Electronic Colloquium on Computational Complexity, Report No. 116 (2009)

Deterministic identity testing of depth 4 multilinear circuits with bounded top fan-in

Zohar S. Karnin * Partha Mukhopadhyay* Amir Shpilka*
Ilya Volkovich*

November 15, 2009

Abstract

We give the first sub-exponential time deterministic polynomial identity testing algorithm for depth-4 multilinear circuits with a small top fan-in. More accurately, our algorithm works for depth-4 circuits with a plus gate at the top (also known as $\Sigma\Pi\Sigma\Pi$ circuits) and has a running time of $\exp(\text{poly}(\log(n), \log(s), k))$ where n is the number of variables, s is the size of the circuit and k is the fan-in of the top gate. In particular, when the circuit is of polynomial (or quasi-polynomial) size, our algorithm runs in quasi-polynomial time. In [AV08], it was shown that derandomizing polynomial identity testing for general $\Sigma\Pi\Sigma\Pi$ circuits implies a derandomization of polynomial identity testing in general arithmetic circuits. Prior to this work sub-exponential time deterministic algorithms were known for depth-3 circuits with small top fan-in and for very restricted versions of depth-4 circuits.

The main ingredient in our proof is a new structural theorem for multilinear $\Sigma\Pi\Sigma\Pi(k)$ circuits. Roughly, this theorem shows that any nonzero multilinear $\Sigma\Pi\Sigma\Pi(k)$ circuit contains an 'embedded' nonzero multilinear $\Sigma\Pi\Sigma(k)$ circuit. Using ideas from previous works on identity testing of sums of read-once formulas and of depth-3 multilinear circuits, we are able to exploit this structure and obtain an identity testing algorithm for multilinear $\Sigma\Pi\Sigma\Pi(k)$ circuits.

*Research supported by the Israel Science Foundation (grant number 439/06). Faculty of Computer Science, Technion, Haifa 32000, Israel. Emails: zkarnin,partha,shpilka,ilyav@cs.technion.ac.il

ISSN 1433-8092

1 Introduction

Polynomial Identity Testing (PIT) is one of the central problems in algebraic complexity theory: Given an arithmetic circuit C over a field $\mathbb{F}$ with input variables $x_1, x_2, \cdots, x_n$, can we check efficiently whether C computes the identically zero polynomial in the polynomial ring $\mathbb{F}[x_1, x_2, \cdots, x_n]$? The same question can be asked in the black-box model too. In the black-box model, C is accessed by a black-box where we are allowed to substitute field elements $a_i \in \mathbb{F}$ for x_i and the black-box returns the value of $C(a_1, a_2, \cdots, a_n)$.

A randomized polynomial-time algorithm (more precisely a coRP algorithm) for this problem is known due to the Schwartz-Zippel Lemma [Sch80, Zip79]. Over the years PIT has played a significant role in our understanding of important complexity theoretic and algorithmic problems. Well-known examples are the randomized NC algorithms for the matching problem in graphs [Lov79, MVV87], and the AKS primality test [AKS04]. The PIT problem has also played an indirect role in important complexity results such as IP = PSPACE [LFKN92, Sha90] and the old proof of PCP theorem [ALM+98].

The main open problem is to come up with a deterministic polynomial-time (or at least subexponential-time) algorithm for PIT. In 2003, Kabanets and Impagliazzo [KI03] show that giving a deterministic polynomial-time (even subexponential-time) identity testing algorithm means either NEXP $\not\subset$ P/poly or that the integer Permanent has no polynomial size arithmetic circuit. Considering the black-box derandomization of PIT, Agrawal further strengthen the connection of PIT with proving circuit lower bounds [Agr05]. More precisely, he shows that the black-box derandomization of PIT implies that an explicit multilinear polynomial has no subexponential size arithmetic circuit.

The results of [KI03] and [Agr05] have triggered a large amount of research for PIT derandomization. So far, most of the derandomization results are known for depth-3 $\Sigma\Pi\Sigma(k, d)$ circuits when the top Σ gate is of bounded fan-in k (d is the fan-in of the Π gates which can be unbounded) [DS06, KS07, KS08, SS09, KS09]. In an important discovery, Agrawal and Vinay [AV08] justified the lack of progress beyond depth-3. What they show is that the *black-box* derandomization of PIT for only depth-4 $\Sigma\Pi\Sigma\Pi$ circuits is almost as hard as that for *general* arithmetic circuits. Their result is based on a depth reduction technique [VSBR83, AJMV98] that converts any arithmetic circuits C to a depth-4 circuit C' such that C and C' compute the same polynomial. Thus, their reduction is suitable for *black-box* PIT derandomization. This connection makes the problem of black-box derandomization of PIT for depth-4 circuits an intriguing open problem.

So far all the black-box derandomization algorithms for depth-3 $\Sigma\Pi\Sigma(k, d)$ circuits [DS06, KS08, SS09, KS09] exploit one common theme: The subspace spanned by the linear forms of an *identically zero* $\Sigma\Pi\Sigma$ circuit (viewing each linear form as a vector in $\mathbb{F}^n$) is of low dimension. More precisely, over a finite field, the current best known bound for the dimension is $\mathcal{O}(k^3 \log d)$ [SS09] and over the field $\mathbb{Q}$ of rational numbers, the bound is $2^{\mathcal{O}(k \log k)}$ which is still a constant for a constant k [KS09]. For multilinear $\Sigma\Pi\Sigma(k, d)$ circuits, over any characteristic, the dimension is bounded by $\mathcal{O}(k^3 \log(k))$ [DS06, SS09]. Yet, the algorithm with the best running time [SV09] was obtained using a different approach. For depth-4 circuits, the situation is very different. It seems unlikely that the method of black-box

derandomization for $\Sigma\Pi\Sigma$ circuits can be adopted/extended for depth-4 circuits. The main difficulty is that there seems to be no notion of a linear space, spanned by the circuit components, that can be used.

In this paper, we study the black-box PIT problem for multilinear depth-4 circuits with bounded fan-in at the top Σ gate. We give new techniques and come up with an efficient black-box algorithm which runs in time *quasi-polynomial* in the input size. We first formally define a depth-4 circuit. A depth-4 $\Sigma\Pi\Sigma\Pi$ circuit has four layers of alternating Σ and Π gates. The top gate is a Σ gate (at level one).[1] The circuit computes a polynomial $C(x_1, x_2, \cdots, x_n)$ of the form $C(x_1, x_2, \cdots, x_n) = \sum_{i=1}^{k} \prod_{j=1}^{d_i} P_{ij}$, where k is the fan-in of the top Σ gate and d_i are the fan-in's of the Π gates at the second level. P_{ij}-s are the polynomials computed at the third level of the circuit (which is a $\Sigma\Pi$ component). It is clear that the number of monomials in each P_{ij} is bounded by the fan-in of the Σ gates at the third level, and in particular, bounded by the circuit size s. In the rest of the paper, we refer to the polynomials P_{ij} as s-sparse polynomials where the sparsity should be understood as a parameter of the circuit size. Also for notational convenience, we denote depth-4 circuits whose top Σ fan-in is at most k by $\Sigma\Pi\Sigma\Pi(k)$ circuits.

We consider the identity testing problem of $\Sigma\Pi\Sigma\Pi(k)$ circuits when each multiplication gate $\prod_{j=1}^{d_i} P_{ij}$ computes a multilinear polynomial and the fan-in of the top Σ gate is a constant k. We call such circuits depth-4 multilinear $\Sigma\Pi\Sigma\Pi$ circuits. We give a deterministic *black-box* PIT algorithm for this model with a running time $\exp(\mathrm{poly}(\log(n), \log(s), k))$ (s is the size of the circuit) which is *quasi-polynomial* in the input size. More formally, we prove the following theorem.

