Electronic Colloquium on Computational Complexity, Revision 2 of Report No. 150 (2015)

Reconstruction of $\Sigma\Pi\Sigma(2)$ Circuits over Reals

Gaurav Sinha *

Abstract

Reconstruction of arithmertic circuits has been heavily studied in the past few years and has connections to proving lower bounds and deterministic identity testing. In this paper we present a polynomial time randomized algorithm for reconstructing $\Sigma\Pi\Sigma(2)$ circuits over $\mathbb{R}$, i.e. depth-3 circuits with fan-in 2 at the top addition gate and having real coefficients.

The algorithm needs only a blackbox query access to the polynomial $f \in \mathbb{R}[x_1, \ldots, x_n]$ of degree d, computable by a $\Sigma\Pi\Sigma(2)$ circuit C. In addition, we assume that the *"simple rank"* of this polynomial (essential number of variables after removing the gcd of the two multiplication gates) is bigger than a fixed constant r. Our algorithm runs in time $poly(n, d)$ and returns an equivalent $\Sigma\Pi\Sigma(2)$ circuit(with high probability).

The problem of reconstructing $\Sigma\Pi\Sigma(2)$ circuits over finite fields was first proposed by Shpilka [Shp07]. The generalization to $\Sigma\Pi\Sigma(k)$ circuits, $k = O(1)$ (over finite fields) was addressed by Karnin and Shpilka in [KS09a]. The techniques in these previous involve iterating over all objects of certain kinds over the ambient field and thus the running time depends on the size of the field $\mathbb{F}$. Their reconstruction algorithm uses lower bounds on the lengths of Linear Locally Decodable Codes with 2 queries. In our settings, such ideas immediately pose a problem and we need new ideas to handle the case of real fields.

Our main techniques are based on the use of Quantitative Syslvester Gallai Theorems from the work of Barak et.al. [BDWY11] to find a small collection of *"nice"* subspaces to iterate over. The heart of our paper lies in subtle applications of the Quantitative Sylvester Gallai theorems to prove why projections w.r.t. the *"nice"* subspaces can be "glued". We also use Brill's Equations from [GKZ94] to construct a small set of candidate linear forms (containing linear forms from both gates). Another important technique which comes very handy is the polynomial time randomized algorithm for factoring multivariate polynomials given by Kaltofen [KT90].

1 Introduction

The last few years have seen significant progress towards interesting problems dealing with arithmetic circuits. Some of these problems include Deterministic Polynomial Identity Testing, Reconstruction of Circuits and recently Lower Bounds for Arithmetic Circuits. There has also been work connecting these three different aspects. In this paper we will primarily be concerned with the reconstruction problem. Even though it's connections to Identity Testing and Lower Bounds are very exciting, the problem in itself has drawn a lot of attention because of elegant techniques and connections to learning. The strongest version of the problem requires that for any $f \in \mathbb{F}[x_1, \ldots, x_n]$ with blackbox access given one wants to construct (roughly) most succint representation i.e. the smallest possible arithmetic circuit computing the polynomial. This general problem appears to be very hard. Most of the work done has dealt with some special type of polynomials i.e.

*Department of Mathematics, California Institute of Technology, Pasadena CA 91106, USA. email : gsinha@caltech.edu

ISSN 1433-8092

the ones which exhibit constant depth circuits with alternating addition and multiplication gates. Our result adds to this by looking at polynomials computed by circuits of this type (alternating addition/multiplication gates but of depth 3). Our circuits will have variables at the leaves, operations $(+, \times)$ at the gates and scalars at the edges. We also assume that the top gate has only two children and the *"simple rank"* of this polynomial (essential number of variables after removing the gcd of the two multiplication gates) is bigger than a constant. The bottom most layer has addition gates and so computes linear forms, the middle layer then multiplies these linear forms together and the top layer adds two such products. Later in Remark 1.2 we discuss that we may assume the linear forms computed at bottom level to be homogeneous and the in-degree of all gates at middle level to be the same ($=$ degree of f). Therefore these circuits compute polynomials with the following form :

$$f(x_1, \ldots, x_n) = G(x_1, \ldots, x_n)(T_0(x_1, \ldots, x_n) + T_1(x_1, \ldots, x_n))$$

where $T_i(x_1, \ldots, x_n) = \prod_{j=1}^{M} l_{ij}$ and $G(x_1, \ldots, x_n) = \prod_{j=1}^{d-M} G_j$ with the l_{ij}'s and G_j's being linear forms for $i \in \{0, 1\}$. Also assume $gcd(T_0, T_1) = 1$. Our condition about the essential number of variables (after removing gcd from the multiplication gates) is called *"simple rank"* of the polynomial and is defined as dimension of the space

$$sp\{l_{ij} : i \in \{0, 1\}, j \in \{1, \ldots, M\}\}$$

When the underlying field is $\mathbb{R}$ (i.e. the field of real numbers) we give an efficient randomized algorithm for reconstructing the circuit representation of such polynomials. Formally our main theorem reads :

Theorem 1.1 ($\Sigma\Pi\Sigma_{\mathbb{R}}(2)$ Reconstruction Theorem)**.** *Let $f = G(T_0 + T_1) \in \mathbb{R}[x_1, \ldots, x_n]$ be any degree d, $n-$ variate polynomial (to which we have blackbox access) which can be computed by a depth 3 circuit with top fan-in 2 (i.e. a $\Sigma\Pi\Sigma(2)$ circuit) with G, T_i being products of homogeneous linear forms and $gcd(T_0, T_1) = 1$. Assume $span\{l : l \mid T_0T_1\}$ is bigger than a fixed constant r (defined later). We give a randomized algorithm which runs in time $poly(n, d)$ and computes the cicuit for f with high probability.*

As per our knowledge this is the first algorithm that efficiently reconstructs such circuits (over the reals). Over finite fields, the same problem has been considered by [Shp07] and our method takes inspiration from their work. They also generalized this finite field version to circuits with arbitrary (but constant) top fan-in in [KS09a]. However we need many new tools and techniques as their methods don't generalize at a lot of crucial steps. For eg:

- They iterate through linear forms in a finite field which we unfortunately cannot do.
- They use lower bounds for Locally Decodable Codes given in [DS07] which again does not work in our setup.

We resolve these issues by

- Constructing candidate linear forms by solving simultaneous polynomial equations obtained from Brill's Equations (Chapter 4, [GKZ94]).
- Using quantitative versions of the Sylvester Gallai Theorems given in [BDWY11]. This new method enables us to construct $nice$ subspaces, take projections onto them and glue the projections back to recover the cicuit representation.

1.1 Previous Work and Connections

Efficient Reconstruction algorithms are known for some concrete class of circuits. We list some here:

- Depth-2 $\Sigma\Pi$ circuits (sparse polynomials) in [KS01]
- Read-once arithmetic formulas in [SV09]
- Non-commutative ABP's [AMS08]
- $\Sigma\Pi\Sigma(2)$ circuits over finite fields in [Shp07], extended to $\Sigma\Pi\Sigma(k)$ circuits (over finite fields) with $k = O(1)$ in [KS09a].
- Random Multilinear Formular in [GKL11]
- Depth 4 ($\Sigma\Pi\Sigma\Pi$) multilinear circuits with top fan-in 2 in [GKL12]
- Random Arithmetic Formulas in [GKQ14]

All of the above work introduced new ideas and techniques and have been greatly appreciated.

It's straightforward to observe that a polynomial time deterministic reconstruction algorithm for a circuit class C also implies a polynomial time Deterministic Identity Testing algorithm for the same class. From the works [Agr05] and [HS80] it has been established that blackbox Identity Testing for certain circuit classes imply superpolynomial circuit lower bounds for an explicit polynomial. Hence the general problem of deterministic reconstruction cannot be easier than proving superpolynomial lower bounds. So one might first try and relax the requirements and demand a randomized algorithm. Another motivation to consider the probabilistic version comes from Learning Theory. A fundamental question called the *exact learning problem using membership queries* asks the following : **Given oracle access to a Boolean function, compute a small description for it.** This problem has attracted a lot of attention in the last few decades. For eg. in [Kha92][GGM86] and [KV94] a negative result stating that a class of boolean circuits containing the trapdoor functions or pseudo-random functions has no efficient learning algorithms. Among positive works [SS96], [BBB$^+$00], [KS06] show that when f has a small circuit (inside some restricted class) exact learning from membership queries is possible. Our problem is a close cousin as we are looking for exact learning algorithms for algebraic functions. Because of this connection with learning theory it makes sense to also allow randomized algorithms for reconstruction.

1.2 Depth 3 Arithmetic Circuits

We will use the definitions from [KS09b]. Let C be an arithmetic circuit with coefficients in the field $\mathbb{F}$. We say C is a $\Sigma\Pi\Sigma(k)$ circuit if it computes an expression of the form.

$$C(\bar{x}) = \sum_{i\in[k]} \prod_{j\in[d]} l_{i,j}(\bar{x})$$

$l_{i,j}(\bar{x})$ are linear forms of the type $l_{i,j}(\bar{x}) = \sum_{s\in[n]} a_s x_s$ where $(a_1, \ldots, a_n) \in \mathbb{F}^n$ and $(x_1, \ldots, x_n)$ is an $n-$ tuple of indeterminates. For convenience we denote the multiplication gates in C as

$$T_i = \prod_{j\in[d]} l_{i,j}(\bar{x})$$

k is the top fanin of our circuit C and d is the fanin of each multiplication gate T_i. With these definitions we will say that our circuit is of type $\Sigma\Pi\Sigma_{\mathbb{F}}(k,d,n)$. When most parameters are understood we will just call it a $\Sigma\Pi\Sigma(k)$ circuit.

Remark Note that we are cosidering homogeneous circuits. There are two basic assumptions:

1. $l_{i,j}$'s have no constant term i.e. they are linear forms.
2. Fanin of each T_i is equal to d.

If these are not satisfied we can homogenize our circuit by considering $Z^d(C(\frac{X_1}{Z},\ldots,\frac{X_n}{Z}))$. Now both the conditions will be taken care of by reconstructing this new homogenized circuit.

Definition 1.2 (Minimal Circuit)**.** We say that the circuit C is minimal if no strict non empty subsets of the $\Pi\Sigma$ polynomials $\{T_1,\ldots,T_k\}$ sums to zero.

Definition 1.3 (Simple Circuit and Simplification)**.** A circuit C is called Simple if the gcd of the $\Pi\Sigma$ polynomials $gcd(T_1,\ldots,T_k)$ is equal to 1 (i.e. is a unit). The simplification of a $\Sigma\Pi\Sigma(k)$ circuit C denoted as $Sim(C)$ is the $\Sigma\Pi\Sigma(k)$ circuit obtained by dividing each term by the gcd of all terms i.e.

$$Sim(C) \stackrel{def}{=} \sum_{i\in[k]} \frac{T_i}{gcd(T_1,\ldots,T_k)}$$

Definition 1.4 (Rank of a Circuit)**.** Identifying each linear form $l(\bar{x}) = \sum_{s\in[n]} a_s x_s$ with the vector $(a_1,\ldots,a_n) \in \mathbb{F}^n$, we define the rank of C to be the dimension of the vector space spanned by the set $\{l_{i,j}|i\in[k], j\in[d]\}$.

Definition 1.5 (Simple Rank of a Circuit)**.** For a $\Sigma\Pi\Sigma(k)$ circuit C we define the *Simple Rank* of C as the rank of the circuit $Sim(C)$.

Before we go further into the paper and explain our algorithm we state some results about uniqueness of these circuits. In a nutshell for a $\Sigma\Pi\Sigma_{\mathbb{R}}(2,d,n)$ circuit C, if one assumes that the *Simple rank* of C is bigger than a constant ($c_{\mathbb{R}}(4)$: defined later) then the circuit is essentially unique.

1.3 Uniqueness of Representation

Shpilka et. al. showed the uniqueness of circuit representation in [Shp07] using rank bounds for Polynomial Identity Testing. The bound they used were from the work of Dvir et. al. in [DS07]. It essentialy states that the rank of a simple, minimal $\Sigma\Pi\Sigma(k)$ circuit ($d\geq 2, k\geq 3$) which computes the identically zero polynomial is $\leq 2^{O(k^2)}\log^{k-2} d$. For circuits over reals improved rank bounds were given by Kayal et.al. in [KS09b].
In a series of following work the rank bounds for identically zero $\Sigma\Pi\Sigma(k)$ circuits got further improved. The best known bounds over real fields were given by Saxena et. al. in [SS10]. We rewrite Theorem 1.5 in [SS10] here for completion.

Theorem 1.6 (Theorem 1.5 in [SS10])**.** *Let C be a $\Sigma\Pi\Sigma(k,d,n)$ circuit over field $\mathbb{R}$ that is simple, minimal and zero. Then, $rk(C) < 3k^2$.*

Let $c_{\mathbb{R}}(k) = 3k^2$. This gives us the following version of Corollary 7, Section 2.1 in [Shp07].

Theorem 1.7 ([Shp07]). *Let $f(\bar{x}) \in \mathbb{R}[x]$ be a polynomial which exhibits a $\Sigma\Pi\Sigma(2)$ circuit*

$$C = G(A + B)$$

$A = \prod_{j\in[M]} A_j, B = \prod_{j\in[M]} B_j, G = \prod_{i\in[d-M]} G_i$, where $A_i, B_j, G_k \in Lin_{\mathbb{R}}[\bar{x}]$. $gcd(A, B) = 1$, and $Sim(C) = A + B$ has rank $\geq c_{\mathbb{R}}(4) + 1$ then the representation is unique. That is if:

$$f = G(A + B) = \tilde{G}(\tilde{A} + \tilde{B})$$

where $A, B, \tilde{A}, \tilde{B}$ are $\Pi\Sigma$ polynomials over $\mathbb{R}$ and $gcd(\tilde{A}, \tilde{B}) = 1$ then we have $G = \tilde{G}$ and $(A, B) = (\tilde{A}, \tilde{B})$ or $(\tilde{B}, \tilde{A})$ (upto scalar multiplication).

Proof. Let $g = gcd(G, \tilde{G})$ and let $G = gG_1, \tilde{G} = g\tilde{G_1}$. Then $gcd(G_1, \tilde{G_1}) = 1$ and we get

$$G_1A + G_1B - \tilde{G_1}\tilde{A} - \tilde{G_1}\tilde{B} = 0$$

This is a simple $\Sigma\Pi\Sigma(4)$ circuit with $rank$ bigger than $c_{\mathbb{R}}(4) + 1$ and is identically 0 so it must be not minimal. Considering the various cases one can easily prove the required equality. □

1.4 Outline of the Algorithm

The broad structure of our algorithm is similar to that of Shpilka in [Shp07] however our techniques are different. We first restrict the blackbox inputs to a low ($O(1)$) dimensional random subspace of $\mathbb{R}^n$ and interpolate this restricted polynomial. Next we try to recover the $\Sigma\Pi\Sigma(2)$ structure of this restricted polynomial and finally lift it back to $\mathbb{R}^n$. The random subspace and unique $\Sigma\Pi\Sigma(2)$ structure will ensure that the lifting is unique. Similar to [Shp07] we try to answer the following questions. However our answers (algorithms) are different from theirs

1. For a $\Sigma\Pi\Sigma(2)$ polynomial f, can one compute a small set of linear forms which contains all factors from both gates?

2. Let V_0 be a co-dimension k subspace($k = O(1)$) and $V_1, \ldots, V_t$ be co-dimension 1 subspaces of a linear space V. Given circuits C_i ($i \in \{0, \ldots, t\}$) computing $f|_{V_i}$(restriction of f to V_i) can we reconstruct from them a single circuit C for $f|_V$?

3. Given co-dimension 1 subspaces $V \subset U$ and circuits $f|_V$ when is the $\Sigma\Pi\Sigma(2)$ circuit representations of lifts of $f|_V$ to $f|_U$ unique?

Our first question is easily solved using Brill's equations (See Chapter 4 [GKZ94]). These provide a set of polynomials whose simultaneous solutions completely characterize coefficients of complex $\Pi\Sigma$ polynomials. A linear form l divides one of the gates of $f \Rightarrow f$ is a $\Pi\Sigma$ polynomial modulo l. When this is applied into Brill's equation we recover possible l's which obviously include linear factors of gates. The extra linear forms we get are not too many and also have some special structure. We call this set $\mathcal{C}$ of linear forms as Candidate linear forms and non-deterministically guess from this set. It should be noted that we do all this when our polynomial is over $O(1)$ variables i.e. the restricted polynomial mentioned in the discussion before these questions.

We deal with the second question while trying to reconstruct the $\Sigma\Pi\Sigma(2)$ representation of the interpolated polynomial $f|_V$, where V is the random low dimensional subspace. There are Easy Cases and a Hard Case.

- For the Easy Cases our algorithm tries to reconstruct one of the multiplication gates of $f|_V$ by first looking at it's restriction to a special co-dimension 1 subspace V_1. If $f = A + B$ with A, B being $\Pi\Sigma$ polynomials, the projection of one of the gates (say A) with respect to V_1 will be 0 and the other (say B) will remain unchanged giving us B and therefore both gates by factoring $f|_V - B$.

- In the Hard Case we will first need V_0, a co-dimension k (where $k = O(1)$) subspace and then iteratively select co-dimension 1 subspaces $V_1, \ldots, V_t$. For some gate (say B), all pairs (V_0, V_i) ($i \in [t]$) will reconstruct some linear factors of B. This process will either completely reconstruct B or we will fall into the Easy Case. Once B is known we can factor $f|_V - B$ to get A.

The restrictions that we compute always factor into product of linear forms and can be easily computed since we know $f|_V$ explicitly. They can then be factorized into product of linear forms using the factorization algorithms from [KT90]. It is the choice of the subspaces $V_0, V_1, \ldots, V_t$ where our algorithm differs from that in [Shp07] significantly. Our algorithm selects V_0 and iteratively selects the V_i's ($i \in [t]$) such that (V_0, V_i) have certain *"nice"* properties which help us recover the gates in $f|_V$. The existence of subspaces with *"nice"* properties is guaranteed by Quantitative Sylvester Gallai Theorems given in [BDWY11]. To use the theorems we had to develop more machinery that has been explained later.

The third question comes up when we want to lift our solution from the random subspace V to the original space. This is done in steps. We first consider random spaces U such that V has co-dimension 1 inside them. Now we reconstruct the circuits for $f|_V$ and $f|_U$. The $\Sigma\Pi\Sigma(2)$ circuits for $f|_V$ and $f|_U$ are unique since the simple ranks are high enough (because U, V are random subspaces of high enough dimension) implying that the circuit for $f|_V$ lifts to a unique circuit for $f|_U$. When this is done for multiple U's we can find the gates exactly.

1.5 Organization of the Paper

Here is how our paper is organised:

- In Subsection 2.1 we go through some definitions and notations we will follow. It is important since some definitions are new and used very frequently.

- Subsection 2.2 talks about removing some non-degenracy from our input by making a random transformation on the variables.

- We talk about some results in incidence geometry in Subsection 2.3, most importantly a Quantitative version of the Sylvester Gallai Theorem given in [BDWY11]. The subsection ends with a corollary we prove to be used later.

- To begin reconstruction we need a constructive description of the variety of $\Pi\Sigma$ (product of linear forms) polynomials. This is given by Brill's Equation explained in Subsection 2.4.

- A method to reconstruct product of linear forms from their projections onto subspaces is described in Subsection 2.5.

- Section 3 is the core of the paper. It solves the reconstruction problem assuming that the number of variables is a large enough constant.

- Section 4 deals with the most general case i.e. the rank being arbitrary (but bigger than a fixed constant). We then use random projections to convert to the constant rank case in Section 3. Then we describe in Subsection 4.2 how to glue different (polynomially many) such reconstructions together and achieve the complete reconstruction.

2 Preliminaries

2.1 Notation

$[n]$ denotes the set $\{1, 2, \ldots, n\}$. Throughout the paper we will work over the field $\mathbb{R}$. Let V be a finite dimensional real vector space and $S \subset V$, $sp(S)$ will denote the linear span of elements of S. $dim(S)$ is the dimension of the subspace $sp(S)$. If $S = \{s_1, \ldots, s_k\} \subset V$ is a set of linearly independent vectors then $fl(S)$ denotes the affine subspace generated by points in S (also called a $(k-1)-flat$ or just $flat$ when dimension is understood). In particular:

$$fl(S) = \{\sum_{i=1}^{k} \lambda_i s_i : \lambda_i \in \mathbb{R}, \sum_{i=1}^{k} \lambda_i = 1\}$$

Let $W \subset V$ be a subspace, then we can extend basis and get another subspace W' (called the complement of W) such that $W \oplus W' = V$. Note that the complement need not be unique. Corresponding to each such decomposition of V we may define orthogonal projections $\pi_W, \pi_{W'}$ onto W, W' respectively. Let $v = w + w' \in V, w \in W, w' \in W'$:

$$\pi_W(v) = w, \pi_{W'}(v) = w'$$

$(\bar{x})$ will be used for the tuple $(x_1, \ldots, x_n)$.

$$Lin_{\mathbb{R}}[\bar{x}] = \{a_1x_1 + \ldots + a_nx_n : a_i \in \mathbb{R}\} \subset \mathbb{R}[\bar{x}]$$

is the vector space of all linear forms over the variables $(x_1, \ldots, x_n)$. For a linear form $l \in Lin_{\mathbb{R}}[\bar{x}]$ and a polynomial $f \in \mathbb{R}[x]$ we write $l \mid f$ if l divides f and $l \nmid f$ if it does not. We say $l^d \mid\mid f$ if $l^d \mid f$ but $l^{d+1} \nmid f$.

$$\Pi\Sigma^d_{\mathbb{R}}[\bar{x}] = \{l_1(\bar{x}) \ldots l_d(\bar{x}) : l_i \in Lin_{\mathbb{R}}[\bar{x}]\} \subset \mathbb{R}[\bar{x}]$$

is the set of degree d homogeneous polynomials which can be written as product of linear forms. This collection for all possible d is called the set

$$\Pi\Sigma_{\mathbb{R}}[\bar{x}] = \bigcup_{d \in \mathbb{N}} \Pi\Sigma^d_{\mathbb{R}}[\bar{x}]$$

also called $\Pi\Sigma$ polynomials for convenience. Let $f(\bar{x}) \in \mathbb{R}[x]$ then $Lin(f) \in \Pi\Sigma_{\mathbb{R}}[\bar{x}]$ denotes the product of all linear factors of $f(\bar{x})$. Let $\mathcal{L}(f)$ denote the set of all linear factors of f. For any set of polynomials $S \subset \mathbb{C}[\bar{x}]$, we denote by $\mathbb{V}(S)$, the set of all complex simultaneous solutions of polynomials in S (this set is called the variety of S), i.e.

$$\mathbb{V}(S) = \{a \in \mathbb{C} : \text{ for all } f \in S, f(a) = 0\}$$

Let $\mathcal{B} = \{b_1, \ldots, b_n\}$ be an ordered basis for $V = Lin_{\mathbb{R}}[\bar{x}]$. We define maps $\phi_{\mathcal{B}} : V \setminus \{0\} \to V$ as

$$\phi_{\mathcal{B}}(a_1b_1 + \ldots + a_nb_n) = \frac{1}{a_k}(a_1b_1 + \ldots + a_nb_n)$$

where k is such that $a_i = 0$ for all $i < k$ and $a_k \neq 0$.

A non-zero linear form l is called <u>normal</u> with respect to $\mathcal{B}$ if $l \in \Phi_{\mathcal{B}}(V)$ i.e. the first non-zero coefficient is 1. A polynomial $P \in \Pi\Sigma_{\mathbb{R}}[\bar{x}]$ is normal w.r.t. $\mathcal{B}$ if it is a product of normal linear forms. For two polynomials $P_1, P_2 \in \Pi\Sigma_{\mathbb{R}}[\bar{x}]$ we define :

$$gcd_{\mathcal{B}}(P_1, P_2) = P \in \Pi\Sigma_{\mathbb{R}}[\bar{x}], P \text{ normal w.r.t. } \mathcal{B} \text{ such that } P \mid P_1, P \mid P_2$$

When a basis is not mentioned we assume that the above definitions are with respect to the standard basis. We can represent any linear form in $Lin_{\mathbb{R}}[\bar{x}]$ as a point in the vector space $\mathbb{R}^n$ and vice versa. To be precise we define the cannonical map $\Gamma : Lin_{\mathbb{R}}[\bar{x}] \to \mathbb{R}^n$ as

$$\Gamma(a_1x_1 + \ldots + a_nx_n) = (a_1, \ldots, a_n)$$

Γ is a linear isomorphism of vector spaces $Lin_{\mathbb{R}}[\bar{x}]$ and $\mathbb{R}^n$. Because of this isomorphism we will interchange between points and linear forms whenever we can. We choose to represent the linear form $a(\bar{x}) = a_1x_1 + \ldots + a_nx_n$ as the point $a = (a_1, \ldots, a_n)$.

LI will be the abbreviation for Linearly Independent and **LD** will be the abbreviation for Linearly Dependent.

Definition 2.1 (Standard Linear Form)**.** A non zero vector v is called $standard$ with respect to basis $\mathcal{B} = \{b_1, \ldots, b_n\}$ if the coefficient of b_1 in v is 1. When a basis is not mentioned we assume we're talking about the standard basis. (Equivalently for linear forms the coefficient of x_1 is 1). A $\Pi\Sigma$ polynomial will be called $standard$ if it is a product of standard linear forms.

We close this section with a lemma telling us when can we replace the span of some vectors with the affine span or flat. We've used this several times in the paper.

