A_\ell \mathbf{b}_\ell$.
18:     Set $\ell = \ell + 1$.
19: end while
20: Set $\mathbf{a} = A_d \mathbf{b}_d$.
21:
22: Pick a point $\mathbf{c}$ uniformly at random from $S^n \subset \mathbb{F}^n$ with $|S| = d.\mathrm{poly}(n)$ and check whether
    $f(\mathbf{c} + \mathbf{a}) = g(\mathbf{c})$. /* With high probability $f(\mathbf{c} + \mathbf{a}) \neq g(\mathbf{c})$ if $f$ and $g$ are not translation equivalent.*/
23: if $f(\mathbf{c} + \mathbf{a}) = g(\mathbf{c})$ then
24:     Output the point $\mathbf{a}$.
25: else
26:     Output "$f$ and $g$ are not translation equivalent."
27: end if
```

Lemma 2.14 (Restated). *There is a randomized algorithm that when given blackbox access to an n variate degree d polynomial f, computes a basis of $\mathfrak{g}_f$ with probability at least $1 - \frac{1}{\mathrm{poly}(n)}$ in time poly(n, d, β) where β is the bit length of the coefficients in f.*

Proof. Recall, the Lie algebra of f is the set of all matrices $E = (e_{ij})_{i,j \in [n]}$ such that $\sum_{i,j \in [n]} e_{ij} x_j \cdot \frac{\partial f}{\partial x_i} = 0$. Hence, $\mathfrak{g}_f$ is the space of linear dependencies of the polynomials $x_j \cdot \frac{\partial f}{\partial x_i}$ for $i, j \in [n]$. Using Claim 2.3, we can derive blackboxes for these n^2 polynomials and then compute a basis of the space of linear dependencies with high probability using Claim 2.2. □

7.3 Proof of Lemmas and Claims in Section 3

Lemma 3.1 (Restated). *Let $\mathcal{W}_1, \mathcal{W}_2, \mathcal{W}_3$ be the following sets (spaces) of matrices:*

(1) *$\mathcal{W}_1$ consists of all matrices $D = (d_{ij})_{i,j \in [n]}$ such that D is diagonal and*

$$\sum_{i=1}^{n} d_{ii} x_i \cdot \frac{\partial \mathsf{IMM}}{\partial x_i} = 0.$$

(2) *$\mathcal{W}_2$ consists of all matrices $B = (b_{ij})_{i,j \in [n]}$ such that*

$$\sum_{i,j \in [n]} b_{ij} x_j \cdot \frac{\partial \mathsf{IMM}}{\partial x_i} = 0,$$

where in every summand $b_{ij} \neq 0$ only if $x_i \neq x_j$ and $x_i, x_j \in \mathbf{x}_l$ for some $l \in [d]$.

(3) *$\mathcal{W}_3$ consists of all matrices $C = (c_{ij})_{i,j \in [n]}$ such that*

$$\sum_{i,j \in [n]} c_{ij} x_j \cdot \frac{\partial \mathsf{IMM}}{\partial x_i} = 0,$$

where in every summand $c_{ij} \neq 0$ only if either $x_i \in \mathbf{x}_2$, $x_j \in \mathbf{x}_1$ or $x_i \in \mathbf{x}_{d-1}$, $x_j \in \mathbf{x}_d$.

Then, $\mathfrak{g}_{\mathsf{IMM}} = \mathcal{W}_1 \oplus \mathcal{W}_2 \oplus \mathcal{W}_3$.

Proof. Since $\mathcal{W}_1 \cap \mathcal{W}_2 = (\mathcal{W}_1 + \mathcal{W}_2) \cap \mathcal{W}_3 = \{\mathbf{0}_n\}$, where $\mathbf{0}_n$ is the $n \times n$ all zero matrix, it is sufficient to show $\mathfrak{g}_{\mathsf{IMM}} = \mathcal{W}_1 + \mathcal{W}_2 + \mathcal{W}_3$. By definition, $\mathcal{W}_1 + \mathcal{W}_2 + \mathcal{W}_3 \subseteq \mathfrak{g}_{\mathsf{IMM}}$. We now show that $\mathfrak{g}_{\mathsf{IMM}} \subseteq \mathcal{W}_1 + \mathcal{W}_2 + \mathcal{W}_3$. Let $E = (e_{ij})_{i,j \in [n]}$ be a matrix in $\mathfrak{g}_{\mathsf{IMM}}$. Then $\sum_{i,j \in [n]} e_{ij} x_j \cdot \frac{\partial \mathsf{IMM}}{\partial x_i} = 0$. We focus on a term $x_j \cdot \frac{\partial \mathsf{IMM}}{\partial x_i}$ and observe the following:

(a) If $x_i = x_j$ then the monomials of $x_i \cdot \frac{\partial \mathsf{IMM}}{\partial x_i}$ are also monomials of IMM. Such monomials do not appear in any term $x_j \cdot \frac{\partial \mathsf{IMM}}{\partial x_i}$, where $x_i \neq x_j$.

(b) If $x_i \neq x_j$ and x_i, x_j belong to the same $\mathbf{x}_l$ then every monomial in $x_j \cdot \frac{\partial \mathsf{IMM}}{\partial x_i}$ has exactly one variable from every $\mathbf{x}_k$ for $k \in [d]$. Such monomials do not appear in a term $x_j \cdot \frac{\partial \mathsf{IMM}}{\partial x_i}$, where $x_i \in \mathbf{x}_l$ and $x_j \in \mathbf{x}_k$ and $l \neq k$.

Due to this monomial disjointness, an equation $\sum_{i,j \in [n]} e_{ij} x_j \cdot \frac{\partial \mathsf{IMM}}{\partial x_i} = 0$ corresponding to E can be split into three equations:

(1) $\sum_{i=1}^{n} d_{ii} x_i \cdot \frac{\partial \mathsf{IMM}}{\partial x_i} = 0$.

(2) $\sum_{i,j \in [n]} b_{ij} x_j \cdot \frac{\partial \mathsf{IMM}}{\partial x_i} = 0$, where $b_{ij} \neq 0$ in a term only if $x_i \neq x_j$ and $x_i, x_j \in \mathbf{x}_l$ for some $l \in [d]$.

(3) $\sum_{i,j \in [n]} c_{ij} x_j \cdot \frac{\partial \mathsf{IMM}}{\partial x_i} = 0$, where $c_{ij} \neq 0$ in a term only if $x_i \in \mathbf{x}_l$ and $x_j \in \mathbf{x}_k$ for $l \neq k$.

Hence, every $E = (e_{ij})_{i,j \in [n]}$ in $\mathfrak{g}_{\mathsf{IMM}}$ equals $D + B + C$ where

- $D \in \mathcal{W}_1$ is a diagonal matrix,
- $B \in \mathcal{W}_2$ is a block-diagonal[37] matrix with diagonal entries zero,
- C is a matrix with nonzero entries appearing outside the above block-diagonal.

To complete the proof of the lemma, we show the following.

[37] An entry is in the block-diagonal if and only if the variables labelling the row and column of the entry are in the same $\mathbf{x}_l$ for some $l \in [d]$.

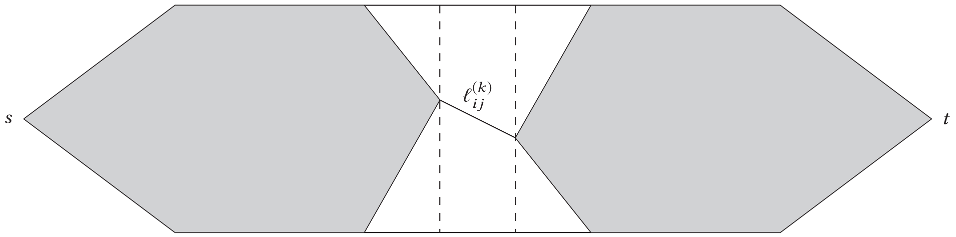

Fig. 6. An ABP computing the term $\ell_{ij}^{(k)} \cdot \frac{\partial \mathsf{IMM}}{\partial x_{ij}^{(k)}}$.

Claim 7.1. *Except those entries of C whose rows and columns are indexed by $\mathbf{x}_2$ and $\mathbf{x}_1$ variables, respectively, or $\mathbf{x}_{d-1}$ and $\mathbf{x}_d$ variables, respectively, all the other entries are zero.*

Proof. In a term $x_{pq}^{(l)} \cdot \frac{\partial \mathsf{IMM}}{\partial x_{ij}^{(k)}}$ where $l \neq k$, every monomial has two variables from $\mathbf{x}_l$ and no variable from $\mathbf{x}_k$. Hence, from the equation corresponding to C, we get separate equations for every pair (l, k) due to monomial disjointness:

$$\sum_{p\in[w_{l-1}],q\in[w_l]} \; \sum_{i\in[w_{k-1}],j\in[w_k]} c_{pq,ij} x_{pq}^{(l)} \cdot \frac{\partial \mathsf{IMM}}{\partial x_{ij}^{(k)}} = 0, \quad \text{where } l \neq k.$$

Collecting coefficients corresponding to $\frac{\partial \mathsf{IMM}}{\partial x_{ij}^{(k)}}$ in the above equation, we get

$$\sum_{i\in[w_{k-1}],j\in[w_k]} \ell_{ij}^{(k)} \cdot \frac{\partial \mathsf{IMM}}{\partial x_{ij}^{(k)}} = 0, \quad \text{where } \ell_{ij}^{(k)} \text{ is a linear form in the variables from } \mathbf{x}_l. \tag{10}$$

Figure 6 depicts a term $\ell_{ij}^{(k)} \cdot \frac{\partial \mathsf{IMM}}{\partial x_{ij}^{(k)}}$ using an ABP that computes it. So the LHS of the above equation can be computed by an ABP B that has edge labels identical to that of the ABP for IMM, except for the edges in layer k. The (i, j)th edge of layer k in B is labelled by $\ell_{ij}^{(k)}$. Suppose $\ell_{ij}^{(k)} \neq 0$ and the coefficient of the variable $x_{pq}^{(l)}$ in $\ell_{ij}^{(k)}$ is nonzero, i.e., $c_{pq,ij} \neq 0$. If (l, k) is neither $(1, 2)$ nor $(d, d-1)$ then the assumption $c_{pq,ij} \neq 0$ leads to a contradiction as follows.

Consider an s to t path P in B that goes through the (i, j)th edge of layer k (which is labelled by $\ell_{ij}^{(k)}$) but excludes the (p, q)th edge of layer l (which is labelled by $x_{pq}^{(l)}$), the (p, i)th edge of layer $k-1$ if $l = k-1$ and the (j, q)th edge of layer $k+1$ if $l = k+1$ (we can notice this is always possible, since (l, k) is neither $(1, 2)$ nor $(d, d-1)$). Then, if we retain the variables labelling the edges of P outside the layer k and the variable $x_{pq}^{(l)}$, and set every other variable to zero then P becomes the unique s to t path in B with nonzero weight (since $c_{pq,ij} \neq 0$). But this contradicts the fact that ABP B is computing an identically zero polynomial (by Equation (10)). □

Therefore, $\mathfrak{g}_{\mathsf{IMM}} \subseteq \mathcal{W}_1 + \mathcal{W}_2 + \mathcal{W}_3$ implying $\mathfrak{g}_{\mathsf{IMM}} = \mathcal{W}_1 \oplus \mathcal{W}_2 \oplus \mathcal{W}_3$.