Theorem 1. *Let k, n, s be integers. There is an explicit set $\mathcal{H}$ of size $n^{\mathcal{O}(k^6 \log(k) \log^2 s)}$, that can be constructed in time $n^{\mathcal{O}(k^6 \log(k) \log^2 s)}$ such that the following holds. Let $P \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ be a non-zero polynomial computed by a multilinear $\Sigma\Pi\Sigma\Pi(k)$ circuit of size s on n variables. Then there is some $\bar{\alpha} \in \mathcal{H}$ such that $P(\bar{\alpha}) \neq 0$.*

To the best of our knowledge, prior to our work, efficient deterministic algorithm were known only in the non black-box setting for very restricted classes of depth-4 circuits [AM07, Sax08, SV09]. For higher depths, efficient black-box algorithms are known only for read-once formulas [SV08, SV09]. Designing efficient PIT algorithms for multilinear circuits is an important problem and our result makes the first step for the important class of depth-4 circuits.

1.1 Overview of Our Algorithm and Proof Technique

To start with, we briefly define the notion of generators and hitting sets for arithmetic circuits which are important for our algorithm. Intuitively, a generator $\mathcal{G}$ for a class of polynomials $\mathcal{M}$, is a function that stretches q independent variables into $n >> q$ dependent variables that can be *plugged* into any polynomial $P \in \mathcal{M}$ without vanishing it. A set $\mathcal{H} \subseteq \mathbb{F}^n$ is

[1] One can consider $\Pi\Sigma\Pi\Sigma$ circuits, however, for identity testing purposes the interesting case is the $\Sigma\Pi\Sigma\Pi$ model.

a hitting set for a class of polynomials $\mathcal{M}$, if for every nonzero polynomial $P \in \mathcal{M}$, there exists $\bar{a} \in \mathcal{H}$, such that $P(\bar{a}) \neq 0$. In identity testing, the role of generators and hitting sets are equivalent. The image of a generator for a class of circuits always contains a hitting set for the same class of circuits. Conversely, given a hitting set for a class of arithmetic circuits, it is fairly easy to construct a generator.

In our algorithm, we use a recursive technique (on the fan-in k of the top Σ gate) to find a generator for $\Sigma\Pi\Sigma\Pi(k)$ circuits and in every stage of the recursion we also construct a hitting set. Recall that the sparsity of the polynomials P_{ij} is bounded by the circuit size s. For $k = 1$, we need to build a generator for product of s-sparse polynomials. It is easy to see that a generator for a single s-sparse polynomial is also a generator for a product of s-sparse polynomials and the construction of a generator for a s-sparse polynomial is well known [KS01].

For $k > 1$, we construct the generator via the following procedure: Let P be a non-zero n-variate polynomial computed by a $\Sigma\Pi\Sigma\Pi(k)$ circuit C of size s and let $\mathcal{G}_{k-1}$ be a generator for $\Sigma\Pi\Sigma\Pi(k-1)$ circuits of size s. We prove that there exist a set $U \subseteq [n]$ of size $\text{poly}(\log s)$ such that a substitution of the generator $\mathcal{G}_{k-1}$ to the variables (indexed by) $[n] \setminus U$ leads to a non-zero polynomial. By going over all possible sets of choice for U, we can produce a small size hitting set for $\Sigma\Pi\Sigma\Pi(k)$ circuits, which in turn is transformed into a generator using the techniques of [SV09]. Notice that the number of choices for U is bounded by $n^{\text{poly}(\log s)}$, which is *quasi-polynomial* in s. Now we justify the existence of U which is enough to justify the correctness of our algorithm. We describe the construction of U in two different cases.

Case I: Assume that there exists some large constant r such that for each i, j, the polynomial P_{ij} depends on at most n/r variables. We show that there exists a subset of the variables $V \subseteq [n]$ of size roughly r/k such that every P_{ij} has at most one variable x_ℓ such that $\ell \in V$. Now, let $\mathcal{G}_{k-1}$ be a generator for $\Sigma\Pi\Sigma\Pi(k-1)$ circuits. Then by suitably fixing the variables whose indices are in $[n] \setminus V$ from the image set of $\mathcal{G}_{k-1}$, we obtain a multilinear depth-3 $\Sigma\Pi\Sigma(k)$ circuit [2]. Using a structural theorem of identically zero depth-3 circuits from [DS06, SS09], our fixing ensures that the resulting depth-3 circuit computes a nonzero polynomial.

Case II: We prove that for any $\Sigma\Pi\Sigma\Pi(k)$ circuit there exists a set $W \subseteq [n]$ of size $\text{poly}(\log s)$ (recall that s is the size of the given circuit) such that the following property holds: For a set $S \subseteq [n]$, let $m(S)$ be the multilinear monomial $\prod_{i \in S} x_i$. Express the polynomial P as $P = \sum_{S \subseteq W} m(S) P_S$ where each polynomial P_S is over $x_{[n] \setminus W}$. We prove that there exists a subset S such that P_S can be computed by a $\Sigma\Pi\Sigma\Pi(k)$ circuit C' and each polynomial P'_{ij} computed in the third level of C' depends only on a small fraction of the total number of variables (which gives a reduction to the first stage). We now explain how to find the set W. Fix r suitably. Let C be the given $\Sigma\Pi\Sigma\Pi(k)$ circuit. Write C as $C = \sum_{i=1}^{k} N_i \cdot A_i$ where $N_i = \prod_j P_{ij}$ such that each P_{ij} depends on at most n/r variables. Similarly, let A_i be the

[2] Notice that after the substitution, at most one variable remains alive in each P_{ij}.

product of the rest of the polynomials under the i-th Π gate. So by definition, each P_{ij} in A_i depends on at least n/r variables. Hence, by multilinearity, each A_i is a product of at most r many P_{ij}-s. We *eliminate* A_i-s by the following process: Consider a variable appearing in some A_i. By either setting the variable to zero or taking a partial derivative with respect to it, we can get rid of at least half of the monomials in A_i. Moreover, we show that such a variable exists with the additional property that both the choices (either setting to zero or taking partial derivative) will result in a non-zero polynomial. Repeating this process at most $\mathcal{O}(\log s)$ times, we can eliminate *all* such A_i.

The final set U that our algorithm considers is defined as $U \triangleq W \cup V$ (the union of the sets found at the first and second stages).

1.2 Organization

We start by giving the required definitions in Sections 2 and 3. We prove our main theorem (Theorem 4.11) in Section 4, showing a construction of a generator for $\Sigma\Pi\Sigma\Pi(k)$ circuits. In Section 4.5 we give as an easy corollary a hitting set for $\Sigma\Pi\Sigma\Pi(k)$ circuits.

2 Preliminaries

For a positive integer n denote $[n] = \{1, \ldots, n\}$. Let $\mathbb{F}$ be a field and $\bar{\mathbb{F}}$ be its algebraic closure. For a polynomial $P(x_1, \ldots, x_n)$, a variable x_i, and $\alpha \in \mathbb{F}$, $P|_{x_i=\alpha}$ is the polynomial resulting after substituting α to the variable x_i. The following definitions are for polynomials $P, Q \in \mathbb{F}[x_1, x_2, \ldots, x_n]$. We say that P *depends* on x_i if there exist $\bar{a} \in \bar{\mathbb{F}}^n$ and $b \in \bar{\mathbb{F}}$ such that:

$$P(a_1, a_2, \ldots, a_{i-1}, a_i, a_{i+1}, \ldots, a_n) \neq P(a_1, a_2, \ldots, a_{i-1}, b, a_{i+1}, \ldots, a_n).$$

We denote $\mathrm{var}(P) \triangleq \{x_i \mid P \text{ depends on } x_i\}$. Intuitively, P depends on x_i if x_i appears when P is listed as a sum of monomials. Given a subset $I \subseteq [n]$ and an assignment $\bar{a} \in \mathbb{F}^n$ we define $P|_{x_I=\bar{a}_I}$ to be the polynomial resulting from substituting a_i to the variable x_i for every $i \in I$. Let P, Q be two non-constant polynomials. We say that P and Q are *similar* and denote $P \sim Q$ if there exist $\alpha, \beta \in \mathbb{F} \setminus \{0\}$ such that $\alpha \cdot P = \beta \cdot Q$. Let $D_i(P, Q)$ be the polynomial defined as follows:

$$D_i(P,Q)(\bar{x}) \triangleq \left| \begin{pmatrix} P & P|_{x_i=0} \\ Q & Q|_{x_i=0} \end{pmatrix} \right| (\bar{x}) = (P \cdot Q|_{x_i=0} - P|_{x_i=0} \cdot Q)(\bar{x})$$

over $\mathbb{F}$. The following is an easy observation.

