Electronic Colloquium on Computational Complexity, Report No. 8 (2025)

Reconstruction of Depth 3 Arithmetic Circuits with Top Fan-in 3

Shubhangi Saraf* Devansh Shringi†

February 9, 2025

Abstract

In this paper, we give the first subexponential (and in fact quasi-polynomial time) reconstruction algorithm for depth 3 circuits of top fan-in 3 ($\Sigma\Pi\Sigma(3)$ circuits) over the fields $\mathbb{R}$ and $\mathbb{C}$. Concretely, we show that given blackbox access to an n-variate polynomial f computed by a $\Sigma\Pi\Sigma(3)$ circuit of size s, there is a randomized algorithm that runs in time quasi-poly(n, s) and outputs a generalized $\Sigma\Pi\Sigma(3)$ circuit computing f. The size s includes the bit complexity of coefficients appearing in the circuit.

Depth 3 circuits of constant fan-in ($\Sigma\Pi\Sigma(k)$ circuits) and closely related models have been extensively studied in the context of polynomial identity testing (PIT). The study of PIT for these models led to an understanding of the structure of identically zero $\Sigma\Pi\Sigma(3)$ circuits and $\Sigma\Pi\Sigma(k)$ circuits using some very elegant connections to discrete geometry, specifically the Sylvester-Gallai Theorem, and colorful and high dimensional variants of them. Despite a lot of progress on PIT for $\Sigma\Pi\Sigma(k)$ circuits and more recently on PIT for depth 4 circuits of bounded top and bottom fan-in, reconstruction algorithms for $\Sigma\Pi\Sigma(k)$ circuits has proven to be extremely challenging.

In this paper, we build upon the structural results for identically zero $\Sigma\Pi\Sigma(3)$ circuits that bound their rank, and prove stronger structural properties of $\Sigma\Pi\Sigma(3)$ circuits (again using connections to discrete geometry). One such result is a bound on the number of codimension 3 subspaces on which a polynomial computed by an $\Sigma\Pi\Sigma(3)$ circuit can vanish on. Armed with the new structural results, we provide the first reconstruction algorithms for $\Sigma\Pi\Sigma(3)$ circuits over $\mathbb{R}$ and $\mathbb{C}$.

Our work extends the work of [Sinha, CCC 2016] who provided a reconstruction algorithm for $\Sigma\Pi\Sigma(2)$ circuits over $\mathbb{R}$ and $\mathbb{C}$ as well as the works of [Shpilka, STOC 2007] who provided a reconstruction algorithms for $\Sigma\Pi\Sigma(2)$ circuits in the setting of small finite fields, and [Karnin-Shpilka, CCC 2009] who provided reconstruction algorithms for $\Sigma\Pi\Sigma(k)$ circuits in the setting of small finite fields.

*Department of Mathematics & Department of Computer Science, University of Toronto, Toronto, Canada. Research partially supported by an NSERC Discovery Grant and a McLean Award. Email: shubhangi.saraf@utoronto.ca

†Department of Computer Science, University of Toronto, Toronto, Canada. Email: devansh@cs.toronto.edu

ISSN 1433-8092

1 Introduction

Arithmetic circuits are directed acyclic graphs (DAGs) that compute multivariate polynomials in a compact form, constructing these polynomials from variables using addition ($+$) and multiplication ($\times$) operations.

Reconstruction of arithmetic circuits is the following problem: given black-box or oracle access to a polynomial computed by a circuit C of size s from a certain class of circuits $\mathcal{C}$, design an efficient algorithm (either deterministic or randomized) to recover a circuit that computes the same polynomial as C. This question is the algebraic equivalent of exact learning in Boolean circuit complexity [Ang88]. If it is additionally required that the output circuit belongs to the same class $\mathcal{C}$ as the input circuit, this process is referred to as *proper learning*.

Reconstruction of arithmetic circuits is a fundamental and challenging problem. In recent years, there has been a flurry of works on developing reconstruction algorithms for various interesting subclasses of arithmetic circuits [BBB+00, KS01, KS06, FS12].

Thanks to the depth reduction results from [AV08, Koi10, Tav13, GKKS13], we now know that even depth-3 and depth-4 circuits are quite expressive. Thus, efficient reconstruction algorithms even for depth three circuits would have significant implications for more general circuit models. Perhaps not surprisingly, we are quite far from achieving efficient reconstruction algorithms for general depth-3 circuits. However, in recent years there have been several works developing reconstruction algorithms for restricted yet interesting subclasses of depth 3 ($\Sigma\Pi\Sigma$) and depth 4 ($\Sigma\Pi\Sigma\Pi$) circuits [Shp07, KS09a, Sin16b, Sin22, BSV21, PSV24, BS24, GKL12, BSV20].

A closely related problem is that of blackbox polynomial identity testing (PIT) which asks for the following. Given blackbox access to a polynomial f computed by some circuit C of size s from some class $\mathcal{C}$, the goal is to decide if f is identically zero. In other words, the goal is to compute an explicit hitting set for the class $\mathcal{C}$[1].

It is easy to see that obtaining deterministic reconstruction algorithms for a class of circuits $\mathcal{C}$ is at least as hard is derandomizing black-box PIT for $\mathcal{C}$. Even randomized reconstruction almost always requires some deep understanding of the structure of the underlying circuit class and in almost every case we know, it seems harder than derandomizing PIT for that class. Indeed for most circuits classes that have been studied, efficient PIT algorithms have been a precursor to understanding reconstruction algorithms for that class. Since reconstruction for depth 3 circuits of constant top fan-in ($\Sigma\Pi\Sigma(k)$ circuits) is the main focus of this paper, we first give some context by describing what is know about PIT for this class.

PIT for $\Sigma\Pi\Sigma(k)$ circuits The recent breakthrough work of [LST22] gives the first subexponential deterministic blackbox PIT for $\Sigma\Pi\Sigma$ (and in fact any constant depth) circuits. If we want truly polynomial time blackbox derandomization, then we only know how do this for restricted classes of depth 3 circuits. When the the top fan-in of the output sum gate is a constant k, then this class is referred to as the class of $\Sigma\Pi\Sigma(k)$ circuits. $\Sigma\Pi\Sigma(k)$ circuits have received a great deal of attention in the context of blackbox PIT, and there has been a large body of beautiful works eventually showing polynomial time blackbox PIT algorithms for $\Sigma\Pi\Sigma(k)$ circuits [DS05, KS08, KS09b, SS13, SS11]. A running theme through several of these works is to show that identically zero $\Sigma\Pi\Sigma(k)$ circuits have some very interesting structure; they must be *low rank*. Along the way some very elegant connections to discrete geometry, specifically the Sylvester-Gallai Theorem, and colorful and high dimensional variants of them were developed and used. In the last few years, there have been several exciting works trying obtaining similar results for interesting subclasses of depth-4 circuits,

[1] With randomness, this problem is easy using the Schwartz-Zippel Lemma [Sch80, Zip79]

in particular for $\Sigma^k\Pi\Sigma\Pi^r$ which are depth 4 circuits with bounded top and bottom fan-in. A sequence of results [Shp19, PS22a, PS21, PS22b, GOS22] developed a beautiful theory of Sylvester-Gallai type configurations for quadratic polynomials and was able to obtain a polynomial time deterministic PIT result for $\Sigma^3\Pi\Sigma\Pi^2$ (think of $\Sigma\Pi\Sigma(3)$ circuits but with product of quadratics computed at the second layer of gates instead of a product of linear forms). Extending this to larger k and r is a very interesting direction and partial results in this direction have been obtained in [OS22, GOPS23, OS24]. Using completely different techniques, a remarkable work by [DDS21] gives quasipolynomial blackbox PIT for $\Sigma^k\Pi\Sigma\Pi^r$ circuits for any constants k and r.

Reconstruction for $\Sigma\Pi\Sigma(k)$ circuits Despite all this progress for PIT, far less is known for reconstruction of $\Sigma\Pi\Sigma(k)$ circuits even when the algorithms are allowed to be randomized.

For now let us assume the underlying field has characteristic 0. In particular let us assume the coefficients lie in $\mathbb{R}$ or $\mathbb{C}$. Without additional restrictions like multilinearity and set-multilinearity, we essentially only how to do efficient reconstruction of $\Sigma\Pi\Sigma(k)$ circuits over infinite fields like $\mathbb{R}$ and $\mathbb{C}$ when $k = 2$ [Sin16b]! The result in [Sin16b] gives a randomized $\text{poly}(n, d)$ time reconstruction algorithm for n variate, degree d polynomials represented by a $\Sigma\Pi\Sigma(2)$ circuit over $\mathbb{R}$ or $\mathbb{C}$.

Note that when $k = 2$, then derandomizing PIT is easy, and it only starts becoming challenging once $k \geq 3$. However reconstruction is much more challenging and despite all the progress on the PIT front, it took a really long time to get efficient reconstruction for $\Sigma\Pi\Sigma(2)$ circuits. The proof in [Sin16b] is quite sophisticated and uses some really beautiful connections to discrete geometry, in particular to the robust Sylvester-Gallai theorems (inspired by the theory developed for PIT of $\Sigma\Pi\Sigma(k)$ circuits). Thus even for the seemingly simple case of $k = 2$, reconstruction can be fairly complex. Already when $k = 3$, the techniques of the above work break down and nothing nontrivial was known for reconstructing $\Sigma\Pi\Sigma(3)$ circuits over $\mathbb{R}$ or $\mathbb{C}$.

In the setting of finite fields, there are some additional very interesting results for reconstruction of $\Sigma\Pi\Sigma(k)$ circuits. Over small (only polynomially large) finite fields, the first (and very nontrivial) reconstruction algorithm for $\Sigma\Pi\Sigma(2)$ circuits was given in [Shp07]. This algorithm run time has a quasipolynomial dependence on $|\mathbb{F}|$ (it crucially needs to iterate over all field constants) and is therefore only efficient for small fields. This work was extended to $\Sigma\Pi\Sigma(k)$ circuits for any constant k in [KS09a], but again it is only efficient for small finite fields due to the quasipolynomial dependence on $|\mathbb{F}|$. When the input is an n-variate, degree d polynomial computed by a size s circuit, both the above algorithms run in quasi-$\text{poly}(n, d, |\mathbb{F}|, s)$ time. In the setting of $k = 2$, only very recently it was shown in [Sin22] that there is a polynomial time reconstruction algorithms with a run time that has a polynomial dependence on $\log |\mathbb{F}|$. In the further restricted setting where we have additional constraints of multilinearity or set-multilinearity[2], there has been a large body of work on reconstruction algorithms for $\Sigma\Pi\Sigma(k)$ circuits [BBB+00, Shp07, KS09a, BSV21, PSV24, BS24].

Thus to summarize, over large fields, or infinite fields such as $\mathbb{R}$ or $\mathbb{C}$, we knew no nontrivial reconstruction algorithms for general $\Sigma\Pi\Sigma(k)$ circuits even for $k = 3$. The main result of this paper is to give the first efficient reconstruction algorithm for $\Sigma\Pi\Sigma(3)$ circuits over infinite fields such as $\mathbb{R}$ and $\mathbb{C}$.

Our result (informal): *Given blackbox access to an n-variate degree d polynomial f over $\mathbb{R}$ or $\mathbb{C}$, computed by a $\Sigma\Pi\Sigma(3)$ circuit, there is a randomized quasi-poly(n, d, s) time reconstruction algorithm for f, where s is the maximum bit complexity of any constant appearing in the circuit*

Before we state our results more formally, we first introduce some definitions and notions related

[2]This setting captures tensor reconstruction for constant rank tensors

to $\Sigma\Pi\Sigma(k)$ circuits.

Some definitions related to $\Sigma\Pi\Sigma(k)$ circuits: The model of depth-3 arithmetic circuits with top fan-in k, which we refer as $\Sigma\Pi\Sigma(k)$ circuits, has three layers of alternating Σ and Π gates and computes a polynomial of the form

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

where the $l_{ij}(\bar{x})$-s are linear polynomials.

We will in fact assume that the circuits are homogeneous and all the d_i's are actually the same. This is because for the purpose of reconstruction and PIT, one can easily reduce to the homogeneous setting (see discussion in Lemma 3.1).

We say that the circuit is simple if $\gcd\{T_i | i \in [k]\} = 1$ and minimal if for all proper subsets $S \subset [k]$, $\sum_{i \in S} T_i \neq 0$. We define $\gcd(C) = \gcd(T_1, \ldots, T_k)$. The simplification of C, denoted by $\mathrm{sim}(C)$, is defined as $C/\gcd(C)$. We define the rank of a circuit ($\mathrm{rank}(C)$) as the dimension of the space spanned by all the linear forms in the circuit $\dim(\mathrm{span}(\{l_{i,j} : i \in [k], j \in [d_i]\}))$. We will often be concerned with $\mathrm{rank}(\mathrm{sim}(C))$.

A generalized depth 3 circuit $\Sigma\Pi\Sigma(k, d, r)$ is of the form

$$C = \sum_{i=1}^{k} \left(\prod_{j=1}^{d_i} l_{ij} \cdot h_i(\bar{l}_{i1}, \ldots, \bar{l}_{ir}) \right)$$

where l_{ij},$\bar{l}_{ik}$ are linear forms in $\mathbb{F}[x_1, \ldots, x_n]$ and $d = max_i(d_i + deg(h_i))$. Notice that when r is small (say constant or $O(\log d)$, the representation looks like a $\Sigma\Pi\Sigma(k)$ circuit where every product gate is further multiplied by a polynomial in few (r) linear forms.