Lemma 3.2 (Restated). The space $\mathcal{W}_3 = \mathcal{W}_3^{(a)} \oplus \mathcal{W}_3^{(b)}$ *where* $\mathcal{W}_3^{(a)} = \mathcal{A}_1 \oplus \mathcal{A}_2 \oplus \cdots \oplus \mathcal{A}_{w_2}$ *and* $\mathcal{W}_3^{(b)} = \mathcal{A}'_1 \oplus \mathcal{A}'_2 \oplus \cdots \oplus \mathcal{A}'_{w_{d-2}}$ *such that for every* $i \in [w_2]$ $\mathcal{A}_i$ *is isomorphic to the space of* $w_1 \times w_1$ *anti-symmetric matrices over* $\mathbb{F}$, *and for every* $j \in [w_{d-2}]$ $\mathcal{A}'_j$ *is isomorphic to the space of* $w_{d-1} \times w_{d-1}$ *anti-symmetric matrices over* $\mathbb{F}$. *Hence,* $\dim(\mathcal{W}_3) = \frac{1}{2}[w_1 w_2(w_1 - 1) + w_{d-1} w_{d-2}(w_{d-1} - 1)]$.

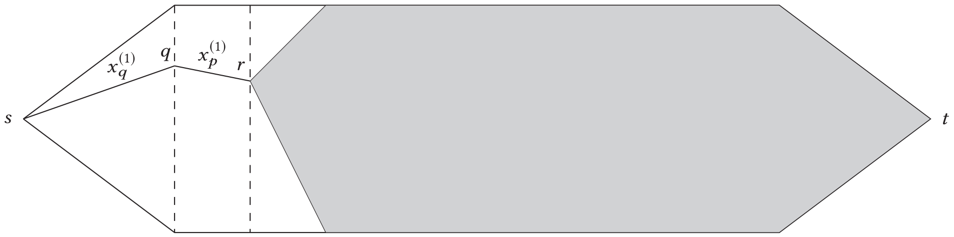

Fig. 7. An ABP computing the term $x_p^{(1)} \cdot \frac{\partial \mathsf{IMM}}{\partial x_{qr}^{(2)}}$.

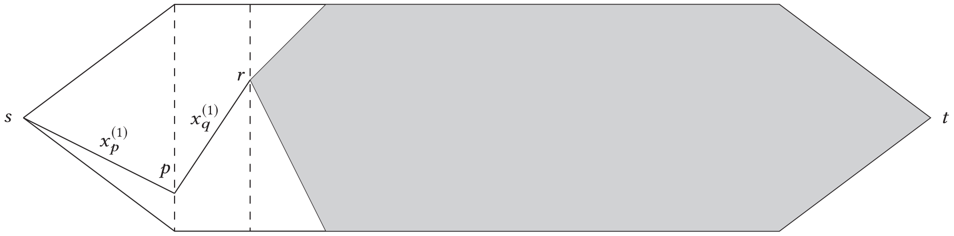

Fig. 8. An ABP computing the term $x_q^{(1)} \cdot \frac{\partial \mathsf{IMM}}{\partial x_{pr}^{(2)}}$.

Proof. Recall, $\mathcal{W}_3$ is the space of all matrices $C = (c_{ij})_{i,j \in [n]}$ such that

$$\sum_{i,j \in [n]} c_{ij} x_j \cdot \frac{\partial \mathsf{IMM}}{\partial x_i} = 0, \tag{11}$$

where in every nonzero summand either $x_i \in \mathbf{x}_2, x_j \in \mathbf{x}_1$ or $x_i \in \mathbf{x}_{d-1}, x_j \in \mathbf{x}_d$. In Equation (11) every monomial in a term $x_p^{(1)} \cdot \frac{\partial \mathsf{IMM}}{\partial x_{qr}^{(2)}}$ has two variables from $\mathbf{x}_1$. Similarly, every monomial in a term $x_p^{(d)} \cdot \frac{\partial \mathsf{IMM}}{\partial x_{qr}^{(d-1)}}$ has two variables from $\mathbf{x}_d$, respectively. Owing to monomial disjointness, Equation (11) gives two equations,

$$\textstyle\sum_{r \in [w_2]} \sum_{p,q \in [w_1]} c_{pqr}^{(1)} x_p^{(1)} \cdot \frac{\partial \mathsf{IMM}}{\partial x_{qr}^{(2)}} = 0, \quad \text{and} \tag{12}$$

$$\textstyle\sum_{q \in [w_{d-2}]} \sum_{p,r \in [w_{d-1}]} c_{pqr}^{(d)} x_p^{(d)} \cdot \frac{\partial \mathsf{IMM}}{\partial x_{qr}^{(d-1)}} = 0. \tag{13}$$

Thus, $\mathcal{W}_3 = \mathcal{W}_3^{(a)} \oplus \mathcal{W}_3^{(b)}$ where $\mathcal{W}_3^{(a)}$ consists of matrices satisfying Equation (12) and $\mathcal{W}_3^{(b)}$ consists of matrices satisfying Equation (13). We argue the following about $\mathcal{W}_3^{(a)}$.

Claim 7.2. *$\mathcal{W}_3^{(a)} = \mathcal{A}_1 \oplus \mathcal{A}_2 \oplus \cdots \oplus \mathcal{A}_{w_2}$ where every $\mathcal{A}_i$ is isomorphic to the space of $w_1 \times w_1$ anti-symmetric matrices over $\mathbb{F}$.*

Proof. Figure 7 depicts an ABP computing the term $x_p^{(1)} \cdot \frac{\partial \mathsf{IMM}}{\partial x_{qr}^{(2)}}$. Every monomial in $c_{pqr}^{(1)} x_p^{(1)} \cdot \frac{\partial \mathsf{IMM}}{\partial x_{qr}^{(2)}}$ is divisible by $x_p^{(1)} x_q^{(1)}$. The only other term in Equation (12) that contains monomials divisible by $x_p^{(1)} x_q^{(1)}$ is $c_{qpr}^{(1)} x_q^{(1)} \cdot \frac{\partial \mathsf{IMM}}{\partial x_{pr}^{(2)}}$. Figure 8 depicts an ABP computing $x_q^{(1)} \cdot \frac{\partial \mathsf{IMM}}{\partial x_{pr}^{(2)}}$. Since the terms in Figures 7 and 8 have no monomials in common with any other term in Equation (12) it must be that $c_{pqr}^{(1)} = -c_{qpr}^{(1)}$. Moreover, if $p = q$ then $c_{pqr}^{(1)} = 0$. Thus, Equation (12) gives an equation for every

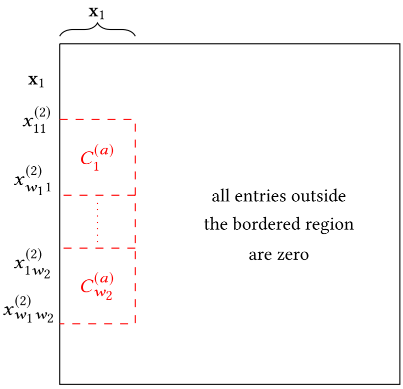

Fig. 9. A matrix $C^{(a)}$ in $\mathcal{W}_3^{(a)}$.

$r \in [w_2]$,

$$\sum_{p,q\in[w_1],p\neq q} c_{pqr}^{(1)} x_p^{(1)} \cdot \frac{\partial \mathrm{IMM}}{\partial x_{qr}^{(2)}} = 0, \tag{14}$$

such that the matrix $C_r = (c_{pqr}^{(1)})_{p,q\in[w_1]} \in \mathbb{F}^{w_1\times w_1}$ is anti-symmetric. Further any anti-symmetric matrix can be used to get an equation like Equation (14). Thus, as shown in Figure 9, every matrix $C^{(a)} \in \mathcal{W}_3^{(a)}$ is such that for every $r \in [w_2]$, the $w_1 \times w_1$ submatrix (say $C_r^{(a)}$) defined by the rows labelled by the $x_{qr}^{(2)}$ variables and the columns labelled by the $x_p^{(1)}$ variables for $p, q \in [w_1]$ is anti-symmetric. Also, any matrix satisfying the above properties belongs to $\mathcal{W}_3^{(a)}$. Naturally, if we define $\mathcal{A}_r$ to be the space of $n \times n$ matrices such that the $w_1 \times w_1$ submatrix defined by the rows labelled by the $x_{qr}^{(2)}$ variables and the columns labelled by the $x_p^{(1)}$ variables for $p, q \in [w_1]$ is anti-symmetric and all other entries are zero then $\mathcal{W}_3^{(a)} = \mathcal{A}_1 \oplus \mathcal{A}_2 \oplus \cdots \oplus \mathcal{A}_{w_2}$. □

Similarly, it can be shown that $\mathcal{W}_3^{(b)} = \mathcal{A'}_1 \oplus \mathcal{A'}_2 \oplus \cdots \oplus \mathcal{A'}_{w_{d-2}}$ where every $\mathcal{A'}_i$ is isomorphic to the space of $w_{d-1} \times w_{d-1}$ anti-symmetric matrices. This completes the proof of Lemma 3.2.

Lemma 3.3 (Restated). *The space $\mathcal{W}_2 = \mathcal{B}_1 \oplus \mathcal{B}_2 \oplus \cdots \oplus \mathcal{B}_{d-1}$ such that for every $k \in [d-1]$, $\mathcal{B}_k$ is isomorphic to the $\mathbb{F}$-linear space spanned by $t_k \times t_k$ matrices of the form*

$$\begin{bmatrix} -Z^T \otimes I_{w_{k-1}} & 0 \\ 0 & I_{w_{k+1}} \otimes Z \end{bmatrix}_{t_k\times t_k} \quad \text{where } Z \in \mathcal{Z}_{w_k} \text{ and } t_k = w_k(w_{k-1}+w_{k+1}).$$

Hence, $\dim(\mathcal{W}_2) = \sum_{k=1}^{d-1}(w_k^2 - w_k)$.

Proof. Recall $w_0 = w_d = 1$ and $\mathcal{Z}_{w_k}$ denotes the space of $w_k \times w_k$ matrix with diagonal entries 0, and $\mathcal{W}_2$ is the space of all matrices $B = (b_{ij})_{i,j\in[n]}$ such that

$$\sum_{i,j\in[n]} b_{ij} x_j \cdot \frac{\partial \mathrm{IMM}}{\partial x_i} = 0, \tag{15}$$

where in every term $b_{ij} \neq 0$ only if $x_i \neq x_j$ and $x_i, x_j \in \mathbf{x}_l$ for some $l \in [d]$. The following observation is easy to verify.

Observation 7.2. *Suppose $l \in [2, d-1]$. A term $x_{i_1 j_1}^{(l)} \cdot \frac{\partial \mathrm{IMM}}{\partial x_{i_2 j_2}^{(l)}}$ where $i_1 \neq i_2$ and $j_1 \neq j_2$ does not share a monomial with any other term in Equation (15).*

Hence, for $l \in [2, d-1]$, terms of the kind $x^{(l)}_{i_1 j_1} \cdot \frac{\partial \mathsf{IMM}}{\partial x^{(l)}_{i_2 j_2}}$ where $i_1 \neq i_2$ and $j_1 \neq j_2$ are absent in Equation (15). A monomial appearing in a nonzero term of Equation (15) is of the form $x^{(1)}_{i_1} \cdot x^{(2)}_{i_1 i_2} \cdots x^{(k)}_{i_{k-1} i_k} \cdot x^{(k+1)}_{i'_k i_{k+1}} \cdots\ x^{(d-1)}_{i_{d-1} i_d} \cdot x^{(d)}_{i_d}$ where $i_k \neq i'_k$, for some $k \in [d-1]$. We say such a monomial is broken at the kth interface. Observe the following.

