Quasi-polynomial Hitting-set for Set-depth-Δ Formulas

Manindra Agrawal
Indian Institute of Technology
Kanpur, India
manindra@iitk.ac.in

Chandan Saha
Indian Institute of Science
Bangalore, India
chandan@csa.iisc.ernet.in

Nitin Saxena
Hausdorff Center for Maths
Bonn, Germany
ns@hcm.uni-bonn.de

ABSTRACT

We call a depth-4 formula C *set-depth-4* if there exists a (unknown) partition $X_1 \sqcup \cdots \sqcup X_d$ of the variable indices $[n]$ that the top product layer respects, i.e. $C(\boldsymbol{x}) = \sum_{i=1}^{k} \prod_{j=1}^{d} f_{i,j}(\boldsymbol{x}_{X_j})$, where $f_{i,j}$ is a *sparse* polynomial in $\mathbb{F}[\boldsymbol{x}_{X_j}]$. Extending this definition to any depth - we call a depth-Δ formula C (consisting of alternating layers of Σ and Π gates, with a Σ-gate on top) a *set-depth-*Δ formula if every Π-layer in C respects a (unknown) partition on the variables; if Δ is even then the product gates of the bottom-most Π-layer are allowed to compute arbitrary monomials. In this work, we give a hitting-set generator for set-depth-Δ formulas (over *any* field) with running time polynomial in $\exp((\Delta^2 \log s)^{\Delta-1})$, where s is the size bound on the input set-depth-Δ formula. In other words, we give a *quasi*-polynomial time *blackbox* polynomial identity test for such constant-depth formulas. Previously, the very special case of $\Delta = 3$ (also known as *set-multilinear* depth-3 circuits) had no known sub-exponential time hitting-set generator. This was declared as an open problem by Shpilka & Yehudayoff (FnT-TCS 2010); the model being first studied by Nisan & Wigderson (FOCS 1995) and recently by Forbes & Shpilka (STOC 2012 & ECCC TR12-115). Our work settles this question, not only for depth-3 but, up to depth $\epsilon \log s / \log\log s$, for a fixed constant $\epsilon < 1$. The technique is to investigate depth-Δ formulas via depth-$(\Delta-1)$ formulas over a *Hadamard algebra*, after applying a 'shift' on the variables. We propose a new algebraic conjecture about the *low-support rank-concentration* in the latter formulas, and manage to prove it in the case of set-depth-Δ formulas.

Categories and Subject Descriptors

F.2.1 [**Analysis of Algorithms and Problem Complexity**]: Numeric Algorithms and Problems—*Computations on polynomials*; I.1.2 [**Symbolic and Algebraic Manipulation**]: Algorithms—*Algebraic Algorithms*

General Terms

Theory, Algorithms

Keywords

identity testing, hitting-set, set-multilinear formula, Hadamard algebra, low-support rank concentration

1. INTRODUCTION

Polynomial identity testing (PIT) - the algorithmic question of examining if an arithmetic circuit computes an identically zero polynomial - has received some attention in the recent times, primarily due to its close connection to circuit lower bounds. It is now known that a complete (blackbox) derandomization of PIT for depth-4 formulas, via a particular kind of pseudorandom generator, implies $\mathsf{VP} \neq \mathsf{VNP}$ [2] (an algebraic analogue of the much coveted goal: $\mathsf{P} \neq \mathsf{NP}$). It is also known that $\mathsf{VP} \neq \mathsf{VNP}$ must necessarily be shown before proving $\mathsf{P} \neq \mathsf{NP}$ [28, 26]. Blackbox identity testing (or, the problem of designing hitting-set generators), being a promising approach to proving lower bounds, naturally calls for a closer examination. Moreover, a line of work [29, 4, 14, 11] has established that a complete blackbox derandomization of depth-3 PIT implies a quasi-polynomial time blackbox PIT algorithm for general *low degree* circuits. Towards this, some progress has been made in the form of polynomial time hitting-set generators for the following formulas:

- depth-2 (cf. [6]),
- depth-3 bounded top fanin [3, 23],
- bounded depth, constant-occur [3],
- (quasi-poly hitting-set for) multilinear constant-read formulas [5], read-once oblivious ABPs [9],

among some others (cf. [22, 25]). The hope is, by studying these special but interesting models we might develop a deeper understanding of the nature of hitting-sets and thereby get a clue as to what techniques can be lifted to solve PIT in general (i.e. for depth-3 formulas). One such potentially effective technique is the study of *partial derivatives* of formulas.

Despite the apparent difference between the approaches of [3] and [5], at a finer level they share a common ingredient - the use of partial derivatives. The partial derivative based method was introduced in the seminal paper by Nisan and Wigderson [16] for proving circuit lower bounds, and since then it has been successfully applied (with more sophistications) to prove various interesting results on lower bounds, identity testing and reconstruction of circuits [3, 5, 10, 12] (cf. [25, 7] for more).

Partial derivatives & shifting: From a geometric viewpoint, partial derivatives *shift* the variables by 'some' amount - for e.g., if $f(x_1, x_2, \ldots, x_n)$ is a multilinear polynomial then its partial derivative with respect to x_1 is $f(x_1+1, x_2, \ldots, x_n) - f(x_1, \ldots, x_n)$. Out of curiosity, one might ask what happens if we shift the polynomial by arbitrary field constants? If we shift a monomial $f(\mathbf{x}) = x_1 x_2 \ldots x_n$ by $\mathbf{c} = (c_1, \ldots, c_n) \in \mathbb{F}^n$, $c_i \neq 0$, we get the polynomial $f(\mathbf{x}+\mathbf{c}) = (x_1+c_1)(x_2+c_2)\ldots(x_n+c_n)$. Something interesting has happened here: The polynomial $f(\mathbf{x}+\mathbf{c})$ has many *low-support* monomials. By a low-support monomial, we mean that the number of variables involved in the monomial is less than a predefined small quantity, say ℓ.

Is it possible that shifting has a similar effect on a more general polynomial $f(\mathbf{x})$, i.e. $f(\mathbf{x}+\mathbf{c})$ has low-support monomials with nonzero coefficients, if $f \neq 0$? Surely, this is true if $\mathbf{c}$ is chosen randomly from $\mathbb{F}^n$ [24, 30]. But, f is not just any arbitrary polynomial, it is a polynomial computed by a formula (say, a depth-3 formula). This makes it an interesting proposition to investigate the following derandomization question: If $f \neq 0$ be a polynomial computed by a formula, is it possible to efficiently compute a *small* collection of points $\mathcal{T} \subset \mathbb{F}^n$, such that there exists a $\mathbf{c} \in \mathcal{T}$ for which $f(\mathbf{x}+\mathbf{c})$ has a low-support monomial with nonzero coefficient?

If the answer to the above question is yes, then it is fairly straightforward to do an efficient blackbox identity test on f: For the right choice of $\mathbf{c} \in \mathcal{T}$, $g(\mathbf{x}) = f(\mathbf{x}+\mathbf{c}) \neq 0$ has a low-support monomial. To witness that $g(\mathbf{x}) \neq 0$, it suffices to keep a set of ℓ variables intact and set the remaining $n-\ell$ variables to zero in g; running over all possible choices of ℓ variables whom we choose to keep intact, we can witness the fact that $g \neq 0$. Since ℓ is presumably small, $g(\mathbf{x})$ restricted to ℓ variables is a sparse polynomial which can be efficiently tested for nonzeroness in a blackbox fashion [6].

Indeed, the above intuition is true for the class of set-depth-Δ formulas (defined in §1.1) - an interesting class capturing many other previously studied models (see §1.1), including *set-multilinear* depth-3 circuits.

Set-multilinear depth-3 circuits: A circuit $C = \sum_{i=1}^{k} \prod_{j=1}^{d} f_{i,j}(\boldsymbol{x}_{X_j})$ is called a set-multilinear depth-3 circuit if $X_1 \sqcup \ldots \sqcup X_d$ is a partition of the variable indices $[n]$ and $f_{i,j}(\boldsymbol{x}_{X_j})$ is a linear polynomial in the variables $\boldsymbol{x}_{X_j}$ i.e. the set of variables corresponding to the partition X_j. The set-multilinear depth-3 model, first defined by [16], kicked off a flurry of activity. Though innocent-looking, it has led researchers to various arithmetic inventions – the *partial derivative* method for circuit lower bounds [16], noncommutative whitebox PIT [18], the relationship between *tensor-rank* and super-polynomial circuit lower bounds [17], hitting-set for tensors, low-rank recovery of matrices, rank-metric codes [8], and reconstruction (or learnability) of circuits [13]. Although, an exponential lower bound for set-multilinear depth-3 circuits is known [16], the closely associated problem of efficient blackbox identity testing on this model remained an open question, until this work.

Hitting-set for set-depth-Δ formulas - A whitebox deterministic polynomial time identity test for set-depth-Δ follows from the noncommutative PIT results [18]. We are interested in *blackbox* PIT and hence, we cannot see inside C and the underlying partitions of $[n]$. The only information we have is the circuit-size bound, s. To our knowledge, there was no sub-exponential time hitting-set known for the set-depth-Δ model. Our work improves this situation to quasi-polynomial for *any* underlying field (refer Theorem 1). We remark that even the very special case of set-multilinear depth-3 circuits had no sub-exponential hitting-set known [25, Problem 27]; closest being the recent result of [8] where they give a quasi-polynomial hitting-set for *tensors*, i.e. the *knowledge of the sets* $X_1, \ldots, X_d$ is required.

Furthermore, set-depth-4 covers other well studied models - diagonal circuits [21] & semi-diagonal circuits [20] - that had whitebox identity tests but no blackbox sub-exponential PIT were known. For these (and set-multilinear depth-3), our hitting-set has time complexity $s^{O(\log s)}$, although, for general set-depth-4 it requires $s^{O(\log^2 s)}$.

Depth-3 formulas being the ultimate frontier for PIT (and lower bounds) [4, 11], one might wonder about the utility of our result on hitting-set for set-depth-Δ formulas beyond $\Delta = 3$. It turns out that there is an interesting connection: We show that a quasi-polynomial hitting-set generator for set-depth-6 formulas implies a quasi-polynomial hitting-set generator for depth-3 formulas of the form $C = \sum_{i=1}^{k} \prod_{j=1}^{d} f_{i,j}(\boldsymbol{x}_{X_j})^{e_{i,j}}$, where $X_1 \sqcup \ldots \sqcup X_d$ defines a partition on $[n]$ and $f_{i,j}$ are linear polynomials. Since arbitrary powers $e_{i,j} \geq 0$ are allowed, the above depth-3 model is stronger than set-multilinear depth-3 formulas (as there is no restriction of multilinearity). This appears to be a step forward towards derandomizing general depth-3 PIT and provides us with a good motivation to understand the strength of our approach against depth-3 formulas.

Few more words on our approach: The $\mathbb{F}$-linear space of polynomials spanned by shifting f by arbitrary points in $\mathbb{F}^n$ is identical to the space spanned by the partial derivatives of f when $\text{char}(\mathbb{F}) = 0$ (if $\text{char}(\mathbb{F}) = p > 0$, the *shifted polynomials space* might strictly subsume the partial derivatives space, e.g. $f(x) = x^p$). The partial derivative based methods are relatively better studied in the literature. But, we choose to work with the shifted polynomials space because of the following reasons: (1) Shifting by points in $\mathbb{F}^n$ can be done even if f is presented as a blackbox, (2) we do not know how to work out the present proofs using partial derivatives, (3) if $\text{char}(\mathbb{F}) > 0$, all the nontrivial partial derivatives might vanish identically giving little information about f, whereas shifted polynomials always retain the 'identity' of f. For these, we work with shifted polynomials (see also [27]).

In fact, we shift the circuit by *formal* variables and examine how it changes by considering a *transfer* matrix T. The transfer matrix originates from the study of a formula with field coefficients via a 'simpler' one having *Hadamard algebra* coefficients. This makes the transfer process more amenable to a study using matrices and linear algebra; proving properties that are vaguely reminiscent of the case of top-fanin $k = 1$. The technicality lies in proving the invertibility of the transfer matrix, which is an exponential-sized matrix.

The use of Hadamard algebra is implicit in the whitebox PIT of [18] and the study of PIT over commutative algebras of [19]. The novelty of the present approach lies in understanding the effect of shift by viewing it through the lens of Hadamard algebra which reveals an interesting phenomenon, we call *low-support rank concentration.*

1.1 The results (stated formally)

Set-depth & set-height formulas - Let C be an arithmetic formula over a field $\mathbb{F}$ in n variables $\boldsymbol{x}$, consisting of alternating layers of addition (Σ) and multiplication (Π) gates, with a Σ-gate on top. The number of layers of Π-gates in C is called the *product-depth* (or simply *height*) of C and will be denoted by H. Naturally, the depth of C - which is the number of layers of gates in C - is either $\Delta = 2H$ or $2H+1$. Counting the Π-layers from the top, we label these layers by numbers in the range $[H]$ and will be referring to a layer as the h-th Π-layer in C, for $h \in [H]$.

We say that C is a *set-depth-Δ* formula if for every h-th Π-layer in C, there exists a partition $X_{h,1} \sqcup \cdots \sqcup X_{h,d_h}$ of variable indices $[n]$ that the product gates of the h-th Π-layer respect. In other words, for every $h \in [H]$ the i-th product gate in the h-th Π-layer computes a polynomial of the form $\prod_{j=1}^{d_h} f_{i,j}(\boldsymbol{x}_{X_{h,j}})$, where each $f_{i,j}(\boldsymbol{x}_{X_{h,j}})$ is a set-depth-$(\Delta - 2h)$ formula of height $H-h$ on the variable set $\boldsymbol{x}_{X_{h,j}}$. If $\Delta = 2H$ then the product gates of the H-th Π-layer are allowed to compute arbitrary monomials, i.e. here the H-th Π-layer *need not* respect any partition of the variables.

