Deterministic Black-Box Identity Testing π-Ordered Algebraic Branching Programs*

Maurice Jansen[1], Youming Qiao[2], and Jayalal Sarma M.N.[3]

1 **School of Informatics, The University of Edinburgh,** maurice.julien.jansen@gmail.com
2 **Institute for Theoretical Computer Science, Tsinghua University,** jimmyqiao86@gmail.com
3 **Department of Computer Science and Engineering, Indian Institute of Technology Madras,** jayalal.sarma@gmail.com

Abstract

In this paper we study algebraic branching programs (ABPs) with restrictions on the order and the number of reads of variables in the program. An ABP is given by a layered directed acyclic graph with source s and sink t, whose edges are labeled by variables taken from the set $\{x_1, x_2, \ldots, x_n\}$ or field constants. It computes the sum of weights of all paths from s to t, where the weight of a path is defined as the product of edge-labels on the path. Given a permutation π of the n variables, for a π-*ordered* ABP (π-OABP), for any directed path p from s to t, a variable can appear at most once on p, and the order in which variables appear on p must respect π. One can think of OABPs as being the arithmetic analogue of ordered binary decision diagrams (OBDDs). We say an ABP A is of read r, if any variable appears at most r times in A.

Our main result pertains to the polynomial identity testing problem, i.e. the problem of deciding whether a given n-variate polynomial is identical to the zero polynomial or not. We prove that over any field $\mathbb{F}$, and in the black-box model, i.e. given only query access to the polynomial, read r π-OABP computable polynomials can be tested in DTIME$[2^{O(r \log r \cdot \log^2 n \log\log n)}]$. In case $\mathbb{F}$ is a finite field, the above time bound holds provided the identity testing algorithm is allowed to make queries to extension fields of $\mathbb{F}$. To establish this result, we combine some basic tools from algebraic geometry with ideas from derandomization in the Boolean domain.

Our next set of results investigates the computational limitations of OABPs. It is shown that any OABP computing the determinant or permanent requires size $\Omega(2^n/n)$ and read $\Omega(2^n/n^2)$. We give a multilinear polynomial p in $2n+1$ variables over some specifically selected field $\mathbb{G}$, such that any OABP computing p must read some variable at least 2^n times. We prove a strict separation for the computational power of read $(r-1)$ and read r OABPs. Namely, we show that the elementary symmetric polynomial of degree r in n variables can be computed by a size $O(rn)$ read r OABP, but not by a read $(r-1)$ OABP, for any $0 < 2r-1 \leq n$. Finally, we give an example of a polynomial p and two variables orders $\pi \neq \pi'$, such that p can be computed by a read-once π-OABP, but where any π'-OABP computing p must read some variable at least 2^n times.

Digital Object Identifier 10.4230/LIPIcs.FSTTCS.2010.296

1 Introduction

The polynomial identity testing problem (PIT) is the question of deciding, given an arithmetic circuit C with input variables $x_1, x_2 \ldots x_n$ over some field $\mathbb{F}$, whether the polynomial computed

* This work was supported in part by the National Natural Science Foundation of China Grant 60553001, 61073174, 61033001 and the National Basic Research Program of China Grant 2007CB807900, 2007CB807901.

© Maurice Jansen, Youming Qiao and Jayalal Sarma M.N.;
licensed under Creative Commons License NC-ND
IARCS Int'l Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2010).
Editors: Kamal Lodaya, Meena Mahajan; pp. 296–307

Leibniz International Proceedings in Informatics
Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

by C is identical to the zero polynomial in the ring $\mathbb{F}[x_1, x_2, \ldots x_n]$. Efficient algorithms for PIT are important both in theory and in practice. Randomized algorithms were given independently by Schwartz [17] and Zippel [22].

Finding *deterministic* algorithms for PIT plays a crucial role in computational complexity theory. Kabanets and Implagliazzo [11] showed that giving a deterministic subexponential time algorithm for PIT implies that either NEXP $\not\subseteq$ P/*poly*, or that the permanent has no *poly*-size arithmetic circuits. Agrawal [1] showed that giving a deterministic *black-box* algorithm for PIT yields an explicit multilinear polynomial that has no subexponential size arithmetic circuits. In [1] a program was outlined explaining how making progress towards the latter kind of algorithm for PIT has the potential of resolving Valiant's Hypothesis, which states that the algebraic complexity classes VP and VNP are distinct. For optimists certainly, the situation is tantalizing, as Agrawal and Vinay [2] showed that the black-box derandomization of PIT for only depth-4 circuits would yield a nearly complete derandomization for general arithmetic circuits. Recent progress on the PIT problem has been impressive. See [16] for a recent survey.

In this paper, we contribute to the above mentioned lower bounds program by considering black-box identity testing *ordered algebraic branching programs* (OABPs), which where introduced in [8]. Algebraic branching programs have computational power somewhere in between arithmetic formulas and circuits. Namely, they can efficiently simulate formulas via a construction by Valiant [21]. Furthermore, their computational power is easily seen to be equivalent to that of skew circuits. For skew circuits, which were introduced by Toda [20], multiplication gates are restricted to have one of their inputs to be a variable or field constant. OABPs can be thought of as being the arithmetic analogue of *ordered binary decision diagrams* (OBDDs), which were introduced by Bryant [4].

Some polynomials can be succinctly represented in the OABP model. For example, we show that the elementary symmetric polynomial of degree k in n variables can be elegantly described by a grid shaped OABP of size $O(kn)$. This can be done for any desired variable order π, and shows small OABPs have some real computing power. We think the OABP model has practical merit for polynomial representation, and being an analogue of the OBDD it should be properly investigated. As our lower bounds show, a succinct OABP-representation is not available for every polynomial. The situation is similar to what is well-known for OBDDs. In practice this may be outweighed by the fact that PIT can be solved efficiently for OABPs. Part of the popularity of OBDDs can be explained by the fact that identity testing (and hence equivalence testing) can be done efficiently for the model, as e.g. Raz and Shpilka [14] remarked.

In [14] a polynomial-time non-black-box algorithm was given for identity testing non-commutative formulas, and more generally non-commutative ABPs. Identity testing OABPs reduces to PIT for non-commutative ABPs, and hence can be done *non-black-box* in polynomial time. Namely, if we take an OABP A computing some polynomial f over commuting variables, and if we let f' be the evaluation of A, where we restrict the variables to be non-commuting, then it can be observed that $f \equiv 0 \Leftrightarrow f' \equiv 0$. Giving a *black-box* algorithm for testing non-commutative formulas and ABPs is currently a major open problem. Our main result implies that for any variable order π, we have a DTIME$[2^{O(polylog(n))}]$ black-box algorithm for testing OABPs with order π that have $polylog(n)$ many reads.