Observation 7.3. *The terms $x^{(k)}_{pr} \cdot \frac{\partial \mathsf{IMM}}{\partial x^{(k)}_{pq}}$ where $p \in [w_{k-1}], q, r \in [w_k], q \neq r$, and $x^{(k+1)}_{mj} \cdot \frac{\partial \mathsf{IMM}}{\partial x^{(k+1)}_{ij}}$ where $i, m \in [w_k], j \in [w_{k+1}], i \neq m$ are the only two whose monomials are broken at the kth interface.*

Thus, from Equation (15), we get $(d-1)$ equations one for each interface by considering cancellations of monomials broken at that interface. For $k \in [2, d-2]$, let $\mathcal{B}_k$ be the space of all $n \times n$ matrices B_k such that

(1) the entry corresponding to the row labelled by $x^{(k)}_{pq}$ and the column labelled by $x^{(k)}_{pr}$ is $b^{(k)}_{pq,pr} \in \mathbb{F}$ for $p \in [w_{k-1}], q, r \in [w_k]$ and $q \neq r$,

(2) the entry corresponding to the row labelled by $x^{(k+1)}_{ij}$ and the column labelled by $x^{(k+1)}_{mj}$ is $b^{(k+1)}_{ij,mj} \in \mathbb{F}$ for $i, m \in [w_k], j \in [w_{k+1}]$ and $i \neq m$,

(3) all other entries of B_k are zero, and

(4)
$$\sum_{p \in [w_{k-1}], q, r \in [w_k], q \neq r} b^{(k)}_{pq,pr} x^{(k)}_{pr} \cdot \frac{\partial \mathsf{IMM}}{\partial x^{(k)}_{pq}} + \sum_{i, m \in [w_k], j \in [w_{k+1}], i \neq m} b^{(k+1)}_{ij,mj} x^{(k+1)}_{mj} \cdot \frac{\partial \mathsf{IMM}}{\partial x^{(k+1)}_{ij}} = 0. \qquad (16)$$

We can define spaces $\mathcal{B}_1$ and $\mathcal{B}_{d-1}$ similarly considering monomials broken at the first and the last interface, respectively. As Equation (15) can be split into $(d-1)$ equations, one for every interface, $\mathcal{W}_2 = \mathcal{B}_1 + \mathcal{B}_2 + \cdots + \mathcal{B}_{d-1}$. Since the spaces $\mathcal{B}_1, \ldots, \mathcal{B}_{d-1}$ control different entries of $n \times n$ matrices, $\mathcal{W}_2 = \mathcal{B}_1 \oplus \mathcal{B}_2 \oplus \cdots \oplus \mathcal{B}_{d-1}$.

Claim 7.3. *For $k \in [2, d-2]$, $\mathcal{B}_k$ is isomorphic to the $\mathbb{F}$-linear space spanned by $t_k \times t_k$ matrices of the form*

$$\begin{bmatrix} -Z^T \otimes I_{w_{k-1}} & 0 \\ 0 & I_{w_{k+1}} \otimes Z \end{bmatrix}_{t_k \times t_k} \quad \textit{where } Z \in \mathcal{Z}_{w_k} \textit{ and } t_k = w_k(w_{k-1} + w_{k+1}).$$

Proof. Collecting the same derivative terms in Equation (16), we get

$$\sum_{p \in [w_{k-1}], q \in [w_k]} \ell^{(k)}_{pq} \cdot \frac{\partial \mathsf{IMM}}{\partial x^{(k)}_{pq}} + \sum_{i \in [w_k], j \in [w_{k+1}]} \ell^{(k+1)}_{ij} \cdot \frac{\partial \mathsf{IMM}}{\partial x^{(k+1)}_{ij}} = 0, \qquad (17)$$

where $\ell^{(k)}_{pq}$ is a linear form containing variables $x^{(k)}_{pr}$ such that $r \neq q$, and $\ell^{(k+1)}_{ij}$ is a linear form containing variables $x^{(k+1)}_{mj}$ such that $m \neq i$. Here is a succinct way to write Equation (17):

$$Q_1 \cdot Q_2 \cdots Q'_k \cdot Q_{k+1} \cdot Q_{k+2} \cdots Q_{d-1} \cdot Q_d + Q_1 \cdot Q_2 \cdots Q_k \cdot Q'_{k+1} \cdot Q_{k+2} \cdots Q_{d-1} \cdot Q_d = 0, \qquad (18)$$

where $Q_1, \ldots, Q_d$ are matrices as in Section 2.3, $Q'_k = (\ell^{(k)}_{pq})_{p \in [w_{k-1}], q \in [w_k]}$ and $Q'_{k+1} = (\ell^{(k+1)}_{ij})_{i \in [w_k], j \in [w_{k+1}]}$. This implies

$$Q'_k \cdot Q_{k+1} + Q_k \cdot Q'_{k+1} = 0,$$

as $Q_1, \ldots, Q_d$ have distinct sets of variables, and the variables appearing in Q'_k and Q'_{k+1} are the same as in Q_k and Q_{k+1}, respectively. The variable disjointness of Q_k and Q_{k+1} can be exploited to infer $Q'_{k+1} = Z \cdot Q_{k+1}$ and $Q'_k = -Q_k \cdot Z$, where Z is in $\mathbb{F}^{w_k \times w_k}$ (even if Q_k, Q_{k+1} may not be square matrices). As the linear form $\ell_{pq}^{(k)}$ is devoid of the variable $x_{pq}^{(k)}$, it must be that $Z \in \mathcal{Z}_{w_k}$. Moreover, any $Z \in \mathcal{Z}_{w_k}$ can be used along with the relations $Q'_{k+1} = Z \cdot Q_{k+1}$ and $Q'_k = -Q_k \cdot Z$ to satisfy Equation (18) and hence also Equations (16) and (17).

Let $Z = (z_{im})_{i,m\in[w_k]}$. Since $Q'_{k+1} = Z \cdot Q_{k+1}$, the coefficient of $x_{mj}^{(k+1)}$ in $\ell_{ij}^{(k+1)}$ is z_{im} for every $j \in [w_{k+1}]$. Hence, in Equation (16), $b_{ij,mj}^{(k+1)} = z_{im}$ for every $j \in [w_{k+1}]$. Similarly, since $Q'_k = -Q_k \cdot Z$ the coefficient of $x_{pr}^{(k)}$ in $\ell_{pq}^{(k)}$ is $-z_{rq}$ for every $p \in [w_{k-1}]$. Hence, in Equation (16) $b_{pq,pr}^{(k)} = -z_{rq}$ for every $p \in [w_{k-1}]$. Thus, the submatrix of B_k defined by the rows and columns labelled by the variables in $\mathbf{x}_k$ and $\mathbf{x}_{k+1}$ looks like

$$\begin{bmatrix} -Z^T \otimes I_{w_{k-1}} & 0 \\ 0 & I_{w_{k+1}} \otimes Z \end{bmatrix}_{t_k \times t_k},$$

where $t_k = w_k(w_{k-1} + w_{k+1})$ and all other entries in B_k are zero. Hence, $\mathcal{B}_k$ is isomorphic to the space generated by $t_k \times t_k$ matrices of the above kind. This proves the claim. □

We can similarly show that $\mathcal{B}_1$ is isomorphic to the space generated by square matrices of the form

$$\begin{bmatrix} -Z^T & 0 \\ 0 & I_{w_2} \otimes Z \end{bmatrix}_{t_1 \times t_1}, \quad \text{where } Z \in \mathcal{Z}_{w_1} \text{ and } t_1 = w_1 + w_1 w_2,$$

and $\mathcal{B}_{d-1}$ is isomorphic to the space generated by square matrices of the form

$$\begin{bmatrix} -Z^T \otimes I_{w_{d-2}} & 0 \\ 0 & Z \end{bmatrix}_{t_{d-1} \times t_{d-1}}, \quad \text{where } Z \in \mathcal{Z}_{w_{d-1}} \text{ and } t_{d-1} = w_{d-1}w_{d-2} + w_{d-1}.$$

This completes the proof of Lemma 3.3. □

Lemma 3.4 (Restated). *The space $\mathcal{W}_1$ contains the space $\mathcal{D}_1 \oplus \mathcal{D}_2 \oplus \cdots \oplus \mathcal{D}_{d-1}$ such that for every $k \in [d-1]$, $\mathcal{D}_k$ is isomorphic to the $\mathbb{F}$-linear space spanned by $t_k \times t_k$ matrices of the form*

$$\begin{bmatrix} -Y \otimes I_{w_{k-1}} & 0 \\ 0 & I_{w_{k+1}} \otimes Y \end{bmatrix}_{t_k \times t_k} \quad \text{where } Y \in \mathcal{Y}_{w_k} \text{ and } t_k = w_k(w_{k-1} + w_{k+1}).$$

Hence, $\dim(\mathcal{W}_1) \geq \sum_{k=1}^{d-1} w_k$.

Proof. The proof is similar to the proof of Lemma 3.3. Recall $w_0 = w_d = 1$ and $\mathcal{Y}_{w_k}$ denotes the space of $w_k \times w_k$ diagonal matrices. Every $D \in \mathcal{W}_1$ satisfies an equation of the following form

$$\sum_{i\in[w_1]} d_i^{(1)} x_i^{(1)} \cdot \frac{\partial \mathsf{IMM}}{\partial x_i^{(1)}} + \sum_{k=2}^{d-1} \sum_{i\in[w_{k-1}], j\in[w_k]} d_{ij}^{(k)} x_{ij}^{(k)} \cdot \frac{\partial \mathsf{IMM}}{\partial x_{ij}^{(k)}} + \sum_{i\in[w_{d-1}]} d_i^{(d)} x_i^{(d)} \cdot \frac{\partial \mathsf{IMM}}{\partial x_i^{(d)}} = 0.$$

A succinct way to write the above equation is

$$\sum_{k=1}^{d} Q_1 Q_2 \cdots Q_{k-1} Q'_k Q_{k+1} \cdots Q_d = 0, \tag{19}$$

where $Q'_1 = (d_i^{(1)} x_i^{(1)})_{i\in[w_1]}$ is a row vector, $Q'_d = (d_i^{(d)} x_i^{(d)})^T_{i\in[w_{d-1}]}$ is a column vector, $Q'_k = (d_{ij}^{(k)} x_{ij}^{(k)})_{i\in[w_{k-1}], j\in[w_k]}$, and $Q_1, \ldots, Q_d$ are matrices as in Section 2.3. For every $k \in [d-1]$, let us focus on those $D_k \in \mathcal{W}_1$ for which the matrices $Q'_1, \ldots, Q'_{k-1}, Q'_{k+2}, \ldots, Q'_d$ are zero in Equation (19). Such a D_k satisfies the following equation,

$$Q_1 \cdot Q_2 \cdot \cdots \cdot Q'_k \cdot Q_{k+1} \cdots Q_d + Q_1 \cdot Q_2 \cdots Q_k \cdot Q'_{k+1} \cdots Q_d = 0. \tag{20}$$

Using a similar argument as in the proof of Lemma 3.3, we get $Q'_{k+1} = Y \cdot Q_{k+1}$ and $Q'_k = -Q_k \cdot Y$ where $Y \in \mathcal{Y}_{w_k}$. Further, any $Y \in \mathcal{Y}_{w_k}$ can be used along with the relations $Q'_{k+1} = Y \cdot Q_{k+1}$ and

$Q'_k = -Q_k \cdot Y$ to satisfy Equation (20). The set of $D_k \in \mathcal{W}_1$ satisfying Equation (20) forms an $\mathbb{F}$-linear space; call it $\mathcal{D}_k$. Every $D_k \in \mathcal{D}_k$ is such that the submatrix defined by the rows and the columns labelled by the variables in $\mathbf{x}_k$ and $\mathbf{x}_{k+1}$ looks like

$$\begin{bmatrix} -Y \otimes I_{w_{k-1}} & 0 \\ 0 & I_{w_{k+1}} \otimes Y \end{bmatrix}_{t_k \times t_k}, \quad \text{where } Y \in \mathcal{Y}_{w_k} \text{ and } t_k = w_k(w_{k-1} + w_{k+1}),$$

and all other entries in D_k are zero. Moreover, any $n \times n$ matrix with this structure is in $\mathcal{D}_k$. Thus, $\mathcal{D}_k$ is isomorphic to the space of all $t_k \times t_k$ matrices of the form shown above. It can also be easily verified that every matrix in $\mathcal{D}_1 + \cdots + \mathcal{D}_{d-1}$ can be expressed *uniquely* as a sum of matrices in these spaces. Hence, $\mathcal{W}_1 \supseteq \mathcal{D}_1 \oplus \mathcal{D}_2 \oplus \cdots \oplus \mathcal{D}_{d-1}$ completing the proof of Lemma 3.4. □

Claim 3.3 (Restated). *No invariant subspace of $\mathfrak{g}_{\mathsf{IMM}}$ is properly contained in $\mathcal{U}_k$ for $k \in [2, d-1]$.*

Proof. Let $\mathcal{U} \subseteq \mathcal{U}_k$ be an invariant subspace of $\mathfrak{g}_{\mathsf{IMM}}$. From Claim 3.2 it follows that $\mathcal{U}$ is a coordinate subspace. For $t \in \mathbb{N}$, let $\tilde{1}_t \overset{\text{def}}{=} 1_t - I_t$, where 1_t is the $t \times t$ all one matrix. From Lemma 3.3, there are matrices B_{k-1} and B_k in $\mathfrak{g}_{\mathsf{IMM}}$ such that the submatrix of B_{k-1} restricted to the rows and the columns labelled by the variables in $\mathbf{x}_{k-1} \uplus \mathbf{x}_k$ looks like

$$\begin{bmatrix} -\tilde{1}_{w_{k-1}} \otimes I_{w_{k-2}} & 0 \\ 0 & I_{w_k} \otimes \tilde{1}_{w_{k-1}} \end{bmatrix}, \text{ and}$$

the submatrix in B_k restricted to the rows and the columns labelled by the variables in $\mathbf{x}_k \uplus \mathbf{x}_{k+1}$ looks like

$$\begin{bmatrix} \tilde{1}_{w_k} \otimes I_{w_{k-1}} & 0 \\ 0 & I_{w_{k+1}} \otimes -\tilde{1}_{w_k} \end{bmatrix}.$$

From Lemma 3.4, there is a diagonal matrix D_{k-1} in $\mathfrak{g}_{\mathsf{IMM}}$ such that the submatrix restricted to the rows and the columns labelled by the variables in $\mathbf{x}_{k-1} \uplus \mathbf{x}_k$ looks like

$$\begin{bmatrix} -I_{w_{k-1}} \otimes I_{w_{k-2}} & 0 \\ 0 & I_{w_k} \otimes I_{w_{k-1}} \end{bmatrix}.$$

Let $L = B_{k-1} + B_k + D_{k-1}$. The submatrix of L restricted to the rows and the columns labelled by the variables in $\mathbf{x}_k$ looks as shown in Figure 10. For notational simplicity, we write w_{k-1} as w in Figure 10. If e_x is a unit vector in $\mathcal{U}$, where $x = x_{ij}^{(k)}$ is a variable in $\mathbf{x}_k$ then the matrix L maps e_x to Le_x, which is the column of L labelled by the variable x. This column vector has all entries zero except for the rows labelled by the variables in $\mathbf{x}_k$. Restricting to these rows and looking at Figure 10, we infer that the rows of Le_x labelled by the variables $x_{1j}^{(k)}, x_{2j}^{(k)}, \ldots, x_{w_{k-1}j}^{(k)}$ are 1 (in particular, these entries are nonzero). We use this knowledge and that $Le_x \in \mathcal{U}$ to make the following observation, the proof of which is immediate from Claim 3.2.