Observation 2.1. *Let $P, Q \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ be two multilinear polynomials such that $x_i \in \mathrm{var}(P) \cap \mathrm{var}(Q)$ then $P \sim Q$ iff $D_i(P, Q) \equiv 0$.*

2.1 Mappings and Generators for arithmetic circuits

In this section, we formally define the notion of generators and hitting sets for polynomials and describe a few useful properties.

A mapping $\mathcal{G} = (\mathcal{G}^1, \ldots, \mathcal{G}^n) : \mathbb{F}^q \to \mathbb{F}^n$, is a *generator* for the circuit class $\mathcal{M}$ if for every non-zero n-variate polynomial $P \in \mathcal{M}$, it holds that $P(\mathcal{G}) \not\equiv 0$. The image of the map $\mathcal{G}$ is denoted as $\mathrm{Im}(\mathcal{G}) = \mathcal{G}(\bar{\mathbb{F}}^q)$. Ideally, q should be very small compared to n. A set $\mathcal{H} \subseteq \mathbb{F}^n$ is a hitting set for a circuit class $\mathcal{M}$, if for every nonzero polynomial $P \in \mathcal{M}$, there exists $\bar{a} \in \mathcal{H}$, such that $P(\bar{a}) \neq 0$. A generator can also be viewed as a mapping containing a hitting set for $\mathcal{M}$ in its image. That is, for every nonzero $P \in \mathcal{M}$ there exists $\bar{a} \in \mathrm{Im}(\mathcal{G})$ such that $P(\bar{a}) \neq 0$. In [SV09] an efficient method of constructing a generator from a hitting set, for a (relatively) *small* q, is given.

Lemma 2.2 (Lemma 4.8 in [SV09])**.** *Let $|\mathbb{F}| > n$. Given a hitting set $\mathcal{H} \subseteq \mathbb{F}^n$ for a circuit class $\mathcal{M}$ there is an algorithm that in time $\mathrm{poly}(|\mathcal{H}|, n)$ constructs a mapping $L(\bar{y}) : \mathbb{F}^q \to \mathbb{F}^n$, which is a generator for $\mathcal{M}$ with $q \stackrel{\Delta}{=} \lceil \log_n |\mathcal{H}| \rceil$ and the individual degrees of L^i are bounded by $n-1$.*

The following is an immediate and important property of a generator:

Observation 2.3. *Let $P = P_1 \cdot P_2, \cdot \ldots \cdot P_k$ be a product of non-zero polynomials $P_i \in \mathcal{M}$ and let $\mathcal{G}$ be a generator for $\mathcal{M}$. Then $P(\mathcal{G}) \not\equiv 0$.*

At times we would like to use only a partial substitution generator to a polynomial. Given a subset $I \subseteq [n]$ we define the mapping: $\mathcal{G}^I$ as $(\mathcal{G}^I)_i = \mathcal{G}^i$ when $i \in I$ and $(\mathcal{G}^I)_i = x_i$ when $i \notin I$. In addition, we define $P|_{x_I = \mathcal{G}^I}$ to be the polynomial resulting from substituting the function $\mathcal{G}^i$ to the variable x_i for each $i \in I$. The following is an immediate observation:

Observation 2.4. *Let $\mathcal{M}$ be a class of polynomials and let $\mathcal{G}$ be a generator for n-variate polynomials in $\mathcal{M}$. Let $I \subseteq [n]$ and $P \in \mathcal{M}$ be a non-zero polynomial. Then $P|_{x_I = \mathcal{G}^I} \not\equiv 0$. Moreover, there exists $\bar{a} \in \mathrm{Im}\left(\mathcal{G}^I\right)$ such that $P(\bar{a}) \neq 0$.*

2.2 Partial Derivatives

Discrete partial derivatives will play an important role in the analysis of our algorithms.

Definition 2.5. *Let P be an n-variate polynomial over a field $\mathbb{F}$. We define the* discrete partial derivative *of P with respect to x_i as $\frac{\partial P}{\partial x_i} = P|_{x_i=1} - P|_{x_i=0}$. For a non-empty subset $I \subseteq [n]$, $I = \left\{i_1, \ldots, i_{|I|}\right\}$, we define the* iterated *partial derivative with respect to I in the following way:*

$$\partial_I P \stackrel{\Delta}{=} \frac{\partial^{|I|} P}{\partial x_{i_1} \partial x_{i_2} \partial x_{i_3} \cdots \partial x_{i_{|I|}}}.$$

Notice that if P is a multilinear polynomial then this definition coincides with the "analytical" one when $\mathbb{F} = \mathbb{R}$ or $\mathbb{F} = \mathbb{C}$. We now state some easy facts about discrete partial

derivatives that can be easily verified. Let $P \in \mathbb{F}[\bar{x}]$ be a multilinear polynomial. Then, P depends on x_i if and only if $\frac{\partial P}{\partial x_i} \not\equiv 0$. For every i and j, $\frac{\partial^2 P}{\partial x_i \partial x_j} = \frac{\partial}{\partial x_i}(\frac{\partial P}{\partial x_j}) = \frac{\partial^2 P}{\partial x_j \partial x_i}$. For two different variables x_i, x_j, derivative and substitution commutes: $\frac{\partial P}{\partial x_i}|_{x_j=a} = \frac{\partial}{\partial x_i}(P|_{x_j=a})$.

2.3 Known Results

In this section, we recall some known results about sparse polynomials and depth-3 $\Sigma\Pi\Sigma$ circuits which play an important role in the design of our algorithm. A *m-sparse polynomial* is a polynomial with at most m non-zero monomials. Equivalently, it is a polynomial computed by a depth-2 circuit with top fan-in m. Using a result of [KS01], we can construct an efficient generator for sparse polynomials.

Lemma 2.6 (Theorem 10 of [KS01])**.** *In time polynomial in m, n, d and $\log|\mathbb{F}|$, one can output a hitting set $\mathcal{H}$ of cardinality $|\mathcal{H}| = \text{poly}(n, m, d)$ for n-variate m-sparse polynomials of degree d over a field $\mathbb{F}$. If $\mathbb{F} = \mathbb{R}$ then each element of each vector in the set has bit-length at most $\mathcal{O}(\log(nd))$. If $\mathbb{F}$ is a finite field with less than $(nd)^6$ elements, then the elements of the vectors lie in the smallest extension of $\mathbb{F}$ with at least $(nd)^6$ elements; otherwise, the vectors contain just elements of $\mathbb{F}$.*

Using Lemma 2.2, we can construct a generator from the hitting set output by the above result.