We will also refer to C as a *set-height-H* formula [1]. *Size* of C, denoted by s or $|C|$, is the number of gates (including the input gates) in C.

Theorem 1 (Main). *There is a hitting-set generator for set-height-H formulas, of size s, that runs in time polynomial in* $\exp((2H^2 \log s)^{H+1})$, *over any field* $\mathbb{F}$.

Remarks - For blackbox PIT of set-multilinear depth-3 formulas this gives a *quasi-polynomial* time complexity of $s^{O(\log s)}$. For constants $H > 1$ the formula may *not* be multilinear, though the hitting-set remains quasi-polynomial. The time complexity remains sub-exponential up to $H = \epsilon \log s / \log\log s$, for a fixed constant $\epsilon < 1$.

An interesting model that is not set-depth-Δ but still Theorem 1 could be applied is - semi-diagonal formula. The reason being the *duality* transformation [21, 20] that helps us view it as a set-depth-4 formula. We recall - a depth-4 ($\Sigma\Pi\Sigma\Pi$) formula C is *semi-diagonal* if, for all i, its i-th (top) product-gate computes a polynomial of the form $m_i \cdot \prod_{j=1}^{b} f_{i,j}^{e_{i,j}}$, where m_i is a monomial, $f_{i,j}$ is a sum of univariate polynomials, and b is a constant. Another interesting application of Theorem 1 is blackbox PIT for set-depth-3 formulas *with powers.*

Corollary 2 (Semi-diag. depth-4). *There is a hitting-set generator for semi-diagonal depth-4 formulas, of size s, that runs in time* $s^{O(\log s)}$.

Corollary 3 (Set-depth-3 with powers). *Consider a depth-3* $C = \sum_{i=1}^{k} \prod_{j=1}^{d} f_{i,j}(\boldsymbol{x}_{X_j})^{e_{i,j}}$, *where $f_{i,j}$ is a linear polynomial in $\mathbb{F}[\boldsymbol{x}_{X_j}]$, $e_{i,j} \in \mathbb{N}$, and $X_1 \sqcup \cdots \sqcup X_d$ partitions $[n]$. There is a hitting-set generator for such formulas, of size s, that runs in time $s^{O(\log^2 s)}$. The result continues to hold even if $f_{i,j}$ is a sum of univariates.*

Remarks - (1) In a recent independent work, Forbes & Shpilka [9] gave a quasi-polynomial time hitting-set generator for read-once oblivious ABPs, when the order of the variables is known. This implies a quasi-polynomial hitting-set for set-depth formulas (of *arbitrary* depth) with the *knowledge of the partition* of the variables. So, although [9] handles a more general model, the a priori knowledge of the partition prevents their algorithm from being *truly* blackbox. Another contribution of [9] is an extension of Saxena's [21] duality trick to fields of *small* characteristic - we use duality in the proofs, of Corollaries 2 & 3, for blackbox PIT of semi-diagonal and diagonal circuits. *(2)* Recently, Mulmuley [15] has shown an interesting connection between blackbox PIT of diagonal circuits (generally, *symbolic trace*), derandomization of Noether's Normalization Lemma and the generalized Riemann Hypothesis; coining the phrase *GCT Chasm.* We refer the reader to [9] for a more detailed discussion on this and a comparison between their and our work. Our techniques (eg. 'rank concentration') suggest that the GCT chasm may be bridgeable in PIT due to the special nature of polynomials, arising from circuits, that is not exploited by the existing general algebraic-geometry notions.

[1]we note that a *multilinear* set-depth-formula is a *pure circuit* (as defined by [16])

1.2 Organization

We develop some terminologies and notations in §2, which would be useful later. §3 proves the first structural property - a small shift ensures *low-block-support rank-concentration* in a product of polynomials, that have disjoint variables and only low-weight monomials. Starting with this as a base case, §4 proves the second structural property - a small shift ensures low-support rank-concentration in set-depth-Δ formulas (thus, achieving the presence of a low-support monomial). Finally, the proofs of our main results (or hitting-sets) are completed in §5.

2. THE BASICS

2.1 Polynomials

Let $\mathbb{N} := \mathbb{Z}_{\geqslant 0}$ and $[n] := \{1, \ldots, n\}$. Let R be a commutative ring. In the motivating cases R will be a field $\mathbb{F}$, which we implicitly assume to be large enough, as the required field extensions are constructible in deterministic polynomial time [1]. Further, as in blackbox PIT we are allowed to evaluate the circuit over any 'small' field extension.

Not always will we use bold notation for a vector, hopefully the context will avoid the confusion. For a vector $e \in \mathbb{Z}^n$, define $|e| := \sum_i e_i$. Let the *support* be $\mathrm{S}(e) := \{i \,|\, e_i \neq 0\}$ and the *weight* $\mathrm{s}(e)$ be its size. For an *exponent vector* $e \in \mathbb{N}^n$ and a polynomial $f \in R[\boldsymbol{x}]$, $\mathrm{Coef}(e)(f)$ denotes the coefficient of x^e in f. Define the *support of f* as $\mathrm{S}(f) := \{e \in \mathbb{N}^n \,|\, \mathrm{Coef}(e)(f) \neq 0\}$ and the *sparsity* $\mathrm{s}(f)$ be its size. The *monomial-weight of f* is $\mu(f) := \max_{e \in \mathrm{S}(f)} \mathrm{s}(e)$. Further, define the *cone of f* as $\mathcal{S}(f) := \{e' \in \mathbb{N}^n \,|\, \exists e \in \mathrm{S}(f), e' \leqslant e\}$, where the inequality is coordinate-wise, and its size as $\mathfrak{s}(f)$. For a sparse polynomial f, $\mathrm{s}(f)$ is small but $\mathfrak{s}(f)$ is usually exponentially large. For $u, v, a \in \mathbb{N}^n$ define $v! := \prod_{i \in [n]} v_i!$, $\binom{v}{u} := \prod_{i \in [n]} \binom{v_i}{u_i} = \frac{v!}{u! \cdot (v-u)!}$, and $a^{v-u} := \prod_{i \in [n]} a_i^{v_i - u_i}$. We follow the convention: For all $a < b \in \mathbb{N}$, $\binom{a}{b} = 0$ and $\binom{a}{0} = 1$.

2.2 Hadamard algebras

For a commutative ring R and $\kappa \in \mathbb{N}$, define the *Hadamard algebra* $\mathrm{H}_\kappa(R) := (R^\kappa, +, \star)$, on the free R-module R^κ, by defining: $u \star v := (u_i \cdot v_i)_{i \in [\kappa]}$, where $\cdot$ is the multiplication in R. The polynomial ring over $\mathrm{H}_\kappa(R)$ is $\mathrm{H}_\kappa(R)[\boldsymbol{x}]$, which inherits the operations $+$ and $\star$. There is an obvious isomorphism between the algebras $\mathrm{H}_\kappa(R)[\boldsymbol{x}]$ and $\mathrm{H}_\kappa(R[\boldsymbol{x}])$. *Low-support coefficient-space* - For any polynomial f over a Hadamard algebra $\mathrm{H}_\kappa(R)$, where R is a field, and $\ell \in \mathbb{N}_{>0}$, define $V_\ell(f) := \mathrm{sp}_R\{\mathrm{Coef}(e)(f) \,|\, e \in \mathbb{N}^n, \mathrm{s}(e) < \ell\} \subseteq \mathrm{H}_\kappa(R)$

(read sp_R as R-span). We call f *ℓ-concentrated over* $\mathrm{H}_\kappa(R)$ if $V_\ell(f) = \mathrm{sp}_R\{\mathrm{Coef}(e)(f) \mid e \in \mathbb{N}^n\}$.

2.3 Proof idea (for set multilinear depth-3)

Let $C(\boldsymbol{x}) = \sum_{i=1}^k \prod_{j=1}^d f_{i,j}(\boldsymbol{x}_{X_j})$, where $f_{i,j}$ is a linear polynomial in $\mathbb{F}[\boldsymbol{x}_{X_j}]$, be a set-depth-3 formula. Consider a $\Pi\Sigma$ formula

$$D(\boldsymbol{x}) := f_1(\boldsymbol{x}_{X_1}) \star \cdots \star f_d(\boldsymbol{x}_{X_d}) \quad \text{over } \mathrm{H}_k(\mathbb{F}),$$

where the i-th coordinate of $f_j(\boldsymbol{x}_{X_j})$ is $f_{i,j}(\boldsymbol{x}_{X_j})$. Note, $C(\boldsymbol{x})$ is $(1,1,\ldots,1)\cdot D(\boldsymbol{x})$, where $\cdot$ is the usual matrix product. For a subspace $V \subseteq \mathbb{F}^k$ and polynomials $D_1, D_2 \in \mathrm{H}_k(\mathbb{F})[\boldsymbol{x}]$, we say $D_1 \equiv D_2 \pmod V$ if each coefficient of $D_1 - D_2$ is in V. Somewhat wishfully, we would like the following *low-support rank-concentration* property to hold:

CONJECTURE 4 (WISHFUL!). *If $\ell > \log|C|$ then $D(\boldsymbol{x}) \equiv 0 \pmod{V_\ell(D)}$.*

If true then the coefficient of $\boldsymbol{x}^e$ in D is in the $\mathbb{F}$-span of those coefficients that correspond to low support, and hence we easily get a blackbox PIT algorithm for set-depth-3 formulas running in time $\mathrm{poly}(n^{\log|C|})$. But, Conjecture 4 is false, eg. $D = x_1 \cdots x_n$. It is here where 'shifting' is useful. The goal in this paper is to prove that after trying out a 'few' shifts of the variables, D satisfies something like Conjecture 4. Looking ahead, we conjecture (without proof) that the phenomena continue to hold in *general* depth-3 formulas.

At this point, the reader may read §3 (by skipping §2.4), which is sufficient to understand the set-depth-3 case. If the reader chooses to do so then please assume $\kappa = k$, f_i's are linear polynomials (i.e. $\delta = 1$), and D is a $\Pi\Sigma$ formula over $\mathrm{H}_k(\mathbb{F})$ in §3. §2.4 and §4 are additionally required to handle set-depth-4 and beyond.

2.4 Set-height over Hadamard algebra

Just as we have defined set-height formulas over a field $\mathbb{F}$ - meaning, the underlying constants come from $\mathbb{F}$, we can also define set-height formula in a natural way over any Hadamard algebra $\mathrm{H}_\kappa(R)$. The reason we can extend the definition to arbitrary $\mathrm{H}_\kappa(R)$ is that the defining property of set-height formulas is the existence of a partition of variables for every Π-layer (irrespective of where the constants of the formula come from). *Size* of a formula C over $\mathrm{H}_\kappa(R)$ is defined as κ times the number of gates in C.

Let C be a set-height-H formula (over $\mathbb{F}$) of depth Δ - we will count depth of C from the top, i.e. the top Σ-gate is at depth 1. If Δ is even (resp. odd) then the gates of the bottom-most Σ-layer compute sparse polynomials (resp. linear polynomials) in the variables. Let k be the maximum among the fanin of the Σ-gates of C (barring the gates of the bottom-most Σ-layer), and d the maximum among the fanin of the Π-gates in C.

Uniform fanin of Σ and Π-gates - With k and d as above, we can assume that the fanin of every Σ-gate in C (barring the gates of the bottom-most Σ-layer) is k, and fanin of every Π-gate is d. This is achieved by introducing 'dummy' gates: The 'dummy' Σ-gates introduced as children of a Π-gate compute the field constant 1, and the 'dummy' Π-gates introduced as children of a Σ-gate also compute 1 except that some of the field constants on the wires are set to zeroes. This process keeps C a set-height-H formula but might blow up the size from s to s^Δ, although it does not change k and d (the way we have defined them). Of course, formula C is not modified physically as it is presented as a blackbox. The point is, even in the blackbox setting we can treat C as a set-height-H formula with *uniform fanin* of Σ and Π-gates. We will call this uniform fanin of the Σ and Π-gates as the Σ-*fanin* and Π-*fanin*, respectively. The definition of Σ-fanin excludes the gates of the bottom-most Σ-layer - they are handled next.

Fanin bound on bottom-most Σ-gates - If Δ is even, denote the set of monomials computed by the H-th Π-layer by M; if Δ is odd then $M := \boldsymbol{x} \cup \{1\}$. The fanin of every gate of the bottom-most Σ-layer is bounded by $\lambda := |M| + 1$, the *sparsity parameter*.

Henceforth, we assume uniform Σ and Π-fanin of C (k and d respectively), keeping in mind that the fanin of every gate of the bottom-most Σ-layer is bounded by λ. k, d and λ are bounded by s. Denote this class of formulas over $\mathbb{F}$ by $\mathcal{C}_0(k, d, \lambda, \boldsymbol{x})$.