Observation 7.4. *If $e_x \in \mathcal{U}$, where $x = x_{ij}^{(k)}$ then $e_{x'} \in \mathcal{U}$ for every $x' \in \{x_{1j}^{(k)}, x_{2j}^{(k)}, \ldots, x_{w_{k-1}j}^{(k)}\}$.*

Moreover, it follows from the presence of I_w matrices in Figure 10 that for every $j' \in [w_k]$ there is the variable $y = x_{ij'}^{(k)}$ such that the row labelled by y in Le_x is 1, implying[38] $e_y \in \mathcal{U}$. Hence, from Observation 7.4, $e_{y'} \in \mathcal{U}$ for every $y' \in \{x_{1j'}^{(k)}, \ldots, x_{w_{k-1}j'}^{(k)}\}$. Since this is true for every $j' \in [w_k]$, $e_y \in \mathcal{U}$ for every variable $y \in \mathbf{x}_k$ implying $\mathcal{U} = \mathcal{U}_k$.

Claim 3.4 (Restated). *The invariant subspaces $\mathcal{U}_{1,2}$ and $\mathcal{U}_{d-1,d}$ are irreducible, and the only invariant subspace properly contained in $\mathcal{U}_{1,2}$ (respectively, $\mathcal{U}_{d-1,d}$) is $\mathcal{U}_2$ (respectively, $\mathcal{U}_{d-1}$).*

[38] Follows again from Claim 3.2.

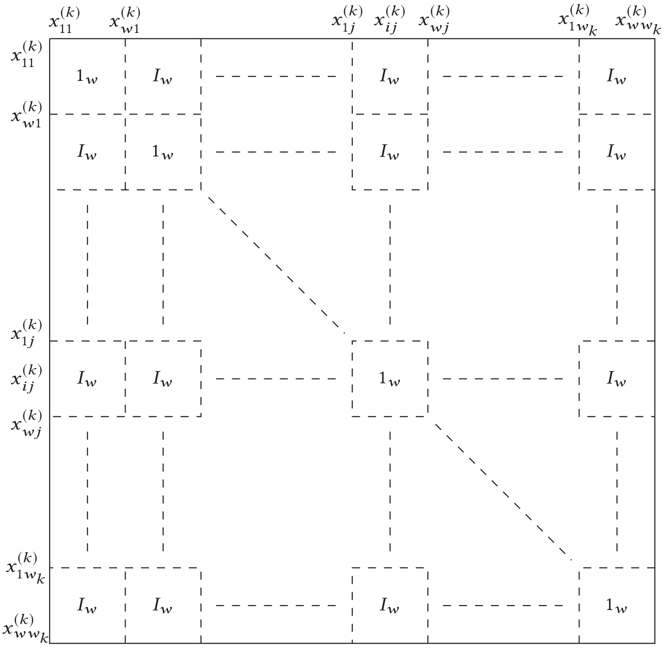

Fig. 10. Submatrix of L restricted to rows/columns indexed by $\mathbf{x}_k$.

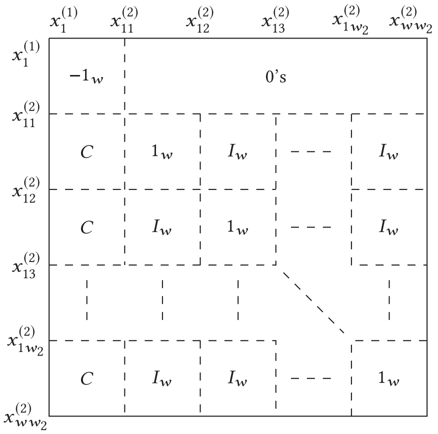

Fig. 11. Submatrix of M matrix restricted to rows/columns indexed by $\mathbf{x}_1 \uplus \mathbf{x}_2$.

Proof. We prove the claim for $\mathcal{U}_{1,2}$, the proof for $\mathcal{U}_{d-1,d}$ is similar. Suppose $\mathcal{U}_{1,2} = \mathcal{V} \oplus \mathcal{W}$ where $\mathcal{V}, \mathcal{W}$ are invariant subspaces of $\mathfrak{g}_{\mathsf{IMM}}$ (and so also coordinate subspaces). A unit vector e_x, where $x \in \mathbf{x}_1$ is either in $\mathcal{V}$ or $\mathcal{W}$. Suppose $e_x \in \mathcal{V}$; we will show that $\mathcal{V} = \mathcal{U}_{1,2}$. Without loss of generality, let $x = x_1^{(1)}$. Arguing as in the proof of the previous claim, we infer that there is a matrix $M \in \mathfrak{g}_{\mathsf{IMM}}$ such that the submatrix of M restricted to the rows and the columns labelled by the variables in $\mathbf{x}_1$ and $\mathbf{x}_2$ looks as shown in Figure 11, in which $w = w_1$ and C is a $w_1 \times w_1$ anti-symmetric matrix with all non-diagonal entries nonzero. All the other entries of M are zero. The vector Me_x is the first column of M and it is zero everywhere except for the rows labelled by the variables in $\mathbf{x}_1 \uplus \mathbf{x}_2$. Among these rows, unless $y \in \{x_{11}^{(2)}, x_{12}^{(2)}, \ldots, x_{1w_2}^{(2)}\}$ the row of Me_x labelled by

y is nonzero. Thus (from Claim 3.2), $e_y \in \mathcal{V}$ for $y \in \mathbf{x}_1$ and $y = x_{ij}^{(2)}$ where $i \in [2, w_1]$ and $j \in [w_2]$. Let $y = x_{ij}^{(2)}$ for some $i \in [2, w_1]$ and $j \in [w_2]$. From Figure 11, the row of Me_y labelled by $x_{1j}^{(2)}$ is nonzero and so, for $y' = x_{1j}^{(2)}$, $e_{y'}$ is also in $\mathcal{V}$. Hence, $\mathcal{V} = \mathcal{U}_{1,2}$ and $\mathcal{U}_{1,2}$ is irreducible. To argue that the only invariant subspace properly contained in $\mathcal{U}_{1,2}$ is $\mathcal{U}_2$, let $\mathcal{V} \subset \mathcal{U}_{1,2}$ be an invariant subspace of $\mathfrak{g}_{\mathsf{IMM}}$. From the above argument it follows that $e_x \notin \mathcal{V}$ for every $x \in \mathbf{x}_1$ (otherwise $\mathcal{V} = \mathcal{U}_{1,2}$). This implies $\mathcal{V} \subseteq \mathcal{U}_2$, and from Claim 3.3, we have $\mathcal{V} = \mathcal{U}_2$. □

7.4 Proof of Claims in Section 4

Claim 4.1 (Restated). *For all $i \in [s]$, let $\mathcal{N}_i$ and $\mathcal{N}_i'$ be the null spaces of $g_i(R)$ and $g_i(R')$. Then,*

(1) $\mathbb{F}^n = \mathcal{N}_1 \oplus \mathcal{N}_2 \oplus \cdots \oplus \mathcal{N}_s = \mathcal{N}_1' \oplus \mathcal{N}_2' \oplus \cdots \oplus \mathcal{N}_s'$.
(2) *For all $i \in [s]$, $dim(\mathcal{N}_i) = dim(\mathcal{N}_i') = deg_x(g_i)$.*

Proof. Since $\mathcal{N}_i' = A^{-1}\mathcal{N}_i$ and $A^{-1} \in \mathsf{GL}(n)$, it is sufficient to show $\mathbb{F}^n = \mathcal{N}_1 \oplus \mathcal{N}_2 \oplus \cdots \oplus \mathcal{N}_s$ and $\dim(\mathcal{N}_i) = \deg_x(g_i)$. Further, observe that each subspace $\mathcal{N}_i$ is non-trivial—if $\mathcal{N}_1 = \{0\}$ then for all $\mathbf{v} \in \mathbb{F}^n$, $h(R) \cdot \mathbf{v} = g_1(R)g_2(R) \cdots g_s(R) \cdot \mathbf{v} = 0$ implying $g_2(R) \cdots g_s(R) \cdot \mathbf{v} = 0$. As the characteristic polynomial and the minimal polynomial have the same irreducible factors this gives a contradiction.