Let us mention the connection of our work to the problem of identity testing multilinear formulas raised by Raz [13]. Our results can be applied to black-box identity testing "ordered multilinear formulas" with few reads (say $polylog(n)$). The latter can be defined for any given variable order π, by requiring that for each multiplication gate $g = g_1 \times g_2$ in the

formula, variables in the subformula rooted at g_1 should either all be smaller or all be larger w.r.t. π than variables in the subformula rooted at g_2. By applying the construction of [21], judiciously to keep the order, a formula of this kind can be simulated by a π-OABP. This then gives another important special case of PIT for multilinear formulas for which a black-box algorithm is known. The other case being sum-of-k read-once formulas, which we elaborate on next.

Any arithmetic *read-once formula* (ROF) can be simulated by an OABP, since the construction of [21] mentioned before preserves the RO-property. Black-box Identity testing sum-of-k ROFs was studied in [19], and this was subsequently generalized to the sum-of-k read-once ABPs in [10]. These results suggest the difficulty of making generalizations in this area to models beyond read-once. For example, by [10] we have an $n^{O(\log n)}$ black-box test for sum-of-two read-once ABPs, but for testing a single read-twice ABP, currently nothing is known beyond brute-force methods. Our result is significant, in that the techniques apply to a model where the multiple reads take place within one *monolithic* ABP. This opens up a new thread of progress in the direction of identity testing unrestricted ABPs. We refer to [9] for a direct connection between this, and proving lower bounds for the *determinantal complexity* of explicit polynomials. The latter is what the separation of VP and VNP requires.

Another point of significance pertains to the techniques we use (on which we will elaborate more below). To obtain the main result, we combine basic tools from algebraic geometry with ideas from derandomization in the Boolean world (specifically, the pseudorandom construction of Impagliazzo, Nisan and Wigderson [7] for network algorithms). As far as we now, this kind of use of basic algebraic geometry is new to the PIT area. We hope our work stimulates more research in this direction.

1.1 Techniques

Towards the identity testing algorithm, first we show that, without increasing the number of reads, any π-OABP can be made π-*oblivious*. For the latter, all variables in a layer must be identical, and all occurences of a variable x_i appear in the same layer. Hence there is some variable order $x_{i_1}, x_{i_2}, \ldots, x_{i_n}$ in which the layers appear (possibly interleaved by constant layers), when going from the source to the sink. The next step is to construct a generator $\mathcal{G}(z)$ for π-oblivious ABPs. This is a mapping $\mathbb{F}^\ell \to \mathbb{F}^n$ such that for any $f \in \mathbb{F}[x_1, \ldots, x_n]$ computed by a π-oblivious ABP, $f \equiv 0 \Leftrightarrow f(\mathcal{G}) \equiv 0$. From this, one obtains an efficient black-box test, if the number of z-variables ℓ and the degree of $\mathcal{G}$ is "small".

For illustrative purposes, let us consider an π-oblivous ABP A with variable order $x_1, x_2, \ldots, x_{2n}$ of small width w, rather than small number of reads, and suppose it computes $f \not\equiv 0$. In order to achieve $\ell = O(w \log n)$, we cut A in the middle layer. This gives a decomposition (say) $f = \sum_{i\in[w]} g_i(x_1, \ldots, x_n)h_i(x_{n+1}, \ldots, x_{2n})$. Then we want $f(\mathcal{G}) = \sum_{i\in[w]} g_i(\mathcal{G}_1, \ldots, \mathcal{G}_n)h_i(\mathcal{G}_{n+1}, \ldots, \mathcal{G}_{2n}) \not\equiv 0$. We would like to use recursion on the g_is and h_is, but in order to get ℓ small, this means $\mathcal{G}^u := (\mathcal{G}_1, \ldots, \mathcal{G}_n)$ and $\mathcal{G}^d := (\mathcal{G}_{n+1}, \ldots, \mathcal{G}_{2n})$ will share most of the variables. Consequently, cancelations might occur and may result in $f(\mathcal{G}) \equiv 0$. However, we do know that $\{g_i(\mathcal{G}^u)\}_{i\in[w]}$ must "communicate" through a small dimensional space $\mathbb{F}^w$. This allows one to take $\mathcal{G}^d$ identical to $\mathcal{G}^u$, except for an additional component to the input that inflates the dimension of any non-empty finite union of *affine varieties*[1], given by the preimage of a single point in $\mathbb{F}^w$. More or less, $\mathcal{G}(z, z')$ will look

[1] Keeping with the terminology in [6], an *algebraic set* is the set of common zeroes of a list of polynomials. Affine varieties are algebraic sets, which are *irreducible* in the *Zariski-topology*.

like $\mathcal{G}^u(z); \mathcal{G}^d(z, z')$, with $\mathcal{G}^d(z, z') = \mathcal{G}^u(z + T(z'))$, where T is a mapping of $O(w)$ many variables that contains any w-dimensional coordinate subspace. Doing so, we only add $O(w)$ many variables per inductive step. This mirrors the pseudorandom generator construction of [7] mentioned before. To make an analogy, $z + T(z')$ can be thought of as similar to taking a vertex (we pick z) and adjacent edge (we move by $T(z)$) on an expander graph.

Necessarily, our final construction will be more complicated than the above sketch, since we assume a bound on the number of reads instead of the width. This will be dealt with by taking a partial derivatives w.r.t. a centrally local variable x_k in the ABP. Taking the derivative w.r.t. x_k has the net effect of cutting down the width of the x_k-layer of A.

2 Preliminaries

For a natural number n, we denote the set $\{1, 2, \ldots, n\}$ by $[n]$. For an n-tuple $a = (a_1, a_2, \ldots, a_n)$ and m-tuple $b = (b_1, b_2, \ldots, b_m)$, we denote $(a_1, a_2, \ldots, a_n, b_1, b_2, \ldots, b_m)$ by $a\#b$. Let $X = \{x_1, x_2, \ldots, x_n\}$ be a set of variables and let $\mathbb{F}$ be a field. For a polynomial $f \in R := \mathbb{F}[X]$, if it is identical to the zero polynomial of the ring R, we write $f \equiv 0$. If the degree of any variable of f is bounded by one, f is said to be *multilinear* (even if f has a constant term). We say f *depends on* x_i, if the formal partial derivative $\partial f/\partial x_i \not\equiv 0$. $Var(f)$ denotes the sets of variables f depends on. For a set of polynomial $f_1, \ldots, f_m \in \mathbb{F}[X]$, we say that they are *independent* if for all $a \in \mathbb{F}^m$ with $a \neq \overline{0}$, $\sum_{i\in[m]} a_i f_i \not\equiv 0$. We use the notation $f_{|x_i=\alpha}$ to denote substitution of x_i with $\alpha \in \mathbb{F}$.

