SIAM J. COMPUT.
Vol. 36, No. 5, pp. 1404–1434

LOCALLY DECODABLE CODES WITH TWO QUERIES AND POLYNOMIAL IDENTITY TESTING FOR DEPTH 3 CIRCUITS*

ZEEV DVIR† AND AMIR SHPILKA‡

Abstract. In this work we study two, seemingly unrelated, notions. *Locally decodable codes* (LDCs) are codes that allow the recovery of each message bit from a constant number of entries of the codeword. *Polynomial identity testing* (PIT) is one of the fundamental problems of algebraic complexity: we are given a circuit computing a multivariate polynomial and we have to determine whether the polynomial is identically zero. We improve known results on LDCs and on polynomial identity testing and show a relation between the two notions. In particular we obtain the following results: (1) We show that if $E : \mathbb{F}^n \mapsto \mathbb{F}^m$ is a linear LDC with two queries, then $m = \exp(\Omega(n))$. Previously this was known only for fields of size $\ll 2^n$ [O. Goldreich et al., *Comput. Complexity*, 15 (2006), pp. 263–296]. (2) We show that from every depth 3 arithmetic circuit ($\Sigma\Pi\Sigma$ circuit), $\mathcal{C}$, with a bounded (constant) top fan-in that computes the zero polynomial, one can construct an LDC. More formally, assume that $\mathcal{C}$ is minimal (no subset of the multiplication gates sums to zero) and simple (no linear function appears in all the multiplication gates). Denote by d the degree of the polynomial computed by $\mathcal{C}$ and by r the rank of the linear functions appearing in $\mathcal{C}$. Then we can construct a linear LDC with two queries that encodes messages of length $r/\text{polylog}(d)$ by codewords of length $O(d)$. (3) We prove a structural theorem for $\Sigma\Pi\Sigma$ circuits, with a bounded top fan-in, that compute the zero polynomial. In particular we show that if such a circuit is simple, minimal, and of polynomial size, then its rank, r, is only polylogarithmic in the number of variables (a priori it could have been linear). (4) We give new PIT algorithms for $\Sigma\Pi\Sigma$ circuits with a bounded top fan-in: (a) a deterministic algorithm that runs in quasipolynomial time, and (b) a randomized algorithm that runs in polynomial time and uses only a polylogarithmic number of random bits. Moreover, when the circuit is multilinear, our deterministic algorithm runs in polynomial time. Previously deterministic subexponential time algorithms for PIT in bounded depth circuits were known only for depth 2 circuits (in the black box model) [D. Grigoriev, M. Karpinski, and M. F. Singer, *SIAM J. Comput.*, 19 (1990), pp. 1059–1063; M. Ben-Or and P. Tiwari, *Proceedings of the* 20*th Annual ACM Symposium on Theory of Computing*, ACM Press, New York, 1988, pp. 301–309; A. R. Klivans and D. Spielman, *Proceedings of the* 33*rd Annual ACM Symposium on Theory of Computing*, ACM Press, New York, 2001, pp. 216–223]. In particular, for the special case of depth 3 circuits with three multiplication gates our result resolves an open question asked by Klivans and Spielman.

Key words. derandomization, polynomial identity test, arithmetic circuits, depth 3, locally decodable codes

AMS subject classifications. 68Q25, 94B65

DOI. 10.1137/05063605X

1. Introduction.

Locally decodable codes (LDCs) are error correcting codes that allow the recovery of each symbol of the message from a constant number of entries of the codeword. Polynomial identity testing (PIT) is one of the fundamental problems of algebraic complexity: we are given a circuit computing a multivariate polynomial, and we have to determine whether the polynomial is identically zero. In this paper we show a relation between these two notions—roughly, from every depth 3 circuit which is identically zero, one can construct an LDC. Using this relation and a new lower bound on LDCs, we devise new PIT algorithms for depth 3 circuits.

*Received by the editors March 14, 2006; accepted for publication (in revised form) July 14, 2006; published electronically January 26, 2007.

http://www.siam.org/journals/sicomp/36-5/63605.html

†Department of Computer Science, Weizmann Institute of Science, Rehovot, Israel (zeev.dvir@weizmann.ac.il).

‡Faculty of Computer Science, Technion, Haifa, Israel (shpilka@cs.technion.ac.il).

1.1. Locally decodable codes. LDCs are error correcting codes that allow the recovery of each symbol of the message, from a corrupted codeword, by looking at only a constant number of entries of the corrupted word. Roughly, a (q, δ, ϵ)-*locally decodable code* encodes $x \in \mathbb{F}^n$ to $E(x) \in \mathbb{F}^m$ such that for each index $i \in [n]$, x_i can be recovered from $E(x)$ with probability[1] $> \frac{1}{|\mathbb{F}|} + \epsilon$ by reading only q (random) entries, even if $E(x)$ was corrupted in δm positions.

LDCs have many applications—they are related to private information retrieval (PIR) schemes [13, 26, 18], and they can be used for amplification of hardness [19, 20, 3] and for the construction of hard-core predicates for one-way permutations [30, 15]. (See [49] for a survey on LDCs.)

The notion of LDCs was explicitly discussed in [4] and explicitly defined in [26]. Implicit constructions of local decoders can be found in the context of random self-reducibility and self-correcting computations (see, e.g., [32, 6, 16, 17, 15]). There are two main questions related to LDCs: finding explicit constructions and proving limits of such constructions (i.e., proving lower bounds on the length of the encoding). Explicit constructions were given by [4, 7, 8]. The best current construction is due to Beimel et al. [8], who gave an LDC with q queries of length $m = \exp(n^{O(\log\log q/q \log q)})$.

The problem of proving lower bounds was first studied by Katz and Trevisan [26], who proved that for every LDC with q queries, the length of the codeword, m, is at least $n^{1+\frac{1}{q-1}}$. This is currently the best lower bound for general LDCs (see also [14]). It is a very challenging open question to give tight lower bounds (or upper bounds) on the length of LDCs. Due to the difficulty of the problem many works focused on the case of codes with two queries ($q = 2$). Exponential lower bounds were first proved for linear codes [18, 37] and then, by techniques from quantum computation, for nonlinear codes over $GF(2)$ [28]. The bound of Goldreich et al. [18] actually holds for linear LDCs with two queries over any finite field, namely, that m is at least $2^{\Omega(n)-\log(|\mathbb{F}|)}$, where $\mathbb{F}$ is the underlined field. This result is (nearly) tight when the field is of constant size; however, it gives no significant bound for infinite fields.

1.2. Polynomial identity testing. PIT is a fundamental problem in algebraic complexity: we are given a multivariate polynomial (in some representation) over some field $\mathbb{F}$, and we have to determine whether it is identically zero.[2] The importance of this problem follows from its many applications: algorithms for primality testing [1, 2], for deciding if a graph contains a perfect matching [33, 34, 11], and more, are based on reductions to the PIT problem. (See the introduction of [31] for more applications.)

Determining the complexity of PIT is one of the greatest challenges of theoretical computer science. It is one of a few problems (and in some sense PIT is the most general problem) for which we have *coRP* algorithms but no deterministic subexponential time algorithms. Kabanets and Impagliazzo [25] suggested an explanation for the lack of algorithms. They showed that efficient deterministic algorithms for PIT imply that $NEXP$ does not have polynomial size arithmetic circuits. Specifically, if PIT has deterministic polynomial time algorithms, then either the permanent cannot be computed by polynomial size arithmetic circuits or $NEXP \not\subset P/\text{poly}$.

The first randomized algorithm for PIT was discovered independently by Schwartz [42] and Zippel [50]. Their well-known algorithm simply evaluates the polynomial at a random point and accepts iff the polynomial vanishes at the point. If the polynomial

[1] If $\mathbb{F}$ is infinite, then the probability of success is $> \epsilon$.

[2] Note that we want the polynomial to be identically zero and not just to be equal to the zero function. For example, $x^2 - x$ is the zero function over $GF(2)$ but not the zero polynomial.

is of degree d and each variable is randomly chosen from a domain S, then the error probability is bounded by $d/|S|$. Two kinds of works followed the Schwartz–Zippel algorithm: randomized algorithms that use fewer random bits [12, 31, 1] and algorithms for restricted models of arithmetic circuits. In [22, 9, 29] polynomial time deterministic PIT algorithms for depth 2 arithmetic circuits were given. More recently, [41] gave a polynomial time PIT algorithm for noncommutative formulas. All algorithms, with the exception of [1, 41], are black box algorithms. That is, these algorithms do not have access to a circuit computing the polynomial, and they can evaluate it only on different inputs (as in the Schwartz–Zippel algorithm).

A result of a different nature was proved by Kabanets and Impagliazzo [25]. They designed a deterministic quasi-polynomial time algorithm based on unproved hardness assumptions. Namely, in Theorem 7.7 of [25] it is shown that if there is a family $\{p_n\}$ of exponential time computable polynomials in n variables over $\mathbb{Z}$ such that the arithmetic circuit complexity of p_n is $\exp(n^{\Omega(1)})$, then there is an $\exp(\text{poly}(\log n))$ time algorithm for identity testing for any polynomial size arithmetic circuit that computes polynomials with at most a polynomial degree and polynomial size coefficients.

1.3. Depth 3 arithmetic circuits. Proving lower bounds for general arithmetic circuits is the greatest challenge of algebraic complexity. Unfortunately, except for the lower bounds of Strassen [47] and Baur and Strassen [5], no lower bounds are known for general arithmetic circuits. Due to the difficulty of the problem, research focused on restricted models such as monotone circuits and bounded depth circuits. Exponential lower bounds were proved on the size of monotone arithmetic circuits [43, 24], and linear lower bounds were proved on their depth [44, 48]. However, unlike the situation in the Boolean case, only weak lower bounds were proved for bounded depth arithmetic circuits [38, 40]. Thus, a more restricted model was considered—the model of depth 3 arithmetic circuits (also known as $\Sigma\Pi\Sigma$ circuits). A $\Sigma\Pi\Sigma$ circuit computes a polynomial of the form

$$\mathcal{C} = \sum_{i=1}^{k} \prod_{j=1}^{d_i} L_{ij}(x), \tag{1}$$

where the L_{ij} are linear functions. Grigoriev and Karpinski [21] and Grigoriev and Razborov [23] proved exponential lower bounds on the size of $\Sigma\Pi\Sigma$ circuits computing the permanent and determinant over finite fields. Over infinite fields exponential lower bounds are known only for the restricted models of *multilinear*[3] $\Sigma\Pi\Sigma$ circuits and for *homogeneous* $\Sigma\Pi\Sigma$ circuits [35, 36]. For general $\Sigma\Pi\Sigma$ circuits over infinite fields only the quadratic lower bound of [46] is known. Thus, proving exponential lower bounds for $\Sigma\Pi\Sigma$ circuits over $\mathbb{C}$ is a major open problem in arithmetic circuit complexity.

In this work we are interested in the problem of PIT for depth 3 circuits. As mentioned earlier there are no efficient PIT algorithms for arithmetic circuits, even if we just consider bounded depth circuits. Thus, finding efficient algorithms for PIT in $\Sigma\Pi\Sigma$ circuits seems like the first step toward proving more general results.

1.4. Our results. Lower bounds for linear LDCs with two queries. We study linear LDCs with two queries over arbitrary fields and prove lower bounds on their length. The first such lower bound was proved by Goldreich et al. [18], as follows.

[3]More accurately for pure multilinear $\Sigma\Pi\Sigma$ circuits.

THEOREM 1.1 (Theorem 1.4 of [18]). *Let $\delta, \epsilon \in [0,1]$, $\mathbb{F}$ be a field, and let $E : \mathbb{F}^n \to \mathbb{F}^m$ be a linear $(2, \delta, \epsilon)$-LDC. Then*

$$m \geq 2^{\frac{\epsilon\,\delta\,n}{16} - 1 - \log_2 |\mathbb{F}|}.$$

Note that this result makes sense only when $|\mathbb{F}|$ is finite. We prove the following theorem.

THEOREM 1.2. *Let $\delta, \epsilon \in [0,1]$, $\mathbb{F}$ be a field, and let $E : \mathbb{F}^n \to \mathbb{F}^m$ be a linear $(2, \delta, \epsilon)$-LDC. Then*

$$m \geq 2^{\frac{\epsilon\,\delta\,n}{4} - 1}.$$

Compared with Theorem 1.4 of [18], our result removes the dependance on the size of the field in the exponent and works for every field size, finite and infinite. The idea of the proof is similar to the one in [18]—we show that, given a linear 2-LDC over an arbitrary field $\mathbb{F}$, we can construct from it a linear 2-LDC over $GF(2)$, with almost the same parameters, and then we use the lower bound of [18] for codes over $GF(2)$.

Relation between depth 3 circuits and LDCs. The main result of the paper is that from every $\Sigma\Pi\Sigma$ circuit that computes the zero polynomial, one can construct a linear LDC with two queries. Relations between arithmetic circuits and error correcting codes were known before [10, 45]; however, this is the first time that LDCs appear in the context of arithmetic circuits. More formally, let $\mathcal{C}$ be a $\Sigma\Pi\Sigma$ circuit, as in (1), computing the zero polynomial. We say that $\mathcal{C}$ is minimal if no proper subset of the multiplication gates sums to zero. We say that $\mathcal{C}$ is simple if there is no linear function that appears in all the multiplication gates (up to a multiplicative constant). Denote with r the rank of the linear functions appearing in $\mathcal{C}$.

THEOREM 1.3. *Let $k \geq 3$, $d \geq 2$, and let $\mathcal{C} \equiv 0$ be a simple and minimal $\Sigma\Pi\Sigma$ circuit of degree d, with k multiplication gates and n inputs. Let $r = \operatorname{rank}(\mathcal{C})$. Then we can construct a linear $(2, \frac{1}{12}, \frac{1}{4})$-LDC $E : \mathbb{F}^{n_1} \to \mathbb{F}^{n_2}$ with*

$$\frac{r}{2^{O(k^2)} \log(d)^{k-3}} \leq n_1 \quad \text{and} \quad n_2 \leq k \cdot d.$$

Thus, if k is a constant, then we can construct a linear $(2, \frac{1}{12}, \frac{1}{4})$-LDC that encodes messages of length $r/\text{polylog}(d)$ by codewords of length $O(d)$. As a corollary of Theorems 1.2 and 1.3 we get the next theorem.

THEOREM 1.4. *Let $k \geq 3$, $d \geq 2$, and let $\mathcal{C} \equiv 0$ be a simple and minimal $\Sigma\Pi\Sigma$ circuit of degree d with k multiplication gates and n inputs; then $r \leq 2^{O(k^2)} \log(d)^{k-2}$.*

Notice that the bound on r depends only on the degree and the number of multiplication gates and not on the number of variables! If the degree is polynomial in n (i.e., the circuit is of polynomial size), then the rank is bounded by $\text{polylog}(n)$, where a priori the rank could have been n.