To show the sum of $\mathcal{N}_i$'s is a direct sum it is sufficient to show the following: if $\sum_{l=1}^{s} \mathbf{u}_l = 0$ where $\mathbf{u}_l \in \mathcal{N}_l$, then $\mathbf{u}_l = 0$ for $l \in [s]$. Define for $i \in [s]$

$$\hat{g}_i := \prod_{j=1, j \neq i}^{s} g_j(x) = \frac{h(x)}{g_i(x)}. \tag{21}$$

Since $\hat{g}_i(R) \cdot \mathbf{u}_j = 0$ for $j \neq i$,

$$\hat{g}_i(R) \cdot \left(\sum_{l=1}^{s} \mathbf{u}_l \right) = \hat{g}_i(R) \cdot \mathbf{u}_i = 0. \tag{22}$$

As $g_i(x)$ and $\hat{g}_i(x)$ are coprime polynomials, there are $p_i(x), q_i(x) \in \mathbb{F}[x]$ such that

$$\begin{aligned} p_i(x)g_i(x) + q_i(x)\hat{g}_i(x) = 1 &\Rightarrow p_i(R)g_i(R) + q_i(R)\hat{g}_i(R) = I_n \\ &\Rightarrow (p_i(R)g_i(R)) \cdot \mathbf{u}_i + (q_i(R)\hat{g}_i(R)) \cdot \mathbf{u}_i = \mathbf{u}_i. \end{aligned}$$

Both $(p_i(R)g_i(R)) \cdot \mathbf{u}_i = 0$ (as $\mathbf{u}_i \in \mathcal{N}_i$) and $(q_i(R)\hat{g}_i(R)) \cdot \mathbf{u}_i = 0$ (by Equation (22)). Hence, $\mathbf{u}_i = 0$ for all $i \in [s]$.

Let $\tilde{R}$ be the linear the linear map R restricted to the subspace $\mathcal{N}_i$ (this is well defined as $\mathcal{N}_i$ is an invariant subspace of R). Then, $g_i(\tilde{R}) = 0$. Since g_i is irreducible, from Cayley-Hamilton theorem it follows that g_i divides the characteristic polynomial of $\tilde{R}$ implying $\deg_x(g_i) \leq \dim(\mathcal{N}_i)$. As a consequence, we have

$$n = \sum_{i=1}^{s} \deg_x g_i \leq \sum_{i=1}^{s} \dim \mathcal{N}_i \leq \dim \mathbb{F}^n = n. \tag{23}$$

Each inequality is an equality, which proves the claim. □

Claim 4.2 (Restated). *Suppose $g_i(x)$ is an irreducible factor of the characteristic polynomial $h_k(x)$ of R_k (depicted in Figure 4) for some $k \in [d]$. Then the following holds:*

(1) *If $k \in [2, d-1]$ then $\mathcal{N}_i \subseteq \mathcal{U}_k$ (equivalently $\mathcal{N}_i' \subseteq A^{-1}\mathcal{U}_k$).*
(2) *If $k = 1$ then $\mathcal{N}_i \subseteq \mathcal{U}_{1,2}$ (equivalently $\mathcal{N}_i' \subseteq A^{-1}\mathcal{U}_{1,2}$), and if $k = d$ then $\mathcal{N}_i \subseteq \mathcal{U}_{d-1,d}$ (equivalently $\mathcal{N}_i' \subseteq A^{-1}\mathcal{U}_{d-1,d}$).*

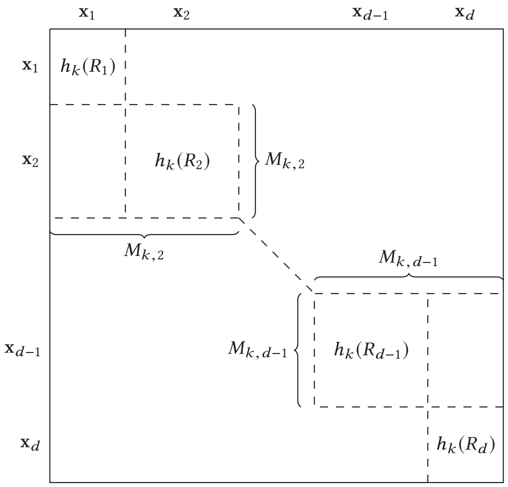

Fig. 12. Matrix $h_k(R)$.

Proof. Figure 12 depicts the matrix $h_k(R)$ and as shown in it, call the submatrix restricted to the rows labelled by variables in $\mathbf{x}_2$ and columns labelled by variables in $\mathbf{x}_1 \uplus \mathbf{x}_2$, $M_{k,2}$; define $M_{k,d-1}$ similarly. Let $\mathbf{v} \in \mathcal{N}_i$. For every $j \in [d]$, let $\mathbf{v}_j$ be the subvector of $\mathbf{v}$ restricted to the rows labelled by variables in $\mathbf{x}_j$, and let $\mathbf{v}_{1,2}$ (respectively, $\mathbf{v}_{d-1,d}$) be the subvector of $\mathbf{v}$ restricted to the rows labelled by variables in $\mathbf{x}_1 \uplus \mathbf{x}_2$ (respectively, $\mathbf{x}_{d-1} \uplus \mathbf{x}_d$). Since $\mathbf{v} \in \mathcal{N}_i$, $g_i(R) \cdot \mathbf{v} = 0$, implying $h_k(R) \cdot \mathbf{v} = 0$. Thus, we have the following set of equations:

$$\begin{aligned} h_k(R_1) \cdot \mathbf{v}_1 &= 0, \\ M_{k,2} \cdot \mathbf{v}_{1,2} &= 0, \\ h_k(R_j) \cdot \mathbf{v}_j &= 0 \ \text{ for } j \in [3, d-2], \\ M_{k,d-1} \cdot \mathbf{v}_{d-1,d} &= 0, \\ h_k(R_d) \cdot \mathbf{v}_d &= 0. \end{aligned} \tag{24}$$

Case a: $k \in [2, d-1]$; since $h_j(x)$ is the characteristic polynomial of R_j, $h_j(R_j) = 0$ implying $h_j(R_j) \cdot \mathbf{v}_j = 0$ for every $j \in [d]$. As $k \neq 1$, $h_k(x)$ and $h_1(x)$ are coprime and from Equation (24) $h_k(R_1) \cdot \mathbf{v}_1 = 0$. Hence, $\mathbf{v}_1 = 0$ and for a similar reason $\mathbf{v}_d = 0$ as $k \neq d$. Thus, in Equation (24), we have

$$\begin{aligned} M_{k,2} \cdot \mathbf{v}_{1,2} = h_k(R_2) \cdot \mathbf{v}_2 &= 0, \\ M_{k,d-1} \cdot \mathbf{v}_{d-1,d} = h_k(R_{d-1}) \cdot \mathbf{v}_{d-1} &= 0. \end{aligned}$$

Therefore, for every $j \in [d]$, $h_k(R_j) \cdot \mathbf{v}_j = 0$. If $j \neq k$, then $h_j(x)$ and $h_k(x)$ are coprime, thus from $h_j(R_j) \cdot \mathbf{v}_j = 0$, we infer $\mathbf{v}_j = 0$, and hence $\mathbf{v} \in \mathcal{U}_k$.

Case b: $k = 1$ or $k = d$; let $k = 1$, the proof for $k = d$ is similar. Since $h_k(R_d) \cdot \mathbf{v}_d = 0$, $h_d(R_d) \cdot \mathbf{v}_d = 0$, and $h_k(x)$, $h_d(x)$ are coprime, we get $\mathbf{v}_d = 0$. Hence, from Equation (24),

$$M_{k,d-1} \cdot \mathbf{v}_{d-1,d} = h_k(R_{d-1}) \cdot \mathbf{v}_{d-1} = 0.$$

Again for $j \in [3, d]$, $h_k(R_j) \cdot \mathbf{v}_j = 0$ and $h_j(x)$, $h_k(x)$ are coprime for every $j \neq k$. Hence, $\mathbf{v}_j = 0$ for $j \in [3, d]$ implying $\mathbf{v} \in \mathcal{U}_{1,2}$. □

7.5 Proof of Lemma and Claim in Section 5

Lemma 5.2 (Restated). *If $f = X_1 \cdot X_2 \cdots X_d$ and $\mathcal{Y}_1, \mathcal{Y}_{1,2}, \mathcal{Y}_3, \ldots, \mathcal{Y}_{d-2}, \mathcal{Y}_{d-1,d}, \mathcal{Y}_d$ is the output of Algorithm 5 then there is a permutation σ on $[3, d-2]$ such that the following hold:*

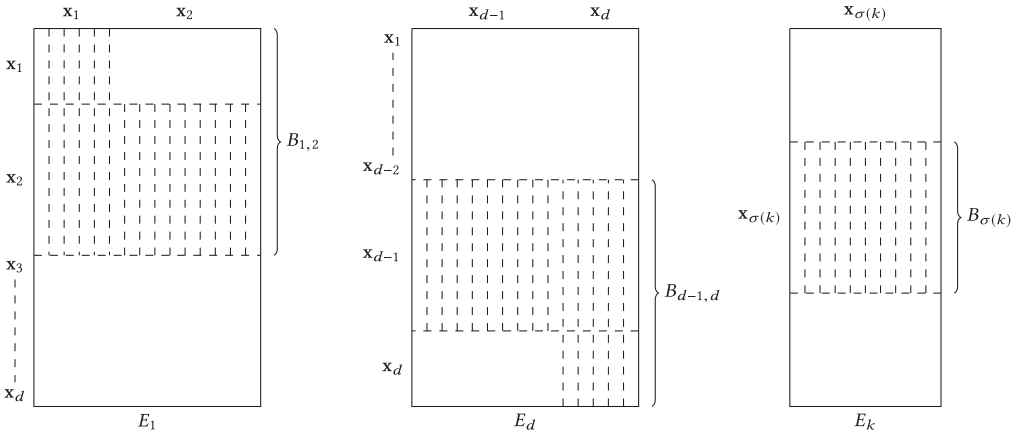

Fig. 13. Matrices E_1, E_d, and E_k.

(1) *For every $k \in [3, d-2]$, $\mathcal{Y}_k = \mathcal{X}_{\sigma(k)}$.*
(2) *Either $\mathcal{Y}_1, \mathcal{Y}_{1,2}$ and $\mathcal{Y}_d, \mathcal{Y}_{d-1,d}$ are $\mathcal{X}_1, \mathcal{X}_{1,2}$ and $\mathcal{X}_d, \mathcal{X}_{d-1,d}$, respectively, or $\mathcal{Y}_1, \mathcal{Y}_{1,2}$ and $\mathcal{Y}_d, \mathcal{Y}_{d-1,d}$ are $\mathcal{X}_d, \mathcal{X}_{d-1,d}$ and $\mathcal{X}_1, \mathcal{X}_{1,2}$, respectively.*

Proof. Assume $\mathcal{V}_1$ and $\mathcal{V}_d$ are the spaces $A^{-1}\mathcal{U}_{1,2}$ and $A^{-1}\mathcal{U}_{d-1,d}$, respectively. In this case, we will show $\mathcal{Y}_1, \mathcal{Y}_{1,2}$ and $\mathcal{Y}_d, \mathcal{Y}_{d-1,d}$ are $\mathcal{X}_1, \mathcal{X}_{1,2}$ and $\mathcal{X}_d, \mathcal{X}_{d-1,d}$, respectively.[39] Hence, $u_1 = w_1 + w_1 w_2$, $u_2 = w_1 w_2$, $u_{d-1} = w_{d-2} w_{d-1}$ and $u_d = w_{d-1} + w_{d-2} w_{d-1}$. From the order of the columns in V_1 and V_d, we have $V_1 = A^{-1}E_1$ and $V_d = A^{-1}E_d$, where E_1 and E_d are $n \times u_1$ and $n \times u_d$ matrices, respectively, and they look as shown in Figure 13. The rows of E_1 and E_d are labelled by n variables in $\mathbf{x}_1$ to $\mathbf{x}_d$, whereas the columns of E_1 are labelled by variables in $\mathbf{x}_1$ and $\mathbf{x}_2$ and the columns of E_d are labelled by variables in $\mathbf{x}_{d-1}$ and $\mathbf{x}_d$. Moreover, the nonzero entries in these matrices are restricted to the shaded region in Figure 13.