We import the following definition and subsequent notations from [10]. An algebraic branching program (ABP) is a 4-tuple $A = (G, w, s, t)$, where $G = (V, E)$ is an edge-labeled directed acyclic graph for which the vertex set V can be partitioned into levels $L_0, L_1, \ldots, L_d$, where $L_0 = s$ and $L_d = t$. Vertices s and t are called the source and sink of A, respectively. Edges may only go between consecutive levels L_i and L_{i+1}. The subgraph induced by $L_i \cup L_{i+1}$ is called a *layer*. The label function $w : E \to X \cup \mathbb{F}$ assigns variables or field constants to the edges of G. For a path p in G, we extend the weight function by $w(p) = \prod_{e\in p} w(e)$. Let $P_{i,j}$ denote the collection of all paths p from i to j in G. The program A computes the polynomial $\sum_{p\in P_{s,t}} w(p)$. The size of A is taken to be $|V|$, and the read of A is the maximum of $|w^{-1}(x_i)|$, over all x_i's. The depth of A equals d, and the width of A equal $\max_i |L_i|$.

Algebraic branching programs were first introduced by Nisan [12]. Our definition differs in the respect that [12] requires edge labels to be linear forms. We remark that the read of an ABP always refers to *global read*, i.e. it bounds the total number of times a variable x_i can be reads in the entire ABP. With some abuse, an ABP A is called a read r ABP, if its read is bounded by r. We also denote this by saying that A is a R_r-ABP. A polynomial $f \in \mathbb{F}[X]$ is called a R_r-ABP-polynomial if there exists a R_r-ABP computing f. We use the following notation: for an arc $e = (v, w)$ in ABP A, $begin(e) = v$ and $end(e) = w$. We let $source(A)$ and $sink(A)$ stand for the source and sink of A. For any nodes v, w in A, we denote the subprogram with source v and sink w by $A_{v,w}$. We use $\widehat{A}$ to denote the polynomial computed by A, and in particular, $\widehat{A_{v,w}}$ is the polynomial computed by the subprogram $A_{v,w}$. A *layer* of an ABP A is the subgraph induced by two consecutive levels L_i and L_{i+1} in A.

▶ **Definition 1.** Let π be a permutation of $[n]$. An ABP A is π-ordered, if on every directed path p in A, if a variable x_i appears before x_j on p, then $\pi(i) < \pi(j)$. For an ABP A we say it is ordered if it is π-ordered w.r.t. some permutation π.

For a π-ordered ABP (π-OABP) variables appear (with possibly omissions) on any path from source to sink in the order $x_{\pi^{-1}(1)}, x_{\pi^{-1}(2)}, \ldots, x_{\pi^{-1}(n)}$. We will speak of the latter

sequence as the *variable order* of A. To stress, for a π-OABP, for any path p, a variable x_i appears at most once on p. Hence, if a π-OABP is read r, each variable x_i can appear at most r times in the ABP, and each occurrence must be on a different path from the source to the sink. Note that the output of a π-OABP must be a multilinear polynomial. Ordered algebraic branching programs where first studied in [8], but with respect to the homogeneous ABP definition of [12]. There the ordering condition states that on any path p, for any edge e_1 appearing before e_2 on p, if e_1 is labeled by $\sum_{i\in[n]} a_i x_i$, and e_2 is labeled by $\sum_{i\in[n]} b_i x_i$, then all variables in $\{x_i : a_i \neq 0\}$ appear before all variables in $\{x_i : b_i \neq 0\}$ in the variable order. The usual "homogenization trick" of splitting nodes into parts computing homogeneous components can be used to convert any OABP to the model of [8] (one also needs to collapse circuitry going over constant wires). This outlines a proof of the second part of the following lemma (the first part being obvious):

▶ **Lemma 2.** *For any permutation π of $[n]$ we have the following:*

1. *A homogeneous π-ordered ABP of size s with linear forms as edge labels can be converted into an equivalent π-OABP with weight function $w : E \to X \cup \mathbb{F}$ of size $O(ns)$.*
2. *For any π-OABP of size s computing a homogeneous polynomial of degree d, there exists an equivalent homogeneous π-ordered ABP of size $O(sd)$ with linear forms as edge labels.*

An ABP is called *oblivious*, if for any layer all variables are the same. We call a layer an x-layer, if x labels some of the edges in that layer, for $x \in X$. Layers with variables are called *variable layers*. Layers without variables are called *constant layers*. We say an ABP is *π-oblivous*, if it is oblivious, *and* for each variable x_i there is at most one x_i-layer, *and* the layers appear in the order $x_{\pi^{-1}(1)}, x_{\pi^{-1}(2)}, \ldots, x_{\pi^{-1}(n)}$ (with possible omissions) in the ABP.

To emphasize, for a read r π-oblivious ABP we have at most n layers where variables are read. These layers appear in the variable order when going from the source to the sink, and can be interleaved with constant labeled layers. Then for a variable layer w.r.t. a variable x, we have at most r occurences of x on an edge in this layer. Any remaining edges in the layer must be labeled by constants. The proof of the following lemma follows by some straightforward circuit manipulations, and it will appear in the full version of the paper. Note the lemma preserves read.

▶ **Lemma 3.** *For any permutation π of $[n]$, given a π-OABP A over n variables of size s and read r, there is an equivalent π-oblivious ABP B of size $O(sn)$, width $\leq 2s$, read r.*

Any subset $X \subseteq \mathbb{F}^n$ which is the set of simultaneous zeroes of a set of polynomials $f_1, \ldots, f_t \in \mathbb{F}[x_1, \ldots, x_n]$ is called an *algebraic set*. For basic definitions we refer to [5, 6]. If X and Y are algebraic sets in $\mathbb{F}^n$, we denote by $X+Y$ the subset $\{x+y \in F^n : x \in X, y \in Y\}$. Note that $X+Y$ may not be an algebraic set. We denote by $\overline{X+Y}$ the closure of $X+Y$ in the Zariski-topology. We need the following two lemmas:

▶ **Lemma 4.** *Let $X \subset \mathbb{F}^n$ be an algebraic set of dimension $0 \leq r < n$. Then for some $(n-r)$-dimensional coordinate subspace $C \subset \mathbb{F}^n$, $\overline{X+C} = \mathbb{F}^n$.*

Proof. For a coordinate subspace C denote the canonical projection to C by π_C. Consider $K = \{0\}^r \times \mathbb{F}^{n-r}$ and $L = \mathbb{F}^r$, which we think of as the complement of K corresponding to the first r coordinates. We have the following two properties: 1) The set $X + K$ equals $\pi_L(X) \times \mathbb{F}^{n-r}$, and 2) $\overline{\pi_L(X) \times \mathbb{F}^{n-r}} = \overline{\pi_L(X)} \times \mathbb{F}^{n-r}$.

By this, $\dim \overline{X+K} = n - r + \dim \overline{\pi_L(X)}$. More generally, it can be seen (by applying isomorphisms to $\mathbb{F}^n$, where we permute the indices), that for any $(n-r)$ -dimensional coordinate subspace C with r-dimensional complement D, $\dim \overline{X+C} = n - r + \dim \overline{\pi_D(X)}$.

