Electronic Colloquium on Computational Complexity, Report No. 109 (2023)

Towards Identity Testing for Sums of Products of Read-Once and Multilinear Bounded-Read Formulae

Pranav Bisht* Nikhil Gupta† Ilya Volkovich‡

July 21, 2023

Abstract

An arithmetic formula is an arithmetic circuit where each gate has fan-out one. An *arithmetic read-once formula* (ROF in short) is an arithmetic formula where each input variable labels at most one leaf. In this paper we present several efficient blackbox *polynomial identity testing* (PIT) algorithms for some classes of polynomials related to read-once formulas. Namely, for polynomial of the form:

- $f = \Phi_1 \cdot \ldots \cdot \Phi_m + \Psi_1 \cdot \ldots \cdot \Psi_r$, where Φ_i, Ψ_j are ROFs for every $i \in [m], j \in [r]$.
- $f = \Phi_1^{e_1} + \Phi_2^{e_2} + \Phi_3^{e_3}$, where each Φ_i is an ROF and e_i-s are arbitrary positive integers.

Earlier, only a whitebox polynomial-time algorithm was known for the former class by Mahajan, Rao and Sreenivasaiah (Algorithmica 2016).

In the same paper, they also posed an open problem to come up with an efficient PIT algorithm for the class of polynomials of the form $f = \Phi_1^{e_1} + \Phi_2^{e_2} + \ldots + \Phi_k^{e_k}$, where each Φ_i is an ROF and k is some constant. Our second result answers this partially by giving a polynomial-time algorithm when $k = 3$. More generally, when each Φ_1, Φ_2, Φ_3 is a multilinear bounded-read formulae, we also give a quasi-polynomial-time blackbox PIT algorithm.

Our main technique relies on the *hardness of representation* approach introduced in Shpilka and Volkovich (Computational Complexity 2015). Specifically, we show hardness of representation for the resultant polynomial of two ROFs in our first result. For our second result, we lift hardness of representation for a sum of three ROFs to sum of their powers.

1 Introduction

Polynomial Identity Testing (PIT) is a central problem in the area of algebraic complexity theory. Given a multivariate polynomial in the form of an arithmetic circuit or a formula

*Computer Science Department, Boston College, Chestnut Hill, MA. Email: pranav.bisht@bc.edu

†Computer Science Department, Boston College, Chestnut Hill, MA. Email: nikhil.gupta.3@bc.edu

‡Computer Science Department, Boston College, Chestnut Hill, MA. Email: ilya.volkovich@bc.edu

ISSN 1433-8092

Φ, one is asked to decide whether Φ computes the identically zero polynomial, i.e. every coefficient in the monomial expansion of the polynomial computed by Φ is zero. There are two types of PIT algorithms - whitebox and blackbox. In the former, one can look inside the circuit or formula while in the latter one can only access evaluations of the formula on field points of their choice.

PIT is one of the important problems in the class BPP (actually in $\mathsf{co\text{-}RP}$) for which a polynomial-time deterministic algorithm is yet to be found. The blackbox randomized algorithm for PIT is extremely simple: given an input polynomial $f \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ of degree d, pick any set $S \subseteq \mathbb{F}$ of size greater than d and evaluate f on a random point sampled from S^n. Declare the polynomial to be an identity if the evaluation is zero and a non-identity otherwise. If the polynomial was actually zero then this algorithm cannot err, otherwise for a non-zero input, this random evaluation can be zero with probability at most $d/|S|$ by the *Schwartz-Zippel-Demillo-Lipton* Lemma [Sch80, Zip79, DL78].

Derandomizing PIT is intimately tied to proving circuit lower bounds. A deterministic sub-exponential-time PIT algorithm yields either a super-polynomial Boolean or a super-polynomial arithmetic circuit lower bound [KI03, HS80, Agr05]. Conversely, a super-polynomial arithmetic circuit lower bound implies a deterministic sub-exponential time PIT algorithm [KI03]. We refer the reader to the excellent survey [KS19] for a detailed exposition on this *hardness vs randomness* trade-off in the algebraic setting. PIT also finds applications in the problems of primality testing [AKS04] and finding perfect matchings in graphs [Lov79].

An arithmetic formula is an arithmetic circuit whose underlying graph is a tree (see Appendix B for formal definitions of arithmetic circuits and formulae). While derandomizing PIT for arithmetic formulae is still open, various interesting restricted classes have found efficient deterministic PIT algorithms. One such natural restriction is to bound the number of times a variable can appear in a formula. An arithmetic read-once formula (ROF in short) is a formula where each variable appears at most once. Shpilka and Volkovich considered the more general class of sum of k ROFs, where k is some constant. They devised a quasi-polynomial-time deterministic algorithm for this class in [SV15], which was later improved to polynomial time in [MV18]. An even more generalized model is a read-k formula, where every variable can appear at most k times, for some constant k. For the class of multilinear read-k formulas, [AvMV15] give a deterministic quasi-polynomial-time PIT algorithm. Note that the class of sum of k ROFs forms a strict subclass of multilinear read-k formulas.

From a single ROF to a sum of ROFs, the next model to consider is sum-of-products of ROFs. More generally, let $\mathcal{C}$ be any natural circuit class like ROFs, then one can define the class $\sum^{[k]} \prod \mathcal{C}$ which consists of polynomials of the form $f = \sum_{i=1}^{k} \prod_{j=1}^{m} f_{ij}$, where each f_{ij} belongs to the class $\mathcal{C}$. One can also define its sub-class $\sum^{[k]} \bigwedge \mathcal{C}$ where each product gate takes the same input. Namely, the class consists of polynomials of the form $f = \sum_{i=1}^{k} f_i^{e_i}$, where each $f_i \in \mathcal{C}$. The work of [RR19] proved lower bounds against the class $\sum^{[k]} \prod$ ROF, when k is constant and the product gates have certain fan-in restriction. For PIT, [MRS16] designed a polynomial-time *whitebox* algorithm for the sub-class $\sum^{[2]} \prod$ ROF. In this work, we give a polynomial-time *blackbox* PIT algorithm for this model. For a constant k, PIT for the class $\sum^{[k]}\bigwedge$ ROF was left as an open problem by [MRS16]. Here, we give a polynomial-time blackbox PIT algorithm for $\sum^{[k]}\bigwedge\mathcal{C}$, when $k = 3$ and $\mathcal{C}$ is the class of read-once or more generally multilinear constant-read formulas.

1.1 Motivations and Related Works

PIT for $\sum^{[k]}\prod\mathcal{C}$. One of the important results in the PIT literature is an efficient deterministic PIT algorithm for the class $\sum^{[k]}\prod\sum$, both in blackbox and whitebox setting, where k is a constant. The first subexponential PIT algorithm was given in [DS07]. The algorithm was in the whitebox setting and had quasi-polynomial run-time. Later, in [KS07], the result was improved by presenting a polynomial-time whitebox algorithm. This was followed by a long line of work [KS08, KS09, SS11, SS12, SS13] which culminated in a polynomial-time blackbox algorithm.

A $\sum^{[k]}\prod\sum$ circuit over a field $\mathbb{F}$ computes polynomials of the kind $\sum_{i=1}^{k}\prod_{j=1}^{d_i}\ell_{i,j}$, where $d_i \in \mathbb{N}$ for every $i \in [k]$ and $\ell_{i,j} \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ are linear polynomials for every $i \in [k], j \in [d_i]$. One natural way to extend this result is to replace the linear polynomials $\ell_{i,j}$'s with more general arithmetic circuits, for which efficient deterministic PIT algorithms are known. Some interesting candidates for such circuits are sparse polynomials (or $\sum\prod$ circuits), ROFs, multilinear bounded-read formulae, etc. Clearly, each of these circuit classes subsume the class of linear polynomials over $\mathbb{F}$. Polynomial-time deterministic blackbox PIT algorithms are known for the classes of sparse polynomials [KS01] and ROFs [MV18], and the class of multilinear bounded-read formulae admits a quasi-polynomial-time blackbox PIT algorithm [AvMV15].

PIT for the class $\sum^{[k]}\prod\sum\prod$ is well-studied: Polynomial-time deterministic blackbox PIT algorithms are known for (syntactically) multilinear $\sum^{[k]}\prod\sum\prod$ circuits [SV18], for *constant-read* $\sum\prod\sum\prod$ circuits [ASSS16, BSV23], and for the class $\sum^{[3]}\prod\sum\prod^{[2]}$ [PS21]; and a quasi-polynomial-time PIT algorithm for $\sum^{[k]}\prod\sum\prod^{[\delta]}$ was given in [DDS21], where δ is also a constant. A deterministic sub-exponential PIT was given for the class $\sum\prod\sum\prod$ in the breakthrough result of [LST21] [1]. Note that there is no top fan-in restriction in their result. However a polynomial-time PIT algorithm continues to be elusive.

In this work, we explore the other route, i.e., in the direction of $\sum^{[k]}\prod$ ROF, which consists of circuits of the kind $\sum_{i=1}^{k}\prod_{j=1}^{d_i}\Phi_{i,j}$, where every $\Phi_{i,j}$ is an ROF over $\mathbb{F}$. The class of ROFs has been studied extensively in the Boolean as well as algebraic worlds. The results in the Boolean world include learning algorithms for Boolean ROFs and some structural properties of Boolean read-once functions [AHK93, KLN$^+$93, BHH95b, BHH95c]. In the arithmetic world, we have the following results for the class of ROFs: A deterministic polynomial-time blackbox PIT algorithm [SV15, MV18], efficient reconstruction algorithms [HH91, BHH95a, BC98, BB98, SV14, MV18], quasi-polynomial-time deterministic blackbox PIT algorithms for the *orbit*[2] of ROFs [MS21, ST21a], a randomized polynomial-time

[1] [LST21] gave a much more general result, which solves PIT for any bounded-depth arithmetic circuit in sub-exponential time.

[2] Let $f \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ of ROFs. Then, the orbit of f is the set of polynomials $f(A\mathbf{x})$, where A varies

reconstruction algorithm for orbits of ROFs [GST23], and characterization of read-once polynomials [Vol16]. The investigation of PIT for $\sum^{[k]}\prod$ ROF might lead to the discovery of new ideas and techniques, which may be helpful in approaching PIT for general arithmetic circuits and formulae.

A polynomial-time deterministic whitebox PIT algorithm for $\sum^{[2]}\prod$ ROF was given in [MRS16]. In this work, we give a polynomial-time deterministic blackbox PIT algorithm for $\sum^{[2]}\prod$ ROF (see Theorem 1). Our algorithm works over any field satisfying some mild condition on its size. The blackbox nature makes the problem quite non-trivial and we introduce a new tool for handling this: **0**-*irreducibility* (for more details, see Definition 2.4 and Observation 2.6). This tool could also be crucial to obtain PIT algorithms for $\sum^{[k]}\prod$ ROF, where $k \geq 3$ is a constant and for other interesting circuit classes.

PIT for $\sum^{[k]}\bigwedge\mathcal{C}$. The circuit class $\sum^{[k]}\bigwedge\mathcal{C}$ consists of arithmetic circuits of the type $\Phi_1^{e_1} + \cdots + \Phi_k^{e_k}$, where $\Phi_1, \ldots, \Phi_k \in \mathcal{C}$ and $e_1, \ldots, e_k \in \mathbb{N}$ are arbitrary. Apart from being a natural and interesting problem in itself, developing efficient PIT algorithm for this class is also important from the viewpoint of PIT for the class $\sum^{[k]}\prod\mathcal{C}$, which subsumes $\sum^{[k]}\bigwedge\mathcal{C}$. Another reason for studying PIT algorithms for $\sum^{[k]}\bigwedge\mathcal{C}$ is that it generalizes the PIT for $\sum^{[k]}\mathcal{C}$, which is comprised of the circuits $\Phi_1 + \cdots + \Phi_k$, where $\Phi_1, \ldots, \Phi_k \in \mathcal{C}$. In this work, we instantiate $\mathcal{C}$ with the classes of ROFs and multilinear bounded-read arithmetic formulae, and take k to be equal to 3 (see Theorems 2 and 3). For the sake of discussion, let $\mathcal{C}_k$ be the class of read-k arithmetic formulae over a field $\mathbb{F}$. Although, $\sum^{[3]}\bigwedge$ ROF is contained in $\sum^{[3]}\bigwedge\mathcal{C}_k$, the reason for mentioning them separately is that in the case of ROFs, we obtain a deterministic polynomial-time blackbox PIT, whereas the time complexity of the blackbox PIT in the case of multilinear bounded-read formulae is quasi-polynomial.

A deterministic polynomial-time PIT algorithm is known for the class $\sum^{[k]}$ ROF [SV15, MV18], which was built over the efficient PIT algorithm for the class of (single) ROFs [MV18]. This is a non-trivial generalization because the class of ROFs is not closed with respect to addition of ROFs. Now, the next level of generalization is to allow arbitrary powers of the ROFs in the sum of k ROFs. This brings us to the class $\sum^{[k]}\bigwedge$ ROF. Obtaining efficient PIT for this class has been mentioned as an open problem in [MRS16]. PIT for the 'bounded-depth variant' of this class has been studied: A polynomial-time blackbox PIT algorithm is given in [ASSS16] for the class of sum of powers of constantly many bounded-depth ROFs[3]. Their algorithm is based on carefully exploiting the *Jacobian* of such circuits and the polynomial running time of their PIT crucially depends on the 'bounded-depth nature' of the underlying formulae. The story of $\sum^{[k]}\bigwedge\mathcal{C}_k$ is also similar. [ASSS16] gives a polynomial-time PIT for the 'bounded-depth' variant of this class. However, it is not clear how to extend their techniques to obtain a polynomial-time PIT for the classes $\sum^{[k]}\bigwedge$ ROF

over all $n \times n$ invertible matrices over $\mathbb{F}$.

[3] In terminology of [ASSS16], such formulae are called as sum of constantly many bounded-depth *occur-once* formulae. In fact, a more general result along with other results was given in [ASSS16] - a polynomial-time deterministic blackbox PIT algorithm for the class of bounded-depth *bounded-occur* arithmetic formulae.

and $\sum^{[k]} \bigwedge \mathcal{C}_k$ in the arbitrary depth setting.

1.2 Our Results

Now we start with our main results. Let $f \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ be an input polynomial of degree d. Our results below hold over *any* field $\mathbb{F}$ satisfying $|\mathbb{F}| > n \cdot d$. Otherwise, we assume to have access to a sufficiently large extension field. We note that the requirement for large enough field size is intrinsically necessary for any *blackbox* PIT algorithm.

Our first result is a blackbox PIT algorithm for $\sum^{[2]} \prod$ ROF. It improves a previous result of [MRS16], which gave a polynomial-time whitebox PIT for the same model.

Theorem 1 (Blackbox PIT for $\sum^{[2]} \prod$ ROF)**.** *Let $f \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ be a polynomial of degree at most d computed as $f = \Phi_1 \cdots \Phi_m + \Psi_1 \cdots \Psi_r$, where $\Phi_1, \ldots, \Phi_m, \Psi_1, \ldots, \Psi_r$ are ROFs. Then there exists a deterministic algorithm that given blackbox access to f decides whether f is identically zero, in time* $\text{poly}(n, d)$.

Remark 1.1. *The parameters m and r used in the above theorem can be arbitrary natural numbers as long as the degree of the polynomial computed by $\Phi_1 \cdot \cdots \Phi_m + \Psi_1 \cdots \Psi_r$ is at most d.*

The proof of this theorem is given in Section 4. It is based on the high-level proof overview given in Section 1.3.

In the following two theorems we give blackbox PIT algorithms for the classes $\sum^{[3]} \bigwedge$ ROF and $\sum^{[3]} \bigwedge \mathcal{C}_k$, where $\mathcal{C}_k$ is the class of multilinear read-k arithmetic formulae (Definition 3.13). Although ROFs are subsumed by multilinear bounded-read formulae, we are stating different results for them since we obtain a polynomial-time PIT for $\sum^{[3]} \bigwedge$ ROF, whereas the runtime for the PIT algorithm for $\sum^{[3]} \bigwedge \mathcal{C}_k$ is quasi-polynomial. An interesting common thread in these results is that the time complexity of the blackbox PIT algorithm for $\sum^{[3]} \bigwedge$ ROF (similarly, $\sum^{[3]} \bigwedge \mathcal{C}_k$) is same as the blackbox PIT algorithm for $\sum^{[3]}$ ROF (respectively, $\sum^{[3]} \mathcal{C}_k$), which is strictly weaker than $\sum^{[3]} \bigwedge$ ROF (respectively, $\sum^{[3]} \bigwedge \mathcal{C}_k$).

Theorem 2 (Blackbox PIT for $\sum^{[3]} \bigwedge$ ROF)**.** *Let $f \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ be a polynomial of degree at most d computed as $f = \Phi_1^{e_1} + \Phi_2^{e_2} + \Phi_3^{e_3}$ where Φ_1, Φ_2, Φ_3 are ROFs and $e_1, e_2, e_3 \in \mathbb{N}$. Then there exists a deterministic algorithm that given blackbox access to f decides whether f is identically zero, in time* $\text{poly}(n, d)$.

Theorem 3 (Blackbox PIT for $\sum^{[3]} \bigwedge \mathcal{C}_k$)**.** *Let $k \in \mathbb{N}$ be a constant and let $f \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ be a polynomial of degree at most d computed as $f = \Phi_1^{e_1} + \Phi_2^{e_2} + \Phi_3^{e_3}$, where Φ_1, Φ_2, Φ_3 are multilinear read-k arithmetic formulae and $e_1, e_2, e_3 \in \mathbb{N}$. Then there exists a deterministic algorithm that given blackbox access to f decides whether f is identically zero, in time* $(n \cdot d)^{O(\log n)}$.

The proofs of these two theorems are given in Subsection 5.3 of Section 5. In fact, we prove a more general result in Section 5 (see Theorem 5.9), which subsumes Theorems 2 and 3. For simplicity of exposition, we give a high-level proof overview of Theorem 2 in Section 1.3.2. The proof overview of Theorem 3 is exactly on the same line as that of Theorem 2.

1.3 Proof Overview and Techniques

In this section, we give the high level overviews of the proofs of Theorems 1, 2, and 3. The underlying theme of these proofs is the *hardness of representation* approach, which was first introduced in [SV15], where PIT algorithms for sums of constantly many ROFs were given. In its initial avatar, hardness of representation was given for sum of constantly many $\mathbf{0}$-*justified* (see Definition 2.1) ROPs, which was the following result: Suppose $\mathbb{F}$ is an arbitrary field, $A_1, \dots, A_k \in \mathbb{F}[x_1, x_2, \dots, x_n]$ are $\mathbf{0}$-justified ROPs and $A \triangleq A_1 + \cdots + A_k \not\equiv 0$. If $n \geq 3k$ then the monomial $x_1 \cdots x_n$ does not divide A. However, it is not difficult to show that this statement is equivalent to the following: Suppose $\mathbb{F}$ is an arbitrary field, $A_1, \dots, A_k \in \mathbb{F}[x_1, x_2, \dots, x_n]$ are $\mathbf{0}$-justified ROPs, and $A \triangleq A_1 + \cdots + A_k$. Either $A \equiv 0$ or if $n \geq 3k$ then for every subset $J \subseteq [n]$ of size $3k$, the monomial $\prod_{j \in J} x_j$ does not divide A. Hardness of representation also sits at the core of the PIT algorithm for the class of multilinear bounded-read arithmetic formulae given in [AvMV15]. In this paper, we work with the second formulation of the hardness of representation approach. See Definition 2.35 and Fact 2.37 in this regard.

There is also an alternate way to view hardness of representation, which is popularly called *low-support concentration* in literature [ASS13, FS13, FSS14, For15, GKST15, GKS16, ST21b]. The idea is to choose a 'nice' point $(a_1, \dots, a_n) \in \mathbb{F}^n$ for an input polynomial $f \in \mathbb{F}[x_1, x_2, \dots, x_n]$ such that the *shifted* polynomial $f(x_1 + a_1, \dots, x_n + a_n)$ has a non-zero monomial of low *support-size* (number of variables appearing in the monomial). For a polynomial with such a low-support monomial, efficient blackbox PIT is known (Fact 2.32). In this work, we shift by a *justifying assignment* or an *irreducibility-preserving assignment* (see Definitions 2.1, 2.4) in order to achieve hardness of representation, which in turn proves existence of a low-support monomial (see Fact 2.37).