For $k \in [3, d-2]$, $\mathcal{V}_k = A^{-1}\mathcal{U}_{\sigma(k)}$, where σ is a permutation on $[3, d-2]$. Hence, $u_k = w_{\sigma(k)-1} w_{\sigma(k)}$ and $V_k = A^{-1}E_k$, where E_k is a $n \times u_k$ matrix and looks as shown in Figure 13. Again the rows of E_k are labelled by the variables $\mathbf{x}_1$ to $\mathbf{x}_d$, whereas the columns of E_k are labelled by variables in $\mathbf{x}_{\sigma(k)}$. The nonzero entries in E_k are restricted to the shaded region in Figure 13 whose rows are labelled by variables in $\mathbf{x}_{\sigma(k)}$. Let E be the concatenation of these matrices, $E = [E_1 | E_3 | E_4 | \ldots | E_{d-2} | E_d]$. The rows of E are labelled by $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_d$ as usual , but now the columns are labelled by $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_{\sigma(3)}, \ldots, \mathbf{x}_{\sigma(d-2)}, \mathbf{x}_{d-1}, \mathbf{x}_d$ in order as shown in Figure 14. Then $V = A^{-1}E$ implying $V^{-1} = E^{-1}A$. Owing to the structure of E, E^{-1} looks as shown in Figure 14. The rows of E^{-1} are labelled by $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_{\sigma(3)}, \ldots, \mathbf{x}_{\sigma(d-2)}, \mathbf{x}_{d-1}, \mathbf{x}_d$ in order, whereas the columns are labelled by the usual ordering $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_d$. The submatrix of E^{-1} restricted to the rows and columns labelled by the variables in $\mathbf{x}_1$ and $\mathbf{x}_2$ is $B_{1,2}^{-1}$ and that labelled by the variables in $\mathbf{x}_{d-1}$ and $\mathbf{x}_d$ is $B_{d-1,d}^{-1}$. For $k \in [3, d-2]$ the submatrix restricted to the rows and columns labelled by $\mathbf{x}_{\sigma(k)}$ is $B_{\sigma(k)}^{-1}$. We infer the following facts:

(I) The space spanned by the first $u_1 - u_2$ (that is w_1) rows of V^{-1} is equal to the space spanned by the first w_1 rows of A, the latter space is $\mathcal{X}_1$.

[39] If $\mathcal{V}_1$ and $\mathcal{V}_d$ are the spaces $A^{-1}\mathcal{U}_{d-1,d}$ and $A^{-1}\mathcal{U}_{1,2}$, respectively, then $\mathcal{Y}_1$, $\mathcal{Y}_{1,2}$ and $\mathcal{Y}_d$, $\mathcal{Y}_{d-1,d}$ are $\mathcal{X}_d$, $\mathcal{X}_{d-1,d}$ and $\mathcal{X}_1$, $\mathcal{X}_{1,2}$, respectively—the proof of this case is similar.

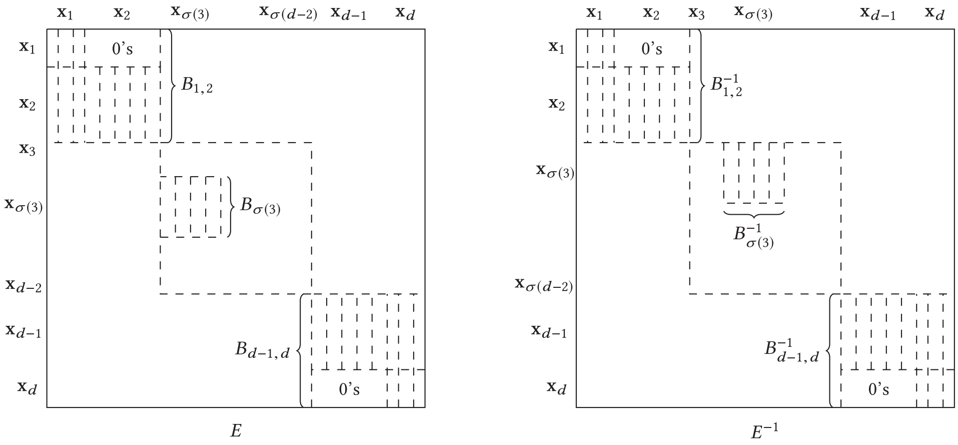

Fig. 14. Matrices E and E^{-1}.

(II) The space spanned by the first u_1 (that is $w_1 + w_1w_2$) rows of V^{-1} is equal to the space spanned by the first $w_1 + w_1w_2$ rows of A, the latter space is $\mathcal{X}_{1,2}$.

(III) The space spanned by the last u_d (that is $w_{d-1} + w_{d-2}w_{d-1}$) rows of V^{-1} is equal to the space spanned by the last $w_{d-1} + w_{d-2}w_{d-1}$ rows of A, the latter space is $\mathcal{X}_{d-1,d}$.

(IV) The space spanned by the last $u_d - u_{d-1}$ (that is w_{d-1}) rows of V^{-1} is equal to the space spanned by the last w_{d-1} rows of A, the latter space is $\mathcal{X}_d$.

(V) For $k \in [3, d-2]$ the space spanned by the rows of V^{-1} that are numbered by $t_{k-1} + 1$ to $t_{k-1} + u_k$ is equal to the space spanned by the rows of A labelled by $\mathbf{x}_{\sigma(k)}$, the latter space is $\mathcal{X}_{\sigma(k)}$. □

Claim 5.1 (Restated). *There is a randomized polynomial time algorithm that takes input the bases of the layer spaces $\mathcal{Y}_1, \mathcal{Y}_{1,2}, \mathcal{Y}_3, \ldots, \mathcal{Y}_{d-2}, \mathcal{Y}_{d-1,d}, \mathcal{Y}_d$ and with probability at least $1 - \frac{1}{\text{poly}(n)}$ reorders these layer spaces and outputs a width vector $\mathbf{w}'$ such that the reordered sequence of spaces and $\mathbf{w}'$ are:*

(1) *either $\mathcal{X}_1, \mathcal{X}_{1,2}, \mathcal{X}_3, \ldots, \mathcal{X}_{d-2}, \mathcal{X}_{d-1,d}, \mathcal{X}_d$ and $(w_1, w_2, \ldots, w_{d-1})$, respectively,*
(2) *or $\mathcal{X}_d, \mathcal{X}_{d-1,d}, \mathcal{X}_{d-2}, \ldots, \mathcal{X}_3, \mathcal{X}_{1,2}, \mathcal{X}_1$ and $(w_d, w_{d-1}, \ldots, w_1)$, respectively.*

Proof. The algorithm employs evaluation dimension to uncover the permutation σ. Assume that $\mathcal{Y}_1, \mathcal{Y}_{1,2}, \mathcal{Y}_3, \ldots, \mathcal{Y}_{d-2}, \mathcal{Y}_{d-1,d}, \mathcal{Y}_d$ are the spaces $\mathcal{X}_1, \mathcal{X}_{1,2}, \mathcal{X}_{\sigma(3)}, \ldots, \mathcal{X}_{\sigma(d-2)}, \mathcal{X}_{d-1,d}, \mathcal{X}_d$, respectively.[40] In this case, the algorithm reorders the spaces to a sequence $\mathcal{X}_1, \mathcal{X}_{1,2}, \mathcal{X}_3, \ldots, \mathcal{X}_{d-2}, \mathcal{X}_{d-1,d}, \mathcal{X}_d$ and outputs $\mathbf{w}' = \mathbf{w}$. For every $k \in [3, d-2]$, let $\mathbf{z}_k$ be a set of $\dim(\mathcal{Y}_k)$ many variables. Let $\mathbf{z}_1$ (similarly, $\mathbf{z}_d$) be a set of $\dim(\mathcal{Y}_1)$ (similarly, $\dim(\mathcal{Y}_d)$) variables, and let $\mathbf{z}_2$ (similarly, $\mathbf{z}_{d-1}$) be a set of $\dim(\mathcal{Y}_{1,2}) - \dim(\mathcal{Y}_1)$ (similarly, $\dim(\mathcal{Y}_{d-1,d}) - \dim(\mathcal{Y}_d)$) variables. Finally, let $\mathbf{z} = \mathbf{z}_1 \uplus \ldots \uplus \mathbf{z}_d$ be the set of these n fresh variables.

Compute a linear map μ that maps $\mathbf{x}$ variables to linear forms in $\mathbf{z}$ variables such that the following conditions are satisfied:

(a) For every $k \in [3, d-2]$, the linear forms corresponding[41] to the basis vectors of $\mathcal{Y}_k$ map to distinct variables in $\mathbf{z}_k$.

[40] The proof of the other case is similar.
[41] Recall, linear forms in $\mathbf{x}$ variables and vectors in $\mathbb{F}^n$ are naturally identified with each other.

(b) The linear forms corresponding to the basis vectors in $\mathcal{Y}_1$ (similarly, $\mathcal{Y}_d$) map to distinct variables in $\mathbf{z}_1$ (similarly, $\mathbf{z}_d$).

(3) The linear forms corresponding to the basis vectors in $\mathcal{Y}_{1,2}$ (similarly, $\mathcal{Y}_{d-1,d}$) map to distinct variables in $\mathbf{z}_1 \uplus \mathbf{z}_2$ (similarly, $\mathbf{z}_{d-1} \uplus \mathbf{z}_d$).

Conditions (b) and (c) can be simultaneously satisfied as the basis of $\mathcal{Y}_1$ (similarly, $\mathcal{Y}_d$) is contained in the basis of $\mathcal{Y}_{1,2}$ (similarly, $\mathcal{Y}_{d-1,d}$) by their very constructions in Algorithm 5. As $f = \mathsf{IMM}_{\mathbf{w},d}(A\mathbf{x})$, the map μ takes f to a polynomial $h(\mathbf{z})$ that is computed by a full rank ABP A' of width $\mathbf{w}$ and length d such that the sets of variables appearing in the d layers of A' from left to right are $\mathbf{z}_1, \mathbf{z}_1 \uplus \mathbf{z}_2, \mathbf{z}_{\sigma^{-1}(3)}, \ldots, \mathbf{z}_{\sigma^{-1}(d-2)}, \mathbf{z}_{d-1} \uplus \mathbf{z}_d, \mathbf{z}_d$ in order.

The following observation, the proof of which is given later, helps find σ^{-1} incrementally from blackbox access to $h(\mathbf{z})$. Let $\mathbf{y}_2 = \mathbf{z}_1 \uplus \mathbf{z}_2$ and $\mathbf{y}_j = \mathbf{z}_1 \uplus \mathbf{z}_2 \uplus \mathbf{z}_{\sigma^{-1}(3)} \uplus \cdots \uplus \mathbf{z}_{\sigma^{-1}(j)}$, for $j \in [3, d-2]$.

Observation 7.5. *For every $j \in [2, d-3]$ and $k \in [3, d-2]$ such that $\mathbf{z}_k \not\subset \mathbf{y}_j$,*

(1) Evaldim$_{\mathbf{y}_j \uplus \mathbf{z}_k}(h) < |\mathbf{z}_k|$, if $k = \sigma^{-1}(j+1)$, and

(2) Evaldim$_{\mathbf{y}_j \uplus \mathbf{z}_k}(h) > |\mathbf{z}_k|$, if $k \neq \sigma^{-1}(j+1)$.

The proof of the observation also includes an efficient randomized procedure to compute $\text{Evaldim}_{\mathbf{y}_j \uplus \mathbf{z}_k}(h)$.