Hence the lemma follows from the fact that for any r-dimensional affine variety there exists a projection τ to some r-dimensional coordinate subspace E such that $\tau(X)$ is dense in E, i.e. $\dim \overline{\pi_D(X)} = r$. For a proof of the latter see [5], p480. ◀

▶ **Lemma 5** (Lemma 2.1 in [3]). *Let $f \in \mathbb{F}[X]$ be a nonzero polynomial such that the degree of f in x_i is bounded by r_i, and let $S_i \subseteq \mathbb{F}$ be of size at least $r_i + 1$, for all $i \in [n]$. Then there exists $(s_1, s_2, \ldots, s_n) \in S_1 \times S_2 \times \ldots \times S_n$ with $f(s_1, s_2, \ldots, s_n) \neq 0$.*

The following lemma gives us a decomposition satisfying some useful independence properties.

▶ **Lemma 6.** *Let $k \geq 1$, and let A be an oblivious ABP of width w with source s and sink t having variable order $x_1, x_2, .., x_{2n}$. Suppose $\widehat{A} \not\equiv 0$. Then we can write for some $w' \leq w$, $f = \sum_{i \in [w']} f_i g_i$, where*

1. *$\{f_1, f_2, \ldots, f_{w'}\} \subseteq \mathbb{F}[x_1, x_2, \ldots, x_n]$ and $\{g_1, g_2, \ldots, g_{w'}\} \subseteq \mathbb{F}[x_{n+1}, x_{n+2}, \ldots, x_{2n}]$ are both independent sets of polynomials.*
2. *$\forall a \in \mathbb{F}^{w'}$, $\sum_{i \in [w']} a_i f_i$ can be computed by an oblivious ABP of width w with variable order $x_1, x_2, .., x_n$.*
3. *$\forall a \in \mathbb{F}^{w'}$, $\sum_{i \in [w']} a_i g_i$ can be computed by an oblivious ABP of width w with variable order $x_{n+1}, x_{n+2}, \ldots, x_{2n}$.*

Proof. Let V be the set of variables used in A. Pick an arbitrary level L of nodes $v_1, v_2, \ldots, v_w$ such that $V \cap \{x_1, x_2, \ldots, x_n\}$ appear on edges in layers before L, and $V \cap \{x_n, x_{n+1}, \ldots, x_{2n}\}$ appear on edges in layers after L. For $i \in [w]$, let $f_i = \widehat{A_{s,v_i}}$ and $g_i = \widehat{A_{v_i,t}}$. We proceed in two phases. First we arrange for a decomposition where the f_is are independent. Then we will deal with the g_is.

Wlog. assume that $f_1, \ldots, f_k$ is a maximum size independent set of polynomials. Since $f \not\equiv 0$, we know that not all $f_i \equiv 0$. So $k \geq 1$. For $j > 0$, any f_{k+j} can be written as a linear combination of $f_1, \ldots, f_k$. Let A' be an equivalent ABP obtained from A as follows. First, A' is just as A from the source up to the level L, except that we drop $v_{k+1}, \ldots v_w$ from L. Let us use L' to denote the modified level L. L' is followed by a constant layer, where $f_1, \ldots, f_w$ are computed (relative to s). After this we attach all the levels of A, just as they followed L in A. We have that $f = \sum_{i \in [k]} f_i g'_i$, where $f_i = \widehat{A'_{s,v_i}}$ and $g'_i = \widehat{A'_{v_i,t}}$. The f_is satisfy the first two conditions of the lemma. The g'_is are in $\mathbb{F}[x_{n+1}, x_{n+2}, \ldots, x_{2n}]$. This completes the first phase.

For the next phase, wlog. assume that $g'_1, \ldots, g'_l$ is a maximum size independent set. Say these correspond to nodes $w_1, \ldots, w_l$, respectively. That is, $\widehat{A'_{w_i,t}} = g'_i$. Since $f \not\equiv 0$, we know that $l \geq 1$. Symmetrically to the first phase, but now going in the direction from sink to source, we modify A' into an equivalent ABP A''. A'' is the same as A' from the sink back to the level L', except that we drop nodes other than $w_1, \ldots, w_l$ from L'. Above this is a constant level, where we compute $g'_1, \ldots, g'_k$ (relative to the sink). Above this we attach all level from A', just as they appear from s to L' in A'. We now have arranged that $f = \sum_{i \in [l]} f''_i g'_i$, where $f''_i = \widehat{A''_{s,w_i}}$ and $g'_i = A''_{w_i,t}$, for $i \in [l]$. Observe that for each $i \in [l]$, $f''_i = f'_i + \text{Linear}(f_{l+1}, \ldots, f_k)$. Hence $\{f''_1, \ldots, f''_l\}$ is an independent set of polynomials. All required properties of the lemma are now clearly satisfied. ◀

▶ **Corollary 7.** *Let $k \geq 1, n \geq 3$ and let $1 < i < n$. Let A be a read r oblivious ABP, with source s and sink t having variable order $x_1, x_2, \ldots, x_{i-1}, x_i, x_{i+1}, \ldots, x_n$. We use y as alias for x_i. Let $f = \partial \widehat{A} / \partial y$. Suppose $\widehat{A}$ depends on y, that is $f \not\equiv 0$. Then we can write for some $r' \leq r$, $f = \sum_{i \in [r']} p_i q_i$, where*

1. $\{p_1, p_2, \ldots, p_{r'}\} \subseteq \mathbb{F}[x_1, x_2, \ldots, x_{i-1}]$ *and* $\{q_1, q_2, \ldots, q_{r'}\} \subseteq \mathbb{F}[x_{i+1}, x_{i+2}, \ldots, x_n]$ *are both independent sets of polynomials.*
2. $\forall a \in \mathbb{F}^{r'}$, $\sum_{i\in[r']} a_i p_i$ *can be computed by a read* r *oblivious ABP with variable order* $x_1, x_2, .., x_{i-1}$.
3. $\forall a \in \mathbb{F}^{r'}$, $\sum_{i\in[r']} a_i q_i$ *can be computed by a read* r *oblivious ABP with variable order* $x_{i+1}, x_{i+2}, \ldots, x_n$.