Our techniques: Rank bounds and connections to discrete geometry As mentioned previously, several of the blackbox PIT results for $\Sigma\Pi\Sigma(k)$ circuits and related models follows from some insight into the structure of identically zero $\Sigma\Pi\Sigma(k)$ circuits. One such remarkable structural result which is also a central ingredient in our proof is that identically zero simple and minimal $\Sigma\Pi\Sigma(k)$ circuits over $\mathbb{R}$ or $\mathbb{C}$ must be of only constant rank [KS09b, SS11, SS13]. In particular, if a simple and minimal $\Sigma\Pi\Sigma(3)$ circuit computes the identically zero polynomial, then the set of all linear forms appearing at any gates in the circuit can only span a constant dimensional space.

In this paper, we develop a deeper understanding of some structural properties of $\Sigma\Pi\Sigma(3)$ circuits. Since we need to learn the linear forms in a given underlying $\Sigma\Pi\Sigma(3)$ circuit (not just determine whether the polynomial is zero or nonzero), thus we need also to understand and develop structural properties of *non-zero* $\Sigma\Pi\Sigma(3)$ circuits. One example of such a structural result we show is that if a polynomial f is computed by an $\Sigma\Pi\Sigma(3)$ circuit (which has some mild non-degeneracy property) then the number of codimension 3 subspaces on which it can vanish is polynomially bounded (see Lemma 4.4).

1.1 Our Results

In this paper, we give the first subexponential time (and in fact quasipolynomial time) algorithm for reconstructing $\Sigma\Pi\Sigma(3)$ circuits over $\mathbb{R}$ and $\mathbb{C}$. When the three multiplication gates in our circuit are sufficiently distant, i.e. when $\forall i, j \in [3], i \neq j, \mathrm{rank}(\mathrm{sim}(T_i + T_j)) \geq c \log d$ for some absolute constant c, then our algorithm does "proper learning", i.e. its output is the unique $\Sigma\Pi\Sigma(3)$ circuit

computing f. If this distance property does not hold, then our algorithm outputs a generalized depth 3 circuit of top fan-in at most 2 with parameters $\Sigma\Pi\Sigma(2, d, c\log d)$. We state our main theorem below. The running time in the statement is suppressing a $\text{poly}(s)$ dependence on the max bit complexity s of any constant appearing in the circuit C.

Theorem 1.1. *Let $\mathbb{F}$ be a field that is $\mathbb{R}$ or $\mathbb{C}$. Let $f \in \mathbb{F}[x_1, \ldots, x_n]$ be a degree d polynomial computed by $\Sigma\Pi\Sigma(3)$ circuit of the form $C = T_1 + T_2 + T_3$. There exist an absolute constant $c > 0$ such that the following holds. There is a randomized algorithm that runs in $(nd)^{O(\log d)}$ time, makes blackbox queries to f, and with probability $1 - o(1)$ does the following:*

1. *If $\forall i, j \in [3], i \neq j, \text{rank}(\text{sim}(T_i + T_j)) \geq c\log d$ then it outputs a $\Sigma\Pi\Sigma(3)$ circuit computing f.*
2. *If $\exists i, j \in [3], i \neq j$, such that $\text{rank}(\text{sim}(T_i + T_j)) < c\log d$ then it outputs a $\Sigma\Pi\Sigma(2, d, c\log d)$ generalized depth 3 circuit computing f.*

Remark 1.2 (Dependence on bit complexity)**.** *If s is the maximum bit complexity of any coefficient appearing in C, then our algorithm run time also depends polynomially on s. In the statement of the above theorem and later in the paper, we have suppressed the $\text{poly}(s)$ dependence in the running time for clarity of exposition.*

Remark 1.3 (Proper vs improper learning)**.** *Note that our algorithm is a proper learning algorithm only when every pair of multiplication gates had enough "distance". Otherwise, the output came from the model of generalized depth 3 circuits. All prior works on reconstruction of $\Sigma\Pi\Sigma(2)$ circuits and $\Sigma\Pi\Sigma(k)$ circuits ([Shp07, KS09a, Sin16b, Sin22]) also had a similar kind of output.*

We discuss the open problems in Section 8.

1.2 Other related works

As the case of general $\Sigma\Pi\Sigma(k)$ is considered hard, several restrictions of the model have been considered for reconstruction. Some well-studied models are powering depth-3 circuits $\Sigma\wedge\Sigma(k)$, multilinear $\Sigma\Pi\Sigma(k)$, and set-multilinear $\Sigma\Pi\Sigma(k)$ circuits. The restrictions of powering circuits $\Sigma\wedge\Sigma(k)$ and set-multilinear $\Sigma\Pi\Sigma(k)$ have received special attention due to their connections with finding symmetric tensor decomposition and tensor decomposition problems [BBB+00, Shp07, KS09a, BSV21, PSV24, BS24].

For the model of depth 2 circuits, $\Sigma\Pi$, the problem of reconstruction is equivalent to sparse multivariate interpolation for which we have a polynomial time algorithm in [BOT88]. The work of [BSV20] studied multilinear depth-4 circuits with bounded top fan-in ($\Sigma\Pi\Sigma\Pi(k)$ circuits), and gave deterministic reconstruction algorithms which ran in $\text{poly}(n, d, |\mathbb{F}|)$ time. The running time is however still at least $\text{poly}(|\mathbb{F}|)$, and hence it does not work over large/infinite fields. Note that when the top fan-in is 2, i.e. for $\Sigma\Pi\Sigma\Pi(2)$ circuits, we do know such efficient polynomial-time reconstruction algorithms by the work of [GKL12]. Read-once oblivious branching programs (ROABPs) are another model that have been well studied in the context of reconstruction. In [KS06], the authors presented a randomized reconstruction (proper learning) algorithm for ran in time $\text{poly}(n, d, s)$. This was derandomized in [FS12], giving a deterministic quasi-$\text{poly}(n, d, s)$ reconstruction algorithm.

Recently, there have also been several works studying *average case* learning algorithms for arithmetic circuits. In [KS19], the authors give a $\text{poly}(n, d, k)$ time reconstruction algorithm for non-degenerate homogeneous depth three circuits $\Sigma\Pi\Sigma(k)$ circuits. A $\text{poly}(n, d, s)$ learning algorithm for generalized depth three circuits in the non-degenerate case is presented in [BGKS22]. Reconstruction algorithms for other constant depth circuits in the non-degenerate case have also been obtained in [GKL11, GKQ14, KNST17, KNS19, GKS20].

2 Proof Overview

Let f be a polynomial that has a $\Sigma\Pi\Sigma(3)$ representation and let

$$C = T_1 + T_2 + T_3$$

be a $\Sigma\Pi\Sigma(3)$ circuit computing f. Thus each T_i is a product of linear forms, and as we describe in the preliminaries, with some simple preprocessing, we can assume that the circuit and all gates within it are homogeneous (see Lemma 3.1). In general, the gates T_i might have nontrivial gcd, which has to be dealt with, but for the purpose of the proof overview, let us assume that $\gcd(T_1, T_2, T_3) = 1$. Note that we cannot easily reduce to the case of $\gcd(T_1, T_2, T_3) = 1$ by factoring and dividing out the linear factors since there might be linear factors which do not divide the gcd, and division by those factors might not preserve the property of the polynomial being representable by a $\Sigma\Pi\Sigma(3)$ circuit.

In order to reconstruct the circuit C, we need to somehow try and learn the linear forms appearing in C. What we have is (randomized) access to a blackbox computing f.

Notice that if l_1 is a linear form dividing T_1, l_2 is a linear form dividing T_2 and l_3 is a linear form dividing T_3 then if we go modulo l_1, l_2 and l_3, then the polynomial f vanishes identically. In other words, for any input where l_1, l_2 and l_3 evaluate to 0, f evaluates to 0 as well.

Let $\mathcal{S}_3(f)$ to be the set of all codimension 3 subspaces of $\mathbb{F}^n$ over which f vanishes. Then if we could somehow "learn" all the spaces in this set, then one of them would correspond to $\mathbb{V}(l_1, l_2, l_3)$ i.e. the codimension 3 space where l_1, l_2, l_3 vanish (or evaluate to 0). Thus, a starting challenge for us is to show that the set $\mathcal{S}_3(f)$ can be learned. It turns out that the set can be infinite. Suppose $\mathbb{V}(l, l')$ is some codimension 2 space on which f vanishes. Then any codimension 3 space contained within $\mathbb{V}(l, l')$ will also be a space on which f vanishes and hence be contained in $\mathcal{S}_3(f)$. This makes the set unwieldy to deal with and hence we modify the definition.

Let $\mathcal{S}_3(f)$ to be the set of all codimension 3 subspaces of $\mathbb{F}^n$ over which f vanishes, such that it is not contained within any codimension 1 or 2 space on which f vanishes. One of our significant structural results is to show that other than in certain degenerate settings, $\mathcal{S}_3(f)$ is finite and in fact has at most $\text{poly}(d)$ distinct elements. This is indeed the first major ingredient of our proof and we prove this in Section 4. We also show that $\mathcal{S}_2(f)$, which is defined similarly with codimension 2 spaces, is finite, and has at most $\text{poly}(d)$ distinct elements. This is the starting point of our analysis.

Once we prove that $\mathcal{S}_3(f)$ and $\mathcal{S}_2(f)$ are finite and polynomially bounded (other than in degenerate settings), the next thing we show is how to actually compute $\mathcal{S}_3(f)$ and $\mathcal{S}_2(f)$. Once we have these sets, we then use them to learn the linear forms appearing in C.

Our reconstruction algorithm for $\Sigma\Pi\Sigma(3)$ circuits over $\mathbb{R}$ or $\mathbb{C}$ (and proof of correctness of the algorithm) follows from the following broad outline

1. Obtain an upper bound on the number of a codimension 2 subspaces ($\mathcal{S}_2(f)$) and codimension 3 subspaces ($\mathcal{S}_3(f)$) on which f vanishes (other than in some degenerate settings). This is shown in Section 4.

2. Algorithmically compute $\mathcal{S}_2(f)$ and $\mathcal{S}_3(f)$. This is shown in Section 5. At a high level, we consider projections of the circuit to constantly many variables, compute $\mathcal{S}_2$ and $\mathcal{S}_3$ for the constant variate polynomials by solving a suitable system of polynomial equations for each projection, and then "glue" or "lift" the solutions to a global solution over the entire original space. The ideas are inspired by the algorithms in [Sin22]. However there are some additional nontrivial challenges that arise in our setting.

3. Use $\mathcal{S}_2(f)$ and $\mathcal{S}_3(f)$ to form a list of linear forms (which we call $\mathcal{L}_{cand}$) such that several of these linear forms actually divide one of the gates of the circuit. We will do this in settings where $\mathcal{S}_2(f)$ and $\mathcal{S}_3(f)$ can be computed as well as in the degenerate cases where they cannot, and for this we will use the structure of the degeneracy. This is shown in Section 6 and is the most technically complex part of the proof.

4. Reconstruct the entirety of the circuit using the few linear forms learned in the previous part. This is achieved by going modulo the linear forms and learning the projected circuit of top fan-in at most 2, and then gluing the projections to recover the original circuit. This part uses several ideas from the reconstruction algorithms of Shpilka [Shp07] and Karnin-Shpilka [KS09a].

We will discuss each of the four parts in some more detail in the following subsections below.

2.1 Overview: Upper bounding the size of $\mathcal{S}_2(f)$ and $\mathcal{S}_3(f)$

For $\mathcal{S}_1(f)$, which is defined to be the number of codimension 1 spaces over which f vanishes, bounding its size is easy since, each member of $\mathcal{S}_1(f)$ corresponds to a linear factor of f, and there can be at most d of those.

We now give a flavor of what goes into bounding $\mathcal{S}_2(f)$. We will only be able to bound $\mathcal{S}_2(f)$ when the circuit $C = T_1 + T_2 + T_3$ is such that $\text{rank}(\text{sim}(C)) > c_2$ for some absolutely constant c_2 which depends on the rank bound for identically zero $\Sigma\Pi\Sigma(3)$ circuits.

By assumption, let $\text{rank}(\text{sim}(C)) > c_2$. For any $\mathbb{V}(l, l')$ which is an $\mathcal{S}_2(f)$ space, let $C' = C \mod \langle l_1, l_2 \rangle$. The C' computes a polynomial that is identically zero. Thus by rank bounds for identically zero $\Sigma\Pi\Sigma(3)$ circuits (see Theorem 3.4), it holds that $\text{rank}(\text{sim}(C')) < \mathcal{R}(3)$ which is an absolute constant much smaller than c_2. Suppose we are only trying to bound the number of those $\mathbb{V}(l, l') \in \mathcal{S}_2(f)$ over which none of the individual T_i vanish (the other case is not so hard to bound) then the fact that $\text{rank}(\text{sim}(C \mod \langle l_1, l_2 \rangle))$ is much smaller than $\text{rank}(\text{sim}(C))$ means that many triples of linear forms coming from distinct gates must "collapse" and become identical when we go mod $\langle l, l' \rangle$. Thus they will no longer contribute to the rank of $\text{sim}(C')$ as they will be in the gcd. (If there is no movement to the gcd, then the overall rank can reduce by at most two). Now if l_1 which divides T_1, l_2 which divides T_2 and l_3 which divides T_3 are three linearly independent linear forms that become identical mod $\langle l, l' \rangle$ then it must hold that $\text{span}(l, l') \subseteq \text{span}(l_1, l_2, l_3)$. Suppose this happens for another triple l'_1, l'_2, l'_3 which is linearly independent and such that $\text{span}(l_1, l_2, l_3)$ is distinct from $\text{span}(l'_1, l'_2, l'_3)$. Then, since $\text{span}(l, l')$ must belong to the spans of both triples, it must hold that $\text{span}(l, l')$ is in fact equal to $\text{span}(l_1, l_2, l_3) \cap \text{span}(l'_1, l'_2, l'_3)$. Thus $l_1, l_2, l_3, l'_1, l'_2, l'_3$ jointly determine $\text{span}(l, l')$ and hence, given the circuit C, there are only $O(d^6)$ choices for $\text{span}(l, l')$.