Finally, the algorithm outputs the reordered layer spaces $\mathcal{Y}_1, \mathcal{Y}_{1,2}, \mathcal{Y}_{\sigma^{-1}(3)}, \ldots, \mathcal{Y}_{\sigma^{-1}(d-2)}, \mathcal{Y}_{d-1,d}, \mathcal{Y}_d$, which is the ordered sequence of spaces $\mathcal{X}_1, \mathcal{X}_{1,2}, \mathcal{X}_3, \ldots, \mathcal{X}_{d-2}, \mathcal{X}_{d-1,d}, \mathcal{X}_d$. The width vector $\mathbf{w}'$ can be readily calculated now by inspecting the dimensions:

$$\begin{aligned}
w_1' &= \dim(\mathcal{X}_1) = w_1,\\
w_2' &= \frac{\dim(\mathcal{X}_{1,2}) - w_1}{w_1} = w_2,\\
w_k' &= \frac{\dim(\mathcal{X}_k)}{w_{k-1}} = w_k, \quad \text{for } k \in [3, d-2],\\
w_d' &= \dim(\mathcal{X}_d) = w_d, \quad \text{and}\\
w_{d-1}' &= \frac{\dim(\mathcal{X}_{d-1,d}) - w_d}{w_d} = w_{d-1}.
\end{aligned}$$

This gives $\mathbf{w}' = \mathbf{w}$.

Proof of Observation 7.5. Let $Z_1 \cdot Z_2 \cdots Z_d$ be equal to A', the full rank ABP of width $\mathbf{w} = (w_1, w_2, \ldots, w_{d-1})$ computing h, where the linear forms in Z_i are in $\mathbf{z}_{\sigma^{-1}(i)}$ variables for $i \in [3, d-2]$, the linear forms in Z_1, Z_d are in variables $\mathbf{z}_1, \mathbf{z}_d$, respectively, and the linear forms in Z_2, Z_{d-1} are in $\mathbf{z}_1 \uplus \mathbf{z}_2, \mathbf{z}_{d-1} \uplus \mathbf{z}_d$ variables, respectively.

Case 1: Suppose $k = \sigma^{-1}(j+1)$, implying $|\mathbf{z}_k| = w_j w_{j+1}$. Let $G = Z_{j+2} \cdot Z_{j+3} \cdots Z_d$ and the tth entry of G be g_t for $t \in [w_{j+1}]$. As the linear forms in $Z_1, Z_2, \ldots, Z_{j+1}$ are $\mathbb{F}$-linearly independent, for every $t \in [w_{j+1}]$ there is a partial evaluation of h at $\mathbf{y}_j \uplus \mathbf{z}_k$ variables that makes h equal to g_t. Also, every partial evaluation of h at $\mathbf{y}_j \uplus \mathbf{z}_k$ variables can be expressed as an $\mathbb{F}$-linear combination of $g_1, g_2, \ldots, g_{w_{j+1}}$. Hence, from Claim 5.2 it follows, $\text{Evaldim}_{\mathbf{y}_j \uplus \mathbf{z}_k}(h) = w_{j+1} < |\mathbf{z}_k|$.

Case 2: Suppose $k \neq \sigma^{-1}(j+1)$. The variables $\mathbf{z}_k$ appear in the matrix $Z_{\sigma(k)}$, so $|\mathbf{z}_k| = w_{\sigma(k)-1} w_{\sigma(k)}$. Let $G = Z_{\sigma(k)+1} \cdot Z_{\sigma(k)+2} \cdots Z_d$ and the tth entry of G be g_t for $t \in [w_{\sigma(k)}]$. Further, let $P = (p_{lm})_{l \in [w_j], m \in [w_{\sigma(k)-1}]}$ be equal to $Z_{j+1} \cdot Z_{j+2} \cdots Z_{\sigma(k)-1}$. As the linear forms in $Z_1, Z_2, \ldots, Z_j$ and $Z_{\sigma(k)}$ are $\mathbb{F}$-linearly independent, there is a partial evaluation of h at the $\mathbf{y}_j \uplus \mathbf{z}_k$ variables that makes h equal to $p_{lm} g_t$ for $l \in [w_j], m \in [w_{\sigma(k)-1}]$ and $t \in [w_{\sigma(k)}]$. By Claim 5.2, $\{g_t | t \in [w_{\sigma(k)}]\}$ are $\mathbb{F}$-linearly independent; using a proof similar to that of Claim 5.2, we can show that the polynomials $\{p_{lm} | l \in [w_j], m \in [w_{\sigma(k)-1}]\}$ are also $\mathbb{F}$-linearly independent. This implies the set of polynomials $\{p_{lm}g_t | l \in [w_j], m \in [w_{\sigma(k)-1}]$ and $t \in [w_{\sigma(k)}]\}$ are $\mathbb{F}$-linearly independent, as p_{lm} and g_t are on disjoint sets of variables. Since every partial evaluation of h at $\mathbf{y}_j \uplus \mathbf{z}_k$ variables can be expressed as an $\mathbb{F}$-linear combination of the set of polynomials $\{p_{lm}g_t | l \in [w_j], m \in [w_{\sigma(k)-1}]$ and $t \in [w_{\sigma(k)}]\}$, $\text{Evaldim}_{\mathbf{y}_j \uplus \mathbf{z}_k}(h) = w_j w_{\sigma(k)-1} w_{\sigma(k)} = w_j \cdot |\mathbf{z}_k| > |\mathbf{z}_k|$.

A randomized procedure to compute $\textit{Evaldim}_{\mathbf{y}_j \uplus \mathbf{z}_k}(h)$. Choose evaluation points $\mathbf{a}_1, \ldots, \mathbf{a}_{n^2}$ for the variables $\mathbf{y}_j \uplus \mathbf{z}_k$ independently and uniformly at random from a set $S^{|\mathbf{y}_j \uplus \mathbf{z}_k|} \subset \mathbb{F}^{|\mathbf{y}_j \uplus \mathbf{z}_k|}$ with $|S| = \text{poly}(n)$. Output the dimension of the $\mathbb{F}$-linear space spanned by the polynomials $h(\mathbf{a}_1), \ldots, h(\mathbf{a}_{n^2})$ using Claim 2.2.

We argue that the above procedure outputs $\text{Evaldim}_{\mathbf{y}_j \uplus \mathbf{z}_k}(h)$ with probability at least $1 - \frac{1}{\text{poly}(n)}$. Let $\text{Evaldim}_{\mathbf{y}_j \uplus \mathbf{z}_k}(h) = e$. Observe that in both Case 1 and 2, $e \le n^2$. Also, in both the cases h can be expressed as

$$h = \sum_{i \in [e]} f_i(\mathbf{y}_j \uplus \mathbf{z}_k) \cdot q_i, \tag{25}$$

where f_i and q_i are variable disjoint. The polynomials $q_1, \ldots, q_e$ are the polynomials $g_1, \ldots, g_{w_{j+1}}$ in Case 1; they are the polynomials $\{p_{lm}g_t | l \in [w_j], m \in [w_{\sigma(k)-1}]$ and $t \in [w_{\sigma(k)}]\}$ in Case 2. Just as we argue that $q_1, \ldots, q_e$ are $\mathbb{F}$-linearly independent, we can show that $f_1, \ldots, f_e$ are also $\mathbb{F}$-linearly independent. So, by Claim 2.2 the rank of the matrix $M = (f_c(\mathbf{a}_r))_{r,c \in [e]}$ is e with high probability. This implies the polynomials $h(\mathbf{a}_1), \ldots, h(\mathbf{a}_e)$ are $\mathbb{F}$-linearly independent also with high probability. The correctness of the procedure follows from the observation that the dimension of the $\mathbb{F}$-linear space spanned by $h(\mathbf{a}_1), \ldots, h(\mathbf{a}_{n^2})$ is upper bounded by e (from Equation (25)). □

Observation 5.1 (Restated). *If h is computable by a full rank almost set-multilinear ABP of width $\mathbf{w}$ then there is a full rank almost set-multilinear ABP of width $\mathbf{w}$ in canonical form computing h.*

Proof. Suppose $X_1 \cdot X_2 \cdots X_d$ is a full rank almost set-multilinear ABP of width $\mathbf{w} = (w_1, w_2, \ldots, w_{d-1})$ computing h. Let $X_1' = (x_1^{(1)}\ x_2^{(1)}\ \ldots\ x_{w_1}^{(1)})$ and $X_d' = (x_1^{(d)}\ x_2^{(d)}\ \ldots\ x_{w_{d-1}}^{(d)})$. We show there are matrices X_2' and X_{d-1}' satisfying conditions (1b) and (2b), respectively, of canonical form (defined in Section 2.4) such that $h = X_1' \cdot X_2' \cdot X_3 \cdots X_{d-2} \cdot X_{d-1}' \cdot X_d'$. We prove the existence of $X_2' = (l_{ij}')_{i \in [w_1], j \in [w_2]}$; the proof for X_{d-1}' is similar. It is sufficient to show that there is such an X_2' satisfying $X_1 \cdot X_2 = X_1' \cdot X_2'$. Denote the jth entry of the $1 \times w_2$ matrix $X_1 \cdot X_2$ as $X_1 \cdot X_2(j)$. Similarly $X_1' \cdot X_2'(j)$ represents the jth entry of $X_1' \cdot X_2'$. Let g_i be the sum of all monomials in $X_1 \cdot X_2(j)$ of the following types: $x_i^{(1)} x_k^{(1)}$ for $k \in [i, w_1]$, and $x_i^{(1)} x_{pq}^{(2)}$ for $p \in [w_1], q \in [w_2]$. Clearly,

$$X_1 \cdot X_2(j) = g_1 + g_2 + \cdots + g_{w_1}.$$

If $l_{ij}' \overset{\text{def}}{=} g_i / x_i^{(1)}$, then

$$X_1 \cdot X_2(j) = x_1^{(1)} l_{1j}' + x_2^{(1)} l_{2j}' + \cdots + x_{w_1}^{(1)} l_{w_1 j}'.$$

Since l_{ij}' is the (i,j)th entry of X_2', we infer $X_1 \cdot X_2(j) = X_1' \cdot X_2'(j)$. By definition, $x_k^{(1)}$ does not appear in l_{ij}' for $k < i$, and thus condition (1b) is satisfied by X_2'. □

Observation 5.2 (Restated). *Let $X_1 \cdot X_2 \cdots X_d$ be a full rank almost set-multilinear ABP, and $C_k = X_k \cdots X_d$ for $k \in [2, d]$. Let the lth entry of C_k be h_{kl} for $l \in [w_{k-1}]$. Then the polynomials $\{h_{k1}, h_{k2}, \cdots, h_{kw_{k-1}}\}$ are $\mathbb{F}$-linearly independent.*

Proof. Suppose $\sum_{p=1}^{w_{k-1}} \alpha_p \cdot h_{kp} = 0$ such that $\alpha_p \in \mathbb{F}$ for $p \in [w_{k-1}]$, and not all $\alpha_p = 0$. Assume without loss of generality $\alpha_1 \neq 0$. Since the linear forms in $X_k, \ldots, X_d$ are $\mathbb{F}$-linearly independent, there is an evaluation of the variables in $\mathbf{x}_k \uplus \cdots \uplus \mathbf{x}_d$ to field constants such that $h_{k1} = 1$ and every other $h_{kp} = 0$ under this evaluation. This implies $\alpha_1 = 0$, contradicting our assumption. □

7.6 Proof of Observation in Section 6

Observation 6.2 (Restated). *There are matrices $A_1, \ldots, A_{d-1}$ with $A_k \in \mathsf{GL}(w_k)$ for every $k \in [d-1]$, such that $X_1 = Q_1 \cdot A_1$, $X_2(\mathbf{x}_2) = A_1^{-1} \cdot Q_2 \cdot A_2$, $X_{d-1}(\mathbf{x}_{d-1}) = A_{d-2}^{-1} \cdot Q_{d-1} \cdot A_{d-1}$, $X_d = A_{d-1}^{-1} \cdot Q_d$, and $X_k = A_{k-1}^{-1} \cdot Q_k \cdot A_k$ for $k \in [3, d-2]$.*

Proof. To simplify notations, we write $X_2(\mathbf{x}_2)$, $X_{d-1}(\mathbf{x}_{d-1})$ as X_2, X_{d-1}, respectively. We have

$$X_1 \cdot X_2 \cdots X_{d-1} \cdot X_d = Q_1 \cdot Q_2 \cdots Q_{d-1} \cdot Q_d = \mathsf{IMM},$$

where the dimensions of the matrices X_k and Q_k are the same, and the set of variables appearing in both X_k and Q_k is $\mathbf{x}_k$, for every $k \in [d]$. Since the linear forms in X_1 are $\mathbb{F}$-linearly independent, there is an $A_1 \in \mathsf{GL}(w_1)$ such that $X_1 = Q_1 \cdot A_1$, implying

$$Q_1 \cdot [A_1 \cdot X_2 \cdots X_{d-1} \cdot X_d - Q_2 \cdots Q_{d-1} \cdot Q_d] = 0$$
$$\Rightarrow X_2 \cdots X_{d-1} \cdot X_d = A_1^{-1} \cdot Q_2 \cdots Q_{d-1} \cdot Q_d,$$

as the formal variable entries of Q_1 do not appear in the matrices X_k, Q_k for $k \in [2, d]$. The rest of the proof proceeds inductively: Suppose for some $k \in [2, d-1]$,

$$X_k \cdots X_{d-1} \cdot X_d = A_{k-1}^{-1} \cdot Q_k \cdots Q_{d-1} \cdot Q_d, \quad \text{where } A_{k-1} \in \mathsf{GL}(w_{k-1}).$$

Let $p_k = \sum_{i=k+1}^{d} |\mathbf{x}_i|$. Since the linear forms in $X_{k+1}, \ldots, X_{d-1}, X_d$ are $\mathbb{F}$-linearly independent, for every $l \in [w_k]$ there is a point $\mathbf{a}_l \in \mathbb{F}^{p_k}$ such that the $w_k \times 1$ matrix $X_{k+1} \cdots X_{d-1} \cdot X_d$ evaluated at $\mathbf{a}_l$ has 1 at the lth position and all its other entries are zero. Let A_k be the $w_k \times w_k$ matrix such that the lth column of A_k is equal to $Q_{k+1} \cdots Q_{d-1} \cdot Q_d$ evaluated at $\mathbf{a}_l$. Then, $X_k = A_{k-1}^{-1} \cdot Q_k \cdot A_k$. As the linear forms in X_k and Q_k are $\mathbb{F}$-linearly independent, it must be that $A_k \in \mathsf{GL}(w_k)$. Putting this expression for X_k in the equation above and arguing as before, we get a similar equation with k replaced by $k+1$. The proof then follows by induction. □

ACKNOWLEDGMENT

We would like to thank Rohit Gurjar for some initial discussion on this work.