Proof. Corollary 7 is now proved as follows. We make changes to A by modifying the edges in the y-layer as follows: for a variable edge (labeled with y), label it with 1. For a constant edge, remove it. The resulting ABP A' computes f. Then in the proof of Lemma 6, take the level L to be the starting level of the original y-layer. As $|L|$ is bounded by the number of y-variables in the y-layer of A, we are done. ◀

3 A Generator for π-Oblivious ABPs

We assume $|\mathbb{F}|$ is large enough. The explicit requirement on $|\mathbb{F}|$ will become clear after the description of the generator. For now, let us fix $S = \{\alpha_1, \ldots, , \alpha_N\} \subseteq \mathbb{F}$, for some N, and let $S_m = \{\alpha_1, \ldots, \alpha_m\}$, for $1 \leq m \leq N$. Let $Z = \{z_1, z_2, \ldots\}$, $Y = \{y_1, y_2, \ldots\}$, $U = \{u_1, u_2, \ldots\}$ and $V = \{v_1, v_2, \ldots\}$ be sets of variables. For $k \geq 1$, we use Z_k to denote the k-tuple of variables $(z_1, z_2, \ldots, z_k)$, similarly for Y_k, U_k and V_k. Define the function ℓ on natural numbers by $\ell(k, r) = 2rk + 1$. Abusing notation, we write $(Z_{\ell(k,r)}, U_k, V_k)$ to denote the tuple $Z_{\ell(k,r)} \# U_k \# V_k$.

For every $k \geq 0$, $r \geq 1$ and a variable w, let $H^{k,r}(w) = (H_1^{k,r}(w), H_2^{k,r}(w), \ldots H_{\ell(k,r)+2k}^{k,r}(w))$, where for each $i \in [\ell(k, r) + 2k]$, $H_i^{k,r}$ is the ith Lagrange interpolation polynomial on the set $S_{\ell(k,r)+2k}$. $H_i^{k,r}$ is a univariate polynomial in w of degree $\ell(k, r) + 2k - 1$, satisfying that $\forall \alpha_j \in S_{\ell(k,r)+2k}$, $H_i^{k,r}(\alpha_j) = 1$ if $i = j$ and 0 otherwise. For $k \geq 1$, and two variables u and v, let $E^k(u, v) = (u \cdot L_1^k(v), \ldots, u \cdot L_{2^k}^k(v))$, in which L_i^k is the ith Lagrange interpolation polynomial on the set S_{2^k}.

For $k \geq 0$ and $r \geq 1$, we define the polynomial mapping $F^{k,r}(Z_{\ell(k,r)}, U_k, V_k) : \mathbb{F}^{\ell(k,r)+2k} \to \mathbb{F}^{2^k}$ inductively as follows:

1. $F^{0,r}(z_1) = z_1$, and
2. For clarity we use $y_1, y_2, \ldots, y_{2r}$ as aliases for the variables $z_{\ell(k,r)+1}, z_{\ell(k,r)+2}, \ldots, z_{\ell(k,r)+2r}$, respectively. We take $F^{k+1,r}(Z_{\ell(k,r)}, Y_{2r}, U_{k+1}, V_{k+1})$ to be equal to the following 2^{k+1}-tuple of polynomials:

$$E^{k+1}(u_{k+1}, v_{k+1}) + \left[F^{k,r}(Z_{\ell(k,r)}, U_k, V_k) \# F^{k,r}\left((Z_{\ell(k,r)}, U_k, V_k) + T^{k,r}(Y_{2r})\right)\right],$$

where $T^{k,r} : \mathbb{F}^{2r} \to \mathbb{F}^{\ell(k,r)+2k}$ is defined by $T^{k,r}(Y_{2r}) = \sum_{i\in[r]} y_i \cdot H^{k,r}(y_{r+i})$.

From the construction we can see that in order to accommodate for S, $|\mathbb{F}|$ should be no less than $\max(\ell(k, r)+2k, 2^k)$. Note that the image of $T^{k,r}$ contains any r-dimensional coordinate subspace of $\mathbb{F}^{\ell(k,r)+2k}$. Namely, for $i \in [r]$, by choosing $y_{r+i} = \alpha_j$, the corresponding vector of $y_i H^{k,r}(y_{r+i})$ becomes $y_i e_j$, where e_j is the jth standard basis vector of $\mathbb{F}^{\ell(k,r)+2k}$. Thus by choosing different α's for the y_{r+i}'s, we can form any r-dimensional coordinate subspace in the image. The term E^{k+1} is there to deal with bounded read, e.g. it would not be needed if we want to have a generator for small width π-oblivious ABPs. Ignoring this term, the generator mimics the construction of [7]. Intuitively, the dimension expanding properties of $T^{k,r}$ will yield that the two sides of the generator appear to be behaving "independently enough", yielding the desired non-cancelation property.

3.1 Properties of the Generator

Let us compute $F^{1,r}$ to get a sense and for later use. We obtain

$$\begin{aligned} F^{1,r} &= E^1(u_1, v_1) + F^{0,r}(z_1) \# F^{0,r}(z_1 + (z_{1+1} + \cdots + z_{1+r})) \\ &= (u_1 L_1^1(v_1) + z_1, u_1 L_2^1(v_1) + z_1 + \cdots + z_{1+r}). \end{aligned}$$

Note that $z_{2+r}, \ldots, z_{1+2r}$ are not used in the Lagrange interpolation in the $T^{0,r}$ part. By a straightforward induction, one can prove the following bound for the individual degree of a variable in $F^{k,r}$.

▶ Proposition 1. $\forall k \geq 2$ and $r \geq 1$, the individual degree of any variable in any component of $F^{k,r}$ is at most $\prod_{j \in [k-1]} (\ell(j,r) + 2j)(\ell(j,r) + 2j - 1))$.

The following theorem shows that the generator $F^{k,r}$ works for the class $\mathcal{C}$ of polynomials computed by read r π-oblivious ABPs, where there is one single fixed order π of the variables for the entire class $\mathcal{C}$. Wlog. the order is assumed to be $x_1, x_2, \ldots$. A generator for any other fixed order, is obtained by permuting the components of the output of the generator in the appropriate way. To make the algebraic geometry go through in the proof, we will assume that $\mathbb{F}$ is algebraically closed. We will remove this requirement subsequently with Corollary 9.

▶ **Theorem 8.** *Let $\mathbb{F}$ be an algebraically closed field. Let $k \geq 0$, and let A be a π-oblivious ABP of read $r \geq 1$ with variable order $x_1, x_2, \ldots, x_{2^k}$. Suppose A computes f, then $f \equiv 0 \iff f(F^{k,r}) \equiv 0$.*

Proof. The "⇒"-direction is trivial, so it suffices to show that if $f \not\equiv 0$, then $f(F^{k,r}) \not\equiv 0$. We prove this by induction on k. For $k = 0$ it is obvious. For $k = 1$, we know there exists (a, b) such that $f(a, b) \neq 0$. Recall $F^{1,r} = (u_1 L_1^1(v_1) + z_1, u_1 L_2^1(v_1) + z_1 + \cdots + z_{1+r})$. Then setting c to be the assignment of (Z_{2r+1}, u_1, v_1) as $z_1 = a$, $z_2 = b - a$ and other variables to 0, would give $f(F^{1,r}) = f(a, b) \neq 0$. So $f(F^{1,r}) \not\equiv 0$.