1.3.1 Proof Overview of Theorem 1

In Theorem 1, we give a polynomial-time blackbox PIT for $\sum^{[2]} \prod$ ROF, which consists of polynomials of the kind $f = A_1 \cdots A_m + B_1 \cdots B_r$, where every $A_\ell, B_t \in \mathbb{F}[x_1, \dots, x_n]$ are read-once polynomials (ROPs), i.e., the polynomials computed by ROFs (Definition 3.1). We are given blackbox access to such an $f = A_1 \cdots A_m + B_1 \cdots B_r$, where $\deg(f) \leq d$ and we want to determine whether $f \equiv 0$ or not. To accomplish this, we design a *generator* (Definition 2.27), that is a polynomial map $\mathcal{G} = (\mathcal{G}^1, \dots, \mathcal{G}^n) : \mathbb{F}^w \to \mathbb{F}^n$ which preserves non-zeroness, formally $f(x_1, \dots, x_n) \equiv 0$ if and only if $f(\mathcal{G}^1, \dots, \mathcal{G}^n) \equiv 0$, where w is a constant and $\max\{\deg(\mathcal{G}^i) : i \in [n]\} \leq \delta$. Then, $f(\mathcal{G})$ becomes a w-variate polynomial, which has degree at most $d \cdot \delta$. Since w is a constant, it is easy to test the zeroness of $f(\mathcal{G})$

in time $\mathrm{poly}(n, d, \delta)$ (see Fact 2.28 in this regard). The map $\mathcal{G}$ in our case is the generator $G_{n,4}$ given in Definition 2.30, with $w = 8$.

Now, let us see why $G_{n,4}$ is a correct generator for the class $\sum^{[2]} \prod$ ROF. Recall $f = A_1 \cdots A_m + B_1 \cdots B_r$. As every A_ℓ, B_t are ROPs, we can assume without loss of generality that they are irreducible (see Fact 3.2). We now apply the standard trick of *simplifying* the polynomial. Formally, let $g \triangleq \gcd(A_1 \cdots A_m, B_1 \cdots B_r)$ and $f' \triangleq \frac{f}{g}$. Then, we can write $f = g \cdot f'$, where g is called the *simple* part of f. Since g is a product of non-zero ROPs (see Fact 3.2), it follows from a result of [MV18] (see Fact 3.11) and the *multiplicative property* of a generator (see Observation 2.29) that $f(G_{n,4}) \equiv 0$ if and only if $f'(G_{n,4}) \equiv 0$. So, we can assume without loss of generality that $f' = f = A_1 \cdots A_m + B_1 \cdots B_r$. Then, there are two possibilities: Either $f \in \mathbb{F}$ or for every $\ell \in [m], t \in [r]$, A_ℓ and B_t are co-prime. The first case is trivial. Now, we talk about the second case.

Fix $A = A_\ell$ and $B = B_t$ arbitrarily. Since A, B are co-prime, if x is an arbitrary variable of A then the resultant of A and B with respect to x, denoted $\mathrm{Res}_x(A, B)$, is a non-zero polynomial (see Definition 2.12 and Fact 2.13). As A, B are ROPs, they are multilinear, and can be written as $A = A'_1 x + A'_0$ and $B = B'_1 x + B'_0$, where $A'_1, A'_0, B'_1, B'_0 \in \mathbb{F}[\mathbf{x} \setminus \{x\}]$. Then, it follows from Definition 2.12 that

$$\mathrm{Res}_x(A, B) = A'_1 B'_0 - A'_0 B'_1.$$

Since A and B are co-prime, $\mathrm{Res}_x(A, B) \not\equiv 0$. If we have a generator $\mathcal{G}$ that hits this resultant, then we are done as $A(\mathcal{G})$ will be co-prime to every $B(\mathcal{G})$, which certifies that $f(\mathcal{G}) \not\equiv 0$. This approach has been utilized earlier also and is formally stated in Fact 2.34. In order to argue that $f(G_{n,4}) \not\equiv 0$, it suffices to show that $G_{n,3}$ hits the resultant $\mathrm{Res}_x(A, B)$, i.e. $(\mathrm{Res}_x(A, B))(G_{n,3}) \not\equiv 0$.

Now, we argue that $(\mathrm{Res}_x(A, B))(G_{n,3}) \not\equiv 0$. For this, we introduce the notion of *zero-irreducibility*. We say that a polynomial $g \in \mathbb{F}[x_1, \ldots, x_n]$ is $\mathbf{0}$*-irreducible*, if for every proper subset $I \subsetneq [n]$, the restricted polynomial $g|_{\mathbf{x}_I = \mathbf{0}_I}$ is irreducible and $g(\mathbf{0}) \neq 0$ (Definition 2.4). Let us first see why $\mathbf{0}$-irreducible ROPs are interesting in this scenario. Let $\widetilde{A}$ and $\widetilde{B}$ be two $\mathbf{0}$-irreducible ROPs and $x \in \mathrm{var}(\widetilde{A})$. We show that there exists a monomial in $\mathrm{Res}_x(\widetilde{A}, \widetilde{B})$, which has at most two variables (see Corollary 4.3). This along with Fact 2.32 implies that $(\mathrm{Res}_x(\widetilde{A}, \widetilde{B}))(G_{n,2}) \not\equiv 0$. To show that a monomial of support at most two exists in $\mathrm{Res}_x(\widetilde{A}, \widetilde{B})$, we prove a *hardness of representation* theorem for the resultant of two $\mathbf{0}$-irreducible ROPs. In particular, we show that if $\widetilde{A}, \widetilde{B}$ are $\mathbf{0}$-irreducible ROPs then there do not exist three distinct variables x_1, x_2, x_3 such that $x_1 x_2 x_3$ divides $\mathrm{Res}_x(\widetilde{A}, \widetilde{B})$ (see Lemma 4.2). This result is the heart of the proof of Theorem 1.

Now let us see how to transform the original irreducible ROPs A, B to $\mathbf{0}$-irreducible ROPs $\widetilde{A}, \widetilde{B}$. We show that there exists an assignment $\mathbf{a}$ in the image of $G_{n,1}$ such that $\widetilde{A} \triangleq A(\mathbf{x}+\mathbf{a})$ and $\widetilde{B} \triangleq B(\mathbf{x} + \mathbf{a})$ are $\mathbf{0}$-irreducible ROPs. For this, we need a tool called the *commutator* of a polynomial $g \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ (Definition 2.19), denoted $\Delta_{i,j} g$, where $i, j \in [n]$. Since A, B are irreducible multilinear polynomials, all the commutators of A and B are non-zero (Corollary 2.21). We show that if we have an $\mathbf{a} \in \mathbb{F}^n$ such that $A(\mathbf{a}) \neq 0, B(\mathbf{a}) \neq 0$ and for every $i, j \in [n], i \neq j, (\Delta_{i,j}A)(\mathbf{a}) \neq 0, (\Delta_{i,j}B)(\mathbf{a}) \neq 0$, then $A(\mathbf{x}+\mathbf{a})$ and $B(\mathbf{x}+\mathbf{a})$ are $\mathbf{0}$-irreducible ROPs. The nice structure of a commutator of an ROP given in Corollary 3.7 turns out to be extremely helpful in showing that the desired tuple $\mathbf{a}$ is in the image of $G_{n,1}$ (see Claim 3.12).

Once putting the things together, we get that $(\mathrm{Res}_x(A, B))(G_{n,2} + G_{n,1}) \not\equiv 0$. Since $G_{n,3} = G_{n,2} + G_{n,1}$ (see Fact 2.31), we have $\mathrm{Res}_x(A, B)(G_{n,3}) \not\equiv 0$.

1.3.2 Proof Overview of Theorems 2 and 3

In Section 5, we prove a result (see Theorem 5.9) which captures both Theorems 2 and 3. However, for the sake of keeping the discussion simple and yet deliver the main ideas, we restrict ourselves to ROFs. In particular, we give a high-level proof overview of Theorem 2.

We are given blackbox access to a polynomial f computed by a circuit in $\sum^{[3]} \bigwedge$ ROF and we want to determine whether f is zero or not. Then, there exist three ROPs $A, B, R \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ and $e_1, e_2, e_3 \in \mathbb{N}$ such that

$$f = A^{e_1} + B^{e_2} + R^{e_3}.$$

We prove that $f \equiv 0$ if and only if $f(G_{n,10}) \equiv 0$, where $G_{n,10}$ is given in Definition 2.30. The main crux of this result is the hardness of representation for f. We show that if an assignment $\mathbf{a} \in \mathbb{F}^n$ is a common *justifying assignment* (Definition 2.1) of A, B, R then $f(\mathbf{x}+\mathbf{a})$ is either zero or for every set $J \subseteq [n]$ of size 10, $f' \overset{\Delta}{=} f(\mathbf{x}+\mathbf{a})$ is not divisible by the monomial $\prod_{j \in J} x_j$. This hardness of representation then implies existence of a monomial of support-size at most 9 in f' and therefore $G_{n,9}$ hits f' (see Fact 2.37). Formally, $f' \equiv 0$ if and only if $f'(G_{n,9}) \equiv 0$. We know from [SV15] that such an $\mathbf{a}$ is in the image of $G_{n,1}$ (see Fact 3.3). Since $G_{n,10} = G_{n,9} + G_{n,1}$ (see Fact 2.31), we get that $G_{n,10}$ is a generator for f. Now, Fact 2.28 implies that given blackbox access to f, we can test whether f is zero or not in $\mathrm{poly}(n, d)$-time, where d is the degree of f.

The hardness of representation theorem mentioned above crucially uses the fact that f is a sum of powers of three ROFs. The proof proceeds by analyzing various cases originating from the comparison of the parameters e_1, e_2, and e_3 mentioned above. Here, we assume without loss of generality that $e_1 \geq e_2 \geq e_3$. If $e_1 > e_2$ then it is easy to show the required hardness of representation result. A major chunk of the proof is devoted to analyze the case when $e_1 = e_2 = e$. In this part, the following factorization becomes pivotal.

$$A^e - B^e = \prod_{\ell \in [e]} (A - \omega^\ell B),$$

where ω is a primitive e-th root of unity (Definition 2.14). It may seem from here that our result only holds over fields that contain ω. However, it is not the case. We show that it is possible to "massage" e in such a way that a primitive e-th root of unity is always present in the underlying field (or an appropriate extension). Our proof crucially exploits the following two properties of ROFs: 1) the class of ROFs is closed under product of variable disjoint ROFs, 2) the hardness of representation result for the sum of any three $\mathbf{0}$-justified ROFs given in [SV15] (see Fact 3.8).

1.4 Organization

We give a set of useful notations and preliminary results in Section 2. Then, we formally define read-once arithmetic formulae and multilinear bounded-read arithmetic formulae, and give some useful properties of these formulae in Section 3. Section 4 is devoted to the proof of Theorem 1 and the proofs of Theorems 2 and 3 are given in Section 5. In fact, we prove a more general result in Section 5, which subsumes these two theorems. We conclude with some open questions in Section 6.

2 Preliminaries

For a field $\mathbb{F}$, its algebraic closure is denoted as $\overline{\mathbb{F}}$. $\mathbb{N}$ represents the set of natural numbers. We use the $\triangleq$ symbol for defining. For a natural number n, $[n] \triangleq \{1, \ldots, n\}$. Unless otherwise specified, we use the shorthand $\mathbf{x}$ for $\{x_1, \ldots, x_n\}$. We denote the sets of variables by $\mathbf{x}, \mathbf{y}, \mathbf{z}$; polynomials over a field $\mathbb{F}$ by $f, g, h, u, v, A, B, F, R$; elements of $\mathbb{F}$ by α, β, a, b; vectors over $\mathbb{F}$ by $\mathbf{a}, \mathbf{b}$; circuit classes by upper case calligraphic letters like $\mathcal{C}$; and sets by I, J, K, L. For a polynomial $f \in \mathbb{F}[\mathbf{x}]$, we denote a monomial $x_1^{e_1} \cdots x_n^{e_n}$ in f by $\mathbf{x}^{\mathbf{e}}$ and for some $i \in [n]$, $\deg_{x_i}(f)$ denotes the degree of variable x_i in f when it is viewed as a polynomial in x_i over the polynomial ring $\mathbb{F}[\mathbf{x} \setminus \{x_i\}]$. The *individual degree* of f is defined as $\max_{i\in[n]}\{\deg_{x_i}(f)\}$. A polynomial f is called *multilinear* if its individual degree is at most one. We define support of a monomial by $\text{supp}(\mathbf{x}^{\mathbf{e}}) \triangleq \{i \in [n] \mid e_i > 0\}$ and denote support-size by $|\text{supp}(\mathbf{x}^{\mathbf{e}})|$.

We say that two polynomials $f, g \in \mathbb{F}[\mathbf{x}]$ are *similar*, denoted $f \sim g$, if there exists a non-zero $\alpha \in \mathbb{F}$ such that $f = \alpha \cdot g$. For a polynomial $f \in \mathbb{F}[\mathbf{x}]$ and a vector $\mathbf{a} = (a_1, \ldots, a_n) \in \mathbb{F}^n$, the shifted polynomial is $f(\mathbf{x} + \mathbf{a}) \triangleq f(x_1 + a_1, \ldots, x_n + a_n)$. We say that $f \in \mathbb{F}[\mathbf{x}]$ *depends* on x_i if there exist $\mathbf{a} = (a_1, \ldots, a_n) \in \mathbb{F}^n$ and $b \in \mathbb{F}$ such that

$$f(a_1, \ldots, a_{i-1}, a_i, a_{i+1}, \ldots, a_n) \neq f(a_1, \ldots, a_{i-1}, b, a_{i+1}, \ldots, a_n).$$

Further, $\text{var}(f) \triangleq \{i \in [n] : f \text{ depends on } x_i\}$. Let $f \in \mathbb{F}[\mathbf{x}], I \subseteq [n]$, and $\mathbf{a} \in \mathbb{F}^n$. Then, $f|_{\mathbf{x}_I=\mathbf{a}_I}$ is the polynomial obtained by substituting $x_i = a_i$ in f for every $i \in I$. Clearly, $\text{var}(f_{\mathbf{x}_I=\mathbf{a}_I}) \subseteq \text{var}(f) \setminus I$. Observe that this containment can be strict. For example, let $f = x_1x_2 + 1, \mathbf{a} = (0, 0)$, and $I = \{1\}$. Then, $\text{var}(f|_{\mathbf{x}_I=\mathbf{a}_I}) \subsetneq \text{var}(f) \setminus \{1\}$ as after setting $x_1 = 0$ in f, it no longer depends on x_2. We are interested in those assignments where such undesirable losses do not happen. Such assignments, known as *justifying assignments*, have been earlier considered in [HH91, BHH95a, SV15]. Consider the following definition, which has been obtained by adding Property 2 to the definition of a justifying assignment given in [SV15, SV14]. This modification has been made to suit our purpose.

Definition 2.1 (Justifying assignment). *Let $f \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ and $\mathbf{a} \in \mathbb{F}^n$. Then, $\mathbf{a}$ is called a justifying assignment of f (equivalently, f is said to be $\mathbf{a}$-justified) if the following properties are satisfied:*

1. *For every $I \subseteq \text{var}(f), \text{var}(f|_{\mathbf{x}_I=\mathbf{a}_I}) = \text{var}(f) \setminus I$.*

2. $f(\mathbf{a}) \neq 0$.

Remark 2.2. *By convention, the identically zero polynomial is always* $\mathbf{a}$*-justified for every tuple* $\mathbf{a} \in \mathbb{F}^n$.

For example, let $f = x_1x_2 + 1 \in \mathbb{F}[\mathbf{x}]$, where $\mathrm{char}(\mathbb{F}) \neq 2$. Let $\mathbf{a} = (0,0)$, and $\mathbf{b} = (1,1)$. Then, f is $\mathbf{b}$-justified but not $\mathbf{a}$-justified. The following nice property of a justifying assignment is implied by Proposition 2.3 of [SV15].

Fact 2.3. *An assignment* $\mathbf{a} \in \mathbb{F}^n$ *is a justifying assignment of a polynomial* $f \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ *if and only if the condition given in Property 1 of Definition 2.1 holds for every* $I \subseteq \mathrm{var}(f)$ *of size* $|\mathrm{var}(f)| - 1$ *and* $f(\mathbf{a}) \neq 0$.

Recall that a non-constant polynomial $f \in \mathbb{F}[\mathbf{x}]$ is *irreducible* if it can not be written as a product of two non-constant polynomials in $\mathbb{F}[\mathbf{x}]$. Otherwise, f is reducible. By convention, every element of $\mathbb{F}$ is reducible.

Definition 2.4 (Irreducibility preserving assignment)**.** *Let* $f \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ *and* $\mathbf{a} \in \mathbb{F}^n$. *Then,* $\mathbf{a}$ *is called an* irreducibility preserving assignment *of* f *if for every proper subset* $I \subsetneq \mathrm{var}(f)$*, the restricted polynomial* $f|_{\mathbf{x}_I = \mathbf{a}_I}$ *is irreducible and* $f(\mathbf{a}) \neq 0$. *Equivalently, we say that* f *is* $\mathbf{a}$*-irreducible.*

For example, let $f = x_1 + x_2 + x_3$ and $\mathbf{a} = (0,0,1)$. Then, f is $\mathbf{a}$-irreducible over every field. Observe that if $f \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ is $\mathbf{a}$-irreducible for any $\mathbf{a} \in \mathbb{F}^n$ then f is irreducible over $\mathbb{F}$. Claim 2.5 below shows that irreducibility preserving assignments capture justifying assignments of irreducible polynomials. Note that the converse of this claim is not true. For example, let $f = (x_1+1)(x_2+1)+x_3$ and $\mathbf{a} = (0,0,0)$. Then, f is irreducible, $\mathbf{a}$-justified but is not $\mathbf{a}$-irreducible. Thus, for irreducible polynomials, the notion of irreducible preserving assignment is strictly stronger than that of justifying assignment.

Claim 2.5. *Let* $f \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ *and* $\mathbf{a} \in \mathbb{F}^n$. *If* f *is* $\mathbf{a}$*-irreducible then* f *is* $\mathbf{a}$*-justified.*

Proof. Suppose for the sake of contradiction that f is not $\mathbf{a}$-justified. Then, Fact 2.3 implies that either $f(\mathbf{a}) = 0$ or there exists an $I \subseteq \mathrm{var}(f)$, $|I| = |\mathrm{var}(f)| - 1$ such that $\mathrm{var}(f|_{\mathbf{x}_I = \mathbf{a}_I}) \subsetneq \mathrm{var}(f) \setminus I$. If the former case holds then $\mathbf{a}$ can not be an irreducibility preserving assignment of f. In the latter case, note that $f|_{\mathbf{x}_I = \mathbf{a}_I} \in \mathbb{F}$. Since every element of $\mathbb{F}$ is reducible, $f|_{\mathbf{x}_I = \mathbf{a}_I}$ is also reducible. As I is a proper subset of $\mathrm{var}(f)$, we get that f is not $\mathbf{a}$-irreducible, which is a contradiction. Hence, f is $\mathbf{a}$-justified. □

The following easy to prove observation allows us to convert an $\mathbf{a}$-irreducible polynomial to a $\mathbf{0}$-irreducible polynomial by shifting $\mathbf{x}$ with $\mathbf{a}$.

Observation 2.6. *Let* $f \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ *and* $\mathbf{a} \in \mathbb{F}^n$. *Then,* f *is* $\mathbf{a}$*-irreducible if and only if* $f(\mathbf{x} + \mathbf{a})$ *is* $\mathbf{0}$*-irreducible.*

2.1 Basic Mathematical Facts

We start this section with the following useful result by Gauss.

Fact 2.7 (Gauss Lemma)**.** *Let $\mathbb{F}$ be a field, $f \not\equiv 0 \in \mathbb{F}[\mathbf{x}, y]$, and $g \in \mathbb{F}[\mathbf{x}]$. Suppose $f|_{y=g(\mathbf{x})} \equiv 0$. Then, $y - g(\mathbf{x})$ is an irreducible factor of f.*

2.1.1 Ideals

Definition 2.8. *Let $(\mathcal{R}, +, .)$ be an arbitrary ring. A subset $\mathcal{I}$ is called a left ideal of $\mathcal{R}$ if*

1. *$(\mathcal{I}, +)$ is a subgroup of $(\mathcal{R}, +)$,*
2. *For every $a \in \mathcal{R}$ and every $x \in \mathcal{I}$, $a \cdot x \in \mathcal{I}$.*

Similarly a right ideal is defined when the condition $a \cdot x \in \mathcal{I}$ is replaced with $x \cdot a \in \mathcal{I}$. A two sided ideal is a left ideal which is also a right ideal and will simply be called the ideal in this work.