PIT algorithms for depth 3 circuits. We design algorithms for PIT of depth 3 circuits with a constant number of multiplication gates. In particular we get a deterministic quasi-polynomial time algorithm and a randomized polynomial time algorithm that uses only polylog random bits. If the circuit is multilinear, i.e., every multiplication gate computes a multilinear polynomial, then we give a deterministic polynomial time algorithm for PIT. Our algorithms are non black box—all of them use the circuit computing the polynomial. The basic idea is to look for a minimal zero subcircuit and then, using Theorem 1.4, to write the linear functions in the circuit

as linear functions in $r \leq 2^{O(k^2)} \log(d)^{k-2}$ variables. Then we expand the monomials computed by the circuit and verify in a brute force manner that the resulting polynomial is zero. Thus the running time of our algorithm is the combined time that it takes to go over all subcircuits and the time that it takes to write all the monomials of a degree d polynomial in $\leq 2^{O(k^2)} \log(d)^{k-2}$ variables. We thus obtain the following result.

THEOREM 1.5. *Let $\mathcal{C}$ be a $\Sigma\Pi\Sigma$ circuit of degree d, with k multiplication gates and n inputs. Then we can check if $\mathcal{C} \equiv 0$:*

1. *Deterministically, in time* $\exp\left(2^{O(k^2)} \log^{k-1}(d)\right)$. *Thus, for a constant k the running time is* $\exp(\text{polylog}(d))$.
2. *Probabilistically, in time* $2^{O(k)} \text{poly}(d, \frac{1}{\epsilon})$, *using* $2^{O(k^2)} \log^{k-2}(d) \log(1/\epsilon)$ *random bits, with error probability ϵ. For constant k the running time is* $\text{poly}(d, \frac{1}{\epsilon})$, *and the number of random bits is* $\text{polylog}(d) \log(1/\epsilon)$.
3. *If $\mathcal{C}$ is also multilinear, then we can check if $\mathcal{C}$ is identically zero deterministically in time* $\exp(2^{O(k^2)}) \cdot \text{poly}(d)$. *For constant k the running time is* $\text{poly}(d)$.

Prior to our work the only algorithms that were designed for bounded depth circuits were the deterministic algorithm of [41] for pure multilinear depth 3 circuits and the black box algorithms of [22, 9, 29] for polynomials computed by depth 2 circuits (also known as sparse polynomials). None of the algorithms for sparse polynomials work in the case of depth 3 circuits, as such circuits can compute polynomials with exponentially many monomials. In fact, Klivans and Spielman [29] ask whether one could derandomize PIT for $\Sigma\Pi\Sigma$ circuits with only three multiplication gates ($k = 3$ in our notation). We give a deterministic algorithm that runs in quasi-polynomial time for this case, thus resolving the question of [29]. We note that a complete derandomization is to give a polynomial time algorithm for the problem, as was recently achieved by Kayal and Saxena [27]. We discuss their result in the next subsection.

1.5. Recent results. Kayal and Saxena [27] managed to give a polynomial time algorithm for PIT of depth 3 circuits with bounded top fan-in. Namely, they give an algorithm that runs in time polynomial in d^k, n, where k is the top fan-in, d is the degree of the circuit, and n is the number of variables. This result gives a complete derandomization of identity testing for depth 3 circuits with bounded top fan-in. In addition Kayal and Saxena give constructions of identically zero depth 3 circuits over $\mathbb{F} = GF(p)$ with $k = p$ for odd p, and $k = 3$ for $p = 2$, of degree d and rank $r = \log_p(d)$ (see Theorem 1.4).

We note, however, that for multilinear depth 3 circuits we give a polynomial time algorithm even when the top fan-in is $O(\sqrt{\log \log n})$ (Theorem 1.5, item 3), whereas [27] is polynomial time only when the top fan-in is constant.

1.6. Organization. In section 2 we analyze linear LDCs and derive Theorem 1.2. Section 3 is devoted to $\Sigma\Pi\Sigma$ circuits and their properties and serves as an introduction to the main part of the paper. In section 4 we give the proof of Theorem 1.3 and discuss the relation between $\Sigma\Pi\Sigma$ circuits and LDCs. Finally, in sections 5 and 6 we use our results to prove a structural theorem for zero $\Sigma\Pi\Sigma$ circuits and devise PIT algorithms based on this theorem.

2. Locally decodable codes. In this section we prove Theorem 1.2. We start by formally defining LDCs.

For a natural number n, let $[n] \triangleq \{1, \ldots, n\}$. Let $\mathbb{F}$ be a field. For a vector $x \in \mathbb{F}^n$ we write x_i for the ith coordinate of x. We denote by e_i the ith unit vector. For two

vectors $y, z \in \mathbb{F}^m$, denote by $\Delta(y,z)$ the number of coordinates in which y and z differ.

DEFINITION 2.1. *Let $\delta, \epsilon \in [0,1]$, and let q be an integer. We say that $E : \mathbb{F}^n \to \mathbb{F}^m$ is a* (q,δ,ϵ)-locally decodable code *if there exists a probabilistic oracle machine A such that*

- *in every invocation, A makes at most q queries (nonadaptively);*
- *for every $x \in \mathbb{F}^n$, for every $y \in \mathbb{F}^m$ with $\Delta(y, E(x)) < \delta m$, and for every $i \in [n]$, we have*

$$|\mathbb{F}| < \infty : \quad \boldsymbol{Pr}[A^y(i) = x_i] \geq \tfrac{1}{|\mathbb{F}|} + \epsilon,$$
$$|\mathbb{F}| = \infty : \quad \boldsymbol{Pr}[A^y(i) = x_i] \geq \epsilon,$$

where the probability is taken over the internal coin tosses of A.

We say that the code E is a linear *code if E is a linear transformation between $\mathbb{F}^n$ and $\mathbb{F}^m$.*

We are now ready to prove Theorem 1.2. We repeat its formulation here.

THEOREM 1.2 (restated). *Let $\delta, \epsilon \in [0,1]$, $\mathbb{F}$ be a field, and let $E : \mathbb{F}^n \to \mathbb{F}^m$ be a linear $(2,\delta,\epsilon)$-LDC. Then*

$$m \geq 2^{\frac{\epsilon\,\delta\,n}{4}-1}.$$

Our proof will build on the methods of [18], together with a novel reduction from LDCs over arbitrary fields to LDCs over $GF(2)$. We start by reviewing the results of [18]. The first step of their proof, given by the following lemma, is a reduction from the problem of proving lower bounds for LDCs to a graph-theoretic problem. The first such reduction was given in [26], where it was used to prove lower bounds on general LDCs. We note that in [18] the lemma was proved only over finite fields; however, it is easy to modify the proof to work for infinite fields as well.

LEMMA 2.2 (*implicit in* [18]). *Let $E : \mathbb{F}^n \to \mathbb{F}^m$ be a linear $(2,\delta,\epsilon)$-LDC, and let $a_1, \ldots, a_m \in \mathbb{F}^m$ be vectors such that*

$$E(x) = (\langle a_1, x\rangle, \ldots, \langle a_m, x\rangle)$$

($\langle\cdot,\cdot\rangle$ denotes the standard inner product). Then, for every $i \in [n]$, there exists a set $M_i \subset [m] \times [m]$ of at least $\frac{\epsilon\,\delta\,m}{4}$ disjoint pairs such that for every $(j_1, j_2) \in M_i$, $e_i \in Span\{a_{j_1}, a_{j_2}\}$.

From Lemma 2.2 we see that to prove lower bounds for two-query LDCs, it is sufficient to deal with the more combinatorial setting in which a given multiset of vectors contains many disjoint pairs spanning each unit vector.

The next step in the proof of [18] is a reduction from arbitrary finite fields to $GF(2)$. The next lemma summarizes the reduction given by [18].

LEMMA 2.3 (*implicit in* [18]). *Let $\mathbb{F}$ be a finite field, and let $a_1, \ldots, a_m \in \mathbb{F}^n$. For every $i \in [n]$ let $M_i \subset [m] \times [m]$ be a set of disjoint pairs of indices such that $e_i \in Span\{a_{j_1}, a_{j_2}\}$ for every $(j_1, j_2) \in M_i$. Then, there exist m' vectors $b_1, \ldots, b_{m'} \in \{0,1\}^n$ and n sets $M'_1, \ldots, M'_n \subset [m'] \times [m']$ of disjoint pairs such that*

1. *for every $(j_1, j_2) \in M'_i$, $b_{j_1} \oplus b_{j_2} = e_i$,*
2. *$m' = (|\mathbb{F}| - 1)m$, and*
3. *$\sum_{i=1}^n |M_i| \leq 2m + \frac{2}{|\mathbb{F}|-1} \sum_{i=1}^n |M'_i|$.*

The third and final step in the proof of [18] is a lemma which bounds the size of the matchings M_i, when the underlying field is $GF(2)$.

LEMMA 2.4 (*see* [18]). *Let $a_1, \ldots, a_m$ be elements of $\{0,1\}^n$. For every $i \in [n]$ let $M_i \subset [m] \times [m]$ be a set of disjoint pairs of indices such that $e_i = a_{j_1} \oplus a_{j_2}$ for every $(j_1, j_2) \in M_i$. Then*

$$\sum_{i=1}^{n} |M_i| \leq \frac{1}{2} m \log(m).$$

Notice that by Lemma 2.3 we have that $m = m'/|\mathbb{F}|$. Therefore, to get significant bounds from the combination of Lemmas 2.2, 2.3, and 2.4, we need $|\mathbb{F}|$ to be much smaller than 2^n. Thus, for very large fields (in particular, infinite fields) we do not get a significant result.

Our proof differs from that of [18] only in its second part—the reduction from $\mathbb{F}$ to $GF(2)$. Our reduction holds for any field, in particular for infinite $\mathbb{F}$, and does not involve the field size as a parameter.

LEMMA 2.5. *Let $\mathbb{F}$ be any field, and let $a_1, \ldots, a_m \in \mathbb{F}^n$. For every $i \in [n]$ let $M_i \subset [m] \times [m]$ be a set of disjoint pairs of indices such that $e_i \in Span\{a_{j_1}, a_{j_2}\}$ for every $(j_1, j_2) \in M_i$. Then, there exist m vectors $b_1, \ldots, b_m \in \{0,1\}^n$, and n sets $M_1'', \ldots, M_n'' \subset [m] \times [m]$ of disjoint pairs, such that*

1. *for every $(j_1, j_2) \in M_i''$, $b_{j_1} \oplus b_{j_2} = e_i$, and*
2. *$\sum_{i=1}^{n} |M_i| \leq 2 \sum_{i=1}^{n} |M_i''| + m$.*

Before giving the proof of the lemma we combine Lemmas 2.2, 2.5, and 2.4 to prove Theorem 1.2.

Proof of Theorem 1.2. Let $a_1, \ldots, a_m \in \mathbb{F}^n$ be vectors such that

$$E(x) = (\langle a_1, x \rangle, \ldots, \langle a_m, x \rangle).$$

From Lemma 2.2, we know that there exist n sets, $M_1, \ldots, M_n \subset [m] \times [m]$, of disjoint pairs of indices, such that for every $(j_1, j_2) \in M_i$ we have $e_i \in Span\{a_{j_1}, a_{j_2}\}$. We also know that

$$\forall\, i \in [n],\ |M_i| \geq \frac{\epsilon\, \delta\, m}{4}.$$

Now, let $b_1, \ldots, b_m \in \{0,1\}^n$ and $M_1'', \ldots, M_n'' \subset [m] \times [m]$ be as in Lemma 2.5. That is,

1. for every $(j_1, j_2) \in M_i''$, $b_{j_1} \oplus b_{j_2} = e_i$, and
2. $\sum_{i=1}^{n} |M_i| \leq 2 \sum_{i=1}^{n} |M_i''| + m$.

Using Lemma 2.4, we now have

$$\sum_{i=1}^{n} |M_i''| \leq \frac{1}{2} m \log(m).$$

This implies

$$n \cdot \frac{\epsilon\, \delta\, m}{4} \leq \sum_{i=1}^{n} |M_i| \leq 2 \sum_{i=1}^{n} |M_i''| + m \leq m \log(m) + m,$$

which, after division by m, gives the bound stated by the theorem. □

We now give the proof of Lemma 2.5.

Proof of Lemma 2.5. The proof will consist of two stages. First, we will remove a relatively small number of "bad" pairs from the given matchings $\{M_i\}$; then we

will transform the vectors $a_1, \ldots, a_m$ to vectors in $\{0,1\}^n$, while preserving a large portion of the pairs spanning the unit vectors.

Let (j_1, j_2) be a pair in M_i for some i such that either a_{j_1} or a_{j_2} are parallel to the unit vector e_i. Without loss of generality (w.l.o.g.) assume $a_{j_1} = c \cdot e_i$. We replace this pair with the pair (j_1, j_1). We do the same for all pairs containing a vector parallel to the unit vector spanned by this pair. This change does not affect the parameters of the lemma and is done only to simplify the analysis.

Next, we define a function $\theta : \mathbb{F}^n \backslash \{0\} \to [n]$ by

$$\theta(v) = \min\{i : v_i \neq 0\}.$$

For the rest of the proof we assume w.l.o.g. that in each pair (j_1, j_2) we have $\theta(a_{j_1}) \leq \theta(a_{j_2})$. (Note that we can assume w.l.o.g. that the vectors $a_1, \ldots, a_m$ are all different from zero.) We remove from each matching M_i all the pairs (j_1, j_2) in which $\theta(a_{j_1}) = i$. (This includes all pairs (j_1, j_1) described in the previous paragraph, and more.) Denote the resulting matching by M_i'. We claim that the total number of pairs removed in this stage is at most m.

Claim 2.6.

$$\sum_{i=1}^{n} |M_i| \leq \sum_{i=1}^{n} |M_i'| + m. \tag{2}$$

Proof. Let $p_1 = (j_1, j_2)$ and $p_2 = (k_1, k_2)$ be two removed pairs. If p_1 and p_2 were in the same matching M_i, then they would be disjoint, and so $j_1 \neq k_1$. If the pairs belonged to two different matchings, say, M_{i_1} and M_{i_2}, then $\theta(a_{j_1}) = i_1$ and $\theta(a_{k_1}) = i_2$, and again we get that $j_1 \neq k_1$. It follows that every removed pair has a distinct first element in the set $[m]$. Therefore, the total number of removed pairs cannot exceed m. ∎

In the following we assume w.l.o.g. that the first nonzero coordinate of each a_j is one. (We can assume that because we are allowed to use arbitrary linear combinations of the a_j when spanning the e_i.) The next claim asserts an important property of the matchings M_i'.