Recursive structure of set-height formulas over Hadamard algebras - Let $\mathcal{C}_h(k, d, \lambda, \boldsymbol{x})$ be the class of set-height-$(H-h)$ formulas, of depth $(\Delta - 2h)$, in the variables $\boldsymbol{x}$ with Σ-fanin k, Π-fanin d and sparsity parameter λ, over the Hadamard algebra $\mathcal{R}_h := \mathrm{H}_{k^h}(\mathbb{F})$. (Eg., to begin with $h = 0$ and the input formula $C \in \mathcal{C}_0(k, d, \lambda, \boldsymbol{x})$.) Assume that k, d and λ are less than s, which is the size of the input formula C. Let C_h be a formula in $\mathcal{C}_h(k, d, \lambda, \boldsymbol{x})$.

$$C_h(\boldsymbol{x}) = \sum_{i\in[k]} c_i \cdot \prod_{j\in[d]} f_{i,j}(\boldsymbol{x}_{X_j}), \tag{1}$$

$c_i \in \mathcal{R}_h$, $f_{i,j}(\boldsymbol{x}_{X_j})$ is a set-height-$(H-h-1)$ formula over $\mathcal{R}_h$ on the variables $\boldsymbol{x}_{X_j}$, and $X_1 \sqcup \cdots \sqcup X_d$ is the partition of $[n]$ that the first Π-layer of $C_h(\boldsymbol{x})$ respects. Let $\mathcal{R}_{h+1} := \mathrm{H}_k(\mathcal{R}_h) = \mathrm{H}_{k^{h+1}}(\mathbb{F})$. Define $f_j(\boldsymbol{x}_{X_j}) := (f_{1,j}(\boldsymbol{x}_{X_j}), \ldots, f_{k,j}(\boldsymbol{x}_{X_j}))^T \in \mathcal{R}_{h+1}[\boldsymbol{x}_{X_j}]$. Let

$$D_h(\boldsymbol{x}) := f_1(\boldsymbol{x}_{X_1}) \star \cdots \star f_d(\boldsymbol{x}_{X_d}) = \prod_{j\in[d]} f_j(\boldsymbol{x}_{X_j})$$

over $\mathcal{R}_{h+1}$, where $\star$ denotes the Hadamard product in the algebra $\mathcal{R}_{h+1}$ (extended naturally to the polynomial ring over $\mathcal{R}_{h+1}$). Evidently,

$$C_h(\boldsymbol{x}) = (c_1, \ldots, c_k) \cdot D_h(\boldsymbol{x}) = \boldsymbol{c}^T \cdot D_h(\boldsymbol{x}), \tag{2}$$

where $\cdot$ is the product for matrices over $\mathcal{R}_h[\boldsymbol{x}]$. We intend to understand the nature of the circuit $C_h(\boldsymbol{x})$ by studying the properties of the circuit $D_h(\boldsymbol{x})$ - it is here that the recursive structure reveals itself as in Lemma 5. Let $\mathcal{P}_h(h') := \{X_{h',1}, \ldots, X_{h',d}\}$ be the partition of $[n]$ that the h'-th Π-layer of C_h respects. (Recall, when the depth of C_h is even, the bottom-most Π-layer need not respect any partition - this attribute would always remain implicit in our discussions.) Define the partition $\mathcal{P}_h(h', X_j) := \{X_{h',1} \cap X_j, \ldots, X_{h',d} \cap X_j\}$ (ignore the empty sets), for every $j \in [d]$. $\mathcal{P}_h(h', X_j)$ is the partition induced on X_j by the h' layer.

LEMMA 5. *For every $j \in [d]$, $f_j(\boldsymbol{x}_{X_j})$ is a set-height-$(H-h-1)$ formula in $\mathcal{R}_{h+1}[\boldsymbol{x}_{X_j}]$ with Σ-fanin k, Π-fanin d and sparsity parameter λ, i.e. $f_j(\boldsymbol{x}_{X_j}) \in \mathcal{C}_{h+1}(k, d, \lambda, \boldsymbol{x}_{X_j})$, such that every h'-th Π-layer of $f_j(\boldsymbol{x}_{X_j})$ respects the partition $\mathcal{P}_h(h'+1, X_j)$.* (Proof in Appendix A)

2.5 Matrices

A matrix M with coefficients in ring R, and the rows (resp. columns) indexed by $\mathcal{I}$ (resp. $\mathcal{J}$) is denoted as: $M \in (\mathcal{I} \times \mathcal{J} \to R)$. When R is an integral domain, we denote the *rank* by $\mathrm{rk}_R M$. We call a matrix $M \in (\mathcal{I} \times \mathcal{J} \to R)$, $|\mathcal{I}| = |\mathcal{J}| - 1$, *strongly full* if for all $u \in \mathcal{J}$, $M_{\mathcal{I}, \mathcal{J}\setminus\{u\}}$ is invertible. For two matrices M_1, M_2 and a R-module V, we write $M_1 \equiv M_2 \pmod V$ to mean that each column of

$M_1 - M_2$ is in V. For a matrix $M \in R^{\kappa\times a}$ and an element $v \in \mathrm{H}_\kappa(R)$, $v \star M$ is the matrix obtained after taking the Hadamard product of each column with v. For two matrices $M_1 \in (\mathcal{I} \times \mathcal{J}_1 \to R), M_2 \in (\mathcal{I} \times \mathcal{J}_2 \to R)$ the *Hadamard-tensor* matrix $M_1 \circledast M_2 \in (\mathcal{I} \times (\mathcal{J}_1 \times \mathcal{J}_2) \to R)$ is defined as: Its (j_1, j_2)-th column is $(M_1)_{\mathcal{I},j_1} \star (M_2)_{\mathcal{I},j_2}$.

3. LOW-BLOCK-SUPPORT RK.-CONC.

For $i \in [\ell]$, let $f_i \in \mathrm{H}_\kappa(\mathbb{F})[\boldsymbol{x}_{X_i}]$ be a polynomial of degree at most δ, where X_i's are disjoint subsets of $[n]$. Define $\mu := \max_i\{\mu(f_i)\}$. The sparsity parameter $\lambda := \max_i\{\mathfrak{s}(f_i)\}$ of the f_i's is bounded by $(\delta+n+\mu)^{O(\mu)}$. Let $\ell := 2\lceil\log_2\kappa\rceil+1$. Consider the depth-3 ($\Pi\Sigma\Pi$) formula over $\mathrm{H}_\kappa(\mathbb{F})$,

$$D := f_1(\boldsymbol{x}_{X_1}) \star \cdots \star f_\ell(\boldsymbol{x}_{X_\ell}) \text{ in } \mathrm{H}_\kappa(\mathbb{F})[\boldsymbol{x}].$$

We *shift* it by formal variables $\boldsymbol{t}$ to get $D(\boldsymbol{x}+\boldsymbol{t}) = f_1(\boldsymbol{x}_{X_1} + \boldsymbol{t}_{X_1})\star\cdots\star f_\ell(\boldsymbol{x}_{X_\ell}+\boldsymbol{t}_{X_\ell})$ in $\mathrm{H}_\kappa(\mathbb{F}[\boldsymbol{t}])[\boldsymbol{x}]$. Wlog we can assume that, $\forall i \in [\ell]$, $f_i(\boldsymbol{t})$ *is a unit in* $\mathrm{H}_\kappa(\mathbb{F}(\boldsymbol{t}))$. This is because not being a unit only means that the vector $f_i \in \mathbb{F}(\boldsymbol{t})^\kappa$ has a zero coordinate, say at place $j \in [\kappa]$. Then the j-th coordinate of $D(\boldsymbol{t})$ is zero, and we can forget this position altogether; project the setting to the simpler algebra $\mathrm{H}_{\kappa-1}(\mathbb{F})$. We *normalize* f_i to $f'_i(\boldsymbol{x}) := f_i(\boldsymbol{t})^{-1} \star f_i(\boldsymbol{x} + \boldsymbol{t})$. Define $D'(\boldsymbol{x}) := f'_1(\boldsymbol{x}_{X_1}) \star\cdots\star f'_\ell(\boldsymbol{x}_{X_\ell})$ in $\mathrm{H}_\kappa(\mathbb{F}(\boldsymbol{t}))[\boldsymbol{x}]$.

$$D(\boldsymbol{x}+\boldsymbol{t}) \;=\; D(\boldsymbol{t}) \star D'(\boldsymbol{x}). \tag{3}$$

Any exponent $e \in \mathbb{N}^n$, possibly appearing in D', can be written uniquely as $e = \sum_{i\in[\ell]} e_i$, where $e_i \in \mathcal{S}(f_i)$, because f_i's are on disjoint set of variables. We will frequently use this identification. We define the *block-support of* e, $\mathrm{bS}(e) := \{i \in [\ell] \mid e_i \neq 0\}$, and let the *block-weight* $\mathrm{bs}(e)$ be its size. Define a relevant vector space, for $l \in \mathbb{N}_{>0}$,

$$\mathcal{V}_l(D') \;:=\; \mathrm{sp}_{\mathbb{F}(\boldsymbol{t})}\left\{\mathrm{Coef}(e)(D') \mid e \in \mathbb{N}^n, \mathrm{bs}(e) < l\right\}.$$

Ordering & Kronecker-based map - We define a *term ordering* on the monomials t^e, $e \in \mathbb{N}^n$, and their inverses. For a $w \in \mathbb{N}^n$ we denote the ordering as $t^e \preceq_w t^{e'}$, or equivalently $1/t^{e'} \preceq_w 1/t^e$, if $\sum_{i\in[n]} w_i e_i \leqslant \sum_{i\in[n]} w_i e'_i$.

For reasons of efficiency, useful later but skippable for now, we assume: $\prec_w$ keeps the monomials $\left\{\prod_{i\in[\ell]} t^{e_i} \mid \forall i \in [\ell],\right.$ $\left. e_i \in \mathcal{S}(f_i)\right\}$ distinct. If we fix such a $w \in \mathbb{N}_{>0}$ (note: it could be found in time $\lambda^{O(\ell)}$ [6]), the Kronecker-like homomorphism $\tau : t_i \mapsto y^{w_i}$ ($\forall i \in [n]$) will obviously also map the aforementioned monomials to distinct *univariate* ones.

We would like to prove something like Conjecture 4 for $D(\boldsymbol{x}+\boldsymbol{t})$. It suffices to focus on $D'(\boldsymbol{x})$ as its coefficients are all scaled-up by the same nonzero 'constant' $D(\boldsymbol{t})$. The rest of the section is devoted to proving the following theorem.

THEOREM 6 (LOW BLOCK-SUPPORT SUFFICES). $D'(\boldsymbol{x}) \equiv 0 \pmod{\mathcal{V}_\ell(D')}$. *Further, it remains true under the map* τ.

3.1 Shift-&-normalizing D

We investigate the effect of shift-&-normalizing on f_i. Write, for $i \in [\ell]$, $f_i(\boldsymbol{x}_{X_i}) =: \sum_{v_i\in \mathrm{S}(f_i)} z_{i,v_i} x^{v_i}$. (Note: $v_i \in \mathbb{N}^n$ and we will denote its j-th coordinate by $v_{i,j} \in \mathbb{N}$.) This yields, after shift-&-normalize

$$f'_i(\boldsymbol{x}) := f_i(\boldsymbol{x}+\boldsymbol{t})/f_i(\boldsymbol{t}) =: \sum_{u_i\in\mathcal{S}(f_i)} z'_{i,u_i} x^{u_i}$$

$\in \mathrm{H}_\kappa(\mathbb{F}(\boldsymbol{t}_{X_i}))[\boldsymbol{x}_{X_i}]$. The last step defines

$$z'_{i,u_i} = \mathrm{Coef}(u_i)(f'_i) = f_i(\boldsymbol{t})^{-1} \star \sum_{v_i\in \mathrm{S}(f_i)} z_{i,v_i} \binom{v_i}{u_i} t^{v_i-u_i} \tag{4}$$

for all exponent vectors $u_i \in \mathrm{S}(f'_i) \subseteq \mathcal{S}(f'_i) = \mathcal{S}(f_i)$. The constant coefficient of f'_i, $z'_{i,0} = 1$.

3.2 Transfer equation of a polynomial

Let f be one of the polynomials $f_1, \ldots, f_\ell$ over $\mathrm{H}_\kappa(\mathbb{F})$. Let $S := S(f)$ and $\mathcal{S} := \mathcal{S}(f)$. For $v \in \mathcal{S}$ define $z_v := \mathrm{Coef}(v)(f)$, and $z'_v := \mathrm{Coef}(v)(f')$. Since f is a unit, obviously, $S \neq \emptyset$ and $\mathcal{S} \neq \emptyset$. Let $Z \in ([\kappa]\times\mathcal{S} \to \mathbb{F})$ be such that: Its v-th column is the vector z_v. Note that exactly $\mathrm{s}(f)$ of these columns are nonzero. Let $Z' \in ([\kappa] \times \mathcal{S} \to \mathbb{F}(\boldsymbol{t}))$ be such that: Its u-th column is the vector z'_u. For any $\mathcal{C} \subseteq \mathcal{S}(f)$ we define a diagonal matrix $N_\mathcal{C} \in (\mathcal{C} \times \mathcal{C} \to \mathbb{F}[\boldsymbol{t}])$ as: Its u-th diagonal element is t^u. Let the *transfer matrix* (of $\Sigma\Pi$ formulas) $T \in (\mathcal{S}\times\mathcal{S} \to \mathbb{F})$ be such that: Its (v, u)-th entry is $\binom{v}{u}$. We are ready to state the *transfer equation.*

LEMMA 7 (TRANSFER EQUATION - PRIMAL). $Z' = f(\boldsymbol{t})^{-1} \star ZN_\mathcal{S}TN_\mathcal{S}^{-1}$. (Proof in Appendix B)

Denote $Z'_{[\kappa],\mathcal{C}}$ by $Z'_\mathcal{C}$, for any $\mathcal{C} \subseteq \mathcal{S}$. Note that the transfer matrix captures a transformation, from Z to Z', which is clearly invertible. Thus, T is an invertible matrix. Define $T' := (T_{\mathcal{S},\mathcal{S}})^{-1} \in (\mathcal{S} \times \mathcal{S} \to \mathbb{F})$ and $\mathcal{S}^* := \mathcal{S} \setminus \{0\}$. If $\mathcal{S}^* = \emptyset$ then it only means that $f \in \mathrm{H}_\kappa(\mathbb{F})$, and is invertible. Such an f could be dropped from D right in the beginning. From now on we assume $\mathcal{S}^* \neq \emptyset$. Recall, $z'_0 = 1$.