We will work over the ring of polynomials $\mathcal{R} = \mathbb{F}[x_1, x_2, \ldots, x_n]$ and the *monomial ideal* $\mathcal{I}_\ell \triangleq \langle x_1 x_2 \cdots x_\ell \rangle$ for some $\ell \leq n$. The observation below follows from the fact that $\mathbb{F}[x_1, x_2, \ldots, x_n]$ is a unique factorization domain.

Observation 2.9. *Let $g \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ and $e, \ell \in \mathbb{N}$ such that $g^e \in \mathcal{I}_\ell$. Then, $g \in \mathcal{I}_\ell$.*

Consider the following useful definition.

Definition 2.10 (polynomial-hat)**.** *For any polynomial f and $\mathcal{I}_\ell$, we can write f as $f = \tilde{f} + \widehat{f}$, where $\tilde{f} \in \mathcal{I}_\ell$ and $\widehat{f} = f \pmod{\mathcal{I}_\ell}$ is the* unique *polynomial obtained from f after going modulo $\mathcal{I}_\ell$.*

Remark 2.11. *Note that the polynomial $\widehat{f}$ may not be unique for general ideals but here for the monomial ideal $\mathcal{I}_\ell$, we define it uniquely by removing all the monomials in f which are divisible by $x_1 \cdots x_\ell$.*

2.1.2 Resultant

The polynomial ring $\mathbb{F}[x_1, \ldots, x_n]$ is a unique factorization domain (UFD) and therefore the gcd of two polynomials is well defined. One can also define gcd w.r.t. a single variable, say x_i, by viewing the polynomials as univariates in x_i, with coefficients in $\mathbb{F}[\mathbf{x} \setminus \{x_i\}]$. Then, $\gcd_{x_i}(f, g)$ is well defined up to multiplication by a rational function in $\mathbb{F}(\mathbf{x} \setminus \{x_i\})$. In this case, we work with the normalized gcd. For example, let $f = x^3y + xy^3$ and $g = xy^2$, then $\gcd(f, g) = xy$ and $\gcd_y(f, g) = y$. The former is gcd in $\mathbb{F}[x, y]$, while the latter is normalized gcd in $\mathbb{F}(x)[y]$. See [GG99] for details.

Let $f, g \in \mathbb{F}[x_1, \ldots, x_n, y]$ be two non-zero polynomials of y-degree d and e, respectively. Suppose $f(y) = \sum_{i=0}^{d} a_i \cdot y^i$ and $g(y) = \sum_{j=0}^{e} b_j \cdot y^j$, where each $a_i, b_j \in \mathbb{F}[x_1, x_2, \ldots, x_n]$. The *Sylvester matrix* M is the following $(d+e) \times (d+e)$ matrix

$$M = \begin{bmatrix} a_d & a_{d-1} & \ldots & a_1 & a_0 & & & & \\ & a_d & a_{d-1} & \ldots & a_1 & a_0 & & & \\ & & \ldots & \ldots & \ldots & \ldots & & & \\ & & & a_d & a_{d-1} & \ldots & a_1 & a_0 \\ b_e & b_{e-1} & \ldots & b_1 & b_0 & & & \\ & b_e & b_{e-1} & \ldots & b_1 & b_0 & & \\ & & \ldots & \ldots & \ldots & \ldots & & \\ & & & b_e & b_{e-1} & \ldots & b_1 & b_0 \end{bmatrix}.$$

Definition 2.12 (Resultant)**.** *For polynomials $f, g \in \mathbb{F}[y, x_1, \ldots, x_n]$, the resultant* $\text{Res}_y(f, g) \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ *is defined as determinant of the Sylvester matrix. That is,* $\text{Res}_y(f, g) = \det(M)$*.*

In literature, Sylvester matrix is sometimes also defined as $M^\intercal$ but it does not affect the definition of resultant. We use the following properties of the Resultant:

Fact 2.13 (See [GCL92, GG99, CLO15])**.** *Let $f, g \in \mathbb{F}[y, x_1, \ldots, x_n]$. Then,*

1. $\gcd_y(f, g) \neq 1$ *if and only if* $\text{Res}_y(f, g) \equiv 0$*. That is, f and g have a non-trivial factor that depends on the variable y (i.e.,* $\deg_y(\gcd(f, g)) > 0$*) if and only if the y-resultant of f, g is the identically zero polynomial.*

2. *Let $\mathbf{a} \in \mathbb{F}^n$. If* $\deg_y(f) = \deg_y(f|_{\mathbf{x}=\mathbf{a}})$ *and* $\deg_y(g) = \deg_y(g|_{\mathbf{x}=\mathbf{a}})$*, then* $\text{Res}_y(f, g)|_{\mathbf{x}=\mathbf{a}} = \text{Res}_y(f|_{\mathbf{x}=\mathbf{a}}, g|_{\mathbf{x}=\mathbf{a}})$*.*

2.1.3 Primitive Roots of Unity in a Field

Definition 2.14 (Primitive root of unity)**.** *Let $\mathbb{F}$ be a field and $r \in \mathbb{N}$. An element $\omega \in \mathbb{F}$ is called an r-*th primitive root of unity *if $\omega^r = 1$ and for every natural number $1 \leq t < r$: $\omega^t \neq 1$.*

For example, -1 is a second primitive root of unity in $\mathbb{R}$ and i is a fourth primitive root of unity in $\mathbb{C}$. The fact below gives a necessary and sufficient condition on the existence of an r-th primitive root of unity in the algebraic closure of a field. See Theorem 8.2 of [Neu07] for a proof of this fact.

Fact 2.15. *Let $\overline{\mathbb{F}}$ be the algebraic closure of a field $\mathbb{F}$ and $r \in \mathbb{N}$. Then $\overline{\mathbb{F}}$ contains an r-th primitive root of unity if and only if $r \not| \operatorname{char}(\mathbb{F})$.*

It is easy to prove the following observation, which would be used in Section 5.

Observation 2.16. *Let $e \in \mathbb{N}$, $\mathbb{F}$ be a field containing an e-th primitive root of unity ω, and x, y be two variables. Then,*

$$x^e - y^e = \prod_{\ell \in [e]} (x - \omega^\ell y).$$

2.2 Partial Derivatives

The following definition of discrete partial derivatives is taken from [SV15].

Definition 2.17 (Discrete Partial Derivative)**.** *Let $f \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ and $x \in \{x_1, \ldots, x_n\}$. Then, the* discrete partial derivative *of f with respect to x is defined as:*

$$\frac{\partial f}{\partial x} \triangleq f|_{x=1} - f|_{x=0}.$$

Further, let $I = \{i_1, \ldots, i_r\} \subseteq [n]$ be a non-empty set of size. Then, the iterated partial derivative *of f with respect to I is defined as*

$$\frac{\partial^r f}{\partial x_{i_1} \cdots \partial x_{i_r}} \triangleq \frac{\partial}{\partial x_{i_1}} \left(\frac{\partial}{\partial x_{i_2}} \cdots \left(\frac{\partial f}{\partial x_{i_r}} \right) \cdots \right).$$

The following fact relates partial derivatives and justifying assignments of multilinear polynomials. A polynomial $f \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ is *multilinear* if the individual degree of every variable in f is at most one.

Fact 2.18. (Lemmas 2.6 [SV15]) *Let $f \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ be a multilinear polynomial and $\mathbf{a} \in \mathbb{F}^n$. Then, $\mathbf{a}$ is a justifying assignment of f if and only if $f(\mathbf{a}) \neq 0$ and for every $x_i \in \mathrm{var}(f)$ we have that $\frac{\partial f}{\partial x_i}(\mathbf{a}) \neq 0$.*

2.3 Commutator

This section is devoted to commutators of polynomials. This tool was defined in [SV10], where it was used in the context of polynomial factorization. It also played a crucial role in the deterministic reconstruction algorithm for read-once formulas (Definition 3.1) given in [SV14]. The following definition of a commutator of a polynomial is taken from [SV14].

Definition 2.19 (Commutator)**.** *Let $f \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ and $i \neq j \in [n]$. Then, the commutator of f with respect to x_i and x_j, denoted $\Delta_{i,j} f$, is defined as*

$$\Delta_{i,j} f = f|_{x_i=1, x_j=1} \cdot f|_{x_i=0, x_j=0} - f|_{x_i=1, x_j=0} \cdot f|_{x_i=0, x_j=1}.$$

We note that this definition of a commutator of a polynomial is different from the definition given in [SV10]. However, it is not difficult to show that both these definitions have same properties for multilinear polynomials. We now note some useful properties of commutators of multilinear polynomials.

Fact 2.20. (Lemma 4.6 of [SV10]) *Let $f \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ be a non-constant multilinear polynomial and $i \neq j \in [n]$. There exist $g, h \in \mathbb{F}[\mathbf{x}]$ where $i \notin \mathrm{var}(h)$ and $j \notin \mathrm{var}(g)$ such that $f = g \cdot h$ if and only if $\Delta_{i,j} f \equiv 0$.*

The fact above implies the following result.

Corollary 2.21. *Let $f \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ be a multilinear polynomial, where $n \geq 2$. Then, f is reducible if and only if there exist $i, j \in \operatorname{var}(f)$ such that $\Delta_{i,j} f \equiv 0$.*

The following property of a commutator immediately follows from Definition 2.19.

Observation 2.22. *Let $f \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ be a multilinear polynomial, $\mathbf{a} \in \mathbb{F}^n, i \neq j \in \operatorname{var}(f)$, and $I \subseteq \operatorname{var}(f) \setminus \{i, j\}$. Then,*

$$\Delta_{i,j}(f|_{\mathbf{x}_I = \mathbf{a}_I}) = (\Delta_{i,j} f)|_{\mathbf{x}_I = \mathbf{a}_I}.$$

The following useful claim relates commutators and irreducibility preserving assignments, which would play an important role in Section 3.1.4.

Claim 2.23 (Commutators and irreducibility preserving assignments)**.** *Let $f \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ be a multilinear polynomial and $\mathbf{a} \in \mathbb{F}^n$ s.t. $f(\mathbf{a}) \neq 0$. Suppose that for every $i \neq j \in \operatorname{var}(f)$: $(\Delta_{i,j} f)(\mathbf{a}) \neq 0$. Then, f is $\mathbf{a}$-irreducible.*

Proof. Suppose f is not $\mathbf{a}$-irreducible. Then, either $f(\mathbf{a}) = 0$ or there exists a proper subset $I \subsetneq \operatorname{var}(f)$ such that $f|_{\mathbf{x}_I = \mathbf{a}_I}$ is reducible. In the former case, we immediately get a contradiction. Now, suppose the latter holds. Then, there exist non-constant multilinear polynomials $g, h \in \mathbb{F}[\mathbf{x}]$ such that

$$f|_{\mathbf{x}_I = \mathbf{a}_I} = g \cdot h.$$

Let $i \in \operatorname{var}(g)$ and $j \in \operatorname{var}(h)$. As $f|_{\mathbf{x}_I = \mathbf{a}_I}$ is multilinear, g and h are variable disjoint. Then, it follows from Corollary 2.21 that $\Delta_{i,j}(f|_{\mathbf{x}_I = \mathbf{a}_I}) \equiv 0$, which implies $(\Delta_{i,j}(f|_{\mathbf{x}_I = \mathbf{a}_I}))(\mathbf{a}) = 0$. Observation 2.22 implies that $(\Delta_{i,j}(f|_{\mathbf{x}_I = \mathbf{a}_I}))(\mathbf{a}) = (\Delta_{i,j} f)(\mathbf{a})$. Since $(\Delta_{i,j}(f|_{\mathbf{x}_I = \mathbf{a}_I}))(\mathbf{a}) = 0$, we get $(\Delta_{i,j} f)(\mathbf{a}) = 0$, which is a contradiction. Hence, f is $\mathbf{a}$-irreducible. □

The next observation follows from Definition 2.19.

Observation 2.24. *Let $f \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ be a multilinear polynomial and $i, j \in [n]$. Then, f can be written as $f = f_{i,j} x_i x_j + f_i x_i + f_j x_j + f_0$, where $f_{i,j}, f_i, f_j, f_0 \in \mathbb{F}[\mathbf{x} \setminus \{x_i, x_j\}]$. Then $\Delta_{i,j} = f_{i,j} \cdot f_0 - f_i \cdot f_j$.*

Using this, we can easily prove the following observation, which would be used in Section 3.

Observation 2.25. *Let $f \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ be a multilinear polynomial and $i \neq j \in [n]$ such that $\frac{\partial^2 f}{\partial x_i \partial x_j} \equiv 0$. Then,*

$$\Delta_{i,j} f = -\frac{\partial f}{\partial x_i} \cdot \frac{\partial f}{\partial x_j}.$$

2.4 Generators

The problem of blackbox PIT asks for a *hitting set*, which is defined as:

Definition 2.26. *Given a circuit class $\mathcal{C}$ of n-variate polynomials, we say that $\mathcal{H} \subseteq \mathbb{F}^n$ is a* hitting set *for $\mathcal{C}$, if for every non-zero polynomial $f \in \mathcal{C}$, there exists some $\mathbf{a} \in \mathcal{H}$ such that $f(\mathbf{a}) \neq 0$.*

A blackbox PIT algorithm is *efficient* when $\mathcal{H}$ is of polynomial size w.r.t the size s and the degree d of the input circuit (i.e. $|\mathcal{H}| = \mathrm{poly}(n, d, s)$) and can also be constructed in polynomial time. For polynomials over finite fields, by convention, we either assume that size of the field is larger than the size of the hitting set or we assume that we have blackbox access to a large enough extension field. For blackbox PIT, there is also the notion of *hitting set generators* or simply generators in short. Hitting set generators are equivalent in power to hitting sets and are often easier to work with. We refer the reader to the survey of [SY10] for a detailed exposition.

Definition 2.27 (Generator)**.** *Let $\mathcal{C}$ be some class of n-variate polynomials. Consider $\mathcal{G} = (\mathcal{G}^1, \mathcal{G}^2, \ldots, \mathcal{G}^n) : \mathbb{F}^t \to \mathbb{F}^n$, an n-tuple of t-variate polynomials where for each $i \in [n]$, $\mathcal{G}^i \in \mathbb{F}[y_1, y_2, \ldots, y_t]$. For a polynomial $f \in \mathbb{F}[x_1, x_2, \ldots, x_n]$, we define action of $\mathcal{G}$ on polynomial f by $f(\mathcal{G}) = f(\mathcal{G}^1, \ldots, \mathcal{G}^n) \in \mathbb{F}[y_1, \ldots, y_k]$. We call $\mathcal{G}$ a t-seeded generator for class $\mathcal{C}$ if for every non-zero $f \in \mathcal{C}$, $f(\mathcal{G}) \not\equiv 0$. Degree of generator $\mathcal{G}$ is defined as $\deg(\mathcal{G}) \triangleq \max\{\deg(\mathcal{G}^i)\}_{i=1}^{n}$. Image of generator $\mathcal{G}$ is defined as $\mathrm{Im}(\mathcal{G}) \triangleq \mathcal{G}(\overline{\mathbb{F}}^t)$.*

A generator $\mathcal{G}$ for class $\mathcal{C}$ acts as a variable reduction map that reduces the number of variables from n to t while preserving non-zeroness. A generator also contains a *hitting set* for $\mathcal{C}$ in its image. That is, for every nonzero $f \in \mathcal{C}$, there exists $\mathbf{a} \in \mathrm{Im}(\mathcal{G})$ such that $f(\mathbf{a}) \neq 0$.

Fact 2.28 (Generator $\implies$ hitting-set, [SV15])**.** *Let $\mathcal{G} = (\mathcal{G}^1, \ldots, \mathcal{G}^n) : \mathbb{F}^t \to \mathbb{F}^n$ be a generator for a circuit class $\mathcal{C}$ such that $\deg(\mathcal{G}) \triangleq \delta$. Let $W \subseteq \mathbb{F}$ be any set of size $nd\delta$. Then, $\mathcal{H} \triangleq \mathcal{G}(W^t)$ is a hitting set, of size $|\mathcal{H}| \leq (nd\delta)^t$, for polynomials $f \in \mathcal{C}$ of individual degrees less than d.*

In other words, when the seed-length t and the degree δ of the generator is constant, we get a polynomial-time blackbox PIT algorithm. The following observation holds since the ring of polynomials forms an integral domain.

Observation 2.29. *Let $\mathcal{G}$ be a generator for a class $\mathcal{C}$ and let $f = f_1 \cdots f_r$ be an arbitrary product of non-zero polynomials such that for each $i \in [r] : f_i \in \mathcal{C}$. Then $f(\mathcal{G}) \not\equiv 0$.*

2.4.1 The Generator $G_{n,k}$ of [SV15]

The generator $G_{n,k}$ was defined in [SV15] for the class of ROFs. It has been a crucial ingredient in PIT algorithms of various other interesting classes also [KMSV13, FSS14, AvMV15, MV18]. We will also be using this generator in our results. We borrow the definition and properties of this generator as presented in [AvMV15, Vol15].

Definition 2.30. *Let $\alpha_1, \ldots, \alpha_n \in \mathbb{F}$ be n distinct elements and for $i \in [n]$, let $L_i(x) \triangleq \prod_{j \in [n] \setminus \{i\}} \frac{x - \alpha_j}{\alpha_i - \alpha_j}$ denote the corresponding Lagrange interpolant. For every $k \in [n]$, let $G_{n,k} : \mathbb{F}^{2k} \to \mathbb{F}^n$ be defined as*

$$G_{n,k}(y_1, \ldots, y_k, z_1, \ldots, z_k) \triangleq \left(\sum_{j=1}^{k} L_1(y_j) z_j, \sum_{j=1}^{k} L_2(y_j) z_j, \ldots, \sum_{j=1}^{k} L_n(y_j) z_j \right)$$

Let $(G_{n,k})_i$ denote the i^{th} component of $G_{n,k}$ and we call α_i as the Lagrange constant *associated with this i^{th} component. We can also define G_k to be the class of generators $\{G_{n,k}\}_{n\in\mathbb{N}}$ for all output lengths.*

For two generators $\mathcal{G}_1, \mathcal{G}_2$ with the same output length, we define their sum $\mathcal{G}_1 + \mathcal{G}_2$ as their component-wise addition, where the seed variables of both generators are implicitly relabelled so as to be disjoint. With this terminology, we can note various useful properties of the generator $G_{n,k}$ from its definition.

Fact 2.31 ([SV15, KMSV13, Vol15])**.** *Let k, k' be positive integers.*

1. $G_{n,k}(\mathbf{y}, \mathbf{0}) \equiv \mathbf{0}$.
2. $G_{n,k}(y_1, \ldots, y_k, z_1, \ldots, z_k)|_{y_k=\alpha_i} = G_{n,k-1}(y_1, \ldots, y_{k-1}, z_1, \ldots, z_{k-1}) + z_k \cdot \mathbf{e}_i$*, where* $\mathbf{e}$ *is the 0-1 vector with a single 1 in coordinate i and α_i the i^{th} Lagrange constant and* $G_{n,0} \triangleq \mathbf{0}$.
3. $G_{n,k}(y_1, \ldots, y_k, z_1, \ldots, z_k) + G_{n,k'}(y_{k+1}, \ldots, y_{k+k'}, z_{k+1}, \ldots, z_{k+k'})$
   $= G_{n,k+k'}(y_1, \ldots, y_{k+k'}, z_1, \ldots, z_{k+k'})$.
4. *For every $\mathbf{b} \in \overline{\mathbb{F}}^n$ with at most k non-zero components, $\mathbf{b} \in \mathrm{Im}(G_{n,k})$.*

It follows from the definition, that $G_{n,k}$ hits any polynomial containing a low-support monomial.

Fact 2.32 ([SV15, FSS14])**.** *Let $f \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ be a polynomial that contains a non-zero monomial of support-size at most k, for some $k \in \mathbb{N}$. Then $f(G_{n,k}) \not\equiv 0$.*

The next property follows from Definition 2.30 and Fact 2.31 and states that $G_{n,k}$ forms a chain.

Observation 2.33. *Let $f \in \mathbb{F}[x_1, \ldots, x_n]$ be a non-zero polynomial and $k \in \mathbb{N}$ such that $f(G_{n,k}) \not\equiv 0$. Then, for every $\ell \geq k$, $f(G_{n,\ell}) \not\equiv 0$.*

Let $\mathcal{C}$ be a circuit class over a field $\mathbb{F}$. Then we define the class,

$$\mathrm{Res}(\mathcal{C}) \triangleq \{\mathrm{Res}_{x_i}(A, B) \mid A, B \in \mathcal{C} \text{ are irreducible and } i \in \mathrm{var}(A) \cup \mathrm{var}(B)\}.$$

We note that $\mathcal{C} \subseteq \mathrm{Res}(\mathcal{C})$ as for any polynomial $f \in \mathcal{C}$, we can write f as $f = \mathrm{Res}_y(P, Q)$, where $P \triangleq (f+1) \cdot y + 1$ and $Q \triangleq y + 1$ and $y \notin \mathrm{var}(P)$.[4] The following fact is implicit in [Vol15] and [BV22]. However, for the sake of completeness, we provide its proof in Section A.1 of the appendix.