Lemma 2.2. *Let $l, l_1, \ldots, l_t \in Lin_{\mathbb{R}}[\bar{x}]$ be* standard *linear forms w.r.t. some basis $\mathcal{B} = \{b_1, \ldots, b_n\}$ such that $l \in sp(\{l_1, \ldots, l_t\})$ then*

$$l \in fl(\{l_1, \ldots, l_t\})$$

Proof. Since $l \in sp(\{l_1, \ldots, l_t\})$, we know that $l = \sum\limits_{i \in [t]} \alpha_i l_i$ for some scalars $\alpha_i \in \mathbb{R}$. All linear forms are $standard$ w.r.t. $\mathcal{B} \Rightarrow$ comparing the coefficients of b_1 we get that $\sum\limits_{i \in [t]} \alpha_i = 1$ and therefore $l \in fl(\{l_1, \ldots, l_t\})$. $\square$

Let $T \subset \mathbb{R}^n$, By a scaling of T we mean a set where all vectors get scaled (possibly by different scalars).

2.2 Random Linear Transformations

This section will prove some results about linear independence and non-degeneracy under random transformations on $\mathbb{R}^r$. This will be required to make our input non-degenerate. From here onwards we fix a natural number $N \in \mathbb{N}$ and assume $0 < k < r$. Let $T \subset \mathbb{R}^r$ be a finite set with $dim(T) = r$. Next we consider two $r \times r$ matrices Ω, Λ. Entries $\Omega_{i,j}, \Lambda_{i,j}$ are picked independently from the uniform distribution on $[N]$. For any basis $\mathcal{B}$ of $\mathbb{R}^r$ and vector $v \in \mathbb{R}^r$, let $[v]_{\mathcal{B}}$ denote the co-ordinate vector of v in the basis $\mathcal{B}$. If $\mathcal{B} = \{b_1, \ldots, b_r\}$ then $[v]_{\mathcal{B}}^i$ denotes the i-th co-ordinate in $[v]_{\mathcal{B}}$. Let $\mathcal{S} = \{e_1, \ldots, e_r\}$ be the standard basis of $\mathbb{R}^r$. Let $E_j = sp(\{e_1, \ldots, e_j\})$ and $E'_j = sp(\{e_{j+1}, \ldots, e_r\})$, then $\mathbb{R}^r = E_j \oplus E'_j$. Let $\pi_{W_{E_j}}$ be the orthogonal projection onto E_j. For any matrix M, we denote the matrix of it's co-factors by $co(M)$. We consider the following events :

- $\mathcal{E}_0 = \{\Omega \text{ is not invertible }\}$
- $\mathcal{E}_1 = \{\exists t (\neq 0) \in T : \pi_{W_{E_1}}(\Omega(t)) = 0\}$
- $\mathcal{E}_2 = \{\exists \{t_1, \ldots, t_r\} \text{ LI vectors in } T : \{\Omega(t_1), \ldots, \Omega(t_r)\} \text{ is LD }\}$

- $\mathcal{E}_3 = \{\exists\{t_1, \ldots, t_r\}$ LI vectors in $T : \{\Omega(t_1), \ldots, \Omega(t_k), \Lambda\Omega(t_{k+1}), \ldots, \Lambda\Omega(t_r)\}$ is LD $\}$
- When t_i, Λ, Ω are clear we define the matrix $M = [M_1 \ldots M_r]$ with columns M_i given as :

$$M_i = \begin{cases} [\Omega(t_i)]_{\mathcal{S}} : i \leq k \\ [\Lambda\Omega(t_i)]_{\mathcal{S}} : i > k \end{cases}$$

M corresponds to the linear map

$$e_i \mapsto \Omega(t_i) \text{ for } i \leq k \text{ and } e_i \mapsto \Lambda\Omega(t_i) \text{ for } i > k$$

$\mathcal{E}_4 = \{\{\exists\{t_1, \ldots, t_r\}$ LI vectors in T and $t \in T \setminus sp(\{t_1, \ldots, t_k\}) : [co(M)[\Omega(t)]_{\mathcal{S}}]^{k+1}_{\mathcal{S}} = 0\}$

- $\mathcal{E}_5 = \mathcal{E}_4 \mid \mathcal{E}_3^c$

Next we show that the probability of all of the above events is small. Before doing that let's explain the events. This will give an intuition to why the events have low probabilities.

- $\mathcal{E}_0$ is the event where Ω is not-invertible. Random Transformations should be invertible.
- $\mathcal{E}_1$ is the event where there is a non-zero $t \in T$ such that the projection to the first co-ordinate (w.r.t. $\mathcal{S}$) of Ω applied on t is 0. We don't expect this for a random linear transformation. Random Transformation on a non-zero vector should give a non-zero coefficient of e_1.
- $\mathcal{E}_2$ is the event such that Ω takes a basis to a LD set i.e. Ω is not invertible (random linear operators are invertible).
- $\mathcal{E}_3$ is the event such that for some basis applying Ω to the first k vectors and $\Lambda\Omega$ to the last $n - k$ vectors gives a LD set. So this operation is not-invertible. For ranrom maps this should not be the case.
- $\mathcal{E}_4$ is the event that there is some basis $\{t_1, \ldots, t_r\}$ and t outside $sp(t_1, \ldots, t_k)$ such that the $(k+1)^{th}$ co-ordinate of $co(M)[\Omega(t)]_{\mathcal{S}}$ w.r.t the standard basis is 0. If M were invertible, clearly the set $\mathcal{B} = \{\Omega(t_1), \ldots, \Omega(t_k), \Lambda\Omega(t_{k+1}), \ldots, \Lambda\Omega(t_r)\}$ would be a basis and $co(M)$ will be a scalar multiple of M^{-1}. So we are asking if the $(k+1)^{th}$ co-ordinate of $\Omega(t)$ in the basis $\mathcal{B}$ is 0. For random Ω, Λ we would expect M to be invertible and this co-ordinate to be non-zero.

Now let's formally prove everything. We will repeatedly use the popular Schawrtz-Zippel Lemma which the reader can find in [Sax09].

Claim 2.3. $Pr[\mathcal{E}_1] \leq \frac{|T|}{N^r}$

Proof. Fix a non-zero $t = \begin{pmatrix} a_1 \\ \cdot \\ \cdot \\ a_r \end{pmatrix}$ with $a_i \in \mathbb{R}$ and let $\Omega = (\Omega_{i,j}), 1 \leq i, j \leq r$. Then the first co-ordinate of $\Omega(t)$ is $\Omega_{1,1}a_1 + \Omega_{1,2}a_2 + \ldots + \Omega_{1,r}a_r$. Since $t \neq 0$, not all a_i are 0 and this is therefore not an identically zero polynomial in $(\Omega_{1,1}, \ldots, \Omega_{1,r})$. Therefore by Schwartz-Zippel lemma $Pr[[\Omega(t)]^1_{\mathcal{S}} = 0] \leq \frac{1}{N^r}$. Using a union bound inside T we get $Pr[\exists t (\neq 0) \in T : [\Omega(t)]^1_{\mathcal{S}} = 0] \leq \frac{|T|}{N^r}$. $\square$

Claim 2.4. $Pr[\mathcal{E}_2] \leq \frac{r}{N^{r^2}}$

Proof. Clearly $\mathcal{E}_2 \subseteq \mathcal{E}_0$ and so $Pr[\mathcal{E}_2] \leq Pr[\mathcal{E}_0]$. $\mathcal{E}_0$ corresponds to the polynomial equation $det(\Omega) = 0$. $det(\Omega)$ is a degree r polynomial in r^2 variables and is also not identically zero, so using Schwartz-Zippel lemma we get $Pr[\mathcal{E}_2] \leq Pr[\mathcal{E}_0] \leq \frac{r}{N^{r^2}}$. ☐

Claim 2.5. $Pr[\mathcal{E}_3] \leq \binom{|T|}{r} \frac{2r}{N^{2r^2}}$

Proof. Fix an LI set $t_1, \ldots, t_r$. The set $\{\Omega(t_1), \ldots, \Omega(t_k), \Lambda\Omega(t_{k+1}), \ldots \Lambda\Omega(t_r)\}$ is LD iff the $r \times r$ matrix M formed by writing these vectors (in basis $\mathcal{S}$) as columns (described in part 2.2 above) has determinant 0. M has entries polynomial (of degree ≤ 2) in $\Omega_{i,j}$ and $\Lambda_{i,j}$ and so $det(M)$ is a polynomial in $\Omega_{i,j}, \Lambda_{i,j}$ of degree $\leq 2r$. For $\Omega = \Lambda = I$ (identity matrix) this matrix just becomes the matrix formed by the basis $\{t_1, \ldots, t_r\}$ which has non-zero determinant and so $det(M)$ is not the identically zero polynomial. By Schwartz-Zippel lemma $Pr[det(M) = 0] \leq \frac{2r}{N^{r^2} N^{r^2}} = \frac{2r}{N^{2r^2}}$. Now we vary the LI set $\{t_1, \ldots, t_r\}$, there are $\leq \binom{|T|}{r}$ such sets and so by a union bound $Pr[\mathcal{E}_3] \leq \binom{|T|}{r} \frac{2r}{N^{2r^2}}$. ☐

Claim 2.6. $Pr[\mathcal{E}_4] \leq \binom{|T|}{r+1} \frac{2r-1}{N^{2r^2}}$

Proof. Fix an LI set $t_1, \ldots, t_r$ and a vector $t \notin sp(\{t_1, \ldots, t_k\})$. Let $t = \sum_{i=1}^{r} a_i t_i$. Since $t \notin sp(\{t_1 \ldots, t_k\})$, $a_s \neq 0$ for some $s \in \{k+1, \ldots, r\}$. Let $\mathcal{B} = \{\Omega(t_1), \ldots, \Omega(t_k), \Lambda\Omega(t_{k+1}), \ldots \Lambda\Omega(t_r)\}$. Let M be the matrix whose columns are from $\mathcal{B}$ (Construction has been explained in part 2.2 above). We know that the co-factors of a matrix are polynomials of degree $\leq r-1$ in the matrix elements. In our matrix M all entries are polynomials of degree ≤ 2 in $\Omega_{i,j}, \Lambda_{i,j}$, so all entries of $co(M)$ are polynomials of degree $\leq 2r-2$ in $\Omega_{i,j}, \Lambda_{i,j}$. Thus $[co(M)[\Omega(t)]_{\mathcal{S}}]_{\mathcal{S}}^{k+1} = \sum_{i=1}^{r} co(M)_{k+1,i} [\Omega(t)]_{\mathcal{S}}^{i}$ is a polynomial of degree $\leq 2r-1$. This polynomial is not identically zero. Define Ω to be the matrix (w.r.t. basis $\mathcal{S}$) of the linear map $\Omega(t_i) = e_i$ and Λ to be the matrix (w.r.t. basis $\mathcal{S}$) of the map

$$\Lambda = \begin{cases} \Lambda(e_i) = e_i : i \notin \{s, k+1\} \\ \Lambda(e_s) = e_{k+1} \\ \Lambda(e_{k+1}) = e_s \end{cases}$$

With these values the set $\mathcal{B}$ becomes $\{e_1, \ldots, e_k, e_s, e_{k+2}, \ldots, e_{s-1}, e_{k+1}, e_{s+1}, \ldots, e_r\}$. If one now looks at M i.e. the matrix formed using entries of $\mathcal{B}$ as columns it's just the permutation matrix that flips e_s and e_{k+1}. This matrix is the inverse of itself and so has determinant $= \pm 1$, thus $co(M) = \pm M^{-1} = \pm M$.

Therefore $co(M)[\Omega(t)]_{\mathcal{S}} = \pm M \begin{pmatrix} a_1 \\ . \\ . \\ a_r \end{pmatrix} = \pm \begin{pmatrix} a_1 \\ . \\ a_k \\ a_s \\ a_{k+2} \\ . \\ a_{s-1} \\ a_{k+1} \\ .a_{s+1} \\ . \\ a_r \end{pmatrix}$. Since $a_s \neq 0$, we get $[co(M)[\Omega(t)]_{\mathcal{S}}]_{\mathcal{S}}^{k+1} \neq 0$.

So the polynomial is not identically zero and we can use Schwartz-Zippel Lemma to say that $Pr[[co(M)[\Omega(t)]_{\mathcal{S}}]_{\mathcal{S}}^{k+1} = 0] \leq \frac{2r-1}{N^{r^2} N^{r^2}} = \frac{2r-1}{N^{2r^2}}$. Now we vary $\{t_1, \ldots, t_r, t\}$ inside T and use union bound to show $Pr[\mathcal{E}_4] \leq \binom{|T|}{r+1} \frac{2r-1}{N^{2r^2}}$. ☐

Even though this is just basic probability we include the following:

Claim 2.7. $Pr[\mathcal{E}_5] \leq \binom{|T|}{r} \frac{2r-1}{N^{2r^2} - \binom{|T|}{r} 2r}$

Proof. $Pr[\mathcal{E}_5] = Pr[\mathcal{E}_4 \mid \mathcal{E}_3^c] = \frac{Pr[\mathcal{E}_4 \cap \mathcal{E}_3^c]}{Pr[\mathcal{E}_3^c]} \leq \frac{Pr[\mathcal{E}_4]}{Pr[\mathcal{E}_3^c]} \leq \binom{|T|}{r+1} \frac{\frac{2r-1}{N^{2r^2}}}{1 - \binom{|T|}{r} \frac{2r}{N^{2r^2}}} = \binom{|T|}{r+1} \frac{2r-1}{N^{2r^2} - \binom{|T|}{r} 2r}$ $\square$

In our application of the above $r = O(1), |T| = poly(d), N = 2^d$ and so all probabilities are very small as d grows. So we will assume that none of the above events occur. By union bound that too will have small probability and so with very high probability $\mathcal{E}_0, \mathcal{E}_1, \mathcal{E}_2, \mathcal{E}_3, \mathcal{E}_4, \mathcal{E}_5$ do not occur.

2.3 Tools from Incidence Geometry

Later in the paper we will use the quantitative version of Sylvester-Gallai Theorem from [BDWY11]. In this subsection we do preparation for the same. Our main application will also involve a corollary we prove towards the end of this subsection.

Definition 2.8 ([BDWY11])**.** Let S be a set of n distinct points in complex space $\mathbb{C}^r$. A $k - flat$ is elementary if its intersection with S has exactly $k+1$ points.

Definition 2.9 ([BDWY11])**.** Let S be a set of n distinct points in $\mathbb{C}^r$. S is called a $\delta - SG_k$ configuration if for every independent $s_1, \ldots, s_k \in S$ there are atleast δn points $t \in S$ such that either $t \in fl(\{s_1, \ldots, s_k\})$ or the $k-$flat $fl(\{s_1, \ldots, s_k, t\})$ contains a point in $S \setminus \{s_1, \ldots, s_k, t\}$.

Theorem 2.10 ([BDWY11])**.** *Let S be a $\delta - SG_k$ configuration then $dim(S) \leq \frac{2^{C^k}}{\delta^2}$. Where $C > 1$ is a universal constant.*

This bound on the dimension of S was further improved by Dvir et. al. in [DSW12]. The latest version now states

Theorem 2.11 ([DSW12])**.** *Let S be a $\delta - SG_k$ configuration then $dim(S) \leq C_k = \frac{C^k}{\delta}$. Where $C > 1$ is a universal constant.*

Corollary 2.12. *Let $dim(S) > C_k$ then S is not a $\delta - SG_k$ configuration i.e. there exists a set of independent points $\{s_1, \ldots, s_k\}$ and $\geq (1-\delta)n$ points t such that $fl(\{s_1, \ldots, s_k, t\})$ is an elementary $k - flat$. That is:*

- $t \notin fl(\{s_1, \ldots, s_k\})$
- $fl(\{s_1, \ldots, s_k, t\}) \cap S = \{s_1, \ldots, s_k, t\}$.

Right now we set δ to be a constant $< 0.5, C_k = \frac{C^k}{\delta}$. Note that $C_i < C_{i+1}$. Using the above theorem we prove the following lemma which will be useful to us later

Lemma 2.13 (Bichromatic semi-ordinary line)**.** *Let X and Y be disjoint finite sets in $\mathbb{C}^r$ satisfying the following conditions.*

1. *$dim(Y) > C_4$.*
2. *$|Y| \leq c|X|$ with $c < \frac{1-\delta}{\delta}$.*

Then there exists a line l such that $|l \cap Y| = 1$ and $|l \cap X| \geq 1$

Proof. We consider two cases:
Case 1 : $c|X| \geq |Y| \geq |X|$
Since $dim(Y) > C_1$, using the corollary above for $S = X \cup Y, k = 1$ we can get a point $s_1 \in X \cup Y$ for which there exist $(1-\delta)(|X|+|Y|)$ points t in $X \cup Y$ such that $t \notin fl\{s_1\}$ and $fl\{s_1, t\}$ is elementary. If $s_1 \in X$ then $(1-\delta)(|X|+|Y|) - |X| \geq (1-2\delta)|X| > 0$ of these flats intersect Y and thus we get such a line l. If $s_1 \in Y$ then $(1-\delta)(|X|+|Y|) - |Y| \geq ((1-\delta)(\frac{1}{c}+1)-1)|Y| > 0$ of these flats intersect X giving us the required line l with $|l \cap X| = 1$ and $|l \cap Y| = 1$.

Case 2: $|Y| \leq |X|$
Now choose a subset $X_1 \subseteq X$ such that $|X_1| = |Y|$. Now using the same argument as above for $S = X_1 \cup Y$ there is a point $s_1 \in X_1 \cup Y$ such that $(1-\delta)(|X_1|+|Y|) = 2(1-\delta)|Y| = 2(1-\delta)|X_1|$ flats through it are elementary in $X_1 \cup Y$. If $s_1 \in Y$ $(1-2\delta)|Y| > 0$ of these flats intersect X_1. If $s_1 \in X_1$, $(1-2\delta)|X_1| > 0$ of these flats intersect Y. In both these above possibilities the flat intersects Y and X_1 in exactly one point each. But it may contain more points from $X \setminus X_1$ so we can find a line l such that $|l \cap Y| = 1$ and $|l \cap X| \geq 1$. □

2.4 Characterizing $\Pi\Sigma$ polynomials (Brill's Equations)

In this section we will explicitly compute a set of polynomials whose common solutions characterize the coefficients of all homogeneous $\Pi\Sigma_{\mathbb{C}}[x_1, \dots, x_r]$ polynomials of degree d. A clean mathematical construction is given by Brill's Equations given in Chapter 4, [GKZ94]. However we still need to calculate the time complexity. But before that we define some operations on polynomials and calculate the time taken by the operation along with the size of the output. Note that all polynomials are over the field of complex numbers $\mathbb{C}$ and all computations are also done for the complex polynomial rings.

Let $\bar{x} = (x_1, \dots, x_r)$ and $\bar{y} = (y_1, \dots, y_r)$ be variables. For any homogeneous polynomial $f(\bar{x})$ of degree d, define

$$f_{\bar{x}^k}(\bar{x}, \bar{y}) = \frac{(d-k)!}{d!}(\sum_i x_i \frac{\partial}{\partial y_i})^k f(\bar{y})$$

Expanding $(\sum_i x_i \frac{\partial}{\partial y_i})^k$ as a polynomial of differentials takes $O((r+k)^r)$ time and has the same order of terms in it. $f(\bar{y})$ has $O((r+k)^r)$ terms. Taking partial derivatives of each term takes constant time and therefore overall computing $(\sum_i x_i \frac{\partial}{\partial y_i})^k f(\bar{y})$ takes $O((r+k)^{2r})$ time. Also the expression obtained will have atmost $O((r+k)^{2r})$ terms. Computing the external factor takes $poly(d)$ time and so for an arbitrary $f(\bar{x})$ computing all $f_{\bar{x}^k}(\bar{x}, \bar{y})$ for $0 \leq k \leq d$ takes $poly((r+d)^r)$ time and has $poly((r+d)^r)$ terms in it. From Section E., Chapter 4 in [GKZ94] we also know that $f_{\bar{x}^k}(\bar{x}, \bar{y})$ is a bihomogeneous form of degree k in $\bar{x}$ and degree $d-k$ in $\bar{y}$. It is called the k^{th} polar of f.

Next we define an $\odot$ opeartion between homogeneous forms. Let $f(\bar{x})$ and $g(\bar{x})$ be homogeneous polynomials of degrees d, define

$$(f \odot g)(\bar{x}, \bar{y}) = \frac{1}{d+1}\sum_{k=0}^{d}(-1)^k \binom{d}{k} f_{\bar{y}^k}(\bar{y}, \bar{x}) g_{\bar{x}^k}(\bar{x}, \bar{y})$$

From the discussion above we know that computing $f_{\bar{y}^k}(\bar{y}, \bar{x}) g_{\bar{x}^k}(\bar{x}, \bar{y})$ takes $poly((r+d)^r)$ time and it is obvious that this product has $poly((r+d)^r)$ terms. Rest of the operations take $poly(d)$ time and therefore computing $(f \odot g)(\bar{x}, \bar{y})$ takes $poly((r+d)^r)$ time and has $poly((r+d)^r)$ terms. From the discussion before we may also easily conclude that the degrees of $\bar{x}, \bar{y}$ in $(f \odot g)(\bar{x}, \bar{y})$ are $poly(d)$. The form $(f \odot g)$ is called the vertical(Young) product of f and g. See Section G., Chapter 4 in [GKZ94].

Next for $k \in \{0, \dots, d\}$ and $\bar{z} = (z_1, \dots, z_r)$ consider homogeneous forms:

$$e_k = \binom{d}{k} f_{\bar{x}^k}(\bar{x}, \bar{z}) f(\bar{z})^{k-1}$$

Following arguments from above, it's straightforward to see that computing e_k takes $poly((r+d)^r)$ time and has $poly((r+d)^r)$ terms. Each e_k is a homogeneous form in $\bar{x}, \bar{z}$ and f. It has degree k in $\bar{x}$, degree $k(d-1)$ in z, and k in coefficients of f. See Section H. of Chapter 4 in [GKZ94]. Let's define the following function of $\bar{x}$ with parameters f, z

$$P_{f,z}(\bar{x}) = (-1)^d d \sum_{i_1+2i_2+\ldots+ri_r=d} (-1)^{(i_1+\ldots+i_r)} \frac{(i_1+\ldots+i_r-1)!}{i_1!\ldots i_r!} e_1^{i_1} \ldots e_r^{i_r}$$

Note that $\{(i_1, \dots, i_r) : i_1 + 2i_2 + \ldots + ri_r = d\} \subseteq \{(i_1, \dots, i_r) : i_1 + i_2 + \ldots + i_r \leq d\}$ and therefore the number of additions in the above summand is $O(poly(r+d)^r)$. For every fixed $(i_1, \dots, i_r)$ computing the coefficient $\frac{(i_1+\ldots+i_r-1)!}{i_1!\ldots i_r!}$ takes $O(poly((r+d)^r))$ time using multinomial coefficients. Each e_k takes $poly((r+d)^r)$ time to compute. There are r of them in each summand and so overall we take $O(poly((r+d)^r))$ time. A similar argument shows that number of terms in this polynomial is $O(poly((r+d)^r))$. Some more analysis shows that $P_{f,z}(\bar{x})$ is a form of degree d in $\bar{x}$ whose coefficients are homogeneous polynomials of dedgree d in f and degree $d(d-1)$ in $\bar{z}$. Let

$$B_f(\bar{x}, \bar{y}, \bar{z}) = (f \odot P_{f,z})(\bar{x}, \bar{y})$$

By the arguments given above calculating this form also takes time $poly((r+d)^r)$ and it has $poly((r+d)^r)$ terms. This is a homogeneous form in $(\bar{x}, \bar{y}, \bar{z})$ of multidegree $(d, d, d(d-1))$ and it's coefficients are forms of degree $(d+1)$ in the coefficients of f. See Section H., Chapter 4 in [GKZ94]. So in time $poly((r+d)^r)$ we can compute $B_f(\bar{x}, \bar{y}, \bar{z})$ explicitly.

Now we arrive at the main theorem

Theorem 2.14 (**Brill's Equation**, See 4.H, [GKZ94])**.** *A form $f(\bar{x})$ is a product of linear forms if and only if the polynomial $B_f(\bar{x}, \bar{y}, \bar{z})$ is identically* 0*.*

We argued above that computing $B_f(\bar{x}, \bar{y}, \bar{z})$ takes $O(poly((r+d)^r))$ time. It's degrees in $\bar{x}, \bar{y}, \bar{z}$ are all $poly(d)$ and so the number of coefficients when written as a polynomial over the $3r$ variables $(x_1, \dots, x_r, y_1, \dots, y_r, z, \dots, z_r)$ is $poly((r+d)^r)$. We mentioned that each coefficient is a polynomial of degree $(d+1)$ in the coefficients of f. Therefore we have the following corollary.

Corollary 2.15. *Let*

$$I \stackrel{def}{=} \{(\alpha_1, \dots, \alpha_n) : \forall i : \alpha_i \geq 0, \sum_{i \in [r]} \alpha_i = d\}$$

be the set capturing the indices of all possible monomials of degree exactly d in r variables $(x_1, \dots, x_r)$. Let $f_{\mathbf{a}}(y_1, \dots, y_r) = \sum_{\alpha \in I} a_\alpha \mathbf{y}^\alpha$ denote an arbitrary homogeneous polynomial. The coefficient vector then becomes $\mathbf{a} = (a_\alpha)_{\alpha \in I}$. Then there exists an explicit set of polynomials $F_1(\mathbf{a}), \dots, F_m(\mathbf{a})$ on $poly((r+d)^r)$ variables ($\mathbf{a} = (a_\alpha)_{\alpha \in I}$), with $m = poly((r+d)^r)$, $deg(F_i) \leq poly(d)$ such that for any particular value of $\mathbf{a}$, the corresponding polynomial $f_{\mathbf{a}}(\mathbf{y}) \in \Pi\Sigma^d_{\mathbb{R}}[\bar{y}]$ if and only if $F_1(\mathbf{a}) = \ldots = F_m(\mathbf{a}) = 0$. Also this set $\{F_i, i \in [m]\}$ can be computed in time $poly((r+d)^r)$ time.

Proof. Clear from the theorem and discussion above. □

Note that in our application $r = O(1)$ and so $poly((d+r)^r) = poly(d)$.