Claim 2.7. *For every $i \in [n]$ and $(j_1, j_2) \in M_i'$,*

$$e_i \in \mathit{Span}\{a_{j_1} - a_{j_2}\}.$$

Proof. Let $u = a_{j_1}$, $v = a_{j_2}$. We know that there exist two nonzero coefficients $\alpha, \beta \in \mathbb{F}$ such that $\alpha u + \beta v = e_i$. (Both coefficients are nonzero because we removed from M_i all pairs containing a vector parallel to e_i.) From this property it is clear that $\theta(u) \leq i$ (remember that $\theta(u) \leq \theta(v)$). As we removed all pairs in which $\theta(a_{j_1}) = i$ we conclude that $\theta(u) < i$. This in turn implies that $\theta(u) = \theta(v) < i$, because if $\theta(v) > \theta(u)$, then the vector $\alpha u + \beta v = e_i$ would have a nonzero coordinate in position $\theta(u) < i$. Now, since $v_{\theta(v)} = u_{\theta(u)} = 1$ we have that $\alpha + \beta = (\alpha u + \beta v)_{\theta(u)} = (e_i)_{\theta(u)} = 0$. Hence $e_i \in \mathit{Span}\{a_{j_1} - a_{j_2}\}$. ∎

Let us now proceed to the second stage of the proof of Lemma 2.5, in which we move from the field $\mathbb{F}$ to $GF(2)$. We will use a probabilistic argument to show the existence of a transformation that maps $\mathbb{F}$ to $GF(2)$, while preserving a large portion of the pairs that span a given unit vector.

For each $i \in [n]$, let a_{ji} denote the ith coordinate of the vector a_j. Let $V = \{a_{ji}\}_{j \in [m],\, i \in [n]}$ be the set of all field elements appearing in one of the vectors $a_1, \ldots, a_m$.

We pick a random function $f : V \to \{0,1\}$ and apply f to all the coordinates in all the vectors. Let

$$b_j = (f(a_{j1}), \ldots, f(a_{jn}))$$

be the vector in $\{0,1\}^n$ obtained from a_j after the transformation. We say that a pair $(j_1, j_2) \in M_i'$ "survived" the transformation if $e_i = b_{j_1} \oplus b_{j_2}$.

CLAIM 2.8. *The expected number of surviving pairs is* $\frac{1}{2}\sum_{i=1}^{n} |M_i'|$.

Proof. Consider a pair $(j_1, j_2) \in M_i'$. Since $e_i \in Span\{a_{j_1} - a_{j_2}\}$ we know that the vectors a_{j_1}, a_{j_2} are identical in all coordinates different from i. Hence, the vectors b_{j_1}, b_{j_2} will also be identical in those coordinates. From this we see that $e_i = b_{j_1} \oplus b_{j_2}$ iff b_{j_1} and b_{j_2} differ in their ith coordinate. This happens with probability of one-half. By linearity of expectation we can conclude that the expected number of surviving pairs is at least half the number of original pairs, which was $\sum_{i=1}^{n} |M_i'|$. ∎

From the above claim we can assert that there exists a function f for which the number of surviving pairs is at least $\frac{1}{2}\sum_{i=1}^{n} |M_i'|$. Thus, we have shown that there exist a set of vectors $b_1, \ldots, b_m \in \{0,1\}^n$ and matchings $M_i'' \subset [m] \times [m]$ such that for every $(j_1, j_2) \in M_i''$, we have $e_i = b_{j_1} \oplus b_{j_2}$. Furthermore, we can assume that

$$\sum_{i=1}^{n} |M_i'| \leq 2 \sum_{i=1}^{n} |M_i''|, \tag{3}$$

which completes the proof of the lemma, since now

$$\sum_{i=1}^{n} |M_i| \leq \sum_{i=1}^{n} |M_i'| + m \leq 2 \sum_{i=1}^{n} |M_i''| + m. \qquad \square$$

The next corollary combines the results of Lemmas 2.5 and 2.4 in a compact form. This corollary will be used in the proof given in section 4.

COROLLARY 2.9. *Let* $\mathbb{F}$ *be any field, and let* $a_1, \ldots, a_m \in \mathbb{F}^n$. *For every* $i \in [n]$ *let* $M_i \subset [m] \times [m]$ *be a set of disjoint pairs of indices* (j_1, j_2) *such that* $e_i \in Span\{a_{j_1}, a_{j_2}\}$. *Then*

$$\sum_{i=1}^{n} |M_i| \leq m \log(m) + m.$$

3. ΣΠΣ circuits. In this section we give some definitions related to ΣΠΣ circuits and describe some elementary operations that can be preformed on them. These definitions and operations will be used in the following sections.

3.1. Definitions. In the following we treat vectors in $\mathbb{F}^n$ also as linear forms in $\mathbb{F}[x_1, \ldots, x_n]$.

DEFINITION 3.1. *Let* $u \in \mathbb{F}^n$, $u = (u_1, \ldots, u_n)$. *Then*

$$u(x) = u_1 x_1 + u_2 x_2 + \cdots + u_n x_n.$$

DEFINITION 3.2. *Let* $v, u \in \mathbb{F}^n \backslash \{0\}$. *We write* $u \sim v$ *if there exists* $c \in \mathbb{F}$ *such that* $u = c \cdot v$.

We proceed to the main definition of this section.

DEFINITION 3.3. *Let $\mathbb{F}$ be a field. A $\Sigma\Pi\Sigma$ circuit, $\mathcal{C}$, over $\mathbb{F}$, with n inputs and k multiplication gates (i.e., top fan-in is k), is the formal expression*

$$\mathcal{C}(x) = \sum_{i=1}^{k} c_i \prod_{j=1}^{d_i} L_{ij}(x),$$

where for each $i \in [k]$, $j \in [d_i]$, L_{ij} is a nonconstant linear function,

$$L_{ij}(x) = L_{ij}^0 + L_{ij}^1 \cdot x_1 + \cdots + L_{ij}^n \cdot x_n,$$

and $c_i, L_{ij}^t \in \mathbb{F}$ for all i, j, t.

For every $i \in [k]$ define N_i to be the ith multiplication gate of $\mathcal{C}$:

$$N_i(x) \triangleq \prod_{j=1}^{d_i} L_{ij}(x).$$

For each $i \in [k]$, d_i is the degree of N_i. The number k denotes the number of different multiplication gates and is referred to as the top fan-in of the circuit. The total degree of $\mathcal{C}$ is $\max\{d_i\}$, and the size of $\mathcal{C}$ is $\sum_{i=1}^{k} d_i$. We denote with $\operatorname{rank}(\mathcal{C})$ the rank of $\mathcal{C}$:

$$\operatorname{rank}(\mathcal{C}) \triangleq \dim(Span\{L_{ij} : i \in [k],\, j \in [d_i]\}).$$

Remark. When dealing with $\Sigma\Pi\Sigma$ circuits, we will always assume that all the linear functions appearing in the circuit are different from zero.

We are interested in $\Sigma\Pi\Sigma$ circuits that compute the zero polynomial in $\mathbb{F}[x_1, \ldots, x_n]$. If $\mathcal{C}$ is such a circuit, we write $\mathcal{C} \equiv 0$. When dealing with circuits of this kind, it is sufficient to consider circuits of limited structure. This notion is made precise by the following definition and the lemma that follows.

DEFINITION 3.4. *Let $k, d > 0$ be integers. A $\Sigma\Pi\Sigma$ circuit $\mathcal{C}$ is called a $\Sigma\Pi\Sigma(k, d)$ circuit if the following three conditions hold:*

- *the top fan-in of $\mathcal{C}$ is k;*
- *$d_1 = d_2 = \cdots = d_k = d$; and*
- *for every $i \in [k]$ and $j \in [d]$, L_{ij} is a homogeneous linear form, that is, $L_{ij}(x) = L_{ij}^1 \cdot x_1 + \cdots + L_{ij}^n \cdot x_n$. (The free coefficient in each linear function is zero.)*

When dealing with $\Sigma\Pi\Sigma(k, d)$ circuits we will treat the linear functions L_{ij} also as vectors in $\mathbb{F}^n$, that is, $L_{ij} = (L_{ij}^1, \ldots, L_{ij}^n)$.

LEMMA 3.5. *There exists a polynomial time algorithm such that, given as input a $\Sigma\Pi\Sigma$ circuit $\mathcal{C}$, with top fan-in k and total degree $d > 0$, it outputs a $\Sigma\Pi\Sigma(k, d)$ circuit $\mathcal{C}'$ such that $\mathcal{C} \equiv 0$ iff $\mathcal{C}' \equiv 0$. The circuit $\mathcal{C}'$ is called the corresponding $\Sigma\Pi\Sigma(k, d)$ circuit of $\mathcal{C}$.*

Proof. We introduce a new variable y and define $\mathcal{C}'$ to be a circuit with input variables $x_1, \ldots, x_n, y$. Let

$$L_{ij}(x) = L_{ij}^0 + \sum_{t=1}^{n} L_{ij}^t \cdot x_t$$

be a linear function appearing in $\mathcal{C}$. Define

$$L'_{ij}(x, y) = L_{ij}^0 \cdot y + \sum_{t=1}^{n} L_{ij}^t \cdot x_t,$$

and define $\mathcal{C}'$ to be

$$\mathcal{C}'(x,y) = \sum_{i=1}^{k} c_i\, y^{d-d_i} \prod_{j=1}^{d_i} L'_{ij}(x,y).$$

Clearly, $\mathcal{C}'$ is a $\Sigma\Pi\Sigma(k,d)$ circuit and can be computed from $\mathcal{C}$ in time polynomial in the size of $\mathcal{C}$. Note that if we write

$$\mathcal{C}(x) = \sum_{i=0}^{d} P_i(x),$$

where $P_i(x)$ denotes the homogeneous part of degree i of $\mathcal{C}(x)$, then

$$\mathcal{C}'(x,y) = \sum_{i=0}^{d} P_i(x) y^{d-i}.$$

Therefore $\mathcal{C} \equiv 0$ iff $\mathcal{C}' \equiv 0$. □

Lemma 3.5 shows that to achieve our final goal, which is to derive PIT algorithms for $\Sigma\Pi\Sigma$ circuits, it is sufficient to consider $\Sigma\Pi\Sigma(k,d)$ circuits. For the rest of the paper we will deal only with $\Sigma\Pi\Sigma(k,d)$ circuits, and we shall sometimes refer to them simply as $\Sigma\Pi\Sigma$ circuits, omitting the suffix (k,d) where it is not needed.

3.2. Identically zero $\Sigma\Pi\Sigma$ circuits.

Simple circuits. It might be the case that there exists a linear function, L, that appears (up to a constant) in all multiplication gates of $\mathcal{C}$. In this case, we can divide each multiplication gate by L and get a simpler circuit $\mathcal{C}'$, whose degree is smaller than that of $\mathcal{C}$ by one. Clearly $\mathcal{C} \equiv 0$ iff $\mathcal{C}' \equiv 0$. The next two definitions deal with this case in a more general way.

DEFINITION 3.6. *Let $\mathcal{C}$ be a $\Sigma\Pi\Sigma$ circuit, and let $N_1, \ldots, N_k$ be its multiplication gates. Define*[4]

$$\gcd(\mathcal{C}) \triangleq \text{g.c.d.}(N_1(x), \ldots, N_k(x)).$$

Since each multiplication gate is a product of linear forms, $N_i(x) = \prod_{j=1}^{d_i} L_{ij}(x)$, we get that $\gcd(\mathcal{C})$ is the product of all the linear forms that appear in all the multiplication gates (up to multiplication by constants). Note also that $\gcd(\mathcal{C})$ can be easily computed from $\mathcal{C}$.

It is clear that $\mathcal{C} \equiv 0$ iff $\frac{\mathcal{C}}{\gcd(\mathcal{C})} \equiv 0$. This fact motivates the following definition.

DEFINITION 3.7. *A $\Sigma\Pi\Sigma$ circuit $\mathcal{C}$ is called* simple *if $\gcd(\mathcal{C}) = 1$. Let us also define $\mathrm{sim}(\mathcal{C})$ to be the simple circuit obtained from $\mathcal{C}$ by dividing each multiplication gate by $\gcd(\mathcal{C})$. It is clear that $sim(\mathcal{C})$ is always simple and that*

$$\mathcal{C}(x) = sim(\mathcal{C})(x) \cdot \gcd(\mathcal{C})(x).$$

Example 3.8. Let

$$\begin{aligned}\mathcal{C}(x) = {} & (\mathbf{x_1} + \mathbf{2x_2} + \mathbf{x_3} + \mathbf{1})(2x_1 + 4x_2 + 5x_3 + 2)(2x_1 + 4x_2 + 2x_3) \\ & + (\mathbf{x_1} + \mathbf{2x_2} + \mathbf{x_3} + \mathbf{1})(6x_1 + 4x_2 + 5x_3)(1x_1 + 1x_2 + 2x_3 + 4) \\ & + (\mathbf{2x_1} + \mathbf{4x_2} + \mathbf{2x_3} + \mathbf{2})(4x_2 + 1x_3)(7x_1 + 4x_2 + 2x_3).\end{aligned}$$

[4] g.c.d. stands for greatest common divisor.

Then

$$\gcd(\mathcal{C}) = x_1 + 2x_2 + x_3 + 1,$$

and

$$\begin{aligned} sim(\mathcal{C})(x) = &(2x_1 + 4x_2 + 5x_3 + 2)(2x_1 + 4x_2 + 2x_3) \\ &+ (6x_1 + 4x_2 + 5x_3)(1x_1 + 1x_2 + 2x_3 + 4) \\ &+ 2 \cdot (4x_2 + 1x_3)(7x_1 + 4x_2 + 2x_3). \end{aligned}$$

Minimal circuits. Suppose we have two $\Sigma\Pi\Sigma$ circuits $\mathcal{C}_1$ and $\mathcal{C}_2$, both of them equal to zero. Let k_1, k_2 denote the top fan-in of $\mathcal{C}_1$ and of $\mathcal{C}_2$, respectively. We can add $\mathcal{C}_1$ to $\mathcal{C}_2$ to create a new circuit $\mathcal{C} = \mathcal{C}_1 + \mathcal{C}_2$, with top fan-in $k_1 + k_2$, that will also be equal to zero. This new circuit $\mathcal{C}$, however, can be broken down into two smaller subcircuits that are zero. In the following we will be interested in circuits that *cannot* be broken down into smaller subcircuits that are equal to zero. The next two definitions deal with circuits of this type.

DEFINITION 3.9. *Let $\mathcal{C}$ be a $\Sigma\Pi\Sigma$ circuit, and let $\emptyset \neq T \subseteq [k]$. Then $\mathcal{C}_T$ is defined to be the subcircuit of $\mathcal{C}$ composed of the multiplication gates whose indices appear in T:*

$$\mathcal{C}_T(x) \triangleq \sum_{i \in T} c_i \prod_{j=1}^{d_i} L_{ij}(x) = \sum_{i \in T} c_i N_i(x).$$

DEFINITION 3.10. *Let $\mathcal{C} \equiv 0$ be a $\Sigma\Pi\Sigma$ circuit. We say that $\mathcal{C}$ is* minimal *if for every nonempty subset $T \subset [k]$, apart from $[k]$ itself, we have $\mathcal{C}_T \not\equiv 0$.*

The following easy claim shows that most properties of a $\Sigma\Pi\Sigma$ circuit $\mathcal{C}$ remain when we move to the corresponding $\Sigma\Pi\Sigma(k, d)$ circuit. The proof is immediate from the proof of Lemma 3.5.