LEMMA 8 (TRANSFER EQUATION - MOD). *We have* $f(\boldsymbol{t})^{-1} \star Z \equiv Z'_{\mathcal{S}^*}N_{\mathcal{S}^*}T'_{\mathcal{S}^*,\mathcal{S}}N_\mathcal{S}^{-1} \pmod{z'_0}$. *Further,* $T'_{\mathcal{S}^*,\mathcal{S}}$ *is strongly full.* (Proof in Appendix B)

3.3 Transfer eqn. of D: Hadamard tensoring

For two subsets $B_1, B_2 \subset \mathbb{N}^n$, define $B_1 + B_2 := \{b_1 + b_2 \mid b_1 \in B_1, b_2 \in B_2\}$, where the sum is coordinate-wise. For $i \in [\ell]$, let $\mathcal{S}_i := \mathcal{S}(f_i)$ and $\mathcal{S}^*_i := \mathcal{S}_i \setminus \{0\}$. Define $\mathcal{S} := \sum_{i\in[\ell]} \mathcal{S}_i$ and $\mathcal{S}' := \sum_{i\in[\ell]} \mathcal{S}^*_i$. There is a natural identification between $\mathcal{S}'$ and $\times_{i\in[\ell]}\mathcal{S}^*_i$. For $i \in [\ell]$, define $Z_i \in ([\kappa]\times\mathcal{S}_i \to \mathbb{F})$ such that: Its u_i-th column is the vector $z_{i,u_i} := \mathrm{Coef}(u_i)(f_i)$. Let $Z \in ([\kappa] \times \mathcal{S} \to \mathbb{F})$ such that: Its u-th column is the vector $z_u := \mathrm{Coef}(u)(D)$. Note that $Z = \circledast_{i\in[\ell]}Z_i$. For $i \in [\ell]$, define $Z'_i \in ([\kappa] \times \mathcal{S}^*_i \to \mathbb{F})$ such that: Its v_i-th column is the vector $z'_{i,v_i} := \mathrm{Coef}(v_i)(f'_i)$. ($Z'_i$ has fewer columns than Z_i.) Let $Z' \in ([\kappa] \times \mathcal{S}' \to \mathbb{F})$ such that: Its v-th column is the vector $z'_{i,v} := \mathrm{Coef}(v)(D')$. Note that $Z' = \circledast_{i\in[\ell]}Z'_i$. For any $\mathcal{C} \subseteq \mathcal{S}$, define a diagonal matrix $N_\mathcal{C} \in (\mathcal{C} \times \mathcal{C} \to \mathbb{F}[\boldsymbol{t}])$ as: Its u-th diagonal element is t^u. For $i \in [\ell]$, define $T'_i := T'_{\mathcal{S}^*_i,\mathcal{S}_i}$. Let the *transfer matrix* (of $\Pi\Sigma\Pi$ formulas) $T' \in (\mathcal{S}' \times \mathcal{S} \to \mathbb{F})$ be $\otimes_{i\in[\ell]}T'_i$.

LEMMA 9 (TRANSFER EQUATION DEPTH-3). $D(\boldsymbol{t})^{-1}\star Z \equiv Z'N_{\mathcal{S}'}T'N_\mathcal{S}^{-1} \pmod{\mathcal{V}_\ell(D')}$. (Pf. in Appendix B)

3.4 To select columns of T'

Recall that T' has rows (resp. columns) indexed by $\mathcal{S}'$ (resp. $\mathcal{S}$) and has entries in $\mathbb{F}$. Let $\mathcal{M}$ be some $\kappa > 0$ columns that we intend to remove from T'; we call them *marked* and the others $\mathcal{S} \setminus \mathcal{M}$ are *unmarked.*

THEOREM 10 (INVERTIBLE MINOR). *There exist unmarked columns* $\mathcal{C} \subseteq \mathcal{S}$, $|\mathcal{C}| = |\mathcal{S}'|$, *s.t.* $|T'_{\mathcal{S}',\mathcal{C}}| \neq 0$. (App. B)

3.5 T' on the nullspace of Z: Finishing Thm. 6

Recall that the columns of Z are indexed by $\mathcal{S}$. Think of these *ordered* by the weight vector w, as discussed in the beginning of this section. Pick a basis $\mathcal{M}$, size at most κ, of the column vectors of Z by *starting from the largest column.* Formally, $\mathcal{M}$ gives the unique (once $\prec$ is fixed) basis such that for each u-th, $u \in \mathcal{S} \setminus \mathcal{M}$, column of Z there exist columns $u_1, \ldots, u_r \in \mathcal{M}$ spanning the u-th column, and $u \prec u_r \prec \cdots \prec u_1$. We think of the columns $\mathcal{M}$ of T' *marked*, and invoke Theorem 10 to get the $\mathcal{C} \subsetneq \mathcal{S}$. We define an $A \in (\mathcal{S} \times \mathcal{C} \to \mathbb{F})$: If a is the v-th column of A then $Z \cdot a = 0$ expresses the $\mathbb{F}$-linear dependence of z_v on $\{z_{v'} \mid v' \in \mathcal{M}, v \prec v'\}$; in particular, the *least* row where a is nonzero is the v-th, the entry being 1.

LEMMA 11 (T' ON NULLSPACE OF Z). $|T' N_{\mathcal{S}}^{-1} A| \neq 0$. *Further, the leading nonzero inverse-monomial in the determinant has the coefficient* $|T'_{\mathcal{S}',\mathcal{C}}|$. (Proof in Appendix B)

PROOF OF THEOREM 6. From the transfer equation, Lemma 9, recall $D(\boldsymbol{t})^{-1} \star Z \equiv Z' N_{\mathcal{S}'} T' N_{\mathcal{S}}^{-1} \pmod{\mathcal{V}_\ell(D')}$. Right-multiplying by A, we get

$$0 = D(\boldsymbol{t})^{-1} \star (ZA) \equiv Z' N_{\mathcal{S}'} T' N_{\mathcal{S}}^{-1} A \pmod{\mathcal{V}_\ell(D')}. \quad (5)$$

Since $T' N_{\mathcal{S}}^{-1} A$ is invertible from Lemma 11 and $N_{\mathcal{S}'}$ is obviously invertible, we get $Z' \equiv 0 \pmod{\mathcal{V}_\ell(D')}$. (Here we do use that the matrices are over $\mathbb{F}(\boldsymbol{t})$ and that $\mathcal{V}_\ell(D')$ is an $\mathbb{F}(\boldsymbol{t})$-vector space.) This implies the first part of Theorem 6, as Z' collected exactly those coefficients of D' that we a priori did not know in $\mathcal{V}_\ell(D')$. The second part of the theorem follows as: (1) τ keeps $D(\boldsymbol{t})$ a unit, and (2) τ corresponds to the correct term ordering $\preceq_w$. These two properties allow the above proof also work after applying τ. □

4. LOW-SUPPORT RK.-CONCENTRATION

We will prove that a set-height-H formula, after a 'small' shift, begins to have 'low'-support rank-concentration. The proof is by induction on the height of the formulas over Hadamard algebras. For $H > h \in \mathbb{N}$, let $\mathbf{t}_h := \{t_{H-1}, \ldots, t_{h+1}, t_h\}$ be a set of formal variables and $\mathbb{F}(\mathbf{t}_h)$ be the function field. These $\mathbf{t}_h$-variables are different from the variables $\boldsymbol{x}$ involved in the formula C. Let $\mathcal{R}'_h := \mathrm{H}_{k^h}(\mathbb{F}(\mathbf{t}_h))$ be a Hadamard algebra over $\mathbb{F}(\mathbf{t}_h)$; $k^h = \dim_{\mathbb{F}(\mathbf{t}_h)} \mathcal{R}'_h$. Further, $\mathcal{R}'_{h+1}[t_h]$ denotes the (univariate) polynomial ring over $\mathcal{R}'_{h+1}$, and $\mathcal{R}'_{h+1}(t_h)$ is the corresponding *ring of fractions*. ($\mathcal{R}'_{h+1}(t_h)$ is basically $\mathrm{H}_{k^{h+1}}(\mathbb{F}(\mathbf{t}_h))$.)

Low-support shift for $\mathcal{C}_h(k, d, \lambda, \boldsymbol{x})$ - Let τ_h be a map from $\mathbb{F}[\boldsymbol{x}]$ to $\mathbb{F}(\mathbf{t}_h)[\boldsymbol{x}]$ defined as,

$$\tau_h : x_i \mapsto x_i + \alpha_{H-1,i}\, t_{H-1}^{a_{H-1,i}} + \cdots + \alpha_{h,i}\, t_h^{a_{h,i}},$$

for $x_i \in \boldsymbol{x}$, $a_{H-1,i}, \ldots, a_{h,i} \in \mathbb{Z}^+$ and $\alpha_{H-1,i}, \ldots, \alpha_{h,i} \in \mathbb{F}$. ($\tau_h$ fixes $\mathbb{F}$, i.e. $\tau_h(c) = c$ for $c \in \mathbb{F}$.) In short, we write $\tau_h : \boldsymbol{x} \mapsto \boldsymbol{x} + \boldsymbol{\alpha}_h\, \mathbf{t}_h^{\mathbf{a}_h}$. For $\ell_h \in \mathbb{N}$, the map τ_h is called an ℓ_h-*support shift* for the class of formulas $\mathcal{C}_h(k, d, \lambda, \boldsymbol{x})$ if for every formula $C_h \in \mathcal{C}_h(k, d, \lambda, \boldsymbol{x})$, the polynomial $\tau_h(C_h(\boldsymbol{x})) = C_h(\boldsymbol{x} + \boldsymbol{\alpha}_h\, \mathbf{t}_h^{\mathbf{a}_h})$ is ℓ_h-concentrated over $\mathcal{R}'_h$.

Fix ℓ_h as - If Δ is even then for $H > h \geq 0$:

$$\ell_h := (2H\lceil H \log_2 k \rceil)^{H-h-1} \cdot 2\lceil H \log_2(k\lambda) \rceil + 1;$$

if Δ is odd then for $H \geq h \geq 0$, $\ell_h := (2H\lceil H \log_2 k \rceil)^{H-h} + 1$.

The above setting satisfies $\ell_h = (\ell_{h+1} - 1)H(\ell - 1) + 1$, where $\ell := 2\lceil H \log_2 k \rceil + 1$, for every $H - 1 > h \geq 0$ (and also for $h = H - 1$ when Δ is odd). Recall Eqn. 2 that says - for each $h \in \{0, \ldots, H-1\}$ and C_h, there exists $\boldsymbol{c} \in \mathrm{H}_k(\mathcal{R}_h)$ such that $C_h = \boldsymbol{c}^T \cdot D_h$. We prove the following theorem.

THEOREM 12 (LOW SUPPORT SUFFICES). *We can construct τ_0 such that $\tau_0 \circ D_0$ is ℓ_0-concentrated over $\mathcal{R}'_1[t_0]$, in time polynomial in $(d + n + \ell_0)^{\ell_0}$, where $n := |\boldsymbol{x}|$.*

Proof strategy ahead - The idea is to construct the map τ_h by applying induction on height $H - h$ of the class $\mathcal{C}_h(k, d, \lambda, \boldsymbol{x})$. By Eqn. 2, $C_h(\boldsymbol{x}) = \boldsymbol{c}^T \cdot (f_1(\boldsymbol{x}_{X_1}) \star \cdots \star f_d(\boldsymbol{x}_{X_d}))$. From Lemma 5, $f_j(\boldsymbol{x}_{X_j}) \in \mathcal{C}_{h+1}(k, d, \lambda, \boldsymbol{x}_{X_j})$. By definition, $\tau_{h+1} : x_i \mapsto x_i + \alpha_{H-1,i}\, t_{H-1}^{a_{H-1,i}} + \cdots + \alpha_{h+1,i}\, t_{h+1}^{a_{h+1,i}}$ is an ℓ_{h+1}-support shift for $\mathcal{C}_{h+1}(k, d, \lambda, \boldsymbol{x}_{X_j})$ for every $1 \leq j \leq d$. Here we use induction on height $H - h$: We will build the map τ_h from the inductive knowledge of τ_{h+1}. We will show that it is possible to efficiently compute $a_{h,1}, \ldots, a_{h,n} \in \mathbb{Z}^+$ and $\alpha_{h,1}, \ldots, \alpha_{h,n} \in \mathbb{F}$ such that $\tau_h : x_i \mapsto \tau_{h+1}(x_i) + \alpha_{h,i}\, t_h^{a_{h,i}}$ is an ℓ_h-support shift for $\mathcal{C}_h(k, d, \lambda, \boldsymbol{x})$.

The proof of Theorem 12. The proof proceeds by induction on height $H - h$ of the class $\mathcal{C}_h(k, d, \lambda, \boldsymbol{x})$ (in other words, *reverse* induction on h). The induction hypothesis is that τ_{h+1}, an ℓ_{h+1}-support shift for the class $\mathcal{C}_{h+1}(k, d, \lambda, \boldsymbol{x})$, can be constructed in time polynomial in $(d + n + \ell_{h+1})^{\ell_{h+1}}$, where $n := |\boldsymbol{x}|$. Overall this means, by varying $h \in [0, ..., H-1]$, we get a hitting-set of size polynomial in $\Pi_{h=0}^{H-1}(d + n + \ell_h)^{\ell_h} \leqslant (d + n + \ell_0)^{\sum_h \ell_h} < (d + n + \ell_0)^{2\ell_0}$. We discuss the base case and the inductive step in separate detail. Keep in mind that $f_j(\boldsymbol{x}_{X_j}) \in \mathcal{C}_{h+1}(k, d, \lambda, \boldsymbol{x}_{X_j})$.