2.5 A Method of Reconstructing Linear Forms

In a lot of circumstances one might reconstruct a linear form (upto scalar multiplication) inside $V = Lin_{\mathbb{R}}[\bar{x}]$ from it's projections (upto scalar multiplication) onto some subspaces of V. For example consider a linear form $L = a_1x_1 + a_2x_2 + a_3x_3 (\in Lin_{\mathbb{R}}[x_1, x_2, x_3])$ with $a_3 \neq 0$, and assume we know scalar multiples of projections of L onto the spaces $\mathbb{R}x_1$ and $\mathbb{R}x_2$ i.e. we know $L_1 = \alpha(a_2x_2 + a_3x_3)$ and $L_2 = \beta(a_1x_1 + a_3x_3)$ for some $\alpha, \beta \in \mathbb{R}$. Scale these projections to $\tilde{L}_1 = x_3 + \frac{a_2}{a_3}x_3$ and $\tilde{L}_2 = x_3 + \frac{a_1}{a_3}x_3$. Using these two define a linear form $x_3 + \frac{a_1}{a_3}x_1 + \frac{a_2}{a_3}x_2$. This is a scalar multiple of our original linear form L. We generalize this a little more below.

Let $\bar{x} \equiv (x_1, \ldots, x_r)$, $\mathcal{B} = \{l_1, \ldots, l_r\}$ be a basis for $V = Lin_{\mathbb{R}}[x_1, \ldots, x_r]$. For $i \in \{0, 1, 2\}$, let S_i be pairwise disjoint non empty subsets of $\mathcal{B}$ such that $S_0 \cup S_1 \cup S_2 = \mathcal{B}$. Let $W_i = sp(S_i)$ and $W_i' = \bigoplus_{j \neq i} W_j$. Clearly $V = W_0 \oplus W_1 \oplus W_2 = W_i \oplus W_i', i \in \{0, 1, 2\}$.

Lemma 2.16. *Assume $L \in V$ is a linear form such that*

- $\pi_{W_2}(L) \neq 0$
- *For $i \in \{0, 1\}$, $L_i = \beta_i \pi_{W_i'}(L)$ are known for some non-zero scalars β_i.*

Then L is unique upto scalar multiplication and we can construct a scalar multiple $\tilde{L}$ of L.

Proof. Let $L = a_1l_1 + \ldots + a_rl_r, a_i \in \mathbb{R}$. Since $\pi_{W_2}(L) \neq 0$, there exists $l_j \in S_2$ such that $a_j \neq 0$. Let $\tilde{L} = \frac{1}{a_j}L$. For $i \in \{0, 1\}$, re-scale L_i to get $\tilde{L}_i$ making sure that coefficient of l_j is 1 in them. Thus for $i = 0, 1$

$$\pi_{W_i'}(\tilde{L}) = \tilde{L}_i$$

Since $W_0' = W_1 \oplus W_2$ and $W_1' = W_0 \oplus W_2$ by comparing coefficients we can get $\tilde{L}$. □

<u>(Algorithm)</u> Assume we know S_0, S_1, S_2 and therefore the basis change matrix to convert vector representations from $\mathcal{S}$ to $\mathcal{B}$. It takes $poly(r)$ time to convert $[v]_{\mathcal{S}}$ to $[v]_{\mathcal{B}}$. Given L_i in the basis $\mathcal{B}$ it takes $poly(r)$ time(by a linear scan) to find $l_j \in S_2$ with $a_j \neq 0$. This l_j has a non zero coefficient in both L_0, L_1. After this we just rescale L_i to get $\tilde{L}_i$ such that coefficient of l_j is 1. Then since $\tilde{L}_i = \pi_{W_i'}(\tilde{L})$ the coefficient of l_t in $\tilde{L}$ is as follows :

$$= \begin{cases} \text{coefficient of } l_t \text{ in } \tilde{L}_1 & : l_t \in S_0 \\ \text{coefficient of } l_t \text{ in } \tilde{L}_0 & : l_t \in S_1 \\ \text{coefficient of } l_t \text{ in } \tilde{L}_0 = \text{ coefficient of } l_t \text{ in } \tilde{L}_1 & : l_t \in S_2 \end{cases}$$

Finding the right coefficients using this also takes $poly(r)$ time.

Next we try and use this to reconstruct $\Pi\Sigma$ polynomials. This case is slightly more complicated and so we demand that the projections have some special form. In particular the projections onto one subspace preserves pairwise linear independence of linear factors and onto the other makes all linear factors scalar multiples of each other.

Corollary 2.17. *Let $S_i, W_i, i \in \{0, 1, 2\}$ be as above and $P \in \Pi\Sigma_{\mathbb{R}}[x_1, \ldots, x_r]$ such that*

1. $\pi_{W_2}(P) \neq 0$

2. For $i \in \{0,1\}$ *there exists* $\beta_i(\neq 0) \in \mathbb{R}$ *such that* $P_0 = \beta_0 \pi_{W_0'}(P) = p^t$ *and* $P_1 = \beta_1 \pi_{W_1'}(P) = d_1 \dots d_t$*. are known i.e.* p, d_j $(j \in [t])$ *and* t *are known.*

Then P *is unique upto scalar multiplication and we can construct a scalar multiple* $\tilde{P}$ *of* P*.*

Proof. Let $P = L_1 \dots L_t$ with $L_i \in Lin_{\mathbb{R}}[\bar{x}]$. There exists $\beta_i^j, i \in \{0,1\}, j \in [t]$, such that $\beta_0^j \pi_{W_0'}(L_j) = p$ and $\beta_1^j \pi_{W_1'}(L_j) = d_j$. Since p, d_j are known by above Lemma 2.16 we find a scalar multiple $\tilde{L_j} = \beta^j L_j$ of L_j and therefore find a scalar multiple $\tilde{P} = \tilde{L_1} \dots \tilde{L_t}$ of P. Note that this method also tells us that such a P is unique upto scalar multiplication. Since we've used the above Algorithm 2.5 at most t times with $t \leq deg(P)$, it takes $poly(deg(P), r)$ time to find $\tilde{P}$. □

This corollary is the backbone for reconstructing $\Pi\Sigma$ polynomials from their projections. But first we formally define a *"Reconstructor"*

Definition 2.18 (Reconstructor)**.** Let $S_i, W_i, i \in \{0,1,2\}$ be as above. Let Q be a standard $\Pi\Sigma$ polynomial and P be a standard $\Pi\Sigma$ polynomial dividing Q with $Q = PR$. Then (Q, P, S_0, S_1, S_2) is called a *Reconstructor* if:

- $\pi_{W_2}(P) \neq 0$.
- $\pi_{W_0'}(P) = \alpha p^t$, for some linear form p.
- Let $l \mid R$ be a linear form and $\pi_{W_2}(l) \neq 0$ then $gcd(\pi_{W_2}(P), \pi_{W_2}(l)) = 1$.

Note :
Let L_1, L_2 be two LI linear forms dividing P , then one can show

$$L_1, L_2 \text{ are LI } \Leftrightarrow \pi_{W_1'}(L_1), \pi_{W_1'}(L_2) \text{ are LI}$$

To see this first observe that the second bullet implies for $i \in [2], L_i \in W_0 + p \Rightarrow sp(\{L_1, L_2\}) \subseteq W_0 + p$. If $\pi_{W_1'}(L_1), \pi_{W_1'}(L_2)$ are LD then

$$sp(\{L_1, L_2\}) \cap W_1 \neq \{0\}$$

$\Rightarrow (W_0 + p) \cap W_1 \neq \{0\}$. Since $W_0 \cap W_1 = \{0\}$ we get that $p \in W_0 \oplus W_1 = W_2' \Rightarrow \pi_{W_2}(p) = 0 \Rightarrow \pi_{W_2}(P) = 0$ contradicting the first bullet.

Geometrically the conditions just mean that all linear forms dividing P have LD projection ($= \gamma p$ for some non zero $\gamma \in \mathbb{R}$) w.r.t. the subspace W_0' and LI linear forms p_1, p_2 dividing P have LI projections (w.r.t. subspace W_1'). Also no linear form l dividing R belongs to $fl(S_0 \cup S_1 \cup \{p\})$.

We are now ready to give an algorithm to reconstruct P using $\pi_{W_0'}(Q)$ and $\pi_{W_1'}(Q)$ by gluing appropriate projections corresponding to P. To be precise:

Claim 2.19. *Let* Q, P *be standard* $\Pi\Sigma$ *polynomials and* $P \mid Q$*. Assume* (Q, P, S_0, S_1, S_2) *is a* Reconstructor*. If we know both* $\pi_{W_0'}(Q)$ *and* $\pi_{W_1'}(Q)$*. Then we can reconstruct* P*.*

Proof. Here is the algorithm:

Algorithm 1 Reconstruct linear forms

procedure RECONSTRUCTOR ($\pi_{W_0'}(Q) \in \Pi\Sigma[\bar{x}], \pi_{W_1'}(Q) \in \Pi\Sigma[\bar{x}], S_0, S_1, S_2$)
  bool flag, $\Pi\Sigma$ polynomial P_0, P_1;
  Let $\pi_{W_0'}(Q) = \gamma \prod_{i\in[s]} c_i^{m_i}$, c_i's pairwise LI and normal, $\gamma \in \mathbb{R}$ (Factor using [KT90]).
  Let $\pi_{W_1'}(Q) = \delta d_1 \dots d_m$, $\delta \in \mathbb{R}$ and d_j normal (Factor using [KT90]).
  for ($i \in [s]$ && $\pi_{W_1'}(c_i) \neq 0$) **do**
    $flag = true, P_0 = c_i^{m_i}$;   // *Assuming projection w.r.t.* W_0' *to be* $c_i^{m_i}$.
    for ($j \in [s]$ && $j \neq i$ && $\pi_{W_1'}(c_j) \neq 0$) **do**
      if ($gcd(\pi_{W_1'}(c_i), \pi_{W_1'}(c_j)) \neq 1$) **then** flag = false;
    if (flag==true) **then**
      $P_1 = 1$;
      for ($j \in [m]$) **do**
        if ($\pi_{W_0'}(d_j) \neq 0$ & & $\{\pi_{W_0'}(d_j), \pi_{W_1'}(c_i)\}$ are LD) **then**
          $P_1 = P_1 d_j$ // *This steps collects projection w.r.t.* W_1' *in* P_1.
      if (($deg(P_1) = m_i$) && ((P_0, P_1) give $\tilde{P} = \beta P$ using Corollary 2.17)) **then**
        Make $\tilde{P}$ *standard* w.r.t. the standard basis $\mathcal{S}$ to get P and finally **return** P

2.5.1 Explanation

- The algorithm takes as input projections $\pi_{W_0'}(Q)$ and $\pi_{W_1'}(Q)$ along with the sets $S_i, i = 0, 1, 2$ which form a partition of a basis $\mathcal{B}$. We know that there exists a polynomial $P \mid Q$ such that (Q, P, S_0, S_1, S_2) is a reconstructor and so we try to compute the projections $\pi_{W_0'}(P), \pi_{W_1'}(P)$.
- If one assumes that $\pi_{W_0'}(Q) = \gamma \prod_{i\in[s]} c_i^{m_i}$ with the c_i's co-prime, then by the properties of a reconstructor the projection (of a scalar multiple of P) onto W_0' say $P_0 = \beta_0 \pi_{W_0'}(P)$ (for some β_0) has to be equal to $c_i^{m_i}$ for some i. We do this assignment inside the first for loop.
- The third property of a reconstructor implies that when we project further to W_2, it should not get any more factors and so we check this inside the second for loop by going over all other factors c_j of $\pi_{W_0'}(Q)$ and checking if c_i, c_j become LD on projecting to W_2.
- Now to find (scalar multiple of) the other projections i.e. $P_1 = \beta_1 \pi_{W_1'}(P)$ (for some β_1), we go through $\pi_{W_1'}(Q)$ and find d_k such that $\pi_{W_1'}(c_i) = \pi_{W_0'}(d_k)$ (i.e. they are projections of the same linear form). We collect the product of all such d_k's. If the choice of c_i were correct then all d_k's would be obtained correctly.
- The last *"if"* statement just checks that the number of d_k's found above is the same as m_i since $P_0 = c_i^{m_i}$ tells us that the degree of P was m_i. We recover a scalar multiple of P using the algorithm explained in Corollary 2.17 and then make it standard to get P.

2.5.2 Correctness

The corectness of our algorithm is shown by the lemma below.

Claim 2.20. *If* (Q, P, S_0, S_1, S_2) *is a reconstructor then Algorithm 1 returns* P.

Proof. (Q, P, S_0, S_1, S_2) is a reconstructor therefore

- $\pi_{W_2}(P) \neq 0$
- $\pi_{W_0'}(P) = \delta p^t$
- $q \mid \frac{Q}{P} \Rightarrow gcd(\pi_{W_2}(q), \pi_{W_2}(P)) = 1$

1. It is clear that for one and only one value of i, c_i divides p. Fix this i. Let $Q = PR$, if $c_i^{m_i} \nmid \pi_{W_0'}(P)$ then $c_i \mid l$ for some linear form $l \mid \pi_{W_0'}(R)$. Condition 3 in definition of Reconstructor implies that $gcd(\pi_{W_2}(P), \pi_{W_2}(l)) = 1$ but $\pi_{W_2}(c_i)$ divides both of them giving us a contradiction. Since $\pi_{W_0'}(P)$ has just one linear factor $\Rightarrow \pi_{W_0'}(P)$ is a scalar multiple of $c_i^{m_i}$ for some i.
2. Assume the correct $c_i^{m_i}$ has been found. Now let $d_j \mid \pi_{W_1'}(Q)$ such that $\{\pi_{W_2}(c_i), \pi_{W_2}(d_j)\}$ are LD. then we can show that $d_j \mid \pi_{W_1'}(P)$. Assume not, then for some linear form $l \mid R = \frac{Q}{P}$, $d_j \mid \pi_{W_1'}(l)$. $\pi_{W_0'}(d_j) \neq 0$ (which we checked) $\Rightarrow \pi_{W_2}(l) \neq 0$. So we get $\pi_{W_2}(c_i) \mid \pi_{W_2}(l)(\neq 0)$ and so $\pi_{W_2}(c_i) \mid gcd(\pi_{W_2}(P), \pi_{W_2}(l))$ which is therefore $\neq 1$ and condition 3 of Definiton 2.18 is violated. So whatever d_j we collect will be a factor of $\pi_{W_1'}(P)$ and we will collect all of them since they are all present in $\pi_{W_1'}(Q)$.
3. We know from proof of Corollary 2.17 that if we know c_i, m_i and d_j's correctly then we can recover a scalar multiple of P correctly. But Q, P are standard so we return P correctly.

□

In fact we can show that if we return something it has to be a factor of Q.

Claim 2.21. *If Algorithm 1 returns a $\Pi\Sigma$ polynomial P, then $P \mid Q$*

Proof.

- If the algorithm returned a $\Pi\Sigma$ polynomial P then $flag$ has to be true at end. So there is an $i \in [s]$ such that $P_0 = c_i^{m_i}$ with the conditions that $\pi_{W_1'}(c_i) \neq 0$ and $gcd(c_i, c_j) = 1$ for $j \neq i$. It also means that for exactly m_i of the d_j's (say $d_1, \ldots, d_{m_i}$) $\{\pi_{W_1'}(c_i), \pi_{W_0'}(d_j)\}$ are LD and $P_1 = d_1 \ldots d_{m_i}$.
- Since $c_i^{m_i} \mid \pi_{W_0'}(Q)$, there exists a factor $\tilde{P} \mid Q$ of degree m_i such that $\pi_{W_0'}(\tilde{P}) = c_i^{m_i}$ and $\pi_{W_1'}(c_i) \neq 0$. This $\Rightarrow \pi_{W_2}(\tilde{P}) \neq 0$. Clearly $\pi_{W_1'}(\tilde{P}) \mid \pi_{W_1'}(Q) = d_1 \ldots d_m$, hence for all linear factors $\tilde{p}$ of $\tilde{P}$, $\pi_{W_1'}(\tilde{p})$ should be some d_j with the condition that $\{\pi_{W_0'}((\pi'_{W_1})(\tilde{p})), \pi_{W_1'}(c_i)\}$ should be LD. The only choice we have are $d_1, \ldots, d_{m_i}$. So $\pi_{W_0'}(\tilde{P}) = d_1 \ldots d_{m_i}$. All conditions of Corollary 2.17 are true and so $\tilde{P}$ is uniquely defined (upto scalar multiplication) by the reconstruction method given in Corollary 2.17. So what we returned was actually a factor of Q.

□

2.5.3 Time Complexity

Factoring $\pi_{W_0'}(Q), \pi_{W_1'}(Q)$ takes $poly(d)$ time (using Kaltofen's Factoring from [KT90]). The nested for loops run $\leq d^3$ times. Computing projections with respect to the known decomposition $W_0 \oplus W_1 \oplus W_2 = \mathbb{R}^r$ of linear forms over r variables takes $poly(r)$ time. Computing gcd and linear independence of linear forms takes $poly(r)$ time. The final reconstruction of P using (P_0, P_1) takes $poly(d, r)$ time as has been explained in Corollary 2.17. Multiplying linear forms to $\Pi\Sigma$ polynomial takes $poly(d^r)$ time. Therefore overall the algorithm takes $poly(d^r)$ time. In our application $r = O(1)$ and therefore the algorithm takes $poly(d)$ time.
□

3 Reconstruction for low rank

For this whole section we fix r to be any constant $> \max(C_{2k-1}+k, c_{\mathbb{R}}(4))$, where $C_k = \frac{C^k}{\delta}$ is the constant that appears in Theorem 2.11. δ is some fixed number in $(0, \frac{7-\sqrt{37}}{6})$ and C comes from Theorem 2.11. $c_{\mathbb{R}}(4) = 3(4)^2 = 48$, is the rankbound needed for uniqueness of $\Sigma\Pi\Sigma(2)$ circuits as shown in Theorem 1.7.

Our main theorem for this section therefore is:

Theorem 3.1. *Let r be as defined above. Consider $f(\bar{x}) \in \mathbb{R}[\bar{x}]$, a multivariate homogeneous polynomial of degree d over the variables $\bar{x} = (x_1, \ldots, x_r)$ which can be computed by a $\Sigma\Pi\Sigma_{\mathbb{R}}(2)[\bar{x}]$ circuit C. Assume that rank of the simplification of C i.e. $Sim(C) = r$. We give a $poly(d)$ time randomized algorithm which computes C given blackbox access to $f(\bar{x})$.*

We assume f has the following $\Sigma\Pi\Sigma_{\mathbb{R}}(2)[\bar{x}]$ representation:

$$f = \tilde{G}(\tilde{\alpha}_0\tilde{T}_0 + \tilde{\alpha}_1\tilde{T}_1)$$

where $\tilde{G}, \tilde{T}_i \in \Pi\Sigma_{\mathbb{R}}[\bar{x}]$ are $normal$ (i.e. leading non-zero coefficient is 1 in every linear factor) and $\tilde{\alpha}_0, \tilde{\alpha}_1 \in \mathbb{R}$ with $gcd(\tilde{T}_0, \tilde{T}_1) = 1$. The $rank(Sim(C)) = r$ condition then becomes

$$sp(\mathcal{L}(\tilde{T}_0) \cup \mathcal{L}(\tilde{T}_1)) = Lin_{\mathbb{R}}[\bar{x}]$$

Consider the set $T = \mathcal{L}(\tilde{G}) \cup \mathcal{L}(\tilde{T}_0) \cup \mathcal{L}(\tilde{T}_1)$. By abuse of notation we will treat these linear forms also as points in $\mathbb{R}^r$. Since linear factors of $\tilde{G}, \tilde{T}_i$ are normal, two linear factors of $\tilde{G}, \tilde{T}_i$ are LD iff they are same.

Random Transformation and Assumptions Let Ω, Λ be two $r \times r$ matrices such that their entries $\Omega_{i,j}$ and $\Lambda_{i,j}$ are picked independently from the uniform distribution on $[N]$. Here $N = 2^d$. We begin our algorithm by making a few assumptions. All of these assumptions are true with very high probability and we assume them in our algorithm. Consider the standard basis of $\mathbb{R}^r$ given as $\mathcal{S} = \{e_1, \ldots, e_r\}$. Let $E_j = sp(\{e_1, \ldots, e_j\})$ and $E'_j = sp(\{e_{j+1}, \ldots, e_r\})$, clearly $\mathbb{R}^r = E_j \oplus E'_j$. Let $\pi_{W_{E_j}}$ be the orthogonal projection onto E_j w.r.t. this decomposition.

- **Assumption 0 :** Ω is invertible. This is just the complement of event $\mathcal{E}_0$ in Section 2.2 and so occurs with high probability.
- **Assumption 1 :** For all $t \in T$, $\pi_{W_{E_1}}(\Omega(t)) \neq 0$ i.e. $[\Omega(t)]^1_{\mathcal{S}} \neq 0$ (coefficient of e_1 is non-zero) . This is the complement of event $\mathcal{E}_1$ in Section 2.2 and so occurs with high probability.
- **Assumption 2 :** For all LI sets $\{t_1, \ldots, t_r\} \subset T$, $\{\Omega(t_1), \ldots, \Omega(t_r)\}$ is LI. This essentially means that Ω is invertible. This is the complement of $\mathcal{E}_2$ in Section 2.2 and so occurs with high probability.
- **Assumption 3 :** Fix a $k < r$. For all LI sets $\{t_1, \ldots, t_r\} \subset T$, $\{\Omega(t_1), \ldots, \Omega(t_k), \Lambda\Omega(t_{k+1}), \ldots, \Lambda\Omega(t_d)\}$ is LI i.e. is a basis. This is the complement of event $\mathcal{E}_3$ in Section 2.2 and so occurs with high probability. It'll be used later in this chapter.
- **Assumption 4 :** Fix a $k < r$. For all LI sets $\tilde{T} = \{t_1, \ldots, t_r\} \subset T$ and define the set $\mathcal{B} = \{\Omega(t_1), \ldots, \Omega(t_k), \Lambda\Omega(t_{k+1}), \ldots, \Lambda\Omega(t_r)\}$. By Assumption 3 this is a basis. Consider any $t \in T$ such that $\Omega(t) \notin sp(\{\Omega(t_1), \ldots, \Omega(t_k)\})$. Then $[\Omega(t)]^{k+1}_{\mathcal{B}} \neq 0$. This event is the complement of $\mathcal{E}_5$ and so it occurs with high probability.

From now onwards we will assume that all the above assumptions are true. Since all of them occur with very high probability, their complements occur with very low probability and by union bound the union of their complements is a low probability event. So intersection of the above assumptions occurs with high probability and we assume all of them are true. **Note that the assumptions will continue to be true if we scale all linear forms (possibly different scaling for different vectors, but non-zero scalars) in T i.e. if the assumptions were true for T then they would have been true had we started with a scaling of T.**

The first step of our algorithm is to apply Ω to f. We have a natural identification between linear forms and points in $\mathbb{R}^r$. This identification converts Ω into a linear map on $Lin_{\mathbb{R}}[\bar{x}]$ which can be further converted to a ring homomorphism on polynomials by assuming that it preserves the products and sums of polynomials. So Ω gets applied to all linear forms in the $\Sigma\Pi\Sigma(2)$ representation of f. Since f is a degree d polynomial in r variables it has atmost $poly(d^r)$ coefficients. Applying Ω to each monomial and expanding it takes $poly(d^r)$ time and gives $poly(d^r)$ terms. So computing $\Omega(f)$ takes $poly(d^r)$ time and has $poly(d^r)$ monomials.

Now we try and reconstruct the circuit for $\Omega(f)$. If this reconstruction can be done correctly, we can apply Ω^{-1} and get back f. Note that **Assumption 1** tells us that the coefficient of x_1 in $\Omega(l)$ is non-zero for all l in T. Let $X = \{x_1, \ldots, x_r\}$ and $\bar{x}$ is used for the tuple $(x_1, \ldots, x_r)$. From this discussion we know that:

$$\Omega(f) = \Omega(\tilde{G})(\tilde{\alpha}_0\Omega(\tilde{T}_0) + \tilde{\alpha}_1\Omega(\tilde{T}_1)) = G(\alpha_0 T_0 + \alpha_1 T_1)$$

where α_i are chosen such that linear factors of G, T_i have their first coefficient(that of x_1) equal to 1. So they are $standard$ $\Pi\Sigma$ polynomials. Note that we've used **Assumption** 1 here. Since we've moved constants to make linear forms standard we can assume $G = \lambda\Omega(\tilde{G}), T_i = \lambda_i\Omega(\tilde{T}_i)$ with $\lambda, \lambda_i \in \mathbb{R}$. Consider some scaling T_{sc} of T such that $\mathcal{X} = \mathcal{L}(G) \cup \mathcal{L}(T_0) \cup \mathcal{L}(T_1)$ is $= \Omega(T_{sc})$. All above assumptions are true for T_{sc} and so we may use the conclusions about $\Omega(T_{sc})$ i.e. $\mathcal{X}$. Also since Ω is invertible $gcd(T_0, T_1) = 1$.

Let $T_i = \prod\limits_{j\in[M]} l_{ij}, i = 0, 1$ and $G = \prod\limits_{k\in[d-M]} G_k$, with $l_{ij}, G_k \in Lin_{\mathbb{R}}[\bar{x}]$ (so $d = deg(f)$).
For simplicity from now onwards we call $\Omega(f)$ by f and try to reconstruct it's circuit. Once this is done we may apply Ω^{-1} to all the linear forms in the gates and get the circuit for f. This step clearly takes $poly(d^r)$ time in the same way as applying Ω took.

Since r is a constant, the steps described above take $poly(d)$ time overall.

Known and Unknown Parts We also define some other $\Pi\Sigma_{\mathbb{R}}[\bar{x}]$ polynomials $K_i, U_i, i = 0, 1$ which satisfy

$$K_i \mid \alpha_i G T_i, U_i = \frac{\alpha_i G T_i}{K_i}.$$

with the extra condition

$$gcd(K_i, U_i) = 1.$$

K_i are the known factors of $\alpha_i G T_i$ and U_i the unknown factors. The gcd condition just means that that known and unknown parts of $\alpha_i G T_i$ don't have common factors. In other words linear forms in $\alpha_i G T_i$ are known with full multiplicity. We initialize $K_i = 1$ and during the course of the algorithm update them as and when we recover more linear forms. At the end $K_i = \alpha_i G T_i$ and so we know both gates.