CLAIM 3.11. *Let $\mathcal{C}$ be a $\Sigma\Pi\Sigma$ circuit, and let $\mathcal{C}'$ be the corresponding $\Sigma\Pi\Sigma(k, d)$ circuit (as defined in Lemma 3.5). Then we have the following:*

- $\operatorname{rank}(\mathcal{C}) \leq \operatorname{rank}(\mathcal{C}') \leq \operatorname{rank}(\mathcal{C}) + 1$.
- *$\mathcal{C}$ is simple iff $\mathcal{C}'$ is simple.*
- *$\mathcal{C}$ is minimal iff $\mathcal{C}'$ is minimal.*

Taking a linear transformation. We start with a simple operation of setting one of the variables to zero. This operation can be looked at as projecting all the linear functions in the circuit on a subspace of codimension 1.

DEFINITION 3.12. *Let $\mathcal{C}$ be a $\Sigma\Pi\Sigma$ circuit, and let $t \in [n]$. Define $\mathcal{C}|_{x_t=0}$ to be the circuit obtained from $\mathcal{C}$ by setting the variable x_t to zero. (This is the same as changing the tth coordinate in each linear form L_{ij} to zero.) The polynomial computed by $\mathcal{C}|_{x_t=0}$ is therefore*

$$(\mathcal{C}|_{x_t=0})(x) = \mathcal{C}(x_1, \ldots, x_{t-1}, 0, x_{t+1}, \ldots, x_n).$$

We can generalize the operation just defined by applying a general linear transformation on the linear functions of the circuit.

DEFINITION 3.13. *Let*

$$\mathcal{C}(x) = \sum_{i=1}^{k} c_i \prod_{j=1}^{d} L_{ij}(x)$$

be a $\Sigma\Pi\Sigma(k,d)$ circuit on n variables, and let $\pi : \mathbb{F}^n \to \mathbb{F}^n$ be a linear transformation. Define $\pi(\mathcal{C})$ to be the circuit obtained from $\mathcal{C}$ by applying π on all linear forms appearing in the circuit.[5] *That is,*

$$\pi(\mathcal{C})(x) = \sum_{i=1}^{k} c_i \prod_{j=1}^{d} \pi(L_{ij})(x).$$

The following claim is easy to verify.

CLAIM 3.14. *Let $\mathcal{C}$ be a $\Sigma\Pi\Sigma(k,d)$ circuit, and let $\pi : \mathbb{F}^n \to \mathbb{F}^n$ be an invertible linear transformation. Then*

- *$\mathcal{C} \equiv 0$ iff $\pi(\mathcal{C}) \equiv 0$,*
- *$\mathcal{C}$ is simple iff $\pi(\mathcal{C})$ is simple,*
- *$\mathcal{C}$ is minimal iff $\pi(\mathcal{C})$ is minimal, and*
- *$\operatorname{rank}(\mathcal{C}) = \operatorname{rank}(\pi(\mathcal{C}))$.*

4. ΣΠΣ circuits and LDCs. In this section we prove Theorem 1.3, which is the main result of the paper. This theorem shows the relation between $\Sigma\Pi\Sigma$ circuits and linear LDCs. It is more convenient to us to prove the theorem for $\Sigma\Pi\Sigma(k,d)$ circuits instead of general $\Sigma\Pi\Sigma$ circuits. From Claim 3.11, we know that moving from $\mathcal{C}$ to its corresponding $\Sigma\Pi\Sigma(k,d)$ circuit does not affect any of the relevant properties of $\mathcal{C}$, so the following theorem is equivalent to Theorem 1.3.

THEOREM 4.1. *Let $k \geq 3$, $d \geq 2$, and let $\mathcal{C} \equiv 0$ be a simple and minimal $\Sigma\Pi\Sigma(k,d)$ circuit, on n inputs, over a field $\mathbb{F}$. Then, there exists a linear $(2, \frac{1}{12}, \frac{1}{4})$-LDC $E : \mathbb{F}^{n_1} \to \mathbb{F}^{n_2}$, with*

$$\frac{\operatorname{rank}(\mathcal{C})}{P(k)\log(d)^{k-3}} \leq n_1 \quad \text{and} \quad n_2 \leq k \cdot d, \quad \text{where} \quad P(k) = 2^{O(k^2)}.$$

We prove Theorem 4.1 by induction on k. We devote section 4.1 to the base case of $k = 3$ and give the proof of the inductive step in section 4.2.

Before moving on to the proof of Theorem 4.1 we should explain why we are dealing only with circuits whose top fan-in is at least 3. The reason for this is that the structure of a zero $\Sigma\Pi\Sigma(k,d)$ circuit with $k = 1, 2$ is trivial. If $\mathcal{C}$ has only one multiplication gate ($k = 1$), then it is zero iff one of the linear functions appearing in it is the zero function. The case of $k = 2$ is equally trivial, as seen by the next claim.

CLAIM 4.2. *Let $\mathcal{C} = c_1 N_1(x) + c_2 N_2(x)$ be a $\Sigma\Pi\Sigma(2,d)$ circuit. Suppose $\mathcal{C} \equiv 0$. Then, the linear functions, appearing in the two multiplication gates N_1 and N_2, are the same, up to an ordering and multiplication by constants.*

Proof. Since $\mathcal{C} \equiv 0$, we have that $c_1 N_1(x) \equiv -c_2 N_2(x)$. Each multiplication gate N_i is a product of linear functions. Since every polynomial can be written, in a unique way, as a product of irreducible polynomials, and since every linear function is irreducible, we have that the linear functions in the two gates must be the same (up to an ordering and multiplication by constants). ∎

4.1. Proof of Theorem 4.1 for $k = 3$. Let $r = \operatorname{rank}(\mathcal{C})$. Then there exist r linearly independent functions $L_1, \ldots, L_r$ in $\mathcal{C}$. Using Claim 3.14, we can assume w.l.o.g. that for every $t \in [r]$, $L_t(x) = x_t$ (or in other words, $L_t = e_t$). Consider the circuit $\mathcal{C}|_{x_t=0}$ for some $t \in [r]$. Clearly $\mathcal{C}|_{x_t=0} \equiv 0$. From the fact that the function $L_t = e_t$ appears in one of the multiplication gates, we know that this gate

[5] Remember that we identify linear forms with vectors in $\mathbb{F}^n$.

will become zero in $\mathcal{C}|_{x_t=0}$. The following claim assures us that neither of the other two multiplication gates will become zero in $\mathcal{C}|_{x_t=0}$.

CLAIM 4.3. *Let L and L' be two linear functions appearing in two different multiplication gates of $\mathcal{C}$. Then $L \not\sim L'$.*

Proof. Assume for a contradiction that L divides both N_1 and N_2. As $c_3N_3(x) = -c_1N_1(x) - c_2N_2(x)$ we get that $N_3(x)$ is also divisible by L. But $\mathcal{C}$ is simple, so this is a contradiction. □

How can a circuit with two nonzero multiplication gates be zero? From Claim 4.2, this is possible only if the two gates contain the same linear functions, up to an ordering and multiplication by constants.

We thus get that every variable x_t, $t \in [r]$, induces a matching on the linear functions of the circuit. This matching contains d pairs of linear functions such that for every pair (L, L') in the matching, we have that L and L' belong to two different multiplication gates and that $L|_{x_t=0} \sim L'|_{x_t=0}$. Denote with M_t the matching induced by x_t. The next claim gives us more information about the pairs appearing in those matchings.

CLAIM 4.4. *Let $t \in [n]$, and let $L, L' \in \mathbb{F}^n$ such that $L \not\sim L'$, and $L|_{x_t=0} \sim L'_{x_t=0}$. Then*

$$e_t \in Span\{L, L'\}.$$

Proof. Let $L = (a_1, \ldots, a_n)$, $L' = (b_1, \ldots, b_n)$. Since $L|_{x_t=0} \sim L'|_{x_t=0}$, we know that there exists a constant $c \in \mathbb{F}$ such that for all $j \neq t$ we have $a_j = c \cdot b_j$. The fact that $L \not\sim L'$ implies that $a_t \neq c \cdot b_t$. It follows that $e_t \sim L - c \cdot L'$. In particular we get that $e_t \in Span\{L, L'\}$. □

From Claim 4.4 we see that every pair $(L, L') \in M_t$ spans the vector e_t. We also have that all the matchings $\{M_t\}_{t \in r}$ are contained in a set of $3d$ linear functions and that each matching contains d pairs. We can now construct a linear LDC in the following way. For each $i \in [3]$, $j \in [d]$, let $l_{ij} \in \mathbb{F}^r$ be the projection of L_{ij} on the first r coordinates. Define $E : \mathbb{F}^r \to \mathbb{F}^{3d}$ by

$$E_{ij}(x) = l_{ij}(x).$$

To show that E is a $(2, \frac{1}{12}, \frac{1}{4})$-LDC, we need to show a decoding algorithm for it. For each $t \in [r]$ we know that there are d disjoint pairs of code positions that span e_t. (Note that taking the projection on the first r coordinates does not affect this property.) To decode x_t we simply pick a random pair, uniformly, among these d pairs, and compute the linear combination giving e_t. Suppose we picked $l_{ij}(x)$ and $l_{i'j'}(x)$. We know that there exist constants $a, b \in \mathbb{F}$ such that

$$a \cdot l_{ij} + b \cdot l_{i'j'} = e_t.$$

Therefore

$$a \cdot E_{i,j}(x) + b \cdot E_{i',j'}(x) = a \cdot l_{ij}(x) + b \cdot l_{i'j'}(x) = e_t(x) = x_t.$$

If our codeword has at most $\frac{1}{12}(3d) = \frac{d}{4}$ corrupted positions, then at least $\frac{3}{4}$ of the d pairs are uncorrupted, and our algorithm will succeed with probability greater than $\frac{3}{4}$.

In the notation of the theorem, we have $n_1 = r$ and $n_2 = 3d = kd$. Let $P(3) = 1$; then

$$n_1 = r \geq \frac{r}{P(k)\log(d)^{k-3}},$$

and the theorem follows for $k = 3$.

4.2. Proof of Theorem 4.1 for $k \geq 4$. The proof is by induction on k. The idea behind the proof is the following. Assume that x_1 appears as a linear function in the circuit. A natural thing to do is to consider $\mathcal{C}|_{x_1=0}$. This circuit contains fewer multiplication gates, and so we would like to find an LDC in it by induction. A possible problem is that the rank of every minimal subcircuit is low. We can overcome this problem by showing that there are many variables $x_1, \ldots, x_m$ ($m \geq r/2^k$) such that there exists $I \subset [k]$ for which $\mathcal{C}_I \not\equiv 0$, but for every $t \in [m]$, $(\mathcal{C}_I)|_{x_t=0}$ is identically zero and minimal. In particular we show that this implies that the rank of $\mathcal{C}_I$ is at least m. We would like to construct a code from $\mathcal{C}_I$, so we consider, say, $(\mathcal{C}_I)|_{x_1=0}$. This circuit is identically zero and minimal, but it is not necessarily simple. Therefore we take $sim((\mathcal{C}_I)|_{x_t=0})$. However, it might be the case that the rank of this circuit is very small, i.e., that we lost a lot of rank when we removed the g.c.d. We overcome this difficulty by proving that there are relatively few ($\approx \log d$) variables, say, $x_1, \ldots, x_{\log d}$, such that the span of the linear functions in $sim((\mathcal{C}_I)|_{x_t=0})_{t=1,\ldots,\log d}$ contains almost all the functions of $\mathcal{C}_I$. In particular, for some t, the rank of $sim((\mathcal{C}_I)|_{x_t=0})$ is relatively high, so we can apply the induction hypothesis on this circuit. Proving the existence of such t is the main technical difficulty of the proof (Claim 4.8). We now give the formal proof.

Let $k \geq 4$, and assume the correctness of Theorem 4.1 for all $3 \leq k' < k$. Let

$$\mathcal{C}(x) = \sum_{i=1}^{k} c_i \prod_{j=1}^{d} L_{ij}(x)$$

be a $\Sigma\Pi\Sigma(k,d)$ circuit satisfying the conditions of the theorem. As in the proof for $k = 3$, let $r = \text{rank}(\mathcal{C})$, and w.l.o.g. assume that the circuit contains the first r unit vectors $e_1, \ldots, e_r$. We can also assume that

$$r \geq P(k) \log(d)^{k-3}, \tag{4}$$

for otherwise the theorem is trivially true, since we can always construct a two-query LDC whose message size is 1, satisfying the requirements of the theorem.

CLAIM 4.5. *For every $t \in [r]$ there exists a set $I_t \subset [k]$ such that*

1. $2 \leq |I_t| \leq k-1$ *and*
2. $(\mathcal{C}|_{x_t=0})_{I_t}$ *is identically zero and minimal.*

Proof. Let $t \in [r]$. Clearly $\mathcal{C}|_{x_t=0} \equiv 0$. Denote with k' the number of multiplication gates in $\mathcal{C}$ that become zero when $x_t = 0$. (These are exactly those multiplication gates that contain a linear function parallel to e_t.) Since we assumed that $\mathcal{C}$ contains e_t, we know that $k' \geq 1$. It is also easy to verify that $k' \leq k-2$. (If $k' = k$, then $\mathcal{C}$ is not simple, and if $k' = k-1$, then $\mathcal{C}$ is not divisible by x_t—as in Claim 4.3.) Therefore, the circuit $\mathcal{C}|_{x_t=0}$ is identically zero and contains at least two (and at most $k-1$) nonzero multiplication gates. Hence, we can decompose $\mathcal{C}|_{x_t=0}$ into minimal subcircuits, each of top fan-in at least two and at most $k-1$. Take I_t to be the index set of any one of these minimal subcircuits. ∎

From Claim 4.5 we can conclude that there are $m \geq \frac{r}{2^k}$ variables (w.l.o.g. $x_1, \ldots, x_m$) that have the same set I_t. Let $I = I_1 = \cdots = I_m$, and define

$$\hat{\mathcal{C}} = sim(\mathcal{C}_I).$$

The next claim summarizes several facts we know about the circuit $\hat{\mathcal{C}}$.

CLAIM 4.6.
1. $\hat{\mathcal{C}}$ *is a* $\Sigma\Pi\Sigma(\hat{k},\hat{d})$ *circuit with* $2 \le \hat{k} \le k-1$, $0 < \hat{d} \le d$.
2. $\hat{\mathcal{C}}$ *is simple.*
3. $\hat{\mathcal{C}} \not\equiv 0$.
4. *For all* $t \in [m]$, $\hat{\mathcal{C}}|_{x_t=0} \equiv 0$ *and is minimal.*
5. *For all* $t \in [m]$, e_t *does not appear in* $\hat{\mathcal{C}}$.