Note that we haven't covered all cases. It could be that the triples which collapse and move to the gcd are not linearly independent and they only span 2-dimensional spaces. This case needs to be handled separately. Also the case where one of the gates (say T_1) identically vanishes over $\mathbb{V}(l, l')$ has to be dealt with. In each of these cases we are able to bound the number of such spaces in $\mathcal{S}_2(f)$ and thus we get a polynomial bound on $|\mathcal{S}_2(f)|$.

Bounding $|\mathcal{S}_3(f)|$ is significantly more challenging, and the analysis breaks down into a larger number of cases. Also, we are not able to bound the size of $\mathcal{S}_3(f)$ whenever $\text{rank}(\text{sim}(C))$ is large, unlike the bounding of $|\mathcal{S}_2(f)|$. Notice that even if $\text{rank}(\text{sim}(C))$ is large, there could exists a linear form l such that $\text{rank}(\text{sim}(C \mod l))$ is quite small. In this case perhaps $f \mod l$ which is computed by $C \mod l$ can vanish on a large (maybe infinite) set of codimension 2 spaces. Then along with l, this will give us a large (possibly unbounded) set of codimension 3 spaces that f vanishes on in $\mathcal{S}_3(f)$. Thus we are only able to bound $\mathcal{S}_3(f)$ when we have the added condition

that there is no linear form l such that rank(sim($C \mod l$)) is small. This is the non-degeneracy condition that we alluded to earlier.

For circuits C that start off with rank(sim(C)) being large but such that there exists a linear form l such that rank(sim($C \mod l$)) is small, we call these circuits "special form" circuits (see Definition 10). We are unable to bound the number of $\mathcal{S}_3$ spaces for polynomials computed by such circuits, but nevertheless, we show that such circuits have other additional nice properties that we will eventually exploit to learn them.

Comparison to [Sin22] In [Sin22], the authors develop a similar structural result where they bound the number of codimension 2 vanishing spaces for the non-linear part for a $\Sigma\Pi\Sigma(2)$ circuit. The structure of $\Sigma\Pi\Sigma(2)$ circuits is simpler. Note for instance that derandomizing PIT for $\Sigma\Pi\Sigma(2)$ circuits is trivial whereas derandomizing PIT for $\Sigma\Pi\Sigma(3)$ circuits is more challenging and is closely related to the rank bound. Indeed (unlike in [Sin22]) we need to crucially use the rank bound to bound vanishing spaces, and the analysis is more intricate.

2.2 Overview: Algorithmically computing $\mathcal{S}_2(f)$ and $\mathcal{S}_3(f)$

We will only show how to compute $\mathcal{S}_2(f)$ and $\mathcal{S}_3(f)$ in the settings where we have proved that their size is polynomially bounded.

Our high level strategy is inspired by the algorithms in [Sin22]. However there are some additional nontrivial challenges that arise in our setting.

For the purpose of the proof overview, we focus on the case of computing $\mathcal{S}_3(f)$, and assume that we know how to compute $\mathcal{S}_2(f)$.

Constant variate case: We first show that if the underlying polynomial is constant variate, then this can be done. We do this by setting up a suitable system of polynomial equations. Setting up equations to find all l_1, l_2, l_3 (the variables are the coefficients of the monomials in l_1, l_2, l_3) such that f vanishes over the codimension 3 space $\mathbb{V}(l_1, l_2, l_3)$ is fairly straightforward. However this might have infinitely many solutions unless we ensure that $\mathbb{V}(l_1, l_2, l_3)$ does not lie within any codimension 2 space on which f vanishes. For this we first compute $\mathcal{S}_1(f)$ and $\mathcal{S}_2(f)$ and for each element in these sets we will require that some certain matrix has large rank. By introducing additional variables (just constantly many) we show how to capture these "large rank" constraints as well using polynomial equations. Thus overall we have polynomially many equations of polynomial degree, but in only constantly many variables. This can be solved (over $\mathbb{R}$ or $\mathbb{C}$) to recover all solutions.

Large variate case: In case f is over a large number of variables, we consider several distinct projections of f to the constant variate case. We learn the $\mathcal{S}_3$ spaces for each of the projections and then glue them together to recover the $\mathcal{S}_3$ spaces for f. Before performing the projections, we first apply a random linear invertible transformation to the variables of f to ensure that the projection has nice properties (such as being able to apply Hilbert irreducibility). For our proof to go through, we will require that the projections of f do not contain new linear factors (which would correspond to new $\mathcal{S}_1$ spaces not arising from projections of original $\mathcal{S}_1$ spaces) and this is easy to obtain using Hilbert irreducibility. However, crucially we will also require that the projections don't generate new $\mathcal{S}_2$ spaces. This is important since if new $\mathcal{S}_2$ spaces were generated, then potentially one could lose out on $\mathcal{S}_3$ spaces when we take a projection (since we are not able to learn codimension 3 spaces contained within an $\mathcal{S}_2$ space). Proving that new $\mathcal{S}_2$ spaces (i.e. not just the ones that are projections of $\mathcal{S}_2$ spaces of f) are not generated does not follow immediately

from Hilbert irreducibility, and indeed we are not able to prove this fact for general polynomials (though perhaps it might be true in general as well). Our proof crucially use the fact that f can be represented as a high rank $\Sigma\Pi\Sigma(3)$ circuit. Indeed a crucial ingredient in our proof (of the fact that no new $\mathcal{S}_2$ spaces are generated in the projections) is the upper bound on the number of $\mathcal{S}_3$ spaces of f.

Once we can prove that no new $\mathcal{S}_2$ spaces are generated in the projections, then it is not hard to show that every space in $\mathcal{S}_3(f)$ gets projected to a distinct space in $\mathcal{S}_3(g_i)$ for each projected polynomial g_i. Now g_i for each i is a constant variate polynomial and we can show that $\mathcal{S}_3(g_i)$ can be computed. Given the structure of how g_i are chosen, it is then not hard to see that the spaces in $\mathcal{S}_3(g_i)$ can be stitched across the different choices of g_i to recover $\mathcal{S}_3(f)$.

Comparison to [Sin22] A similar algorithm (Algorithm 7) appeared in [Sin22] for computing the set of codimension 2 vanishing spaces for $\Sigma\Pi\Sigma(2)$ circuits. Our algorithm for learning the set of codimension 3 vanishing spaces is inspired by this work, but there is one crucial difficulty/difference. In [Sin22], when $\mathcal{S}_2(f)$ is being learnt, one needs to discard those vanishing spaces that are contained in an $\mathcal{S}_1$ space. This can be easily achieved by just dividing out the linear factors. This process needs to be modified when learning $\mathcal{S}_3(f)$ since there is no way of just "factoring out" the spaces in $\mathcal{S}_2(f)$. Instead, we have to set up a modified system of polynomial equations that handles this. We also need to prove a structural result that ensures that after projecting the large variate case to the constant variate case, the bounds on the number of $\mathcal{S}_3$ spaces still holds (no linear form exists modulo which the rank of the simple part drops below c_2), and no new $\mathcal{S}_2$ spaces are generated. These issues did not arise in [Sin22].

2.3 Overview: Using $\mathcal{S}_2(f)$ and $\mathcal{S}_3(f)$ to learn some linear forms appearing in C

Recall, f is a polynomial that has a $\Sigma\Pi\Sigma(3)$ representation and let

$$C = T_1 + T_2 + T_3$$

be the $\Sigma\Pi\Sigma(3)$ circuit computing f. Each T_i is product of linear forms and for now we are assuming that $\gcd(T_1, T_2, T_3) = 1$. We will show that we are able to learn $\Omega(\log d)$ independent linear forms from one of the gates in C.

If l_1 is a linear form dividing T_1, l_2 is a linear form dividing T_2 and l_3 is a linear form dividing T_3 such that $\dim(\text{span}(l_1, l_2, l_3)) = 3$, then $\mathbb{V}(l_1, l_2, l_3)$ will belong to $\mathcal{S}_3(f)$ unless it is contained within some space in $\mathcal{S}_2(f)$ or is some space in $\mathcal{S}_1(f)$. In such a case, we will say $\mathbb{V}(l_1, l_2, l_3)$ got "blocked" by an $\mathcal{S}_1$ or $\mathcal{S}_2$ space and hence did not get learned.

Now suppose that $\mathbb{V}(l_1, l_2, l_3)$ did not get blocked and hence lies in $\mathcal{S}_3(f)$. Thus we can use $\mathcal{S}_3(f)$ to learn $\text{span}(l_1, l_2, l_3)$. Suppose there exists l_3' dividing T_3 such that $\mathbb{V}(l_1, l_2, l_3')$ is some other distinct space in $\mathcal{S}_3(f)$. Thus we can also learn $\text{span}(l_1, l_2, l_3')$. The intersection of these two spaces allows us to learn $\text{span}(l_1, l_2)$. Just like we managed to learn $\text{span}(l_1, l_2)$, if we could also somehow learn $\text{span}(l_1, l_2')$ for some other linear form l_2' dividing T_2, then we could take the intersection of $\text{span}(l_1, l_2)$ and $\text{span}(l_1, l_2')$ to learn l_1 and hence learn one of the linear forms appearing in C !

Thus intersections of kernels of spaces in $\mathcal{S}_3(f)$ can be useful in learning linear forms in C. Can this strategy always be carried out to learn linear forms? Or could it be that we cannot learn any linear forms because $\mathcal{S}_3(f)$ is empty, since every codimension 3 space on which f vanishes got blocked by a space in $\mathcal{S}_1(f)$ or $\mathcal{S}_2(f)$?

It turns out that this part proved to be surprisingly challenging to show and is the technically most difficult and intricate part of the paper. Indeed we are not able to show that intersections of

spaces in $\mathcal{S}_3(f)$ will suffice in learning linear forms. However, we do show that in most interesting cases, either intersections of kernels of $\mathcal{S}_3$ spaces, or intersections of kernels of $\mathcal{S}_2$ spaces will suffice. When these do not suffice, then we show that the underlying circuit has some other nice structure which can be exploited to learn the linear forms. We first prove some useful structural properties of $\mathcal{S}_1(f)$ and $\mathcal{S}_2(f)$.

In the rest of the proof overview, we will just try to give some high level ideas of the kinds of structural results we need to prove, and some of the algorithmic ideas that go into the reconstruction. It will be a considerable simplification of all the actual cases we need to consider and their analyses.

$\mathcal{S}_1(f)$ is essentially low dimensional: $\mathcal{S}_1(f)$ is in correspondence with the linear factors of f. The linear factors could either divide $\gcd(T_1 + T_2 + T_3)$ (which we assume is 1 for the proof overview) or it could divide $\mathrm{sim}(T_1 + T_2 + T_3)$. We show that the set of linear factors dividing $\mathrm{sim}(T_1 + T_2 + T_3)$ can span dimension at most $O(\log d)$. This uses lower bounds for 2-query locally decodable codes (similar such bounds also appeared in [DS05, Shp07, KS09a]). See Lemma 6.4 for the precise statement.

Understanding the structure of $\mathcal{S}_2(f)$: If we could show that linear forms defining the kernels of spaces in $\mathcal{S}_2(f)$ are also low dimensional then that would be very convenient, since then if the gates in the circuit start off having many high rank linear forms, it would show that many of the spaces in $\mathcal{S}_3(f)$ remain unblocked, and then we can use them to learn linear forms.

However, this turns out to be not true and this causes the proof to become quite a bit more involved. Though we are not able to bound the dimension of $\mathcal{S}_2(f)$ as a whole, we still manage to prove some structural results that suffice for our purpose.

Consider any $\mathbb{V}(l, l') \in \mathcal{S}_2(f)$. Thus when we consider f modulo l and l', it is identically zero. There are two ways this can happen. Either each of the T_i's vanishes modulo l and l' (in other words, each T_i has a linear form dividing it that is in the span of l and l') or the T_i's don't *all* individually vanish, but still their sum vanishes.

We would like to partition the set $\mathcal{S}_2(f)$ based on the above two possibilities. We say $\mathbb{V}(l, l') \in \mathcal{S}_2(f)$ is in $\mathcal{S}_2^{sp}(f)$ if each T_i has a linear factor lying in $\mathrm{span}(l, l')$ and we say it is in $\mathcal{S}_2^{reg}(f)$ otherwise. (We are cheating a bit here - our actual definitions of these two sets is a bit more subtle, but for intuition, this is good enough. In reality the set $\mathcal{S}_2^{reg}(f)$ is a bit larger. It could be that each T_i has a linear form in $\mathrm{span}(l, l')$ but when we remove those linear forms (taking into account multiplicities) then the resulting circuit still vanishes when we go mod l and l'. In this case we add $\mathbb{V}(l, l')$ to $\mathcal{S}_2^{reg}(f)$).