4.1 Base case ($h + 1 \geqslant H - 1$)

The base case is when $H - h - 1 = 1$ or 0, i.e. $f_j(\boldsymbol{x}_{X_j})$'s are sparse polynomials or linear polynomials over $\mathcal{R}_{h+1}$, depending on whether Δ is even or odd, respectively. These two base cases have varying level of difficulty. If $H - h - 1 = 0$ then $\ell_{h+1} = \ell_H = 2$, hence taking τ_H as the identity map suffices (since $f_j(\boldsymbol{x}_{X_j})$'s are linear polynomials) as an ℓ_H-support shift for the class $\mathcal{C}_H(k, d, \lambda, \boldsymbol{x})$. If $H - h - 1 = 1$ then $f_j(\boldsymbol{x}_{X_j})$'s are sparse polynomials. We first prove,

LEMMA 13 (SPARSE POLYNOMIAL). *Let $f \in \mathrm{H}_\kappa(\mathbb{F})[\boldsymbol{x}]$ be a polynomial with degree bound δ. Let $\ell' := 1 + \min\{2\lceil \log_2(\kappa \cdot \mathrm{s}(f)) \rceil, \mu(f)\}$. We can construct a map $\sigma : x_i \mapsto x_i + t^{b_i}$, in time polynomial in $(\delta + n + \ell')^{\ell'}$, such that $\sigma(f)$ is ℓ'-concentrated over $\mathrm{H}_\kappa(\mathbb{F}(t))$.* (Proof in Appendix C)

Apply the lemma to the sparse polynomial $f_j(\boldsymbol{x}_{X_j})$, which has the sparsity parameter λ. Define $\tau_{h+1} = \tau_{H-1} : x_i \mapsto x_i + t_{H-1}^{b_i}$ (in other words, $a_{H-1,i} := b_i$). This, by Lemma 13, ensures that the concentration parameter is $2\lceil \log_2(k^{H-1} \cdot \lambda) \rceil + 1 \leqslant 2\lceil H \log_2(k\lambda) \rceil + 1 = \ell_{H-1} = \ell_{h+1}$. Finally, τ_{H-1} is an ℓ_{H-1}-support shift for the class $\mathcal{C}_{H-1}(k, d, \lambda, \boldsymbol{x})$, and it can be constructed in time polynomial in $(d + n + \ell_{H-1})^{\ell_{H-1}}$.

4.2 Induction ($h + 1$ to h)

Let $\widehat{f}_j(\boldsymbol{x}_{X_j}) := \tau_{h+1}(f_j(\boldsymbol{x}_{X_j}))$. Then,

$$\widehat{D}_h(\boldsymbol{x}) := \tau_{h+1}(D_h(\boldsymbol{x})) = \widehat{f}_1(\boldsymbol{x}_{X_1}) \star \cdots \star \widehat{f}_d(\boldsymbol{x}_{X_d}),$$

where every $\widehat{f}_j$ is ℓ_{h+1}-concentrated over $\mathcal{R}'_{h+1}$ (by induction hypothesis). Let $\mathbf{t} := \{t_{h,1}, \ldots, t_{h,n}\}$ be a set of 'fresh' formal variables. (The $\mathbf{t}$-variables would be eventually set as univariates in a variable t_h.) As in Eqn. 3, $\widehat{D}_h(\boldsymbol{x} + \mathbf{t}) = \prod_{j\in[d]} \widehat{f}_j(\boldsymbol{x}_{X_j} + \mathbf{t}_{X_j}) = \prod_{j\in[d]} \widehat{f}_j(\mathbf{t}_{X_j}) \star \widehat{f}'_j(\boldsymbol{x}_{X_j}) = \widehat{D}_h(\mathbf{t}) \star \widehat{D}'_h(\boldsymbol{x})$. We would like to show $\widehat{D}'_h(\boldsymbol{x}) \equiv 0 \pmod{\mathcal{V}_\ell(\widehat{D}'_h)}$, where $\mathcal{V}_\ell(\widehat{D}'_h) := \mathrm{sp}_{\mathbb{F}(\mathbf{t}_{h+1}, \mathbf{t})}\{\mathrm{Coef}(e)(\widehat{D}'_h) \mid e \in \mathbb{N}^n, \mathrm{bs}(e) < \ell\}$, and $\ell = 2\lceil H \log_2 k \rceil + 1$. As before (see 'key argument' in Lemma 13), it is sufficient to prove the typical case, $\widehat{D}'_{h,\ell}(\boldsymbol{x}) := \prod_{j\in[\ell]} \widehat{f}'_j(\boldsymbol{x}_{X_j}) \equiv 0 \pmod{\mathcal{V}_\ell(\widehat{D}'_{h,\ell})}$. Towards this, define the *truncated* polynomials, $\widehat{g}_j(\boldsymbol{x}_{X_j}) := \sum_{e : s(e) < \ell_{h+1}} \mathrm{Coef}(e)(\widehat{f}_j)\, \boldsymbol{x}_{X_j}^{e}$ and let the corresponding product be $\widehat{E}_h(\boldsymbol{x}) := \prod_{j\in[d]} \widehat{g}_j(\boldsymbol{x}_{X_j})$. Sparsity of $\widehat{g}_j(\boldsymbol{x}_{X_j})$ over $\mathcal{R}'_{h+1}$ is bounded by $(d^{H-h-1} + n + \ell_{h+1})^{\ell_{h+1}} =: \lambda_h$. Let, $\widehat{E}_h(\boldsymbol{x} + \mathbf{t}) = \prod_{j\in[d]} \widehat{g}_j(\boldsymbol{x}_{X_j} + \mathbf{t}_{X_j}) = \widehat{E}_h(\mathbf{t}) \star \widehat{E}'_h(\boldsymbol{x})$ and $\widehat{E}'_{h,\ell}(\boldsymbol{x}) := \prod_{j\in[\ell]} \widehat{g}'_j(\boldsymbol{x}_{X_j})$. By Theorem 6, we can find

$a_{h,1},\ldots a_{h,n} \in \mathbb{Z}^+$ in time $(d\lambda_h)^{O(\ell)} = (d+n+\ell_h)^{O(\ell_h)}$ such that by setting $t_{h,i} = \alpha_{h,i}\, t_h^{a_{h,i}}$ (any $\alpha_{h,i} \in \mathbb{F}\setminus\{0\}$ works), where t_h is a 'fresh' formal variable, we can ensure that the following is satisfied:

$$\widehat{E}'_{h,\ell}(\boldsymbol{x}) \equiv 0 \pmod{\mathcal{V}_\ell(\widehat{E}'_{h,\ell})}. \quad (6)$$

The claim is, the same setting $t_{h,i} = \alpha_{h,i}\, t_h^{a_{h,i}}$ (with carefully chosen $\alpha_{h,i}$'s) also ensures that $\widehat{D}'_{h,\ell}(\boldsymbol{x}) \equiv 0 \pmod{\mathcal{V}_\ell(\widehat{D}'_{h,\ell})}$. Consequently, $\widehat{D}'_h$ is $(\ell-1)(\ell_{h+1}-1)+1 < \ell_h$ concentrated over $\mathcal{R}'_{h+1}(t_h)$. We argue this next. Eqn. 6 implies

$$\begin{aligned}\widehat{E}_{h,\ell}(\boldsymbol{x}+\boldsymbol{\alpha}\,\mathbf{t}) &= \prod_{j\in[\ell]} \widehat{g}_j(\boldsymbol{x}_{X_j} + \boldsymbol{\alpha}_{X_j}\,\mathbf{t}_{X_j}) \quad (7)\\ &= \widehat{E}_{h,\ell}(\boldsymbol{\alpha}\,\mathbf{t}) \star \widehat{E}'_{h,\ell}(\boldsymbol{x})\\ &\equiv 0 \pmod{\mathcal{V}_\ell(\widehat{E}_{h,\ell}(\boldsymbol{x}+\boldsymbol{\alpha}\,\mathbf{t}))},\end{aligned}$$

where (reusing symbol) $\mathbf{t} := (t_h^{a_{h,1}},\ldots,t_h^{a_{h,n}})$ and $\boldsymbol{\alpha} := (\alpha_{h,1},\ldots,\alpha_{h,n})$. Let, $\widehat{D}_{h,\ell}(\boldsymbol{x}) := \prod_{j=1}^{\ell} \widehat{f}_j(\boldsymbol{x}_{X_j})$. Define, $\widehat{z}_{j,u_j} := \mathrm{Coef}(u_j)(\widehat{f}_j(\boldsymbol{x}_{X_j})) \in \mathcal{R}'_{h+1}$; $\widehat{z}'_{j,u_j} := \mathrm{Coef}(u_j)(\widehat{f}_j(\boldsymbol{x}_{X_j} + \boldsymbol{\alpha}_{X_j}\,\mathbf{t}_{X_j})) \in \mathcal{R}'_{h+1}[t_h]$; $\widetilde{z}_{j,u_j} := \mathrm{Coef}(u_j)(\widehat{g}_j(\boldsymbol{x}_{X_j})) \in \mathcal{R}'_{h+1}$; $\widetilde{z}'_{j,u_j} := \mathrm{Coef}(u_j)(\widehat{g}_j(\boldsymbol{x}_{X_j}+\boldsymbol{\alpha}_{X_j}\mathbf{t}_{X_j})) \in \mathcal{R}'_{h+1}[t_h]$. Note that $\widetilde{z}_{j,u_j} = \widehat{z}_{j,u_j}$ if $u_j \in S(\widehat{g}_j)$. Let, $\widehat{B}_j := \{u_j : \widehat{z}_{j,u_j}$ is in the $\mathbb{F}(\mathbf{t}_{h+1})$-basis of the coefficients of $\widehat{f}_j\}$ and $\widetilde{B}_j := \{u_j : \widetilde{z}_{j,u_j}$ is in the $\mathbb{F}(\mathbf{t}_{h+1})$-basis of the coefficients of $\widehat{g}_j\}$ with respect to some fixed basis that comprises coefficients of monomials of as low support as possible. Note, $\widehat{B}_j = \widetilde{B}_j =: B_j$, as $\widehat{f}_j$ is ℓ_{h+1}-concentrated over $\mathcal{R}'_{h+1}$. The crucial observation is - for any $v_j \in B_j$, $\widehat{z}'_{j,v_j}$ gets a t_h-free contribution only from the monomial x^{v_j}, thus, its basis representation looks like: $\widehat{z}'_{j,v_j} = (1+a(v_j,v_j))\cdot \widehat{z}_{j,v_j} + \sum_{u_j\in B_j\setminus\{v_j\}} a(u_j,v_j)\cdot \widehat{z}_{j,u_j}$, where a's are in $\mathbb{F}(\mathbf{t}_{h+1})[t_h]$ and t_h divides each $a(\cdot,v_j)$. Also, $\widetilde{z}'_{j,v_j} = (1+b(v_j,v_j))\cdot\widehat{z}_{j,v_j} + \sum_{u_j\in B_j\setminus\{v_j\}} b(u_j,v_j)\cdot \widehat{z}_{j,u_j}$, where b's are in $\mathbb{F}(\mathbf{t}_{h+1})[t_h]$ and t_h divides each $b(\cdot,v_j)$. Now define the matrices $\widehat{Z}_j, \widehat{Z}'_j$ and $\widetilde{Z}'_j$ as follows: $\widehat{Z}_j \in ([k^{h+1}]\times B_j \to \mathbb{F}(\mathbf{t}_{h+1}))$ with u_j-th column $\widehat{z}_{j,u_j}$; $\widehat{Z}'_j \in ([k^{h+1}]\times B_j \to \mathbb{F}(\mathbf{t}_h))$ with u_j-th column $\widehat{z}'_{j,u_j}$; and $\widetilde{Z}'_j \in ([k^{h+1}]\times B_j \to \mathbb{F}(\mathbf{t}_h))$ with u_j-th column $\widetilde{z}'_{j,u_j}$.

From the above crucial observation,

$$\widehat{Z}'_j = \widehat{Z}_j \cdot \widehat{M}' \quad\text{and}\quad \widetilde{Z}'_j = \widehat{Z}_j \cdot \widetilde{M}', \quad (8)$$

where $\widehat{M}', \widetilde{M}' \in (B_j \times B_j \to \mathbb{F}(\mathbf{t}_{h+1})[t_h])$ with rows indexed by $u_j \in B_j$ and columns by $v_j \in B_j$. The (u_j,v_j)-th entry of $\widehat{M}'$ contains $a(u_j,v_j)$ if $u_j \neq v_j$, otherwise $1+a(u_j,v_j)$ if $u_j = v_j$. Similarly, the (u_j,v_j)-th entry of $\widetilde{M}'$ contains $b(u_j,v_j)$ if $u_j \neq v_j$, otherwise $1+b(u_j,v_j)$ if $u_j=v_j$. Both $\widehat{M}'$ and $\widetilde{M}'$ are invertible over $\mathbb{F}(\mathbf{t}_{h+1})(t_h)$ as $\det(\widehat{M}') \equiv \det(\widetilde{M}') \equiv 1 \pmod{t_h}$. Therefore,

$$\widehat{Z}'_j = \widetilde{Z}'_j \cdot (\widetilde{M}'^{-1}\widehat{M}') \quad\text{and}\quad \widetilde{Z}'_j = \widehat{Z}'_j \cdot (\widetilde{M}'^{-1}\widehat{M}')^{-1}. \quad (9)$$