Proof. Parts 1 and 2 follow from the definition of $\hat{\mathcal{C}}$ (the fact that $0 < \hat{d}$ follows from 3 and 4). Part 3 is true because we assumed that $\mathcal{C}$ is minimal. Part 4 follows from the fact that $\hat{\mathcal{C}} = sim(\mathcal{C}_I)$ and that $(\mathcal{C}_I)|_{x_t=0} \equiv 0$ is minimal for all $t \in [m]$. Finally, 5 is a direct consequence of 4. □

Let $\hat{r} \triangleq \text{rank}(\hat{\mathcal{C}})$. The next claim shows that the rank of our chosen subcircuit $\hat{\mathcal{C}}$ is not considerably smaller than the rank of $\mathcal{C}$.

CLAIM 4.7. $\hat{r} \ge m$ $(\ge \frac{r}{2^k})$.

Proof. To prove the claim, we will show that the linear functions of $\hat{\mathcal{C}}$ span the unit vectors $e_1, \ldots, e_m$. Suppose, on the contrary, that there exists an index $t \in [m]$ for which e_t is not spanned by the linear functions of $\hat{\mathcal{C}}$. Assume w.l.o.g. that $t = 1$. There exists an invertible linear transformation $\pi : \mathbb{F}^n \to \mathbb{F}^n$ that satisfies the following two constraints:

- $\pi(e_1) = e_1$.
- The variable x_1 does not appear in the circuit $\pi(\hat{\mathcal{C}})$. (Equivalently, all the linear functions in $\pi(\hat{\mathcal{C}})$ are orthogonal to e_1.)

From Claim 4.6 we know that $\hat{\mathcal{C}} \not\equiv 0$ and that $\hat{\mathcal{C}}|_{x_1=0} \equiv 0$. Hence $\hat{\mathcal{C}}(x)$ can be written as

$$\hat{\mathcal{C}}(x) \equiv x_1 \cdot g(x),$$

where $g(x)$ is a nonzero polynomial. We can look at the transformation π as a linear change of variables and denote with $\pi(g)$ the polynomial obtained from $g(x)$ after this change. Thus,

$$\pi(\hat{\mathcal{C}})(x) \equiv \pi(x_1) \cdot \pi(g)(x) \equiv x_1 \cdot \pi(g)(x). \tag{5}$$

Now, since $g(x) \not\equiv 0$, and since π is invertible, Claim 3.14 implies[6] that $\pi(g)(x) \not\equiv 0$. From this and from (5) we see that $\pi(\hat{\mathcal{C}})(x)$ is a nonzero polynomial divisible by x_1. This is a contradiction, since we assumed that x_1 does not appear in $\pi(\hat{\mathcal{C}})$. □

We would like to use the inductive hypothesis on a well-chosen circuit among $\hat{\mathcal{C}}|_{x_1=0}, \ldots, \hat{\mathcal{C}}|_{x_m=0}$. However, there are two obstacles in the way. The first is that the top fan-in of $\hat{\mathcal{C}}$ might be equal to 2 (the theorem holds only for $k \ge 3$). This case is rather simple, since we can use the analysis given in section 4.1 to construct an LDC satisfying the conditions of the theorem. (A detailed analysis of this special case is deferred to the end of this section.) From now on we assume that $\hat{k} \ge 3$. The second obstacle is that these circuits are not necessarily simple. We overcome this obstacle by using the inductive hypothesis on $sim(\hat{\mathcal{C}}|_{x_t=0})$ instead. The next claim, whose proof is deferred to section 4.3, tells us which of these circuits we should pick.

For each $t \in [m]$, let $r_t \triangleq \text{rank}(sim(\hat{\mathcal{C}}|_{x_t=0}))$.

CLAIM 4.8. *There exists* $t \in [m]$ *such that*

$$r_t \ge \frac{\hat{r}}{2^{k+1}\log(d)}.$$

[6] It is easy to see that this part of Claim 3.14 holds also for general polynomials and not just $\Sigma\Pi\Sigma$ circuits.

Claim 4.8 assures us that one of the r_t is large (we assume w.l.o.g. that $t = 1$). We get that

$$r_1 \geq \frac{\hat{r}}{2^{k+1}\log(d)}. \tag{6}$$

Our next step is to apply the induction hypothesis to the circuit $sim(\hat{\mathcal{C}}|_{x_1=0})$. However, to use Theorem 4.1, we require that the degree of the given circuit be at least two. The next claim shows that the degree of $sim(\hat{\mathcal{C}}|_{x_1=0})$ is indeed at least two.

CLAIM 4.9. *Let d_1 denote the degree of $sim(\hat{\mathcal{C}}|_{x_1=0})$. Then $d_1 \geq 2$.*

Proof. If $d_1 < 2$, then $r_1 < k$. (The number of linear functions is at most $\hat{k} < k$.) By (6) we get that

$$\hat{r} \leq k2^{k+1}\log(d).$$

Now, using the fact that $\hat{r} \geq m \geq \frac{r}{2^k}$ (Claim 4.7), we conclude that

$$r \leq 2^k\hat{r} \leq k2^{2k+1}\log(d),$$

contradicting (4), for an appropriate choice of $P(k) = 2^{O(k^2)}$. ∎

Therefore $sim(\hat{\mathcal{C}}|_{x_1=0})$ satisfies all the conditions of Theorem 4.1. The induction hypothesis, applied on $sim(\hat{\mathcal{C}}|_{x_1=0})$, asserts that there exists a $(2, \frac{1}{12}, \frac{1}{4})$-LDC, $E : \mathbb{F}^{n_1} \to \mathbb{F}^{n_2}$, with

$$n_1 \geq \frac{r_1}{P(\hat{k})\log(d_1)^{\hat{k}-3}} \quad \text{and} \quad n_2 \leq \hat{k} \cdot d_1 \ (\leq k \cdot d).$$

Using (6) and the facts that $\hat{k} \leq k - 1$ and $\hat{r} \geq m \geq \frac{r}{2^k}$, we derive the following inequalities:

$$\begin{aligned} n_1 &\geq \frac{r_1}{P(\hat{k})\log(d_1)^{\hat{k}-3}} \\ &\geq \frac{r_1}{P(k-1)\log(d)^{k-4}} \\ &\geq \frac{\hat{r}}{2^{k+1}P(k-1)\log(d)^{k-3}} \\ &\geq \frac{r}{2^{2k+1}P(k-1)\log(d)^{k-3}} \\ &\geq \frac{r}{P(k)\log(d)^{k-3}} \end{aligned}$$

(for an appropriate choice of $P(k) = 2^{O(k^2)}$). This completes the proof of the inductive step and of Theorem 4.1.

4.2.1. A special case: $\hat{k} = 2$. In this subsection we analyze a special case of the proof of Theorem 4.1. This case is when $\hat{k}$ (the top fan-in of the circuit $\hat{C}$, whose properties are detailed in Claim 4.6) is equal to 2. The analysis of this case differs from the analysis of the general ($\hat{k} \geq 3$) case because we cannot apply the inductive hypothesis on $\hat{C}$ (or more precisely, on the circuits $C|_{x_t=0}$). We now show how to complete the proof of the theorem (that is, to construct an LDC satisfying the requirements of the thorem) in this case.

Denote by $\hat{N}_1$ and $\hat{N}_2$ the two multiplication gates of $\hat{\mathcal{C}}$. We can write

$$\hat{\mathcal{C}}(x) \equiv c_1\hat{N}_1(x) + c_2\hat{N}_2(x).$$

Now, since $\hat{\mathcal{C}}$ is simple and nonzero, we have

$$\gcd(\hat{\mathcal{C}}) \equiv \text{g.c.d.}(\hat{N}_1(x), \hat{N}_2(x)) \equiv 1.$$

Next, let $t \in [m]$, and consider what happens to $\hat{\mathcal{C}}$ after we set x_t to zero. We know that $\hat{\mathcal{C}}|_{x_t=0} \equiv 0$, and so

$$c_1\hat{N}_1|_{x_t=0} \equiv -c_2\hat{N}_2|_{x_t=0}.$$

Now, since $\hat{N}_1|_{x_t=0}$ and $\hat{N}_2|_{x_t=0}$ are both nonzero (by Claim 4.6, $e_1, \ldots, e_m$ do not appear in $\hat{\mathcal{C}}$), we can deduce, as we did in section 4.1, that there exist m matchings M_t, $t \in [m]$, of size $|M_t| = \hat{d}$, of linear functions appearing in $\hat{\mathcal{C}}$, such that for every pair $(L, L') \in M_t$, $e_t \in Span\{L, L'\}$. Projecting each linear function in $\hat{\mathcal{C}}$ on the first m coordinates, and using the construction from section 4.1, we see that there exists a $(2, \frac{1}{12}, \frac{1}{4})$-LDC,[7] $E : \mathbb{F}^m \to \mathbb{F}^{2\hat{d}}$. In the notation of the theorem, we have

$$n_2 = 2\hat{d} \leq kd$$

and

$$n_1 = m \geq \frac{r}{2^k} \geq \frac{r}{P(k)\log(d)^{k-3}},$$

as required by the theorem.

4.3. Proof of Claim 4.8. In this section we prove Claim 4.8. The following notation is required for the proof.

4.3.1. Notation. Let $\hat{N}_1, \ldots, \hat{N}_{\hat{k}}$ denote the multiplication gates of $\hat{\mathcal{C}}$. We will treat $\hat{\mathcal{C}}, \hat{N}_1, \ldots, \hat{N}_{\hat{k}}$ also as sets of indices. We shall abuse notation and write

$$\hat{\mathcal{C}} = \{(i,j) \mid i \in [\hat{k}],\, j \in [\hat{d}]\},$$

$$\hat{N}_i = \{(i,j) \mid j \in [\hat{d}]\}.$$

For a set $H \subset \hat{\mathcal{C}}$, we denote with $\text{rank}(H)$ the dimension of the vector space spanned by the linear functions whose indices appear in H. That is,

$$\text{rank}(H) \triangleq \dim\left(Span\{L_{ij} : (i,j) \in H\}\right).$$

For the rest of the proof we will treat subsets of $\hat{\mathcal{C}}$ interchangeably as sets of indices and as (multi)sets of linear functions.

We would next like to define, for each $t \in [m]$, certain subsets of $\hat{\mathcal{C}}$ that capture the structure of $\hat{\mathcal{C}}|_{x_t=0}$. Fix some $t \in [m]$, and consider what happens to $\hat{\mathcal{C}}$ when we set x_t to be zero. The resulting circuit $\hat{\mathcal{C}}|_{x_t=0}$ is generally not simple and can therefore be partitioned (see Definition 3.7) into two disjoint sets: a set containing the indices of the linear functions appearing in $\gcd(\hat{\mathcal{C}}|_{x_t=0})$, and a set containing the indices of the

[7] We could have taken δ to be $\frac{1}{8}$ instead of $\frac{1}{12}$, because the number of multiplication gates is two and not three.

remaining linear functions (these are the linear functions appearing in $sim(\hat{\mathcal{C}}|_{x_t=0})$). To be more precise, denote by δ_t the degree of $\gcd(\hat{\mathcal{C}}|_{x_t=0})$. In every multiplication gate $\hat{N}_i$, there are δ_t linear functions such that the restriction of their product to the linear space defined by the equation $x_t = 0$ is equal to $\gcd(\hat{\mathcal{C}}|_{x_t=0})$. In other words, the product of these δ_t linear functions is equal to $\gcd(\hat{\mathcal{C}})$ under the restriction $x_t = 0$. Denote the set of indices of these functions by G_t^i, and let $R_t^i \triangleq \hat{N}_i \backslash G_t^i$ be the set of indices of the remaining linear functions of this multiplication gate. We thus have (for some choice of constants $\{c_i\}$)

$$sim(\hat{\mathcal{C}}|_{x_t=0}) = \sum_{i=1}^{\hat{k}} c_i \prod_{(i,j)\in R_t^i} (L_{ij}|_{x_t=0})$$

and

$$\forall i \in [\hat{k}], \qquad \gcd(\hat{\mathcal{C}}|_{x_t=0}) = \prod_{(i,j)\in G_t^i} (L_{ij}|_{x_t=0}).$$

We now define, for each $t \in [m]$, the sets $R_t \triangleq \bigcup_{i=1}^{\hat{k}} R_t^i$ and $G_t \triangleq \bigcup_{i=1}^{\hat{k}} G_t^i$. The following claim summarizes some facts that we will later need.

CLAIM 4.10. *For every $t \in [m]$,*

1. $R_t \cap G_t = \emptyset$.
2. $\hat{\mathcal{C}} = R_t \cup G_t$.
3. $|G_t^i| = |G_t^{i'}|$ *for all* i, i'.
4. $|G_t| = \hat{k} \cdot deg(\gcd(\hat{\mathcal{C}}|_{x_t=0})) = \hat{k} \cdot \delta_t$.
5. R_t *contains the indices of the linear functions appearing in* $sim(\hat{\mathcal{C}}|_{x_t=0})$.
6. $r_t = \mathrm{rank}(sim(\hat{\mathcal{C}}|_{x_t=0})) = \mathrm{rank}(R_t)$.

Proof. Items 1 and 2 follow directly from the definition of R_t and G_t as R_t^i and G_t^i give a partition of the indices in $\hat{N}_i$. Items 3 and 4 hold as the linear factors of $\gcd(\hat{\mathcal{C}}|_{x_t=0})$ belong to all the multiplication gates. By definition, R_t^i is the set of linear functions in $\hat{N}_i$ that belong to $sim(\hat{\mathcal{C}}|_{x_t=0})$, which implies item 5. Finally, by definition, $r_t = \mathrm{rank}(sim(\hat{\mathcal{C}}|_{x_t=0}))$ and by item 5 we have that R_t is the set of linear functions appearing in $sim(\hat{\mathcal{C}}|_{x_t=0})$. ∎

4.3.2. The proof. We finally give the proof of Claim 4.8. For convenience we restate it here.

CLAIM 4.8 (restated). *There exists $t \in [m]$ such that*

$$r_t \geq \frac{\hat{r}}{2^{k+1}\log(d)}.$$

We start by assuming that the claim is false. In other words, we assume that for every $t \in [m]$

$$r_t < \frac{\hat{r}}{2^{k+1}\log(d)}. \tag{7}$$

Having defined, for each $t \in [m]$, the sets R_t and G_t, we would now like to show that there exist a small ($\sim \log(d)$) number of sets R_t such that their union covers almost all of $\hat{\mathcal{C}}$. As $\mathrm{rank}(\hat{\mathcal{C}})$ is relatively high, and for each t, $r_t = \mathrm{rank}(R_t)$ is (assumed to be) relatively small, we will get a contradiction. We construct the cover step by step, and in each step we will find an index $t \in [m]$ such that the set R_t covers

at least half the linear functions not yet covered. This idea is made precise by the following claim.

CLAIM 4.11. *For every integer* $1 \le q \le \log(\hat{d})$ *there exist* q *indices* $t_1, \ldots, t_q \in [m]$ *for which*

$$\left| \bigcup_{s=1}^{q} R_{t_s} \right| \ge \hat{k}\hat{d}(1 - 2^{-q}).$$

Proof. The proof proceeds by induction on q.