REFERENCES

[1] Manindra Agrawal. 2005. Proving lower bounds via pseudo-random generators. In *Proceedings of the 25th International Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS'05)*. Springer, Berlin, 92–105.

[2] Manindra Agrawal, Rohit Gurjar, Arpita Korwar, and Nitin Saxena. 2015. Hitting-sets for ROABP and sum of set-multilinear circuits. *SIAM J. Comput.* 44, 3 (2015), 669–697.

[3] Manindra Agrawal and Nitin Saxena. 2006. Equivalence of F-algebras and cubic forms. In *Proceedings of the 23rd Annual Symposium on Theoretical Aspects of Computer Science (STACS'06)*. Springer, Berlin, 115–126.

[4] Dana Angluin. 1988. Queries and concept learning. *Mach. Learn.* 2, 4 (1988), 319–342.

[5] Vikraman Arvind, Partha Mukhopadhyay, and Srikanth Srinivasan. 2008. New results on noncommutative and commutative polynomial identity testing. In *Proceedings of the 23rd Annual IEEE Conference on Computational Complexity (CCC'08)*. 268–279.

[6] Amos Beimel, Francesco Bergadano, Nader H. Bshouty, Eyal Kushilevitz, and Stefano Varricchio. 2000. Learning functions represented as multiplicity automata. *J. ACM* 47, 3 (2000), 506–530.

[7] Elwyn Berlekamp. 1967. Factoring polynomials over finite fields. *Bell Syst. Tech. J.* 46 (1967), 1853–1859.

[8] Lenore Blum, Mike Shub, and Steve Smale. 1988. On a theory of computation over the real numbers; NP completeness, recursive functions and universal machines (extended abstract). In *Proceedings of the 29th Annual Symposium on Foundations of Computer Science*. 387–397.

[9] David G. Cantor and Hans Zassenhaus. 1981. A new algorithm for factoring polynomials over finite fields. *Math. Comp.* 36 (1981), 587–592.

[10] Enrico Carlini. 2006. Reducing the number of variables of a polynomial. In *Algebraic Geometry and Geometric Modelling, Mathematics and Visualization*. Springer, 237–247.

[11] Marco L. Carmosino, Russell Impagliazzo, Valentine Kabanets, and Antonina Kolokolova. 2016. Learning algorithms from natural proofs. In *Proceedings of the 31st Conference on Computational Complexity (CCC'16).* 10:1–10:24.

[12] Zeev Dvir, Rafael Mendes de Oliveira, and Amir Shpilka. 2014. Testing equivalence of polynomials under shifts. In *Proceedings of the 41st International Colloquium on Automata, Languages, and Programming (ICALP'14).* 417–428.

[13] Michael A. Forbes and Amir Shpilka. 2013. Quasipolynomial-time identity testing of non-commutative and read-once oblivious algebraic branching programs. In *54th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2013, 26-29 October, 2013, Berkeley, CA, USA*. 243–252.

[14] Michael A. Forbes, Amir Shpilka, and Ben Lee Volk. 2017. Succinct hitting sets and barriers to proving algebraic circuits lower bounds. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing (STOC'17).* 653–664.

[15] Fulvio Gesmundo. 2016. Gemetric aspects of iterated matrix multiplication. *J. Alg.* 461 (2016), 42–64.

[16] Joshua A. Grochow. 2012. *Symmetry and Equivalence Relations in Classical and Geometric Complexity Theory.* Ph.D. Dissertation. The University of Chicago.

[17] Joshua A. Grochow, Mrinal Kumar, Michael E. Saks, and Shubhangi Saraf. 2017. Towards an algebraic natural proofs barrier via polynomial identity testing. *CoRR* abs/1701.01717 (2017).

[18] Ankit Gupta, Neeraj Kayal, and Satyanarayana V. Lokam. 2011. Efficient reconstruction of random multilinear formulas. In *Proceedings of the IEEE 52nd Annual Symposium on Foundations of Computer Science (FOCS'11).* 778–787.

[19] Ankit Gupta, Neeraj Kayal, and Satyanarayana V. Lokam. 2012. Reconstruction of depth-4 multilinear circuits with top fan-in 2. In *Proceedings of the 44th Symposium on Theory of Computing Conference (STOC'12).* 625–642.

[20] Ankit Gupta, Neeraj Kayal, and Youming Qiao. 2013. Random arithmetic formulas can be reconstructed efficiently. In *Proceedings of the 28th Conference on Computational Complexity (CCC'13).* 1–9.

[21] Johan Håstad. 1989. Tensor rank is NP-complete. In *Proceedings of the 16th International Colloquium on Automata, Languages and Programming (ICALP'89).* 451–460.

[22] Joos Heintz and Claus-Peter Schnorr. 1980. Testing polynomials which are easy to compute (extended abstract). In *Proceedings of the 12th Annual ACM Symposium on Theory of Computing.* 262–272.

[23] Erich Kaltofen and Barry M. Trager. 1988. Computing with polynomials given by black boxes for their evaluation: Greatest common divisors, factorization, separation of numerators and denominators. In *Proceedings of the 29th Annual Symposium on Foundations of Computer Science.* 296–305.

[24] Zohar Shay Karnin and Amir Shpilka. 2009. Reconstruction of generalized depth-3 arithmetic circuits with bounded top fan-in. In *Proceedings of the 24th Annual IEEE Conference on Computational Complexity (CCC'09).* 274–285.

[25] Neeraj Kayal. 2011. Efficient algorithms for some special cases of the polynomial equivalence problem. In *Proceedings of the 22nd Annual ACM-SIAM Symposium on Discrete Algorithms (SODA'11).* 1409–1421.

[26] Neeraj Kayal. 2012. Affine projections of polynomials: Extended abstract. In *Proceedings of the 44th Symposium on Theory of Computing Conference (STOC'12).* 643–662.

[27] Neeraj Kayal. 2012. An exponential lower bound for the sum of powers of bounded degree polynomials. *Proceedings of the Electronic Colloquium on Computational Complexity (ECCC'12)*, vol. 19, 81.

[28] Neeraj Kayal, Vineet Nair, and Chandan Saha. 2018. Average-case linear matrix factorization and reconstruction of low width algebraic branching programs. *Proceedings of the Electronic Colloquium on Computational Complexity (ECCC'18)*, vol. 25, 29. Retrieved from https://eccc.weizmann.ac.il/report/2018/029.

[29] Adam Klivans, Pravesh Kothari, and Igor Carboni Oliveira. 2013. Constructing hard functions using learning algorithms. In *Proceedings of the 28th Conference on Computational Complexity (CCC'13).* 86–97.

[30] Adam Klivans and Amir Shpilka. 2003. Learning arithmetic circuits via partial derivatives. In *Proceedings of the 16th Annual Conference on Computational Learning Theory and 7th Kernel Workshop on Computational Learning Theory and Kernel Machines (COLT/Kernel'03).* 463–476.

[31] Adam Klivans and Daniel A. Spielman. 2001. Randomness efficient identity testing of multivariate polynomials. In *Proceedings on 33rd Annual ACM Symposium on Theory of Computing.* 216–223.

[32] A. K. Lenstra, H. W. jun. Lenstra, and Lászlo Lovász. 1982. Factoring polynomials with rational coefficients. *Math. Ann.* 261 (1982), 515–534.

[33] Meena Mahajan and V. Vinay. 1997. Determinant: Combinatorics, algorithms, and complexity. *Chicago J. Theor. Comput. Sci.* 1997, 5 (1997).

[34] Daniel Minahan and Ilya Volkovich. 2016. Complete derandomization of identity testing and reconstruction of read-once formulas. *Proceedings of the Electronic Colloquium on Computational Complexity (ECCC'16)*, vol. 23, 171.

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

[36] Jacques Patarin. 1996. Hidden fields equations (HFE) and isomorphisms of polynomials (IP): Two new families of asymmetric algorithms. In *Proceedings of the International Conference on the Theory and Application of Cryptographic Techniques: Advances in Cryptology (EUROCRYPT'96).* 33–48.

[37] Alexander A. Razborov and Steven Rudich. 1997. Natural proofs. *J. Comput. Syst. Sci.* 55, 1 (1997), 24–35.

[38] Amir Shpilka. 2007. Interpolation of depth-3 arithmetic circuits with two multiplication gates. In *Proceedings of the 39th Annual ACM Symposium on Theory of Computing*. 284–293.

[39] Amir Shpilka and Ilya Volkovich. 2009. Improved polynomial identity testing for read-once formulas. In *Proceedings of the 12th International Workshop on Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX'09), and 13th International Workshop (RANDOM'09)*. 700–713.

[40] Amir Shpilka and Amir Yehudayoff. 2010. Arithmetic circuits: A survey of recent results and open questions. *Found. Trends Theor. Comput. Sci.* 5, 3-4 (2010), 207–388.

[41] Gaurav Sinha. 2016. Reconstruction of real depth-3 circuits with top fan-in 2. In *Proceedings of the 31st Conference on Computational Complexity (CCC'16)*. 31:1–31:53.

[42] Thomas Thierauf. 1998. The isomorphism problem for read-once branching programs and arithmetic circuits. *Chicago J. Theor. Comput. Sci.* 1998, 1 (1998).

[43] Ilya Volkovich. 2016. A guide to learning arithmetic circuits. In *Proceedings of the 29th Conference on Learning Theory (COLT'16)*. 1540–1561.

Received October 2017; revised June 2018; accepted August 2018