The set $\mathcal{S}_2^{sp}(f)$ is actually a nice helpful set. It can be quite useful in learning linear forms that appear in the circuit. For l_1 dividing T_1, l_2 dividing T_2 and l_3 dividing T_3 suppose that $\mathbb{V}(l_1, l_2, l_3)$ did not belong to $\mathcal{S}_3(f)$ since it got blocked by a space in $\mathcal{S}_2(f)$. Then we show that if that was a space in $\mathcal{S}_2^{sp}(f)$, that is usually not a big problem, since the space in $\mathcal{S}_2^{sp}(f)$ can actually be used to learn the space $\mathbb{V}(l_1, l_2, l_3)$ unless one of l_1, l_2, or l_3 is in the kernel of the $\mathcal{S}_2^{sp}(f)$ space. This argument is not immediate to see. The details appear in Lemma 6.8. If a large fraction $\mathbb{V}(l_1, l_2, l_3)$ spaces are blocked by $\mathcal{S}_2^{sp}(f)$ spaces whose kernel contains one of l_1, l_2, or l_3, then we learn these linear forms from the intersection of kernels of $\mathcal{S}_2^{sp}(f)$ spaces, so this case turns out not be be a problem either.

The bigger issue is when $\mathbb{V}(l_1, l_2, l_3)$ gets blocked by a space in $\mathcal{S}_2^{reg}(f)$. We show that this cannot happen *too often*. Though we cannot say that the union of kernels of spaces in $\mathcal{S}_2^{reg}(f)$ is low dimensional, we can say something close. Consider the maximum number of spaces, k, in

$\mathcal{S}_2^{reg}(f)$ such that their kernels are completely linearly independent. In other words, the k kernels (that are each 2 dimensional) in total span a $2k$ dimensional space. We show that k is at most $O(\log d)$. (See Lemma 6.6 for details). This structure ends up being sufficient to show that $\mathcal{S}_2^{reg}(f)$ cannot just block all spaces of the form $\mathbb{V}(l_1, l_2, l_3)$ that we wanted to learn, assuming that each of the T_i's started off with enough linearly independent linear forms present in all of them.

Though our target is to learn $\Omega(\log d)$ independent linear forms appearing in one gate, in most cases it is sufficient to learn two independent linear forms (not in kernels of $\mathcal{S}_1$ spaces) from one gate. We can then reconstruct the circuit mod these linear forms as $\Sigma\Pi\Sigma(2)$ circuits using the reconstruction algorithm in [Sin16b] and use it to get projections of a high-rank gate, which we can then glue to obtain $\Omega(\log d)$ independent linear forms from one gate.

We now discuss a few additional algorithmic tools that go into the analysis of one specific case that cannot be learned using the above mentioned ideas. Suppose that there is one gate such that all linear forms appearing in it only span a constant dimensional space.

Learning linear forms when some of the gates have low rank Assume the linear forms appearing in T_3 span a c dimensional space for some constant c. In this case, it can be that $\mathcal{S}_3(f)$ and $\mathcal{S}_2^{sp}(f)$ are both empty, and spaces in $\mathcal{S}_1(f)$ and $\mathcal{S}_2^{reg}(f)$ block any codimension 3 space from appearing in $\mathcal{S}_3(f)$. For details on this case, see Lemma 6.13 and Lemma 6.14. We mention a few ingredients that go into the analysis.

We need a new algorithmic insight, since $\mathcal{S}_1(f)$, $\mathcal{S}_2(f)$ and $\mathcal{S}_3(f)$ might be useless. This method also shows up in the learning of a few other other instances (of the structural partitioning) of $\Sigma\Pi\Sigma(3)$ circuits.

When T_3 has only low rank linear forms, it turns out we can then (essentially) compute $\mathcal{S}_2(T_1 + T_2)$. Note that we do not have blackbox access to $T_1 + T_2$. We would still like to compute $\mathcal{S}_2(T_1 + T_2)$. The key observation is that if $\mathbb{V}(l, l') \in \mathcal{S}_2(T_1 + T_2)$ then even though we don't know that $C \mod \langle l, l' \rangle$ is zero, we can still conclude that $C \mod \langle l, l' \rangle$ has *few essential variables* (see Definition 6) , i.e. it can be written as a polynomial depending on constantly many linear forms. In order to compute $\mathcal{S}_2(T_1 + T_2)$ we will attempt to find all $\mathbb{V}(l, l')$ such that $C \mod \langle l, l' \rangle$ has few essential variables. We show that this can be done using our algorithms for finding $\mathcal{S}_2$ spaces of a polynomial, combined with a suitable modification of an algorithm by Carlini [Car06] (see Theorem 3.11) which can compute the number of essential variables in a polynomial. Once we can compute $\mathcal{S}_2(T_1 + T_2)$, we can use intersections of the kernels of these spaces to compute linear forms appearing in one of T_1 or T_2[3].

Learning linear forms when C is of special form We say a circuit C is of special form if there is a linear form l such that when we go modulo l, the simple part of the circuit has low rank. Note, this is the case when we don't know how to bound and hence compute $\mathcal{S}_3(f)$. Thus again new ideas are needed. In this case we inspect the structure of the circuit and show that it must be one of three types (see Section 6.4). In each of these types, though $\mathcal{S}_3(f)$ cannot be learned, we show how to bound and learn an interesting subset of $\mathcal{S}_3(f)$ (which we call $\mathcal{S}_3^*(f)$) that still contains enough useful codimension 3 spaces that allow us to learn some of the linear forms appearing in C. We then again have to consider cases based on whether all gates are high rank or not, and construct learning algorithms similar to the non-special form cases, but with $\mathcal{S}_3^*(f)$ playing the role of $\mathcal{S}_3(f)$.

[3]This is again a bit of a simplification of our algorithm. It could be that the rank of T_3 is super constant, or it could be that most of $\mathcal{S}_2(T_1 + T_2)$ is blocked by the $\mathcal{S}_1$ spaces. For details, see Section 6.2.2

2.4 Overview: From a few linear forms to reconstructing the entire circuit

Once we learn a few linear independent linear factors (we will actually ensure we learn $\Omega(\log d)$ linear factors) appearing in one of the gates (say T_1) then the high-level plan is to consider the circuit modulo each of the factors to obtain a projected circuit of top fan-in at most two. This can be learnt using a reconstruction algorithm for $\Sigma\Pi\Sigma(2)$ circuits from [Sin16b] (see Theorem 3.10) as a unique $\Sigma\Pi\Sigma(2)$ circuit when the distance between the projected gates is high or as a $\Sigma\Pi\Sigma(1, d, r)$ circuit when the distance is low. Moreover by a result by Shpilka [Shp07] (See Theorem 3.15), given enough linearly independent projections of $T_2 + T_3$ suffices in recovering $T_2 + T_3$. An extension of this result by Karnin and Shpilka [KS09a] (See Lemma 3.16), gives a way of recovering $T_2 + T_3$ from enough $\Sigma\Pi\Sigma(1, d, r)$ projections of low distance $T_2 + T_3$.

Comparison to [KS09a] As described in our overview, once we have a few linear forms from a gate, the process of learning the entire circuit follows closely the outline from [KS09a]. The main difference stems from how the few linear forms are obtained. The main technical contribution of this paper is to show how to efficiently compute these linear forms over large fields. In the works of [KS09a, Shp07], the authors obtained the linear forms using a brute-force search approach, by searching over all possible linear forms with $O(\log d)$ variables, which took quasi-poly($|\mathbb{F}|$) time.

3 Preliminaries

Notations. Let $\mathbb{N} := \{0, 1, 2, \ldots\}$ and $\mathbb{N}^+ := \{1, 2, \ldots\}$. Denote $\{1, 2, \ldots, n\}$ by $[n]$. The cardinality of a set S is denoted by $|S|$. $\mathbb{F}$ is usually used to denote the underlying field. $\mathbb{R}$ refers to the field of real numbers, and $\mathbb{C}$ refers to the field of complex numbers. Denote by $\log a$ the logarithm of a with base two.

Throughout the paper, we use uppercase letters X, Y to denote sets of variables, lowercase x_i to denote variables, $\mathbf{x}, \mathbf{y}$ or $\bar{x}, \bar{y}$ to denote vectors/tuples of variables, and $\mathbf{v}$ to denote a vector/tuple of field constants.

Whenever we say linear forms divide a multiplication gate, we mean up to scalar multiples. For a polynomial f, $\mathrm{Lin}(f)$ denotes the multiset of linear factors of f (including multiplicities), and $\mathrm{NonLin}(f)$ refers to $\frac{f}{\prod_{l \in \mathrm{Lin}(f)} l}$. We use $\mathrm{span}(l_1, \ldots, l_r)$ to refer to the vector space that is the span of the linear forms $\{l_1, \ldots, l_r\}$. In other words, it is the set of all vectors of the form $\sum_{i=1}^{r} \alpha_i l_i$ for $\alpha_i \in \mathbb{F}$. For a vector space $\mathbf{V}$, $\dim(\mathbf{V})$ denotes the dimension of $\mathbf{V}$.

Given k linearly independent linear forms $l_1, l_2, \cdots l_k$, let $\mathbb{V}(l_1, l_2, \ldots, l_k) \subseteq \mathbb{F}^n$ denote the codimension k subspace of $\mathbb{F}^n$ corresponding to those vectors where $l_1, l_2, \cdots l_k$ evaluate to 0. We say that $\mathbb{V}(l_1, l_2, \ldots, l_k)$ is a vanishing space for a polynomial f if f vanishes on all points of $\mathbb{V}(l_1, l_2, \ldots, l_k)$.

Consider any invertible linear transformation $\phi \in \mathbb{F}^{n \times n}$ such that $\phi(l_i) = x_i$ for all $i \in [k]$. Let $\phi \cdot f = f(\phi(\bar{x}))$. Then setting $x_1, x_2, \ldots x_k$ to 0 in $\phi \cdot f$ results in the identically 0 polynomial. The polynomial $f \mod \langle l_1, \ldots, l_k \rangle$ is equivalent (up to an invertible linear map) to $\phi \cdot f$ after setting $x_1, x_2, \ldots x_k$ to 0. Often it is easier to think in terms of $\phi \cdot f$, and once we learn $\phi \cdot f$, one can recover f after applying the inverse linear map. We abuse notation and use $f \mod l$ (and $C \mod l$ where C is a circuit computing f) to denote $f \mod \langle l \rangle$ (or $C \mod \langle l \rangle$) for a linear form l.

3.1 Depth-3 Circuits

In this section, we formally introduce the general model of depth-3 circuits which is the focus of our paper.

Definition 1. *A depth-3 $\Sigma\Pi\Sigma(k)$ circuit C computes a polynomial of the form*

$$C(X) = \sum_{i=1}^{k} T_i(X) = \sum_{i=1}^{k} \prod_{j=1}^{d_i} l_{i,j}(X),$$

where the $l_{i,j}$-s are linear functions; $l_{i,j}(X) = \sum_{t=1}^{n} a_{i,j}^t x_t + a_{i,j}^0$ with $a_{i,j}^t \in \mathbb{F}$.

We say that C is minimal *if no strict subset of the multiplication gates sums to zero. We define* $\gcd(C)$ *as the linear product of all the non-constant linear functions that belong to all the T_i-s. I.e.* $\gcd(C) = \gcd(T_1, \ldots, T_k)$. *We say that C is* simple *if* $\gcd(C) = 1$. *The simplification of C, denoted by* $\mathrm{sim}(C)$, *is defined as* $C/\gcd(C)$. *In other words, the circuit resulting upon the removal of all the linear functions that appears in* $\gcd(C)$.

Definition 2 (Homogeneous Depth 3 circuit)**.** *A depth 3 circuit $\Sigma\Pi\Sigma(k)$ computing a polynomial $f \in \mathbb{F}[x_1, \ldots, x_n]$ is a homogeneous depth 3 circuit $\Sigma\Pi\Sigma(k)$ if f is homogeneous and the polynomial computed in every gate of the circuit is homogeneous as well. It will have the following form*

$$C(X) = \sum_{i=1}^{k} T_i(X) = \sum_{i=1}^{k} \prod_{j=1}^{d} l_{i,j}(X),$$

where the $l_{i,j}$-s are linear functions; $l_{i,j}(X) = \sum_{t=1}^{n} a_{i,j}^t x_t + a_{i,j}^0$ with $a_{i,j}^t \in \mathbb{F}$ and $a_{i,j}^0 = 0$.

Definition 3 (Rank of circuit)**.** *The* rank *of a circuit $C(X) = \sum_{i=1}^{k} T_i(X) = \sum_{i=1}^{k} \prod_{j=1}^{d_i} l_{i,j}(X)$ is defined as the dimension of the space spanned by all the linear forms in the circuit* $\dim(\mathrm{span}(\{l_{i,j} : i \in [k], j \in [d_i]\}))$. *We denote it by* $\mathrm{rank}(C)$.

Definition 4 (Rank of Simple part of circuit)**.** *The* rank *of the simple part of the circuit $C(X) = \sum_{i=1}^{k} T_i(X) = \sum_{i=1}^{k} \prod_{j=1}^{d_i} l_{i,j}(X)$ is defined as the rank of the simple part (obtained after removing the gcd of T_i's). We will denote the simple rank of C using $\Delta(C) = \mathrm{rank}(\mathrm{sim}(C))$. This also defines a distance measure between 2 circuits C_1, C_2 as $\Delta(C_1, C_2) = rank(sim(C_1 + C_2))$.*

In the following lemma, we will reduce the problem of reconstruction for any polynomial f computed by a $\Sigma\Pi\Sigma(3)$ circuit to the reconstruction of a homogeneous polynomial f^{hom} computed by a homogeneous $\Sigma\Pi\Sigma(3)$ circuit. Therefore, from now on, we are only concerned with the reconstruction of $\Sigma\Pi\Sigma(3)$ circuits in this paper and all $\Sigma\Pi\Sigma(3)$ circuits we consider will be assume to be homogeneous.