Base case $q = 1$. To prove the claim for $q = 1$, it is sufficient to show that there exists $t \in [m]$ for which $|R_t| \ge \frac{1}{2}\hat{k}\hat{d}$. Suppose, on the contrary, that for all $t \in [m]$, $|R_t| < \frac{1}{2}\hat{k}\hat{d}$. Claim 4.10 implies that for all $t \in [m]$, $|G_t| \ge \frac{1}{2}\hat{k}\hat{d}$. This in turn implies (by item 3 of Claim 4.10) that for all $t \in [m]$

$$|G_t^1| \ge \frac{1}{2}\hat{d}. \tag{8}$$

The next lemma shows that, under the conditions just described, the linear functions of $\hat{\mathcal{C}}$ "contain" a two-query LDC. We will then apply our results on LDCs from section 2 (namely, Corollary 2.9) to derive a contradiction. Lemma 4.12 is more general than what is required at this point; however, we will need it in its full generality when we handle $q > 1$.

LEMMA 4.12. *Let* $\mathcal{C}$ *be a simple* $\Sigma\Pi\Sigma(k, d)$ *circuit with* n *inputs. Let* $t \in [n]$, $i_t \in [k]$. *Denote* $\delta_t = deg(\gcd(\mathcal{C}|_{x_t=0}))$. *Suppose that the linear functions in* N_{i_t} *are ordered such that*

$$\gcd(\mathcal{C}|_{x_t=0}) = (L_{i_t 1}|_{x_t=0})(x) \cdot (L_{i_t 2}|_{x_t=0})(x) \cdot \cdots \cdot (L_{i_t \delta_t}|_{x_t=0})(x).$$

Then, there exists a matching, $M = \{\mathcal{P}_1, \ldots, \mathcal{P}_g\} \subseteq \mathcal{C} \times \mathcal{C}$, *consisting of* δ_t *disjoint pairs of linear functions, such that for each* $j \in [\delta_t]$,

- *the two linear functions in* $\mathcal{P}_j$ *span* e_t, *and*
- *the first element of* $\mathcal{P}_j$ *is* $L_{i_t j}$.

Proof. As the linear factors of $\gcd(\mathcal{C}|_{x_t=0})$ belong to all the multiplication gates (of $\mathcal{C}|_{x_t=0}$) we can reorder the linear functions in each gate N_i, $i \ne i_t$, such that

$$\forall j \in [\delta_t] \quad : \quad L_{1j}|_{x_t=0} \sim L_{2j}|_{x_t=0} \sim \cdots \sim L_{i_t j}|_{x_t=0} \sim \cdots \sim L_{kj}|_{x_t=0}.$$

As $\mathcal{C}$ is simple, it cannot be the case that, for some j, $L_{i_t j}$ divides all the multiplication gates. Therefore, for every $j \in [\delta_t]$ there exists an index $\alpha(j) \in [k]$ such that $L_{i_t j} \not\sim L_{\alpha(j) j}$. From Claim 4.4 it follows that

$$\forall j \in [\delta_t] \quad : \quad e_t \in Span\{L_{i_t j}, L_{\alpha(j) j}\}.$$

For each $j \in [\delta_t]$ let $\mathcal{P}_j = (L_{i_t j}, L_{\alpha(j) j})$. Set $M = \{\mathcal{P}_1, \ldots, \mathcal{P}_{\delta_t}\}$. It is clear that each $\mathcal{P}_j$ satisfies the two conditions of the lemma and that the $\mathcal{P}_j$ are disjoint. ∎

We continue with the proof of Claim 4.11. From (8) and Lemma 4.12 we conclude that for each $t \in [m]$ there exists a matching $M_t \subset \mathcal{C} \times \mathcal{C}$, containing at least $\frac{1}{2}\hat{d}$ disjoint pairs of linear functions, such that every pair in M_t spans e_t. Corollary 2.9 implies that

$$\frac{1}{2}\hat{d}m \le \sum_{t=1}^{m} |M_t| \le \hat{k}\hat{d}\log(\hat{k}\hat{d}) + \hat{k}\hat{d},$$

which gives

$$m \leq 2\hat{k}\log(\hat{k}\hat{d}) + 2\hat{k} < \log(d)^{k-3}P(k)2^{-k}$$

(for an appropriate choice of $P(k) = 2^{O(k^2)}$). Now, since $m \geq \frac{r}{2^k}$, we have that

$$r < \log(d)^{k-3}P(k),$$

contradicting (4). Therefore our initial assumption was wrong and we conclude that there exists t_1 with $|R_{t_1}| \geq \frac{1}{2}\hat{k}\hat{d}$. This completes the proof of Claim 4.11 for the case of $q = 1$.

Induction step. Let us now assume that we have found $q-1$ indices $t_1, \ldots, t_{q-1} \in [m]$ for which

$$\left|\bigcup_{s=1}^{q-1} R_{t_s}\right| \geq \hat{k}\hat{d}(1 - 2^{-(q-1)}).$$

Let

$$R \triangleq \bigcup_{s=1}^{q-1} R_{t_s}, \tag{9}$$

$$S \triangleq \hat{\mathcal{C}} \backslash R. \tag{10}$$

Then, by our assumption,

$$|S| \leq \hat{k}\hat{d}2^{-(q-1)}. \tag{11}$$

The proof goes along the same lines as the proof for $q = 1$: we show that there exists an index $t \in [m]$ such that R_t covers at least half of S. We will argue that if such an index does not exist, then a contradiction to (4) can be derived. Our main tools in doing so are Lemma 4.12 and Corollary 2.9.

CLAIM 4.13. *There exists $t \in [m]$ such that for all $i \in [\hat{k}]$,*

$$|G_t^i \cap S| < \hat{d}2^{-q}.$$

Roughly, the lemma states that there exists some variable, x_t, such that most of the linear functions in S do not belong to $\gcd(\hat{\mathcal{C}}|_{x_t=0})$. In particular it implies that R_t covers a large fraction of S, as needed.

Proof. Assume, on the contrary, that for every $t \in [m]$ there exists $i_t \in [\hat{k}]$ for which

$$|G_t^{i_t} \cap S| \geq \hat{d}2^{-q}.$$

From Lemma 4.12 we get that, for every $t \in [m]$, there exists a matching M_t, consisting of $\hat{d}2^{-q}$ disjoint pairs of linear functions, such that each pair spans e_t, and that the first element in each pair is in $G_t^{i_t} \cap S$ (from the lemma we actually get that M_t contains $deg(\gcd(\hat{\mathcal{C}}|_{x_t=0}))$ number of pairs, but we are interested only in the pairs whose first element is in $G_t^{i_t} \cap S$).

We would now like to apply Corollary 2.9 on the matchings $\{M_t\}_{t\in[m]}$; however, for our needs, we would also like that all the linear functions in all the matchings will belong to S. We achieve this by projecting all functions in R to zero. As the

dimension of the linear functions in R is small (by our assumption that each r_{t_s} is small) we can find a linear transformation that sends the linear functions in R to zero but leaves many of the linear functions $\{x_t\}$ linearly independent. This is formalized in the next claim.

CLAIM 4.14. *There exists a subset $A \subset [m]$ of size $|A| \geq \frac{m}{2}$ and a linear transformation $\pi : \mathbb{F}^n \to \mathbb{F}^n$ such that*

- $\ker(\pi) = Span(R)$,
- *for all $t \in A$, $\pi(e_t) = e_t$.*

Proof. Calculating, we get that

$$\begin{aligned} \operatorname{rank}(R) &= \operatorname{rank}\left(\bigcup_{s=1}^{q-1} R_{t_s}\right) \leq \sum_{s=1}^{q-1} \operatorname{rank}(R_{t_s}) = \sum_{s=1}^{q-1} r_{t_s} \\ &\leq (q-1)\frac{\hat{r}}{\log(d)2^{k+1}} \leq \frac{r}{2^{k+1}} \leq \frac{m}{2}, \end{aligned} \tag{12}$$

where the second inequality follows from (7), the third inequality follows from the fact that $q \leq \log \hat{d} \leq \log d$ and $\hat{r} \leq r$, and the last inequality follows from the fact that $\frac{r}{2^k} \leq m$. Let $m' = m - \operatorname{rank}(R)$. From (12) we get that $m' \geq m/2$. In particular, there exists a subset $A \subset [m]$, of size $|A| = m'$, such that $Span(\{x_t \mid t \in A\}) \cap Span(R) = \{0\}$. Hence, there exists a linear transformation $\pi : \mathbb{F}^n \to \mathbb{F}^n$ such that

- $\ker(\pi) = Span(R)$,
- for all $t \in A$, $\pi(e_t) = e_t$.

This completes the proof of Claim 4.14. ∎

Let A be the set obtained from the above claim and π the corresponding linear transformation. We assume, w.l.o.g., that $A = [m']$. From here on, we consider only variables x_t such that $t \in [m']$ (i.e., $t \in A$). Fix such $t \in [m']$, and let $M'_t = \pi(M_t)$. In other words, $M'_t = \{(\pi(L), \pi(L'))\}_{(L,L') \in M_t}$. Clearly,

$$|M'_t| = |M_t| \geq \hat{d}2^{-q}. \tag{13}$$

Note that the pairs in M'_t still span e_t, as for any pair $(L.L') \in M_t$, with $e_t = \alpha L + \beta L'$, we have that

$$e_t = \pi(e_t) = \pi(\alpha L + \beta L') = \alpha\pi(L) + \beta\pi(L').$$

Since all the linear functions appearing in R were projected to zero, we know that all the pairs in each M'_t are contained in the multiset[8] $S' \triangleq \{\pi(L) : L \in S\}$.

After this long preparation we apply Corollary 2.9 to the matchings M'_t and derive the following inequality:

$$\sum_{t=1}^{m'} |M'_t| \leq |S'| \log(|S'|) + |S'|. \tag{14}$$

As $|S'| = |S|$ (remember that S' is a multiset), we get by (11) that

$$|S'| \leq \hat{k}\hat{d}2^{-(q-1)}. \tag{15}$$

[8] Note that, as in the proof of Lemma 2.5, we can replace each pair in M'_t that contains the zero vector with a singleton.

By (13), (14), and (15), it follows that

$$m' \cdot (\hat{d}2^{-q}) \leq \sum_{t=1}^{m'} |M'_t| \leq |S'| \log(|S'|) + |S'| \\ \leq \hat{k}\hat{d}2^{-(q-1)} \log(\hat{k}\hat{d}2^{-(q-1)}) + \hat{k}\hat{d}2^{-(q-1)}.$$

From the fact that $k \geq 4$ and $m' \geq m/2$ (and some simple manipulations), we see that for an appropriate choice of $P(k) = 2^{O(k^2)}$

$$m < 2^{-k} P(k) \log(d)^{k-3}.$$

As $m \geq \frac{r}{2^k}$, we get that

$$r < P(k) \log(d)^{k-3},$$

contradicting (4). This completes the proof of Claim 4.13. □

Let us now proceed with the proof of Claim 4.11. Take t_q to be the index described by Claim 4.13, that is,

$$\forall i \in [\hat{k}] \quad : \quad |G^i_{t_q} \cap S| < \hat{d}2^{-q}.$$

In particular,

$$|G_{t_q} \cap S| < \hat{k}\hat{d}2^{-q}.$$

Notice that by (9) and (10) and by the fact that R_{t_q} and G_{t_q} give a partition of $\hat{\mathcal{C}}$, we get that the complement of $\bigcup_{s=1}^{q} R_{t_s}$ is exactly $G_{t_q} \cap S$. From this we get that adding R_{t_q} to R gives

$$\left| \bigcup_{s=1}^{q} R_{t_s} \right| \geq \hat{k}\hat{d}(1 - 2^{-q}).$$

This completes the proof of Claim 4.11. □

Having proved Claim 4.11, we are now just steps away from completing the proof of Claim 4.8. Taking q to be $\lfloor \log(\hat{d}) \rfloor$ in Claim 4.11, we get that there exist indices $t_1, \ldots, t_{\lfloor \log(\hat{d}) \rfloor} \in [m]$ such that

$$\left| \bigcup_{s=1}^{\lfloor \log(\hat{d}) \rfloor} R_{t_s} \right| \geq \hat{k}\hat{d} - 2\hat{k} \ldots.$$

Thus

$$\hat{r} - 2\hat{k} \leq \operatorname{rank} \left(\bigcup_{s=1}^{\lfloor \log(\hat{d}) \rfloor} R_{t_s} \right) \leq \sum_{s=1}^{\lfloor \log(\hat{d}) \rfloor} r_{t_s}.$$

The last inequality tells us that there exists some $t \in [m]$ for which

$$r_t \geq \frac{\hat{r} - 2\hat{k}}{\lfloor \log(\hat{d}) \rfloor} \geq \frac{\hat{r} - 2\hat{k}}{\log(d)}. \tag{16}$$

In order to finish the proof of Claim 4.8 we prove the following inequality.

CLAIM 4.15.

$$\hat{r} - 2\hat{k} \geq \frac{\hat{r}}{2^{k+1}}.$$

Proof. Using (4), we get

$$\hat{r} \geq m \geq 2^{-k} r \geq 2^{-k} P(k) \log(d)^{k-3}.$$

Therefore we can choose $P(k) = 2^{O(k^2)}$ such that

$$\hat{r} > 2\hat{k} \frac{2^{k+1}}{2^{k+1} - 1}.$$

This implies the inequality in the claim. □

Combining Claim 4.15 with (16), we conclude that there exists $t \in [m]$ for which

$$r_t \geq \frac{\hat{r}}{\log(d) 2^{k+1}},$$

which contradicts our initial assumption (7). This completes the proof of Claim 4.8.

5. A structural theorem for zero ΣΠΣ circuits. The main result of this section is a structural theorem for ΣΠΣ circuits which are identically zero. The proof is based on the results of section 4. To ease the notation we will prove our results only for $\Sigma\Pi\Sigma(k, d)$ circuits; however, from Claim 3.11 it will follow that all the results also hold for ΣΠΣ circuits with k multiplication gates of degree d.

THEOREM 5.1 (*structural theorem*). *Let $\mathcal{C} \equiv 0$ be a $\Sigma\Pi\Sigma(k, d)$ circuit. Then, there exists a partition of $[k]$: $T_1, \ldots, T_s \subset [k]$ with the following properties:*

- $\mathcal{C} = \sum_{i=1}^{s} \mathcal{C}_{T_i} = \sum_{i=1}^{s} \gcd(\mathcal{C}_{T_i}) \cdot sim(\mathcal{C}_{T_i})$.
- *For all $i \in [s]$, $sim(\mathcal{C}_{T_i}) \equiv 0$ and is simple and minimal.*
- *For all $i \in [s]$, $\operatorname{rank}(sim(\mathcal{C}_{T_i})) \leq 2^{O(k^2)} \log(d)^{k-2}$.*

In other words, the theorem says that every zero ΣΠΣ circuit can be broken down into zero subcircuits of low rank (ignoring the g.c.d.). This fact will be used in the next section, in which we devise PIT algorithms for ΣΠΣ circuits.