Observe that any coefficient of $\widehat{D}_{h,\ell}(\boldsymbol{x}+\boldsymbol{\alpha}\,\mathbf{t})$ is an $\mathbb{F}(\mathbf{t}_h)$-linear combination of the columns of $\circledast_{j\in[\ell]}\widehat{Z}_j$ (by the definition of B_j), which by Eqn. 8 is an $\mathbb{F}(\mathbf{t}_h)$-linear combination of the columns of $\circledast_{j\in[\ell]}\widehat{Z}'_j$ - this in turn is an $\mathbb{F}(\mathbf{t}_h)$-linear combination of the columns of $\circledast_{j\in[\ell]}\widetilde{Z}'_j$ (by Eqn. 9). By Eqn. 7, any $\mathbb{F}(\mathbf{t}_h)$-linear combination of the columns of $\circledast_{j\in[\ell]}\widetilde{Z}'_j$ can be expressed as an $\mathbb{F}(\mathbf{t}_h)$-linear combination of those columns u of $\circledast_{j\in[\ell]}\widetilde{Z}'_j$ for which $\mathrm{bs}(u) < \ell$, which in turn can be expressed as an $\mathbb{F}(\mathbf{t}_h)$-linear combination of those columns u of $\circledast_{j\in[\ell]}\widehat{Z}'_j$ for which $\mathrm{bs}(u) < \ell$ (by Eqn. 9 again). In other words, we have shown the following: $\widehat{D}_{h,\ell}(\boldsymbol{x}+\boldsymbol{\alpha}\,\mathbf{t}) \equiv 0 \pmod{\mathcal{V}_\ell(\widehat{D}_{h,\ell}(\boldsymbol{x}+\boldsymbol{\alpha}\,\mathbf{t}))}$. This would imply $\widehat{D}'_{h,\ell}(\boldsymbol{x}) \equiv 0 \pmod{\mathcal{V}_\ell(\widehat{D}'_{h,\ell})}$, if we choose $\boldsymbol{\alpha}$ so that the map $t_{h,i} \mapsto \alpha_{h,i}\, t_h^{a_{h,i}}$ ensures $\widehat{f}_j(\boldsymbol{\alpha}_{X_j}\,\mathbf{t}_{X_j})^{-1}$ is *well-defined* in $\mathcal{R}'_{h+1}(t_h)$. Such an $\boldsymbol{\alpha}$ can be constructed, by Lemma 14, in time polynomial in $\lambda_h = (d^{H-h-1}+n+\ell_{h+1})^{\ell_{h+1}}$. Therefore, $\tau_h : x_i \mapsto \tau_{h+1}(x_i) + \alpha_{h,i}\, t_h^{a_{h,i}}$ is such that $\tau_h(D_h(\boldsymbol{x}))$ is ℓ_h-concentrated over $\mathcal{R}'_{h+1}[t_h]$. Since $C_h(\boldsymbol{x}) = \boldsymbol{c}^T \cdot D_h(\boldsymbol{x})$, $\tau_h(C_h(\boldsymbol{x}))$ is ℓ_h-concentrated over $\mathcal{R}'_h$. This finishes the construction of τ_h, given τ_{h+1}, in time $(d+n+\ell_h)^{O(\ell_h)}$. □

LEMMA 14 (PRESERVE INVERTIBILITY). *Let $f \in \mathrm{H}_\kappa(\mathbb{F})[\boldsymbol{x}]$ be a polynomial with degree bound δ. Assume that f is ℓ'-concentrated over $\mathrm{H}_\kappa(\mathbb{F})$, and that $f^{-1} \in \mathrm{H}_\kappa(\mathbb{F}(\boldsymbol{x}))$. Then, we can contruct an $\boldsymbol{\alpha} \in \mathbb{F}^n$, in time polynomial in $\kappa(\delta+n+\ell')^{\ell'}$, such that $f(\boldsymbol{\alpha})^{-1} \in \mathrm{H}_\kappa(\mathbb{F})$.* (Proof in Appendix C.)

5. READING OFF THE HITTING-SET

5.1 Proof of Theorem 1

Suppose we are given a blackbox access to a set-height-H nonzero formula $C = C_0 \in \mathcal{C}_0(k,d,\lambda,\boldsymbol{x})$ of size s. Using Theorem 12 we can construct a map $\tau_0 : \mathbb{F}[\boldsymbol{x}] \mapsto \mathbb{F}[\boldsymbol{t}_0][\boldsymbol{x}]$ such that $\widehat{D} := \tau_0 \circ D_0$ is ℓ_0-concentrated over $\mathcal{R}'_1[t_0]$, in time $(d+n+\ell_0)^{O(\ell_0)}$. Clearly, $\widehat{D} \in \mathrm{H}_k(\mathbb{F}[\mathbf{t}_0])[\mathbf{x}]$ and $C' := \tau_0 \circ C = \mathbf{c}^T \cdot \widehat{D}$. For $X \subseteq [n]$ of size at most ℓ_0, define $\sigma_X : x_j \mapsto (x_j$ if $j \in X$, else $0)$ for all $j \in [n]$. Clearly, $\sigma_X \circ C'$ is only ℓ_0-variate, thus it has sparsity $(d^H+\ell_0)^{O(\ell_0)}$. By the assumption on $\widehat{D}$ we know that there exists such an X for which $\sigma_X \circ C' \neq 0$. Using sparse PIT methods [6] we can construct a hitting-set for C', in time $(d^H+n+\ell_0)^{O(\ell_0)} = 2^{O(\ell_0 H \log(s+\ell_0))} = \exp(O(\ell_0 H^2 \log s))$, which is time polynomial in $\exp((2H^2\log s)^{H+1})$. □

5.2 Proof of Corollary 2

We are given a blackbox access to a semi-diagonal formula $C = \sum_{i=1}^k m_i \cdot \prod_{j=1}^b f_{i,j}^{e_{i,j}}$ of size s, where m_i is a monomial, $f_{i,j}$ is a sum of univariate polynomials, and b is a constant. Using the *duality* trick (cf. [20], [9]), there exists another representation of C as $C' := \sum_{i=1}^{k'} \prod_{j=1}^n g_{i,j}(x_j)$ of size $s^{O(b)}$. Rewrite this, using the Hadamard algebra $\mathrm{H}_{k'}(\mathbb{F})$, as $C' = \boldsymbol{c}^T \cdot D$, where $D = G_1(x_1)\star\cdots\star G_n(x_n) \in \mathrm{H}_{k'}(\mathbb{F})[\boldsymbol{x}]$. The monomial-weight of each G_j is bounded by 1. By Theorem 6 (& Lemma 13) we can shift D, in time $s^{O(\log k')}$, such that it becomes $O(\log k')$-concentrated. On top of the shift, the sparse PIT gives a hitting-set in time $s^{O(\log s)}$. □

5.3 Proof of Corollary 3

Suppose we are given a blackbox access to the formula $C = \sum_{i=1}^k \prod_{j=1}^d f_{i,j}(\boldsymbol{x}_{X_j})^{e_{i,j}}$, where $f_{i,j}$ is a sum of univariate polynomials in $\mathbb{F}[\boldsymbol{x}_{X_j}]$, $e_{i,j} \in \mathbb{N}$, and $X_1 \sqcup \cdots \sqcup X_d$ partitions $[n]$. Let the formula size be s. Using duality, there exists another representation of $f_{i,j}(\boldsymbol{x}_{X_j})^{e_{i,j}}$ as $F_{i,j} := \sum_{p=1}^{k_{i,j}} \prod_{q\in X_j} g_{i,j,p,q}(x_q)$ of size $s^{O(1)}$. The monomial-weight of each $g_{i,j,p,q}$ is bounded by 1. Overall, we can represent C as $C' := \sum_{i=1}^k \prod_{j=1}^d F_{i,j}$, which is a set-depth-6 formula. The inductive proof of Theorem 12 on C' will have $H=3$ inductive steps. In the base case (dealing with sparse polynomials) we can use a better bound $\ell' = 2$ in Lemma 13, as $\mu(g_{i,j,p,q}) \leqslant 1$. This leads us to an improvement on Theorem 12 - we construct τ_0 such that $\tau_0 \circ D_0$ is $O(\log^2 s)$-concentrated over $\mathcal{R}'_1[t_0]$, in time $s^{O(\log^2 s)}$. On top of the shift, the sparse PIT gives a hitting-set for C in time $s^{O(\log^2 s)}$. □

6. CONCLUSION

We have identified a natural phenomena - low-support rank-concentration - in constant-depth formulas, that is directly useful in their blackbox PIT. In this work, we gave a proof for the interesting special case of set-depth-Δ formulas. More work is needed to prove such rank-concentration in full generality, if true. Next, it would be interesting to prove rank-concentration for multilinear depth-3 formulas.

Acknowledgements MA thanks Humboldt Forschungspreis, and we all thank Max Planck Institute for Informatics for its kind hospitality. CS and NS would also thank Hausdorff Center for Mathematics for the generous support.

7. REFERENCES

[1] L. M. Adleman and H. W. Lenstra. Finding irreducible polynomials over finite fields. In *STOC*, pages 350–355, 1986.

[2] M. Agrawal. On the Arithmetic Complexity of Euler Function. In *CSR*, pages 43–49, 2011.

[3] M. Agrawal, C. Saha, R. Saptharishi, and N. Saxena. Jacobian hits circuits: Hitting-sets, lower bounds for depth-D occur-k formulas & depth-3 transcendence degree-k circuits. In *STOC*, pages 599–614, 2012.

[4] M. Agrawal and V. Vinay. Arithmetic circuits: A chasm at depth four. In *FOCS*, pages 67–75, 2008.

[5] M. Anderson, D. van Melkebeek, and I. Volkovich. Derandomizing polynomial identity testing for multilinear constant-read formulae. In *CCC*, pages 273–282, 2011.

[6] M. Bläser, M. Hardt, R. J. Lipton, and N. K. Vishnoi. Deterministically testing sparse polynomial identities of unbounded degree. *Inform. Process. Lett.*, 109(3):187–192, 2009.

[7] X. Chen, N. Kayal, and A. Wigderson. Partial Derivatives in Arithmetic Complexity and Beyond. *FnT-TCS*, 6(1-2):1–138, 2011.

[8] M. Forbes and A. Shpilka. On identity testing of tensors, low-rank recovery and compressed sensing. In *STOC*, pages 163–172, 2012.

[9] M. Forbes and A. Shpilka. Quasipolynomial-time Identity Testing of Non-Commutative & Read-Once Oblivious ABP. *ECCC*, 2012.

[10] A. Gupta, P. Kamath, N. Kayal, and R. Saptharishi. Approaching the chasm at depth four. In *CCC*, 2013.

[11] A. Gupta, P. Kamath, N. Kayal, and R. Saptharishi. Arithmetic circuits: A chasm at depth three. *ECCC*, 2013.

[12] A. Gupta, N. Kayal, and Y. Qiao. Random Arithmetic Formulas can be Reconstructed Efficiently. In *CCC*, 2013.

[13] A. Klivans and A. Shpilka. Learning restricted models of arithmetic circuits. *ToC*, 2(1):185–206, 2006.

[14] P. Koiran. Arithmetic circuits: The chasm at depth four gets wider. *Theor. Comput. Sci.*, 448:56–65, 2012.

[15] K. Mulmuley. Geometric Complexity Theory V: Equivalence between blackbox derandomization of polynomial identity testing and derandomization of Noether's Normalization Lemma. In *FOCS*, 2012.

[16] N. Nisan and A. Wigderson. Lower bounds on arithmetic circuits via partial derivatives. *Comp. Complex.*, 6(3):217–234, 1997.

[17] R. Raz. Tensor-rank and lower bounds for arithmetic formulas. In *STOC*, pages 659–666, 2010.

[18] R. Raz and A. Shpilka. Deterministic PIT in non-commutative models. *Comp. Complex.*, 14(1):1–19, 2005.

[19] C. Saha, R. Saptharishi, and N. Saxena. The power of depth 2 circuits over algebras. In *FSTTCS*, pages 371–382, 2009.

[20] C. Saha, R. Saptharishi, and N. Saxena. A case of depth-3 identity testing, sparse factorization and duality. *Comp. Complex.*, 2012.

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

[22] N. Saxena. Progress on polynomial identity testing. *Bulletin of EATCS*, (99):49–79, 2009.

[23] N. Saxena and C. Seshadhri. Blackbox identity testing for bounded top fanin depth-3 circuits: the field doesn't matter. In *STOC*, pages 431–440, 2011.

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

[25] A. Shpilka and A. Yehudayoff. Arithmetic circuits: A survey of recent results and open questions. *FnT-TCS*, 5(3–4):207–388, 2010.

[26] S. Skyum and L. G. Valiant. A Complexity Theory Based on Boolean Algebra. *J. ACM*, 32(2):484–502, 1985.

[27] R. Smolensky. On Interpolation by Analytic Functions with Special Properties and Some Weak Lower Bounds on the Size of Circuits with Symmetric Gates. In *FOCS*, pages 628–631, 1990.

[28] L. G. Valiant. Completeness classes in algebra. In *STOC*, pages 249–261, 1979.

[29] L. G. Valiant, S. Skyum, S. Berkowitz, and C. Rackoff. Fast Parallel Computation of Polynomials Using Few Processors. *SIAM J. Comput.*, 12(4):641–644, 1983.

[30] R. Zippel. Probabilistic algorithms for sparse polynomials. In *EUROSAM*, pages 216–226, 1979.

APPENDIX

A. PROOFS OF SECTION 2

A.1 Proof of Lemma 5

PROOF. Recall that $f_j(\boldsymbol{x}_{X_j}) = (f_{1,j}(\boldsymbol{x}_{X_j}), \ldots, f_{k,j}(\boldsymbol{x}_{X_j}))^T$, where every $f_{i,j}(\boldsymbol{x}_{X_j})$ is a set-height-$(H-h-1)$ formula over $\mathcal{R}_h$. The proof is by induction on height $(H-h-1)$ of $f_j(\boldsymbol{x}_{X_j})$. *Base case ($h+1 \geqslant H-1$):* The base case is when $H-h-1 = 1$ or 0, i.e. $f_{i,j}(\boldsymbol{x}_{X_j})$'s are sparse polynomials or linear polynomials depending on whether Δ is even or odd, repectively. In this case, $f_j(\boldsymbol{x}_{X_j})$ is a set-height-$(H-h-1)$ formula over $\mathcal{R}_{h+1}$. Also, the sparsity parameter λ remains the same by its definition. Hence, $f_j(\boldsymbol{x}_{X_j}) \in \mathcal{C}_{h+1}(k, d, \lambda, \boldsymbol{x}_{X_j})$. *Inductive step ($h+2$ to $h+1$):* The formulas $f_{i,j}(\boldsymbol{x}_{X_j})$'s appear as sub-formulas of C_h at depth-3 (Eqn. 1). Therefore, the *corresponding* Π-layers of $f_{1,j}(\boldsymbol{x}_{X_j}), \ldots, f_{k,j}(\boldsymbol{x}_{X_j})$ respect the *same* partitions of $\boldsymbol{x}_{X_j}$. In particular, we can express every $f_{i,j}(\boldsymbol{x}_{X_j})$ as, $f_{i,j}(\boldsymbol{x}_{X_j}) = \sum_{p=1}^{k} b_{i,j,p} \cdot \prod_{q=1}^{d} g_{i,j,p,q}(\boldsymbol{x}_{Y_{j,q}})$, where $b_{i,j,p} \in \mathcal{R}_h$, $g_{i,j,p,q}(\boldsymbol{x}_{Y_{j,q}})$ is a set-height-$(H-h-2)$ formula over $\mathcal{R}_h$, and the first Π-layer of all $f_{i,j}(\boldsymbol{x}_{X_j})$, for $1 \leq i \leq k$, respect the same partition $\mathcal{P}_h(2, X_j)$, i.e., $Y_{j,q}$'s partition X_j as do $X_{2,q} \cap X_j$.