Lemma 3.1 (Section 1.5, [Sin16a])**.** *Let $f \in \mathbb{F}[x_1, \ldots, x_n]$ be degree d polynomial computed by a $\Sigma\Pi\Sigma(k)$ circuit C. Then in time* $\mathrm{poly}(n, d)$ *(per query) one can simulate a blackbox access to a homogeneous $\Sigma\Pi\Sigma(3)$ circuit computing a homogeneous $f^{hom} \in \mathbb{F}[x_1, \ldots, x_n, z]$ such that any reconstruction algorithm for f^{hom} immediately implies a reconstruction algorithm for f, with only a* $\mathrm{poly}(n, d)$ *overhead in time complexity.*

Proof. Let f^d denote the degree d homogeneous part of f. Given f, define f^{hom} to be

$$f^{hom}(x_1, \ldots, x_n, z) = \begin{cases} z^d f\left(\frac{x_1}{z}, \ldots, \frac{x_n}{z}\right) & z \neq 0 \\ f^d(x_1, \ldots, x_n) & z = 0 \end{cases}$$

If f is computed by a $\Sigma\Pi\Sigma(k)$ circuit C of the form

$$C = \sum_{i=1}^{k} T_i(X) = \sum_{i=1}^{k} \prod_{j=1}^{d_i} (l_{i,j}(X) + a_{i,j}^0)$$

circuit, then one can easily construct the following $\Sigma\Pi\Sigma(k)$ circuit C^{hom} computing f^{hom}

$$C^{hom} = \sum_{i=1}^{k} T_i'(X) = \sum_{i=1}^{k} z^{d-d_i} \cdot \prod_{j=1}^{d_i} (l_{i,j}(X) + a_{i,j}^0 \cdot z).$$

Given black-box access to f, one can easily simulate black-box access to f^{hom}. To query $f^{hom}(x_1, \ldots, x_n, z)$, query $f(\frac{x_1}{z}, \ldots, \frac{x_n}{z})$ and multiply the result with z^d if $z \neq 0$. From Lemma 2.1 of [DS05], we can get black-box access to $f^d(x_1, \ldots, x_n)$ using black-box access to f in poly(n) time. Therefore, we can query $f^d(x_1, \ldots, x_n)$ when $z = 0$.

Finally, if we can reconstruct a circuit computing f^{hom}, we can get a circuit computing f by simply substituting $z = 1$. □

3.2 Polynomial Identity Testing and Rank Bounds

Lemma 3.2 (Schwartz-Zippel Lemma,[Sch80, Zip79]). *Let $f \in \mathbb{F}[x_1, \ldots, x_n]$ be a polynomial of total degree d such that it is not identically zero. Let $S \subset \mathbb{F}$ be any finite set. For $s_1, \ldots, s_n$ picked independently and uniformly from S,*

$$\Pr[f(s_1, \ldots, s_n) = 0] \leq \frac{d}{|S|}.$$

A finite set of points S with the property that every line through two points of S passes through a third point in S is called a Sylvester-Gallai configuration. The famous Sylvester-Gallai theorem states that the only Sylvester-Gallai configurations in $\mathbb{R}^n$ are those formed by collinear points. This basic theorem about point-line incidences was extended to higher dimensional flats in [Han65, BE67] over the Real numbers and in [BDWY13, DSW14] over $\mathbb{C}$. We define the *rank* of a set of vectors to be the dimension of the linear space they span.

Definition 5 ($\mathrm{SG}_k(\mathbb{F}, m)$). *Let S be a set of non-zero vectors in $\mathbb{F}^{n+1}$ without multiples: ie no two vectors in S are scalar multiples of each other. Suppose that for every set $V \subseteq S$ of k linearly independent vectors, the linear span of V contains at least $k+1$ vectors of S. Then, the set S is said to be SG_k-closed. The largest possible rank of an SG_k-closed set of at most m vectors in $\mathbb{F}^n$ (for any n) is denoted by $\mathrm{SG}_k(\mathbb{F}, m)$.*

Over the field of Real numbers, it is known that $\mathrm{SG}_k(\mathbb{R}, m) = 2(k-1)$ [Han65, BE67]. The rank of high dimensional Sylvester-Gallai configurations over $\mathbb{C}$ was bounded by 2^{c^k} for a fixed constant c in [BDWY13]. This bound was further improved to $\mathrm{SG}_k(\mathbb{C}, m) = c^k$(for a fixed constant c) in [DSW14].

The polynomial time blackbox PIT algorithms for $\Sigma\Pi\Sigma(k)$ circuits over $\mathbb{R}$ and $\mathbb{C}$ follow from some strong structural properties of identically zero $\Sigma\Pi\Sigma(k)$ circuits. In [KS09b] it was shown that the rank of any identically zero, simple and minimal $\Sigma\Pi\Sigma(k)$ circuit is at most some constant depending on k. This bound was improved in [SS11, SS13], and the theorem below gives the best bound we know.

Theorem 3.3 ([SS13]). *Let C be a $\Sigma\Pi\Sigma(k)$ circuit, over field $\mathbb{F}$, that is simple, minimal and zero. Then, we have* $\mathrm{rank}(C) \leq 2k^2 + k \cdot \mathrm{SG}_k(\mathbb{F}, d)$.

Combining the above theorem with the best bounds we know for $\mathrm{SG}_k(\mathbb{R}, m)$ and $\mathrm{SG}_k(\mathbb{C}, m)$ we obtain the following,

Theorem 3.4. *Let C be a simple, minimal and identically zero $\Sigma\Pi\Sigma(k)$ circuit over $\mathbb{R}$ or $\mathbb{C}$. Then there is an absolute constant $\mathcal{R}(k)$ depending only on k such that $\mathrm{rank}(C) < \mathcal{R}(k)$. If C is over $\mathbb{R}$ then we can bound $\mathrm{rank}(C)$ by $3k^2$. If C is over $\mathbb{C}$ then we can bound $\mathrm{rank}(C)$ by $2k^2 + k \cdot c^k$ for some absolute constant c.*

3.3 Other Known Results

Theorem 3.5 (Theorem 1.1[KSS14])**.** *[Effective Hilbert irreducibility] Let $S \subseteq \mathbb{F}$ be a finite set and $g(X, A_1, \ldots, A_n)$ a monic polynomial in X of total degree at most d. If g is irreducible then it holds that*

$$\mathbb{P}_{\boldsymbol{\alpha},\boldsymbol{\beta}}[g(X, \alpha_1 T + \beta_1, \ldots, \alpha_n T + \beta_n) \text{ is not irreducible}] < O(d^5/|S|),$$

where $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ are chosen uniformly and independently from S^n.

Lemma 3.6 (Black-box multivariate polynomial interpolation,[BOT88])**.** *Let n, m, d be parameters and $\mathbb{F}$ be a field that is $\mathbb{R}$ or $\mathbb{C}$ (or large enough). There exists a deterministic algorithm that runs in time $(nmd)^{O(1)}$, and outputs a set S of points in $\mathbb{F}^n$, such that given black-box access to any degree d polynomial $f \in \mathbb{F}[x_1, \ldots, x_n]$ with at most m monomials, the coefficients of all monomials can be recovered in $(nmd)^{O(1)}$ time using evaluations from the set $\{f(s) : s \in S\}$.*

Lemma 3.7 (Blackbox Factoring, [KT90])**.** *There exists a randomized algorithm that takes as input black-box access to a degree d, n-variate polynomial f with coefficients in some field $\mathbb{F}$, runs in time $\mathrm{poly}(nd)$ and outputs black-box access to polynomials $f_1, \ldots, f_m$ ($m \leq d$) along with integers $e_1, \ldots, e_m$ such that,*

$$Pr[f = f_1^{e_1} \ldots f_m^{e_m} \bigwedge f_1, \ldots, f_m \text{ are irreducible}] \geq 1 - o(1).$$

Using the above, we can also decompose any circuit into its linear factors(which we can interpolate) and $\mathrm{NonLin}(f)$ in randomized $\mathrm{poly}(n, d)$ time.

3.4 Solving a System of Polynomial Equations

We obtain the vanishing spaces of our circuit by solving a system of polynomial equations. We will need to find all possible solutions of the system that we set up, and in order to do this, we show that the number of solutions is finite, and in particular polynomially bounded.

A longer discussion on the complexity of finding a single solution to a system of polynomial equations for various fields can be found in [BSV21].

In this work, the polynomial systems we solve have a small ($O(1)$) number of variables, and hence once can find solutions efficiently. The theorem we state below is a variant of an analogous one that appears in [BSV21], and it describes the current known upper bounds for solving a system of polynomial equations for various fields.

Let $\bar{F}$ denote the algebraic closure of $\mathbb{F}$.

Theorem 3.8. *Let $f_1, f_2, \ldots f_m \in \mathbb{F}[x_1, \ldots, x_n]$ be n-variate polynomials of degree at most d. Suppose that the system of equations $f_1(x) = 0, \ldots, f_m(x) = 0$ has finitely many solutions in the algebraic closure of $\mathbb{F}$. Then, the complexity of finding all the solutions in $\bar{F}$ is as follows:*

1. *[GVJ88] For $\mathbb{F} = \mathbb{R}$, there is a deterministic $\text{poly}((md)^{n^2})$ time algorithm. Here the authors assumed that the constants appearing in the system are integers (or rationals). Note that for all computational applications we can WLOG assume this by simply approximating/truncating a given real number at some number of bits.*

2. *[Ier89, Buc76] For $\mathbb{F} = \mathbb{C}$ (or any algebraically closed field), there is a deterministic $(mn)^{O(n)} \cdot d^{O(n^n)}$ time algorithm.*

Thus in deterministic time $(mdn)^{O(n^n)}$, we can find all the solutions of $f_1(x) = 0, \ldots, f_m(x) = 0$ if it has finitely many solutions in the algebraic closure of $\mathbb{F}$.

Remark 3.9. *In the results used above, we have suppressed $\text{poly}(s)$ multiplicative dependence (when n is a constant) in the running time where s is the maximum bit complexity of any coefficient appearing in the input circuit. We use the above algorithm only in cases where n is constant, and hence there is an additional $\text{poly}(s)$ running time factor, which we suppress throughout the paper.*

3.5 Reconstruction for Top Fan-in 2

In [Sin16b], for fields $\mathbb{F}$ of characteristic 0 ($\mathbb{Q}, \mathbb{R}$ or $\mathbb{C}$ for simplicity), a reconstruction algorithm for $\Sigma\Pi\Sigma(2)$ circuits was presented as below:

Theorem 3.10 (Theorem 1.1, [Sin16b])**.** *Let $f \in \mathbb{F}[x_1, \ldots, x_n]$ be any degree d, n-variate polynomial (to which we have blackbox access) which can be computed by a depth 3 circuit of the form $C = G \times (T_0 + T_1)$ with top fan-in 2 (i.e. a $\Sigma\Pi\Sigma(2)$ circuit). Let $\mathcal{R}(k)$ be as defined in Theorem 3.4. Assume $\gcd(T_0, T_1) = 1$ and $\dim(\text{span}\{l : l|T_0T_1\})$ is bigger than $\mathcal{R}(4) + 1$. Then there exists a randomized algorithm that runs in time $\text{poly}(n, d)$ and computes a $\Sigma\Pi\Sigma(2)$ circuit computing f with high probability.*

3.6 Essential Variables of a Polynomial

This notion will be useful in reconstruction when the input circuit is low rank, as well as when one of more gates is low rank. We start by defining essential variables in a polynomial.

Definition 6. *[[Kay11]](Essential Variables) The number of essential variables in $f(x_1, \ldots, x_n)$ is the smallest t such that there exists an invertible linear transformation $A \in \mathbb{F}^{(n\times n)}$ on the variables such that $f(A \cdot \overline{x})$ depends on only t variables.*

The number of redundant variables is the number of essential variables subtracted from n. We will use the following result from [Car06] that allows us to compute t, the number of essential variables, and the linear transformation A.

Theorem 3.11 ([Car06],[Kay11])**.** *Let n, d be positive integers and $\mathbb{F}$ be a field with $\text{char}(\mathbb{F}) > d$ or 0. There is a randomized algorithm that takes as input black-box access to an n-variate degree d polynomial $f(x) \in \mathbb{F}[\bar{x}]$ having m essential variables, that runs in time $(nd)^{O(1)}$ and outputs an invertible matrix $A \in \mathbb{F}^{(n\times n)}$ such that $f(A \cdot \bar{x})$ depends only on the first m-variables.*

The partial derivative $\partial_i f$ is used to represent $\frac{\partial f}{\partial x_i}$. We use ∂f to denote $(\partial_1 f, \ldots, \partial_n f)$. We define the partial derivative matrix of a polynomial f, $M(f)$, as the matrix with columns indexed by monomials over n variables and degree $d-1$, while the rows are indexed by $[n]$, and $M_{i,j} = \text{coeff}_j(\partial_i f)$ where $\text{coeff}_j(g)$ is the coefficient of monomial j (represented as vector) in g.

We denote $\partial f^{\perp}$ as the set of vectors $\mathbf{a} \in \mathbb{F}^n$ such that $\mathbf{a} \cdot \partial f = 0$. The proof of the above theorem relies on the following lemma which describes the relation between the partial derivative matrix and the number of essential variables.

Lemma 3.12 ([Car06], Lemma B.1[Kay11]). *The number of redundant variables in a polynomial $f(x_1, \ldots, x_n)$ equals the dimension of $\partial f^{\perp}$. In particular, the number of essential variables of f is the rank of the partial derivative matrix $M(f)$.*

The following lemma from [Shp07] will also be useful to us.