3.1 Outline of the algorithm

1. **Set $\mathcal{C}$ of Candidate Linear Forms :**

We compute a $poly(d)$ size set $\mathcal{C}$ of linear forms which contains $\mathcal{L}(T_i), i = 0, 1$. We will non-deterministically guess from this set $\mathcal{C}$ making only a constant number of guesses everytime(thus polynomial work overall). It is important to note that the uniqueness of our circuit guarantees that our answer if computed can always be tested to be right.

2. **Easy Case 1 : -** $\boxed{dim(^{sp(T_{1-i}) + sp(T_i)}/_{sp(T_i)}) \geq 2 \text{ for some } i \in \{0, 1\} :}$

   So T_{1-i} has two linear factors $l_{(1-i)1}, l_{(1-i)2}$ such that $sp(\{l_{(1-i)1}, l_{(1-i)2}\}) \cap sp(T_i) = \{0\}$. In this case we show that the only linear factors of f are those which appear in G. So we can first factorize f using Kaltofen's factoring ([KT90]) and obtain G. Update $K_j = G, j = 0, 1$. So $U_j = \alpha_j T_j$ for $j = 0, 1$. Clearly we also have $\mathcal{L}(T_{1-i}) \subsetneq sp(T_i) = sp(U_i)$ and we can go to **Easy Case 2** below.

3. **Easy Case 2 :** $\boxed{\mathcal{L}(T_{1-i}) \subsetneq sp(U_i), \text{ for some } i \in \{0, 1\} :}$

   So T_{1-i} has a linear factor $l_{(i-1)1}$ such that

   $$sp(\{l_{(i-1)1}\}) \cap sp(U_i) = \{0\} \tag{1}$$

   Let $W = sp(\{l_{(1-i)1}\})$ and extend to a basis of V and in the process obtain another subspace $W' \subset V$ such that $W \oplus W' = V$. We can see from Equation 1 that LI linear forms in U_i remain LI when we project to W'. We use this to compute U_i and then since $K_i U_i = \alpha_i G T_i$ we know one of the gates. To find the other gate simply factorize $f - \alpha_i G T_i$. If it factors into a product of linear forms we have the reconstruction.

4. **Hard Case :** $\boxed{\mathcal{L}(T_{1-i}) \subseteq sp(U_i), \text{ for } i = 0 \text{ and } 1 :}$

   We know that we are not in **Easy Case 1** and so $dim(sp(T_0) + sp(T_1)) - sp(T_i) \leq 1$ for $i = 0, 1$. Also $dim(sp(T_0) + sp(T_1)) = r$ by assumption on the simple rank of our polynomial. So this guarantees that $dim(sp(T_{1-i})) \geq r - 1 \Rightarrow$ (by the condition of this hard case) $dim(sp(U_i)) \geq r - 1$ for $i = 0, 1$ and therefore enables us to use the Quantitative Sylvester Gallai theorems with the sets $\mathcal{L}(T_i), \mathcal{L}(U_i)$. Our first step is to identify a certain *"bad"* factor I of G and get rid of it to get $G^\star = \frac{G}{I}$ and thus $f^\star = \frac{f}{I}$. This is done using something we call a Detector Pair (See 3.5) whose existence is shown using the Quantitative Sylvester Galai Theorems mentioned above. So now we try reconstructing $f^\star$ with known (and unknown resp.) parts as $K_0^\star, K_1^\star$ ($U_0^\star, U_1^\star$ resp.). If $sp(U_{1-i}^\star)$ becomes small we may fall in **Easy Case** 2 and recover the whole circuit directly. Otherwise the same detector pairs then provide certain *"nice"* subspaces corresponding to linear forms in T_i. Projection of $U_{1-i}^\star$ onto these subspaces can be easily glued together to recover some linear factors(with multiplicities) of $U_{1-i}^\star$, which will then be multiplied to $K_{1-i}^\star$. The process continues as long as $sp(U_{1-i}^\star)$ remains large. As soon as this condition fails we end up in **Easy Case** 2 and the gates are recovered.

3.2 Set $\mathcal{C}$ of Candidate Linear Forms

This section deals with constructing a $poly(d)$ size set $\mathcal{C}$ which contains each $l_{ij}, (i, j) \in \{0, 1\} \times [M]$. First we define the set and prove a bound on it's size.

3.2.1 Structure and Size of $\mathcal{C}$

Let's recall $f = G(\alpha_0 T_0 + \alpha_1 T_1)$ and define two other polynomials:

$$g = \frac{f}{G} = \alpha_0 T_0 + \alpha_1 T_1$$

$$h = \frac{f}{Lin(f)} = \frac{g}{Lin(g)}$$

Assume $deg(h) = d_h$

Definition 3.2. Our candidate set is defined as:

$$\mathcal{C} \stackrel{def}{=} \{l = x_1 - a_2x_2 - \ldots - a_rx_r \in Lin_{\mathbb{R}}[\bar{x}] : h(a_2x_2 + \ldots + a_rx_r, x_2, \ldots, x_r) \in \Pi\Sigma_{\mathbb{R}}^{d_h}[x_2, \ldots, x_r]\}$$

(for definition of $\Pi\Sigma_{\mathbb{R}}^{d_h}[x_2, \ldots, x_r]$ See Section 2.1)

In the claim below we show that linear forms dividing polynomials $T_i, i = 0, 1$ are actually inside $\mathcal{C}$ (first part of claim). The remaining linear forms in $\mathcal{C}$ (which we call *"spurious"*) have a nice structure (second part of claim). In the third part of our claim we arrive at a bound on the size of $\mathcal{C}$. Recall the definition of $c_{\mathbb{R}}(k)$ from Theorem 1.6.

Claim 3.3. *The following are true about our candidate set $\mathcal{C}$.*

1. $\mathcal{L}(T_i) \subseteq \mathcal{C}, i = 0, 1$.
2. *Let $k = c_{\mathbb{R}}(3) + 2$ and suppose $\{l_j; j \in [k]\} \subset \mathcal{L}(T_i)$ are LI . Then for any $l \in \mathcal{C} \setminus (\mathcal{L}(T_0) \cup \mathcal{L}(T_1))$, there exists $j \in [k]$ such that $fl(\{l, l_j\}) \cap \mathcal{L}(T_{1-i}) \neq \phi$ i.e. the line joining l and l_j does not intersect the set $\mathcal{L}(T_{1-i})$.*
3. $|\mathcal{C}| \leq M^4 + 2M \leq d^4 + 2d$.

Proof. See A.1 in Appendix. □

Let's now give an algorithm to construct this set.

3.2.2 Constructing the set $\mathcal{C}$

Here is an algorithm to construct the set $\mathcal{C}$. An explanation is given in the lemma below.

Algorithm 2 Find the set $\mathcal{C}$ of candidate linear forms (returns a set)

---

procedure $Candidates(f \in \Sigma\Pi\Sigma_{\mathbb{R}}(2)[\bar{x}])$
  Define $\mathcal{C} = \phi$;
  Use polynomial factorization from [KT90] to find $Lin(f)$ i.e. the product of all linear factors of f.
  Consider polynomial $h = \frac{f}{Lin(f)}$
  Let $a_2, \ldots, a_r$ be variables.
  Compute the coefficient vector **b** of $h(a_2x_2 + \ldots + a_rx_r, x_2, \ldots, x_r)$ with entries as polynomials in $a_2, \ldots, a_r$.
  Consider the polynomials $\{F_i, i \in [m]\}$ constructed in Corollary 2.15.
  Using your favorite algorithm (eg. Buchberger's [Buc76]) to solve polynomial equations, find all complex solutions to the system $\{F_i(\mathbf{b}) = 0, i \in [m]\}$.
  For each solution $(a_2, \ldots, a_r) \in \mathbb{R}^r$ do : $\mathcal{C} = \mathcal{C} \cup \{(1, a_2, \ldots, a_r)\}$.
   return $\mathcal{C}$;

Lemma 3.4. *Given a polynomial $f \in \mathbb{R}[x_1, \ldots, x_r]$ of degree d in r independent variables which admits a $\Sigma\Pi\Sigma_{\mathbb{R}}(2)[x_1, \ldots, x_r]$-representation : $f = \prod_{i \in [d-M]} G_i(\alpha_0 \prod_{j \in [M]} l_{0j} + \alpha_1 \prod_{k \in [M]} l_{1k})$ such that $G_t, l_{ij}(t \in [d-M], i \in \{0, 1\}, j \in [M])$ are standard w.r.t. the standard basis $\{x_1, \ldots, x_n\}$ then we can find in deterministic time $poly(d)$, the corresponding candidate set $\mathcal{C}$ (see Definition 3.2) described above.*

Proof. The proof also contains an explanation of the algorithm above

- Let $l = x_1 - a_2x_2 - \ldots - a_rx_r \in \mathcal{C}$ be a candidate linear form. We know that $h(a_2x_2 + \ldots + a_rx_r, x_2, \ldots, x_r) \in \Pi\Sigma_{\mathbb{R}}^{d_h}[x_2, \ldots, x_r] \subset \Pi\Sigma_{\mathbb{C}}^{d_h}[x_1, \ldots, x_r]$.
- Using Theorem 2.15 we know that $h(a_2x_2 + \ldots + a_rx_r, x_2, \ldots, x_r) \in \Pi\Sigma_{\mathbb{C}}^{d_h}[x_2, \ldots, x_r] \Leftrightarrow$ for the coefficient vector $\mathbf{b}$ of $h(a_2x_2 + \ldots + a_rx_r, x_2, \ldots, x_r)$ inside $\mathbb{C}[x_2, \ldots, x_r]$ satisifes $F_1(\mathbf{b}) = \ldots = F_m(\mathbf{b}) = 0$ for the polynomials $\{F_i : i \in [m]\}$ obtained in Corollary 2.15. .
- For any $t \le d_h$, computing $(a_2x_2 + \ldots + a_rx_r)^t$ requires $poly(t^r)$ time and it also has $poly(t^r)$ terms and degree t. Multiplying such powers to other variables and adding $poly(d_h^r)$ many such expressions also requires $poly(d_h^r)$ time. Hence computing the coefficient vector **b** takes polynomial time since r is a constant. Each co-ordinate of this coefficient vector is a polynomial in $r-1$ variables $(a_2, \ldots, a_r)$ of degree $poly(d_h^r)$.
- Now we think of the a_i's as our unknowns and obtain them by solving the polynomial system $\{F_i(\mathbf{b}) = 0, i \in [m]\}$. The number of polynomials is $m = poly(d^r)$ and degrees are $poly(d)$. F_i's are polynomials in $poly(d^r)$ variables. Expanding $F_i(\mathbf{b})$ will clearly take $poly(d^r)$ time and now we will have $poly(d^r)$ polynomials in r variables of degrees $poly(d^r)$. Note that $r = O(1)$ and so we need to solve $poly(d)$ polynomials of degree $poly(d)$ in constant many variables. Also Claim 3.3 implies that the number of solutions $\le M^4 + 2M = O(poly(d))$. So using Buchberger's algorithm [Buc76] we can solve the system for $(a_2, \ldots, a_r)$ in $poly(d)$ time. Once we have the solutions we consider only those linear forms which are in $\mathbb{R}[x_1, \ldots, x_r]$ and add them to $\mathcal{C}$.

□

We give algorithms for **Easy Case** 1 **and** 2. **Hard Case** will require more prepration and will be done after these subsections.

3.3 Easy Case 1 : $\boxed{dim({}^{sp(T_{1-i}) + sp(T_i)}/_{sp(T_i)}) \ge 2 \text{ for some } i \in \{0, 1\}}$

Claim 3.5. *If* $dim({}^{sp(T_{1-i}) + sp(T_i)}/_{sp(T_i)}) \ge 2$ *then* $\mathcal{L}(\alpha_iT_i + \alpha_{1-i}T_{1-i}) = \phi$.

Proof. $dim({}^{sp(T_{1-i}) + sp(T_i)}/_{sp(T_i)}) \ge 2 \Rightarrow$, there exists $l_1', l_2' \in \mathcal{L}(T_{1-i}) \setminus sp(T_i)$ be such that $dim(\{l_1', l_2'\} \cup \mathcal{L}(T_i)) = dim(\mathcal{L}(T_i)) + 2$. Assume there exist $l \in \mathcal{L}(\alpha_iT_i + \alpha_{1-i}T_{1-i})$.

$$l \mid \alpha_iT_i + \alpha_{1-i}T_{1-i} \Rightarrow l \nmid T_i \text{ and } l \nmid T_{1-i} \text{ (since they are coprime)}$$

$$0 \ne \alpha_i \prod_{j \in [M]} l_{ij} = -\alpha_{1-i} \prod_{j \in [M]} l_{(1-i)j} \pmod{\{l\}}.$$

Thus there exist $l_1, l_2 \in \mathcal{L}(T_i)$ and scalars $\gamma_j, \delta_j, j \in [2]$ such that $l = \gamma_jl_j + \delta_jl_j'$. Since $l \nmid T_0, l \nmid T_1$ we get γ_j, δ_j are non zero.
$\delta_1, \delta_2 \ne 0 \Rightarrow$,

$$l_1', l_2' \in sp(\{l\} \cup \mathcal{L}(T_i)) \Rightarrow dim(\{l_1', l_2'\} \cup \mathcal{L}(T_i)) \le dim(\mathcal{L}(T_i)) + 1$$

which is a contradiction. So $\mathcal{L}(\alpha_iT_i + \alpha_{1-i}T_{1-i}) = \phi$. □

Therefore the only linear factors of f are present in G, which can now be correctly found by using Kaltofen's algorithm [KT90] and identifying the linear factors. Update $K_j = G$ for $j = 0, 1$, therefore $U_j = T_j$. Also this case implies that $\mathcal{L}(T_{1-i}) \subsetneq sp(T_i) = sp(U_i)$. and so we can go to the next case.

Algorithm 3 Easy Case 1 - Gates span different spaces

procedure *bool gatespan_uneven*$(f \in \Sigma\Pi\Sigma_{\mathbb{R}}(2)[\bar{x}], \mathcal{C} \subset Lin_{\mathbb{R}}[\bar{x}])$
  Apply Kaltofen's factoring algorithm [KT90] and find $Lin(f)$;
  if ($(recon\_uneven(f, Lin(f), Lin(f), \mathcal{C}) == true)$) **then return** Both Gates and true;
  return false;

The above algorithm does exactly what has been explained in the preceeding paragraph. It works in $poly(d)$ time if $recon\_uneven(f, K_0, K_1, \mathcal{C})$ works in $poly(d)$ time. Kaltofen's factoring and all other steps are $poly(d)$ time.

3.4 Easy Case 2 : $\boxed{\mathcal{L}(T_{1-i}) \subsetneq sp(U_i), \textbf{ for some } i \in \{0, 1\}}$

Claim 3.6. *Suppose for some $i \in \{0, 1\}$, $\mathcal{L}(T_{1-i}) \subsetneq sp(U_i)$ then we can reconstruct f.*

Algorithm 4 Easy Case 2 - Some gate has extra dimensions

procedure *bool* $recon\_uneven(f \in \Sigma\Pi\Sigma_{\mathbb{R}}(2)[\bar{x}], K_0 \in \Pi\Sigma_{\mathbb{R}}[\bar{x}], K_1 \in \Pi\Sigma_{\mathbb{R}}[\bar{x}], \mathcal{C} \subset Lin_{\mathbb{R}}[\bar{x}])$
  for $(i \in \{0, 1\}$) **do** // *i such that T_{1-i} has extra dimensions*
    for (every LI set $\{l_1, l_2, \ldots, l_r\} \subset \mathcal{C}$) **do** // *Guess $l_1 \in \mathcal{L}(T_{1-i})$, $sp(U_i) \subset sp(\{l_2, \ldots, l_r\})$*
      $K'_i = K_i$;
      Compute t such that $l_1^t \mid\mid f$; // *i.e. $l_1^t \mid f$ && $l_1^{t+1} \nmid f$*
      $W = sp(\{l_1\})$; and $W' = sp(\{l_2, \ldots, l_r\})$; // $V = W \oplus W'$
      if $l_1^t \mid\mid K'_i$ **then**
        $\tilde{f} = \frac{f}{l_1^t}$; $\tilde{K}_i = \frac{K'_i}{l_1^t}$;
        if $U_i = \frac{\pi_{W'}(\tilde{f})}{\pi_{W'}(\tilde{K}_i)} \in \Pi\Sigma_{\mathbb{R}}[\bar{x}]$ && $f - K_iU_i \in \Pi\Sigma_{\mathbb{R}}[\bar{x}]$ **then**
          $K_i = K_iU_i$; $K_{1-i} = f - K_iU_i$;
          return Both gates and true;
  return false;

Explanation and Correctness Analysis

- The first for loop just guesses the gate with extra dimensions i.e. it's not contained in span of the unknown part of the other gate.
- If for some basis $\{l_1, \ldots, l_r\} \subset \mathcal{C}$ the algorithm actually computes a $\Sigma\Pi\Sigma(2)$ representation in the end then it ought to be correct since the last 'if' also checks if it is correct.
- If our guess for i is correct, we show that there exists a basis $\{l_1, \ldots, l_r\} \subset \mathcal{C}$ for which all conditions will be satisfied and we actually arrive at a $\Sigma\Pi\Sigma(2)$ representation in the end. Since $\mathcal{L}(T_{1-i}) \subsetneq sp(U_i)$ and $\mathcal{L}(T_{1-i}), sp(U_i) \subset \mathcal{C}$ there exists $l_1 \in \mathcal{L}(T_{1-i}) \setminus sp(U_i) \subset \mathcal{C}$. Choose a basis $\{l_2, \ldots, l_s\}$ of $sp(U_i)$, then $\{l_1, \ldots, l_s\}$ is an LI set. Now extend this to a basis $\{l_1, \ldots, l_s, l_{s+1}, \ldots, l_r\} \subset \mathcal{C}$ of V. We go over all such choices of basis in $\mathcal{C}$ and will arrive at the right one.

- We initialize a dummy polynomial K'_i to represent K_i since we do not want to update K_i till we actually have a solution. Let's assume $l_1^t \mid\mid f$. We know $l_1 \mid T_{1-i} \Rightarrow l_1 \nmid T_i \Rightarrow l_1 \nmid \alpha_i T_i + \alpha_{1-i} T_{1-i}$. Therefore $l_1^t \mid\mid G \Rightarrow l_1^t \mid\mid \alpha_i G T_i = K_i U_i$. Also $l_1 \notin sp(U_i) \Rightarrow l_1 \nmid U_i$ thus $l_1^t \mid\mid K_i \Rightarrow l_1^t \mid\mid K'_i$. We remove l_1^t from both f, K'_i to get $\tilde{f}, \tilde{K}_i$. Let $W = sp(\{l_1\})$ and $W' = sp(\{l_2, \dots, l_r\})$, therefore $V = W \oplus W'$. Note that since $l_1 \in \mathcal{L}(T_{1-i})$

$$\pi_{W'}(\tilde{f}) = \pi_{W'}(U_i)\pi_{W'}(\tilde{K}_i)$$

Since $\pi_{W'}(\tilde{K}_i) \neq 0$, we get $\pi_{W'}(U_i) = \frac{\pi_{W'}(\tilde{f})}{\pi_{W'}(\tilde{K}_i)}$. If $U_i = u_1 \dots u_s$ with $u_j \in W'$, we see that $\pi_{W'}(U_i) = \pi_{W'}(u_1) \dots \pi_{W'}(u_s) = u_1 \dots u_s = U_i$. So we get U_i and hence $\alpha_i G T_i = K_i U_i$. Once $\alpha_i G T_i$ is known we factorize $f - \alpha_i G T_i$ to get $\alpha_{1-i} G T_{1-i}$. For the correct choice of our basis this will factorize completely into a $\Pi\Sigma$ polynomial. Now we update $K_i = K_i U_i$ and $K_{1-i} = f - k_i U_i$ and **return true**. Throughout the algorithm we use Kaltofen's factoring [KT90] wherever necessary.

- If we were not able to find the $\Sigma\Pi\Sigma(2)$ representation then we were not in this case and **return false**.

Time Complexity - We can see above all loops run only $poly(d)$ many times. The most expensive step is choosing r vectors from $\mathcal{C}$. But recall that r is a constant and so this also takes only polynomial time in d. Other steps like factoring polynomials (using Kaltofen's factoring algorithm from [KT90]), taking projection onto known subspaces, divding by polynomials require $poly(d)$ time (r is a constant) as has been explained multiple times before.

Now we need to handle the **Hard Case**. This is quite technical and so we do some more preparation. We devise a technique to get rid of some factors of f to get a new polynomial $f^\star$ without destroying the $\Sigma\Pi\Sigma(2)$ structure. If Easy Case 2 holds for $f^\star$ we stop there itself. Otherwise we will use combination of different subspaces of V, project $f^\star$ onto them and glue projections to get gates for $f^\star$.

3.5 Detector Pair, Reducing Factors, Hard Case Preparation

We outline an approach to identify some factors of f. These factors will divide G but won't divide g. This is going to be useful in the Hard Case. The linear factors left after removing these identified factors will have very strong structural properties and so will be instrumental in reconstruction. The main tool in this identification is a pair (S, D) (defined below) inside one of the $\mathcal{L}(T_i)$'s. This pair will be called a *"Detector Pair"*. It will also decide the subspaces on which we take projections of f and glue back to get the gates.

Detector Pairs (S, D) Fix $k = c_{\mathbb{R}}(3) + 2$ (See Theorem 1.6 for definition of $c_{\mathbb{R}}(m)$). Let $S = \{l_1, \dots, l_k\} \subset \mathcal{L}(T_i)$ be an LI set of linear forms. Let $D(\neq \phi) \subseteq \mathcal{L}(T_i)$ .We say that (S, D) is a *"Detector Pair"* in $\mathcal{L}(T_i)$ if the following are satisfied for all $l_{k+1} \in D$:

- $\{l_1, \dots, l_k, l_{k+1}\}$ is an LI set. Let $\mathcal{F} = fl(\{l_1, \dots, l_k, l_{k+1}\})$. $\mathcal{F}$ is elementary in $\mathcal{L}(T_i)$ i.e. $\mathcal{F} \cap \mathcal{L}(T_i) = \{l_1, \dots, l_k, l_{k+1}\}$. See Definition 2.8.

- $\mathcal{F} \cap \mathcal{L}(T_{1-i}) \subseteq fl(\{l_1, \dots, l_k\})$ i.e. $\mathcal{F}$ contains only those points from $\mathcal{L}(T_{1-i})$ which lie inside $fl(\{l_1, \dots, l_k\})$.

3.5.1 Identifying Some Factors Which Don't Divide g

The two claims below give results about structure of linear forms which divide g. The proofs are easy but technical and so we move them to the appendix.

Claim 3.7. *Let $(S = \{l_1 \dots, l_k\}, D)$ be a Detector set in $\mathcal{L}(T_i)$. Let $l_{k+1} \in D$. For a $standard$ linear form $l \in V$, if $l \mid g$ then $l \notin sp(\{l_1, \dots, l_k\})$.*

Proof. See B.1 in Appendix ☐

Claim 3.8. *Let $l \in Lin_{\mathbb{R}}[\bar{x}]$ be $standard$ such that $l \mid g$ and $\mathcal{C}$ be the candidate set. Assume $(S = \{l_1, \dots, l_k\}, D(\neq \phi))$ is a Detector pair in $\mathcal{L}(T_i)$. Then $|\mathcal{L}(T_{1-i}) \cap (fl(S \cup \{l\}) \setminus fl(S))| \geq 2$. That is the flat $fl(\{l_1, \dots, l_k, l\})$ contains atleast two distinct points from $\mathcal{L}(T_{1-i})(\subseteq \mathcal{C})$ outside $fl(\{l_1, \dots, l_k\})$.*

Proof. See B.2 in Appendix ☐

Claim 3.9. *Suppose $(S = \{l_1, \dots, l_k\}, D(\neq \phi))$ is a Detector in $\mathcal{L}(T_i)$. The following algorithm identifies some factors in $\mathcal{L}(G) \setminus \mathcal{L}(g)$. It returns the product of all linear forms identified.*

Algorithm 5 Identifies linear forms dividing $\mathcal{L}(G)$ but not $\mathcal{L}(g)$

procedure $IdentifyFactors(f \in \Sigma\Pi\Sigma_{\mathbb{R}}(2)[\bar{x}], \mathcal{C} \subset Lin_{\mathbb{R}}[\bar{x}], S = \{l_1, \dots, l_k\} \subset Lin_{\mathbb{R}}[\bar{x}])$
    I $= 1$, *bool* $flag$;
    for (each factor l of f) **do**
        $flag = false$;
        if $(l, l_1, \dots, l_k$ are LI) **then**
            for ($l'_1 \neq l'_2 \in \mathcal{C} \setminus fl(\{l_1, \dots, l_k\})$) **do**
                if $(l'_1, l'_2 \in sp(\{l, l_1, \dots, l_k\}))$ **then**
                    $flag = true$;   *//This factor should not identified*
                    break();
        if $(!flag)$ **then**
            I = I $\times l$ *// identified $l \in \mathcal{L}(G) \setminus \mathcal{L}(g)$*
    return I;

Proof. The proof of the claim is a part of Lemma 3.10 below. ☐

Time Complexity - Since $\mathcal{C}$ has size $poly(d)$ and $deg(f) = d$, the nested loops run $poly(d)$ times. k, r are constants so checking linear independence of $k+1$ linear forms in r variables takes constant time. Checking if some vectors belong to a $k + 1$ dimensional space also takes constant time. Multiplying linear forms to **I** takes $poly(d)$ time. So overall the algorithm runs in $poly(d)$ time.