Fact 2.34 (Generator for $\sum^{[2]} \prod \mathcal{C}$)**.** *Let $\mathcal{C}$ be a class of arithmetic circuits over a field $\mathbb{F}$ and $\mathcal{G}$ be a generator for the class* $\mathrm{Res}(\mathcal{C})$*. Then, $H = \mathcal{G} + G_1$ is a generator for the class $\sum^{[2]} \prod \mathcal{C}$.*

[4] We are assuming that both the polynomials $(f+1) \cdot y + 1$ and $y + 1$ are also in $\mathcal{C}$, which is true for all natural classes of polynomials.

2.5 Hardness of Representation

We start with the following definition.

Definition 2.35. *Let $k, n, m \in \mathbb{N}$ and let $A_1 \dots, A_k \in \mathbb{F}[x_1, x_2, \dots, x_n]$ be polynomials. Define $A \triangleq A_1 + \cdots + A_k$. We say that the set $\{A_1, \dots, A_k\}$ is m-*hard*, if and only if either $A \equiv 0$ or for every set $J \subseteq [n]$ of size $|J| = m$, the monomial $\prod_{j \in J} x_j$ does not divide A.*

Remark 2.36. *It follows from the definition that if $n < m$ then any set $\{A_1, \dots, A_k\}$ is m-hard.*

The following fact has been used in many works like [SV15, AvMV15] etc. We provide a proof of this fact in Section A.2 of the appendix.

Fact 2.37 (Hardness of representation implies PIT)**.** *Let $m, n, k \in \mathbb{N}$ and $A_1, \dots, A_k \in \mathbb{F}[x_1, x_2, \dots, x_n]$ such that $A \triangleq A_1 + \cdots + A_k \not\equiv 0$. Suppose further that for every subset $I \subseteq [n]$, the set of restricted polynomials $\{A_1|_{\mathbf{x}_I = \mathbf{0}_I}, \dots, A_k|_{\mathbf{x}_I = \mathbf{0}_I}\}$ is m-hard. Then A contains a non-zero monomial of support-size at most $(m-1)$ and in particular $A(G_{n,m-1}) \not\equiv 0$. Here $G_{n,m-1}$ is the generator given in Definition 2.30.*

3 ROFs and Multilinear Bounded-Read Arithmetic Formulae

3.1 ROFs and ROPs

We start this section with the following definition of a read-once formula.

Definition 3.1 (Read-once formulas, [SV15])**.** *Let $\mathbb{F}$ be a field and $\mathbf{x} = \{x_1, \dots, x_n\}$. A read-once formula (in short, ROF) Φ over $\mathbb{F}$ in $\mathbf{x}$-variables is a binary tree whose leaves are labelled with variables in $\mathbf{x}$ and non-leaf nodes are labelled with $+$ and $\times$. Every variable in $\mathbf{x}$ labels at most one leaf of Φ and every node of Φ is associated with a pair $(\alpha, \beta) \in \mathbb{F}^2$. The computation in Φ proceeds as follows: A leaf node of Φ labelled with $x \in \mathbf{x}$ and (α, β) computes $\alpha x + \beta$. A node v labelled with $\circ \in \{+, \times\}$ and (α, β), and having children v_1 and v_2 computes $\alpha(\Phi_{v_1} \circ \Phi_{v_2}) + \beta$, where Φ_{v_i} is the sub-formula of Φ rooted at v_i.*

We say that a polynomial $A \in \mathbb{F}[\mathbf{x}]$ is a *read-once polynomial* (in short, ROP) if it is computed by an ROF. Note that every ROP is multilinear.

3.1.1 Some Useful Properties of ROFs and ROPs

The following fact shows that the class of read-once formulas is closed under factorization and partial derivatives.

Fact 3.2. (Lemmas 3.6 and 3.12 of [SV15]) *Let $A \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ be an ROP, $i \in [n], I \subseteq [n]$, and $\mathbf{a} \in \mathbb{F}^n$. Then, the substituted polynomial $A|_{\mathbf{x}_I=\mathbf{a}_I}$, the partial derivative $\frac{\partial A}{\partial x_i}$, and factors of A are ROPs.*

Below fact shows that we can make ROPs $\mathbf{0}$-justified by shifting.

Fact 3.3 ([SV15, MV18])**.** *Let $n, k \in \mathbb{N}, \mathbb{F}$ be a field, and $A_1, \ldots, A_k \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ be ROPs. Then, we there exists an $\mathbf{a} \in \mathrm{Im}(G_{n,1})$ in such that for every $t \in [k], A_t(\mathbf{x}+\mathbf{a})$ is $\mathbf{0}$-justified.*

The fact below follows from Theorem 3.10 of [SV14].

Fact 3.4. *Let $A \in \mathbb{F}[\mathbf{x}]$ be a $\mathbf{0}$-justified ROP. Then, every partial derivative of A is also a $\mathbf{0}$-justified ROP.*

The next observation follows from Definition 3.1.

Observation 3.5. *Let $\mathbb{F}$ be a field and $A, B \in \mathbb{F}[\mathbf{x}]$ be two variable disjoint ROPs. Then, $A \cdot B$ is also an ROP.*

3.1.2 Commutator of an ROP

Fact 3.6. (Lemma 3.14 of [SV14]) *Let $A \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ be an ROP and $i \neq j \in [n]$ such that $\frac{\partial^2 A}{\partial x_i \partial x_j} \not\equiv 0$. Then, there exist variable disjoint ROPs $B(\mathbf{x}), R(\mathbf{x}, y)$ such that $A = R(\mathbf{x}, B(\mathbf{x}))$ and*

$$\Delta_{i,j}A = R(\mathbf{x}, 0) \cdot \frac{\partial^2 A}{\partial x_i \partial x_j}.$$

Facts 3.2, Observation 2.25, and Fact 3.6 imply the following useful result.

Corollary 3.7 (Structure of a commutator of an ROP)**.** *Let $A \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ be an ROP and $i \neq j \in [n]$. Then, $\Delta_{i,j}A$ is a product of ROPs in $\mathbb{F}[x_1, x_2, \ldots, x_n]$.*

3.1.3 The Hardness of Representation Theorem for Sum of ROPs

Now, we list results related to the hardness of representation for sum of constantly many $\mathbf{0}$-justified ROPs. These would be used in Sections 4 and 5. Recall Definition 2.1.

Fact 3.8 (Hardness of representation for sum of k $\mathbf{0}$-justified ROPs, Theorem 6.1 of [SV15])**.** *Let $n, k \in \mathbb{N}$ and $A_1, \ldots, A_k \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ be $\mathbf{0}$-justified ROPs. Suppose $n \geq 3k$. Then, for every collection of sets $J_1, \ldots, J_k \subseteq [n]$ and every collection of field elements $\alpha_1, \ldots, \alpha_k \in \mathbb{F}$, the set $\{\alpha_1 \cdot A_1|_{\mathbf{x}_{J_1}=\mathbf{0}_{J_1}}, \ldots, \alpha_k \cdot A_k|_{\mathbf{x}_{J_k}=\mathbf{0}_{J_k}}\}$ is $3k$-hard.*

When $k = 2$, we can, in fact, show that the polynomials are 3-hard rather than 6-hard. Although a minor improvement, we present it formally in the following fact as it would be used in Section 4. We provide a proof of this fact in Section A.3 of the appendix.

Fact 3.9 (Hardness of representation for sum of two **0**-justified ROPs)**.** *Let $\mathbb{F}$ be a field and $A, B \in \mathbb{F}[x_1, x_2, \dots, x_n]$ be two* **0***-justified ROPs. Then the set $\{A, B\}$ is 3-hard.*

Using this fact, we give the following useful result used in Section 4.

Claim 3.10. *Let $n \geq 3$ be a natural number and $A, B \in \mathbb{F}[x_1, x_2, \dots, x_n]$ be two* **0***-justified ROPs. Suppose there exists a $J \subseteq [n], |J| = 3$ such that for every $j \in J, A|_{x_j=0} = \alpha_j \cdot B|_{x_j=0}$ for some $\alpha_j \in \mathbb{F}$. Then, $A \sim B$.*

Proof. Let $j, k \in J$ be distinct. As A, B are **0**-justified, it follows from Definition 2.1 that

$$A|_{x_j=0,x_k=0} = \alpha_j \cdot B|_{x_j=0,x_k=0} \not\equiv 0,$$
$$A|_{x_j=0,x_k=0} = \alpha_k \cdot B|_{x_j=0,x_k=0} \not\equiv 0.$$

These two equations immediately imply that there exists a non-zero $\alpha \in \mathbb{F}$ such that for every $j \in J, \alpha_j = \alpha$. Hence, for every $j \in J, A|_{x_j=0} = \alpha \cdot B|_{x_j=0}$. Now, suppose $A - \alpha \cdot B \not\equiv 0$. As for every $j \in J, A|_{x_j=0} - \alpha \cdot B|_{x_j=0} \equiv 0$, Fact 2.7 implies that for every $j \in J, x_j$ divides $A - \alpha \cdot B$. Since B is a **0**-justified ROP, $\alpha \cdot B$ is also a **0**-justified ROP. Hence, $\prod_{j \in J} x_j$ divides $A - \alpha \cdot B$, which can not happen because of Fact 3.9. Thus, $A = \alpha \cdot B$. Hence proved. □

3.1.4 Obtaining 0-Irreducible ROPs

In this section, we give a procedure to convert an irreducible ROP to a **0**-irreducible ROP (Definition 2.4). In particular, if $A \in \mathbb{F}[\mathbf{x}]$ is an irreducible n-variate ROP then we compute an assignment $\mathbf{a} \in \mathbb{F}^n$ such that A is $\mathbf{a}$-irreducible. Observation 2.6 implies that $A(\mathbf{x} + \mathbf{a})$ is a **0**-irreducible ROP.

It follows from Claim 2.23 that if all the commutators of a multilinear polynomial $f \in \mathbb{F}[\mathbf{x}]$ are non-zero and if we can efficiently hit all these commutators, i.e., we can efficiently compute an $\mathbf{a} \in \mathbb{F}^n$ such that for every $i \neq j \in \text{var}(f), (\Delta_{i,j} f)(\mathbf{a}) \neq 0$, then using Observation 2.6, we transform f to a **0**-irreducible polynomial. It follows from Corollary 2.21 that for every $i \neq j \in \text{var}(f), \Delta_{i,j} f \not\equiv 0$ if and only if f is irreducible. Thus, only irreducible multilinear polynomials are eligible to be transformed into **0**-irreducible polynomials. The following fact from [MV18] would be used in Claim 3.12.

Fact 3.11 (Theorem 4.2 of [MV18])**.** *Let $A \in \mathbb{F}[x_1, x_2, \dots, x_n]$ be a non-zero ROP and $G_{n,1}$ be the generator given in Definition 2.30. Then, $A(G_{n,1}) \not\equiv 0$.*

Claim 3.12 (Converting a set of irreducible ROPs to **0**-irreducible ROPs)**.** *Let $n, m \in \mathbb{N}$ and $A_1, \dots, A_m \in \mathbb{F}[x_1, x_2, \dots, x_n]$ be irreducible ROPs. Then, there exists an assignment $\mathbf{a} \in \mathbb{F}^n$ in the image of $G_{n,1}$ (see Definition 2.30) such that $A_\ell(\mathbf{x} + \mathbf{a})$ is a* **0***-irreducible ROP for every $\ell \in [m]$.*

Proof. As $A_1, \dots, A_m$ are irreducible, Corollary 2.21 implies that for every $\ell \in [m], i \neq j \in \text{var}(A_\ell), \Delta_{i,j} A_\ell \not\equiv 0$. It follows from Claim 2.23 that it is sufficient to show that $G_{n,1}$ hits $A_\ell, \Delta_{i,j} A_\ell$ for every $\ell \in [m], i \neq j \in \text{var}(A_\ell)$. Let

$$\Phi(\mathbf{x}) \triangleq \prod_{\ell \in [m]} A_\ell \prod_{i,j \in \text{var}(A_\ell), i \neq j} \Delta_{i,j} A_\ell.$$

As every A_ℓ is irreducible, it is non-zero (recall that every element of $\mathbb{F}$ is reducible). Then, it follows from Corollary 3.7 that $\Phi(\mathbf{x})$ is a product of non-zero ROPs in $\mathbb{F}[\mathbf{x}]$. Now, Fact 3.11 and Observation 2.29 together imply that $\Phi(G_{n,1}) \not\equiv 0$. Hence, it follows from Fact 2.28 that there exists an $\mathbf{a}$ in the image of $G_{n,1}$ such that $\Phi(\mathbf{a}) \neq 0$. Now, Claim 2.23 and Observation 2.6 imply that $A_\ell(\mathbf{x}+\mathbf{a})$ is a $\mathbf{0}$-irreducible ROP for every $\ell \in [m]$. □

3.2 Multilinear Bounded-Read Arithmetic Formulae

Definition 3.13 (Read-k formula)**.** *Let $\mathbb{F}$ be a field and $k \in \mathbb{N}$. A read-k arithmetic formula F is a tree where every leaf node is labelled either by a variable or an element of $\mathbb{F}$; every other node (or internal node) is labelled by either $+$ or $\times$; every edge is labelled by an element of $\mathbb{F}$; and every variable labels at most k leaves of F. Every leaf node of F computes its label. Suppose v is an internal node of F labelled by $\circ \in \{+, \times\}$ such that $v_1, \ldots, v_m$ are the children of v, for every $i \in [m]$, v_i computes $F_{v_i} \in \mathbb{F}[\mathbf{x}]$ and the edge between v and v_i is labelled by $\alpha_i \in \mathbb{F}$. Then, v computes the following polynomial*

$$F_v \triangleq \alpha_1 \cdot F_{v_1} \circ \cdots \circ \alpha_m \cdot F_{v_m}.$$

Further, if every node of F computes a multilinear polynomial then F is called a multilinear read-k arithmetic formula.

An ROF is a special case of a multilinear bounded-read formula. One of the reasons for studying multilinear bounded-read arithmetic formulae is that developing deep understanding of such formulae might give us good insights about the class of multilinear formulae, which is an important class of arithmetic circuits. Deterministic algorithms for blackbox and whitebox PIT for multilinear bounded-read arithmetic formulae were given in [AvMV15]. For the rest of this section, let $k \in \mathbb{N}$ be a fixed constant and $\mathcal{C}_k$ be the class of multilinear read-k formulae over a field $\mathbb{F}$. The following result would be used in the proof of Theorem 3.

Observation 3.14. *Let $k \in \mathbb{N}, \mathbb{F}$ be a field, and $A, B \in \mathcal{C}_k$ be two variable disjoint polynomials over $\mathbb{F}$. Then, $A \cdot B \in \mathcal{C}_k$.*

The following fact would play a crucial role in the proof of Theorem 3.

Fact 3.15 (Implicit in [AvMV15])**.** *Let $k \in \mathbb{N}$, $m \triangleq (8k \cdot (k+1)^2)^k$, and $\widetilde{A}, \widetilde{B}, \widetilde{R} \in \mathcal{C}_k$ compute n-variate polynomials over a field $\mathbb{F}$. Let $\ell \triangleq m + 3k\lceil \log n \rceil$. Then, there exists an assignment $\mathbf{a} \in \mathrm{Im}(G_{n,\ell})$ such that the polynomials $A \triangleq \widetilde{A}(\mathbf{x}+\mathbf{a}), B \triangleq \widetilde{B}(\mathbf{x}+\mathbf{a})$, and $R \triangleq \widetilde{R}(\mathbf{x}+\mathbf{a})$ satisfy Properties 1, 2, and 3 given in Theorem 5.6.*

4 PIT for $\sum^{[2]}\prod$ ROF

This section is devoted to the proof of Theorem 1, which is based on the proof overview given in Section 1.3.1. We first present some results related to the resultant of two co-prime and $\mathbf{0}$-irreducible ROPs in Section 4.1. These results are required for proving Theorem 4.5. Then, using Theorem 4.5, we give a proof of Theorem 1 in Section 4.2.

4.1 Properties of the Resultant of two 0-Irreducible ROPs

In this section, we note some important results related to the resultant of two n-variate, co-prime, and $\mathbf{0}$-irreducible ROPs A, B (Definition 2.4). The most important result that we prove is a hardness of representation theorem for $\mathrm{Res}_x(A, B)$ (Lemma 4.2). A key consequence of this result is that there exists a monomial of support (at most) two in $\mathrm{Res}_x(A, B)$ (Corollary 4.3). Using this, we show in Lemma 4.4 that $\mathrm{Res}_x(A, B)(G_{n,3}) \not\equiv 0$, where A, B are any co-prime ROPs (not necessarily $\mathbf{0}$-irreducible). We start with the following claim.

Claim 4.1. *Let $f, g \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ be multilinear polynomials, where f is $\mathbf{0}$-irreducible and $i \neq j \in [n]$ such that $\mathrm{var}(f) \setminus \{i, j\} \neq \emptyset$. Suppose x_j divides $\mathrm{Res}_{x_i}(f, g)$. Then, $f|_{x_j=0}$ divides $g|_{x_j=0}$.*

Proof. As f, g are multilinear, there exist multilinear $f_i, f_0, g_i, g_0 \in \mathbb{F}[\mathbf{x} \setminus \{x_i\}]$ such that $f = f_i \cdot x_i + f_0$ and $g = g_i \cdot x_i + g_0$. Then,

$$f|_{x_j=0} = f_i|_{x_j=0} \cdot x_i + f_0|_{x_j=0}.$$

We first claim that $f_0|_{x_j=0}$ is non-constant and $f_i|_{x_j=0} \not\equiv 0$. Since f is $\mathbf{0}$-irreducible, and $\mathrm{var}(f) \setminus \{i, j\} \neq \emptyset$, by Definition 2.4, $f_{x_i=0,x_j=0}$ is irreducible. As f_0 is x_i-free, note that $f|_{x_i=0,x_j=0} = f_0|_{x_j=0}$, which implies $f_0|_{x_j=0}$ is a non-constant polynomial. Now, suppose $f_i|_{x_j=0} \equiv 0$. Then, clearly $f|_{x_j=0}$ does not depend on x_i. This can not happen as f is $\mathbf{0}$-justified, which follows from Claim 2.5. Now, we claim that $f_i|_{x_j=0}$ and $f_0|_{x_j=0}$ are co-prime. Suppose not. Let $h \in \mathbb{F}[\mathbf{x}]$ be a non-constant polynomial that divides $f_i|_{x_j=0}$ and $f_0|_{x_j=0}$. Then, there exist $v_0, v_i \in \mathbb{F}[\mathbf{x}]$ such that $f_i|_{x_j=0} = v_i \cdot h$ and $f_0|_{x_j=0} = v_0 \cdot h$. Then,

$$f|_{x_j=0} = f_i|_{x_j=0} \cdot x_i + f_0|_{x_j=0} = h(v_i x_i + v_0).$$

Thus, $f|_{x_j=0}$ is reducible, which contradicts that f is $\mathbf{0}$-irreducible. Hence, $f_i|_{x_j=0}$ and $f_0|_{x_j=0}$ are co-prime polynomials. As $f = f_i \cdot x_i + f_0$, it follows from Definition 2.12 that

$$\mathrm{Res}_{x_i}(f, g) = f_i \cdot g_0 - g_i \cdot f_0.$$

Since x_j divides $\mathrm{Res}_{x_i}(f, g)$, we get $\mathrm{Res}_{x_i}(f, g)|_{x_j=0} \equiv 0$. Then, the above equation implies

$$f_i|_{x_j=0} \cdot g_0|_{x_j=0} = g_i|_{x_j=0} \cdot f_0|_{x_j=0}. \tag{1}$$

Since both $f_0|_{x_j=0}$ and $f_i|_{x_j=0}$ are non-zero, the above equation implies that $g_0|_{x_j=0} \equiv 0$ if and only if $g_i|_{x_j=0} \equiv 0$. If this happens then $g|_{x_j=0} \equiv 0$ and in this case, $f|_{x_j=0}$ obviously divides $g|_{x_j=0}$. Now, suppose $g|_{x_j=0} \not\equiv 0$, which implies $g_0|_{x_j=0}$ and $g_i|_{x_j=0}$ are non-zero. As $f_i|_{x_j=0}$ and $f_0|_{x_j=0}$ are co-prime and $f_0|_{x_j=0}$ is a non-constant polynomial, Equation (1) implies that there exists a non-zero polynomial $v \in \mathbb{F}[\mathbf{x}]$ such that

$$g_i|_{x_j=0} = f_i|_{x_j=0} \cdot v \quad \text{and} \quad g_0|_{x_j=0} = f_0|_{x_j=0} \cdot v.$$

As $f|_{x_j=0} = f_i|_{x_j=0} \cdot x_i + f_0|_{x_i=0}$ and $g|_{x_j=0} = g_i|_{x_j=0} \cdot x_i + g_0|_{x_i=0}$, the equation above implies that $g|_{x_j=0} = v \cdot f|_{x_j=0}$. □

The following lemma lies at the heart of the proof of Theorem 1. We show that the resultant of two **0**-irreducible ROPs is 3-hard (see Definition 2.35).