Lemma 2.7. *There exists a generator $\mathcal{S}_m \triangleq (\mathcal{S}_m^1, \mathcal{S}_m^2, \ldots, \mathcal{S}_m^n) : \mathbb{F}^q \to \mathbb{F}^n$ for m-sparse multilinear polynomials with the individual degrees of each $\mathcal{S}_m^i$ are bounded by $n-1$ and $q(n,m) = \mathcal{O}(\log_n m)$.*

Proof. Since the degree of a multilinear polynomial is bounded by n, we apply Lemma 2.2 on the hitting set $\mathcal{H}$ output by Lemma 2.6. Note that as $|\mathcal{H}| = \text{poly}(n, m)$, we obtain that $q(n,m) = \mathcal{O}(\log_n m)$. □

The proof technique of our main result involves a reduction from identity testing of a class of depth-4 circuits to depth-3 circuits. Here, we define depth-3 circuits formally and recall some of their relevant properties. A depth-3 $\Sigma\Pi\Sigma(k,d)$ circuit C computes a polynomial of the form

$$C(\bar{x}) = \sum_{i=1}^{k} F_i(\bar{x}) = \sum_{i=1}^{k} \prod_{j=1}^{d_i} L_{ij}(\bar{x})$$

where the $L_{ij}(\bar{x})$-s are linear functions $L_{ij}(\bar{x}) = \sum_\ell a_{ij\ell} x_\ell + a_{ij0}$ with $a_{ij\ell} \in \mathbb{F}$, and $d_i \leq d$. We refer to the F_m-s as the multiplication gates of the circuit. A *subcircuit* of C is defined as a sum of a subset of the multiplication gates in C. Let $\gcd(C) \triangleq \gcd(F_1, F_2, \ldots, F_k)$. We say that a circuit is *simple* if $\gcd(C) = 1$. We say that a circuit is *minimal* if no proper subcircuit of C computes the zero polynomial. Define the *rank* of C, denoted by $\text{rank}(C)$, as the rank of its linear functions, viewed as $(n+1)$-dimensional vectors over $\mathbb{F}^{n+1}$. That is, $\text{rank}(C) \triangleq \dim(\text{span}\{L_{ij}\})$.

A *multilinear* $\Sigma\Pi\Sigma(k,d)$ circuit has the additional requirement that each F_i is a multilinear polynomial. We require an important structural theorem regarding the rank of an identically zero $\Sigma\Pi\Sigma(k,d)$ multilinear circuit.

Theorem 2.8. *[DS06, SS09] There exists an increasing integer function $R(k)$ upper bounded by $\mathcal{O}(k^3\log(k))$ with the following property: Let C be an n-variate multilinear, simple and minimal $\Sigma\Pi\Sigma(k,d)$ circuit computing the zero polynomial. Then* $\mathrm{rank}(C) < R(k)$.

We conclude this section with a well-known lemma concerning polynomials, giving a trivial (yet possibly large) hitting set. A proof can be found in [Alo99].

Lemma 2.9. *Let $P \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ be a polynomial. Suppose that for every $i \in [n]$ the individual degree of x_i is bounded by d_i, and let $S_i \subseteq \mathbb{F}$ be such that $|S_i| > d_i$. We denote $S = S_1 \times S_2 \times \cdots \times S_n$ then $P \equiv 0$ iff $P|_S \equiv 0$.*

3 Depth-4 Multilinear Circuits

In this section, we recall the model of depth-4 multilinear circuits and present a simple structural property of such circuits which is useful for our main result.

Definition 3.1. *A multilinear depth-4 $\Sigma\Pi\Sigma\Pi(k)$ circuit C has four layers of alternating Σ and Π gates (the top Σ gate is at level one) and it computes a polynomial of the form*

$$C(\bar{x}) = \sum_{i=1}^{k} F_i(\bar{x}) = \sum_{i=1}^{k} \prod_{j=1}^{d_i} P_{ij}(\bar{x})$$

where the $P_{ij}(\bar{x})$-s are multilinear polynomials computed by the last two layers of $\Sigma\Pi$ gates of the circuit and are the inputs to the Π gates at the second level. Each multiplication gate F_i computes a multilinear polynomial.

Note that the requirement that the F_i-s compute multilinear polynomials implies that for each i the polynomials $\{P_{ij}\}_{j\in[d_i]}$ are variable-disjoint. It is clear that if the circuit size is s, then the number of monomials in P_{ij} (i.e. its sparsity) is bounded by s. In this paper, we often refer to the polynomials P_{ij} as s-sparse where the sparsity should be understood in terms of the circuit size s. Similar to the case of depth-3 circuits, a (proper) subcircuit of C is defined as the sum of a (proper) subset of multiplication gates of C. Also, a $\Sigma\Pi\Sigma\Pi(k)$ circuit is simple when no P_{ij} appears in all the multiplication gates at the second level. Namely, $\gcd(C) \triangleq \gcd(F_1, \ldots, F_k) = 1$. When C is not simple we define its simplification to be

$$\mathrm{sim}(C) \triangleq C/\gcd(C)\ .$$

Note that $\mathrm{sim}(C)$ is a simple $\Sigma\Pi\Sigma\Pi(k)$ circuit.

Our identity testing algorithm builds on a reduction from identity testing of multilinear $\Sigma\Pi\Sigma\Pi(k)$ circuits to identity testing of a special type of such circuits where for every i, j, $|\mathrm{var}(P_{ij})| \leq n/r$. We call such circuits *r-compressed* circuits. Now we prove an easy structural property of r-compressed circuits which is useful for our algorithm design.

Lemma 3.2. *Let $P \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ be computed by a $\Sigma\Pi\Sigma\Pi(k)$ multilinear r-compressed circuit $C = \sum_{i=1}^{k} \prod_{j=1}^{d_i} P_{ij}(\bar{x})$. Then there exists a set $V \subseteq [n]$ of size $|V| \geq r/k$ such that for each i, j: $|V \cap \mathrm{var}(P_{ij})| \leq 1$.*

Proof. The first element of V can be arbitrarily set to x_1. Let $T_1 \subseteq [n]$ be the set of variables that appear in some P_{ij} along with x_1. As $|\mathrm{var}(P_{ij})| \leq n/r$, we get that $|T_1| \leq k \cdot (\frac{n}{r} - 1)$. Hence, the set $W = [n] \setminus (T_1 \cup \{x_1\})$ is non empty. We pick one arbitrarily (say, the one with lowest index) from W and construct a set analogical to T_1 for it. Due to the size restriction of T_1 (and the other T's), we can continue this process at least r/k times. The set V is the set of these (at least) r/k chosen indices. □

4 Black-Box PIT

In this section we give an efficient black-box PIT algorithm for multilinear $\Sigma\Pi\Sigma\Pi(k)$ circuits. We construct a generator for such circuits, which gives us a small hitting set. We start by describing the construction of a polynomial map which we eventually use as our generator.

4.1 The Construction and Some Easy Properties

In this section we construct a map from $\mathbb{F}^{2t}$ to $\mathbb{F}^n$ with the following property: Its image contains all vectors $\bar{a} \in \mathbb{F}^n$ with at most t non-zero entries. This map will later be put to use in the construction of the generator for depth-4 circuits. We assume that $|\mathbb{F}| > n$ as we are allowed to use elements from an appropriate extension field. Throughout the entire section we fix a set $A = \{\alpha_1, \alpha_2, \ldots, \alpha_n\} \subseteq \mathbb{F}$ of n distinct elements.

Definition 4.1. *For every $i \in [n]$ let $u_i(w) : \mathbb{F} \to \mathbb{F}$ be the i-th Lagrange Interpolation polynomial for the set A. That is, each $u_i(w)$ is polynomial of degree $n-1$ satisfying $u_i(\alpha_j) = 1$ if $i = j$ and zero otherwise. For every $i \in [n]$ and $t \geq 1$ we define $G_t^i(y_1, \ldots, y_t, z_1, \ldots, z_t) : \mathbb{F}^{2t} \to \mathbb{F}$ as*

$$G_t^i(y_1, \ldots, y_t, z_1, \ldots, z_t) \stackrel{\Delta}{=} \sum_{j=1}^{t} u_i(y_j) \cdot z_j \ .$$

Finally, let $G_t(y_1, \ldots, y_t, z_1, \ldots, z_t) : \mathbb{F}^{2t} \to \mathbb{F}^n$ be defined as

$$G_t(y_1, \ldots, y_t, z_1, \ldots, z_t) \stackrel{\Delta}{=} \left(G_t^1, G_t^2, \ldots, G_t^n\right) = \left(\sum_{j=1}^{t} u_1(y_j) \cdot z_j, \sum_{j=1}^{t} u_2(y_j) \cdot z_j, \ldots, \sum_{j=1}^{t} u_n(y_j) \cdot z_j\right).$$

We will use the following immediate observations:

Observation 4.2. *For every $t \geq 1$, it holds $G_t(\bar{y}, \bar{0}) \equiv 0$.*

Observation 4.3. *Denote with $\bar{e}_i \in \{0,1\}^n$ the vector that has 1 in the i-th coordinate and 0 elsewhere. Then*

$$G_{t+1} = G_t + \sum_{i=1}^{n} u_i(y_{t+1}) \cdot z_{t+1} \cdot \bar{e}_i \ .$$

Hence, for every $t \geq 1$ and $\alpha_m \in A$ we have that

$$G_{t+1}|_{y_{t+1}=\,\alpha_m} = G_t + z_{t+1} \cdot \bar{e}_m \ .$$

We now state a simple but crucial property of the generator G that follows from the above observations. (Recall the notation above Observation 2.4).