So the above algorithm identified a factor **I** of G for us. Let us define new polynomials

$$G^{\star} = \frac{G}{\mathbf{I}} = \prod_{t \in [N_1]} G_t$$

and

$$f^{\star} = \frac{f}{\mathbf{I}} = G^{\star}(\alpha_0 T_0 + \alpha_1 T_1)$$

Lemma 3.10. *The following are true:*

1. *If $l \mid I$ (i.e. l was identified) then $l \in \mathcal{L}(G) \setminus \mathcal{L}(g)$.*

2. *If $l \mid G^\star$ (i.e. l was retained) then $(fl(\{l_1, \ldots, l_k, l\}) \setminus fl(\{l_1, \ldots, l_k\})) \cap (\mathcal{L}(T_{1-i}) \cup (\mathcal{L}(T_i) \setminus D)) \neq \phi$ that is:*

   $(fl(\{l_1, \ldots, l_k, l\}) \setminus fl(\{l_1, \ldots, l_k\}))$ contains a point from $\mathcal{L}(T_i) \setminus D$ or $\mathcal{L}(T_{1-i})$.

3. *If $l \mid G^\star$ and $l_{k+1} \in D$ then $l \notin sp(\{l_1, \ldots, l_k, l_{k+1}\})$.*

Proof. See B.3 in Appendix. □

3.5.2 Overestimating the set D of the detector pair (S, D)

Lemma 3.10 is going to help us actually find an overestimate of D corresponding to $S = \{l_1, \ldots, l_k\}$ in the detector pair (S, D) as described in the lemma below. This will be important since we need D during our algorithm for the Hard Case.

Lemma 3.11. *Let $(S = \{l_1, \ldots, l_k\}, D)$ be a detector in $\mathcal{L}(T_i)$. For each $(l, l_j) \in \mathcal{C} \times S$ define the space $U_{\{l,l_j\}} = sp(\{l, l_j\})$. Extend $\{l, l_j\}$ to a basis and in the process obtain $U'_{\{l,l_j\}}$ such that $V = U_{\{l,l_j\}} \oplus U'_{\{l,l_j\}}$. Define the set:*

$$X = \{l \in \mathcal{C} : \pi_{U'_{\{l,l_j\}}}(f^\star) \neq 0, \textit{ for all } l_j \in S\}$$

Then $D \subset X \subset \mathcal{L}(T_i)$.

Proof. See B.4 in Appendix. □

This set X is an overestimate of D inside $\mathcal{L}(T_i)$ and also easy to compute. Given S we may easily construct X in time $poly(d)$ because of it's simple description. Let's give an algorithm to compute X given $f^\star, S, \mathcal{C}$.

Algorithm 6 Overestimate the Detector (Returns a set of linear forms)

procedure $overest\_detector(f^\star \in \Sigma\Pi\Sigma_{\mathbb{R}}(2)[\bar{x}], S = \{l_1, \ldots, l_k\} \subset Lin_{\mathbb{R}}[\bar{x}], \mathcal{C} \subset Lin_{\mathbb{R}}[\bar{x}])$
    $bool\ flag$;
    Define $X = \phi$;
    for (each $l \in \mathcal{C}$) **do**   *// Searching for linear forms in D*
        $flag = true$;
        for (each $l_j \in S$) **do**
            Find $\{l'_1, \ldots, l'_{r-2}\} \subset \mathcal{C}$ such that $\{l, l_j, l'_1, \ldots, l'_{r-2}\}$ is LI
            $U = \mathbb{R}l \oplus \mathbb{R}l_j; U' = \mathbb{R}l'_1 \oplus \ldots \oplus \mathbb{R}l'_{r-2}$;   *// Spaces for projection*
            if ($\pi_{U'}(f^\star) = 0$) **then**   *// l is not in D*
                $flag = false$;
        if $(flag)$ **then**
            $X = X \cup \{l\}$;
    return X;

Time Complexity - Inside the inner for loop we look for $(r-2)$ linear forms from $\mathcal{C}$. $|\mathcal{C}| = poly(d)$ and r is a constant and so this step only needs $poly(d)$ time. The nested loops run polynomially many times. Checking linear independece of r linear forms and projecting to known constant dimensional subspaces also take $poly(d)$ time as has been discussed before. So the algorithm runs in $poly(d)$ time.

3.6 Hard Case : $\boxed{\mathcal{L}(T_{1-i}) \subseteq sp(U_i), \textbf{ for } i = 0 \textbf{ and } 1}$

This Subsection will involve the most non trivial ideas. We handled the case $dim(^{sp(T_{1-i})\,+\,sp(T_i)}/_{sp(T_i)}) \geq 2$ completely in Easy Case 1 and 2 (See Sections 3.3 and 3.4), so let's assume $dim(^{sp(T_{1-i})\,+\,sp(T_i)}/_{sp(T_i)}) \leq 1 \Rightarrow dim(\mathcal{L}(T_{1-i}) \cup \mathcal{L}(T_i)) \leq dim(\mathcal{L}(T_i)) + 1$ for both $i = 0, 1$. We already know that $rank(f) = r$, implying $dim(\mathcal{L}(T_i) \cup \mathcal{L}(T_{1-i})) = r$. Thus for $i = 0, 1$; $dim(\mathcal{L}(T_i)) \geq r - 1$. This works in our favour for applying the quantitative version of the Sylvester Gallai theorems given in [BDWY11]. To be precise we will use Corollary 2.13 from Section 2.3 in this paper.

1. Our first application (See Lemma 3.13) of Quantitative Sylvester Gallai will help us prove the existence of a Detector pair $(S = \{l_1, \ldots, l_k\}, D)$ in $\mathcal{L}(T_i)$ with $k = c_{\mathbb{R}}(3) + 2$ (See defn of $c_{\mathbb{R}}(.)$ in Theorem 1.6) and large size of D. For this we will only need $dim(\mathcal{L}(T_i)) \geq C_{2k-1}$ for $i = 0, 1$(See Section 2.3 for definition of C_{2k-1}). So we can go back and fix our r to be $\geq C_{2k-1} + 1$ where $k = c_{\mathbb{R}}(3) + 2$.
2. The above point shows the existence of a detector pair (S, D) in $\mathcal{L}(T_i)$ with large $|D|$. So now we go back to Subsection 3.5 and remove some factors of f to get $f^\star = G^\star(\alpha_0 T_0 + \alpha_1 T_1)$ such that linear factors of $G^\star$ satisfy properties given in Lemma 3.10. We also compute the overestimate X of D using Algorithm 3.5.2. Let the known and unknown parts of $f^\star$ be $K_0^\star, K_1^\star$ and $U_0^\star, U_1^\star$ respectively. If for some $i \in \{0, 1\}$, $\mathcal{L}(T_i) \subsetneq sp(U_{1-i}^\star)$ then we are in Easy Case 2 for $f^\star$ and can recover the gates for $f^\star$. Otherwise for both $i = 0, 1$; $\mathcal{L}(T_i) \subseteq sp(U_{1-i}^\star) \Rightarrow dim(\mathcal{L}(U_{1-i}^\star)) \geq r - 1$ and we continue with reconstruction below.
3. Next to actually reconstruct linear forms in $U_{1-i}^\star$, we will use it's high-dimensionality ($\geq r - 1 \geq C_{2k-1}$) discussed above. Corollary 2.13 from Section 2.3 will enable us to prove the existence of a $d_1 \in D$ which together with the set S found above will give the existence of a *"Reconstructor"*(See Claim 2.19 and Algorithm 1) which recovers some linear factors of $U_{1-i}^\star$ with multiplicity (See Theorem 3.16) .

3.6.1 Large Size of Detector Sets

w.l.o.g. we assume $|\mathcal{L}(T_0)| \leq |\mathcal{L}(T_1)|$. First we point out a simple calculation that will be needed later. For $\delta \in (0, \frac{7-\sqrt{37}}{6})$ and $\theta \in (\frac{3\delta}{1-\delta}, 1 - 3\delta)$, let $v(\delta, \theta)$ be defined as follows:

$$v(\delta, \theta) = \begin{cases} 1 - \delta - \theta & \text{if } |\mathcal{L}(T_0)| \leq \theta |\mathcal{L}(T_1)| \\ (1 - \delta)(1 + \theta) - 1 & \text{if } \theta |\mathcal{L}(T_1)| < |\mathcal{L}(T_0)| \leq |\mathcal{L}(T_1)| \end{cases}$$

Claim 3.12. *The following is true*

$$\frac{(2 - v(\delta, \theta))}{v(\delta, \theta)} \leq \frac{1 - \delta}{\delta}$$

Proof. See C.1 in Appendix. □

Lemma 3.13. *Let $k = c_{\mathbb{R}}(3) + 2$ (see defn of $c_{\mathbb{R}}(m)$ in Theorem 1.6). Fix δ, θ in range given in Claim 3.12 above . Then for some $i \in \{0, 1\}$ there exists a Detector $(S = \{l_1, \ldots, l_k\}, D)$ in $\mathcal{L}(T_i)$ with $|D| \geq v(\delta, \theta) \max(|\mathcal{L}(T_0)|, |\mathcal{L}(T_1)|)$.*

Proof. See C.2 in Appendix. □

3.6.2 Assuming $\mathcal{L}(T_i) \subseteq sp(\mathcal{L}(U^\star_{1-i}))$ and reconstructing factors of $U^\star_{1-i}$

Consider the set of linear forms (points) $\mathcal{X} = \mathcal{L}(G^\star) \cup \mathcal{L}(T_0) \cup \mathcal{L}(T_1)$. We know that $sp(\mathcal{X}) = V = Lin_{\mathbb{R}}[\bar{x}] \simeq \mathbb{R}^r$ (By abuse of notation we will use linear forms as points in $\mathbb{R}^r$ wherever required). Let $(S_0 = \{l_1, \ldots, l_k\}, D)$ be a detector in $\mathcal{L}(T_i)$ with $|D| \geq v(\delta, \theta) \max(|\mathcal{L}(T_0)|, |\mathcal{L}(T_1)|)$ and $W_0 = sp(S_0)$. Extend S_0 to a basis $\{l_1, \ldots, l_k, l'_{k+1}, \ldots, l'_r\}$. Now it's time to use the other random matrix Λ. Since we had applied Ω in the beginning, $\{\Omega^{-1}(l_1), \ldots, \Omega^{-1}(l_k)\}$ are linear forms in our input polynomial for this section. By **Assumption 3** we know that the set $\{\Omega(\Omega^{-1}l_1), \ldots, \Omega(\Omega^{-1}l_k), \Lambda\Omega(\Omega^{-1}l'_{k+1}), \ldots, \Lambda\Omega(\Omega^{-1}l'_r)\}$ is LI. Let $l_j = \Lambda l'_j, j \in \{k+1, \ldots, r\}$. So $\mathcal{B} = \{l_1, \ldots, l_r\}$ is a basis. and define $\tilde{W_0} = sp(\{l_{k+1}, \ldots, l_r\})$. Clearly $V = W_0 \oplus \tilde{W_0}$. Also by **Assumption 4** for any $l \in \mathcal{X} \setminus W_0$, $[l]^{k+1}_{\mathcal{B}} \neq 0$. We define a normalization for linear forms $l \in \mathcal{X}$:

$$\widehat{l} = \begin{cases} \frac{1}{[l]^{k+1}_{\mathcal{B}}} l & : l \in W_0^c \cap \mathcal{X} \\ 0 & : l \in W_0 \cap \mathcal{X} \end{cases}$$

i.e. normalize the $(k+1)^{th}$ co-ordinate w.r.t. the basis $\mathcal{B}$. For any subset $\mathcal{T} \subset \mathcal{X}$, we define :

$$\widehat{\mathcal{T}} = \{\widehat{l} : l \in \mathcal{T}\} \setminus \{0\}$$

With this notation we proceed towards detecting linear factors of the unknown parts. But first let's show that even after projecting onto $\tilde{W_0}$, the detector is larger in size (upto a function of δ) compared to one of the unknown parts.

Lemma 3.14. *The following are true:*

1. $dim(\pi_{\tilde{W_0}}(\widehat{\mathcal{L}(U^\star_{1-i})})) > C_4$
2. $\pi_{\tilde{W_0}}(\widehat{\mathcal{L}(U^\star_{1-i})}) \cap \pi_{\tilde{W_0}}(\widehat{D}) = \phi$
3. $|\pi_{\tilde{W_0}}(\widehat{\mathcal{L}(U^\star_{1-i})})| \leq \frac{1-\delta}{\delta} |\pi_{\tilde{W_0}}(\widehat{D})|$

Proof. See C.3 Appendix. □

This Lemma enables us to apply Lemma 2.13 from Section 2.3. Consider the sets $\pi_{\tilde{W_0}}(\widehat{\mathcal{L}(U^\star_{1-i})})$ and $\pi_{\tilde{W_0}}(\widehat{D})$. We've shown above that they are disjoint, span high enough dimension and

$$|\pi_{\tilde{W_0}}(\widehat{\mathcal{L}(U^\star_{1-i})})| \leq \frac{1-\delta}{\delta} |\pi_{\tilde{W_0}}(\widehat{D})|$$

Lemma 2.13 shows the existence of a line $\vec{L}_1$ (called a *"semiordinary bichromatic"* line) in $\tilde{W_0}$ such that $|\vec{L}_1 \cap \pi_{\tilde{W_0}}(\widehat{\mathcal{L}(U^\star_{1-i})})| = 1$ and $|\vec{L}_1 \cap \pi_{\tilde{W_0}}(\widehat{D})| \geq 1$.

For technical reasons we need a different *"semiordinary bichromatic"* line. We construct it here:

1. Pick a $d_1 \in D$ such that $e = \pi_{\tilde{W}_0}(\hat{d}_1) \in \vec{L}_1$. Clearly $e \in sp(\{l_1, \ldots, l_k, d_1\})$. Observe $[d_1]_{\mathcal{B}}^{k+1} \neq 0 \Rightarrow [e]_{\mathcal{B}}^{k+1} \neq 0$, further implying that $\mathcal{B}_1 = \{l_1, \ldots, l_k, e, l_{k+2}, \ldots, l_r\}$ and $\mathcal{B}_2 = \{l_1, \ldots, l_k, d_1, l_{k+2}, \ldots, l_r\}$ are bases.

2. For $v \in V$, denote by $[v]_{\mathcal{B}_2}^{d_1}$ the coefficient of d_1 when v is written in basis $\mathcal{B}_2$. We know that for $v \in \mathcal{X}$, $[v]_{\mathcal{B}}^{k+1} \neq 0$, this clearly implies that $[v]_{\mathcal{B}_2}^{d_1} \neq 0$. We define another normalization for linear forms $l \in \mathcal{X}$:

$$\widetilde{l} = \begin{cases} \frac{1}{[l]_{\mathcal{B}_2}^{d_1}} l & : l \notin W_0 \cap \mathcal{X} \\ 0 & : l \in W_0 \cap \mathcal{X} \end{cases}$$

i.e. normalize the coefficient of d_1 when l is written in basis $\mathcal{B}_2$. For any subset $\mathcal{T} \subset \mathcal{X}$, we define :

$$\widetilde{\mathcal{T}} = \{\widetilde{l} : l \in \mathcal{T}\} \setminus \{0\}$$

This leads us to the following lemma :

Lemma 3.15. *Let $S_1 = \{d_1\}$ and $S_2 = \{l_{k+2}, \ldots, l_r\}$, $W_1 = sp(S_1)$ and $W_2 = sp(S_2)$. So $V = W_0 \oplus W_1 \oplus W_2$ and let $W_0' = W_1 \oplus W_2$. For $u \in \mathcal{L}(U_{1-i}^\star)$ such that $\pi_{\tilde{W}_0}(\widehat{u}) \in \vec{L}_1 \cap \pi_{\tilde{W}_0}(\widehat{\mathcal{L}(U_{1-i}^\star)})$ consider the following line inside W_0'*

$$\vec{L}_2 = fl(\{d_1, \pi_{W_0'}(\widetilde{u})\})$$

then $|\vec{L}_2 \cap \pi_{W_0'}(\widetilde{D})| \geq 1$ and $|\vec{L}_2 \cap \pi_{W_0'}(\widetilde{\mathcal{L}(U_{1-i}^\star)})| = 1$, i.e. $\vec{L}_2$ is also a "semiordinary bichromatic" *like $\vec{L}_1$.*

Proof. See C.4 in Appendix. □

Finally it's time to give the main theorem for this subsection which helps us design an algorithm for the Hard Case.

Theorem 3.16. *There exist pairwise disjoint LI sets S_0, S_1, S_2 with $S_0 \cup S_1 \cup S_2$ being a basis, and non constant polynomials P, Q dividing $U_{1-i}^\star$ such that $P \mid Q$ and (Q, P, S_0, S_1, S_2) is a Reconstructor.*

Proof. We do this in steps:

- Let S_0, S_1, S_2 be as defined in the discussion above.
- Let Q be the largest factor of $U_{1-i}^\star$ such that for all linear forms $q \mid Q$, $\pi_{W_2}(q) \neq 0$. So $\pi_{W_2}(Q) \neq 0$ and if $u^\star \mid \frac{U_{1-i}^\star}{Q}$ is a linear form then $\pi_{W_2}(u^\star) = 0$. Let P be the $\Pi\Sigma$ polynomial with the largest possible degree such that for all linear factors p of P, $\pi_{W_0'}(\widetilde{p}) = \pi_{W_0'}(\widetilde{u})$ (which was a non zero vector on $\vec{L}_2$). Since $\pi_{W_0'}(\widetilde{u})$ and $\pi_{W_0'}(\widetilde{d}_1)$ were LI this also means that $\pi_{W_2}(u) \neq 0 \Rightarrow \pi_{W_2}(p) \neq 0$ for all $p \mid P$. Clearly P is non constant since $u \mid P$, also by definition $P \mid Q$. Then (Q, P, S_0, S_1, S_2) is a *Reconstructor* (See Subsection 2.5 for definition) for P. Let's check this is true:
  - $\pi_{W_2}(Q) \neq 0$ - By definition of Q we know this for all it's factors and therefore for Q itself.
  - $\pi_{W_0'}(P) = \delta(\pi_{W_0'}(\widetilde{u}))^t$, for some $\delta \in \mathbb{R}$ (by definition of P).

– Let $q \mid \frac{Q}{P}$ such that $gcd(\pi_{W_2}(P), \pi_{W_2}(q)) \neq 1 \Rightarrow$ there exists some linear factor $p \mid P$ such that $\pi_{W_2}(p), \pi_{W_2}(q)$ are LD. $\{\pi_{W_2}(p), \pi_{W_2}(q)\}$ are LD and non-zero implies $q \in sp(\{l_1, \ldots, l_k, d_1, p\})$.

$$\Rightarrow \pi_{W_0'}(q) \in sp(\{\pi_{W_0'}(d_1), \pi_{W_0'}(p)\}) = sp(\{d_1, \pi_{W_0'}(\widetilde{u})\})$$

So clearly :

$$\pi_{W_0'}(\widetilde{q}) \in sp(\{d_1, \pi_{W_0'}(\widetilde{u})\})$$

Since coefficient of d_1 in $\pi_{W_0'}(\widetilde{q}), d_1$, and $\pi_{W_0'}(\widetilde{u})$ is 1, it's easy to see that $\pi_{W_0'}(\widetilde{q}) \in fl(\{d_1, \pi_{W_0'}(\widetilde{u})\}) = \vec{L}_2$. Since $Q \mid U^\star_{1-i}$ we have $\pi_{W_0'}(\widetilde{q}) \in \pi_{W_0'}(\widetilde{\mathcal{L}(U^\star_{1-i})}) \Rightarrow \pi_{W_0'}(\widetilde{q}) \in \vec{L}_2 \cap \pi_{W_0'}(\widetilde{\mathcal{L}(U^\star_{1-i})}) = \{\pi_{W_0'}(\widetilde{u})\}$. So $\pi_{W_0'}(\widetilde{q}) = \pi_{W_0'}(\widetilde{u})$ which can't be true since P is the largest polynomial dividing Q where linear factors have this property and $q \nmid P$. So such a q does not exist.

$\square$

Corollary 3.17. *Using $f, K_{1-i}, S_0, S_1, S_2$ from above we can compute $\pi_{W_0'}(Q), \pi_{W_1'}(Q)$ for Q defined in the proof above.*

Proof. See the following algorithm. The $r \times r$ matrix Λ with enries picked independently (and independent of entries of Ω) from the uniform distribution on $[N]$ is also sent as an input. Fix $k = c_{\mathbb{R}}(3) + 2$.

$\square$

Algorithm 7 Hard Case - $\mathcal{L}(T_i) \subseteq \mathcal{L}(U_{1-i})$ for $i = 0, 1$

procedure $bool\ recon\_even(f \in \Sigma\Pi\Sigma_{\mathbb{R}}(2)[\bar{x}], \mathcal{C} \subset Lin_{\mathbb{R}}[\bar{x}], \Lambda \in \mathbb{R}^{r \times r})$

i) For each $i \in \{0, 1\}$ (guessing the gate which contains detector pair with large $|D|$) do the following

a) Iterate over all choices of LI vectors $\mathcal{B}' = \{l_1, \dots, l_k, l'_{k+1}, \dots, l'_r\} \subset \mathcal{C}$.

1) Let $S_0 = \{l_1, \dots, l_k\}$ (guess for the S in the good detector pair) and apply Λ to the last $r - k$ vectors to get $l_j = \Lambda(l'_j)$ for $j \in \{k+1, \dots, r\}$.

2) Check if $\mathcal{B} = \{l_1, \dots, l_r\}$ is a basis, if not **reject this $\mathcal{B}'$ and go to next (or next i if $\mathcal{B}'$ is not available anymore)**

3) Compute the $\Pi\Sigma$ polynomial $I = IdentifyFactors(f, \mathcal{C}, S_0)$. If $I \mid f$ let $f^\star = \frac{f}{I}$ and define known parts of gates $K_0^\star = 1, K_1^\star = 1$. Else **reject this $\mathcal{B}'$ and go to next (or next i if $\mathcal{B}'$ is not available anymore)**

4) Compute an overestimate X of D in $\mathcal{L}(T_i)$ using $X = overest\_detector(f^\star, \mathcal{C}, S_0)$.

5) **while** $(deg(K_{1-i}^\star) < deg(f^\star))$ do the following -

(i) Check whether we've landed in Easy Case 2. Invoke $recon\_uneven(f^\star, K_0^\star, K_1^\star, \mathcal{C})$. If this returns true with $\Pi\Sigma$ gates A, B, check if $f = IA + IB$ and return true with gates IA, IB. If it returned false or $f \neq IA + IB$, continue reconstruction.

(ii) Iterate over all $d_1 \in X$ and do the following:

(a) If $\mathcal{B}_2 = \{l_1, \dots, l_k, d_1, l_{k+2}, \dots, l_r\}$ is not an LI set **reject this d_1 and go to next d_1 in loop**.

(b) To take projections define the following spaces : $V_j = \mathbb{R}l_j, j \in [r] \setminus \{k+1\}, V_{k+1} = \mathbb{R}d_1 \Rightarrow V = \bigoplus_{j=1}^{r} V_j$. Define $V'_j = \bigoplus_{t \in [r] \setminus \{j\}} V_t$. Also define $S_0 = \{l_1, \dots, l_k\}, S_1 = \{u_{k+1}\}, S_2 = \{l_{k+2}, \dots, l_r\}, W_j = sp(S_j), W'_j = \bigoplus_{j_1 \neq j} W_{j_1}$ for $j \in \{0, 1, 2\}$.

(c) Consider the largest $Q \mid U_{1-i}^\star$ such that for all $q \mid Q, q \notin W'_2 = W_0 \oplus W_1$(defined above). Next we try to compute $\pi_{W'_0}(Q), \pi_{W'_1}(Q)$.

(d) To compute $\pi_{W'_0}(Q)$ first compute $Q_0 = \frac{\pi_{V'_1}(f^\star)}{\pi_{V'_1}(K_{1-i}^\star)}$. If Q_0 is a non-zero $\Pi\Sigma$ polynomial continue else **reject d_1 and go to next in loop**. For each linear form $q_0 \mid Q_0$ if $q_0 \in W'_2$ then $Q_0 = \frac{Q_0}{q_0}$. This removes some projections and gives $Q_0 = \pi_{V'_1}(Q)$. Compute $Q_0 = \pi_{W'_0}(Q_0)$. If Q_0 is a non-zero $\Pi\Sigma$ polynomial continue else **reject this d_1 and go to next d_1 in the loop**. Since $V_1 \subset W_0$ and $W'_0 \subset V'_1$, $\pi_{W'_0}(Q) = \pi_{W'_0}(\pi_{V'_1}(Q)) = Q_0$.

(e) Compute $Q_1 = \frac{\pi_{W'_1}(f)}{\pi_{W'_1}(K_{1-i}^\star)}$. If Q_1 is a non-zero $\Pi\Sigma$ polynomial continue else **reject this d_1 and go to next d_1 in the loop**. Again remove projections i.e. for linear form $q_1 \mid Q_1$ if $q_1 \in W'_2$ then $Q_1 = \frac{Q_1}{q_1}$. This process makes sure that $Q_1 = \pi_{W'_1}(Q)$.

(f) If $Reconstructor(Q_0, Q_1, S_0, S_1, S_2)$ returns a non-trivial $\Pi\Sigma$ polynomial update $K_{1-i}^\star = K_{1-i}^\star \times Reconstructor(Q_0, Q_1, S_0, S_1, S_2)$. Also update $count = count + deg(Reconstructor(Q_0, Q_1, S_0, S_1, S_2))$. Else **reject this d_1 and go to next d_1 in the loop**.

6) Define $K_{1-i} = IK_{1-i}^\star$. Factor $K_i = f - K_{1-i}$ and if it is a $\Pi\Sigma$ polynomial return gates as K_0, K_1 and return true.

ii) Outside all the loops return false.

Correctness

1. **If we return true with gates A, B :** then we ought to be right since we check if $f = A + B$. Since the representation is unique this will be the correct answer.