Before giving the proof of the theorem we prove a lemma that bounds the rank of a zero, simple, and minimal ΣΠΣ circuit. Note that Theorem 1.4 follows from this lemma and Claim 3.11.

LEMMA 5.2. *Let $k \geq 3$, $d \geq 2$, and let $\mathcal{C} \equiv 0$ be a simple and minimal $\Sigma\Pi\Sigma(k, d)$ circuit. Then*

$$\operatorname{rank}(\mathcal{C}) \leq 2^{O(k^2)} \log(d)^{k-2}.$$

Proof. From Theorem 4.1 we know that there exists a linear $(2, \frac{1}{12}, \frac{1}{4})$-LDC $E : \mathbb{F}^{n_1} \to \mathbb{F}^{n_2}$ with

$$\frac{\operatorname{rank}(\mathcal{C})}{P(k) \log(d)^{k-3}} \leq n_1 \quad \text{and} \quad n_2 \leq k \cdot d, \quad \text{where} \quad P(k) = 2^{O(k^2)}.$$

Theorem 1.2 now tells us that

$$n_2 \geq 2^{\frac{1}{96} n_1 - 1}.$$

Combining the above inequalities, we get the required bound on $\text{rank}(\mathcal{C})$. ∎

We now use Lemma 5.2 to prove Theorem 5.1.

Proof of Theorem 5.1. Since $\mathcal{C}$ is equal to zero, we can find a partition $T_1, \ldots, T_s \subset [k]$ such that the circuits $\mathcal{C}_{T_1}, \ldots, \mathcal{C}_{T_s}$ are all zero and minimal. Thus, the circuits $sim(\mathcal{C}_{T_1}), \ldots, sim(\mathcal{C}_{T_s})$ are all zero, simple, and minimal. By Lemma 5.2 we get that if $|T_i| \geq 3$ and $\deg(sim(\mathcal{C}_{T_i})) \geq 2$, then

$$\text{rank}(sim(\mathcal{C}_{T_i})) \leq 2^{O(k^2)} \log(d)^{k-2}.$$

If $|T_i| = 2$, then by Claim 4.2 we get that $\deg(sim(\mathcal{C}_{T_i})) = 0$ and so its rank is 1. If $\deg(sim(\mathcal{C}_{T_i})) \leq 1$, then its rank is at most k. Thus, we have covered all the possible cases, and the lemma follows.

6. PIT algorithms. In this section we use the structural theorem (Theorem 5.1), proved in the previous section, to devise the PIT algorithms of Theorem 1.5. Again, to simplify the notation, we give algorithms for $\Sigma\Pi\Sigma(k, d)$ circuits, which work in the same manner also for $\Sigma\Pi\Sigma$ circuits with k multiplication gates of degree d. We state our results for a general k; however, our algorithms will be most applicable when k is a constant.[9]

From Theorem 5.1 we know that every zero $\Sigma\Pi\Sigma$ circuit can be broken down into zero subcircuits whose ranks are small. The next two lemmas show that checking whether these low-rank circuits are zero can be done efficiently.

LEMMA 6.1. *Let $\mathcal{C}$ be a $\Sigma\Pi\Sigma(k, d)$ circuit with $\text{rank}(\mathcal{C}) = r$. Then, there exists a polynomial time algorithm, transforming $\mathcal{C}$ into a $\Sigma\Pi\Sigma(k, d)$ circuit $\mathcal{C}'$, such that*

- *$\mathcal{C} \equiv 0$ iff $\mathcal{C}' \equiv 0$,*
- *$\mathcal{C}'$ contains only r variables.*

Proof. The proof is a direct consequence of Claim 3.14: we apply an invertible linear transformation on $\mathcal{C}$, taking a set of r linearly independent vectors to $e_1, \ldots, e_r$. The transformed circuit will contain only the first r variables and will be zero iff $\mathcal{C}$ is zero. ∎

LEMMA 6.2. *Let $\mathcal{C}$ be a $\Sigma\Pi\Sigma(k, d)$ circuit, and let $r = \text{rank}(\mathcal{C})$, $s = size(\mathcal{C})$. Then we can check if $\mathcal{C} \equiv 0$*

1. *deterministically, in time $\text{poly}(s) \cdot (r + d)^r$;*
2. *probabilistically, in time $\text{poly}(s + \frac{1}{\epsilon})$, using $r \cdot \left(\log(d) + \log(\frac{1}{\epsilon})\right)$ random bits, with error probability ϵ.*

Proof. Using Lemma 6.1, we can transform $\mathcal{C}$ into a circuit $\mathcal{C}'$ with at most r variables, such that $\mathcal{C} \equiv 0$ iff $\mathcal{C}' \equiv 0$. Since $\mathcal{C}'$ contains only r variables, the number of different monomials in $\mathcal{C}'(x)$ is bounded by $\binom{r+d-1}{r-1} < (r + d)^r$. We can thus check if $\mathcal{C}' \equiv 0$ by computing the coefficients of all the monomials and seeing if they are all zero. This can be done in time $\text{poly}(s) \cdot (r + d)^r$. For the second part of the corollary, note that we can also check if $\mathcal{C}' \equiv 0$ probabilistically using the well-known Schwartz–Zippel algorithm [42, 50]. ∎

We are now ready to describe our PIT algorithm for $\Sigma\Pi\Sigma(k, d)$ circuits.

THEOREM 6.3. *Let $\mathcal{C}$ be a $\Sigma\Pi\Sigma(k, d)$ circuit, $s = size(\mathcal{C})$. Then, Algorithm 1 will check if $\mathcal{C} \equiv 0$. Further, the algorithm will run in time $\text{poly}(s) \cdot \exp\left(2^{O(k^2)} \log(d)^{k-1}\right)$.*

Proof. First, note that if $\mathcal{C}$ is nonzero, then the algorithm will never accept. (The algorithm accepts only when a partition of $\mathcal{C}$ into zero subcircuits is found.) Assume that $\mathcal{C}$ is zero. Then, by Theorem 5.1, there exists a partition, $T_1, \ldots, T_s \subset [k]$, of

[9] Our methods give subexponential time ($2^{o(n)}$) algorithms also if $k = o(\sqrt{\log n})$.

ALGORITHM 1. Deterministic algorithm.

input: A $\Sigma\Pi\Sigma(k,d)$ circuit $\mathcal{C}$.

(1) For every subset $T \subset [k]$ do the following:
- (1.1) Compute $r_T = \text{rank}(sim(\mathcal{C}_T))$.
- (1.2) If $r_T \leq 2^{O(k^2)} \log(d)^{k-2}$, then:
- check if $sim(\mathcal{C}_T) \equiv 0$ using part 1 of Lemma 6.2.

(2) If there exists a partition of $[k]$, such that for every set $T \subset [k]$ in the partition $sim(\mathcal{C}_T) \equiv 0$, then **accept**. Otherwise **reject**.

$[k]$ such that the circuits $sim(\mathcal{C}_{T_1}), \ldots, sim(\mathcal{C}_{T_s})$ are all zero and that for all $i \in [s]$ the rank of $sim(\mathcal{C}_{T_i})$ is bounded by $2^{O(k^2)} \log(d)^{k-2}$. Therefore, for every $\mathcal{C}_{T_i}$ we will check whether $sim(\mathcal{C}_{T_i}) \equiv 0$ in step (1.2) of the algorithm. Since we go over all subsets of $[k]$, we are bound to find the above partition and accept.

As for the running time of the algorithm, notice that we apply the algorithm from Lemma 6.2 only on circuits whose rank is smaller than $2^{O(k^2)} \log(d)^{k-2}$. Therefore, by Lemma 6.2, the time spent in each invocation of step (1.2) is at most

$$\text{poly}(s) \cdot \exp\left(2^{O(k^2)} \log(d)^{k-1}\right).$$

Step (1.2) is run at most 2^k times, and so the total running time is also

$$\text{poly}(s) \cdot \exp\left(2^{O(k^2)} \log(d)^{k-1}\right).$$

(The running times of all the other steps of the algorithm are "swallowed up" by the running time of step (1.2).) □

We can modify Algorithm 1 so that it will use a probabilistic check in step (1.2). This will result in a probabilistic PIT algorithm for $\Sigma\Pi\Sigma$ circuits, which uses fewer random bits than previous algorithms.

ALGORITHM 2. Probabilistic algorithm.

input: A $\Sigma\Pi\Sigma(k,d)$ circuit $\mathcal{C}$. An error probability ϵ.

(1) For every subset $T \subset [k]$ do the following:
- (1.1) Compute $r_T = \text{rank}(sim(\mathcal{C}_T))$.
- (1.2) If $r_T \leq 2^{O(k^2)} \log(d)^{k-2}$, then: check if $sim(\mathcal{C}_T) \equiv 0$ probabilistically,
- using part 2 of Lemma 6.2, with error probability $\epsilon 2^{-k}$.

(2) If there exists a partition of $[k]$, such that for every set $T \subset [k]$ in the partition $sim(\mathcal{C}_T) \equiv 0$, then **accept**. Otherwise **reject**.

THEOREM 6.4. *Let $\mathcal{C}$ be a $\Sigma\Pi\Sigma(k,d)$ circuit, $s = size(\mathcal{C})$. Then, Algorithm 2 will check if $\mathcal{C} \equiv 0$. Further, the algorithm will run in time* $\text{poly}\left(s + \frac{2^k}{\epsilon}\right)$, *will use $2^{O(k^2)} \log(d)^{k-1} \log(\frac{1}{\epsilon})$ random bits, and will make an error with probability less than ϵ.*

Proof. Using the same reasoning as in the proof of Theorem 6.3, we see that the algorithm can make an error only if one of the checks in step (1.2) fails. By the union bound, this happens with probability of at most ϵ.

Each check in step (1.2) takes time poly $\left(s+\frac{2^k}{\epsilon}\right)$. And so the total running time is

$$2^k \cdot \text{poly}\left(s+\frac{2^k}{\epsilon}\right) = \text{poly}\left(s+\frac{2^k}{\epsilon}\right).$$

By part 2 of Lemma 6.2, the number of random bits used in step (1.2) is at most $r_T \cdot \left(\log(d)+\log(\frac{1}{\epsilon})\right)$. Since we run the probabilistic check only when $r_T \leq 2^{O(k^2)}\log(d)^{k-2}$, it follows that the number of random bits used in each invocation of step (1.2) is bounded by $2^{O(k^2)}\log(d)^{k-1}\log(\frac{1}{\epsilon})$. As we can use the same random bits in all tests, this is also the total number of random bits needed. □

We restate the last two theorems for the case when k is a constant.

THEOREM 6.5. *Let $\mathcal{C}$ be a $\Sigma\Pi\Sigma(k,d)$ circuit, k a constant, $s = size(\mathcal{C})$. Then we can check if $\mathcal{C} \equiv 0$*

1. *deterministically, in quasi-polynomial time,*
2. *probabilistically, in time* $\text{poly}(s+\frac{1}{\epsilon})$*, using* $O\left(\log(d)^{k-1}\log(\frac{1}{\epsilon})\right)$ *random bits, with error probability ϵ.*

Note that Theorems 6.3, 6.4, and 6.5 imply the first two claims of Theorem 1.5.

6.1. Multilinear circuits. This section deals with a special kind of $\Sigma\Pi\Sigma$ circuit, described by the following definition.

DEFINITION 6.6. *A $\Sigma\Pi\Sigma$ circuit $\mathcal{C}$ is* multilinear *if each of its multiplication gates computes a multilinear polynomial. (A polynomial is multilinear if the degree of every variable is at most one.)*

Let

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

be a $\Sigma\Pi\Sigma$ circuit. Denote by $V_{ij} \subset [n]$ the set of variables appearing in the linear form L_{ij}. From Definition 6.6 we see that $\mathcal{C}$ is multilinear iff for every $i \in [k]$, and for every $j_1 \neq j_2$, we have

$$V_{ij_1} \cap V_{ij_2} = \emptyset.$$

This condition implies that for every $i \in [k]$ the linear functions $\{L_{ij}\}_{j\in[d_i]}$ are linearly independent. This leads to the following observation.

Observation 6.7. If $\mathcal{C}$ is a multilinear $\Sigma\Pi\Sigma$ circuit of degree d, then $\text{rank}(\mathcal{C}) \geq d$.

Combining this observation and Theorem 1.4, we get the following theorem.

THEOREM 6.8. *Let $\mathcal{C} \equiv 0$ be a multilinear $\Sigma\Pi\Sigma$ circuit with k multiplication gates ($k \geq 3$), which is simple and minimal. Let $d = deg(\mathcal{C})$; then*

$$d \leq 2^{O(k^2)}\log(d)^{k-2}. \tag{17}$$

COROLLARY 6.9. *There exists an integer function $D(k) = 2^{O(k^2)}$ such that every multilinear $\Sigma\Pi\Sigma$ circuit $\mathcal{C}$ with k multiplication gates, which is simple and equal to zero, and of degree $d = deg(\mathcal{C}) > D(k)$, is not minimal.*

Proof. Fix k, and consider (17). This inequality holds only if $d \leq 2^{O(k^2)} = D(k)$. Thus, if $d > D(k)$, then the conditions of Theorem 6.8 are not satisfied. In particular, if $\mathcal{C} \equiv 0$ and is simple, then it is not minimal. □

We can use Corollary 6.9 to improve the algorithm given in section 6, in the case that the given circuit is multilinear.

THEOREM 6.10. *Let $\mathcal{C}$ be a multilinear $\Sigma\Pi\Sigma$ circuit, of size s, with k multiplication gates. We can check if $\mathcal{C} \equiv 0$ in time* $\mathrm{poly}(s) \cdot \exp\left(2^{O(k^2)}\right)$. *Thus, if k is constant, the algorithm runs in polynomial time.*

Proof. The algorithm is the same as Algorithm 1. (It does not matter that our circuit is not a $\Sigma\Pi\Sigma(k,d)$ circuit.) The only difference is that by Corollary 6.9 we only have to consider subcircuits $\mathcal{C}_T$ such that the degree of $sim(\mathcal{C}_T)$ is less than $D(|T|) = 2^{O(k^2)}$. The running time is computed in a similar fashion. In step (1) we go over at most $2^{O(k^2)}$ partitions of $[k]$. Computing the rank of the subcircuit $\mathcal{C}_T$ can be done in polynomial time in s. Finally, by part 1 of Lemma 6.2, step (1.2) requires time $O\left(D(|T|)^{2^{O(k^2)}}\right) = \exp\left(2^{O(k^2)}\right)$. ∎

Theorem 6.10 implies the third claim of Theorem 1.5, thus completing the proof of the theorem.