Observation 4.4. *Let $\ell, t \in \mathbb{N}$, $I \subseteq [n]$ and $|I| \leq t$. Then, it holds that*

$$\mathrm{Im}\left(G_\ell^{[n]\setminus I}\right) \subseteq \mathrm{Im}\left(G_{\ell+t}\right).$$

4.2 A Restricted Case: r-compressed $\Sigma\Pi\Sigma\Pi(k)$ Circuits

In this section we consider a restricted class of $\Sigma\Pi\Sigma\Pi(k)$ circuits: For a fixed r, we assume that the polynomial is computed by a simple r-compressed $\Sigma\Pi\Sigma\Pi(k)$ circuit of size s. Using a generator that works for sparse polynomials as well as for $\Sigma\Pi\Sigma\Pi(k-1)$ circuits of size s, we construct a generator for r-compressed $\Sigma\Pi\Sigma\Pi(k)$ circuits of size s. To do so, the key idea is to use the set $V \subseteq [n]$ that we obtain from Lemma 3.2. Recall that V has the following property: The size of V is at least r/k and for every P_{ij} in C, $|V \cap \mathrm{var}(P_{ij})| \leq 1$. Let $\mathcal{G}_{k-1}$ be a generator for $\Sigma\Pi\Sigma\Pi(k-1)$ circuits and for sparse polynomials of suitable sparsity. In the following lemma we show that if $r = R(k) \cdot k$ then when we restrict the variables in $[n] \setminus V$ to $\mathcal{G}_{k-1}$, we obtain a non-zero polynomial.

Lemma 4.5. *Let $k \geq 2$ and $0 \not\equiv P \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ be computed by a simple, multilinear, $k \cdot R(k)$-compressed $\Sigma\Pi\Sigma\Pi(k)$ circuit of size s. In addition, let $\mathcal{G}_{k-1}$ be a generator for $\Sigma\Pi\Sigma\Pi(k-1)$ circuits of size s and $(2s^2)$-sparse polynomials.[3] Then there exists a subset $V \subseteq \mathrm{var}(P)$ of size $|V| \leq R(k)$ such that*

$$P|_{x_{\mathrm{var}(P)\setminus V} = \mathcal{G}_{k-1}^{\mathrm{var}(P)\setminus V}} = P \circ \mathcal{G}_{k-1}^{\mathrm{var}(P)\setminus V} \not\equiv 0 \ .$$

Proof. Let $C = \sum_{i=1}^{k} \prod_{j=1}^{d_i} P_{ij}(\bar{x})$ be a simple, multilinear, and $(k \cdot R(k))$-compressed $\Sigma\Pi\Sigma\Pi(k)$ circuit of size s, computing P. If C is not minimal, then P can be computed by a $\Sigma\Pi\Sigma\Pi(k-1)$ circuit of size s and we are done (set $V = \emptyset$). Assume w.l.o.g. that C is minimal. Let V be a set promised by Lemma 3.2. We can assume w.l.o.g that $|V| = k \cdot R(k)/k = R(k)$ by keeping $R(k)$ arbitrary indices. Define the set T as $T = [n] \setminus V$.

We now describe a way to find an assignment for x_T such that the resulting polynomial is non-zero. We do so via a reduction to depth-3 circuits. Let $C_1, \ldots, C_{2^k-2}$ be the proper

[3] Namely, $\mathcal{G}_{k-1}$ is a generator for both models.

subcircuits of C (excluding the empty circuit). Clearly they are all $\Sigma\Pi\Sigma\Pi(k-1)$ circuits of size s. For any $P_{i_1j_1}$ and $P_{i_2j_2}$ appearing in C, and a variable x_ℓ such that $\mathrm{var}(P_{i_1j_1}) \cap \mathrm{var}(P_{i_2j_2}) \cap V = \{x_\ell\}$, define the polynomial Q as $D_\ell(P_{i_1j_1}, P_{i_2j_2})$ (recall D_ℓ and its property from Observation 2.1). Let $\mathcal{Q}$ be the set of all non-zero such Q's. The following lemma gives a sufficient condition that a given partial assignment for x_T results in a simple, minimal and nonzero depth-3 circuit.

Lemma 4.6. *Let*

$$\varphi = \prod_{i=1}^{2^k-1} C_i \cdot \prod_{Q\in\mathcal{Q}} Q.$$

Let $\bar{a} \in \bar{\mathbb{F}}^n$ be such that $\varphi|_{x_T=\bar{a}_T} \not\equiv 0$. Then $C|_{x_T=\bar{a}_T}$ is a simple, minimal multilinear $\Sigma\Pi\Sigma(k)$ circuit.

Proof. The minimality of $C|_{x_T=\bar{a}_T}$ is clear since all of the subcircuits of C are factors of φ. If one of them is zero, then so is $\varphi|_{x_T=\bar{a}_T}$. Notice that due to the same reason, no P_{ij} is reduced to zero. In order to prove that $C|_{x_T=\bar{a}_T}$ is simple, notice two following simples facts. First, by the definition of V, for every i, j it holds that $|\mathrm{var}(P_{ij}|_{x_T=\bar{a}_T})| \leq 1$. Second, consider $i_1 \neq i_2$ and j_1, j_2 such that $P_{i_1j_{i_1}} \nsim P_{i_2j_{i_2}}$. If $\mathrm{var}(P_{i_1j_{i_1}}|_{x_T=\bar{a}_T}) \neq \mathrm{var}(P_{i_2j_{i_2}}|_{x_T=\bar{a}_T})$ then we still have $P_{i_1j_{i_1}}|_{x_T=\bar{a}_T} \nsim P_{i_2j_{i_2}}|_{x_T=\bar{a}_T}$. If, on the other hand, $\mathrm{var}(P_{i_1j_{i_1}}|_{x_T=\bar{a}_T}) = \mathrm{var}(P_{i_2j_{i_2}}|_{x_T=\bar{a}_T}) = \{x_\ell\}$, then $D_\ell(P_{i_1j_{i_1}}, P_{i_2j_{i_2}})$ is a factor of φ and so

$$D_\ell(P_{i_1j_{i_1}}, P_{i_2j_{i_2}})|_{x_T=\bar{a}_T} = D_\ell(P_{i_1j_{i_1}}|_{x_T=\bar{a}_T}, P_{i_2j_{i_2}}|_{x_T=\bar{a}_T}) \not\equiv 0\ .$$

Hence, by Observation 2.1, $P_{i_1j_{i_1}}|_{x_T=\bar{a}_T} \nsim P_{i_2j_{i_2}}|_{x_T=\bar{a}_T}$. Since C is itself a simple circuit, the claim follows from those two facts. □

Now we return to the proof of Lemma 4.5. The polynomial φ is a product of $(2s^2)$-sparse polynomials and $\Sigma\Pi\Sigma\Pi(k-1)$ circuits of size s. By Observations 2.3 and 2.4 we get that $\varphi|_{x_T=\mathcal{G}_{k-1}^T} \not\equiv 0$. It follows that there exists some $\bar{a} \in \mathrm{Im}\,(\mathcal{G}_{k-1})$ for which $C|_{x_T=\bar{a}_T}$ is a simple, minimal, and multilinear $\Sigma\Pi\Sigma(k)$ circuit. Notice now that $C|_{x_T=\bar{a}_T}$ contains $R(k)$ variables (the previous proof shows that all the variables in V 'survived') and any linear function appearing in it contains only one variable. Hence, the rank of $C|_{x_T=\bar{a}_T}$ is $R(k)$. By the definition of $R(k)$ (Theorem 2.8) it cannot be a zero circuit. We thus proved that $P|_{x_T=\mathcal{G}_{k-1}^T} \not\equiv 0$. □