Lemma 3.13 (Lemma 23,[Shp07]). *Let $f(x_1, \ldots, x_n) \in \mathbb{F}[x_1, \ldots, x_n]$ be a polynomial with k essential variables, such that it has two different representations: $f = g(l_1, \ldots, l_k) = h(l'_1, \ldots, l'_k)$ for polynomial $g, h \in \mathbb{F}[y_1, \ldots, y_k]$ and $l_1, \ldots, l_k, l'_1, \ldots, l'_k$ are linear forms in $\mathbb{F}[x_1, \ldots, x_n]$. Then* $\text{span}(\{l_i\}_{i\in[k]}) = \text{span}(\{l'_i\}_{i\in[k]})$.

3.7 Generalized Depth-3 circuits

Definition 7. *A generalized depth 3 circuit $\Sigma\Pi\Sigma(k, d, r)$ is of the form*

$$C = \sum_{i=1}^{k} \left(\prod_{j=1}^{d_i} l_{ij} \cdot h_i(\bar{l}_{i1}, \ldots, \bar{l}_{ir}) \right)$$

where l_{ij},$\bar{l}_{ik}$ are linear forms in $\mathbb{F}[x_1, \ldots, x_n]$ and $d = max_i(d_i + deg(h_i))$.

In particular in the setting when r is small (say constant or $O(\log d)$), the representation looks like a $\Sigma\Pi\Sigma(k)$ circuit where every product gate is further multiplied by a polynomial in few (r) linear forms.

3.8 Locally Decodable Codes

Locally decodable codes are error-correcting codes that allow the recovery of each symbol of the message, from a corrupted code word, by looking at only a constant or small number of entries of the corrupted code word.

Definition 8. *A (q, δ, ϵ)-locally decodable code encodes $x \in \mathbb{F}^n$ to $E(x) \in \mathbb{F}^m$ such that for each index $i \in [n]$, x_i can be recovered from $E(x)$ with probability $> \epsilon$ by reading only q (random) entries, even if $E(x)$ was corrupted in δm positions.*

The following lower bound for 2-query LDCs over arbitrary fields is from [DS05]. (Exponential lower bounds for 2-query LDCs over small finite fields were proved in [GKST06]).

Theorem 3.14 ([DS05], Theorem 1.2). *Let $\delta, \epsilon \in [0, 1]$, $\mathbb{F}$ be an arbitrary field, and let $E : \mathbb{F}^n \to \mathbb{F}^m$ be a linear $(2, \delta, \epsilon)$-LDC. Then*

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

The LDC lower bounds have been influential in obtaining reconstruction results like [Shp07, KS09a]. One application (that was useful for reconstruction) is the following lemma, which bounds the dimension of the set of "rank-reducing" linear forms. We use these in Section 6, specifically Lemma 6.3.

Lower bounds for 2-query LDCs also formed the basis for the method of gluing projections of a product of linear forms to reconstruct a multiplication gate in [Shp07] and [KS09a] (which we will discuss next).

3.9 Gluing Projections

Using locally Decodable lower bounds, in [Shp07], the authors gave an algorithm that could learn a product of linear forms exactly with multiplicities if given access to $\Omega(\log d)$ independent non-zero projections of the product. This is summarized in the theorem below.

Theorem 3.15 (Implicit in [Shp07]). *Let L be a multiset containing d linear functions in n variables. Let $\{\varphi_1, \ldots, \varphi_m\}$ be a set of linearly independent linear functions such that $m \geq 100 \log(d)$. For each $j \in [m]$ define the multiset*

$$L_j \triangleq \{l \mod \varphi_j : l \in L\}.$$

Then there exists a deterministic algorithm that given $\{L_j\}_{j=1}^m$ outputs L in $\mathrm{poly}(n, d)$ time.

In [KS09a], the authors gave a way to similarly glue projections of gates in a generalized circuit. The two key ingredients were the theorem above, and the following lemma below that allows one to glue the projections of low-rank polynomials.

Lemma 3.16 (Special case of Lemma 4.20 in [KS09a]). *Let h be a non-zero n-variate polynomial of degree d. Let $r \in \mathbb{N}^+$ be such that h has r essential variables. Let l_1, l_2 be two independent linear forms in $\mathbb{F}[x_1, \ldots, x_n]$ such that $h \mod \langle l_1, l_2 \rangle$ has r essential variables. Then there exists a deterministic algorithm which when given as the input the two polynomials $\{h \mod l_1, h \mod l_2\}$, outputs a representation of h as a polynomial of r linear functions in $O(n \cdot d^r)$ time.*

4 Upper bounding the size of $\mathcal{S}_2(f)$ and $\mathcal{S}_3(f)$

In this section, we will show that if a degree d polynomial $f \in \mathbb{F}[\bar{x}]$ is computed by a $\Sigma\Pi\Sigma(3)$ circuit C then other then in certain degenerate cases, it will "vanish" on only finitely many codimension 3 subspaces. In the next sections we will show how to compute these subspaces and then how to extract the linear forms of C from these subspaces.

Given k linearly independent linear forms $l_1, l_2, \cdots l_k$, let $\mathbb{V}(l_1, l_2, \ldots, l_k) \subseteq \mathbb{F}^n$ denote the codimension k subspace of $\mathbb{F}^n$ corresponding to those vectors where $l_1, l_2, \cdots l_k$ evaluate to 0. We say that $\mathbb{V}(l_1, l_2, \ldots, l_k)$ is a vanishing space for a polynomial f if f vanishes on all points of $\mathbb{V}(l_1, l_2, \ldots, l_k)$.

Equivalently, consider any invertible linear transformation $\phi \in \mathbb{F}^{n \times n}$ such that $\phi(l_i) = x_i$ for all $i \in [k]$. Let $\phi \cdot f = f(\phi(\bar{x}))$. Then setting $x_1, x_2, \ldots x_k$ to 0 in $\phi \cdot f$ results in the identically 0 polynomial.

For a polynomial f defined over $\mathbb{F}^n$, we will define $\mathcal{S}_1(f)$ to be the set of codimension 1 subspaces over which f vanishes. $\mathcal{S}_1(f) = \{\mathbb{V}(l) \mid \mathbb{V}(l) \text{ is a vanishing space for} f\}$

We would like to define $\mathcal{S}_2(f)$ to be the set of codimension 2 subspaces over which f vanishes and try to show that this is finite. However note that if f has even one codimension 1 subspace on which it vanishes, then there will be infinitely many codimension 2 subspaces on which it vanishes, since for any $W \in \mathcal{S}_1(f)$, f vanishes on every single codimension 2 subspace of W. Thus when we define $S_2(f)$, we will not consider such subspaces.

Let $\mathcal{S}_2(f) = \{W \mid W$ is a codimension 2 subspace of $\mathbb{F}^n, f$ vanishes over W and $W \not\subset W'$ for any $W' \in \mathcal{S}_1(f)\}$.

Note that any $W \in \mathcal{S}_2(f)$ is of the form $\mathbb{V}(l_1, l_2)$ for some two linear forms l_1, l_2. Moreover any two independent linear forms in the span of l_1 and l_2 will result in the same space W.

Similarly we define $\mathcal{S}_3(f)$ to be the set of codimension 3 subspaces W over which f vanishes such that W is not contained in any subspace from $\mathcal{S}_2(f)$ or $\mathcal{S}_1(f)$.

$\mathcal{S}_3(f) = \{W \mid W$ is a codimension 3 subspace of $\mathbb{F}^n, f$ vanishes over W and $W \not\subset W'$ for any $W' \in \mathcal{S}_1(f) \cup \mathcal{S}_2(f)\}$.

Note again that any $W \in \mathcal{S}_3(f)$ is of the form $\mathbb{V}(l_1, l_2, l_3)$ for some three linearly independent linear forms l_1, l_2, l_3. Moreover any three independent linear forms in the span of l_1, l_2 and l_3 will result in the same space W.

Lemma 4.1. *Let f be a degree d polynomial. Then, $|\mathcal{S}_1(f)| \leq d$.*

Proof. The proof is quite simple. Any linear form l such that f vanishes on $\mathbb{V}(l)$ must be a linear factor of f. Since f has degree d, it can have at most d distinct factors. □

We will also show how to bound the size of $\mathcal{S}_2(f)$ and $\mathcal{S}_3(f)$ under additional structural assumptions of f. Note that if f is computed by an $\Sigma\Pi\Sigma(3, d)$ circuit of the form $T_1 + T_2 + T_3$, then by picking one linear form from each of the three multiplication gates, we can obtain codimension 3 spaces on which f vanishes. Learning these spaces will eventually allow us to learn the linear forms, and it will be an important ingredient of our final algorithm. Note however that f might have other codimension 3 vanishing subspaces that are not of this form. The main goal of this section is to show that nevertheless under some structural assumptions, we are able to bound the number of these spaces.

Before we state and prove our upper bounds for $\mathcal{S}_2$ and $\mathcal{S}_3$ spaces we state and prove a slightly modified version of a result from [Sin22] that we will be useful for us.

Claim 4.2. *Let P_1, P_2, P_3 be distinct 2-dim subspaces of some vector space V such that $\dim(\text{span}(P_1, P_2, P_3)) \geq 4$ and P_3 is not contained in $\text{span}(P_1, P_2)$. Suppose that P is a 2-dim subspace such that for each $i \in [3]$, $\dim(P_i \cap P) = 1$. Then either*

- *$\dim(P_1 \cap P_2) = 1$ and $P_1 \cap P_2 \subset P$ or*
- *$\dim(P_3 \cap \text{span}(P_1, P_2)) = 1$ and $P_3 \cap \text{span}(P_1, P_2) \subset P$.*

Proof. Case (1): $P_1 \cap P = P_2 \cap P$. Then since P_1 and P_2 are distinct, clearly $\dim(P_1 \cap P_2) = 1$ and $P_1 \cap P_2 \subset P$.

Case(2): $P_1 \cap P \neq P_2 \cap P$.

Here, we have $\dim(\text{span}(P \cap P_1, P \cap P_2)) = \dim(P \cap \text{span}(P_1, P_2))) = 2$, but as $\dim(P) = 2$, we have $P \subset \text{span}(P_1, P_2)$. As the P_1, P_2 and P_3 together span a space of dimension ≥ 4 and $P_3 \not\subset \text{span}(P_1, P_2)$, thus $\dim(P_3 \cap \text{span}(P_1, P_2)) \leq 1$. Now as $P \subset \text{span}(P_1, P_2)$, thus $(P_3 \cap P) \subset (P_3 \cap \text{span}(P_1, P_2))$. But we also know $\dim(P_3 \cap P) = 1$, and therefore $\dim(P_3 \cap \text{span}(P_1, P_2)) = 1$ and $P_3 \cap P = P_3 \cap \text{span}(P_1, P_2)$, which means $P_3 \cap \text{span}(P_1, P_2) \subset P$. □

4.1 Bounding the number of vanishing codimension 2 subspaces

Lemma 4.3. *Let $\mathbb{F}$ be a field that is $\mathbb{R}$ or $\mathbb{C}$. Let f be a degree d, n variate polynomial in $\mathbb{F}[x_1, \ldots, x_n]$. Then there is an absolute constant c_2 such that if f is computed by a $\Sigma\Pi\Sigma(3)$ circuit $C = T_1 + T_2 + T_3$ with $\text{rank}(\text{sim}(C)) \geq c_2$, then*

$$\mathcal{S}_2(f) \leq O(d^7).$$

Proof. We will be dividing our analysis into the following cases.

1. **Case 1:** We bound the number of $W = \mathbb{V}(l_1, l_2) \in \mathcal{S}_2(f)$ such that for some $i \in [3]$, T_i vanishes on $\mathbb{V}(l_1, l_2)$.

Wlog $T_i = T_1$. Note that if T_1 vanishes on $\mathbb{V}(l_1, l_2)$, then there must be some $l \in \text{span}(l_1, l_2)$ such that l divides T_1. Note that since $\mathbb{V}(l_1, l_2) \notin \mathbb{V}(l)$ for any $\mathbb{V}(l) \in \mathcal{S}_1(f)$, thus $C' = (T_2 + T_3 \mod l)$ is nonzero, and moreover there is some linear form l' with $\text{span}(l, l') = \text{span}(l_1, l_2)$ such that $C' \mod l' \equiv 0$. There are at most d choices for l' from the factors of $(T_2 + T_3 \mod l)$, and there were at most d choices for l (once we fix i). Therefore, there are at most $O(d^2)$ possibilities for $W \in \mathcal{S}_2(f)$.

2. **Case 2:** We bound the number of $W = \mathbb{V}(l_1, l_2) \in \mathcal{S}_2(f)$ such that no T_i vanishes on $\mathbb{V}(l_1, l_2)$.

   Consider $C \mod \langle l_1, l_2 \rangle$ which is of the form $G \times (T_1' + T_2' + T_3')$ where $T_1' + T_2' + T_3'$ is a simple, minimal $\Sigma\Pi\Sigma(3)$ circuit computing the identically zero polynomial and G is a product of linear forms. By the rank bound given in Theorem 3.4, $\text{rank}(T_1' + T_2' + T_3') < \mathcal{R}(3)$ where $\mathcal{R}(3)$ is some absolute constant. Over the real numbers $\mathcal{R}(3) = 5$.