2. **If we return false:** Let's assume f actually has a $\Sigma\Pi\Sigma(2)$ representation. If we were in Easy Case 1 or 2 we would have already found the circuit using their algorithms. So we are in the Hard Case. So by Lemma 3.13 there exists i such that $\mathcal{L}(T_i)$ has a detector pair (S_0, D) with $|D|$ large. For this i there exists such an S_0, so sometime during the algorithm we would have guessed the correct i and the correct S_0. **Now let's analyze what happens inside the while and the third for loop when the first two guesses are correct.** Note that this also implies that the I we have identified is correct and now we need to solve for

$$f^\star = G^\star(\alpha_0 T_0 + \alpha_1 T_1)$$

Let $K_0^\star, K_1^\star$ (initialized to 1) be the known parts of the gates for this polynomial $f^\star$ and $U_0^\star, U_1^\star$ be the unknown parts. Note that T_0, T_1 are same for both polynomials so $rank(f^\star) = rank(f)$ and for $j = 0, 1$; $K_j \mid G^\star T_j$.

Assume till the m^{th} iteration of the while loop $K_t^\star \mid G^\star T_t$ for $t \in \{0, 1\}$, we show that after the $(m+1)^{th}$ iteration, this property continues to hold and $deg(K_{1-i}^\star)$ increases.

- If after the m^{th} iteration of the while loop for some $j \in \{0, 1\}$, $\mathcal{L}(T_j) \subsetneq sp(\mathcal{L}(U_{1-j}^\star))$ we are in Easy Case 2 for $f^\star$. The first step in while loop is to call $recon\_uneven(f^\star, \mathcal{C}, K_0^\star, K_1^\star)$. This will clearly recover the circuit for $f^\star$ and return true since $K_t^\star \mid G^\star T_t$ for $t \in \{0, 1\}$. However this does not happen so for both $j = 0, 1$, we have $\mathcal{L}(T_i) \subsetneq \mathcal{L}(U_{1-i}^\star)$. This means that we can use the ideas in Subsection 3.6.2, specifically Theorem 3.16.
- The first two guesses are correct imply that $D \subseteq X \subseteq \mathcal{L}(T_i)$.
- If d gets rejected then $K_t, t \in \{0, 1\}$ remain unchanged. If some d_1 does not get rejected then since $d_1 \in \mathcal{L}(T_i)$, $Q_0 = \pi_{V_1'}(U_{1-i}^\star)$ is a non zero $\Pi\Sigma$ polynomial. Then some factors (the ones $\in W_2'$) are removed from Q_0. Also on projecting to W_0' this still remains non-zero (as d_1 was not rejected).
- We know that $d_1 \in \mathcal{L}(T_i)$ and d_1 not getting rejected implies that $Q_1 = \pi_{W_1'}(U_{1-i}^\star)$ is a non-zero $\Pi\Sigma$ polynomial. We again remove some factors (i.e. the ones in W_2') from Q_1. The non-zeroness of Q_0, Q_1 imply that $Q_0 = \pi_{W_1'}(Q), Q_1 = \pi_{W_1'}(Q)$ i.e. they are projections of the same polynomial Q which is the largest factor of $U_{1-i}^\star$ with the property that any linear form $q \mid Q$ is not in $W_2' = W_0 \oplus W_1$.
- d_1 was not rejected implies that $Reconstructor(Q_0, Q_1, S_0, S_1, S_2)$ returned a non-trivial polynomial P. This has to be a factor of Q by Claim 2.21 following Algorithm 1 and therefore a factor of $U_{1-i}^\star$.
- Proof of Theorem 3.16 implies that in every iteration atleast some d_1 will not be rejected.
- So clearly the new $K_{1-i}^\star = K_{1-i}^\star \times P$ divides $G^\star T_{1-i}$. K_i remains unchanged. Therefore even after the $(m+1)^{th}$ iteration $K_t \mid G^\star T_t$ for both $j = 0, 1$ but degree of $K_{1-i}^\star$ increases.

So the while loop cannot run more than $deg(f^\star)$ times and in the end $G^\star T_{1-i}$ will be reconstructed completely and correctly and we should have returned true. Therefore we have a contradiction and so f did not have a $\Sigma\Pi\Sigma(2)$ circuit and we correctly returned false.

Running Time

- First for loop runs twice.
- Inside it chossing r linear forms from $\mathcal{C}$ ($|\mathcal{C}| = poly(d)$) takes $poly(d)$ time.
- Applying Λ to $r - k$ vectors takes $poly(r) = O(1)$ time.
- Checking if a set of size r inside $\mathbb{R}^r$ is LI takes $poly(r) = O(1)$ time since it is equivalent to computing determinant.
- $IdentifyFactors()$ takes $poly(d)$ time and computing $f^\star$ also takes $poly(d)$ time.
- $overest\_detector()$ runs in $poly(d)$ time.
- while loop runs atmost d times
- $recon\_uneven()$ runs in $poly(d)$ time and so does polynomial multiplication.
- $X \subseteq \mathcal{L}(T_i)$ and $|\mathcal{L}(T_i)| \leq deg(f^\star)$ and so for loop runs d times.
- Change of bases in $\mathbb{R}^r$ and application to a polynomial of degree d takes $poly(d)$ time.
- Therefore projecting to subspaces also takes $poly(d)$ time.
- $Reconstructor()$ runs in $poly(d)$ time (since r is a constant) and so does polynomial multiplication and factoring by [KT90].

Since all of the above steps run in $poly(d)$ time, nesting them a constant number of times also takes $poly(d)$ time. Therefore the running time of our algorithm is $poly(d)$.

3.7 Algorithm including all cases :

The algorithm we give here will be the final algorithm for rank r $\Sigma\Pi\Sigma$ polynomials. It will use the previous three cases. Our input will be a $\Sigma\Pi\Sigma(2)$ polynomial $f(x_1, \ldots, x_r)$ and output will be a circuit computing the same.

Algorithm 8 Reconstruction of rank r polynomials

1: **procedure** $void\ lowdim\_reconstruct(f \in \Sigma\Pi\Sigma_{\mathbb{R}}(2)[\bar{x}])$
2: Pick $(\Omega_{i,j}), (\Lambda_{i,j})$, $r \times r$ matrices with entries chosen uniformly randomly from $[N]$. Make them visible to all functions.
3: Consider the linear forms $L_i(\bar{x}) = \sum\limits_{k=1}^{r} \Omega_{i,k} x_k$ and redefine $f(x_1, \ldots, x_r) = f(L_1(\bar{x}), \ldots, L_r(\bar{x}))$.
4: $\mathcal{C} = Candidates(f(x_1, \ldots, x_r))$; *//Compute the set of candidate linear forms $\mathcal{C}$.*
5: **if** $(gatespan\_uneven(f, \mathcal{C}))$ **then** *//Assuming Easy Case 1 where $\mathcal{L}(g) = \phi$*
6: **else if** $(recon\_uneven(f, K_0, K_1, \mathcal{C}))$ **then** *//Assuming Easy Case 2 where some gate has extra dimensions*
7: **else** $(recon\_even(f, \mathcal{C}, \Lambda))$

Explanation : Here we explain every step of the given algorithm:

- The function $reconstruct(f)$ takes as input a polynomial $f \in \Sigma\Pi\Sigma_{\mathbb{R}}(2)[\bar{x}]$ of $rank = r$ and outputs two polynomials $K_0, K_1 \in \Pi\Sigma_{\mathbb{R}}[\bar{x}]$ which are the two gates in it's circuit representation.
- Steps $2, 3$ picks a random matrix M and transforms each variable using the linear transformation this matrix defines. With high probability this will be an invertible transformation. We do the reconstruction for this new polynomial since the linear factors of it's gates satisfy some non-degenerate conditions(because they have been randomly transformed) our algorithm needs. We apply M^{-1} after the reconstruction and get back our original f.
- The next step constructs the set of candidate linear forms $\mathcal{C}$. We've talked about the size, construction and structure of this set in Section 3.2.
- We first assume Easy Case 1. It that was not the case we check for Easy Case 2. If both did not occur we can be sure we are in the Hard case.
- If none of the called functions gave true we can be sure that f did not have a $\Sigma\Pi\Sigma_{\mathbb{R}}(2)[\bar{x}]$ representation.

4 Reconstruction for arbitrary $rank$

This section reduces the problem from $\Sigma\Pi\Sigma(2)$ Circuits with arbitrary rank $n\ (> r)$ to one with constant rank $(= r)$. Also once the problem has been solved efficiently in the low rank case we use multiple instances of such solutions to lift to the general $\Sigma\Pi\Sigma(2)$ circuit. The idea is to project the polynomial to a small (polynomial) number of random subspaces of dimension r, reconstruct these low rank polynomials and then lift back to the original polynomial. The uniqueness of our circuit's representation plays a major role in both the projection and lifting steps. Let

$$f = G(\alpha_0 T_0 + \alpha_1 T_1)$$

G, T_i are normal $\Pi\Sigma$ polynomials. All notations are borrowed from the previous section. It is almost identical to the restriction done in [Shp07] except that the dimension of random subspaces is different. For more details see Section 4.2.1 and 4.2.3. in [Shp07]. Since all proofs have been done in detail in [Shp07] we do not spend much time here. A clear sketch with some proofs is however given.

4.1 Projection to a Random Low Dimensional Subspace

We explain the procedure of projecting to the random subspace below. In this low dimensional setup we can get white-box access to $\pi_V(f)$, also some important properties of f are retained by $\pi_V(f)$. Proofs are simple and standard so we discuss them in the appendix at end.

Pick n vectors $v_i, i \in [n]$ with each co-ordinate chosen independently from the uniform distribution on $[N]$. Let $V = sp(\{v_i : i \in [r]\})$ and $V' = sp\{v_i : i \in \{r+1, \dots, n\}\}$. Then $V \oplus V' = \mathbb{R}^n$ Let π_V denote the orthogonal projection onto V. With high probability the following hold :

1. This set $\{v_i : i \in [n]\}$ is linearly independent (See Appendix D.1 for proof).
2. Let $\{l_1, \dots, l_r\}$ be a set of r linearly independent linear forms in $\mathcal{L}(T_0) \cup \mathcal{L}(T_1)$. Then $\pi_V(\{l_1, \dots, l_r\})$ is linearly independent with high probability. So $rank(\pi_V(f)) = r$ (See Appendix D.2 for Proof).

3. Let $l_{01} \in \mathcal{L}(T_0), l_{11} \in \mathcal{L}(T_1)$, then $\pi_V(l_{01}), \pi_V(l_{11})$ are linearly independent with high probability and so $gcd(\pi_V(T_0), \pi_V(T_1)) = 1$.

Pick large number of ($\geq d^r$) random points $p_i, i = 1, \ldots, d^r$ in the space V. Use the values $\{f(p_i)\}$ and get a white-box (coefficient) representation for $\pi_V(f)$. With high probability over the choice of points lagrange interpolation will work (See Appendix D.3 for Proof). Note that the number of coefficients in $f|_V = O(d^r)$. Now this white box representation of $\pi_V(f)$ is reconstructed using the algorithm in Chapter 3. A number of such reconstructions are then glued to reconstruct the original polynomial.

4.2 Lifting from the Random Low Dimensional Subspace

1. Consider spaces $V_i = V \oplus \mathbb{R}v_i$ for $i = r+1, \ldots, n$.

2. Reconstruct $\pi_{V_i}(f)$ and $\pi_V(f)$ for each $i \in \{r+1, \ldots, n\}$.

3. Let $l = \sum_{i=1}^{n} a_i v_i$ be a linear form dividing one of the gates of f say T_0. $\pi_V(l) = \sum_{i=1}^{r} a_i v_i$ and $\pi_{V_i}(l) = \sum_{j=1}^{r} a_j v_j + a_i v_i$. Using our algorithm discussed in Chapter 3 we would have reconstructed $\pi_V(f)$ and $\pi_{V_i}(f)$. So we know the triples $(\pi_V(G), \pi_V(T_0), \pi_V(T_1))$ and $(\pi_{V_i}(G), \pi_{V_i}(T_0), \pi_{V_i}(T_1))$ On restricting V_i to V :

   a) **Only Factors become factors** with high probability so we can easily find the correspondence between $\pi_V(G)$ and $\pi_{V_i}(G)$.

   b) $\pi_V(\pi_{V_i}(T_0)) = \pi_V(T_0)$ and $\neq \pi_V(T_1)$ because of uniqueness of representation and therefore we get the correspondence between gates.

   c) Now to get correspondence between linear forms. Let $\pi_V(l)$ have multiplicity k in $\pi_V(T_0)$. Then with high probability l has multiplicity k in T_0 Since two LI vectors remain LI on projecting to a random subspace of dimension ≥ 2 (again See Appendix D.2 for proof). Therefore $\pi_{V_i}(l)$ has multiplicity k and is the unique lift of $\pi_V(l)$ for all i. Let $\pi_{V_i}(l) = \pi_V(l) + a_i v_i$. Then $l = \pi_V(l) + \sum_{i=r+1}^{n} a_i v_j$. This finds G, T_0, T_1 for us

4.3 Time Complexity

- Interpolation to find whitebox representation $\pi_V(f)$ which is a degree d polynomial over r variables clearly takes $poly(d^r)$ time (accounts to solving a linear system of size $poly(d^r)$).
- Solving $n - r$ instances of the low rank problem (simple ranks r and $r+1$) takes $npoly(d^r)$ time.
- The above mentioned approach to glue the linear forms in the gates clearly takes $poly(n, d)$ time.
- Overall the algorithm takes $poly(n, d)$ time since r is a constant.

5 Conclusion and Future Work

We described an efficient randomized algorithm to reconstruct circuit representation of multivariate polynomials which exhibit a $\Sigma\Pi\Sigma(2)$ representation. Our algorithm works for all polynomials with rank(number of independent variables greater than a constant r). In future we would like to address the following:

- **Reconstruction for Lower Ranks -** As can be seen in the paper, rank of the polynomial for uniqueness (i.e. $c_{\mathbb{R}}(4)$) and the rank we've assumed in the low rank reconstruction (i.e. r) are both $O(1)$ but $c_{\mathbb{R}}(4)$ is smaller than r. Since one would expect a reconstruction algorithm whenever the circuit is unique we would like to close this gap.
- $\Sigma\Pi\Sigma(k)$ **circuits -** It would be interesting to consider more general top fan-in. In particular we could consider $\Sigma\Pi\Sigma(k)$ circuits with $k = O(1)$.
- **Derandomization -** We would like to derandomize the algorithm as it was done in the finite field case in [KS09a].

6 Acknowledgements

I am extremely thankful to Neeraj Kayal for introducing me to this problem. Sukhada Fadnavis, Neeraj Kayal and myself started working on the problem together during my summer internship at Microsoft Research India Labs in 2011. We solved the first important case together. I'm grateful to them for all helpful discussions, constant guidance and encouragement.

I would like to thank Zeev Dvir for communicating the most recent rank bounds on $\delta - SG_k$ configurations from [DSW12] and his feedback on the work. This reduces the gap in the first problem we mentioned above.

I would also like to thank Vinamra Agrawal, Pravesh Kothari and Piyush Srivastava for helpful discussions. Lastly I would like to thank Microsoft Research for giving me the opportunity to intern at their Bangalore Labs with the Applied Mathematics Group.

A Proofs from Subsection 3.2

Claim A.1. *The following are true about our candidate set* $\mathcal{C}$.

1. $\mathcal{L}(T_i) \subseteq \mathcal{C}, i = 0, 1.$
2. *Let* $k = c_{\mathbb{R}}(3) + 2$ *and suppose* $\{l_j; j \in [k]\} \subset \mathcal{L}(T_i)$ *are LI . Then for any* $l \in \mathcal{C} \setminus (\mathcal{L}(T_0) \cup \mathcal{L}(T_1))$*, there exists* $j \in [k]$ *such that* $fl(\{l, l_j\}) \cap \mathcal{L}(T_{1-i}) \neq \phi$ *i.e. the line joining* l *and* l_j *does not intersect the set* $\mathcal{L}(T_{1-i})$.
3. $|\mathcal{C}| \leq M^4 + 2M \leq d^4 + 2d.$

Proof. Let's first recall the definition of our candidate set

$$\mathcal{C} \stackrel{def}{=} \{l = x_1 - a_2x_2 - \ldots - a_rx_r \in Lin_{\mathbb{R}}[\bar{x}] : h(a_2x_2 + \ldots + a_rx_r, x_2, \ldots, x_r) \in \Pi\Sigma_{\mathbb{R}}^{d_h}[x_2, \ldots, x_r]\}$$

Also recall that

$$h = \frac{g}{Lin(g)} = \frac{f}{Lin(f)}$$

1. Let $l = x_1 - a_2x_2 - \ldots - a_rx_r \in \mathcal{L}(T_{1-i})$. Let's denote the tuple $v \equiv (a_2x_2 + \ldots + a_rx_r, x_2, \ldots, x_r)$. Since $gcd(T_0, T_1) = 1$ and $l \mid T_{1-i}$ we know that $l \nmid T_i$ and therefore $Lin(g)(v) \neq 0$. We can then compute
$$h(v) = \frac{\alpha_i T_i(v)}{Lin(g)(v)} = \alpha_i H_1(v) \ldots H_{d_h}(v) \in \Pi\Sigma_{\mathbb{R}}^{d_h}[x_2, \ldots, x_r]$$
where $H_j \in Lin_{\mathbb{R}}[x_2, \ldots, x_r]$. So $\mathcal{L}(T_i) \subseteq \mathcal{C}$ for $i = 0, 1$.

2. Consider $l = x_1 - a_2x_2 - \ldots - a_rx_r \in \mathcal{C} \setminus (\mathcal{L}(T_0) \cup \mathcal{L}(T_1))$ and assume that $sp(\{l, l_j\}) \cap \mathcal{L}(T_{1-i}) = \phi$ for all $j \in [k]$. We know that
$$g(v) = Lin(g)(v)H_1(v) \ldots H_{d_h}(v) = \alpha_0 T_0(v) + \alpha_1 T_1(v)$$
Let g' be the following identically zero $\Sigma\Pi\Sigma(3)[x_2, \ldots, x_r]$ polynomial (with circuit $\mathcal{C}'$)
$$g' = Lin(g)(v)H_1(v) \ldots H_{d_h}(v) - \alpha_0 T_0(v) - \alpha_1 T_1(v)$$
We know
$$\mathcal{C}' = gcd(\mathcal{C}')Sim(\mathcal{C}') \Rightarrow Sim(\mathcal{C}') \equiv 0$$
Recall that $l_j(v) \mid T_i(v)$, therefore the $l_j(v)$ cannot be factors of $gcd(\mathcal{C}')$ because if they did then there exist pair $l_j, l_{(1-i)t}$ such that $\{l_j(v), l_{(1-i)t}(v)\}$ is LD or in other words $sp(\{l, l_j\}) \cap \mathcal{L}(T_{1-i}) \neq \phi$ and we have a contradiction. Also the set $\{l_j(v) : j \in [k]\}$ has dimension $\geq k - 1$ since the dimension could fall only by 1 when we go modulo a linear form (project to hyperplane). This means that $rank(Sim(\mathcal{C}')) \geq k - 1 \geq c_{\mathbb{R}}(3) + 1$.

   If $Sim(\mathcal{C}')$ **were not minimal** $\Rightarrow$ $\mathcal{C}'$ is not minimal $\Rightarrow$ one of it's gates would be 0. Since $l \notin \mathcal{L}(T_0) \cup \mathcal{L}(T_1) \Rightarrow \alpha_0 T_0(v) + \alpha_1 T_1(v) \equiv 0 \Rightarrow$ for every $j \in [k]$ there exist $l_{(1-i)j} \mid T_{1-i}$ such that $l_{(1-i)j}(v), l_j(v)$ are LD. $\Rightarrow sp(\{l, l_j\}) \cap \mathcal{L}(T_{1-i}) \neq \phi$ for $j \in [k]$, a contradiction to our assumption.

   If $Sim(\mathcal{C}')$ **were minimal**, we have an identically zero simple minimal circuit $Sim(\mathcal{C}')$ with $rank(Sim(\mathcal{C}')) \geq c_{\mathbb{R}}(3) + 1$ contradicting Theorem 1.6.

   So our assumption is wrong and $sp(\{l, l_j\}) \cap \mathcal{L}(T_{1-i}) \neq \phi$ for some $j \in [k]$.

3. Let $l \in \mathcal{C} \setminus (\mathcal{L}(T_0) \cup \mathcal{L}(T_1))$. Consider a set $\{l_1, \ldots, l_{k+2}\} \subset \mathcal{L}(T_i)$ of $k+2$ LI linear forms. By the above argument there exist three distinct elements in this set say l_1, l_2, l_3 such that $sp(\{l_j, l\}) \cap \mathcal{L}(T_{1-i}) \neq \phi$ for $j \in [3]$. Let $\{l'_1, l'_2, l'_3\} \subset \mathcal{L}(T_{1-i})$ such that $l'_j \in sp(\{l_j, l\})$ for $j \in [3]$. Then $gcd(l_j, l'_j) = 1$ implies that $l \in sp(\{l_j, l'_j\})$ for $j \in [3]$. Since l, l_j, l'_j are all standard (coefficient of x_1 is 1), Lemma 2.2 tells us
$$l \in fl(\{l_j, l'_j\})$$
for $j \in [3]$. So l lies on the lines $\vec{L_j} = fl(\{l_j, l'_j\})$ for $j \in [3]$. Atleast two of these lines should be distinct otherwise $dim(\{l_1, l_2, l_3\}) \leq 2$ which is a contradiction. So l is the intersection of these two lines. There are M^2 such lines and so M^4 such intersections. If $l \in \mathcal{L}(T_0) \cup \mathcal{L}(T_1)$ we have $\leq 2M$ other possibilities. So $|\mathcal{C}| \leq M^4 + 2M = O(d^4)$.

□

B Proofs from Subsection 3.5

Claim B.1. *Let $(S = \{l_1 \ldots, l_k\}, D)$ be a Detector pair in $\mathcal{L}(T_i)$. Let $l_{k+1} \in D$. For a standard linear form $l \in V$, if $l \mid g$ then $l \notin sp(\{l_1, \ldots, l_k\})$.*

Proof. Assume $l \mid g$ and $l \in sp(\{l_1, \ldots, l_k\})$. Let $W = sp(\{l\})$, extend it to a basis and in the process obtain W' such that $W \oplus W' = V$. We get
$$\pi_{W'}(\alpha_0 T_0 + \alpha_1 T_1) = 0$$
$\pi_{W'}(\alpha_i T_i) \neq 0$ (i.e. $l \nmid T_0 T_1$), otherwise l divides both T_0, T_1 and $gcd(T_0, T_1)$ won't be 1. So we have an equality of non zero $\Pi\Sigma$ polynomials
$$\alpha_0 \prod_{j=1}^{M} \pi_{W'}(l_{0j}) = -\alpha_1 \prod_{j=1}^{M} \pi_{W'}(l_{1j})$$
Therefore there exists a permutation $\theta : [M] \to [M]$ such that $\{\pi_{W'}(l_{(1-i)j}), \pi_{W'}(l_{i\theta(j)})\}$ are LD $\Rightarrow l \in sp(\{l_{(1-i)j}, l_{i\theta(j)}\})$. Since $l \nmid T_0 T_1$ this also means that $l_{(1-i)j} \in sp(\{l, l_{i\theta(j)}\})$ and $l_{i\theta(j)} \in sp(\{l, l_{(1-i)j}\})$.

In particular there is an $l'_{k+1} \in \mathcal{L}(T_{1-i})$ such that $l'_{k+1} \in sp(\{l, l_{k+1}\})$ and $l_{k+1} \in sp(\{l, l'_{k+1}\})$.

Since $l \in sp(\{l_1, \ldots, l_k\}) \Rightarrow l'_{k+1} \in sp(\{l_1, \ldots, l_k, l_{k+1}\})$. All linear forms here are standard(i.e. coefficient of x_1 is 1) and so by Lemma 2.2, $l'_{k+1} \in fl(\{l_1, \ldots, l_k, l_{k+1}\})$. Below we use the definition of detector pair and get
$$l'_{k+1} \in fl(\{l_1, \ldots, l_k, l_{k+1}\}) \cap \mathcal{L}(T_{1-i}) \subseteq fl(\{l_1, \ldots, l_k\})$$
And $l_{k+1} \in sp(\{l, l'_{k+1}\}) \Rightarrow l_{k+1} \in sp(\{l_1, \ldots, l_k\})$ which is a contradiction to (S, D) being a detector pair..

□

Claim B.2. *Let $l \in Lin_{\mathbb{R}}[\bar{x}]$ be standard such that $l \mid g$ and $\mathcal{C}$ be the candidate set. Assume $(S = \{l_1, \ldots, l_k\}, D(\neq \phi))$ is a Detector pair in $\mathcal{L}(T_i)$. Then $|\mathcal{L}(T_{1-i}) \cap (fl(S \cup \{l\}) \setminus fl(S))| \geq 2$. That is the flat $fl(\{l_1, \ldots, l_k, l\})$ contains atleast two distinct points from $\mathcal{L}(T_{1-i})(\subseteq \mathcal{C})$ outside $fl(\{l_1, \ldots, l_k\})$.*

Proof. From the previous claim we know that $\{l_1,\ldots,l_k,l\}$ is an LI set. Also like above we know there exists $l'_j \in \mathcal{L}(T_{1-i}), j \in [3]$ such that $l_j \in sp(\{l,l'_j\}), l'_j \in sp(\{l,l_j\})$. Since $\{l_1,l_2,l_3\}$ are LI, atleast two of the l'_j's, $j \in [3]$ must be distinct, otherwise $sp(\{l_1,l_2,l_3\}) \subset sp(\{l,l'_1\})$ which is not possible as LHS has dimension 3 and RHS has dimension 2. Thus there exist two distinct $l'_1, l'_2 \in sp(\{l_1,l_2,l_3,l\}) \subset sp(\{l_1,\ldots,l_k,l\})$. Note that $l_1,\ldots,l_k,l,l'_1,l'_2$ are all standard (i.e. coefficient of x_1 is 1) and so by Lemma 2.2

$$l'_j \in fl(\{l_1,\ldots,l_k,l\})$$

for $j \in [2]$.