4.3 A Reduction to r-compressed Multilinear $\Sigma\Pi\Sigma\Pi(k)$ Circuits

In this section we prove a structural theorem for multilinear $\Sigma\Pi\Sigma\Pi(k)$ circuits. This theorem enables us to reduce the identity testing of multilinear $\Sigma\Pi\Sigma\Pi(k)$ circuits to the identity testing of r-compressed multilinear $\Sigma\Pi\Sigma\Pi(k)$ circuits for any $r > 0$. Roughly, the theorem says that there exists a small set of variables W with the following property. Let $P = \sum_{T\subseteq W} m(T)F_T$, where F_T are polynomials defined over the variables $[n] \setminus W$ and $m(T) = \prod_{i\in T} x_i$. Then there exists T such that F_T can be computed by an r-compressed $\Sigma\Pi\Sigma\Pi(k)$ circuit of size s. Now we state the theorem formally.

Theorem 4.7. *Let P be an n-variate polynomial computable by a $\Sigma\Pi\Sigma\Pi(k)$ circuit of size s. Let $r > 0$ be a parameter. Then there exists a set W of size $|W| \leq 2\log n \cdot \log s \cdot kr$ for which the following holds: Write*

$$P = \sum_{T \subseteq W} m(T) F_T$$

where the F_T's are polynomials independent of the variables in W. Then there exists at least one set $T \subseteq W$ for which $F_T = Q \cdot H$ where Q is a product of s-sparse polynomials and H is computable by a simple, r-compressed $\Sigma\Pi\Sigma\Pi(k)$ circuit of size s.

An alternative view of the theorem states that there exist two sets I, J of the following properties: If we set the variables of J to zero and take a partial derivative w.r.t. I (i.e. compute $\partial_I P|_{x_J=\bar{0}_J}$) then we get an r-compressed $\Sigma\Pi\Sigma\Pi(k)$ circuit of size s, multiplied by s-sparse polynomials. The set I corresponds to the variables in the monomial (i.e. T) and J to the variables of W outside the monomial (i.e. $W \setminus T$). We find this alternative view more convenient for the purpose of proving the theorem.

Lemma 4.8. *Let $n, s, r, k > 1$ be integers. Let $P \not\equiv 0$ be an n variate polynomial computable by a multilinear $\Sigma\Pi\Sigma\Pi(k)$ circuit C of size s. Then there exist disjoint subsets $I, J \subseteq [n]$ such that $|I| + |J| \leq 2kr\log(s)$ and $\partial_I P|_{x_J=\bar{0}_J}$ is a non-zero polynomial computed by a $\Sigma\Pi\Sigma\Pi(k)$ circuit C_{IJ} with at least one of the following properties:*

- $\mathrm{var}\left(\mathrm{sim}(C_{IJ})\right) < \mathrm{var}\left(\mathrm{sim}(C)\right)/2$ *(recall the definition of* $\mathrm{sim}(C)$ *from Section 3).*
- $\mathrm{sim}(C_{IJ})$ *is an r-compressed $\Sigma\Pi\Sigma\Pi(k)$ circuit of size s.*

Proof. Assume that C itself does not meet any of the needed conditions. Let $\mathrm{sim}(C) = \sum_{j=1}^{k} M_j$ where each M_j is a multiplication gate of a $\Sigma\Pi\Sigma\Pi(k)$ circuit of size s. Write $M_j = N_j \cdot A_j$ where N_j is a product of s-sparse polynomials which are defined on at most $|\mathrm{var}(\mathrm{sim}(C))|/2r$ variables and $A_j = M_j/N_j$. Clearly, A_j is a product of s-sparse polynomials, each defined on at least $|\mathrm{var}(\mathrm{sim}(C))|/2r$ variables. Hence, due to the multilinearity of M_j, A_j must be s^{2r}-sparse. Let $\mathrm{mon}(A)$ denote the number of monomials in a polynomial A. Let $\Phi(C) \stackrel{\Delta}{=} \sum_{j=1}^{k} \log\left(\mathrm{mon}\left(A_j\right)\right)$ be a potential function that will aid us during the proof. We assume w.l.o.g. that $\Phi(C)$ is minimal w.r.t. all possible $\Sigma\Pi\Sigma\Pi(k)$ circuits of size s computing P. Notice that $\Phi(C) \leq 2kr\log(s)$.

Let $I_0 = J_0 = \emptyset$, $P_0 = P$ and $A_{j,0} = A_j$ for each $j \in [k]$. We now describe an algorithm that produces the required sets I, J. The algorithm is composed of enumerated steps, starting from 1. At each step, we add a single element either to I or J. Denote by I_ℓ and J_ℓ the sets at the end of step ℓ. Correspondingly, define $P_\ell = \partial_{I_\ell} P|_{x_{J_\ell}=\bar{0}_{J_\ell}}$. Also, define

$$C_\ell = \gcd(C_\ell) \cdot \mathrm{sim}(C_\ell) = \gcd(C_\ell) \cdot \sum_{j=1}^{k} N_{j,\ell} A_{j,\ell}$$

where the $N_{j,\ell}$'s are s-sparse polynomials that rely on at most $|\text{var}(\text{sim}(C))|/2r$ variables (notice that we used C and not C_ℓ) and $C_\ell \equiv P_\ell$. Let $\Phi_C(C_\ell) \triangleq \sum_{j=1}^{k} \log(\text{mon}(A_{j,\ell}))$.[4] Define C_ℓ as the circuit achieving the minimal Φ_C among all size s $\Sigma\Pi\Sigma\Pi(k)$ circuits computing P_ℓ.

We now describe the process of adding a single variable to I or J. The idea is to take a variable appearing in some $A_{j,\ell}$ and add it to a set that will result in a maximal reduction to the monomials of $A_{j,\ell}$. One of the choices must reduce the number of monomials by a factor of 2 and thus reduce the Φ_C function by at least 1. This is since adding the variable to I means keeping only monomials in which it appears and adding it to J means keeping only the monomials in which it does not appear. The problem is to ensure that the resulting circuit computes a non-zero polynomial. The following lemma guarantees the existence of a variable for which neither action would result in a non-zero polynomial.

Lemma 4.9. *Let $\ell \geq 0$. Assume that $P_\ell \not\equiv 0$ and that C_ℓ does not meet the conditions of Lemma 4.8. Then there exist some $i \in [n]$ and $j \in [k]$ such that $A_{j,\ell}$ and P_ℓ depend on x_i and x_i is not a factor of P_ℓ.*

Proof. Assume that the claim is false. We have one of the following cases:
case 1: For some i, j, $A_{j,\ell}$ depends on x_i and P_ℓ does not. We can replace $A_{j,\ell}$ with $A_{j,\ell}|_{x_i=0}$ and result in a circuit C' computing the same polynomial P with a $\Phi(C') < \Phi(C_\ell)$. This is a contradiction to the minimality of C_ℓ.

case 2: All the $A_{j,\ell}$'s are constant. That is: $C_\ell = \gcd(C_\ell) \cdot \sum_{j=1}^{k} N_{j,\ell}$. In this case, either $\text{var}(\text{sim}(C_\ell)) < \text{var}(\text{sim}(C))/2$ or $\text{sim}(C_\ell)$ is an r-compressed $\Sigma\Pi\Sigma\Pi(k)$ circuit of size s. Either way this is a contradiction.
case 3: There exists a variable x_i in A_ℓ that divides P_ℓ. Let $C' \triangleq C|_{x_i=1} \cdot x_i$. Then $P_\ell \equiv C'$ and $\Phi_C(C') < \Phi_C(C_\ell)$. This is a contradiction to the minimality of C_ℓ w.r.t. Φ_C. □

We return to the proof of lemma 4.8. Due to the lemma, there exist some x_i that appears in some $A_{j,\ell}$ where both $\frac{\partial P_\ell}{\partial x_i}$ and $P_\ell|_{x_i=0}$ are non-zero polynomials. Clearly, one of these choices for $P_{\ell+1}$ results in a non-zero polynomial for which $\Phi_C(C_{\ell+1}) \leq \Phi_C(C_\ell) - 1$. Since Φ_C is always non-negative, after at most $2kr\log(s)$ (the initial value for Φ) steps, we get the required circuit. □

By repeating Lemma 4.8 at most $\log n$ times, we get Theorem 4.7 (indeed, in each step if we do not have the conclusion of 4.7 then $|\text{var}(\text{sim}(C_\ell))|$ is reduced by a factor of 2).