(Note: With j fixed, $X_{2,q} \cap X_j$ are the only relevant variable indices.) Hence,

$$f_j(\boldsymbol{x}_{X_j}) = \sum_{p=1}^{k} b_{j,p} \cdot \prod_{q=1}^{d} g_{j,p,q}(\boldsymbol{x}_{Y_{j,q}}), \qquad (10)$$

where $b_{j,p} = (b_{1,j,p}, \cdots, b_{k,j,p})^T \in \mathcal{R}_{h+1}$ and $g_{j,p,q}(\boldsymbol{x}_{Y_{j,q}}) = (g_{1,j,p,q}(\boldsymbol{x}_{Y_{j,q}}), \ldots, g_{k,j,p,q}(\boldsymbol{x}_{Y_{j,q}}))^T \in \mathcal{R}_{h+1}[\boldsymbol{x}_{Y_{j,q}}]$. In order to apply induction, we make a comparison between $f_{i,j}(\boldsymbol{x}_{X_j})$ and $g_{i,j,p,q}(\boldsymbol{x}_{Y_{j,q}})$ (and between $f_j(\boldsymbol{x}_{X_j})$ and $g_{j,p,q}(\boldsymbol{x}_{Y_{j,q}})$). Just like $f_{i,j}(\boldsymbol{x}_{X_j})$ is a set-height-$(H-h-1)$ formula over $\mathcal{R}_h$ occurring as a sub-formula at depth-3 of the formula C_h, $g_{i,j,p,q}(\boldsymbol{x}_{Y_{j,q}})$ is a set-height-$(H-h-2)$ formula over $\mathcal{R}_h$ occurring as a sub-formula at depth-5 of the formula C_h. By induction, $g_{j,p,q}(\boldsymbol{x}_{Y_{j,q}})$ is a set-height-$(H-h-2)$ formula in $\mathcal{R}_{h+1}[\boldsymbol{x}_{Y_{j,q}}]$ with Σ-fanin k, Π-fanin d and sparsity parameter λ i.e., $g_{j,p,q}(\boldsymbol{x}_{Y_{j,q}}) \in \mathcal{C}_{h+2}(k, d, \lambda, \boldsymbol{x}_{Y_{j,q}})$, such that every h'-th Π-layer of $g_{j,p,q}(\boldsymbol{x}_{Y_{j,q}})$ respects the partition $\mathcal{P}_h(h'+2, Y_{j,q})$. Since $g_{j,p,q}(\boldsymbol{x}_{Y_{j,q}})$ has only variables $\boldsymbol{x}_{Y_{j,q}}$ and $Y_{j,q} \subseteq X_j$, we can say that every h'-th Π-layer of $g_{j,p,q}(\boldsymbol{x}_{Y_{j,q}})$ respects the partition $\mathcal{P}_h(h'+2, X_j)$. The h'-th Π-layers of the $g_{j,p,q}(\boldsymbol{x}_{Y_{j,q}})$'s (for $1 \leq q \leq d$) correspond to the $(h'+1)$-th Π-layer of $f_j(\boldsymbol{x}_{X_j})$. Hence, by Eqn. 10, we infer that every h'-th Π-layer of $f_j(\boldsymbol{x}_{X_j})$ respects the partition $\mathcal{P}_h(h'+1, X_j)$. Note that the Σ-fanin, Π-fanin and the sparsity parameter remain k, d and λ, respectively. □

B. PROOFS OF SECTION 3

B.1 Proof of Lemma 7

PROOF. Consider a column $u \in \mathcal{S}$ of Z'; it is z'_u. By Eqn. 4, $z'_u = f(\boldsymbol{t})^{-1} \star \sum_{v \in \mathcal{S}} z_v \binom{v}{u} t^{v-u} = f(\boldsymbol{t})^{-1} \star \sum_{v \in \mathcal{S}} z_v \cdot t^v \cdot \binom{v}{u} \cdot t^{-u} = f(\boldsymbol{t})^{-1} \star Z \cdot (u\text{-th column of } N_{\mathcal{S}} T N_{\mathcal{S}}^{-1})$. Running over all $u \in \mathcal{S}$ gives us the result. □

B.2 Proof of Lemma 8

PROOF. Lemma 7 gives $Z'_{\mathcal{S}} = f(\boldsymbol{t})^{-1} \star Z N_{\mathcal{S}} T_{\mathcal{S},\mathcal{S}} N_{\mathcal{S}}^{-1}$. Rewrite it as, $f(\boldsymbol{t})^{-1} \star Z = Z'_{\mathcal{S}} N_{\mathcal{S}} T' N_{\mathcal{S}}^{-1}$. Going modulo the subspace $\mathrm{sp}_{\mathbb{F}(\boldsymbol{t})}\{z'_0\}$ kills the 0-th column of $Z'_{\mathcal{S}}$ and yields, $f(\boldsymbol{t})^{-1} \star Z \equiv Z'_{\mathcal{S}^*} N_{\mathcal{S}^*} T'_{\mathcal{S}^*,\mathcal{S}} N_{\mathcal{S}}^{-1} \pmod{z'_0}$. For the second part we exploit the independence of $T'_{\mathcal{S}^*,\mathcal{S}}$ from Z and the Hadamard algebra. Formally, fix a large enough $\widetilde{\kappa}$, say $|\mathcal{S}|$, and the Hadamard algebra $\mathrm{H}_{\widetilde{\kappa}}(\mathbb{F})$. Let $e \in \mathcal{S}$. Fix $\widetilde{Z}$ as: Its e-th column is 0 and the rest are linearly independent modulo 1 (note: $1 = \widetilde{z}'_0$). For this 'generic' setting we still have the equation, $\widetilde{f}(\boldsymbol{t})^{-1} \star \widetilde{Z} \equiv \widetilde{Z}'_{\mathcal{S}^*} N_{\mathcal{S}^*} T'_{\mathcal{S}^*,\mathcal{S}} N_{\mathcal{S}}^{-1} \pmod{\widetilde{z}'_0}$. Implying, $\widetilde{f}(\boldsymbol{t})^{-1} \star \widetilde{Z}_{\mathcal{S}\setminus\{e\}} \equiv \widetilde{Z}'_{\mathcal{S}^*} N_{\mathcal{S}^*} T'_{\mathcal{S}^*,\mathcal{S}\setminus\{e\}} N_{\mathcal{S}\setminus\{e\}}^{-1} \pmod{\widetilde{z}'_0}$. Since the LHS is a matrix of rank $|\mathcal{S}|-1$, deduce that $T'_{\mathcal{S}^*,\mathcal{S}\setminus\{e\}}$ is invertible. That is, $T'_{\mathcal{S}^*,\mathcal{S}}$ is strongly full. □

B.3 Proof of Lemma 9

PROOF. For $i \in [\ell]$, we can apply Lemma 8 to f_i and get,

$$f_i(\boldsymbol{t})^{-1} \star Z_i \equiv Z'_i N_{\mathcal{S}_i^*} T'_i N_{\mathcal{S}_i}^{-1} \pmod 1 \qquad (11)$$

where the 1 is the unity, the all one vector, in $\mathrm{H}_\kappa(\mathbb{F})$. Denote the u_i-th column of the matrix on the RHS, of the above congruence, by C_{i,u_i}. Consider a column $u \in \mathcal{S}$ of Z; it is z_u. Now $D(\boldsymbol{t})^{-1} \star z_u = \prod_{i \in [\ell]} f_i(\boldsymbol{t})^{-1} \star z_{i,u_i} = \prod_{i \in [\ell]} (\alpha_i + C_{i,u_i})$, for some $\alpha_i \in \mathbb{F}(\boldsymbol{t})$ by Eqn. 11. Hence, $D(\boldsymbol{t})^{-1} \star z_u \equiv \prod_{i \in [\ell]} C_{i,u_i} \pmod{\mathcal{V}_\ell(D')}$, as the product of ℓ or less C_{i,u_i} vanishes. Running over all $u \in \mathcal{S}$ gives us, $D(\boldsymbol{t})^{-1} \star Z \equiv \circledast_{i \in [\ell]} \left(Z'_i N_{\mathcal{S}_i^*} T'_i N_{\mathcal{S}_i}^{-1} \right) \equiv \left(\circledast_{i \in [\ell]} Z'_i \right) \cdot \otimes_{i \in [\ell]} \left(N_{\mathcal{S}_i^*} T'_i N_{\mathcal{S}_i}^{-1} \right) \equiv Z' \cdot N_{\mathcal{S}'} \cdot T' \cdot N_{\mathcal{S}}^{-1} \pmod{\mathcal{V}_\ell(D')}$. □

B.4 Proof of Theorem 10

PROOF. We know that $T' = \otimes_{i \in [\ell]} T'_i$, where each $T'_i \in (\mathcal{S}_i^* \times \mathcal{S}_i \to \mathbb{F})$ is strongly full (Lemma 8 for f_i). Thus, we can apply invertible row operations $E_i \in (\mathcal{S}_i^* \times \mathcal{S}_i^* \to \mathbb{F})$ such that $E_i T'_i$ has a $|\mathcal{S}_i^*|$-sized identity submatrix, and another column that has only nonzero entries. Since, from now on, we are not going to use the properties of the index sets $\mathcal{S}_i^*, \mathcal{S}_i$, we replace them by a more readable identification: Define, for $i \in [\ell]$, $n_i := |\mathcal{S}_i^*| > 0$ and identify $\mathcal{S}_i^*$ (resp. $\mathcal{S}_i$) with $U_i := [n_i]$ (resp. $W_i := [0..n_i]$). Let $U := \times_{i \in [\ell]} U_i$ and $W := \times_{i \in [\ell]} W_i$. Wlog we keep the following setting: For all $i \in [\ell]$, $(T'_i)_{U_i,U_i} = I_{n_i}$ [taking $E_i T'_i$ to be our new T'_i], and the column $(T'_i)_{U_i,0}$ is zero free. Define an *indicator* function (note: $\delta(\cdot)$ equals 1, if the boolean condition is true, else 0)

$$\begin{aligned} \varepsilon : \mathbb{N}_{>0} \times \mathbb{N} &\to \{0,1\}; \\ (u, w) &\mapsto \delta\left((w = 0) \vee (w \neq 0 \wedge w = u)\right). \end{aligned}$$

Extend it to $\mathbb{N}_{>0}^{\ell} \times \mathbb{N}^{\ell}$ by defining $\varepsilon : (u, w) \mapsto \prod_{r \in [\ell]} \varepsilon(u_r, w_r)$. Note that the (u, w)-th entry in T'_i is nonzero iff $\varepsilon(u, w) = 1$. Thus, ε *exactly indicates the non-zeroness in* T'_i. Similarly, by tensoring, the (u, w)-th entry in $T' \in (U \times W \to \mathbb{F})$ is nonzero iff $\varepsilon(u, w) = 1$. Thus, ε *exactly indicates the non-zeroness in* T'. We will build $\mathcal{C}$ incrementally, starting with $\mathcal{C} = \emptyset$. During this build up we might apply row permutations R on T'.

Consider a column u, $u \in U \subset W$, of T'. This column has exactly one nonzero entry; appearing at the row indexed by $u \in U$. Put all these unmarked columns u in $\mathcal{C}$, and collect the marked ones in $\mathcal{M}_1$. If $\mathcal{M}_1 = \emptyset$, we already have $|\mathcal{C}| = |U|$ and we are done (infact, $T'_{U,\mathcal{C}}$ is identity). So assume $|\mathcal{M}_1| =: m_1 \in [\kappa]$ and define $m_2 := \kappa - m_1 < \kappa$. Let the other marked columns be $\mathcal{M}_2 := \mathcal{M} \setminus \mathcal{M}_1$; they lie in $W \setminus U$ and are m_2 many. Consider the unmarked columns in $W \setminus U$; collect them in $\mathcal{L} := W \setminus (U \cup \mathcal{M}_2)$. We will now focus on the submatrix $T'_{\mathcal{M}_1, W \setminus U} =: T'_1$. Note that its column-indices are ℓ-tuples with at least one zero.

CLAIM 15. *There exists a row-permutation $R_1 \in \mathbb{F}^{m_1 \times m_1}$, and m_1 unmarked columns $\mathcal{C}_1 \subseteq \mathcal{L}$ such that: $(R_1 T'_1)_{\mathcal{M}_1, \mathcal{C}_1}$ is a lower-triangular $m_1 \times m_1$ matrix with w-th ($w \in \mathcal{C}_1$) diagonal entry being nonzero.*

Proof of Claim 15. We will again build $\mathcal{C}_1$ incrementally, starting from $\emptyset$. Recall that each row of T'_1 is indexed by an ℓ-tuple u in U. For $i \in [\ell]$ we denote the i-th coordinate in u by $u(i)$, and for an $I \subseteq [\ell]$, $u(I)$ denotes the ordered set $\{u(i) | i \in I\}$. For $w \in W$, define the *support* $\mathrm{S}(w) := \{i \in [\ell] \mid w(i) \neq 0\}$. We want to permute the rows so that the coordinates of the row-indices appear in a *decreasing order of frequency.* Formally, pick $R_1 \in \mathbb{F}^{m_1 \times m_1}$ to reorder the rows of T'_1 as $\mathcal{M}_1 = (u_1, \ldots, u_{m_1})$ such that:

- The ordered list $u_1(1), \ldots, u_{m_1}(1)$ has repetitions only in contiguous locations and the frequencies are non-increasing. In equation terms: The list has some r distinct elements $\alpha_1, \ldots, \alpha_r \in U_1$ with respective frequencies $i_1 \geqslant \cdots \geqslant i_r$ (summing to m_1), and they appear as $\alpha_1(i_1 \text{ times}), \ldots, \alpha_r(i_r \text{ times})$.
- The ordered list $(u_1(1), u_1(2)), \ldots, (u_{m_1}(1), u_{m_1}(2))$ has repetitions only in contiguous locations and the frequencies are non-increasing.
- The same as above holds for 3-tuples, 4-tuples,…,ℓ-tuples.