If for any $j \in [2]$, $l'_j \in sp(\{l_1,\ldots,l_k\})$ then $l \in sp(\{l_j,l'_j\}) \Rightarrow l \in sp(\{l_1,\ldots,l_k\})$ which is a contradiction. This also shows that $l'_j \notin fl(\{l_1,\ldots,l_k\})$ for $j \in [2]$.

From what we showed above we may conclude:

$$l'_j \in fl(\{l_1,\ldots,l_k,l\}) \setminus fl(\{l_1,\ldots,l_k\})$$

for $j \in [2]$. Hence Proved.

□

Lemma B.3. *The following are true:*

1. *If $l \mid I$ (i.e. l was identified) then $l \in \mathcal{L}(G) \setminus \mathcal{L}(g)$.*
2. *If $l \mid G^\star$ (i.e. l was retained) then $(fl(\{l_1,\ldots,l_k,l\}) \setminus fl(\{l_1,\ldots,l_k\})) \cap (\mathcal{L}(T_{1-i}) \cup (\mathcal{L}(T_i) \setminus D)) \neq \phi$ that is*

   $(fl(\{l_1,\ldots,l_k,l\}) \setminus fl(\{l_1,\ldots,l_k\}))$ contains a point from $\mathcal{L}(T_i) \setminus D$ or $\mathcal{L}(T_{1-i})$.
3. *If $l \mid G^\star$ and $l_{k+1} \in D$ then $l \notin sp(\{l_1,\ldots,l_k,l_{k+1}\})$.*

Proof. 1. Assume $l \mid I$ (i.e. l was identified) and $l \mid g$. Then by Claim 3.7 we know that $\{l_1,\ldots,l_k,l\}$ are LI and so the first "if" condition is true. By Claim 3.8 we know that there are two other points $\{l'_1,l'_2\} \subset \mathcal{C} \cap (fl(\{l_1,\ldots,l_k,l\}) \setminus fl(\{l_1,\ldots,l_k\}))$, so the second "$if$" condition will also be true and thus l will not be identified which is a contradiction. Therefore $l \in \mathcal{L}(G) \setminus \mathcal{L}(g)$.

2. Assume $l \mid G^\star$ (i.e. l was not identified). This means both "if" statements were true for l. Thus $\{l_1,\ldots,l_k,l\}$ is LI. Also there exist distinct $\{l'_1,l'_2\} \in \mathcal{C} \cap (fl(\{l_1,\ldots,l_k,l\}) \setminus fl(\{l_1,\ldots,l_k\}))$. If

   $$l'_1 \in (\mathcal{L}(T_{1-i}) \cup (\mathcal{L}(T_i) \setminus D)) \text{ or } l'_2 \in (\mathcal{L}(T_{1-i}) \cup (\mathcal{L}(T_i) \setminus D))$$

   we are done so assume both are in

   $$\mathcal{C} \setminus ((\mathcal{L}(T_{1-i}) \cup (\mathcal{L}(T_i) \setminus D)))) = (\mathcal{C} \setminus (\mathcal{L}(T_i) \cup \mathcal{L}(T_{1-i}))) \cup D$$

   If one of them say $l'_1 \in \mathcal{C} \setminus (\mathcal{L}(T_i) \cup \mathcal{L}(T_{1-i}))$, then by Part 2 of Claim 3.3, for some $j \in [k]$, $sp(\{l'_1,l_j\}) \cap \mathcal{L}(T_{1-i}) \neq \phi$. Let $\tilde{l}_j \in sp(l'_1,l_j) \cap \mathcal{L}(T_{1-i}) \Rightarrow$

   $$\tilde{l}_j \in sp(\{l'_1,l_j\}) \subseteq sp(\{l_1,\ldots,l_k,l\})$$

   Since all linear forms $\tilde{l}_j, l_1,\ldots,l_k,l$ are standard (coefficient of x_1 is 1) by Lemma 2.2

   $$\tilde{l}_j \in fl(\{l_1,\ldots,l_k,l\})$$

Also $\tilde{l}_j, l_j$ are LI and $\tilde{l}_j \in sp(\{l'_1, l_j\})$ together imply $l'_1 \in sp(\{l_j, \tilde{l}_j\})$. Note that $l'_1 \notin fl(\{l_1, \dots, l_k\}) \Rightarrow l'_1 \notin sp(\{l_1, \dots, l_k\})$ which along with $l'_1 \in sp(\{l_j, \tilde{l}_j\})$ will then give

$$\tilde{l}_j \notin sp(\{l_1, \dots, l_k\})$$

So we found $\tilde{l}_j \in \mathcal{L}(T_{1-i}) \cap (fl(\{l_1, \dots, l_k, l\}) \setminus fl(\{l_1, \dots, l_k\}))$ and we are done.

So the only case that remains now is that $l'_1, l'_2 \in D$. Let's complete the proof in the following steps

- $l'_1 \in fl(\{l_1, \dots, l_k, l\}) \setminus fl(\{l_1, \dots, l_k\}) \Rightarrow l \in sp(\{l_1, \dots, l_k, l'_1\})$
- Using the above bullet, $l'_2 \in fl(\{l_1, \dots, l_k, l\}) \Rightarrow l'_2 \in sp(\{l_1, \dots, l_k, l'_1\})$. Linear forms $l'_2, l_1, \dots, l_k, l$ are standard (coefficient of x_1 is 1) so using Lemma 2.2, $l'_2 \in fl(\{l_1, \dots, l_k, l'_1\})$
- $l'_2 \in D \Rightarrow l'_2 \notin fl(\{l_1, \dots, l_k\})$
- The above two bullets and $\{l'_1, l'_2\} \subset \mathcal{L}(T_i)$ tell us that $fl(\{l_1, \dots, l_k, l'_1\})$ is not elementary which is a contradiction.

So atleast one of l'_1, l'_2 is inside $\mathcal{L}(T_{1-i}) \cup (\mathcal{L}(T_i) \setminus D)$

3. Let $l_{k+1} \in D$ and $l \in sp(\{l_1, \dots, l_k, l_{k+1}\})$. Since $l, l_1, \dots, l_k, l_{k+1}$ are standard, by Lemma 2.2, $l \in fl(\{l_1, \dots, l_k, l_{k+1}\})$. Clearly $l \notin fl(\{l_1, \dots, l_k\})$ otherwise it would get identified at the first "if". Therefore $l \in fl(\{l_1, \dots, l_k, l_{k+1}\}) \setminus fl(\{l_1, \dots, l_k\})$ By Part 2 above let $l'_1 \in (fl(\{l_1, \dots, l_k, l\}) \setminus fl(\{l_1 \dots, l_k\})) \cap (\mathcal{L}(T_{1-i}) \cup (\mathcal{L}(T_i) \setminus D))$. So $l'_1 \in \mathcal{L}(T_{1-i})$ or $l'_1 \in \mathcal{L}(T_i) \setminus D$.

   This tells us that $l'_1 \in sp(\{l_1, \dots, l_k, l_{k+1}\}) \setminus fl(\{l_1, \dots, l_k\})$. All linear forms $l'_1, l_1, \dots, l_k, l_{k+1}$ are standard (i.e. coefficients of x_1 is 1) so by Lemma 2.2 we get that $l'_1 \in fl(\{l_1, \dots, l_k, l_{k+1}\}) \setminus fl(\{l_1, \dots, l_k\})$. Now using the definition of detector pair $l'_1 \notin \mathcal{L}(T_{1-i})$ since $fl(\{l_1, \dots, l_k, l_{k+1}\}) \cap \mathcal{L}(T_{1-i}) \subseteq fl(\{l_1, \dots, l_k\})$. The flat $fl(\{l_1, \dots, l_k, l_{k+1}\})$ is elementary in $\mathcal{L}(T_i)$, so l'_1 can belong here only if $l'_1 = l_{k+1}$ which is not possible since $l'_1 \notin D$. So we have a contradiction. Hence Proved. ∎

Lemma B.4. *Let $(S = \{l_1, \dots, l_k\}, D)$ be a detector in $\mathcal{L}(T_i)$. For each $(l, l_j) \in \mathcal{C} \times S$ define the space $U_{\{l,l_j\}} = sp(\{l, l_j\})$. Extend $\{l, l_j\}$ to a basis and in the process obtain $U'_{\{l,l_j\}}$ such that $V = U_{\{l,l_j\}} \oplus U'_{\{l,l_j\}}$. Define the set:*

$$X = \{l \in \mathcal{C} : \pi_{U'_{\{l,l_j\}}}(f^\star) \neq 0, \textit{ for all } l_j \in S\}$$

Then $D \subset X \subset \mathcal{L}(T_i)$.

Proof. $\underline{(D \subset X):}$ Consider $l_{k+1} \in D$. Since $D \subset \mathcal{L}(T_i) \Rightarrow l_{k+1} \in \mathcal{C}$. Assume $l_{k+1} \notin X$, so there exists a $j \in [k]$ such that $\pi_{U'_{\{l_{k+1},l_j\}}}(f^\star) = 0$. That is:

$$\pi_{U'_{\{l_{k+1},l_j\}}}(G^\star(\alpha_0 T_0 + \alpha_1 T_1)) = 0.$$

So

$$\prod_{t \in [N_1]} \pi_{U'_{\{l_{k+1},l_j\}}}(G_t)(\alpha_0 \prod_{s \in [M]} \pi_{U'_{\{l_{k+1},l_j\}}}(l_{0s}) + \alpha_1 \prod_{s \in [M]} \pi_{U'_{\{l_{k+1},l_j\}}}(l_{1s})) = 0$$

Now

$$l_j \in \mathcal{L}(T_i) \Rightarrow \pi_{U'_{\{l_{k+1},l_j\}}}(T_i) = 0 \Rightarrow \prod_{t\in[N_1]} \pi_{U'_{\{l_{k+1},l_j\}}}(G_t) \prod_{s\in[M]} \pi_{U'_{\{l_{k+1},l_j\}}}(l_{(1-i)s}) = 0.$$

Since $G_t \mid G^{\star}$, by Part 3 of Lemma 3.10 $\pi_{U'_{\{l_{k+1},l_j\}}}(G_t) \neq 0$ for all $t \in [N_1]$. If for some $s \in [M]$, $\pi_{U'_{\{l_{k+1},l_j\}}}(l_{(1-i)s}) = 0$ then $l_{(1-i)s} \in sp(\{l_j, l_{k+1}\}) \Rightarrow l_{(1-i)s} \in sp(\{l_1, \ldots, l_k, l_{k+1}\}) \Rightarrow l_{(1-i)s} \in sp(\{l_1, \ldots, l_k\})$ (by definition of Detector Pair in 3.5).

$$l_{(1-i)s} \in sp(\{l_j, l_{k+1}\}) \text{ and } \{l_{(1-i)s}, l_j\} \text{ LI } \Rightarrow l_{k+1} \in sp(\{l_{(1-i)s}, l_j\})$$

This means $l_{k+1} \in sp(\{l_1, \ldots, l_k, l_{(1-i)s}\}) \subset sp(\{l_1, \ldots, l_k\})$ which is a contradiction to $l_{k+1} \in D$. So $\pi_{U'_{\{l_{k+1},l_j\}}}(f^{\star}) \neq 0$ for all $j \in [k] \Rightarrow l_{k+1} \in X$. Therefore $D \subset X$.

$(X \subset \mathcal{L}(T_i))$: Consider $l \in X$. We need to show $l \in \mathcal{L}(T_i)$. We already know $l \in \mathcal{C}$.

- If $l \in \mathcal{L}(T_{1-i})$, then $\pi_{U'_{\{l,l_j\}}}(f^{\star}) = 0$ for all $j \in [k]$ since $l \mid T_{1-i}$ and $l_j \mid T_i$. Contradiction to $l \in X$.
- If $l \in \mathcal{C} \setminus (\mathcal{L}(T_i) \cup \mathcal{L}(T_{1-i}))$ by Part 2 of Claim 3.3 we know that there exists $j \in [k]$ such that $sp(\{l_j, l\}) \cap \mathcal{L}(T_{1-i}) \neq \phi$. Let $l'_j \in sp(\{l_j, l\}) \cap \mathcal{L}(T_{1-i})$. We show that $sp(\{l'_j, l_j\}) = sp(\{l_j, l\}) = U_{\{l_j,l\}}$.
  - $l'_j \in sp(\{l_j, l\}) \Rightarrow sp(\{l'_j, l_j\}) \subset sp(\{l_j, l\})$.
  - Let $l'_j = \alpha l_j + \beta l$. We know that $\{l_j, l'_j\}$ are LI since $l_j \in \mathcal{L}(T_i)$ and $l'_j \in \mathcal{L}(T_{1-i})$. So $\beta \neq 0 \Rightarrow l \in sp(\{l'_j, l_j\}) \Rightarrow sp(\{l, l_j\}) \subset sp(\{l'_j, l_j\}) \Rightarrow sp(\{l, l_j\}) = sp(\{l'_j, l_j\})$.

  Use the same extension for $sp(\{l, l_j\}) = sp(\{l'_j, l_j\}) = U_{\{l_j,l\}}$ to get $\pi_{U'_{\{l,l_j\}}}(f^{\star}) = \pi_{U'_{\{l'_j,l_j\}}}(f^{\star}) = 0$ (since $l'_j \mid T_{1-i}$ and $l_j \mid T_i$). Contradiction to $l \in X$.

Therefore $l \in \mathcal{L}(T_i) \Rightarrow X \subset \mathcal{L}(T_i)$. □

C Proofs from Subsection 3.6

Claim C.1. *The following is true*

$$\frac{(2 - v(\delta, \theta))}{v(\delta, \theta)} \leq \frac{1-\delta}{\delta}$$

Proof. Note that

$$\frac{(2 - v(\delta, \theta))}{v(\delta, \theta)} = \begin{cases} \frac{1+\delta+\theta}{1-\delta-\theta} & \text{if } |\mathcal{L}(T_0)| \leq \theta|\mathcal{L}(T_1)| \\ \frac{3-(1-\delta)(1+\theta)}{(1-\delta)(1+\theta)-1} & \text{if } \theta|\mathcal{L}(T_1)| < |\mathcal{L}(T_0)| \leq |\mathcal{L}(T_1)| \end{cases}$$

By simple computation $\delta \in (0, \frac{7-\sqrt{37}}{6})$ gives

$$3\delta^2 - 7\delta + 1 > 0 \Rightarrow 0 < \frac{3\delta}{1-\delta} < 1 - 3\delta < 1 \Rightarrow \frac{1+\delta+\theta}{1-\delta-\theta} < \frac{1-\delta}{\delta}$$

Also

$$\theta > \frac{3\delta}{1-\delta} \Rightarrow \frac{3-(1-\delta)(1+\theta)}{(1-\delta)(1+\theta)-1} < \frac{1-\delta}{\delta}$$

□

Lemma C.2. *Let $k = c_{\mathbb{R}}(3) + 2$ (see defn of $c_{\mathbb{R}}(k)$ in Theorem 1.6). Fix δ, θ in range given in Claim 3.12 above . Then for some $i \in \{0, 1\}$ there exists a Detector Pair $(S = \{l_1, \ldots, l_k\}, D)$ in $\mathcal{L}(T_i)$ with $|D| \geq v(\delta, \theta) \max(|\mathcal{L}(T_0)|, |\mathcal{L}(T_1)|)$.*

Proof. We assume $|\mathcal{L}(T_0)| \leq \mathcal{L}(T_1)$. The other case gives the same result for(maybe) a different value of i . We will consider linear forms as points in the space $\mathbb{R}^r$. Let's consider the two cases used in the definition of $v(\delta, \theta)$.

- **Case 1 : $|\mathcal{L}(T_0)| \leq \theta|\mathcal{L}(T_1)|$ (i.e. $\mathcal{L}(T_0)$ is much smaller) $\Rightarrow v(\delta, \theta) = 1 - \delta - \theta$:**

  Since $dim(\mathcal{L}(T_1)) \geq r - 1 \geq C_{2k-1} > C_k$ (See Section 2.3 for definition of C_k) by Corollary 2.12 there exists a set S of k LI points say $S = \{l_1, \ldots, l_k\} \subseteq \mathcal{L}(T_1)$ and a set $Z \subseteq \mathcal{L}(T_1)$ of size $\geq (1 - \delta)|\mathcal{L}(T_1)|$ such that for any $l_{k+1} \in Z$

  - $l_{k+1} \notin fl(\{l_1, \ldots, l_k\})$.
  - $fl(\{l_1, \ldots, l_k, l_{k+1}\})$ is elementary in $\mathcal{L}(T_1)$.

  Next we define our set D according to the condition we needed in the definition of detector (See Subsection 3.5).

  $$D \overset{def}{=} \{l_{k+1} \in Z : fl(\{l_1, \ldots, l_k, l_{k+1}\}) \cap \mathcal{L}(T_0) \subset fl(\{l_1, \ldots, l_k\})\}$$

  In the following lines we will show that this set D has large size, to be precise:

  $$|D| \geq (1 - \delta - \theta)|\mathcal{L}(T_1)|$$

  We do this in steps:

  1. First we define a special subset of Z

     $$\tilde{Z} = \{l_{k+1} \in Z : (fl(\{l_1, \ldots, l_{k+1}\}) \setminus fl(\{l_1, \ldots, l_k\})) \cap \mathcal{L}(T_0) \neq \phi\}$$

     We claim that $Z \setminus \tilde{Z} \subset D$. Let $l_{k+1} \in Z \setminus \tilde{Z} \Rightarrow (fl(\{l_1, \ldots, l_{k+1}\}) \setminus fl(\{l_1, \ldots, l_k\})) \cap \mathcal{L}(T_0) = \phi \Rightarrow fl(\{l_1, \ldots, l_{k+1}\}) \cap \mathcal{L}(T_0) \subset fl(\{l_1, \ldots, l_k\})$ and so $l_{k+1} \in D$.
  2. Next we show that for distinct $l_{k+1}, \tilde{l}_{k+1} \in Z (\subseteq \mathcal{L}(T_1))$

     $$(fl(\{l_1, \ldots, l_k, l_{k+1}\}) \setminus fl(\{l_1, \ldots, l_k\})) \cap (fl(\{l_1, \ldots, l_k, \tilde{l}_{k+1}\}) \setminus fl(\{l_1, \ldots, l_k\})) = \phi$$

     If not then there exist scalars $\mu_j, \nu_j, j \in [k + 1]$ such that

     $$\nu_1 l_1 + \ldots \nu_k l_k + \nu_{k+1} l_{k+1} = \mu_1 l_1 + \ldots \mu_k l_k + \mu_{k+1} \tilde{l}_{k+1}$$

     with $\nu_{k+1} \neq 0$ implying that $l_{k+1} \in sp(\{l_1, \ldots, l_k, \tilde{l}_{k+1}\})$. Since all linear forms are *standard* this implies $l_{k+1} \in fl(\{l_1, \ldots .l_k, \tilde{l}_{k+1}\})$ (See Lemma 2.2). Also $l_{k+1} \in Z \Rightarrow l_{k+1} \notin fl(\{l_1, \ldots, l_k\})$. Together this means that $l_{k+1} \in fl(\{l_1, \ldots, l_k, \tilde{l}_{k+1}\}) \setminus fl(l_1, \ldots, l_k)$ and we arrive at a contradiction to $fl(\{l_1, \ldots, l_k, \tilde{l}_{k+1}\})$ being elementary.
  3. From what we showed above every $l \in \mathcal{L}(T_0)$ can belong to atmost one of the sets $fl(\{l_1, \ldots, l_{k+1}\}) \setminus fl(\{l_1, \ldots, l_k\})$ with $l_{k+1} \in Z$ (since intersection between two such sets is ϕ) and therefore there can be atmost $|\mathcal{L}(T_0)|$ such l_{k+1}'s in $\tilde{Z} \Rightarrow |\tilde{Z}| \leq |\mathcal{L}(T_0)|$.

  So we get :

  $$|D| \geq |Z| - |\mathcal{L}(T_0)| \geq (1 - \delta - \theta)|\mathcal{L}(T_1)|$$

  (S, D) is a detector pair in $\mathcal{L}(T_1)$ by the choice of Z and D.

- **Case 2 :** $\theta|\mathcal{L}(T_1)| < |\mathcal{L}(T_0)| \leq |\mathcal{L}(T_1)|$ **(i.e. there sizes are comparable)** $\Rightarrow v(\delta,\theta) = (1-\delta)(1+\theta)-1$ **:**

  Since $dim(\mathcal{L}(T_0) \cup \mathcal{L}(T_1)) = r > C_{2k-1}$, by Corollary 2.12 we know that there exist $2k-1$ independent points $l_1,\ldots,l_{2k-1} \in \mathcal{L}(T_0)\cup\mathcal{L}(T_1)$ and a set $Z \subseteq \mathcal{L}(T_0)\cup\mathcal{L}(T_1)$ of size $\geq (1-\delta)(|\mathcal{L}(T_0)|+|\mathcal{L}(T_1)|)$ such that for all $l \in Z$

  - $l \notin fl(\{l_1,\ldots,l_{2k-1}\})$.
  - $fl(\{l_1,\ldots,l_{2k-1},l\})$ is elementary in $\mathcal{L}(T_0)\cup\mathcal{L}(T_1)$.

  By pigeonhole principle, k of the $\{l_j\}_{j=1}^{2k-1}$ points must belong to either $\mathcal{L}(T_0)$ or $\mathcal{L}(T_1)$. Let's assume they belong to $\mathcal{L}(T_i)$ (for some $i \in \{0,1\}$) (say the points are $l_1,\ldots,l_k$), then consider $D = Z\cap\mathcal{L}(T_i)$. Clearly for every $l \in D$, $l \notin fl(\{l_1,\ldots,l_k\})$ and $fl(\{l_1,\ldots,l_k,l\})$ is elementary in $\mathcal{L}(T_0)\cup\mathcal{L}(T_1)$. This immediately tells us that $(S=\{l_1,\ldots,l_k\},D)$ satisfies all properties of being a detector pair in $\mathcal{L}(T_i)$. We defined $D = Z\cap\mathcal{L}(T_i)$. Since $Z \subseteq \mathcal{L}(T_i)\cup\mathcal{L}(T_{1-i})$ we have $Z = (Z\cap\mathcal{L}(T_i))\cup(Z\cap\mathcal{L}(T_{1-i})) \subset D\cup\mathcal{L}(T_{1-i})$ giving

  $$|D|+|\mathcal{L}(T_{1-i})| \geq |Z| \Rightarrow |D| \geq |Z|-|\mathcal{L}(T_{1-i})| \geq (1-\delta)(|\mathcal{L}(T_0)|+|\mathcal{L}(T_1)|)-|\mathcal{L}(T_{1-i})|$$
  $$\geq ((1-\delta)(1+\theta)-1)\max(|\mathcal{L}(T_0)|,|\mathcal{L}(T_1)|)$$

Combining the two cases we see that for some $i \in \{0,1\}$ there exists a Detector set $(S=\{l_1,\ldots,l_k\},D)$ in $\mathcal{L}(T_i)$ with $|D| \geq v(\delta,\theta)\max(|\mathcal{L}(T_0)|,|\mathcal{L}(T_1)|)$.

□

Lemma C.3. *The following are true:*

1. $dim(\pi_{\tilde{W}_0}(\widehat{\mathcal{L}(U^\star_{1-i})})) > C_4$
2. $\pi_{\tilde{W}_0}(\widehat{\mathcal{L}(U^\star_{1-i})}) \cap \pi_{\tilde{W}_0}(\widehat{D}) = \phi$
3. $|\pi_{\tilde{W}_0}(\widehat{\mathcal{L}(U^\star_{1-i})})| \leq \frac{1-\delta}{\delta}|\pi_{\tilde{W}_0}(\widehat{D})|$

Proof.

1. Since $dim(\widehat{\mathcal{L}(U^\star_{1-i})}) \geq r-1$ we get $dim(\pi_{\tilde{W}_0}(\widehat{\mathcal{L}(U^\star_{1-i})})) \geq r-1-k > C_4$.

2. Assume $\exists\, d_1 \in D, u \in \mathcal{L}(U^\star_{1-i})$ such that $\pi_{\tilde{W}_0}(\widehat{d}) = \pi_{\tilde{W}_0}(\widehat{u}) \Rightarrow \exists\lambda,\nu \in \mathbb{R}$ such that $\nu d_1+\lambda u \in \tilde{W}_0$. Since $\pi_{\tilde{W}_0}(d_1) \neq 0$ both $\nu,\lambda \neq 0$. Thus $u \in sp(\{l_1,\ldots,l_k,d_1\}) \Rightarrow u \in fl(\{l_1,\ldots,l_k,d_1\})$ (using Lemma 2.2 since all linear forms involved are *standard* i.e. have coefficient of x_1 equal to 1). Also $u \in \mathcal{L}(G^\star T_{1-i}) \Rightarrow u \in fl(\{l_1,\ldots,l_k,d_1\})\cap(\mathcal{L}(G^\star)\cup\mathcal{L}(T_{1-i}))$. We know from Part 2 of Lemma 3.10 that $fl(\{l_1,\ldots,l_k,d_1\})\cap\mathcal{L}(G^\star) = \phi \Rightarrow u \in fl(\{l_1,\ldots,l_k,d_1\})\cap\mathcal{L}(T_{1-i}) \subseteq fl\{l_1,\ldots,l_k\}$ because (S,D) was a detector pair. But $u \in fl(\{l_1,\ldots,l_k\}) \Rightarrow d_1 \in sp(\{l_1,\ldots,l_k\})$ which is a contradiction because $d_1 \in D$ and (S,D) is a detector pair.