4.4 Rounding the Components Together

In the previous sections we found that there exists a small set of variables W and an additional disjoint small set of variables V with the following properties: When picking the

[4] We use a different notation (Φ_C and not Φ) as $\text{var}(\text{sim}(C))$ affects the definition of the $A_{j,\ell}$-s.

variables outside of W and V from a generator for $\Sigma\Pi\Sigma\Pi(k-1)$ circuits and for $(2s^2)$-sparse polynomials, we get a non-zero polynomial. The following is the formal claim:

Lemma 4.10. *Let $k \geq 2$ and $P \not\equiv 0 \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ be computed by a multilinear $\Sigma\Pi\Sigma\Pi(k)$ circuit of size s. In addition, let $\mathcal{G}_{k-1}$ be a generator for $\Sigma\Pi\Sigma\Pi(k-1)$ circuits of size s and $(2s^2)$-sparse polynomials. Then there exists a subset $U \subseteq [n]$, depending only on P (i.e. U does not depend on the generator), of size $|U| \leq 3k^2 R(k)\log(s)\log(n)$ such that $P|_{x_{[n]\setminus U}=\mathcal{G}_{k-1}^{[n]\setminus U}} \not\equiv 0$ for any such generator.*

Proof. Let T_0 and W be the sets guaranteed by Theorem 4.7. Namely, when writing $P = \sum_{T\subseteq W} m(T)F_T$, we have that $F_{T_0} = Q \cdot H$ where Q is a product of s-sparse polynomials and H can be computed by a simple, multilinear $(k \cdot R(k))$-compressed $\Sigma\Pi\Sigma\Pi(k)$ circuit. Note that Q, H are defined on $[n] \setminus W$ (that is $\mathrm{var}(Q) \cup \mathrm{var}(H) \subseteq [n] \setminus W$). By Lemma 4.5 there exists a subset $V \subseteq \mathrm{var}(H)$ of size $|V| \leq R(k)$ such that

$$H|_{x_{\mathrm{var}(H)\setminus V}=\mathcal{G}_{k-1}^{\mathrm{var}(H)\setminus V}} \not\equiv 0 \,.$$

Let $U \triangleq V \cup W$. It holds that

$$H|_{x_{[n]\setminus U}=\mathcal{G}_{k-1}^{[n]\setminus U}} \not\equiv 0 \,.$$

As Q is a product of s-sparse polynomials we get, by Observations 2.3 and 2.4, that

$$Q|_{x_{[n]\setminus U}=\mathcal{G}_{k-1}^{[n]\setminus U}} \not\equiv 0 \,.$$

It follows that under the restriction $x_{[n]\setminus U} = \mathcal{G}_{k-1}^{[n]\setminus U}$, F_{T_0} is a non-zero polynomial. As we did not substitute anything to the variables in W the claim clearly follows. □

We now establish the generator for $\Sigma\Pi\Sigma\Pi(k)$ circuits. This is our main theorem and it guarantees that we get the requires black-box algorithm. In particular Theorem 1 is an immediate corollary.

Theorem 4.11 (Main)**.** *Let $P \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ be a non-zero polynomial computed by a multilinear $\Sigma\Pi\Sigma\Pi(k)$ circuit of size s. Then for every $\ell \geq 3k^3 R(k)\log(s)\log(n)$ it holds that $P\left(G_\ell(\bar{y}, \bar{z}) + \mathcal{S}_{2s^2}(\bar{w})\right) \not\equiv 0$, where $\bar{y}, \bar{z}$ and $\bar{w}$ are new sets of variables.*

Proof. We prove the claim by induction on k. For $k = 1$ we note that P is a product of $(2s^2)$-sparse polynomials. By definition of $\mathcal{S}_{2s^2}$ (recall Lemma 2.7), and Observations 2.3 and 4.2, we get that $P(G_\ell + \mathcal{S}_{2s^2}) \not\equiv 0$ and the claim follows. Assume that $k \geq 2$. Let $U \subseteq [n]$ be the subset guaranteed by Lemma 4.10. By the induction hypothesis, we get that for $v = \lceil 3(k-1)^3 R(k-1)\log(s)\log(n)\rceil$ the mapping $G_v(\bar{y}, \bar{z}) + \mathcal{S}_{2s^2}(\bar{w})$ is a generator for both $\Sigma\Pi\Sigma\Pi(k-1)$ circuits and for $(2s^2)$-sparse polynomials. ¿From Lemma 4.10 and Observation 2.4 it follows that $\mathrm{Im}\left(G_v^{[n]\setminus U} + \mathcal{S}_{2s^2}^{[n]\setminus U}\right)$ contains a point $\bar{a}$ for which $P(\bar{a}) \neq 0$. Since $|U| \leq 3k^2 R(k)\log(s)\log(n) \leq \ell - v$, Observation 4.4 gives

$$\mathrm{Im}\left(G_v^{[n]\setminus U} + \mathcal{S}_{2s^2}^{[n]\setminus U}\right) \subseteq \mathrm{Im}\left(G_v^{[n]\setminus U} + \mathcal{S}_{2s^2}\right) \subseteq \mathrm{Im}\left(G_\ell + \mathcal{S}_{2s^2}\right)$$

and thus $\bar{a} \in \mathrm{Im}\left(G_\ell(\bar{y}, \bar{z}) + \mathcal{S}_{2s^2}(\bar{w})\right)$ and the claim holds. □

4.5 An explicit hitting set

The hitting set for $\Sigma\Pi\Sigma\Pi(k)$ circuit is an immediate corollary of Theorem 4.11. Basically, as $G_\ell(\bar{y},\bar{z}) + \mathcal{S}_{2s^2}(\bar{w})$ are (relatively) low degree polynomials defined on $m = \mathcal{O}(k^3 R(k)\log(s)\log(n))$ many variables, we can simply evaluate $P \circ (G_\ell(\bar{y},\bar{z}) + \mathcal{S}_{2s^2}(\bar{w}))$ on all inputs from E^m where $E \subseteq \mathbb{F}$ is a set of size $\text{poly}(n)$. Algorithm 1 follows exactly this intuition and produces the hitting set.