Now let $k \geq 1$. For the induction step from k to $k+1$, we need to prove that $F^{k+1,r}$ works for an oblivious read r ABP polynomial f with variables $x_1, \ldots, x_{2^{k+1}}$. We use X as an alias for x_{2^k}, and Λ as an alias for α_{2^k}. Let $g = \partial f / \partial X$, and note that $f = g \cdot X + f \mid_{X=0}$, since f is multilinear. Wlog. we can assume that f depends on X. Namely, since f is multilinear, if f does not depend on any variable, i.e. $\forall i, \partial f / \partial x_i \equiv 0$, then $f \in \mathbb{F}$ (even if $char(\mathbb{F}) > 0$). Clearly the theorem holds in this case. Otherwise, the rest of the proof goes through mutatis mutandis by selecting X to be the median variable (w.r.t. the variable order $x_1, x_2, \ldots$) of variables that f depends on. Thus $g \not\equiv 0$. We claim that the following holds:

▶ Claim 1. $h := g(F^{k+1,r}) \mid_{v_{k+1}=\Lambda} \not\equiv 0$

Before proving Claim 1, let us show that this is sufficient to complete the proof of Theorem 8. We will prove Claim 1 in the next subsection. Consider $f(F^{k+1,r}) \mid_{v_{k+1}=\Lambda}$. It is equal to the following:

$$\begin{aligned} h \cdot \left(F_{2^k}^{k+1,r} \mid_{v_{k+1}=\Lambda}\right) + \left((f \mid_{X=0})(F^{k+1,r})\right) \mid_{v_{k+1}=\Lambda} &= \\ h \cdot \left((E_{2^k}^{k+1} + P(Z_{\ell(k,r)}, U_k, V_k)) \mid_{v_{k+1}=\Lambda}\right) + \left((f \mid_{X=0})(F^{k+1,r})\right) \mid_{v_{k+1}=\Lambda} &= \\ h \cdot (u_{k+1} + P(Z_{\ell(k,r)}, U_k, V_k)) + \left((f \mid_{X=0})(F^{k+1,r})\right) \mid_{v_{k+1}=\Lambda}, & \end{aligned}$$

for some polynomial P in variables $(Z_{\ell(k,r)}, U_k, V_k)$. Observe that $\left((f \mid_{X=0})(F^{k+1,r})\right) \mid_{v_{k+1}=\Lambda}$ does not contain the variable u_{k+1}. The same holds for $P(Z_{\ell(k,r)}, U_k, V_k))$. Hence $h \cdot u_{k+1}$ cannot be canceled, and therefore $f(F^{k+1,r}) \mid_{v_{k+1}=\Lambda} \not\equiv 0$. This implies $f(F^{k+1,r}) \not\equiv 0$. ◀

▶ **Corollary 9.** *Let $\mathbb{F}$ be any field. Let $k \geq 0$, and let A be a π-oblivious ABP over $\mathbb{F}$ of read $r \geq 1$ with variable order $x_1, x_2, .., x_{2^k}$. Suppose A computes the polynomial $f \in \mathbb{F}[x_1, x_2, .., x_{2^k}]$. Then in the construction of $F^{k,r}$ selecting any set S of size $\max(\ell(k,r) + 2k, 2^k)$ contained in $\mathbb{F}$ (or an arbitrary field extension $\mathbb{G}$ of $\mathbb{F}$, if $\mathbb{F}$ is not large enough) yields that $f \equiv 0 \iff f(F^{k,r}) \equiv 0$,*

Proof. First consider the case when $\text{char}(\mathbb{F}) = 0$. In this case we take $S = \{0, 1, 2, \dots\}$. Let $\overline{\mathbb{F}}$ be the algebraic closure of $\mathbb{F}$. Interpreting A as an ABP over $\bar{\mathbb{F}}$, we can apply Theorem 8 to conclude $f \equiv 0 \iff f(F^{k,r}) \equiv 0$. All coefficients of $F^{k,r}$ are rational numbers and thus lie inside $\mathbb{F}$. Hence the property $f \equiv 0 \iff f(F^{k,r}) \equiv 0$ also holds when considering we work over $\mathbb{F}$.

In case $\text{char}(\mathbb{F}) > 0$, if $|\mathbb{F}|$ is not large enough, by allowing ourselves to use elements from the extension $\mathbb{G}$, we can still get the required S. Then similarly as above, by considering the algebraic closure of $\mathbb{G}$ and applying Theorem 8, the required generator property follows, considering one works over $\mathbb{G}$. ◀

3.2 Proof of Claim 1

Let $F'^{k+1,r} = F^{k+1,r} - E^{k+1}$. Note that since f is multilinear, g does not depend on X. Hence $g(F^{k+1,r}) \mid_{v_{k+1}=\Lambda} = g(F'^{k+1,r})$. We will show that $g(F'^{k+1,r}) \not\equiv 0$. We have that

$$F'^{k+1,r} = F^{k,r}(Z_{\ell(k,r)}, U_k, V_k) \# F^{k,r}\left((Z_{\ell(k,r)}, U_k, V_k) + T^{k,r}(Y_{2r})\right).$$

Again we will use $y_1, y_2, \dots, y_{2r}$ as alias for the variables $z_{\ell(k,r)+1}, z_{\ell(k,r)+2}, \dots, z_{\ell(k,r)+2r}$, respectively. Corollary 7 gives us that we can write $g = \sum_{i\in[r']} p_i q_i$, for some $r' \leq r$, where

1. $\{p_1, p_2, \dots, p_{r'}\} \subseteq \mathbb{F}[x_1, x_2, \dots, x_{2^k-1}]$ and $\{q_1, q_2, \dots, q_{r'}\} \subseteq \mathbb{F}[x_{2^k+1}, x_{2^k+2}, \dots, x_{2^{k+1}}]$ are both independent sets of polynomials.
2. $\forall a \in \mathbb{F}^{r'}$, $\sum_{i\in[r']} a_i p_i$ can be computed by an oblivious ABP of read r with variable order $x_1, x_2, .., x_{2^k-1}$.
3. $\forall a \in \mathbb{F}^{r'}$, $\sum_{i\in[r']} a_i q_i$ can be computed by an oblivious ABP of read r with variable order $x_{2^k+1}, x_{2^k+2}, \dots, x_{2^{k+1}}$.

For any $a \in \mathbb{F}^{r'}$ with $a \neq \overline{0}$, $\sum_{i\in[r']} a_i p_i \not\equiv 0$, and this sum can be computed by an oblivious ABP of read r with variable order $x_1, x_2, .., x_{2^k-1}$. Hence by induction hypothesis $\sum_{i\in[r']} a_i p_i(F^{k,r}) \not\equiv 0$. Let $\hat{p}_i = p_i(F^{k,r}(z_1, \dots, z_{\ell(k,r)}, U_k, V_k))$. The above shows that $P := \{\hat{p}_1, \hat{p}_2, \dots, \hat{p}_{r'}\}$ is an independent set of polynomials. Let $\hat{q}_i = q_i(F^{k,r}(z_1, \dots, z_{\ell(k,r)}, U_k, V_k))$. Similarly we have that $Q := \{\hat{q}_1, \hat{q}_2, \dots, \hat{q}_{r'}\}$ is an independent set of polynomials.