3. We first plan to show $\pi_{\tilde{W}_0}(\widehat{\mathcal{L}(U^\star_{1-i})}) \subset \pi_{\tilde{W}_0}(\widehat{\mathcal{L}(T_{1-i})})\cup\pi_{\tilde{W}_0}(\widehat{\mathcal{L}(T_i)}\setminus D)$. Clearly $U^\star_{1-i} \mid G^\star T_{1-i} \Rightarrow \mathcal{L}(U^\star_{1-i}) \subset \mathcal{L}(G^\star T_{1-i}) \Rightarrow \pi_{\tilde{W}_0}(\widehat{\mathcal{L}(U^\star_{1-i})}) \subset \pi_{\tilde{W}_0}(\widehat{\mathcal{L}(G^\star T_{1-i})}) \subset \pi_{\tilde{W}_0}(\widehat{\mathcal{L}(G^\star)})\cup\pi_{\tilde{W}_0}(\widehat{\mathcal{L}(T_{1-i})})$. Now consider any $l \in \mathcal{L}(G^\star)$. We know that $(S_0=\{l_1,\ldots,l_k\},D)$ is a detector pair, so by Part 2 of Lemma 3.10 we get

   $$(fl(\{l_1,\ldots,l_k,l\})\setminus fl(\{l_1,\ldots,l_k\}))\cap(\mathcal{L}(T_{1-i})\cup(\mathcal{L}(T_i)\setminus D)) \neq \phi$$

So there exists $l' \in \mathcal{L}(T_{1-i}) \cup (\mathcal{L}(T_i) \setminus D)$ such that $\pi_{\tilde{W_0}}(l), \pi_{\tilde{W_0}}(l')$ are both non-zero and are LD $\Rightarrow \pi_{\tilde{W_0}}(\widehat{l}) = \pi_{\tilde{W_0}}(\widehat{l'})$ implying that $\pi_{\tilde{W_0}}(\widehat{\mathcal{L}(G^\star)}) \subset \pi_{\tilde{W_0}}(\widehat{\mathcal{L}(T_{1-i}) \cup (\mathcal{L}(T_i) \setminus D)})$ giving us $\pi_{\tilde{W_0}}(\widehat{\mathcal{L}(U^\star_{1-i})}) \subset \pi_{\tilde{W_0}}(\widehat{\mathcal{L}(T_{1-i})}) \cup \pi_{\tilde{W_0}}(\widehat{\mathcal{L}(T_i) \setminus D})$ and therefore

$$|\pi_{\tilde{W_0}}(\widehat{\mathcal{L}(U^\star_{1-i})})| \leq |\pi_{\tilde{W_0}}(\widehat{\mathcal{L}(T_{1-i})})| + |\pi_{\tilde{W_0}}(\widehat{\mathcal{L}(T_i) \setminus D})|$$

Now we try to show $|\pi_{\tilde{W_0}}(\widehat{\mathcal{L}(T_i) \setminus D})| = |\pi_{\tilde{W_0}}(\widehat{\mathcal{L}(T_i)})| - |D|$

(a) It's straightforward to see $\pi_{\tilde{W_0}}(\widehat{\mathcal{L}(T_i)}) = \pi_{\tilde{W_0}}(\widehat{D}) \cup \pi_{\tilde{W_0}}(\widehat{\mathcal{L}(T_i) \setminus D})$. Also $\pi_{\tilde{W_0}}(\widehat{\mathcal{L}(T_i) \setminus D}) \cap \pi_{\tilde{W_0}}(\widehat{D}) = \phi$. If not then there exists $l' \in \mathcal{L}(T_i) \setminus D, l'' \in D$ such that $0 \neq \pi_{\tilde{W_0}}(\widehat{l''}) = \pi_{\tilde{W_0}}(\widehat{l'}) \Rightarrow \pi_{\tilde{W_0}}(l''), \pi_{\tilde{W_0}}(l')$ are LD $\Rightarrow l' \in sp\{l_1, \ldots, l_k, l''\} \setminus sp\{l_1, \ldots, l_k\} \Rightarrow$ (by Lemma 2.2), $l' \in fl\{l_1, \ldots, l_k, l''\} \setminus fl\{l_1, \ldots, l_k\}$ which is a contradiction to the flat being elementary inside $\mathcal{L}(T_i)$. So $|\pi_{\tilde{W_0}}(\widehat{\mathcal{L}(T_i)})| = |\pi_{\tilde{W_0}}(\widehat{D})| + |\pi_{\tilde{W_0}}(\widehat{\mathcal{L}(T_i) \setminus D})|$.

(b) $\pi_{\tilde{W_0}}$ is injective on $\widehat{D}$. Let $\pi_{\tilde{W_0}}(\widehat{l'}) = \pi_{\tilde{W_0}}(\widehat{l''})$ for LI forms $\{l', l''\} \subset D$, then $l' \in sp(\{l_1, \ldots, l_k, l''\}) \Rightarrow$ (by Lemma 2.2), $l' \in fl(\{l_1, \ldots, l_k, l''\})$ and clearly $l' \notin fl\{l_1, \ldots, l_k\}$ (since it's in D), which is again a contradiction to the flat being elementary , thus $|\pi_{\tilde{W_0}}(\widehat{D})| = |\widehat{D}| = |D|$ (since D is a set of $normal$ linear forms).

Combining these with Claim 3.12 and Lemma 3.13 we get

$$|\pi_{\tilde{W_0}}(\widehat{\mathcal{L}(U^\star_{1-i})})| \leq 2\max(|\mathcal{L}(T_0)|, |\mathcal{L}(T_1)|) - |D| \leq (2 - v(\delta, \theta))\max(|\mathcal{L}(T_0)|, |\mathcal{L}(T_1)|)$$

$\Rightarrow$

$$\frac{|\pi_{\tilde{W_0}}(\widehat{\mathcal{L}(U^\star_{1-i})})|}{|\pi_{\tilde{W_0}}(\widehat{D})|} \leq \frac{(2 - v(\delta, \theta))}{v(\delta, \theta)} \leq \frac{1 - \delta}{\delta}$$

□

Lemma C.4. *Let* $S_1 = \{d_1\}$ *and* $S_2 = \{l_{k+2}, \ldots, l_r\}$, $W_1 = sp(S_1)$ *and* $W_2 = sp(S_2)$. *So* $V = W_0 \oplus W_1 \oplus W_2$ *and let* $W_0' = W_1 \oplus W_2$. *For* $u \in \mathcal{L}(U^\star_{1-i})$ *such that* $\pi_{\tilde{W_0}}(\widehat{u}) \in \vec{L}_1 \cap \pi_{\tilde{W_0}}(\widehat{\mathcal{L}(U^\star_{1-i})})$ *consider the line*

$$\vec{L}_2 = fl(\{d_1, \pi_{W_0'}(\widetilde{u})\})$$

then $|\vec{L}_2 \cap \pi_{W_0'}(\widetilde{D})| \geq 1$ *and* $|\vec{L}_2 \cap \pi_{W_0'}(\widetilde{\mathcal{L}(U^\star_{1-i})})| = 1$, *i.e.* $\vec{L}_2$ *is also a* "semiordinary bichromatic" *like* $\vec{L}_1$.

Proof. We first show the following : Let $u_2 \in U^\star_{1-i}, d_2 \in D$ then

$$\pi_{W_0'}(\widetilde{u_2}) \neq \pi_{W_0'}(\widetilde{d_2})$$

- Assume not, then $\exists\, \nu, \lambda \in \mathbb{R}$ such that $\nu d_2 + \lambda u_2 \in W_0$. ν, λ cannot be 0 since this would mean $\pi_{W_0'}(\widetilde{d_2}) = 0$. Thus $u_2 \in sp(\{l_1, \ldots, l_k, d_2\}) \Rightarrow u_2 \in fl(\{l_1, \ldots, l_k, d_2\})$ (using Lemma 2.2 since all linear forms involved are *standard* i.e. have coefficient of x_1 equal to 1). Also $u_2 \in \mathcal{L}(G^\star T_{1-i}) \Rightarrow u_2 \in fl(\{l_1, \ldots, l_k, d_2\}) \cap (\mathcal{L}(G^\star) \cup \mathcal{L}(T_{1-i}))$. We know from Part 2 of Lemma 3.10 that $fl(\{l_1, \ldots, l_k, d_2\}) \cap \mathcal{L}(G^\star) = \phi \Rightarrow u_2 \in fl(\{l_1, \ldots, l_k, d_2\}) \cap \mathcal{L}(T_{1-i}) \subseteq fl\{l_1, \ldots, l_k\}$ because (S, D) was a detector pair. But $u_2 \in fl(\{l_1, \ldots, l_k\}) \Rightarrow d_2 \in sp(\{l_1, \ldots, l_k\})$ which is a contradiction because $d_2 \in D$ and (S, D) is a detector pair.

Now let's go back to proving this lemma.

$|\vec{L}_2 \cap \pi_{W_0'}(\widetilde{D})| \geq 1$ is clearly true since $d_1 \in \vec{L}_2 \cap \pi_{W_0'}(\widetilde{D})$. For the other part assume there exist $u_1 \neq u$ inside $\mathcal{L}(U^\star_{1-i})$ such that $\pi_{W_0'}(\widetilde{u}), \pi_{W_0'}(\widetilde{u}_1)$ are distinct points on $\vec{L}_2 \cap \pi_{W_0'}(\mathcal{L}(U^\star_{1-i}))$ implying that the set $\{\pi_{W_0'}(\widetilde{u}), \pi_{W_0'}(\widetilde{u}_1), \pi_{W_0'}(\widetilde{d}_1) = d_1\}$ is an LD set and there exist κ, ν, θ with one of these non-zero such that

$$\kappa\pi_{W_0'}(\widetilde{u}) + \nu\pi_{W_0'}(\widetilde{u}_1) + \theta\pi_{W_0'}(\widetilde{d}_1) = 0 \Rightarrow \kappa u + \nu u_1 + \theta d_1 \in W_0$$

From what we showed at the beginning of this proof, we can conclude that κ, ν are non-zero. $\theta \neq 0$ since $\pi_{W_0'}(\widetilde{u}), \pi_{W_0'}(\widetilde{u}_1)$ are distinct. Put $d_1 = \delta_1 l_1 + \ldots + \delta_k l_k + \delta_{k+1} e$ with $\delta_{k+1} \neq 0$, then the above equation becomes

$$\kappa u + \nu u_1 + \theta\delta_{k+1} e \in W_0$$

Taking projection onto $\tilde{W}_0$ for the decomposition $W_0 \oplus \tilde{W}_0 = V$ and normalizing their coefficients of l_{k+1} when they are written in basis $\mathcal{B}$

$$\kappa\pi_{\tilde{W}_0}(\widehat{u}) + \nu\pi_{\tilde{W}_0}(\widehat{u}_1) + \theta\pi_{\tilde{W}_0}(\widehat{d}_1) = 0$$

Since coefficient of l_{k+1} is 1 in all of them and $\nu \neq 0$ we get that

$$\pi_{\tilde{W}_0}(\widehat{u}_1) \in fl(\{\pi_{\tilde{W}_0}(\widehat{u}), \pi_{\tilde{W}_0}(\widehat{d}_1)\}) = \vec{L}_1$$

Since $|\vec{L}_1 \cap \pi_{\tilde{W}_0}(\widehat{\mathcal{L}(U^\star_{1-i})})| = 1 \Rightarrow \pi_{\tilde{W}_0}(\widehat{u}) = \pi_{\tilde{W}_0}(\widehat{u}_1) \neq 0 \Rightarrow \exists \delta, \psi$ both non-zero such that $\delta u + \psi u_1 \in W_0$. We could eliminate u_1 to conclude that there exist constants α, β with $\beta \neq 0$ such that $\alpha u + \beta d_1 \in W_0 \Rightarrow \pi_{W_0'}(\widetilde{d}_1) = \pi_{W_0'}(\widetilde{u})$ which cannot happen by what we showed in the beggining of the proof or $\pi_{W_0'}(d_1) = 0 \Rightarrow d_1 \in sp(\{l_1, \ldots, l_k\})$ which is a contradiction to (S, D) being a detector pair. Therefore such a u_1 does not exist and $|\vec{L}_2 \cap \pi_{W_0'}(\widetilde{\mathcal{L}(U^\star_{1-i})})| = 1$. □

D Proofs from Section 4

All random selections are done from the set $[N] = \{1, \ldots, N\}$.

Lemma D.1. *Let $\mathbb{R}^n$ be the n dimensional vector space over $\mathbb{R}$. Suppose $v_i : i = 1, \ldots, n$ are n vectors in $\mathbb{R}^n$ with each co-ordinate chosen independently from the uniform distribution on $[N]$. Consider the event*

$$\mathcal{E} = \{\{v_1, \ldots, v_n\} \text{ are LI }\}$$

Then $Pr[\mathcal{E}] \geq 1 - \frac{n}{N^{n^2}}$.

Proof. Each $v_i \in \mathbb{R}^n$ is chosen such that each co-ordinate is chosen uniformly randomly from the set $[N]$. Let v_i be the vector $(V_{i,1}, \ldots, V_{i,n})$. Consider the matrix $\tilde{V} = (V_{i,j})$. The v_i's will be linearly independent if and only if $\tilde{V}$ is invertible i.e. $det(V_{i,j}) \neq 0$. Note that $det(V_{i,j})$ is not the zero polynomial since the monomial $v_1^1 v_2^2 .. v_n^n$ has coefficient 1. Now we can use Schwartz-Zippel Lemma [Sax09] on this polynomial to yield:

$$Pr[det(\tilde{V}) = 0] \leq \frac{n}{N^{n^2}}$$

Therefore $Pr[v_i, i = 1, \ldots n \text{ are LI }] = Pr[det(\tilde{V}) \neq 0] \geq 1 - \frac{n}{N^{n^2}}$. Therefore $Pr[\mathcal{E}] \geq 1 - \frac{n}{N^{n^2}}$. □

Lemma D.2. *Assume conditions in the previous lemma. Consider the subspaces* $V = sp\{v_1, \dots, v_r\}$ *and* $V' = sp\{v_{r+1}, \dots, v_n\}$. *Let's assume that that* $\mathcal{E}$ *occurs. So* $dim(V) = r$. *We know Then* $\mathbb{R}^n = V \oplus V'$. *Let* $\pi_V : \mathbb{R}^n \to V$ *be the orthogonal projection onto* V *under this decomposition . Let* $T \subset \mathbb{R}^n$ *be a finite set from which linear forms are chosen. Consider the event*

$$\mathcal{F} = \{\exists \textit{ an LI set } \{l_1, \dots, l_r\} \subset T \textit{ such that } \{\pi_V(l_1), \dots, \pi_V(l_r)\} \textit{ is LD } \}$$

Then $Pr[\mathcal{F}] \leq \binom{|T|}{r}\{\frac{n}{N^{n^2}} + \frac{r(n-1)}{N^{n^2}}\}$

Proof. Fix $\{l_1, \dots, l_r\} \subset T$ an LI set. Extend it to get a basis $\{l_1, \dots, l_n\}$ of $\mathbb{R}^n$. Let $l_i = \sum\limits_{j\in[n]} L_{i,j}e_j$. Let L be the matrix $(L_{i,j})_{(i,j)\in[n]\times[n]}$. From the discussion above we have $\tilde{V} = (V_{i,j})$. Now let P_r be the $n \times n$ matrix

$$P_r = \begin{bmatrix} I_r & 0_{r,n-r} \\ 0_{n-r,r} & 0_{n-r,n-r} \end{bmatrix}$$

where I_r is the $r \times r$ identity matrix and $0_{p,q}$ is the $p \times q$ matrix with all 0 entries. Also for any $n \times n$ matrix A, define $M_r(A)$ to be the principal $r \times r$ minor of A. Consider the equation given by

$$det(M_r(P_rLco(\tilde{V}))) = 0$$

where $co(\tilde{V})$ is the co-factor matrix of $\tilde{V}$. Since entries of $co(\tilde{V})$ are polynomials in the $V_{i,j}$'s and L is a fixed matrix, the entries of $P_rLco(\tilde{V})$ are polynomials in $V_{i,j}$'s. So $det(M_r(P_rLco(\tilde{V})))$ is a polynomial in $V_{i,j}$'s. This polynomial can't be identically 0. Choose $V_{i,j} = L_{i,j}$, then $\tilde{V}$ is invertible and $Lco(\tilde{V}) = det(L)I$ and so $P_rLco(\tilde{V}) = det(L)P_r \Rightarrow det(M_r(P_rLco(\tilde{V}))) = det(L) \neq 0$. Degree of the polynomial $det(M_r(P_rLco(\tilde{V})))$ is clearly $\leq r(n-1)$. Therefore by Schwartz Zippel Lemma

$$Pr[det(M_r(P_rLco(\tilde{V}))) = 0] \leq \frac{r(n-1)}{N^{n^2}}$$

Consider the set

$$S(\{l_1, \dots, l_r\}) = \{(V_{i,j}) : det(\tilde{V}) \neq 0, det(M_r(P_rLco(\tilde{V})) \neq 0\}$$

On this set $S(\{l_1, \dots, l_r\})$, $\{v_1, \dots, v_n\}$ is a basis and we have the following matrix equations :

$$\begin{bmatrix} v_1 \\ \cdot \\ \cdot \\ v_n \end{bmatrix} = \tilde{V} \begin{bmatrix} e_1 \\ \cdot \\ \cdot \\ e_n \end{bmatrix} \text{ and } \begin{bmatrix} l_1 \\ \cdot \\ \cdot \\ l_n \end{bmatrix} = L \begin{bmatrix} e_1 \\ \cdot \\ \cdot \\ e_n \end{bmatrix} \Rightarrow \begin{bmatrix} l_1 \\ \cdot \\ \cdot \\ l_n \end{bmatrix} = L\tilde{V}^{-1} \begin{bmatrix} v_1 \\ \cdot \\ \cdot \\ v_n \end{bmatrix}$$

and so

$$\begin{bmatrix} \pi_V(l_1) \\ \cdot \\ \pi_V(l_r) \end{bmatrix} = \frac{1}{det(\tilde{V})} M_r(P_rLco(\tilde{V})) \begin{bmatrix} v_1 \\ \cdot \\ v_r \end{bmatrix}$$

Therefore $\{\pi_V(l_1), \dots, \pi_V(l_r)\}$ is an LI set. Now $S(\{l_1, \dots, l_r\})^c = \{(V_{i,j}) : det(\tilde{V}) = 0 \text{ or } det(M_rLco(M)) = 0\} \Rightarrow Pr[S(\{l_1, \dots, l_r\})^c] \leq \frac{n}{N^{n^2}} + \frac{r(n-1)}{N^{n^2}}$. Next we vary $\{l_1, \dots, l_r\}$ and apply union bound to get

$$Pr[\mathcal{F}] \leq \sum_{\{l_1,\dots,l_r\}\subset T} S(\{l_1, \dots, l_r\})^c \leq \binom{|T|}{r}\{\frac{n}{N^{n^2}} + \frac{r(n-1)}{N^{n^2}}\}$$

In our application $|T| = poly(d)$ and r is a constant, so we choose $N = 2^{d+n}$ and make this probability very small.

$\square$

Lemma D.3. *Let* $f|_V(\bar{X}) = \sum_{\{\bar{\alpha}:|\bar{\alpha}|=d\}} a_{\bar{\alpha}}\bar{X}^{\bar{\alpha}}$ *be a homogeneous multivariate polynomial of degree* d *in* r *variables* $X_1, \ldots, X_r$. *Let* $p_i : 1 \leq i \leq \binom{d+r-1}{r-1}$ *be randomly chosen points in* V *(dimension* r *random subspace of* $\mathbb{R}^n$ *chosen in the above lemmas). Then with high probability one can find all the* $a_{\bar{\alpha}}$.

Proof. We evaluate the polynomial at each of the p_i's. So we have $\binom{d+r-1}{r-1}$ evaluations. The number of coefficients is also $\binom{d+r-1}{r-1}$ so we get a linear system in the coefficients where the matrix (X) entries are just monomials evaluated at the p_i's. Since f is not identically zero clearly there exist values for the points p_i's such that the determinant of this matrix is non zero polynomial so it cannot be identically zero. Now the degree of the determinant polynomial is bounded by $d\binom{d+r-1}{r-1} \leq poly((d+r)^r)$. So by Schwarz Zippel lemma

$$Pr[a_{\bar{\alpha}} \text{ is recovered correctly }] = Pr[det(X) \neq 0] \geq 1 - \frac{poly(d^r)}{N^{n^2}}$$

□

References

[Agr05] Manindra Agrawal. Proving lower bounds via pseudo-random generators. In *FSTTCS 2005: Foundations of Software Technology and Theoretical Computer Science, 25th International Conference, Hyderabad, India, December 15-18, 2005, Proceedings, volume 3821 of Lecture*, pages 92–105. Springer, 2005.

[AMS08] V. Arvind, Partha Mukhopadhyay, and Srikanth Srinivasan. New results on noncommutative and commutative polynomial identity testing. *2012 IEEE 27th Conference on Computational Complexity*, 0:268–279, 2008.

[BDWY11] B. Barak, Z. Dvir, A. Wigderson, and A. Yehudayoff. **Rank bounds for design matrices with applications to combinatorial geometry and locally correctable codes**. In *Proceedings of the 43rd annual ACM symposium on Theory of computing*, STOC '11, pages 519–528, New York, NY, USA, 2011. ACM.

[BBB$^+$00] Amos Beimel, Francesco Bergadano, Nader H. Bshouty, Eyal Kushilevitz, and Stefano Varricchio. Learning functions represented as multiplicity automata. *J. ACM*, 47(3):506–530, May 2000.

[Buc76] B. Buchberger. A theoretical basis for the reduction of polynomials to canonical forms. *SIGSAM Bull.*, 10(3):19–29, August 1976.

[DSW12] Z. Dvir, S. Saraf, and A. Wigderson. **Improved rank bounds for design matrices and a new proof of Kelly's theorem.** Forum of mathematics - Sigma (to appear), 2012.

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

[GKZ94] Izrail Moiseevitch Gelfand, Mikhail M. Kapranov, and Andrei V. Zelevinsky. *Discriminants, resultants, and multidimensional determinants*. Mathematics : theory & applications. Birkhäuser, Boston, Basel, Berlin, 1994. Autre tirage de l'édition Birkhäuser chez Springer Science+ Business Media.

[GGM86] Oded Goldreich, Shafi Goldwasser, and Silvio Micali. How to construct random functions. *J. ACM*, 33(4):792–807, August 1986.

[GKL12] Ankit Gupta, Neeraj Kayal, and Satya Lokam. Reconstruction of depth-4 multilinear circuits with top fan-in 2. In *Proceedings of the Forty-fourth Annual ACM Symposium on Theory of Computing*, STOC '12, pages 625–642, New York, NY, USA, 2012. ACM.

[GKL11] Ankit Gupta, Neeraj Kayal, and Satyanarayana V. Lokam. Efficient reconstruction of random multilinear formulas. In *IEEE 52nd Annual Symposium on Foundations of Computer Science, FOCS 2011, Palm Springs, CA, USA, October 22-25, 2011*, pages 778–787, 2011.

[GKQ14] Ankit Gupta, Neeraj Kayal, and Youming Qiao. Random arithmetic formulas can be reconstructed efficiently. *computational complexity*, 23(2):207–303, 2014.

[HS80] J. Heintz and C. P. Schnorr. Testing polynomials which are easy to compute (extended abstract). In *Proceedings of the Twelfth Annual ACM Symposium on Theory of Computing*, STOC '80, pages 262–272, New York, NY, USA, 1980. ACM.

[KT90] Erich Kaltofen and Barry M. Trager. Computing with polynomials given byblack boxes for their evaluations: Greatest common divisors, factorization, separation of numerators and denominators. *J. Symb. Comput.*, 9(3):301–320, March 1990.

[KS09a] Zohar S. Karnin and Amir Shpilka. Reconstruction of generalized depth-3 arithmetic circuits with bounded top fan-in. In *Proceedings of the 24rd Annual CCC*, pages 274–285, 2009.

[KS09b] Neeraj Kayal and Shubhangi Saraf. Blackbox polynomial identity testing for depth 3 circuits. In *Proceedings of the 2009 50th Annual IEEE Symposium on Foundations of Computer Science*, FOCS '09, pages 198–207, Washington, DC, USA, 2009. IEEE Computer Society.

[KV94] Michael Kearns and Leslie Valiant. Cryptographic limitations on learning boolean formulae and finite automata. *J. ACM*, 41(1):67–95, January 1994.

[Kha92] Michael Kharitonov. Cryptographic lower bounds for learnability of boolean functions on the uniform distribution. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory*, COLT '92, pages 29–36, New York, NY, USA, 1992. ACM.

[KS06] Adam Klivans and Amir Shpilka. Learning restricted models of arithmetic circuits. *Theory of computing*, 2(10):185–206, 2006.

[KS01] Adam R. Klivans and Daniel Spielman. Randomness efficient identity testing of multivariate polynomials. In *Proceedings of the Thirty-third Annual ACM Symposium on Theory of Computing*, STOC '01, pages 216–223, New York, NY, USA, 2001. ACM.

[Sax09] Nitin Saxena. Progress on polynomial identity testing. 2009.

[SS10] Nitin Saxena and C. Seshadhri. From sylvester-gallai configurations to rank bounds: Improved black-box identity test for depth-3 circuits. *CoRR*, abs/1002.0145, 2010.

[SS96] Robert E. Schapire and Linda M. Sellie. Learning sparse multivariate polynomials over a field with queries and counterexamples. In *In Proceedings of the Sixth Annual ACM Workshop on Computational Learning Theory*, pages 17–26, 1996.

[Shp07] Amir Shpilka. Interpolation of depth-3 arithmetic circuits with two multiplication gates. In *In STOC '07: Proceedings of the thirty-ninth annual ACM symposium on Theory of computing*, pages 284–293. ACM Press, 2007.

[SV09] Amir Shpilka and Ilya Volkovich. Improved polynomial identity testing for read-once formulas. pages 700–713, 2009.

ECCC ISSN 1433-8092

http://eccc.hpi-web.de