   Let c_2 be any constant greater than $\mathcal{R}(3) + 10$. Therefore, $\text{rank}(\text{sim}(C)) \geq \mathcal{R}(3) + 10$. When we consider $C \mod \langle l_1, l_2 \rangle$, the linear forms appearing in the gates of $\text{sim}(C)$ get mapped to linear forms in G or in $(T_1' + T_2' + T_3')$. The rank of those linear forms that get mapped to $(T_1' + T_2' + T_3')$ can be at most $\mathcal{R}(3) + 2$. Thus the rank of the set of linear forms that gets mapped to G is at least 8.

   Consider 3 linear forms (not all the same) l_{1j}, l_{2j}, l_{3j} from distinct product gates of $\text{sim}(C)$ such that when we consider them $\mod \langle l_1, l_2 \rangle$, they map to the same linear form l (we don't distinguish between a linear form and its multiple) and hence all get mapped to the linear forms of G. Call such triple of linear forms a "collapsing" triple when we go $\mod \langle l_1, l_2 \rangle$. These linear forms must be of the form

$$l_{1j} = l + \alpha_1 l_1 + \alpha_2 l_2$$
$$l_{2j} = l + \beta_1 l_1 + \beta_2 l_2$$
$$l_{3j} = l + \gamma_1 l_1 + \gamma_2 l_2$$

   where the α, β, γ denote field constants.

   A collapsing triple can either span a 2 or 3 dimensional space. Now, since the rank of linear forms mapping to G is at least 8, one of the following two scenarios must occur.

   - **Case 2(a):** There are two collapsing triples (l_{1j}, l_{2j}, l_{3j}) and (l_{1k}, l_{2k}, l_{3k}) such that each spans a 3-dim space and jointly the two triples span at least a 4-dim space.

     Let $V_j = \text{span}(l_{1j}, l_{2j}, l_{3j})$ and $V_k = \text{span}(l_{1k}, l_{2k}, l_{3k})$. Let $U = span(l_1, l_2)$. Since V_j and V_k both become 1-dimensional when $U = \text{span}(l_1, l_2)$ is projected to 0, it must hold that $U \subset V_j$ and $U \subset V_k$. Moreover as V_j and V_k are distinct 3-dim spaces, thus it must hold that $V_j \cap V_i = U$. Thus V_j and V_k determine U. In particular, the two triples of linear forms determine U.

     Since, there can be d possibilities for each of $l_{1j}, l_{2j}, l_{3j}, l_{1k}, l_{2k}, l_{3k}$, we have $O(d^6)$ possibilities for U, and hence for $\mathbb{V}(l_1, l_2) \in \mathcal{S}_2(f)$.

   - **Case 2(b):** The collapsing triples that span 2-dim spaces have combined rank at least 5.

     Clearly there must be at least 3 collapsing triples. Let V_i, V_j and V_k be the span of of 3 of the triples such that $\text{span}(V_i \cup V_j \cup V_k) \geq 5$. Since V_i, V_j and V_k all become 1-dimensional when $U = \text{span}(l_1, l_2)$ is projected to 0, it follows that each of V_i, V_j, V_k intersects U

in a 1-dim space. By Claim 4.2, it follows that knowing V_i, V_j and V_k is enough to determine a vector $l \in U$. Once l is determined, then the rest of the argument is similar to case 1. $C' = (T_2 + T_3 \mod l)$ is nonzero, and moreover there is some linear form l' with $\text{span}(l, l') = \text{span}(l_1, l_2)$ such that $C' \mod l' \equiv 0$. There are at most d choices for l' from the factors of $(T_2 + T_3 \mod l)$. Since there were at most $O(d^6)$ possibilities for V_i, V_j and V_k, and hence at most $O(d^6)$ possibilities for the choice of l, thus overall there are at most $O(d^7)$ possibilities for U and hence for for $W \in \mathcal{S}_2(f)$.

□

4.2 Bounding the number of vanishing codimension 3 subspaces

Lemma 4.4. *Let $\mathbb{F}$ be a field that is $\mathbb{R}$ or $\mathbb{C}$. Let f be a degree d, n variate polynomial in $\mathbb{F}[x_1, \ldots, x_n]$. Then there is an absolute constant c_3 such that if f is computed by a $\Sigma\Pi\Sigma(3)$ circuit $C = T_1 + T_2 + T_3$ with the following properties*

1. $\text{rank}(\text{sim}(C)) \geq c_3$,
2. *There is no linear form l such that $(C \mod l)$ is nonzero and* $\text{rank}(\text{sim}(C \mod l)) < c_2$.

then

$$\mathcal{S}_3(f) \leq O(d^{15}).$$

Proof. We will be dividing our analysis into the following cases.

1. **Case 1:** We bound the number of $W = \mathbb{V}(l_1, l_2, l_3) \in \mathcal{S}_3(f)$ such that for some $i \in [3]$, T_i vanishes on $\mathbb{V}(l_1, l_2, l_3)$.

   Note that if T_i vanishes on $\mathbb{V}(l_1, l_2, l_3)$, then there must be some $l \in \text{span}(l_1, l_2, l_3)$ such that l divides T_i. Note that since $\mathbb{V}(l_1, l_2, l_3) \notin \mathbb{V}(l)$ for any $\mathbb{V}(l) \in \mathcal{S}_1(f)$ thus $C' = (T_2 + T_3 \mod l)$ is nonzero. Moreover by assumption, $\text{rank}(\text{sim}(C \mod l)) = \text{rank}(\text{sim}(T_2 + T_3 \mod l)) \geq c_2$.

   Observe that $(T_2 + T_3 \mod l)$ must vanish when we consider it $\mod \langle l_a, l_b \rangle$ for any l_a, l_b such that $\text{span}(l, l_a, l_b) = \text{span}(l_1, l_2, l_3)$. Thus once l is fixed, any such $W' = \text{span}(l_a, l_b)$ is a vanishing codimension 2 space for $(T_2 + T_3 \mod l)$. By Lemma 4.3, there are at most $O(d^7)$ choices for W'. Given that there are at most $O(d)$ choices for l, thus there are totally $O(d^8)$ possibilities for $W \in \mathcal{S}_3(f)$ in this case.

2. **Case 2:** We bound the number of $W = \mathbb{V}(l_1, l_2, l_3) \in \mathcal{S}_3(f)$ such that no T_i vanishes on $\mathbb{V}(l_1, l_2, l_3)$.

   Consider $C \mod \langle l_1, l_2, l_3 \rangle$ which is of the form $G \times (T_1' + T_2' + T_3')$ where $T_1' + T_2' + T_3'$ is a simple, minimal $\Sigma\Pi\Sigma(3)$ circuit computing the identically zero polynomial and G is a product of linear forms. By the rank bound given in Theorem 3.4, $\text{rank}(T_1' + T_2' + T_3') < \mathcal{R}(3)$ where $\mathcal{R}(3)$ is some absolute constant. Over the real numbers $\mathcal{R}(3) = 5$.

   Let c_3 be any constant greater than $\mathcal{R}(3) + 17$. Therefore, $\text{rank}(\text{sim}(C)) \geq \mathcal{R}(3) + 17$. When we consider $C \mod \langle l_1, l_2, l_3 \rangle$, the linear forms appearing in the gates of $\text{sim}(C)$ get mapped to linear forms in G or in $(T_1' + T_2' + T_3')$. The rank of those linear forms that get mapped to $(T_1' + T_2' + T_3')$ can be at most $\mathcal{R}(3) + 3$. Thus the rank of the set of linear forms that gets mapped to G is at least 14.

   Note that by assumption, there is no linear form l such that $C \mod l$ is nonzero and $\text{rank}(\text{sim}(C \mod l)) \leq c_2$. However there could exists two independent linear forms $l, l' \in$

$\text{span}(l_1, l_2, l_3)$, such that $\text{rank}(\text{sim}(C \mod \langle l, l' \rangle)) < \mathcal{R}(3)$. We consider two further sub-cases based on whether this happens or not.

A **Case 2(a):** We bound the number of $W = \mathbb{V}(l_1, l_2, l_3) \in \mathcal{S}_3(f)$ such that no T_i vanishes on $\mathbb{V}(l_1, l_2, l_3)$ but there exists two independent linear forms $l, l' \in \text{span}(l_1, l_2, l_3)$, such that $\text{rank}(\text{sim}(C \mod \langle l, l' \rangle)) < \mathcal{R}(3)$.

The analysis for this case is almost identical to that of Case 2 in Lemma 4.3. By the analysis of Case 2 in Lemma 4.3, the number of spaces $W' = \mathbb{V}(l, l')$ such that $\text{rank}(\text{sim}(C \mod \langle l, l' \rangle)) < \mathcal{R}(3)$ is at most $O(d^7)$. For each fixing of $W' = \mathbb{V}(l, l')$ we now count the number of $W = \mathbb{V}(l_1, l_2, l_3) \in \mathcal{S}_3(f)$ such that $l, l' \in \text{span}(l_1, l_2, l_3)$. Thus any such W is of the form $\mathbb{V}(l, l', l'')$, Note that since for any space in $\mathcal{S}_3(f)$, it is not contained in any space in $\mathcal{S}_2(f)$, we only need to consider those $W' = \mathbb{V}(l, l')$ such that $C \mod \langle l, l' \rangle \neq 0$. Since $C' = (C \mod \langle l, l' \rangle) \neq 0$ but $(C \mod \langle l, l', l'' \rangle) = 0$ thus l'' must be a linear factor of C' and hence there can be only d choices for l''. Thus in this case there are at most $O(d^8)$ possibilities for $W \in \mathcal{S}_3(f)$.

B **Case 2(b):** We bound the number of $W = \mathbb{V}(l_1, l_2, l_3) \in \mathcal{S}_3(f)$ such that no T_i vanishes on $\mathbb{V}(l_1, l_2, l_3)$ and there do not exist two independent linear forms $l, l' \in \text{span}(l_1, l_2, l_3)$, such that $\text{rank}(\text{sim}(C \mod \langle l, l' \rangle)) < \mathcal{R}(3)$.

Consider 3 linear forms (not all the same) l_{1j}, l_{2j}, l_{3j} from distinct product gates of $\text{sim}(C)$ such that when we consider them $\mod \langle l_1, l_2, l_3 \rangle$, they map to the same linear form l (we don't distinguish between a linear form and its multiple) and hence all get mapped to the linear forms of G. Call such triple of linear forms a "collapsing" triple when we go $\mod \langle l_1, l_2, l_3 \rangle$. These linear forms must be of the form

$$l_{1j} = l + \alpha_1 l_1 + \alpha_2 l_2 + \alpha_3 l_3$$
$$l_{2j} = l + \beta_1 l_1 + \beta_2 l_2 + \beta_3 l_3$$
$$l_{3j} = l + \gamma_1 l_1 + \gamma_2 l_2 + \gamma_3 l_3$$

where the α, β, γ denote field constants.

A collapsing triple can either span a 2 or 3 dimensional space. Now, since the rank of linear forms mapping to G is at least 14, one of the following two scenarios must occur. Either (i) The set of collapsing triples such that each spans a 3-dim space has combined rank (over all the triples) of at least 7 or (ii) The set of collapsing triples such that each spans a 2-dim space has combined rank at least 8.

We will separately analyze both these subcases.

- **Case 2(b)(i):**The set of collapsing triples such that each spans a 3-dim space has combined rank (over all the triples) of at least 7.
  Let V_1, V_2, V_3 be the vector spaces spanned by three of the triples such that the V_1, V_2 and V_3 are distinct and $\text{span}(V_1 \cup V_2 \cup V_3) \geq 5$. Three such vector spaces must exist.
  Now since going $\mod \langle l_1, l_2, l_3 \rangle$ maps these triples to a line l, letting
  $U = \text{span}(l_1, l_2, l_3)$, it must hold that for all $i \in [3]$, $\dim(V_i \cap U) = 2$. It follows that for all $i, j \in [3]$, $\dim(V_i \cap V_j) \geq 1$. Since the spaces are distinct, thus $\dim(V_i \cap V_j)$ can equal 1 or 2.
  Now if for any $i, j \in [3]$, $\dim(V_i \cap V_j) = 1$ then thus intersection must be contained in U. Thus knowing V_i and V_j determines a 1-dim subspace of U. Let l' be the linear

form representing this space. Since l' is determined by two of the triples, thus there are at most $O(d^6)$ possibilities for l'. Once l' is determined, we consider $C' = C \mod l'$. This is nonzero and by assumption, the rank of its simple part is at least c_2. By Lemma 4.3 there are only $O(d^7)$ choices of co-dim 2 subspaces modulo which C' vanishes (we disregard those co-dim 2 spaces which contain a co-dim 1 space on which C' vanishes. Thus in total, in this case there are at most $O(d^{13})$ choices of $W = \mathbb{V}(l_1, l_2, l_3) \in \mathcal{S}_3(f)$.

We need to still consider the case when for all distinct $i, j \in [3]$, $\dim(V_i \cap V_j) = 2$. We first prove the following simple claim.