Since $\hat{p}_1 + \hat{p}_2 + \ldots + \hat{p}_{r'} \not\equiv 0$, there exists input $c \in \mathbb{F}^{\ell(k,r)+2k}$ so that if we let $a_i = \hat{p}_i(c)$, then $a = (a_1, a_2, \dots, a_{r'}) \neq \overline{0}$. Let $V \subseteq \mathbb{F}^{\ell(k,r)+2k}$ be the algebraic set defined by the system of equations

$$\{\hat{p}_i(z_1, \dots, z_{\ell(k,r)}, U_k, V_k) = a_i : \forall i \in [r']\}$$

We know this system has a solution namely c. Since $\mathbb{F}$ is assumed to be algebraically closed, by Exercise 1.9 p. 8 in [6], we know that each irreducible component of V has dimension at least $\ell(k,r) + 2k - r'$. Since the system is solvable there must exist at least one irreducible component, and since $r \geq 1$, $\ell(k,r) + 2k - r' \geq 3$.

Let $W \subseteq \mathbb{F}^{\ell(k,r)+2k}$ be the algebraic set defined by the equation $\sum_{i\in[r']} a_i \hat{q}_i(z_1, \dots, z_{\ell(k,r)}, U_k, V_k) = 0$. Since Q is an independent set of polynomials the l.h.s. of the above equation is a nonzero polynomial. In case the l.h.s. is a non-zero constant, then we are done. Namely, letting $b \in \mathbb{F}^{\ell(k+1,r)+2(k+1)}$ be the assignment where we

set $(z_1, \ldots, z_{\ell(k,r)}, U_k, V_k)$ to c, $y_1, \ldots, y_r$ to 0, and the remaining variables arbitrarily, would give $g(F'^{k+1,r})(b) = \sum_{i\in[r']} a_i \hat{q}_i(z_1, \ldots, z_{\ell(k,r)}, U_k, V_k)(b) \neq 0$. Otherwise, we know by Proposition 1.13 in [6], that W is a finite union of hypersurfaces each of dimension $\ell(k,r) + 2k - 1$ (these correspond to the irreducible factors of $\sum_{i\in[r']} a_i \hat{q}_i(z_1, \ldots, z_{\ell(k,r)}, U_k, V_k)$). We want to argue that $V + \operatorname{Im} T$ cannot be contained in W. Namely, to see the consequence, suppose we have $c' = c'' + T(d)$, for $c'' \in V$ and $d \in \mathbb{F}^{2r}$, with $c' \notin W$. Then letting $b \in \mathbb{F}^{\ell(k+1,r)+2k}$ be the assignment where we set $(z_1, \ldots, z_{\ell(k,r)}, U_k, V_k)$ to c'' and $Y_{2r} := d$ gives that $g(F'^{k+1,r})(b) = \sum_{i\in[r']} p_i(F^{k,r}(c''))q_i(F^{k,r}(c'' + T(d))) = \sum_{i\in[r']} \hat{p}_i(c'')\hat{q}_i(c'' + T(d)) = \sum_{i\in[r']} a_i \hat{q}_i(c') \neq 0$.

We complete the proof by showing that the Zariski-closure of $V + \operatorname{Im} T$ has dimension greater than $\dim W$.

▶ Claim 2. $\dim \overline{V + \operatorname{Im} T} = \ell(k,r) + 2k$.

Proof. As remarked upon before, for any $r'' \leq r$, $\operatorname{Im} T$ contains any r''-dimensional coordinate subspace of $\mathbb{F}^{\ell(k,r)+2k}$. Namely, by setting $y_{r+i} = \alpha_{j_i}$, for all $i \in [r]$, where $\alpha_{j_1}, \alpha_{j_2}, \ldots, \alpha_{j_r}$ are distinct elements of $S_{\ell(k,r)+2k}$, we obtain $\sum_{i\in[r]} y_i \cdot H^{k,r}(y_{r+i}) = \sum_{i\in[r]} y_i \cdot H^{k,r}(\alpha_{j_i}) = \sum_{i\in[r]} y_i \cdot e_{j_i}$, where $e_1, e_2, \ldots, e_{\ell(k,r)+2k}$ are standard basis vectors of $\mathbb{F}^{\ell(k,r)+2k}$. Hence the claim follows from Lemma 4. ◀

The above claim implies that $V + \operatorname{Im} T \not\subset W$. By the above remarks, this gives that $g(F'^{k+1,r})(b) \neq 0$, for some b. This proves Claim 1. ◀

4 A Black-Box PIT Algorithm for π-OABPs

Algorithm 1 PIT Algorithm for read r π-OABPs.

Input: Black-box access to $f \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ computed by a π-OABP with read r.
Output: returns **true** iff $f \equiv 0$.

1: let k be such that $2^{k-1} < n \leq 2^k$.
2: let $D = \prod_{j\in[k-1]}(\ell(j,r) + 2j)(\ell(j,r) + 2j - 1)$.
3: let S_{D+1} be an arbitrary subset of $\mathbb{F}$ (or an extension field of $\mathbb{F}$ if $|\mathbb{F}| < D + 1$) of size $D + 1$.
4: let $R = S_{D+1}^{\ell(k,r)+2k}$.
5: compute $A = F^{k,r}(R)$.
6: permute the vectors in A according to π.
7: For every $a \in A$, check whether $f(a) = 0$.
8: **return** **true** if in the previous stage no nonzero was found, **false** otherwise.

▶ **Theorem 10.** *Let $\mathbb{F}$ be an arbitrary field. Using black-box Algorithm 1 we can check deterministically in time $2^{O(r \log r \cdot \log^2 n \log\log n)}$ whether a given polynomial $f \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ computed by a read r π-OABP is identically zero or not. If $char(\mathbb{F}) > 0$, the algorithm is granted black-box access to extension fields of $\mathbb{F}$.*

Proof. By Lemma 3, we can assume wlog. that f is computed by a read r π-oblivious ABP. By Theorem 8, we see that $f \equiv 0 \Leftrightarrow f(F^{k,r}) \equiv 0$. By Proposition 1, the individual degree of variables of $f(F^{k,r})$ can be bounded by $D = \prod_{j\in[k-1]}(\ell(j,r) + 2j)(\ell(j,r) + 2j - 1)$. Correctness now follows from Lemma 5. Bounding D by $(2rk + 2k)^{2k}$, and knowing that the number of variables of $f(G^{k,r})$ is $2rk + 2k + 1$, the theorem follows by straightforward arithmetic.