Lemma 4.2 (Hardness of representation for the resultant of two **0**-irreducible ROPs)**.** *Let $n \geq 3$ be a natural number and $A, B \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ be **0**-irreducible ROPs. Let $i \in [n]$ be such that $\mathrm{Res}_{x_i}(A, B) \not\equiv 0$. Then $\mathrm{Res}_{x_i}(A, B)$ is 3-hard.*

Proof. Suppose for contradiction that $\mathrm{Res}_{x_i}(A, B)$ is not 3-hard, that is, $\mathrm{Res}_{x_i}(A, B) \not\equiv 0$ and there exists a $J \subseteq [n], |J| = 3$ such that for every $j \in J, x_j$ divides $\mathrm{Res}_{x_i}(A, B)$. We claim that this implies either $J \subseteq \mathrm{var}(A)$ or $J \subseteq \mathrm{var}(B)$. As x_j divides $\mathrm{Res}_{x_i}(A, B)$ for every $j \in J$, we get that $j \in \mathrm{var}(A) \cup \mathrm{var}(B)$. Observe that this implies either $|\mathrm{var}(A) \cap J| \geq 2$ or $|\mathrm{var}(B) \cap J| \geq 2$. Suppose the former is true and $j \neq k \in \mathrm{var}(A) \cap J$. As A is **0**-irreducible, it follows from Claim 4.1 that $A|_{x_j=0}$ divides $B|_{x_j=0}$ and $A|_{x_k=0}$ divides $B|_{x_k=0}$. This implies that $j, k \in \mathrm{var}(B)$. Let $\ell \in J \setminus \{j, k\}$. Since $\ell \in \mathrm{var}(A) \cup \mathrm{var}(B)$, we get that either $J \subseteq \mathrm{var}(A)$ or $J \subseteq \mathrm{var}(B)$.

By using the fact that B is also **0**-irreducible and by using a similar argument as above, it follows that $J \subseteq \mathrm{var}(A) \cap \mathrm{var}(B)$. This implies that for every $j \in J$, there exists a non-zero $\alpha_j \in \mathbb{F}$ such that $A|_{x_j=0} = \alpha_j \cdot B|_{x_j=0}$. Since B is **0**-irreducible, $|\mathrm{var}(B)| \geq 3$, and $\alpha_j \neq 0$ we get that $\alpha_j \cdot B|_{x_j=0} \not\equiv 0$. Since A and B are **0**-irreducible ROPs, Claim 2.5 implies that these are also **0**-justified. Then, it follows from Claim 3.10 that there exists an $\alpha \neq 0 \in \mathbb{F}$ such that $A = \alpha \cdot B$. But this means that A, B are not co-prime, thus Fact 2.13 implies that $\mathrm{Res}_{x_i}(A, B) \equiv 0$, which is a contradiction. Hence, $\mathrm{Res}_{x_i}(A, B)$ is 3-hard. □

Using Lemma 4.2, we can now show that $\mathrm{Res}_{x_i}(A, B)$ has a monomial of support-size at most 2 and hence we can hit $\mathrm{Res}_{x_i}(A, B)$ using $G_{n,2}$. Note that $\mathrm{Res}_{x_i}(A, B)$ by definition does not depend on x_i but we can still consider it as a polynomial in $\mathbb{F}[x_1, \ldots, x_n]$. In this case, when we apply any generator $\mathcal{G} : \mathbb{F}^t \to \mathbb{F}^n$, it will not substitute anything for variable x_i. Henceforth, we follow this convention for any $(n-1)$-variate polynomial.

Corollary 4.3. *Let $A, B \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ be **0**-irreducible ROPs such that $\mathrm{Res}_{x_i}(A, B) \not\equiv 0$ for some $i \in [n]$. Then, $\mathrm{Res}_{x_i}(A, B)$ contains a monomial of support-size at most 2. In particular, $(\mathrm{Res}_{x_i}(A, B))(G_{n,2}) \not\equiv 0$.*

Proof. Let $R \triangleq \mathrm{Res}_{x_i}(A, B) \not\equiv 0$. Let J be any subset of $\mathrm{var}(R)$. Note that $i \notin J$. Since A, B are **0**-irreducible and multilinear, we have that $\deg_{x_i}(P) = \deg_{x_i}(P|_{\mathbf{x}_J=\mathbf{0}_J}) = 1$ for both $P = A, B$. Then by Fact 2.13,

$$R|_{\mathbf{x}_J=\mathbf{0}_J} \;=\; \mathrm{Res}_{x_i}(A, B)|_{\mathbf{x}_J=\mathbf{0}_J} \;=\; \mathrm{Res}_{x_i}(A|_{\mathbf{x}_J=\mathbf{0}_J}, B|_{\mathbf{x}_J=\mathbf{0}_J}). \tag{2}$$

Since A, B are **0**-irreducible, both $A|_{\mathbf{x}_J=\mathbf{0}_J}, B|_{\mathbf{x}_J=\mathbf{0}_J}$ are also **0**-irreducible by definition. Then by Lemma 4.2 we deduce that $R|_{\mathbf{x}_J=\mathbf{0}_J}$ is also 3-hard for any $J \subseteq \mathrm{var}(R)$. Now, we can use Fact 2.37 to show that R has a monomial of support-size at most 2 and thus $R(G_{n,2}) \not\equiv 0$. □

In Corollary 4.3 we showed how to hit the resultant of two **0**-irreducible ROPs. In the lemma below, we show how to hit resultant of two general ROPs. Recall the definition of the class $\mathrm{Res}(\mathcal{C})$ and Fact 2.34 from Section 2.

Lemma 4.4. *G_3 is a generator for the class* $\mathrm{Res}(\mathrm{ROF})$.

Proof. Let A, B be two irreducible ROPs in $\mathbb{F}[x_1, x_2, \ldots, x_n]$. We show that for every $i \in [n]$ s.t. $\mathrm{Res}_{x_i}(A, B) \not\equiv 0$ we have that $(\mathrm{Res}_{x_i}(A, B))(G_{n,3}) \not\equiv 0$. Since, $\mathrm{Res}_{x_i}(A, B) \not\equiv 0$, Fact 2.13 implies that A and B are co-prime with respect to x_i. By Claim 3.12, there exists an $\mathbf{a} \in \mathrm{Im}(G_{n,1})$ such that $\tilde{A} \triangleq A(\mathbf{x} + \mathbf{a})$ and $\tilde{B} \triangleq B(\mathbf{x} + \mathbf{a})$ are $\mathbf{0}$-irreducible polynomials. Moreover they are co-prime with respect to x_i, since A, B are. From Corollary 4.3, we deduce that $(\mathrm{Res}_{x_i}(\tilde{A}, \tilde{B}))(G_{n,2}) \not\equiv 0$.

Let $A = A_i(\mathbf{x}) \cdot x_i + A_0(\mathbf{x})$ and $B = B_i(\mathbf{x}) \cdot x_i + B_0(\mathbf{x})$, where A_i, A_0, B_i, B_0 do not depend on x_i. Then, we get that

$$\begin{aligned}
\tilde{A} &= A_i(\mathbf{x} + \mathbf{a})(x_i + a_i) + A_0(\mathbf{x} + \mathbf{a}) \\
&= A_i(\mathbf{x} + \mathbf{a}) \cdot x_i \ + \ a_i A_i(\mathbf{x} + \mathbf{a}) + A_0(\mathbf{x} + \mathbf{a}) \\
\tilde{B} &= B_i(\mathbf{x} + \mathbf{a}) \cdot x_i \ + \ a_i B_i(\mathbf{x} + \mathbf{a}) + B_0(\mathbf{x} + \mathbf{a}).
\end{aligned}$$

On computing the resultant, we get $\mathrm{Res}_{x_i}(\tilde{A}, \tilde{B}) = (\mathrm{Res}_{x_i}(A, B))(\mathbf{x} + \mathbf{a})$. Thus, we deduce that $(\mathrm{Res}_{x_i}(A, B))(G_{n,2} + \mathbf{a}) = (\mathrm{Res}_{x_i}(\tilde{A}, \tilde{B}))(G_{n,2}) \not\equiv 0$. Since $\mathbf{a} \in \mathrm{Im}(G_{n,1})$, it follows that

$$(\mathrm{Res}_{x_i}(A, B))(G_{n,2} + G_{n,1}) = (\mathrm{Res}_{x_i}(A, B))(G_{n,3}) \not\equiv 0. \qquad \square$$

4.2 Proof of Theorem 1: PIT for $\sum^{[2]} \prod$ ROF

In Section 1.3, we outlined why it suffices to hit the resultants of two ROFs in order to hit the class $\sum^{[2]} \prod$ ROF. For a general class $\mathcal{C}$, it is implicitly shown in previous works like [Vol15, BV22] that it suffices to hit the class $\mathrm{Res}(\mathcal{C})$, which we formally stated in Fact 2.34 and proved in Appendix A.1. In Lemma 4.4, we have shown that G_3 is a generator for the class $\mathrm{Res}(\mathrm{ROF})$. As a consequence, we get that $G_{n,4} = G_{n,3} + G_{n,1}$ is a generator for any n-variate polynomial in the class $\sum^{[2]} \prod$ ROF.

Theorem 4.5 (G_4 hits $\sum^{[2]} \prod$ ROF)**.** *Let $f \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ be a polynomial in the class $\sum^{[2]} \prod$ ROF and let $G_{n,4}$ be the generator given in Definition 2.30. Then, $f \equiv 0$ if and only if $f(G_{n,4}) \equiv 0$.*

Proof. In Lemma 4.4, we showed that G_3 is a generator for the class $\mathrm{Res}(\mathrm{ROF})$. Then by Fact 2.34, $G_4 = G_3 + G_1$ is a generator for the class $\sum^{[2]} \prod$ ROF, that is, given an n-variate polynomial $f \in \sum^{[2]} \prod$ ROF, $f \equiv 0$ if and only if $f(G_{n,4}) \equiv 0$. $\square$

Now, we are ready to prove Theorem 1.

Proof of Theorem 1. We are given an n-variate polynomial $f \in \sum^{[2]} \prod$ ROF such that $\deg(f) \leq d$. Then, Theorem 4.5 implies that $f(G_{n,4}) \equiv 0$ if and only if $f \equiv 0$. Since $f(G_{n,4})$ is an eight-variate polynomial and has degree at most $n \cdot d$, it follows from Fact 2.28 that the zeroness of $f(G_{n,4})$ can be tested in $\mathrm{poly}(n, d)$ time. This completes the proof of Theorem 1.

5 PIT for $\sum^{[3]} \bigwedge \mathcal{C}$

This section is devoted to the proofs of Theorems 2 and 3. Here, we prove a more general result in Section 5.2, which subsumes these two theorems (see Theorem 5.9). Its proof goes via a hardness of representation result given in Theorem 5.6. Before coming to this theorem, we discuss some useful results in the next section.

5.1 Some Useful Results

In this section, we give a set of results required in the proof of Theorem 5.6, which lies at the core of the proofs of Theorems 2 and 3. Recall definition of a m-hard set of polynomials (Definition 2.35). The following result generalizes Claim 3.10.

Claim 5.1. *Let $n, m \in \mathbb{N}$, $n \geq m \geq 2$ and $A, B \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ be two $\mathbf{0}$-justified polynomials such that for every $\beta \in \mathbb{F}$, the set $\{A,\ \beta \cdot B\}$ is m-hard. Suppose there exists a set $J \subseteq [n], |J| = m$ such that for every $j \in J$, there exists an $\alpha_j \in \mathbb{F}$ satisfying $A|_{x_j=0} = \alpha_j \cdot B|_{x_j=0}$. Then, $A \sim B$*

Proof. Since A, B are $\mathbf{0}$-justified polynomials, it follows from Definition 2.1 that for every $j \in J$, $A|_{x_j=0} \not\equiv 0$ and $B|_{x_j=0} \not\equiv 0$ and therefore $\alpha_j \neq 0$. Let $i, j \in J$ be arbitrary distinct indices. As $m \geq 2$, such indices exist. As $\alpha_i \neq 0$ and $\alpha_j \neq 0$, we get

$$A|_{x_i=0,x_j=0} = \alpha_i \cdot B|_{x_i=0,x_j=0} = \alpha_j \cdot B|_{x_i=0,x_j=0}.$$

Since A, B are $\mathbf{0}$-justified, $A|_{x_i=0,x_j=0} \not\equiv 0$ and $B|_{x_i=0,x_j=0} \not\equiv 0$ and therefore $\alpha_i = \alpha_j$. Then, the above equation implies that there exists an $\alpha \in \mathbb{F} \setminus \{0\}$ such that for every $j \in J, \alpha_j = \alpha$. We claim that $A = \alpha \cdot B$. Suppose not. As for every $j \in J, A|_{x_j=0} = \alpha \cdot B|_{x_j=0}$, Fact 2.7 implies that x_j divides $A - \alpha \cdot B$. Thus, the monomial $\prod_{j \in J} x_j$ divides $A - \alpha B$. Since $|J| = m$ and $A - \alpha \cdot B \not\equiv 0$, this contradicts our assumption that the set $\{A, -\alpha \cdot B\}$ is m-hard. Hence, $A = \alpha \cdot B$. □

Claim 5.2. *Let $\mathbb{F}$ be a field, $n, e, d \in \mathbb{N}$, such that $e \geq 2, \mathrm{char}(\mathbb{F})$ does not divide e, and $d \leq e$. Let $f, g \in \mathbb{F}[x_1, x_2, \ldots, x_n]$, and $\alpha \in \mathbb{F} \setminus \{0\}$. Suppose $f^e - g^e = \alpha$. Then $f, g \in \mathbb{F}$.*

Proof. As $\mathrm{char}(\mathbb{F})$ does not divide e, Fact 2.15 implies that there exists an e-th primitive root of unity $\omega \in \mathbb{F}$. Then, it follows from Fact 2.16 that

$$f^e - g^e = \prod_{\ell \in [e]} (f - \omega^\ell g) = \alpha. \tag{3}$$

As $e \geq 2$ and $\alpha \neq 0$, we get from the above equation that for every $\ell_1 \neq \ell_2 \in [e]$, there exist non-zero $\alpha_1, \alpha_2 \in \mathbb{F}$ such that

$$f - \omega^{\ell_1} g = \alpha_1 \ \text{ and } \ f - \omega^{\ell_2} g = \alpha_2.$$

These two equations can be expressed as follows.

$$\begin{bmatrix} 1 & -\omega^{\ell_1} \\ 1 & -\omega^{\ell_2} \end{bmatrix} \cdot \begin{bmatrix} f \\ g \end{bmatrix} = \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix}.$$

As ω is a primitive root of unity, the 2×2 matrix, say M, in the L.H.S. of the above equation is invertible over $\mathbb{F}$. On multiplying the above equation with M^{-1}, we get that $f, g \in \mathbb{F}$. □

The following two claims would be used to handle two important cases in the proof of Theorem 5.6. Recall Definition 2.14 and Fact 2.15.

Claim 5.3. *Let $\mathbb{F}$ be a field, $n, e, d \in \mathbb{N}$, such that $2 \leq d \leq e$ and* $\text{char}(\mathbb{F})$ *does not divide e. Let $f, g, h \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ be multilinear polynomials such that h is non-constant. Suppose $f^e - g^e = h^d$. Then, we get the following.*

1. *If $d \geq 2$ then $d = e$ and $f \sim g \sim h$.*

2. *If $d = 1$ then $e = 2$.*

Proof. As $\text{char}(\mathbb{F})$ does not divide e, Fact 2.15 implies that there exists an e-th primitive root of unity $\omega \in \mathbb{F}$. Then, it follows from Fact 2.16 that

$$f^e - g^e = \prod_{\ell \in [e]} (f - \omega^\ell g) = h^d. \tag{4}$$

1. Suppose $d \geq 2$. Let v be an irreducible factor of h. As $d \geq 2$ and f, g are multilinear polynomials, the above equation along with uniqueness of factorization property of $\mathbb{F}[x_1, x_2, \ldots, x_n]$ implies that there exist distinct $\ell_1, \ell_2 \in [e]$ and two non-zero polynomials $u_1, u_2 \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ such that

   $$f - \omega^{\ell_1} g = u_1 \cdot v \;\text{ and }\; f - \omega^{\ell_2} g = u_2 \cdot v.$$

   These two equations can be expressed as follows.

   $$\begin{bmatrix} 1 & -\omega^{\ell_1} \\ 1 & -\omega^{\ell_2} \end{bmatrix} \cdot \begin{bmatrix} f \\ g \end{bmatrix} = \begin{bmatrix} u_1 \cdot v \\ u_2 \cdot v \end{bmatrix}.$$

   Recall from the proof of Claim 5.2 that we called the 2×2 matrix in the L.H.S. of the above equation as M and argued that it is invertible. On multiplying the above equation with M^{-1}, we get that v divides both f and g. Then, it follows from Equation (4) that v^e divides h^d, which along with the assumption that $e \geq d$ implies $d = e$. Thus,

   $$f^e - g^e = h^e.$$

   As v is an arbitrary irreducible factor of h, we get that h divides both f and g. Since $f^e - g^e = h^e$, by using a similar argument, we get g divides f and h. Hence, $g \sim h$. Then, there exists an $\alpha \in \mathbb{F}$ such that

   $$f^e = \alpha \cdot g^e.$$

   This implies that f, g, h are similar polynomials.

2. Suppose $d = 1$. Then, Equation (4) implies

$$f^e - g^e = \prod_{\ell \in [e]} (f - \omega^{\ell} g) = h.$$

Let $\ell_1 \neq \ell_2 \in [e]$. For $t \in [2]$, let $h_t \triangleq f - \omega^{\ell_t} g$. Then, h_1 and h_2 are factors of h. Now, using a similar matrix based argument as above, it is not difficult to show that $\mathrm{var}(f), \mathrm{var}(g) \subseteq \mathrm{var}(h_1) \cup \mathrm{var}(h_2)$. Since h is multilinear, all it factors are variable disjoint. Note that we can not have $e > 2$ since it would violate the variable disjointness of factors of h. Thus, $e = 2$. □

Let $n, r \in \mathbb{N}, r \leq n$, $\mathcal{P}_r$ be the monomial $x_1 \dots x_r$, and $\mathcal{I}_r \triangleq \langle \mathcal{P}_r \rangle$ be the monomial ideal in $\mathbb{F}[x_1, x_2, \dots, x_n]$. For the lemma below, recall Definition 2.10. Here we consider a polynomial f w.r.t. ideal $\mathcal{I}_r$ and write $f = \tilde{f} + \widehat{f}$, where $\widehat{f} = f \pmod{\mathcal{I}_r}$.