Input: $n, k, s \in \mathbb{N}$.
Output: A set $\mathcal{H}$
Let $W \subseteq \mathbb{F}$ be of size $|W| = n^2$;
Let $\ell \triangleq \lceil 3k^3 R(k)\log(s)\log(n)\rceil$ where $R(k)$ is defined in Theorem 2.8;
Let $q \triangleq q(n, 2s^2)$ as defined in Lemma 2.7;
Initialize $\mathcal{H} = \emptyset$;
foreach $\bar{a}, \bar{b} \in W^\ell$ *and* $\bar{c} \in W^q$ **do**
  Evaluate $G_\ell(\bar{a},\bar{b}) + \mathcal{S}_{2s^2}(\bar{c})$ and add it to $\mathcal{H}$.
end

Algorithm 1: Construction of a hitting set for $\Sigma\Pi\Sigma\Pi(k)$ circuits

Theorem 4.12. *Let $n, s, k > 0$. Algorithm 1, given n, s, k as input runs in $n^{\mathcal{O}(k^3 R(k)\log^2 s)} = n^{\mathcal{O}(k^6 \log(k)\log^2 s)}$ time. The set $\mathcal{H}$ it produces is of size $n^{\mathcal{O}(k^3 R(k)\log^2 s)} = n^{\mathcal{O}(k^6\log(k)\log^2 s)}$ and is a hitting set for n-variate polynomials that can be computed by a $\Sigma\Pi\Sigma\Pi(k)$ circuit of size s.*

Proof. Let $P \in \mathbb{F}[x_1, x_2, \ldots, x_n]$ be a polynomial computed by a multilinear $\Sigma\Pi\Sigma\Pi(k)$ circuit of size s. Let $\mathcal{H}$ be the set given by Algorithm 1. We claim that $P \equiv 0$ if and only if $P|_{\mathcal{H}} \equiv 0$. If $P \equiv 0$ then the claim is trivial. If $P \not\equiv 0$, by Theorem 4.11 we get that $P(G_\ell + \mathcal{S}_{2s^2}) \not\equiv 0$. According to their definition, the degrees of all the output variables of G_ℓ and $\mathcal{S}_{2s^2}$ are at most $n-1$. Therefore, the degrees of the variables in $P(G_\ell + \mathcal{S}_{2s^2})$ are bounded by $(n-1)n < n^2$. Since $P(G_\ell + \mathcal{S}_{2s^2}) \not\equiv 0$, Lemma 2.9 implies that $P|_{\mathcal{H}} \not\equiv 0$.

We now bound the size of $\mathcal{H}$ and the time required to construct it. From their definition, G_ℓ depends on 2ℓ variables and $\mathcal{S}_{2s^2}$ depends on $\mathcal{O}(\log_n s)$ variables. Hence,

$$|\mathcal{H}| \leq n^{4\ell + 2q(n,2s^2)} = n^{\mathcal{O}(k^3 R(k)\log s \log n \cdot \log_n s)} = n^{\mathcal{O}(k^3 R(k)\log^2 s)}.$$

The time required to construct $\mathcal{S}_{2s^2}$ and G_ℓ is polynomial in n, s. The time to evaluate $(G_\ell + \mathcal{S}_{2s^2})$ on a point from $W^{2\ell + q(n,2s^2)}$ is polynomial in n. Hence, the time to construct $\mathcal{H}$ is $|\mathcal{H}| \cdot (ns)^{\mathcal{O}(1)} = n^{\mathcal{O}(k^3 R(k)\log^2 s)}$. □

5 Conclusion

Derandomizing the Polynomial Identity Testing problem for depth-4 arithmetic circuits is an outstanding open problem in complexity theory [AV08]. Any efficient derandomized

algorithm for depth-4 circuits will imply strong lower bounds [KI03, Agr05]. So far, the progress in depth-4 identity testing is very limited [AM07, Sax08, SV09]. In this paper, we improve the situation by giving a quasi-polynomial time *black-box* identity testing algorithm for depth-4 multilinear circuits with bounded fan in top gate. Our algorithm is based on new structural theorems about such circuits.

In identity testing and explicit lower bound proofs, multilinear circuits have already received significant attention from the community [DS06, KS08, SV08, SV09, Raz04a, Raz04b, RSY08, RY08]. In [Raz04a], Raz asked whether one could design efficient identity testing algorithms for multilinear formulas. The best algorithms today are for sums of read-once formulas [SV09] and for set-multilinear depth 3 formulas (non black-box) [RS05]. For depth-4 multilinear circuits with bounded fan in top gate, our result gives the first efficient identity testing algorithm.

It will be very interesting to generalize our result for non-multilinear circuits with bounded fan in top gate. Another problem is to give a deterministic *polynomial time* identity testing algorithm for such circuits in *non black-box* model.

References

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

[AJMV98] E. Allender, J. Jiao, M. Mahajan, and V. Vinay. Non-commutative arithmetic circuits: Depth reduction and size lower bounds. *Theor. Comput. Sci.*, 209(1-2):47–86, 1998.

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

[ALM+98] S. Arora, C. Lund, R. Motwani, M. Sudan, and M. Szegedy. Proof verification and the hardness of approximation problems. *JACM*, 45(3):501–555, 1998.

[Alo99] N. Alon. Combinatorial nullstellensatz. *Combinatorics, Probability and Computing*, 8:7–29, 1999.

[AM07] V. Arvind and P. Mukhopadhyay. The monomial ideal membership problem and polynomial identity testing. In *Proceedings of the 18th International Symposium on Algorithm and Computation (ISAAC)*, pages 800–811, 2007.

[AV08] M. Agrawal and V. Vinay. Arithmetic circuits: A chasm at depth four. In *Proceedings of the fourty ninth Annual FOCS*, pages 67–75, 2008.

[DS06] 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, 2006.

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

[KS01] A. Klivans and D. Spielman. Randomness efficient identity testing of multivariate polynomials. In *Proceedings of the 33rd Annual STOC*, pages 216–223, 2001.

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

[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 CCC*, pages 280–291, 2008.

[KS09] N. Kayal and S. Saraf. Blackbox polynomial identity testing for depth 3 circuits. *Electronic Colloquium on Computational Complexity (ECCC)*, (32), 2009.

[LFKN92] C. Lund, L. Fortnow, H. Karloff, and N. Nisan. Algebraic methods for interactive proof systems. *JACM*, 39(4):859–868, 1992.

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

[MVV87] K. Mulmuley, U. Vazirani, and V. Vazirani. Matching is as easy as matrix inversion. *Combinatorica*, 7(1):105–113, 1987.

[Raz04a] R. Raz. Multi-linear formulas for permanent and determinant are of super-polynomial size. In *Proceedings of the 36th Annual STOC*, pages 633–641, 2004.

[Raz04b] R. Raz. Multilinear $NC_1 \neq$ Multilinear NC_2. In *Proceedings of the 45th Annual FOCS*, pages 344–351, 2004.

[RS05] R. Raz and A. Shpilka. Deterministic polynomial identity testing in non commutative models. *Computational Complexity*, 14(1):1–19, 2005.

[RSY08] R. Raz, A. Shpilka, and A. Yehudayoff. A lower bound for the size of syntactically multilinear arithmetic circuits. *SIAM J. on Computing*, 38(4):1624–1647, 2008.

[RY08] R. Raz and A. Yehudayoff. Lower bounds and separations for constant depth multilinear circuits. In *IEEE Conference on Computational Complexity*, pages 128–139, 2008.

[Sax08] N. Saxena. Diagonal circuit identity testing and lower bounds. In *ICALP (1)*, pages 60–71, 2008.

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

[Sha90] Adi Shamir. IP=PSPACE. In *Proceedings of the Thirty First Annual Symposium on Foundations of Computer Science*, pages 11–15, 1990.

[SS09] N. Saxena and C. Seshadhri. An almost optimal rank bound for depth-3 identities. In *Proceedings of the 24rd Annual CCC*, 2009.

[SV08] A. Shpilka and I. Volkovich. Read-once polynomial identity testing. In *Proceedings of the 40th Annual STOC*, pages 507–516, 2008.

[SV09] A. Shpilka and I. Volkovich. Improved polynomial identity testing for read-once formulas. In *APPROX-RANDOM*, pages 700–713, 2009. Full version available at http://www.cs.technion.ac.il/ shpilka/publications/PROF.pdf.

[VSBR83] L. G. Valiant, S. Skyum, S. Berkowitz, and C. Rackoff. Fast parallel computation of polynomials using few processors. *SIAM J. on Computing*, 12(4):641–644, November 1983.

[Zip79] R. Zippel. Probabilistic algorithms for sparse polynomials. In *Symbolic and algebraic computation*, pages 216–226. 1979.

ECCC ISSN 1433-8092

http://eccc.hpi-web.de