We describe an iterative process to build $\mathcal{C}_1$ one element at a time. In the i-th iteration, $i \in [m_1]$, we will add an unmarked, *unpicked* column $w_i \in \mathcal{L}$ to $\mathcal{C}_1$. The process maintains the invariant: $(R_1 T_1')_{\mathcal{M}_1,\mathcal{C}_1}$ is *lower-triangular.*

Iteration $i=1$ - The row u_1 of T_1' has exactly $2^\ell - 1$ nonzero columns. (Why? Zero-out at least one coordinate of u_1.) Since $2^\ell - 1 \geqslant \kappa > |\mathcal{M}_2|$ we can pick a column $w_1 \in \mathcal{L}$ such that $\varepsilon(u_1, w_1) \neq 0$, thus $(T_1')_{u_1,w_1} \neq 0$. Add w_1 to $\mathcal{C}_1$.

Iteration $i \geqslant 2$ - Consider the list $u_1, \ldots, u_i$. We claim that there are positions $I \subset [\ell]$, $|I| \leqslant \lceil \lg i \rceil$, such that $u_i(I)$ is not contained in any of the previous sets in the list. The proof is by *binary-search* in the list. Start with $I = \emptyset$. Pick the least $j_1 \in [\ell]$ such that $u_1(j_1), \ldots, u_i(j_1)$ are not all the same; add j_1 to I. By the ordering on u's the frequency μ_1 of $u_i(j_1)$ is at most $i/2$. If it is one then we stop with this I, otherwise we zoom-in on the 'halved' list $u_{i-\mu_1+1}, \ldots, u_i$. Again we pick the least $j_2 \in [j_1+1, \ell]$ such that $u_{i-\mu_1+1}(j_2), \ldots, u_i(j_2)$ are not all the same; add j_2 to I. This leads to a further halving of the list, and so on. Finally, we do have our positions I, $|I| \leqslant \lceil \lg i \rceil$, such that $u_i(I)$ appears for the first time in u_i. Deduce that each column w of T_1', with $I \subseteq \mathrm{S}(w) \subsetneq [\ell]$ and $w(\mathrm{S}(w)) = u_i(\mathrm{S}(w))$, has the first nonzero entry at the u_i-th row. (Consider $\varepsilon(u_j, w) = \varepsilon(u_j(\mathrm{S}(w)), w(\mathrm{S}(w))) = \varepsilon(u_j(\mathrm{S}(w)), u_i\ (\mathrm{S}(w)))$.) The number of such columns w, that are unmarked and unpicked, is at least $(2^{\ell-|I|}-1) - m_2 - (i-1) \geqslant 2^{\ell-|I|} - \kappa \geqslant 2^{\ell - \lceil \lg i \rceil} - \kappa \geqslant 2^{\ell - \lceil \lg \kappa \rceil} - \kappa = 2^{\lceil \lg \kappa \rceil + 1} - \kappa > 0$. So we can pick such a column, say, $w_i \in \mathcal{L} \setminus \mathcal{C}_1$ and add to $\mathcal{C}_1$. The square submatrix of T_1' thus far, $(R_1 T_1')_{\{u_1,\ldots,u_i\},\mathcal{C}_1}$ is lower-triangular with a nonzero diagonal.

After the iteration $i = m_1$ - The square matrix $(R_1 T_1')_{\mathcal{M}_1,\mathcal{C}_1}$ is lower-triangular with a nonzero diagonal. $\square$

Since R_1 permutes the rows of T_1', its action can be lifted to the rows of T'; call this action R. Append $\mathcal{C}_1$ to the current $\mathcal{C}$ (making its size $|U|$). Define $\overline{\mathcal{M}}_1 := U \setminus \mathcal{M}_1$ and $\overline{\mathcal{C}}_1 := \mathcal{C} \setminus \mathcal{C}_1$. The square matrix $(RT')_{U,\mathcal{C}}$ looks like,

$$\left[\begin{array}{c|c} (RT')_{\overline{\mathcal{M}}_1,\overline{\mathcal{C}}_1} & (RT')_{\overline{\mathcal{M}}_1,\mathcal{C}_1} \\ \hline (RT')_{\mathcal{M}_1,\overline{\mathcal{C}}_1} & (RT')_{\mathcal{M}_1,\mathcal{C}_1} \end{array}\right]$$

$$= \left[\begin{array}{c|c} I_{\overline{\mathcal{M}}_1,\overline{\mathcal{C}}_1} & (RT')_{\overline{\mathcal{M}}_1,\mathcal{C}_1} \\ \hline 0_{\mathcal{M}_1,\overline{\mathcal{C}}_1} & (R_1 T_1')_{\mathcal{M}_1,\mathcal{C}_1} \end{array}\right].$$

Its determinant equals $|(R_1 T_1')_{\mathcal{M}_1,\mathcal{C}_1}| \neq 0 \Rightarrow |T'_{U,\mathcal{C}}| \neq 0$. $\square$

B.5 Proof of Lemma 11

PROOF. Let a be the v-th column of A. Let $a' \in \mathbb{F}^{|\mathcal{M}|}$ be the vector having the entries of a appearing at the rows $\mathcal{M}$. Consider $(T' N_{\mathcal{S}}^{-1}) \cdot a$. By the property of a we can write, $(T' N_{\mathcal{S}}^{-1}) a = (T' N_{\mathcal{S}}^{-1})_{\mathcal{S}',v} + (T' N_{\mathcal{S}}^{-1})_{\mathcal{S}',\mathcal{M}} \cdot a' = T'_{\mathcal{S}',v} \cdot t^{-v} + (T' N_{\mathcal{S}}^{-1})_{\mathcal{S}',\mathcal{M}} \cdot a'$. Thus, the v-th column of A has the leading monomial t^{-v} which 'contributes' the vector $T'_{\mathcal{S}',v}$. Going over the columns a, running $v \in \mathcal{C}$, by the column-linearity of determinant and the multiplicativity of the inverse-monomial ordering, we deduce that the largest possible (inverse-monomial) term in the expression $|T' N_{\mathcal{S}}^{-1} A|$ is: $|T'_{\mathcal{S}',\mathcal{C}}| \cdot t^{-\sum_{v \in \mathcal{C}} v}$. This is nonzero, by the property of $\mathcal{C}$, thus it is *indeed* the leading term, i.e. $|T' N_{\mathcal{S}}^{-1} A| \neq 0$. $\square$

C. PROOFS OF SECTION 4

C.1 Proof of Lemma 13

PROOF. If $2\lceil \log_2(\kappa \cdot \mathrm{s}(f)) \rceil \geqslant \mu(f)$ then $\ell' = 1 + \mu(f)$. In this case trivially, for any shift σ, $\sigma(f)$ is ℓ'-concentrated over $\mathrm{H}_\kappa(\mathbb{F}(t))$. So, from now on we assume $2\lceil \log_2(\kappa \cdot \mathrm{s}(f)) \rceil < \mu(f)$, thus $\ell' = 1 + 2\lceil \log_2(\kappa \cdot \mathrm{s}(f)) \rceil$. Define $\mathcal{R} := \mathrm{H}_{\mathrm{s}(f)}(\mathrm{H}_\kappa(\mathbb{F}))$. Let $f =: \sum_{e \in \mathrm{S}(f)} z_e x^e$. Define a column vector $D \in (\mathrm{S}(f) \times [1] \to \mathrm{H}_\kappa(\mathbb{F}[\boldsymbol{x}]))$ with e-th entry being $z_e x^e$; D can be seen as a polynomial over $\mathcal{R}$. Rewrite D as a product of univariate polynomials over $\mathcal{R}$ as: $D(\boldsymbol{x}) = g_1(x_1) \star \cdots \star g_n(x_n)$. Clearly, each g_i has degree, hence sparsity, bounded by δ, and can be seen as an element in $\mathrm{H}_{\kappa \cdot \mathrm{s}(f)}(\mathbb{F})[x_i]$.

For any $X \subseteq [n]$ of size ℓ', define $D_X(\boldsymbol{x}) := \prod_{i \in X} g_i(x_i)$. Recalling Theorem 6 we can construct a shift σ for D_X, such that $\sigma \circ D_X$ is ℓ'-concentrated, in time polynomial in $(\delta + n + \ell')^{\ell'}$. Using induction on the number of variables, it is easy to see that if $\sigma \circ D_X$ is ℓ'-concentrated ($\forall X \in \binom{[n]}{\ell'}$) then so is $\sigma \circ D$. The *key argument* is: Since the constant coefficient in each g_i' (i.e. shift-&-normalized g_i) is one, deduce that the coefficient of any term in D' (i.e. shift-&-normalized D) of block-weight $\leqslant \ell'$ is produced by the product of some $\leqslant \ell'$ g_i''s, so this case is covered by some $X \in \binom{[n]}{\ell'}$. Also, deduce that the coefficient of any term in D' of block-weight $> \ell'$ can be inductively written down as a linear combination of $\{\mathrm{Coef}(e)(D') \mid e \in \mathbb{N}^n, \mathrm{s}(e) < \ell'\}$. Finally, $\sigma \circ D$ inherits this concentration property from D'.

Recall $f = 1^T \cdot D$, where 1 is the unity in $\mathcal{R} = \mathrm{H}_{\mathrm{s}(f)}(\mathrm{H}_\kappa(\mathbb{F}))$. Thus, from the ℓ'-concentration of $\sigma \circ D$ (over $\mathcal{R}$), we can deduce the ℓ'-concentration of $\sigma \circ f$ (over $\mathrm{H}_\kappa(\mathbb{F})$). This completes the construction of σ. $\square$

C.2 Proof of Lemma 14

PROOF. View f as a vector with κ coordinates; each entry is in $\mathbb{F}[\boldsymbol{x}] \setminus \{0\}$. Call the i-th entry f_i. Clearly, f_i has variables (resp. degree) at most n (resp. δ). Also, by the concentration property there exists $e_i \in \mathbb{N}^n$, with $\mathrm{s}(e_i) \leqslant \ell'$, such that $\mathrm{Coef}(e_i)(f_i) \neq 0$. For $X \subseteq [n]$ of size at most ℓ', define $\sigma_X : x_j \mapsto (x_j \text{ if } j \in X, \text{ else } 0)$ for all $j \in [n]$. Clearly, $\sigma_X \circ f_i$ is only ℓ' variate, thus it has sparsity $(\delta + \ell')^{O(\ell')}$. By the assumption on f_i we know that $X_i := \mathrm{S}(e_i)$ is of size at most ℓ', and $\sigma_{X_i} \circ f_i \neq 0$. Using standard sparse PIT methods [6], we can construct a hitting-set for $\sigma_{X_i} \circ f_i$ in time $(\delta + \ell')^{O(\ell')}$. Varying over all subsets $X \subseteq [n]$ of size at most ℓ', we get a hitting-set for f_i in time $(\delta + n + \ell')^{O(\ell')}$. For convenience, denote this hitting-set as a set of evaluation-maps $\{\sigma_{i,1}, \ldots, \sigma_{i,r}\}$; each map is from $\boldsymbol{x}$ to $\mathbb{F}$ and we write $\sigma_{i,j} \circ f_i$ to mean $f_i(\sigma_{i,j}(\boldsymbol{x}))$. Overall we are ensured the existence of a j, for a given i, such that $\sigma_{i,j} \circ f_i \neq 0$. We will now show how to combine all these into a single map.

Pick distinct κr elements $\beta_{1,1}, \ldots, \beta_{\kappa,r} \in \mathbb{F}$. Consider the univariate polynomial $g(u) := \prod_{i \in [\kappa], j \in [r]} (u - \beta_{i,j})$. Define $g_{i,j}(u) := g(u)/(u - \beta_{i,j})$, for all i, j. Consider an evaluation map from $\mathbb{F}[\boldsymbol{x}]$ to $\mathbb{F}[u, v]$ - $\sigma := v \cdot \sum_{i \in [\kappa], j \in [r]} g_{i,j}(u) \cdot \sigma_{i,j}$. We claim that, for all $i \in [\kappa]$, $\sigma \circ f_i \neq 0$. To see this, note that there is some $j \in [r]$ for which $\sigma_{i,j} \circ f_i \neq 0$. Further, let f_i' be a homogeneous part of f_i, say of degree δ_i, such that $\sigma_{i,j} \circ f_i' \neq 0$. Consider the partial evaluation $(\sigma \circ f_i)(\beta_{i,j}, v) = f_i(v \cdot g_{i,j}(\beta_{i,j}) \cdot \sigma_{i,j}(\boldsymbol{x}))$. Here the coefficient of the monomial v^{δ_i} is $g_{i,j}(\beta_{i,j})^{\delta_i} \cdot (\sigma_{i,j} \circ f_i') \neq 0$. Consequently, $\sigma \circ f_i \neq 0$.

Thus, for all $i \in [\kappa]$, $\sigma \circ f_i$ is a nonzero bivariate polynomial in $\mathbb{F}[u, v]$. Since its degree remains bounded by $\delta \cdot \kappa r$, we can again apply [6] to replace u, v by a hitting-set. Finally, we hit an $\boldsymbol{\alpha} \in \mathbb{F}^n$, in time polynomial in $\kappa(\delta + n + \ell')^{\ell'}$, such that for all $i \in [\kappa]$, $f_i(\boldsymbol{\alpha}) \neq 0$. This finishes the proof. $\square$