Lemma 5.4. *Let $f \in \mathbb{F}[x_1, x_2, \dots, x_n]$ be a non-constant $\mathbf{0}$-justified multilinear polynomial such that $f = g \cdot h + v \cdot \mathcal{P}_r$, where $3 \leq r \leq n$ and $v, g, h \in \mathbb{F}[x_1, x_2, \dots, x_n]$ are arbitrary polynomials (possibly non-multilinear). Then $\widehat{g}$ and $\widehat{h}$ are variable disjoint.*

Proof. Since $g = \tilde{g} + \widehat{g}$ and $h = \tilde{h} + \widehat{h}$, where $\tilde{g}, \tilde{h} \in \mathcal{I}_r$, there exists a $v' \in \mathbb{F}[\mathbf{x}]$ such that f can be rewritten as

$$f = \widehat{g} \cdot \widehat{h} + v' \cdot \mathcal{P}_r. \tag{5}$$

For the sake of contradiction, suppose $\widehat{g}$ and $\widehat{h}$ depend on a common variable, say x_ℓ. Write $\widehat{g} = \widehat{g}_{d_1} x_\ell^{d_1} + \cdots + \widehat{g}_1 x_\ell + \widehat{g}_0$ and $\widehat{h} = \widehat{h}_{d_2} x_\ell^{d_2} + \cdots + \widehat{h}_1 x_\ell + \widehat{h}_0$, where $d_1, d_2 \geq 1$, $\widehat{g}_{d_1} \not\equiv 0$, $\widehat{h}_{d_2} \not\equiv 0$ and for each $0 \leq i \leq d_1$, $0 \leq j \leq d_2$, $\widehat{g}_i, \widehat{h}_j \in \mathbb{F}[\mathbf{x} \setminus \{x_\ell\}]$. Then, we can rewrite Equation (5) as

$$f = \left(\sum_{i \in [d_1]} \widehat{g}_i \cdot x_\ell^i + \widehat{g}_0 \right) \cdot \left(\sum_{j \in [d_2]} \widehat{h}_j \cdot x_\ell^j + \widehat{h}_0 \right) + v' \cdot \mathcal{P}_r. \tag{6}$$

Note that the leading term on R.H.S. is $\widehat{g}_{d_1} \cdot \widehat{h}_{d_2} \cdot x_\ell^{d_1 + d_2}$, where $d_1 + d_2 \geq 2$. Since f is multilinear, this term must get cancelled by some term in $v' \mathcal{P}_r$ and hence it belongs to the ideal $\mathcal{I}_r$. First assume that x_ℓ appears in the monomial $\mathcal{P}_r$. Then, we get that $\frac{\mathcal{P}_r}{x_\ell}$ must divide the product $\widehat{g}_{d_1} \cdot \widehat{h}_{d_2}$. First, suppose $\frac{\mathcal{P}_r}{x_\ell}$ divides $\widehat{g}_{d_1}$. As $d_1 \geq 1$, we get that $\mathcal{P}_r$ divides $\widehat{g}_\ell \cdot x_\ell^{d_1}$. In other words, $\widehat{g}$ contains a monomial divisible by $\mathcal{P}_r$ but this contradicts the fact that $\widehat{g} = g \pmod{\mathcal{I}_r}$ must not have any term divisible by $\mathcal{P}_r$. Similarly, the case when $\frac{\mathcal{P}_r}{x_\ell}$ divides $\widehat{h}_{d_2}$ leads to a contradiction. Thus, we come to the case when $\frac{\mathcal{P}_r}{x_\ell}$ divides the product $\widehat{g}_{d_1} \cdot \widehat{h}_{d_2}$ but neither of them alone. Since $\frac{\mathcal{P}_r}{x_\ell}$ does not divide $\widehat{g}_{d_1}$, there exists a variable x_t, where $t \in [r] \setminus \{\ell\}$ such that x_t does not divide $\widehat{g}_{d_1}$. But since $\frac{\mathcal{P}_r}{x_\ell}$ divides $(\widehat{g}_{d_1} \cdot \widehat{h}_{d_2})$, x_t must divide $\widehat{h}_{d_2}$. Similarly, as $\frac{\mathcal{P}_r}{x_\ell}$ does not divide $\widehat{h}_{d_2}$, there exists another variable x_s, where

$s \in [r] \setminus \{\ell, t\}$ such that x_s does not divide $\widehat{h}_{d_2}$ but x_s divides $\widehat{g}_{d_1}$. Since $r \geq 3$, we indeed have sufficiently many variables for this to happen. Observe that

$$\begin{aligned} f|_{x_t=0} &= \widehat{g}|_{x_t=0} \cdot \widehat{h}|_{x_t=0} \\ f|_{x_t=0} &= (\widehat{g}_{d_1}|_{x_t=0} \cdot x_\ell^{d_1} + \ldots + \widehat{g}_0|_{x_t=0}) \cdot (\widehat{h}_{d_2}|_{x_t=0} \cdot x_\ell^{d_2} + \ldots + \widehat{h}_0|_{x_t=0}). \end{aligned} \tag{7}$$

Since f is $\mathbf{0}$-justified, $f|_{x_t=0} \not\equiv 0$. Although we know that x_t divides $\widehat{h}_{d_2}$, we note that x_t does not divide $\widehat{h}$, otherwise R.H.S. in Equation (7) becomes 0, while L.H.S. is non-zero. Then from Equation (7), we deduce that $\deg_{x_\ell}(\widehat{g}|_{x_t=0} \cdot \widehat{h}|_{x_t=0}) \geq d_1 \geq 1$. This implies that $x_\ell \in \mathrm{var}(f)$ but since f is multilinear, $\deg_{x_\ell}(f|_{x_t=0}) = 1$ and hence $d_1 = 1$. Arguing similarly for the variable x_s, we get that $d_2 = 1$. Thus, we can rewrite Equation (6) as:

$$f = (\widehat{g}_1 \cdot x_\ell + \widehat{g}_0) \cdot (\widehat{h}_1 \cdot x_\ell + \widehat{h}_0) + v' \cdot \mathcal{P}_r. \tag{8}$$

Now consider the polynomial $f|_{x_t=0,x_s=0}$. Since $d_1 = d_2 = 1$, x_t and x_s divide $\widehat{h}_1$ and $\widehat{g}_1$, respectively. We observe that

$$f|_{x_t=0,x_s=0} = \widehat{g}_0|_{x_t=0,x_s=0} \cdot \widehat{h}_0|_{x_t=0,x_s=0}.$$

Note that since f is $\mathbf{0}$-justified, L.H.S. depends on x_ℓ while R.H.S. does not, as $\widehat{g}_0, \widehat{h}_0$ were x_ℓ-free. This is a contradiction.

The case when x_ℓ does not appear in the monomial $\mathcal{P}_r$ is proved similarly. There, instead of $\frac{\mathcal{P}_r}{x_\ell}$, we get that $\mathcal{P}_r$ divides the product $\widehat{g}_{d_1} \cdot \widehat{h}_{d_2}$ but does not divide $\widehat{g}_{d_1}$ or $\widehat{h}_{d_2}$. Then there exist distinct $t, s \in [r]$ such that x_t does not divide $\widehat{g}_{d_1}$ but divides $\widehat{h}_{d_2}$ and x_s does not divide $\widehat{h}_{d_2}$ but divides $\widehat{g}_{d_1}$. Rest of the argument is exactly the same. Thus, we show that $\widehat{g}$ and $\widehat{h}$ cannot depend on any common variable and are therefore variable disjoint. □

We shall now use Lemma 5.4 to prove the following claim that will be used inside the proof of Theorem 5.6. In this claim, for a polynomial f, we work with the ideal $\mathcal{I}_{m+1}$ and express f as $f = \tilde{f} + \widehat{f}$, where $\widehat{f} = f \pmod{\mathcal{I}_{m+1}}$.

Claim 5.5. *Let $n \geq 3$ be a natural number and let $A, B, R \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ be multilinear polynomials that satisfy properties 1, 2 in Theorem 5.6. Let $H_1 \triangleq A - B$, $H_2 \triangleq A + B$ and*

$$F = H_1 \cdot H_2 - R = v \cdot \mathcal{P}_{m+1}.$$

For each $i \in \{1, 2\}$, let $J_i = \mathrm{var}(\widehat{H}_i)$ and $I_i = [n] \setminus J_i$. Then $J_1 \cap J_2 = \phi$, $H_1 \sim R|_{\mathbf{x}_{I_1}=\mathbf{0}_{I_1}}$ and $H_2 \sim R|_{\mathbf{x}_{I_2}=\mathbf{0}_{I_2}}$.

Proof. Similar to (5), we can also write

$$\widehat{H}_1 \cdot \widehat{H}_2 - R = v' \cdot \mathcal{P}_{m+1}. \tag{9}$$

Then from Lemma 5.4, we get that $\widehat{H}_1, \widehat{H}_2$ are variable disjoint. Therefore, $J_1 \cap J_2 = \phi$. Then note that $J_2 \subseteq I_1$. Consider the substitution $\mathbf{x}_{I_1} = \mathbf{0}_{I_1}$ in (9). We get

$$\alpha \cdot \widehat{H}_1 - R|_{\mathbf{x}_{I_1}=\mathbf{0}_{I_1}} = v'' \cdot \mathcal{P}_{m+1}, \tag{10}$$

for some $v'' \in \mathbb{F}[\mathbf{x}]$, where $\alpha = \widehat{H}_2|_{\mathbf{x}_{I_1}=\mathbf{0}_{I_1}} \in \mathbb{F}$ and v'' may or may not be zero. For example, if $[m+1] \cap I_1 \neq \phi$, then $v'' \equiv 0$.

Note that if $\alpha = 0$, then

$$-R|_{\mathbf{x}_{I_1}=\mathbf{0}_{I_1}} = v'' \cdot \mathcal{P}_{m+1} \implies R|_{\mathbf{x}_{I_1}=\mathbf{0}_{I_1}} \in \mathcal{I}_{m+1}.$$

But R is $\mathbf{0}$-justified (property 1 in Theorem 5.6), thus $R|_{\mathbf{x}_{I_1}=\mathbf{0}_{I_1}} \notin \mathcal{I}_{m+1}$, which gives a contradiction. Hence, $\alpha \neq 0$. Now add $\alpha \cdot \tilde{H}_1$ on both sides of (10). Since $H_1 = \tilde{H}_1 + \widehat{H}_1$ and $\tilde{H}_1 \in \mathcal{I}_{m+1}$, we get

$$\begin{aligned} \alpha \cdot \tilde{H}_1 + \alpha \cdot H_1 - R|_{\mathbf{x}_{I_1}=\mathbf{0}_{I_1}} &= \alpha \cdot \tilde{H}_1 + v'' \cdot \mathcal{P}_{m+1} \\ \alpha \cdot H_1 - R|_{\mathbf{x}_{I_1}=\mathbf{0}_{I_1}} &= v''' \cdot \mathcal{P}_{m+1} \\ \alpha \cdot A - \alpha \cdot B - R|_{\mathbf{x}_{I_1}=\mathbf{0}_{I_1}} &= v''' \cdot \mathcal{P}_{m+1}, \end{aligned}$$

for some $v''' \in \mathbb{F}[\mathbf{x}]$. By our hypothesis (property 2 in Theorem 5.6), we know that the set $\{\alpha \cdot A, -\alpha \cdot B, -R|_{\mathbf{x}_{I_1}=\mathbf{0}_{I_1}}\}$ is m-hard (also $(m+1)$-hard). Therefore, from the equation above, we deduce that $v''' \equiv 0$. Hence, $\alpha \cdot H_1 = R|_{\mathbf{x}_{I_1}=\mathbf{0}_{I_1}}$, or equivalently $H_1 \sim R|_{\mathbf{x}_{I_1}=\mathbf{0}_{I_1}}$. Similarly, we also get $H_2 \sim R|_{\mathbf{x}_{I_2}=\mathbf{0}_{I_2}}$. □

5.2 The Hardness of Representation Theorem and PIT

The theorem below is the main technical result of this section. Instead of talking about a particular class like ROF, we state it more generally for possible future use. The theorem below essentially lifts hardness of representation for a set of three polynomials to the set of their (arbitrary) powers. Recall Definition 2.35 for a m-hard set of polynomials.

Theorem 5.6 (Hardness of representation for $A^{e_1} - B^{e_2} - R^{e_3}$)**.** *Let $n, m \in \mathbb{N}, m \geq 2$ and $\mathbb{F}$ be a field. Suppose $A, B, R \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ are multilinear polynomials which satisfy the following properties:*

1. *$A, B,$ and R are $\mathbf{0}$-justified.*
2. *For every $J_1, J_2, J_3 \subseteq [n]$ and $\alpha_1, \alpha_2, \alpha_3 \in \mathbb{F}$, the set of polynomials $\{\alpha_1 \cdot A|_{\mathbf{x}_{J_1}=\mathbf{0}_{J_1}},\ \alpha_2 \cdot B|_{\mathbf{x}_{J_2}=\mathbf{0}_{J_2}},\ \alpha_3 \cdot R|_{\mathbf{x}_{J_3}=\mathbf{0}_{J_3}}\}$ is m-hard.*
3. *For any disjoint sets $J_1, J_2 \subseteq [n]$ and for every $\alpha, \beta \in \mathbb{F}$, the set of polynomials $\{\alpha \cdot R|_{\mathbf{x}_{I_1}=\mathbf{0}_{I_1}} \cdot R|_{\mathbf{x}_{I_2}=\mathbf{0}_{I_2}},\ \beta \cdot R\}$ is m-hard, where $I_1 = [n] \setminus J_1$ and $I_2 = [n] \setminus J_2$.*

Let $e_1, e_2, e_3 \in \mathbb{N}$ such that $e_1 \geq e_2 \geq e_3$. Then, the set $\{A^{e_1}, -B^{e_2}, -R^{e_3}\}$ is $(m+1)$-hard.

Proof. Let $F \overset{\Delta}{=} A^{e_1} - B^{e_2} - R^{e_3}$. To prove that $\{A^{e_1}, -B^{e_2}, -R^{e_3}\}$ is $(m+1)$-hard, either we have to show that $F \equiv 0$ or for every subset $J \subseteq [n], |J| = m+1$, the monomial $\prod_{j \in J} x_j$ does not divide F. If $F \equiv 0$, there is nothing to prove. If $n < m+1$ then $F \equiv 0$. Therefore, we can assume without loss of generality that $n \geq m+1$ and $F \not\equiv 0$. Assume for the sake of contradiction that there exists a set $J \subseteq [n], |J| = m+1$ such that $\prod_{j \in J} x_j$

divides F. Without loss of generality, let $J = [m+1]$, which implies that the monomial $\mathcal{P}_{m+1} \triangleq x_1 \cdots x_{m+1}$ divides F. In other words, $F \in \mathcal{I}_{m+1}$, where $\mathcal{I}_{m+1}$ is the ideal in $\mathbb{F}[x_1, x_2, \ldots, x_n]$ generated by $\mathcal{P}_{m+1}$. For this proof, we can assume without loss of generality that $\mathbb{F} = \overline{\mathbb{F}}$. This is so because if the monomial $\mathcal{P}_{m+1}$ divides F over the field $\mathbb{F}$ then it also divides it over $\overline{\mathbb{F}}$. As $F \in \mathcal{I}_{m+1}$, there exists a non-zero $v \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ such that

$$F = v \cdot \mathcal{P}_{m+1}. \tag{11}$$

If $A, B, R \in \mathbb{F}$ then we immediately get a contradiction. So, we assume without loss of generality that A is non-constant (otherwise, B and R are also constants). Now, we analyze the situation in the following cases.

- **Case 1.** $e_1 > e_2$: Let $j \in [m+1]$ be such that $\text{var}(A) \setminus \{j\} \neq \emptyset$. Since $m+1 \geq 2$, such a j always exists. Then, Equation (11) implies that

  $$F|_{x_j=0} = (A|_{x_j=0})^{e_1} - (B|_{x_j=0})^{e_2} - (R|_{x_j=0})^{e_3} \equiv 0.$$

  As A is **0**-justified, non-constant, and $\text{var}(A) \setminus \{j\} \neq \emptyset$, we get that $A|_{x_j=0}$ is a non-constant polynomial. Since $A|_{x_j=0}, B|_{x_j=0}, R|_{x_j=0}$ are multilinear, $A|_{x_j=0}$ is non-constant, and $e_1 > e_2$, $F|_{x_j=0} \not\equiv 0$. This means that F is not divisible by x_j, which contradicts the assumption that $F \in \mathcal{I}_{m+1}$.

- **Case 2.** $e_1 = e_2 \geq e_3$: Fix $e = e_1, d = e_3$. Then,

  $$F = A^e - B^e - R^d. \tag{12}$$

  For the further discussion, we need an e-th primitive root of unity ω in the field $\mathbb{F}$ (see Definition 2.14). As $\mathbb{F}$ is algebraically closed, Fact 2.15 tells us that if $p \triangleq \text{char}(\mathbb{F})$ does not divide e then ω is always present in $\mathbb{F}$. If $p = 0$ then ω exists. Suppose p is a prime number and p divides e. Then, there exists an $e' \in \mathbb{N}$ such that $e = e' \cdot p$. As $p = \text{char}(\mathbb{F})$, Equations (11) and (12) imply that

  $$(A^{e'} - B^{e'})^p - R^d = v \cdot \mathcal{P}_{m+1}. \tag{13}$$

  First, suppose that R is non-constant. Clearly, there exists a $j \in [m+1]$ such that $\text{var}(R) \setminus \{j\} \neq \emptyset$. It follows from Equation (13) that

  $$((A|_{x_j=0})^{e'} - (B|_{x_j=0})^{e'})^p = (R|_{x_j=0})^d.$$

  As $\text{var}(R) \setminus \{j\} \neq \emptyset$, $R|_{x_j=0}$ is a non-constant multilinear polynomial. Then, the equation above implies that p divides d.

  Now, suppose that $R \in \mathbb{F}$. As $\mathbb{F}$ is algebraically closed, we know that $\alpha \triangleq R^{\frac{1}{p}}$ is present in $\mathbb{F}$. Then, $R^d = (\alpha)^{d \cdot p}$. In this case, without loss of generality, we can replace R with α. This is so because observe that Properties 1, 2, and 3 of A, B, R remain intact if R is constant and we replace it with any other constant. This implies that $\{A^e, -B^e, -R^d\}$ is $(m+1)$-hard if and only if $\{A^e, -B^e, -\alpha^{d \cdot p}\}$ is $(m+1)$-hard.

Thus, in both the cases discussed above i.e., $R \in \mathbb{F}$ and R is non-constant, there exists a $d' \in \mathbb{N}$ such that $d = d' \cdot p$. Then, again using the fact that $p = \operatorname{char}(\mathbb{F})$, it follows from Equation (11) that

$$F = (F')^p = \upsilon \cdot \mathcal{P}_{m+1}.$$

As $F \in \mathcal{I}_{m+1}$, Observation 2.9 implies that $F' \in \mathcal{I}_{m+1}$. Thus, we can work with F' instead of F. This argument allows us to assume without loss of generality that p does not divide e. Hence, by Fact 2.15, we get that an e-th primitive root of unity ω is present in $\mathbb{F}$. Then, on substituting $x = A$ and $y = B$ in $x^e - y^e$ in Observation 2.16, we get the following useful factorization of $A^e - B^e$.

$$A^e - B^e = \prod_{\ell \in [e]} (A - \omega^\ell B). \tag{14}$$

This factorization would be immensely helpful for further analysis. We first assume that $e = 1$. As $d \leq e = 1$, Equation (12) implies that $F = A - B - \beta R$, where $\beta \in \{0, 1\}$. It follows from Property 2 that the set $\{A, -B, -\beta R\}$ is m-hard. Then Definition 2.35 implies that F can not be divisible by any multilinear monomial having support m. This contradicts our assumption that $F \in \mathcal{I}_{m+1}$.

Henceforth, we assume that $e \geq 2$. Now, we analyse this case in the following sub-cases.

- **Sub-case 2.a. R is a field constant**: Suppose $R \equiv 0$. It follows from Equations (12) and (14) that for every $j \in [n]$, there exists an $\ell_j \in [e]$ such that $A|_{x_j=0} = \omega^{\ell_j} B|_{x_j=0}$. As A is $\mathbf{0}$-justified and $n \geq m+1$, there exists a $J \subseteq [n], |J| = m$ such that for every $j \in J, A|_{x_j=0} \not\equiv 0$. Since $m \geq 2$, Claim 5.1 implies $A \sim B$. Thus, there exists an $\alpha \in \mathbb{F}$ such that $F = \alpha \cdot A^e$. Now, it follows from Observation 2.9 that $A \in \mathcal{I}_{m+1}$. As $n \geq m+1$, we get from Definition 2.35 that $\{A\}$ is not $(m+1)$-hard. On the other hand, as $\{A\}$ is m-hard by assumption (see Property 2), observe that it is also $(m+1)$-hard. This is a contradiction.

  Now, suppose $R \in \mathbb{F} \setminus \{0\}$. Let $j \in [m+1]$ be such that $\operatorname{var}(A) \setminus \{j\} \neq \emptyset$. It follows from Equations (11) and (12) that

  $$(A|_{x_j=0})^e - (B|_{x_j=0})^e = (R|_{x_j=0})^d.$$

  Since $R \in \mathbb{F} \setminus \{0\}$, Claim 5.2 implies that $A|_{x_j=0}, B|_{x_j=0} \in \mathbb{F}$. But this can not happen as $\operatorname{var}(A) \setminus \{j\} \neq \emptyset$, A is non-constant and $\mathbf{0}$-justified. Thus, $F|_{x_j=0} \not\equiv 0$, which means that x_j does not divide F and hence F is not in $\mathcal{I}_{m+1}$. This is a contradiction.