7. Conclusions and open problems. Finding efficient deterministic PIT algorithms for general arithmetic circuits is a long-standing open problem. We made the first step toward an efficient algorithm for PIT for depth 3 circuits by giving PIT algorithms for depth 3 circuits with bounded top fan-in; however, the general case of depth 3 circuits is still open. In view of [25] it is natural to look for algorithms for PIT for restricted models of arithmetic circuits in which lower bounds are known. Raz [39] proved a quasi-polynomial lower bound for *multilinear* arithmetic formulas computing the determinant and the permanent. Thus, giving PIT algorithms for multilinear formulas is a very interesting, and maybe even a solvable, problem.

The key to our result is the relation we have found between LDCs and depth 3 circuits. Previously, relations between circuits and error correcting codes were known only for bilinear circuits over finite fields [10, 45]. It should be very interesting to find new relations between codes and arithmetic circuits. Another interesting question is whether the relation that we have found is tight. In particular we believe that in Theorem 1.3 one should be able to replace $r/2^{O(k^2)}\log(d)^{k-3}$ with $O(r/k)$. A related question regards how to improve Theorem 1.4. We believe that for minimal and simple circuits over fields of characteristic zero the rank should be $O(k)$. We have found circuits that are minimal and simple, with $r = 3k-2$, and we think that it would be an interesting task to come up with (minimal and simple) circuits that have larger rank. As mentioned in the introduction, [27] showed that over characteristic p there are identically zero depth 3 circuits with top fan-in p for odd p (for $p = 2$ the top fan-in is 3) whose rank is $\log_p(d)$.

We conclude this section with a geometrical problem related to depth 3 circuits with three multiplication gates. The famous Sylvester–Gallai theorem states that every set of n points in the plane having the property that every line that contains two points from the set also contains a third point from the set is contained in a line. Consider the following generalization of the problem (colored version in the projective plane): instead of one set of points we have three different sets. Each set is of size n. The points in the sets correspond to vectors from the r-dimensional sphere, and every two such vectors are linearly independent. The condition on the sets is that every two-dimensional subspace that contains points from two different sets also contains a point from the third set.[10] What can be said about r in this case? Clearly the

[10] Alternatively, the points belong to the r-dimensional projective space, and every line that contains points from two different sets also contains a point from the third set.

r-dimensional sphere can be embedded into the $(r+1)$-dimensional sphere so we only consider "irreducible" arrangements in which the vectors corresponding to the points span the whole space. Using our lower bound on LDCs, we can show that r is at most $O(\log n)$; however, we think that this can be improved. In particular we conjecture that r is bounded (maybe even $r = 2$). If our conjecture is true, then it will serve as evidence that for $k = 3$ the rank of every simple and minimal depth 3 circuit, which is identically zero, is bounded.

We now give an example that shows the relation of the problem to identically zero depth 3 circuits with three multiplication gates. Consider the following equality $x_1^n - x_2^n = \prod_{i=0}^{n-1}(x_1 - w^i x_2)$, where w is a primitive nth root of unity. We get that

$$\sum_{i=1}^{k-1}\prod_{j=0}^{n-1}(x_i - w^j x_{i+1}) + \prod_{j=0}^{n-1}(x_k - w^j x_1) = 0.$$

Notice that this is an identically zero depth 3 circuit with k multiplication gates. For the special case of $k = 3$ we get that

$$\prod_{j=0}^{n-1}(x_1 - w^j x_2) + \prod_{j=0}^{n-1}(x_2 - w^j x_3) + \prod_{j=0}^{n-1}(x_3 - w^j x_1) = 0.$$

Each multiplication gate corresponds to a different set of points. We map each linear function $x_1 - w^j x_2$ from the first gate to the point $(\frac{1}{\sqrt{2}}, \frac{-w^j}{\sqrt{2}}, 0)$; similarly, we map the functions of the second multiplication gate to $\{(0, \frac{1}{\sqrt{2}}, \frac{-w^j}{\sqrt{2}})\}_{j=0,\ldots,n-1}$, etc. Clearly all the points belong to the two-dimensional sphere in $\mathbb{C}^3$. It is easy to see that for each point from the first set (i.e., points coming from the first multiplication gate) and each point from the second set there is a unique point from the third set that belongs to the same two-dimensional space (similarly if we pick the first and third sets, etc.). Therefore this construction satisfies our requirements. Our question is, Can such arrangements be found in higher dimensions?

Acknowledgments. The authors would like to thank Ran Raz and Avi Wigderson for helpful discussions during various stages of this work. A. S. would like to thank Boaz Barak, Valentine Kabanets, and Salil Vadhan for useful conversations on the topic of the work. We are grateful to Ran Raz for many valuable comments that improved the presentation of the results.

REFERENCES

[1] M. Agrawal and S. Biswas, *Primality and identity testing via Chinese remaindering*, J. ACM, 50 (2003), pp. 429–443.

[2] M. Agrawal, N. Kayal, and N. Saxena, *PRIMES is in p*, Ann. of Math., 160 (2004), pp. 781–793.

[3] A. Akavia, S. Goldwasser, and M. Safra, *A unifying approach for proving hardcore predicates using list decoding*, in Proceedings of the 44th IEEE Symposium on Foundations of Computer Science, Cambridge, MA, 2003, pp. 146–155.

[4] L. Babai, L. Fortnow, L. A. Levin, and M. Szegedy, *Checking computations in polylogarithmic time*, in Proceedings of the 23rd Annual ACM Symposium on Theory of Computing, ACM Press, New York, 1991, pp. 21–32.

[5] W. Baur and V. Strassen, *The complexity of partial derivatives*, Theoret. Comput. Sci., 22 (1983), pp. 317–330.

[6] D. Beaver and J. Feigenbaum, *Hiding instances in multioracle queries*, in Proceedings of the Seventh Annual Symposium on Theoretical Aspects of Computer Science, Springer, New York, 1990, pp. 37–48.

[7] A. Beimel and Y. Ishai, *Information-theoretic private information retrieval: A unified construction*, in Automata, Languages and Programming, Lecture Notes in Comput. Sci. 2076, Springer, New York, 2001, pp. 912–926.

[8] A. Beimel, Y. Ishai, E. Kushilevitz, and J.-F. Raymond, *Breaking the $O(n^{1/2k-1})$ barrier for information-theoretic private information retrieval*, in Proceedings of the 43rd Annual IEEE Symposium on Foundations of Computer Science, 2002, pp. 261–270.

[9] M. Ben-Or and P. Tiwari, *A deterministic algorithm for sparse multivariate polynomial interpolation*, in Proceedings of the 20th Annual ACM Symposium on Theory of Computing, ACM Press, New York, 1988, pp. 301–309.

[10] N. H. Bshouty, *A lower bound for matrix multiplication*, SIAM J. Comput., 18 (1989), pp. 759–765.

[11] S. Chari, P. Rohatgi, and A. Srinivasan, *Randomness-optimal unique element isolation with applications to perfect matching and related problems*, SIAM J. Comput., 24 (1995), pp. 1036–1050.

[12] Z.-Z. Chen and M.-Y. Kao, *Reducing randomness via irrational numbers*, in Proceedings of the 29th Annual ACM Symposium on Theory of Computing, ACM Press, New York, 1997, pp. 200–209.

[13] B. Chor, O. Goldreich, E. Kushilevitz, and M. Sudan, *Private information retrieval*, in Proceedings of the 36th Annual IEEE Symposium on Foundations of Computer Science, 1995, pp. 41–50.

[14] A. Deshpande, R. Jain, T. Kavitha, J. Radhakrishnan, and S. V. Lokam, *Lower bounds for adaptive locally decodable codes*, Random Structures Algorithms, 27 (2005), pp. 358–378.

[15] J. Feigenbaum and L. Fortnow, *Random-self-reducibility of complete sets*, SIAM J. Comput., 22 (1993), pp. 994–1005.

[16] P. Gemmell, R. J. Lipton, R. Rubinfeld, M. Sudan, and A. Wigderson, *Self-testing/correcting for polynomials and for approximate functions*, in Proceedings of the 23rd Annual ACM Symposium on Theory of Computing, ACM Press, New York, 1991, pp. 33–42.

[17] P. Gemmell and M. Sudan, *Highly resilient correctors for polynomials*, Inform. Process. Lett., 43 (1992), pp. 169–174.

[18] O. Goldreich, H. J. Karloff, L. J. Schulman, and L. Trevisan, *Lower bounds for linear locally decodable codes and private information retrieval*, Comput. Complexity, 15 (2006), pp. 263–296.

[19] O. Goldreich and L. A. Levin, *A hard core predicate for all one way functions*, in Proceedings of the 21st ACM Symposium on Theory of Computing, Seattle, WA, 1989, pp. 25–32.

[20] O. Goldreich, R. Rubinfeld, and M. Sudan, *Learning polynomials with queries: The highly noisy case*, SIAM J. Discrete Math., 13 (2000), pp. 535–570.

[21] D. Grigoriev and M. Karpinski, *An exponential lower bound for depth* 3 *arithmetic circuits*, in Proceedings of the 30th Annual ACM Symposium on Theory of Computing, ACM Press, New York, 1998, pp. 577–582.

[22] D. Grigoriev, M. Karpinski, and M. F. Singer, *Fast parallel algorithms for sparse multivariate polynomial interpolation over finite fields*, SIAM J. Comput., 19 (1990), pp. 1059–1063.

[23] D. Grigoriev and A. A. Razborov, *Exponential complexity lower bounds for depth* 3 *arithmetic circuits in algebras of functions over finite fields*, in Proceedings of the 39th Annual IEEE Symposium on Foundations of Computer Science (FOCS), 1998, pp. 269–278.

[24] M. Jerrum and M. Snir, *Some Exact Complexity Results for Straight-Line Computations over Semi-Rings*, Technical report CRS-58-80, University of Edinburgh, Edinburgh, UK, 1980.

[25] V. Kabanets and R. Impagliazzo, *Derandomizing polynomial identity tests means proving circuit lower bounds*, in Proceedings of the 35th Annual ACM Symposium on Theory of Computing, ACM Press, New York, 2003, pp. 355–364.

[26] J. Katz and L. Trevisan, *On the efficiency of local decoding procedures for error-correcting codes*, in Proceedings of the 32nd Annual ACM Symposium on Theory of Computing, ACM Press, New York, 2000, pp. 80–86.

[27] N. Kayal and N. Saxena, *Polynomial identity testing for depth* 3 *circuits*, in Proceedings of the 21st Annual IEEE Conference, 2006, pp. 9–17.

[28] I. Kerenidis and R. de Wolf, *Exponential lower bound for* 2*-query locally decodable codes via a quantum argument*, in Proceedings of the 35th Annual ACM Symposium on Theory of Computing, ACM Press, New York, 2003, pp. 106–115.

[29] A. R. Klivans and D. Spielman, *Randomness efficient identity testing of multivariate polynomials*, in Proceedings of the 33rd Annual ACM Symposium on Theory of Computing, ACM Press, New York, 2001, pp. 216–223.

[30] L. A. Levin, *One–way functions and pseudorandom generators*, Combinatorica, 7 (1987), pp. 357–363.

[31] D. Lewin and S. Vadhan, *Checking polynomial identities over any field: Towards a derandomization?*, in Proceedings of the 30th Annual ACM Symposium on Theory of Computing, ACM Press, New York, 1998, pp. 438–447.
[32] R. J. Lipton, *Efficient checking of computations*, in STACS 90: Proceedings of the 7th Annual Symposium on Theoretical Aspects of Computer Science, C. Choffrut and T. Lengauer, eds., Springer, Berlin, Heidelberg, 1990, pp. 207–215.
[33] L. Lovász, *On determinants, matchings, and random algorithms*, in Fundamentals of Comptuation Theory: Proceedings of the Conference on Algebraic, Arithmetic, and Categorial Methods in Computation Theory, Vol. 2, Akademie-Verlag, Berlin, 1979, pp. 565–574.
[34] K. Mulmuley, U. V. Vazirani, and V. V. Vazirani, *Matching is as easy as matrix inversion*, in Proceedings of the Nineteenth Annual ACM Conference on Theory of Computing, ACM Press, New York, 1987, pp. 345–354.
[35] N. Nisan, *Lower bounds for noncommutative computation*, in Proceedings of the 23rd Annual ACM Symposium on Theory of Computing, ACM Press, New York, 1991, pp. 410–418.
[36] N. Nisan and A. Wigderson, *Lower bounds on arithmetic circuits via partial derivatives*, Comput. Complexity, 6 (1997), pp. 217–234.
[37] K. Obata, *Optimal lower bounds for 2-query locally decodable linear codes*, in Proceedings of the Sixth International Workshop on Randomization and Approximation Techniques, Springer, New York, 2002, pp. 39–50.
[38] P. Pudlak, *Communication in bounded depth circuits*, Combinatorica, 14 (1994), pp. 203–216.
[39] R. Raz, *Multi-linear formulas for permanent and determinant are of super-polynomial size*, in Proceedings of the 36th Annual ACM Symposium on Theory of Computing, ACM Press, New York, 2004, pp. 633–641.
[40] R. Raz and A. Shpilka, *Lower bounds for matrix product, in bounded depth circuits with arbitrary gates*, in Proceedings of the 33rd Annual ACM Symposium on Theory of Computing, ACM Press, New York, 2001, pp. 409–418.
[41] R. Raz and A. Shpilka, *Deterministic polynomial identity testing in non-commutative models*, Comput. Complexity, 14 (2005), pp. 1–19.
[42] J. T. Schwartz, *Fast probabilistic algorithms for verification of polynomial identities*, J. ACM, 27 (1980), pp. 701–717.
[43] E. Shamir and M. Snir, *Lower Bounds on the Number of Multiplications and the Number of Additions in Monotone Computations*, Research report RC6757, IBM Thomas J. Watson Research Center, Yorktown Heights, NY, 1977.
[44] E. Shamir and M. Snir, *On the depth complexity of formulas*, Math. Systems Theory, 13 (1980), pp. 301–322.
[45] A. Shpilka, *Lower bounds for matrix product*, SIAM J. Comput., 32 (2003), pp. 1185–1200.
[46] A. Shpilka and A. Wigderson, *Depth-3 arithmetic formulae over fields of characteristic zero*, in Proceedings of the 14th Annual IEEE Conference on Computational Complexity, IEEE Computer Society, Piscataway, NJ, 1999, p. 87.
[47] V. Strassen, *Die berechnungskomplexität von elementarsymmetrischen Funktionen und von Interpolationskoeffizienten*, Numer. Math., 20 (1972/73), pp. 238–251.
[48] P. Tiwari and M. Tompa, *A direct version of Shamir and Snir's lower bounds on monotone circuit depth*, Inform. Process. Lett., 49 (1994), pp. 243–248.
[49] L. Trevisan, *Some applications of coding theory in computational complexity*, Quad. Mat. 13 (2004), pp. 347–424.
[50] R. Zippel, *Probabilistic algorithms for sparse polynomials*, in Proceedings of the International Symposium on Symbolic and Algebraic Computation, Springer, New York, 1979, pp. 216–226.