We remark that the hitting set A, will be constructed over an extension field of $\mathbb{F}$ if $|\mathbb{F}| < \max(\ell(k,r)+2k, 2^k)$ or $|\mathbb{F}| < D+1$. In the former case, it is because of having enough interpolation points to define the generator. In the latter case it is in order to apply Lemma 5, as was done in the above. To work over the extension field the algorithm by Shoup [18] can be used to obtain an irreducible polynomial of degree d over $\mathbb{F}$ in time $poly(d)$. For us, it suffices for the degree of this polynomial to be bounded by $O(\log n \log r + \log n \log\log n)$. Field operations in the extension field then take time $poly(\log n, \log r)$, assuming a unit cost model for operations in $\mathbb{F}$. The cost of constructing A this way, can easily be seen to be subsumed by the time bound given in the theorem. ◀

The above implies that read $polylog(n)$ π-OABPs can be tested in DTIME$[2^{O(polylog(n))}]$.

5 Separation Results and Lower Bounds for OABPs

Omitted proofs in this section will appear in the full version of the paper.

▶ **Theorem 11.** *Any OABP computing the permanent or determinant of an $n \times n$ matrix of variables has size $\Omega(2^n/n)$ and read $\Omega(2^n/n^2)$.*

By extending the construction in [15], we can prove the following theorem:

▶ **Theorem 12.** *Let $X = \{x_i\}_{i\in[2n+1]}$ and $\mathcal{W} = \{w_{i,j,k}\}_{i,j,k\in[2n+1]}$ be sets of variables. We can construct an explicit polynomial $p \in \mathbb{F}[X, \mathcal{W}]$ such that any OABP A over variables $X \cup \mathcal{W}$ using constants from $\mathbb{F}$ computing p requires some variable to be read at least 2^n times.*

We can also reinterpret the above result to be giving a stronger lower bound (seen as a function of the number of variables), but for a polynomial which uses $O(n^3)$ transcendental constants in its definition.

▶ **Corollary 13.** *For any field $\mathbb{F}$, and any extension field $\mathbb{G}$ of $\mathbb{F}$ of transcendence degree at least $(2n+1)^3$, there exists an explicit polynomial $p \in \mathbb{G}[x_1, x_2, \ldots, x_{2n+1}]$, such that any OABP over $\mathbb{G}$ computing p requires some variable to be read at least 2^n times.*

Consider the elementary symmetric polynomial $S_n^k = \sum_{S\subset[n],|S|=k} \prod_{i\in S} x_i$.

▶ **Theorem 14.** *S_n^k can not be computed by an R_{k-1}-OABP, for $n \geq 2k-1$, $k \geq 2$.*

▶ **Theorem 15.** *S_n^k can be computed by an R_k-OABP of size $O(kn)$, for $n \geq k \geq 1$.*

The following theorem shows that under different permutations π and π', the gap between the number of reads for the models π-OABP and π'-OABP can be exponentially large.

▶ **Theorem 16.** *Given $X = \{x_0, x_1, \ldots, x_{2n-1}, x_{2n}\}$, $n \geq 1$, there exists a polynomial p on X, and two permutations π and π' on X, such that 1) There exists a read-once π-OABP computing p, and 2) Any π'-OABP computing p requires read 2^n.*

References

1 M. Agrawal. Proving lower bounds via pseudo-random generators. In *Proc. 25th Annual Conference on Foundations of Software Technology and Theoretical Computer Science*, pages 92–105, 2005.

2 M. Agrawal and V. Vinay. Arithmetic circuits: A chasm at depth four. In *Proc. 49th FOCS*, pages 67–75, 2008.

3 N. Alon. Combinatorial nullstellensatz. *Combinatorics, Probability and Computing*, 8(1–2):7–29, 1999.

4 R.E. Bryant. On the complexity of vlsi implementations and graph representations of boolean functions with application to integer multiplication. *IEEE Trans. Computers*, 40(2):205–213, 1991.

5 D. Cox, J. Little, and D. O'Shea. *Ideals, Varieties, and Algorithms, Second Edition*. Undergraduate Texts in Mathematics. Springer Verlag, 1996.

6 R. Hartshorne. *Algebraic Geometry*. Graduate Texts in Mathematics, Vol 52. Springer Verlag, 1977.

7 R. Impagliazzo, N. Nisan, and A. Wigderson. Pseudorandomness for network algorithms. In *Proc. 26th STOC*, pages 356–364, 1994.

8 M. Jansen. Lower bounds for syntactically multilinear algebraic branching programs. In *Proc. 33rd MFCS*, volume 5162 of *Lect. Notes in Comp. Sci.*, pages 407–418, 2008.

9 M. Jansen. Weakening assumptions for deterministic subexponential time non-singular matrix completion. In *27th STACS*, volume 5 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 465–476, 2010.

10 M. Jansen, Y. Qiao, and J. Sarma M.N. Deterministic identity testing of read-once algebraic branching programs, 2009. http://arxiv.org/abs/0912.2565.

11 V. Kabanets and R. Impagliazzo. Derandomizing polynomial identity testing means proving circuit lower bounds. *Computational Complexity*, 13(1–2):1–44, 2004.

12 N. Nisan. Lower bounds for non-commutative computation: extended abstract. In *Proc. 23rd Annual ACM STOC*, pages 410–418, 1991.

13 R. Raz. Multilinear formulas for permanent and determinant are of super-polynomial size. *J. Assn. Comp. Mach.*, 56(2):1–17, 2009.

14 R. Raz and A. Shpilka. Deterministic polynomial identity testing in non commutative models. *Computational Complexity*, 14(1):1–19, 2005.

15 R. Raz and A. Yehudayoff. Balancing syntactically multilinear arithmetical circuits. *Computational Complexity*, 17(4):515–535, 2008.

16 N. Saxena. Progress of polynomial identity testing. Technical Report ECCC TR09-101, Electronic Colloquium in Computational Complexity, 2009.

17 J.T. Schwartz. Fast probabilistic algorithms for polynomial identities. *J. Assn. Comp. Mach.*, 27:701–717, 1980.

18 V. Shoup. New algorithms for finding irreducible polynomials over finite fields. In *Proc. 29th FOCS*, pages 283–290, 1988.

19 A. Shpilka and I. Volkovich. Improved polynomial identity testing of read-once formulas. In *Approximation, Randomization and Combinatorial Optimization. Algorithms and Techniques, volume 5687 of LNCS*, pages 700–713, 2009.

20 S. Toda. Classes of arithmetic circuits capturing the complexity of computing the determinant. *IEICE Trans. Inf. Syst.*, E75-D:116–124, 1992.

21 L. Valiant. Completeness classes in algebra. Technical Report CSR-40-79, Dept. of Computer Science, University of Edinburgh, April 1979.

22 R. Zippel. Probabilistic algorithms for sparse polynomials. In *Proceedings of the International Symposium on Symbolic and Algebraic Manipulation (EUROSAM '79)*, volume 72 of *Lect. Notes in Comp. Sci.*, pages 216–226. Springer Verlag, 1979.