- **Sub-case 2.b.** $d \geq 2$: Let $J \subseteq [m+1]$ such that $|J| = m$ and for every $j \in J, \operatorname{var}(R) \setminus \{j\} \neq \emptyset$. Let $j \in J$ be arbitrary. Since R is $\mathbf{0}$-justified and non-constant, the restricted polynomial $R|_{x_j=0}$ is non-constant. Then, Equations (11) and (12) imply

  $$(A|_{x_j=0})^e - (B|_{x_j=0})^e = (R|_{x_j=0})^d.$$

It follows from Point 1 of Claim 5.3 that $d = e$ and for every $j \in J$, $A|_{x_j=0} \sim B|_{x_j=0} \sim R|_{x_j=0}$. Since $|J| = m \geq 2$, it is not difficult to see from Claim 5.1 that $A \sim B \sim R$. Thus, there exists an $\alpha \in \mathbb{F}$ such that $F = \alpha \cdot A^e$. Since, by assumption, $F \in \mathcal{I}_{m+1}$, Observation 2.9 implies that $A \in \mathcal{I}_{m+1}$. But this can not happen as A is m-hard, which implies that A can not be divisible by any multilinear monomial of support m. Thus, we get a contradiction.

- **Sub-case 2.c.** $d = 1$: From Claim 5.3, point 2, we deduce that $e = 2$. Then from Equations (11), (12), and (14), we have the following scenario:

  $$F = (A - B)(A + B) - R = v \cdot \mathcal{P}_{m+1}.$$

  Let $H_1 = A - B$ and $H_2 = A + B$. Then, we can write $R = H_1 \cdot H_2 + v' \cdot \mathcal{P}_{m+1}$, where $v' = -v$. Since $n \geq m + 1 \geq 3$, by Lemma 5.4, $\widehat{H}_1, \widehat{H}_2$ are variable disjoint. Moreover Claim 5.5 shows that for disjoint sets $J_1 = \text{var}(\widehat{H}_1), J_2 = \text{var}(\widehat{H}_2)$, we have $H_1 \sim R|_{\mathbf{x}_{I_1}=\mathbf{0}_{I_1}}$ and $H_2 \sim R|_{\mathbf{x}_{I_2}=\mathbf{0}_{I_2}}$, where $I_1 = [n] \setminus J_1$ and $I_2 = [n] \setminus J_2$. By Point 3 in our hypothesis, we deduce that the set $\{H_1H_2,\ -R\}$ is m-hard and hence also $(m + 1)$-hard. This contradicts our assumption that $F = H_1H_2 - R \in \mathcal{I}_{m+1}$. □

In Theorem 5.6, it was convenient to work with the set $\{A^{e_1}, -B^{e_2}, -R^{e_3}\}$, as we could exploit the factorization of $A^e - B^e$ in Equation (14). But as shown below, we can also drop the $-$ signs and prove that the set $\{A^{e_1}, B^{e_2}, R^{e_3}\}$ is m-hard provided A, B, R satisfy the hypothesis of Theorem 5.6.

Corollary 5.7 (Hardness of representation for $A^{e_1} + B^{e_2} + R^{e_3}$)**.** *Let A, B, and R be the multilinear polynomials given in Theorem 5.6 and m be the parameter mentioned in Theorem 5.6. Let $e_1, e_2, e_3 \in \mathbb{N}$ such that $e_1 \geq e_2 \geq e_3$*[5]*. Then, the set $\{A^{e_1}, B^{e_2}, R^{e_3}\}$ is $(m+1)$-hard.*

Proof. Let $F \triangleq A^{e_1} + B^{e_2} + R^{e_3}$. As argued in the proof of Theorem 5.6, we can assume without loss of generality that $\mathbb{F} = \overline{\mathbb{F}}$. Let $\alpha, \beta \in \overline{\mathbb{F}}$ be roots of the univariate polynomials $y^{e_2} + 1, y^{e_3} + 1 \in \mathbb{F}[y]$ respectively. Observe that

$$F = A^{e_1} - (\alpha \cdot B)^{e_2} - (\beta \cdot R)^{e_3}.$$

Since A, B, and R satisfy Properties 1,2, and 3 given in Theorem 5.6, it is easy to see that these properties are also satisfied by $A, \alpha \cdot B, \beta \cdot R$. Now it follows from Theorem 5.6 that the set $\{A^{e_1}, -(\alpha \cdot B)^{e_2}, -(\beta \cdot R)^{e_3}\}$ is $(m + 1)$-hard. □

Using the hardness of representation proved above, we can now hit any polynomial of the form $A^{e_1} + B^{e_2} + R^{e_3}$, provided that A, B, R satisfy the hypothesis of Theorem 5.6.

[5] We have this condition because we require only R to satisfy Property 3 in Theorem 5.6. If A, B also satisfy this property then we can drop the restriction $e_1 \geq e_2 \geq e_3$, which is indeed the case for ROFs and $\mathcal{C}_k$ as shown in Section 5.3. More generally, given the structures of ROFs and $\mathcal{C}_k$, our hardness of representation result actually extends to any sum of three powers of ROFs and $\mathcal{C}_k$.

Corollary 5.8 (Generator for $A^{e_1}+B^{e_2}+R^{e_3}$)**.** *Let $A, B, R \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ be the polynomials given in Theorem 5.6 and m be the parameter mentioned in Theorem 5.6. Suppose $F \triangleq A^{e_1}+B^{e_2}+R^{e_3}$, where $e_1, e_2, e_3 \in \mathbb{N}, e_1 \geq e_2 \geq e_3$. Then, $F \equiv 0$ if and only if $F(G_{n,m}) \equiv 0$, where $G_{n,m}$ is described in Definition 2.30.*

This corollary immediately follows from Corollary 5.7 and Fact 2.37. We can apply Fact 2.37 here because of the following reason: For Fact 2.37, we need to argue that for every subset $I \subset [n]$, the set of restricted polynomials $\{(A|_{\mathbf{x}_I=\mathbf{0}_I})^{e_1}, (B|_{\mathbf{x}_I=\mathbf{0}_I})^{e_2}, (R|_{\mathbf{x}_I=\mathbf{0}_I})^{e_3}\}$ is $(m+1)$-hard. To show this, we invoke Corollary 5.7 by replacing A, B, and R with $A|_{\mathbf{x}_I=\mathbf{0}_I}, B|_{\mathbf{x}_I=\mathbf{0}_I}$, and $R|_{\mathbf{x}_I=\mathbf{0}_I}$ respectively. Observe that these restricted polynomials also satisfy Properties 1, 2, and 3. Thus, Corollary 5.7 implies that the set of restricted polynomials $\{(A|_{\mathbf{x}_I=\mathbf{0}_I})^{e_1}, (B|_{\mathbf{x}_I=\mathbf{0}_I})^{e_2}, (R|_{\mathbf{x}_I=\mathbf{0}_I})^{e_3}\}$ is $(m+1)$-hard.
Now, we are ready to prove the following result, which generalizes Theorems 2 and 3.

Theorem 5.9. *Let $n \in \mathbb{N}$ and let $\widetilde{A}, \widetilde{B}, \widetilde{R} \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ be multilinear polynomials. Suppose $F \triangleq \widetilde{A}^{e_1}+\widetilde{B}^{e_2}+\widetilde{R}^{e_3}$, where $e_1, e_2, e_3 \in \mathbb{N}, e_1 \geq e_2 \geq e_3$. Let m be the parameter mentioned in Theorem 5.6. Let $H : \mathbb{F}^t \to \mathbb{F}^n$ be a generator such that there exist an assignment $\mathbf{a} \in \operatorname{Im}(H)$ for which the polynomials $A \triangleq \widetilde{A}(\mathbf{x}+\mathbf{a}), B \triangleq \widetilde{B}(\mathbf{x}+\mathbf{a})$, and $R \triangleq \widetilde{R}(\mathbf{x}+\mathbf{a})$ satisfy Properties 1, 2, and 3 given in Theorem 5.6. Then, $F \equiv 0$ if and only if $F(H+G_{n,m}) \equiv 0$, where $G_{n,m}$ is described in Definition 2.30.*

Proof. Suppose F is non-zero. Let $F' = A^{e_1}+B^{e_1}+R^{e_3}$. Since $F' = F(\mathbf{x}+\mathbf{a})$, we also have $F' \not\equiv 0$. Since A, B, R satisfy Properties 1, 2, and 3 in Theorem 5.6, we deduce that $F'(G_{n,m}) \not\equiv 0$ from Corollary 5.8. As the assignment $\mathbf{a} \in \operatorname{Im}(H)$, $F'(G_{n,m}) \not\equiv 0$ implies that $F(H+G_{n,m}) \not\equiv 0$. □

5.3 Proofs of Theorem 2 and 3: PIT for $\sum^{[3]}\bigwedge$ ROF and $\sum^{[3]}\bigwedge\mathcal{C}_k$

Now, we are ready to prove our main results. Let us see them one by one.

Proof of Theorem 2. In this theorem, we give a blackbox PIT for the class $\sum^{[3]}\bigwedge$ ROF. Let $f \in \sum^{[3]}\bigwedge$ ROF be a polynomial of degree at most d. Then, there exist three ROPs $\widetilde{A}, \widetilde{B}, \widetilde{R}$ and $e_1, e_2, e_3 \in \mathbb{N}$ such that

$$f = \widetilde{A}^{e_1}+\widetilde{B}^{e_2}+\widetilde{R}^{e_3}.$$

It follows from Fact 3.3 that there exists an assignment $\mathbf{a} \in \operatorname{Im}(G_{n,1})$ (see Definition 2.30) such that $A \triangleq \widetilde{A}(\mathbf{x}+\mathbf{a}), B \triangleq \widetilde{B}(\mathbf{x}+\mathbf{a}), R \triangleq \widetilde{R}(\mathbf{x}+\mathbf{a})$ are $\mathbf{0}$-justified polynomials. This, along with Fact 3.8 and Observation 3.5 imply that A, B, and R satisfy Properties 1, 2, and 3 of Theorem 5.6 with $m = 9$. Note that since the class of ROFs is closed with respect to the product of variable disjoint formulae, (see Observation 3.5) we get that each of the three polynomials A, B, R satisfy Property 3. Thus, we can assume without loss of generality that $e_1 \geq e_2 \geq e_3$. Now, it follows from Theorem 5.9 and Fact 2.28 that we can determine in poly(n, d) time whether f is zero or not.

Proof of Theorem 3. In this theorem, we give a blackbox PIT for the class $\sum^{[3]} \bigwedge \mathcal{C}_k$. Recall that $\mathcal{C}_k$ is the class of multilinear read-k arithmetic formulae. Let $f \in \sum^{[3]} \bigwedge \mathcal{C}_k$ be a polynomial of degree at most d. Then, there exist three polynomials $\widetilde{A}, \widetilde{B}, \widetilde{R}$ computed by multilinear read-k arithmetic formulae and $e_1, e_2, e_3 \in \mathbb{N}$ such that

$$f = \widetilde{A}^{e_1} + \widetilde{B}^{e_2} + \widetilde{R}^{e_3}.$$

Fact 3.15 implies that there exists an $\mathbf{a} \in \mathrm{Im}(G_{n,\ell})$, where $\ell = m + 3k\lceil \log n \rceil$ and $m = (8k \cdot (k+1))^k$, such that $A \triangleq \widetilde{A}(\mathbf{x} + \mathbf{a}), B \triangleq \widetilde{B}(\mathbf{x} + \mathbf{a}), R \triangleq \widetilde{R}(\mathbf{x} + \mathbf{a})$ satisfy Properties 1, 2, and 3. Since $\mathcal{C}_k$ is closed with respect to the product of variable disjoint formulae, (see Observation 3.14) we get that each of the three polynomials A, B, R satisfy Property 3. Thus, we can assume without loss of generality that $e_1 \geq e_2 \geq e_3$. Now, Theorem 5.9 and Fact 2.28 imply that we can determine in $(nd)^{O(\log n)}$ time whether $f \equiv 0$ or not.

6 Discussion and Future Work

In this work, we give a polynomial-time blackbox PIT algorithm for the class $\sum^{[2]} \prod$ ROF. We improve upon a result of [MRS16], which gave a whitebox PIT algorithm for the same class. We also took a step forward in solving an open question in [MRS16]. An efficient deterministic PIT algorithm for the class $\sum^{[k]} \bigwedge$ ROF was listed as an open problem in [MRS16]. We give a polynomial-time deterministic blackbox PIT algorithm for $\sum^{[3]} \bigwedge$ ROF. In addition to these two results, we also give a quasi-polynomial-time deterministic blackbox PIT for $\sum^{[3]} \bigwedge \mathcal{C}_k$, where $\mathcal{C}_k$ is the class of multilinear read-k arithmetic formulae over a field $\mathbb{F}$. All our results work over any field. The common thread between these three results is the hardness of representation approach (see Section 2.5). Now we list some open questions.

- PIT for $\sum^{[3]} \prod$ ROF: Our PIT algorithm for $\sum^{[2]} \prod$ ROF crucially depends on the fact that the fan-in of the topmost + gate in the circuits of this class is exactly two. In particular, the resultant based approach used in our algorithm only works in the top fan-in equal to two regime. It is not clear how to lift the resultant-based approach to $\sum^{[3]} \prod$ ROF. Can we come up with some technique that not only yields efficient PIT algorithm for $\sum^{[3]} \prod$ ROF, but also has the potential to extend to PIT for $\sum^{[k]} \prod$ ROF, where k is a constant?

- PIT for $\sum^{[k]} \bigwedge$ ROF: An efficient PIT algorithm for this class would solve an open question given in [MRS16]. Our PIT algorithm for $\sum^{[3]} \bigwedge$ ROF is based on a hardness of representation theorem, which we prove for this class. Can we prove the hardness of representation for $\sum^{[k]} \bigwedge$ ROF? In this direction, we note the following conjecture.

  Conjecture 6.1. *Let $k, n \in \mathbb{N}$, and $A_1, \ldots, A_k \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ be $\mathbf{0}$-justified ROPs. Then, there exists a monotone function $\varphi : \mathbb{N} \to \mathbb{N}$ such that for any $e_1, \ldots, e_k \in \mathbb{N}$, the set $\{A_1^{e_1}, \ldots, A_k^{e_k}\}$ is $\varphi(k)$-hard (see Definition 2.35).*

We remark that this conjecture is true when every $e_i = 1$. In particular, [SV15] showed that for any constant k, the set $\{A_1, \ldots, A_k\}$ is $3k$-hard (see Fact 3.8). In addition, for the special case when the A_i-s are products of linear forms over the reals, it was shown in [SV15], based on a result of [SS11], that set $\{A_1^{e_1}, \ldots, A_k^{e_k}\}$ is $R_{\mathbb{R}}(k)$-hard (for arbitrary e_i-s) where $R_{\mathbb{R}}(k)$ is the so-called "Rank Bound over the reals". Finally, in [KS09] it was shown that $R_{\mathbb{R}}(k) = k^{O(k)}$ and improved to $R_{\mathbb{R}}(k) = O(k^2)$ in [SS13].

- PIT for $\sum^{[2]} \prod \mathcal{C}_k$: The approach used in the proof of Theorem 1 would immediately solve this problem, provided we are able to efficiently compute a common irreducibility preserving assignment (see Definition 2.4) of a set of multilinear read-k arithmetic formulae. We know that if we could efficiently hit all the commutators of these formulae then such an assignment can be computed efficiently (see Claim 2.23 in this regard). In case of ROFs, it turns out that a commutator of an ROF is a product of ROF (see Corollary 3.7). What can we say about the structure of commutators of multilinear bounded-read arithmetic formulae?

References

[Agr05] M. Agrawal. Proving lower bounds via pseudo-random generators. In *Proceedings of the 25th FSTTCS*, volume 3821 of *LNCS*, pages 92–105, 2005. 2

[AHK93] D. Angluin, L. Hellerstein, and M. Karpinski. Learning read-once formulas with queries. *J. ACM*, 40(1):185–210, jan 1993. 3

[AKS04] M. Agrawal, N. Kayal, and N. Saxena. Primes is in P. *Annals of Mathematics*, 160(2):781–793, 2004. 2

[ASS13] M. Agrawal, C. Saha, and N. Saxena. Quasi-polynomial hitting-set for set-depth-*delta* formulas. In *Proceedings of the 45th Annual ACM Symposium on Theory of Computing (STOC)*, pages 321–330, 2013. 6

[ASSS16] M. Agrawal, C. Saha, R. Saptharishi, and N. Saxena. Jacobian hits circuits: Hitting sets, lower bounds for depth-d occur-k formulas and depth-3 transcendence degree-k circuits. *SIAM J. Comput.*, 45(4):1533–1562, 2016. 3, 4

[AvMV15] M. Anderson, D. van Melkebeek, and I. Volkovich. Derandomizing polynomial identity testing for multilinear constant-read formulae. *Computational Complexity*, 24(4):695–776, 2015. 2, 3, 6, 15, 17, 20

[BB98] D. Bshouty and N. H. Bshouty. On interpolating arithmetic read-once formulas with exponentiation. *JCSS*, 56(1):112–124, 1998. 3

[BC98] N. H. Bshouty and R. Cleve. Interpolating arithmetic read-once formulas in parallel. *SIAM J. on Computing*, 27(2):401–413, 1998. 3

[BHH95a] N. H. Bshouty, T. R. Hancock, and L. Hellerstein. Learning arithmetic read-once formulas. *SIAM J. on Computing*, 24(4):706–735, 1995. 3, 9

[BHH95b] N. H. Bshouty, T. R. Hancock, and L. Hellerstein. Learning boolean read-once formulas with arbitrary symmetric and constant fan-in gates. *JCSS*, 50:521–542, 1995. 3

[BHH95c] N.H. Bshouty, T.R. Hancock, and L. Hellerstein. Learning boolean read-once formulas over generalized bases. *J. Comput. Syst. Sci.*, 50(3):521–542, jun 1995. 3

[BSV23] V. Bhargava, S. Saraf, and I. Volkovich. Linear independence, alternants and applications. In *STOC '23: 55th Annual ACM SIGACT Symposium on Theory of Computing, Orlando, Florida, June 20-23, 2023*. ACM, 2023. 3

[BV22] P. Bisht and I. Volkovich. On solving sparse polynomial factorization related problems. In *42nd IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS 2022, December 18-20, 2022,* *IIT Madras, Chennai, India*, volume 250 of *LIPIcs*, pages 10:1–10:22. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2022. 16, 23

[CLO15] D. A. Cox, J. Little, and D. O'Shea. *Ideals, varieties, and algorithms - an introduction to computational algebraic geometry and commutative algebra (4. ed.)*. Undergraduate texts in mathematics. Springer, 2015. 12

[DDS21] P. Dutta, P. Dwivedi, and N. Saxena. Deterministic identity testing paradigms for bounded top-fanin depth-4 circuits. In Valentine Kabanets, editor, *36th Computational Complexity Conference, CCC 2021, July 20-23, 2021, Toronto, Ontario, Canada (Virtual Conference)*, volume 200 of *LIPIcs*, pages 11:1–11:27. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021. 3

[DL78] R. A. DeMillo and R. J. Lipton. A probabilistic remark on algebraic program testing. *Inf. Process. Lett.*, 7(4):193–195, 1978. 2

[DS07] Z. Dvir and A. Shpilka. Locally decodable codes with 2 queries and polynomial identity testing for depth 3 circuits. *SIAM J. on Computing*, 36(5):1404–1434, 2007. 3

[For15] M. A. Forbes. Deterministic divisibility testing via shifted partial derivatives. In *FOCS*, 2015. 6

[FS13] M. A. Forbes and A. Shpilka. Explicit noether normalization for simultaneous conjugation via polynomial identity testing. In *APPROX-RANDOM*, pages 527–542, 2013. 6

[FSS14] M. A. Forbes, R. Saptharishi, and A. Shpilka. Pseudorandomness for multilinear read-once algebraic branching programs, in any order. In *Proceedings of the 46th Annual ACM Symposium on Theory of Computing (STOC)*, pages 867–875, 2014. Full version at https://eccc.weizmann.ac.il/report/2013/132. 6, 15, 16

[GCL92] K. O. Geddes, S. R. Czapor, and G. Labahn. *Algorithms for computer algebra.* Kluwer, 1992. 12

[GG99] J. von zur Gathen and J. Gerhard. *Modern computer algebra.* Cambridge University Press, 1999. 11, 12

[GKS16] R. Gurjar, A. Korwar, and N. Saxena. Identity testing for constant-width, and commutative, read-once oblivious abps. In *31st Conference on Computational Complexity, CCC*, pages 29:1–29:16, 2016. 6

[GKST15] R. Gurjar, A. Korwar, N. Saxena, and N. Thierauf. Deterministic identity testing for sum of read-once oblivious arithmetic branching programs. In *30th Conference on Computational Complexity, CCC*, pages 323–346, 2015. 6

[GST23] N. Gupta, C. Saha, and B. Thankey. Equivalence test for read-once arithmetic formulas. In Nikhil Bansal and Viswanath Nagarajan, editors, *Proceedings of the 2023 ACM-SIAM Symposium on Discrete Algorithms, SODA 2023, Florence, Italy, January 22-25, 2023*, pages 4205–4272. SIAM, 2023. 4

[HH91] T. R. Hancock and L. Hellerstein. Learning read-once formulas over fields and extended bases. In *Proceedings of the 4th Annual Workshop on Computational Learning Theory (COLT)*, pages 326–336, 1991. 3, 9

[HS80] J. Heintz and C. P. Schnorr. Testing polynomials which are easy to compute (extended abstract). In *Proceedings of the 12th Annual ACM Symposium on Theory of Computing (STOC)*, pages 262–272, 1980. 2

[KI03] V. Kabanets and R. Impagliazzo. Derandomizing polynomial identity tests means proving circuit lower bounds. In *Proceedings of the 35th Annual ACM Symposium on Theory of Computing (STOC)*, pages 355–364, 2003. 2

[KLN$^+$93] M. Karchmer, N. Linial, I. Newman, M. Saks, and A. Wigderson. Combinatorial characterization of read-once formulae. *Discrete Math.*, 114(1–3):275–282, April 1993. 3

[KMSV13] Z. S. Karnin, P. Mukhopadhyay, A. Shpilka, and I. Volkovich. Deterministic identity testing of depth 4 multilinear circuits with bounded top fan-in. *SIAM J. on Computing*, 42(6):2114–2131, 2013. 15, 16

[KS01] A. Klivans and D. Spielman. Randomness efficient identity testing of multivariate polynomials. In *Proceedings of the 33rd Annual ACM Symposium on Theory of Computing (STOC)*, pages 216–223, 2001. 3

[KS07] N. Kayal and N. Saxena. Polynomial identity testing for depth 3 circuits. *Computational Complexity*, 16(2):115–138, 2007. 3

[KS08] Z. S. Karnin and A. Shpilka. Deterministic black box polynomial identity testing of depth-3 arithmetic circuits with bounded top fan-in. In *Proceedings of the 23rd Annual IEEE Conference on Computational Complexity (CCC)*, pages 280–291, 2008. 3

[KS09] N. Kayal and S. Saraf. Blackbox polynomial identity testing for depth 3 circuits. In *Proceedings of the 50th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 198–207, 2009. Full version at https://eccc.weizmann.ac.il/report/2009/032. 3, 34

[KS19] M. Kumar and R. Saptharishi. Hardness-randomness tradeoffs for algebraic computation. *Bulletin of EATCS*, 3(129), 2019. 2

[Lov79] L. Lovasz. On determinants, matchings, and random algorithms. In L. Budach, editor, *Fundamentals of Computing Theory*. Akademia-Verlag, 1979. 2

[LST21] N. Limaye, S. Srinivasan, and S. Tavenas. Superpolynomial lower bounds against low-depth algebraic circuits. In *62nd IEEE Annual Symposium on Foundations of Computer Science, FOCS 2021, Denver, CO, USA, February 7-10, 2022*, pages 804–814. IEEE, 2021. 3

[MRS16] M. Mahajan, B.V.R. Rao, and K. Sreenivasaiah. Building above read-once polynomials: Identity testing and hardness of representation. *Algorithmica*, 76:890–909, 2016. 2, 3, 4, 5, 33

[MS21] D. Medini and A. Shpilka. Hitting sets and reconstruction for dense orbits in vp_{e} and $\Sigma\Pi\Sigma$ circuits. In Valentine Kabanets, editor, *36th Computational Complexity Conference, CCC 2021, July 20-23, 2021, Toronto, Ontario, Canada (Virtual Conference)*, volume 200 of *LIPIcs*, pages 19:1–19:27. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021. 3

[MV18] D. Minahan and I. Volkovich. Complete derandomization of identity testing and reconstruction of read-once formulas. *TOCT*, 10(3):10:1–10:11, 2018. 2, 3, 4, 7, 15, 18, 19

[Neu07] M. Neunhöffer, 2007. Lecture notes on finite fields - Module MT 5826, Chapter 4, Link - http://www.math.rwth-aachen.de/homes/Max.Neunhoeffer/Teaching/ff/ffchap4.pdf. 12

[PS21] S. Peleg and A. Shpilka. Polynomial time deterministic identity testing algorithm for $\Sigma^{[3]}\Pi\Sigma\Pi^{[2]}$ circuits via edelstein-kelly type theorem for quadratic polynomials. In Samir Khuller and Virginia Vassilevska Williams, editors, *STOC '21: 53rd Annual ACM SIGACT Symposium on Theory of Computing, Virtual Event, Italy, June 21-25, 2021*, pages 259–271. ACM, 2021. 3

[RR19] C. Ramya and B.V.R. Rao. Lower bounds for sum and sum of products of read-once formulas. *ACM Transactions on Computation Theory (TOCT)*, 11(2):1–27, 2019. 2

[Sch80] J. T. Schwartz. Fast probabilistic algorithms for verification of polynomial identities. *J. ACM*, 27(4):701–717, 1980. 2

[SS11] N. Saxena and C. Seshadhri. An almost optimal rank bound for depth-3 identities. *SIAM J. Comput.*, 40(1):200–224, 2011. 3, 34

[SS12] N. Saxena and C. Seshadhri. Blackbox identity testing for bounded top-fanin depth-3 circuits: The field doesn't matter. *SIAM J. Comput.*, 41(5):1285–1298, 2012. 3

[SS13] N. Saxena and C. Seshadhri. From sylvester-gallai configurations to rank bounds: Improved blackbox identity test for depth-3 circuits. *J. ACM*, 60(5):33, 2013. 3, 34

[ST21a] C. Saha and B. Thankey. Hitting sets for orbits of circuit classes and polynomial families. In Mary Wootters and Laura Sanità, editors, *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques, APPROX/RANDOM 2021, August 16-18, 2021, University of Washington, Seattle, Washington, USA (Virtual Conference)*, volume 207 of *LIPIcs*, pages 50:1–50:26. Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 2021. 3

[ST21b] A. Sinhababu and T. Thierauf. Factorization of polynomials given by arithmetic branching programs. *computational complexity*, 30(2):1–47, 2021. 6

[SV10] A. Shpilka and I. Volkovich. On the relation between polynomial identity testing and finding variable disjoint factors. In *Automata, Languages and Programming, 37th International Colloquium (ICALP)*, pages 408–419, 2010. Full version at https://eccc.weizmann.ac.il/report/2010/036. 13

[SV14] A. Shpilka and I. Volkovich. On reconstruction and testing of read-once formulas. *Theory of Computing*, 10:465–514, 2014. 3, 9, 13, 18

[SV15] A. Shpilka and I. Volkovich. Read-once polynomial identity testing. *Computational Complexity*, 24(3):477–532, 2015. 2, 3, 4, 6, 8, 9, 10, 13, 15, 16, 17, 18, 34, 40

[SV18] S. Saraf and I. Volkovich. Blackbox identity testing for depth-4 multilinear circuits. *Combinatorica*, 38(5):1205–1238, 2018. 3

[SY10] A. Shpilka and A. Yehudayoff. Arithmetic circuits: A survey of recent results and open questions. *Foundations and Trends in Theoretical Computer Science*, 5(3-4):207–388, 2010. 15, 42

[Vol15] I. Volkovich. Deterministically factoring sparse polynomials into multilinear factors and sums of univariate polynomials. In *APPROX-RANDOM*, pages 943–958, 2015. 15, 16, 23, 40

[Vol16] I. Volkovich. Characterizing arithmetic read-once formulae. *ACM Transactions on Computation Theory (ToCT)*, 8(1):2, 2016. 4

[Zip79] R. Zippel. Probabilistic algorithms for sparse polynomials. In *Proceedings of the International Symposium on Symbolic and Algebraic Computation*, pages 216–226, 1979. 2

A Missing proofs from Sections 2 and 3

For a generator, we can keep a variable untouched while applying the map on rest of the variables. This is formally called reviving a variable and is defined formally below. This operation will be needed for Section A.1.

Definition A.1 (Reviving, [Vol15])**.** *Let $f \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ be a polynomial and $\mathcal{G} : \mathbb{F}^t \to \mathbb{F}^n$ be a polynomial map such that $\mathcal{G} = (\mathcal{G}^1(\mathbf{w}), \ldots, \mathcal{G}^n(\mathbf{w}))$. Let $k \leq n$. Define $H_{n,1} : \mathbb{F}^{t+2} \to \mathbb{F}^n$ as $H_{n,1} \triangleq \mathcal{G}(\mathbf{w}) + G_{n,1}(y, z)$. Let $i \in [n]$. By Fact 2.31,*

$$f(H_{n,1})|_{y=\alpha_i, z=x_i-\mathcal{G}^i} = f(\mathcal{G}^1(\mathbf{w}), \ldots, \mathcal{G}^{i-1}(\mathbf{w}), x_i, \mathcal{G}^{i+1}(\mathbf{w}), \ldots, \mathcal{G}^n(\mathbf{w})),$$

where α_i is the i^{th} Lagrange constant.

The following fact from [SV15] gives the complete description of the structure of an ROP. This will be useful in Section A.3.

Fact A.2. (Lemma 3.3 of [SV15]) *Let $A \in \mathbb{F}[\mathbf{x}]$ be an ROP such that $|\mathrm{var}(A)| \geq 2$. Then, there exist non-constant variable disjoint ROPs $A_1, A_2 \in \mathbb{F}[\mathbf{x}]$ such that exactly one of the following is true.*

1. $A = A_1 + A_2$
2. $A = A_1 \cdot A_2 + \alpha$, *where* $\alpha \in \mathbb{F}$.

A.1 Proof of Fact 2.34

We are given a non-zero $f = A_1 \cdots A_m + B_1 \cdots B_r$. Then, we have two cases: either $A_1 \cdots A_m \sim B_1 \cdots B_r$ or $A_1 \cdots A_m \not\sim B_1 \cdots B_r$. In the former case, $f = \alpha \cdot A_1 \cdots B_m$ for some $\alpha \in \mathbb{F}$. Then, clearly $f(\mathcal{G}) \equiv 0$ if and only if $f \equiv 0$. Now, suppose $A_1 \cdots A_m \not\sim B_1 \cdots B_r$. We can assume that f is simple, that is, there is no common factor among any two polynomials A_j, B_k, $j \in [m], k \in [r]$. Otherwise we can simply take out the gcd, which can be hit by $\mathcal{G}$ as it is a product of polynomials from $\mathcal{C}$ (Observation 2.29). Now consider the irreducible factorization of LHS and RHS. Without loss of generality, there exist an irreducible factor u of some A_j, $j \in [m]$ such that u does not divide any B_k, $k \in [r]$. Let x_i be any variable in $\mathrm{var}(u)$. Then $\mathrm{Res}_{x_i}(u, B_k) \not\equiv 0$ for all $k \in [r]$. Since $\mathcal{G} : \mathbb{F}^t \to \mathbb{F}^n$ hits $\mathrm{Res}(\mathcal{C})$, we deduce that $\mathrm{Res}_{x_i}(u, B_k)(\mathcal{G}_{-i}) \not\equiv 0$, where $G_{-i} = (\mathcal{G}^1, \ldots, \mathcal{G}^{i-1}, \mathcal{G}^{i+1}, \mathcal{G}^n)$. Using definition of resultant, one can then show that $\mathrm{Res}_{x_i}(u(x_i, \mathcal{G}_{-i}), B_k(x_i, \mathcal{G}_{-i})) \not\equiv 0$. We need to show this for every $i \in [n]$, which would imply that u does not share a gcd with any B_k even after applying the generator map. This then proves that $A_1 \cdots A_m \not\sim B_1 \cdots B_r$ after the map, which shows that f remains non-zero after applying the map. To iterate over every variable x_i, $i \in [n]$ in a blackbox way, we do this additional step of composing $\mathcal{G}$ with $G_{n,1}$. Let $H_{n,1} = \mathcal{G} + G_{n,1}$. Then from Definition A.1, we deduce that for every $i \in [n]$, $u(H_{n,1})|_{y=\alpha_i, z=x_i-\mathcal{G}^i} = u(x_i, \mathcal{G}_{-i})$ and $B_k(H_{n,1})|_{y=\alpha_i, z=x_i-\mathcal{G}^i} = B_k(x_i, \mathcal{G}_{-i})$. Hence, $f(H_{n,1}) \not\equiv 0$. □

A.2 Proof of Fact 2.37

Let M be a minimal support monomial in A, i.e., for every monomial M' of A, $|\mathrm{supp}(M)| \leq |\mathrm{supp}(M')|$. We claim that $|\mathrm{supp}(M)| < m$. Suppose this is not true. Let $J := [n] \setminus \mathrm{supp}(M)$. Then, it is not difficult to see that $A|_{\mathbf{x}_J = \mathbf{0}_J}$ is non-zero. Since $A = A_1 + \cdots + A_k$, we get

$$A|_{\mathbf{x}_J=\mathbf{0}_J} = A_1|_{\mathbf{x}_J=\mathbf{0}_J} + \ldots + A_k|_{\mathbf{x}_J=\mathbf{0}_J} \not\equiv 0,$$

and $A|_{\mathbf{x}_J=\mathbf{0}_J}$ is divisible by the support at least m monomial $\prod_{j \in \mathrm{supp}(M)} x_j$. On the other hand, by assumption, the set $\{A_1|_{\mathbf{x}_J=\mathbf{0}_J}, \ldots, A_k|_{\mathbf{x}_J=\mathbf{0}_J}\}$ is m-hard. Since $A|_{\mathbf{x}_J=\mathbf{0}_J} \not\equiv 0$, it follows from Definition 2.35 that $A|_{\mathbf{x}_J=\mathbf{0}_J}$ can not be divisible by any monomial of support m. This is a contradiction. Thus, $|\mathrm{supp}(M)| < m$. As A has a monomial of support less than m, it follows from Fact 2.32 and Observation 2.33 that $A(G_{n,m-1}) \not\equiv 0$. □

A.3 Proof of Fact 3.9

For the sake of contradiction, suppose the set $\{A, B\}$ is not 3-hard. Then $A + B \not\equiv 0$ and there exists a set $J \subseteq [n], |J| = 3$ such that $A + B$ is divisible by the monomial $\prod_{j \in J} x_j$. Without loss of generality, let $J = \{1, 2, 3\}$. Since $A+B$ is multilinear, there exists a non-zero $R \in \mathbb{F}[\mathbf{x} \setminus \{x_1, x_2, x_3\}]$ such that

$$A + B = R \cdot x_1 x_2 x_3. \tag{15}$$

We get the above equation for restrictions of A, B. For the sake of simplicity, we still call these restricted polynomials A, B which are now $\mathbf{0}$-justified polynomials in $\mathbb{F}[x_1, x_2, x_3]$. First, suppose there exist $i, j \in [3]$, $i \neq j$ such that $\frac{\partial^2 A}{\partial x_i \partial x_j} \equiv 0$. Then from Equation (15),

$$\frac{\partial^2 B}{\partial x_i \partial x_j} = R \cdot \frac{x_1 x_2 x_3}{x_i x_j}.$$

Note that $\frac{\partial^2 B}{\partial x_i \partial x_j} \not\equiv 0$ as R.H.S. is non-zero. Moreover, it follows from Fact 3.4 that $\frac{\partial^2 B}{\partial x_i \partial x_j}$ is $\mathbf{0}$-justified since B is $\mathbf{0}$-justified. Since we have 3 variables, pick x_ℓ such that $\ell \notin \{i, j\}$ and fix x_ℓ to 0. Then R.H.S. of the above equation is 0 but L.H.S. is non-zero, which is a contradiction. Similarly the case, where $\frac{\partial^2 B}{\partial x_i \partial x_j} \equiv 0$ leads to a contradiction.

Now, we are in the case where for every $i, j \in [3]$, $i \neq j$, we have $\frac{\partial^2 A}{\partial x_i \partial x_j} \not\equiv 0$ and $\frac{\partial^2 B}{\partial x_i \partial x_j} \not\equiv 0$. In this case, every gate in the ROFs of A and B is a multiplication gate. This is because if there was an addition gate in A (similarly, B), it would be the first common gate for some pair of x_i, x_j variables, which implies $\frac{\partial^2 A}{\partial x_i \partial x_j} \equiv 0$ (respectively, $\frac{\partial^2 B}{\partial x_i \partial x_j} \equiv 0$). Therefore, by Fact A.2, we can write $A + B$ as $g_1 \cdot h_1 + g_2 \cdot h_2 + c$, where g_1, h_1 are variable disjoint and g_2, h_2 are variable disjoint and $c \in \mathbb{F}$. Then, without loss of generality, we can write (15) as

$$g_1 \cdot h_1 + c = g_2 \cdot h_2 + R \cdot x_1 x_2 x_3. \tag{16}$$

Since g_1, h_1 are variable disjoint, pick some $i \in \mathrm{var}(g_1) \setminus \mathrm{var}(h_1)$ and $j \in \mathrm{var}(h_1) \setminus \mathrm{var}(g_1)$. First note that $i, j \in \mathrm{var}(g_2 \cdot h_2)$. To see this, substitute x_ℓ to 0 for some $\ell \notin \{i, j\}$. Since

A was $\mathbf{0}$-justified, L.H.S. depends on x_i, x_j. Hence $g_2 \cdot h_2$ must also depend on x_i, x_j. Since g_2, h_2 are variable disjoint, without loss of generality, we can assume $i \in \mathrm{var}(g_2) \setminus \mathrm{var}(h_2)$ and $j \in \mathrm{var}(h_2) \setminus \mathrm{var}(g_2)$. Now apply the commutator Δ_{ij} on Equation (16). It follows from Definition 2.19 that

$$\frac{\partial g_1}{\partial x_i} \cdot \frac{\partial h_1}{\partial x_j} \cdot c = g_2|_{x_i=0} \cdot h_2|_{x_j=0} \cdot R \cdot \frac{x_1 x_2 x_3}{x_i x_j}.$$

Substitute x_ℓ to 0 above. Then R.H.S. is 0 but since A was $\mathbf{0}$-justified, L.H.S. is non-zero. This implies $c = 0$. But this is a contradiction, as now without the substitution L.H.S. above is zero while R.H.S. is a non-zero polynomial. This is because $g_2|_{x_i=0} \not\equiv 0$ and $h_2|_{x_j=0} \not\equiv 0$ as B was $\mathbf{0}$-justified. Hence $A + B$ is not divisible by any monomial of length 3. □

B Arithmetic circuits and formulas

In this section, we give a brief overview of the various algebraic models of computation discussed in this work. For a detailed exposition, we refer the reader to the excellent survey of [SY10].

An *arithmetic circuit* is defined as a directed acyclic graph, where input variables or field constants label the leaf nodes, while intermediate nodes are labelled with either + or ×. A '+' node adds all the polynomials on its incoming edges, while a × node multiplies. We have a single output node at the top and the circuit computes from bottom to top. The edges can also be labeled with field constants which get multiplied. An unlabelled edge can be thought to be labelled with the constant 1. The in-degree of a node is called it *fan-in* and out-degree is called *fan-out*. An arithmetic circuit has two important resource parameters: size and depth. *Size* of the circuit is size of the underlying graph, given by the number of edges and nodes. *Depth* of the circuit is the length of the longest path from some leaf node to the output node. *Degree* of the circuit is the maximum degree of a polynomial computed at any node in the circuit. VP is defined as the class of poly(n)-sized and poly(n)-degree arithmetic circuits. Without loss of generality, an arithmetic circuit is assumed to be an alternating layered graph, which alternates between a layer of addition and a layer of multiplication nodes. The class of *depth-2* $\Sigma\Pi$ *circuits* computes sparse polynomials. Depth-3 $\Sigma\Pi\Sigma$ circuits compute polynomials of the form $f = \sum_{i=1}^{k} \prod_{j=1}^{m} \ell_{ij}$, where l_{ij}'s are linear polynomials. Depth-4 $\Sigma\Pi\Sigma\Pi$ circuits compute polynomials of the form $f = \sum_{i=1}^{k} \prod_{j=1}^{m} f_{ij}$, where f_{ij}'s are sparse polynomials.

An *arithmetic formula* or simply formula in short is defined as an arithmetic circuit where every node has at most one outgoing edge. The underlying graph for a formula has a tree structure.