Claim 4.5. *Let V_1, V_2, V_3 be distinct 3-dim subspaces of some vector space V such that $\dim(\text{span}(V_1, V_2, V_3)) \geq 5$. Suppose that for any distinct $i, j \in [3]$, $\dim(V_i \cap V_j) = 2$. Then there exists a subspace $U' \subseteq V$ such that $\dim(U') = 2$ such that for any distinct $i, j \in [3]$, $(V_i \cap V_j) = U'$.*

Proof. Since V_1, V_2 are distinct 3-dimensional spaces with $\dim(V_1 \cap V_2) = 2$, we have $\dim(V_1 \cup V_2) = 4$. As $\dim(\text{span}(V_1, V_2, V_3)) \geq 5$, we have $V_3 \not\subset V_1 \cup V_2$. If V_3 intersects V_1 and V_2 in distinct 2-dimensional spaces, then $\dim((V_3 \cap V_1) \cup (V_3 \cap V_2)) \geq 3$. But $(V_3 \cap V_1) \cup (V_3 \cap V_2) = V_3 \cap (V_1 \cup V_2) \subseteq V_3$ and as $\dim(V_3) = 3$, $V_3 = V_3 \cap (V_1 \cup V_2) \subseteq (V_1 \cup V_2)$, which is a contradiction. Therefore, V_3 intersects V_1 and V_2 in the same 2-dimensional plane, let it be $U' = V_1 \cap V_3 = V_2 \cap V_3$. Moreover, this means $U' \subset V_1$ and $U' \subset V_2$, and therefore $U' \subseteq (V_1 \cap V_2)$. As $\dim(V_1 \cap V_2) = 2$, we have $U' = (V_1 \cap V_2)$. □

By the above claim, it follows that $V_1 \cap V_2 \cap V_3 = U'$ for some 2-dim space U'. Moreover we know that U is a 3-dim space intersecting each V_i in a 2-dim space. It must hold that $U' \subseteq U$. If not, then $\dim(U' \cap U) \leq 1$. Thus U needs to still intersect each V_i in a vector outside of U'. These three additional vectors will be distinct and also linearly independent since $\text{span}(V_1 \cup V_2 \cup V_3) \geq 5$. This is not possible since $\dim(U) = 3$.

Thus $U' \subseteq U$. Hence V_1 and V_2 determine U' and hence determine a 2-dim subspace of U. Since U' is determined by two of the triples, thus there are at most $O(d^6)$ possibilities for U'. Once we fix $U' \subseteq U$, it remains to find the number of $W = \mathbb{V}(l_1, l_2, l_3) \in \mathcal{S}_3(f)$ such that $U' \in \text{span}(l_1, l_2, l_3)$. When we go mod U', we can assume that C does not vanish (since we are only counting those W that are not contained in an $\mathcal{S}_2$ space), and hence for each fixing of U', since $C \mod U'$ has only at most d linear factors, thus there are at most $O(d^7)$ possibilities for W.

- **Case 2(b)(ii):**The set of collapsing triples such that each spans a 2-dim space, has combined rank at least 8.

  Let P_1, P_2, P_3, P_4 be the vector spaces spanned by four of the triples such that the P_1, P_2, P_3 and P_4 are distinct and $\text{span}(P_1, P_2, P_3, P_4) \geq 5$. Moreover $P_3 \not\subseteq \text{span}(P_1, P_2)$ and $P_4 \not\subseteq \text{span}(P_1, P_2, P_3)$. Four such vector spaces must exist.

  Now since going $\mod \langle l_1, l_2, l_3 \rangle$ maps these triples to a single line l, letting $U = \text{span}(l_1, l_2, l_3)$, it must hold that for all $i \in [4]$, $\dim(P_i \cap U) = 1$.

  There are three subcases that we will consider. In each case we show that the knowledge of the four subspaces P_i allows us to determine a single linear form $l' \in U$. There are at most hence d^8 choices for l'. Once we find and fix $l' \in U$ we consider $C' = C \mod l'$. This is nonzero and by assumption, the rank of its simple part is at least c_2. By Lemma 4.3 there are only $O(d^7)$ choices of co-dim 2 subspaces modulo which C' vanishes (we disregard those co-dim 2 spaces which

contain a co-dim 1 space on which C' vanishes. Thus in total, the number of choices of $W = \mathbb{V}(l_1, l_2, l_3) \in \mathcal{S}_3(f)$ is $O(d^{15})$.

- **subcase 1:** P_1, P_2 are such that $P_1 \cap U = P_2 \cap U$.
  In particular $P_1 \cap P_2 \subset U$. Thus knowing P_1 and P_2 determines a 1-dim subspace of U.
- **subcase 2:** The 2-dim space defined by $P' = \text{span}(U \cap P_1, U \cap P_2) \subset U$ contains the line $U \cap P_3$. Since $P_3 \not\subseteq \text{span}(P_1, P_2)$, thus $\dim(\text{span}(P_1, P_2, P_3)) \geq 4$ and $\dim(P' \cap P_1) = \dim(P' \cap P_2) = \dim(P' \cap P_3) = 1$.
  Using Claim 4.2, it follows that P' and hence U contains the line $P_3 \cap \text{span}(P_1, P_2)$.
- **subcase 3:** $\dim(\text{span}(U \cap P_1, U \cap P_2, U \cap P_3)) = 3$ i.e. the three 1-dim intersections are independent. But as $\dim(U) = 3$ and $\text{span}(U \cap P_1, U \cap P_2, U \cap P_3) \subseteq U$, we have $U = U \cap \text{span}(P_1, P_2, P_3)$, which means $U \subset \text{span}(P_1, P_2, P_3)$. As $\dim(P_4 \cap U) = 1$, we have $\dim(P_4 \cap \text{span}(P_1, P_2, P_3)) \geq 1$. By assumption, $P_4 \not\subseteq \text{span}(P_1, P_2, P_3)$ and therefore $\dim(P_4 \cap \text{span}(P_1, P_2, P_3)) \leq 1$. So, $P_4 \cap \text{span}(P_1, P_2, P_3) = U \cap P_4$ is a 1-dim space that is contained in U, and is determined by knowing P_1, P_2, P_3 and P_4

□

5 Algorithmically computing $\mathcal{S}_2(f)$ and $\mathcal{S}_3(f)$

This section aims to compute the set of vanishing spaces of codimension up to 3 for polynomials that can be computed by $\Sigma\Pi\Sigma(3)$ circuits. We showed in the previous section, under some rank constraints, these sets have size $d^{O(1)}$. The set of codimension 1 space is easy to compute as it is just the set of linear factors of the polynomial, which we can compute using the randomized black box factoring algorithm in Lemma 3.7. We will be working on proving the following two lemmas

Lemma 5.1. *Let $\mathbb{F}$ be a field that is $\mathbb{R}$ or $\mathbb{C}$. Let f be a n-variate, degree d polynomial in $\mathbb{F}[x_1, \ldots, x_n]$ that is computed by a $\Sigma\Pi\Sigma(3)$ circuit C with $\text{rank}(\text{sim}(C)) \geq c_2$, where c_2 is as in Lemma 4.3. Then, there exists a randomized algorithm(Algorithm 2) that outputs $\mathcal{S}_2(f)$ in $\text{poly}(n, d)$-time with probability $1 - o(1)$.*

Lemma 5.2. *Let $\mathbb{F}$ be a field that is $\mathbb{R}$ or $\mathbb{C}$. Let f be a n-variate, degree d polynomial in $\mathbb{F}[x_1, \ldots, x_n]$ that is computed by a $\Sigma\Pi\Sigma(3)$ circuit C such that $\text{rank}(\text{sim}(C)) \geq c_3$ and there doesn't exist a linear form $l \in \mathbb{F}[x_1, \ldots, x_n]$ with $\text{rank}(\text{sim}(C \mod l)) < c_2$, where c_2 is as in Lemma 4.3 and c_3 is as in Lemma 4.4. Then, there exists a randomized algorithm (Algorithm 4) that outputs $\mathcal{S}_3(f)$, in $\text{poly}(n, d)$-time with probability $1 - o(1)$.*

We divide the remaining section into two subsections for computing the codimension 2 and 3 spaces over which f vanishes, i.e. $\mathcal{S}_2(f)$ and $\mathcal{S}_3(f)$ respectively.

5.1 Computing $\mathcal{S}_2(f)$

Recall that in Lemma 4.3, we have defined $\mathcal{S}_2(f)$ as

$$\mathcal{S}_2(f) = \{W \mid W \text{ is a codimension 2 subspace of } \mathbb{F}^n, f \text{ vanishes over } W \text{ and } W \not\subset W' \text{ for any } W' \in \mathcal{S}_1(f)\}$$

We will first discuss computing $\mathcal{S}_2(f)$ when f depends only on a constant number of variables t such that $t \geq c_2$. Then, solve the general case computation of $\mathcal{S}_2$ spaces by solving it on multiple instances of constant variate cases and gluing them together.

5.1.1 Computing $\mathcal{S}_2(f)$ for constant variate polynomials

Lemma 5.3. *Let $\mathbb{F}$ be a field that is $\mathbb{R}$ or $\mathbb{C}$. Let f be a t-variate, degree d polynomial in $\mathbb{F}[x_1, \ldots, x_t]$ that is computed by a $\Sigma\Pi\Sigma(3)$ circuit C with $\text{rank}(\text{sim}(C)) = t \geq c_2$, where c_2 is as in Lemma 4.3. Then, there exists a randomized algorithm(Algorithm 1) that outputs $\mathcal{S}_2(f)$ in $\text{poly}(d^{O(t^t)})$-time with probability $1 - o(1)$.*

Proof. Let Φ be a random linear isomorphism on $\mathbb{F}[x_1, \ldots, x_t]$ such that $\forall i \in [t], \Phi(x_i) = \sum_{j=1}^{t} \alpha_{ij} x_j$ where α_{ij} are sampled randomly from $[d^t]$. We first observe that if f vanishes over a codimension 2 space $\mathbb{V}(l_1, l_2)$, then after a random linear isomorphism Φ on the variables, $g = \Phi(f) = f(\Phi(x))$ will vanish over a space $\mathbb{V}(\Phi(l_1), \Phi(l_2))$ and moreover this space can be represented in the form $\mathbb{V}(x_1 - l_a, x_2 - l_b)$ for linear forms $l_a, l_b \in \mathbb{F}[x_3, \ldots, x_n]$.

Let $l_a = a_3 x_3 + \ldots + a_t x_t$ and $l_b = b_3 x_3 + \ldots + b_t x_t$. The basic idea in the constant variate case is that we can substitute $x_1 = a_3 x_3 + \ldots + a_t x_t = l_a$ and $x_2 = b_3 x_3 + \ldots + b_t x_t = l_b$ into the monomial representation of g, and obtain the polynomial $g \mod \langle x_1 - l_a, x_2 - l_b \rangle$. Since we are interested in the case when f vanishes over codimension 2 spaces, we equate coefficients of monomials over the variables $x_3, \ldots, x_t$ to 0 to get a system of polynomial equations treating a_i, b_i as formal variables, and therefore it will have $2t-4$ variables. This system of polynomial equations might have infinitely many solutions unless we discard all those codimension 2 spaces that are contained in $\mathcal{S}_1$ spaces. We also know from Lemma 4.3 that this suffices as $t \geq c_2$. The challenge remains to remove the codimension 2 over which f vanishes that are contained in spaces in $\mathcal{S}_1(f)$. To do that, we add an additional polynomial equation to the system of polynomial equations, that ensures for any $\mathbb{V}(l) \in \mathcal{S}_1(g)$, $\dim(\text{span}(l, x_1 - l_a, x_2 - l_b)) = 3$. Finally, having computed the spaces $\mathbb{V}(x_1 - l_a, x_2 - l_b)$ on which g vanishes, we simply apply Φ^{-1} to get $\mathbb{V}(l_1, l_2)$.

We now give a more detailed analysis.

We first observe that in Step 1 of Algorithm 1, the random linear forms $l'_1, \ldots, l'_t$ will be independent with high probability (as otherwise it will correspond to a certain determinant evaluating to 0, which happens with probability at most $d^{-(t-1)}$ due to Lemma 3.2).

Thus will high probability Φ is a random isomorphism, and we obtain the polynomial $g = \Phi(f)$ which is also computable by a $\Sigma\Pi\Sigma(3)$ circuit over t variables, and the simple part of the circuit has rank t. From now onwards let us assume that Φ is an isomorphism.

g vanishes on spaces of the form $\mathbb{V}(x_1 - l_a, x_2 - l_b)$ As Φ is a random isomorphism, f vanishes on a codimension 2 space $\mathbb{V}(l_1, l_2)$ if and only if g vanishes on $\mathbb{V}(\Phi(l_1), \Phi(l_2))$. We will first observe that with high probability, for any space $\mathbb{V}(\Phi(l_1), \Phi(l_2)) \in \mathcal{S}_2(g)$, there are linear forms $l_a, l_b \in \mathbb{F}[x_3, \ldots, x_n]$ such that $\mathbb{V}(x_1 - l_a, x_2 - l_b) = \mathbb{V}(\Phi(l_1), \Phi(l_2))$. The reason is the following: Let $l_1 = u_1 x_1 + \ldots + u_t x_t$ and $l_2 = v_1 x_1 + \ldots + v_t x_t$. As $\mathbb{V}(l_1, l_2)$ is a codimension 2 space, l_1 and l_2 are not scalar multiples of each other. After applying the isomorphism Φ, they remain independent with high probability. The coefficients of x_i in $\Phi(l_1)$ can be expressed as $\sum_{j=1}^{n} \alpha_{i,j} u_j x_j$ and similarly for $\Phi(l_2)$ they will be $\sum_{j=1}^{n} \alpha_{i,j} v_j x_j$. As they were independent, the determinant of the 2×2 matrix formed by the coefficients of x_1, x_2 from $\Phi(l_1), \Phi(l_2)$ will be a non-zero polynomial in $\alpha_{1,1}, \ldots, \alpha_{2,n}$ and will vanish with vanishingly small probability due to Lemma 3.2. This means that for space $\mathbb{V}(\Phi(l_1), \Phi(l_2)) \in \mathcal{S}_2(g)$ there is a space $\mathbb{V}(x_1 - l_a, x_2 - l_b) \in \mathcal{S}_2(g)$ where $l_a, l_b \in \mathbb{F}[x_3, \ldots, x_n]$.

Setting up a system of equations Observe that we can use interpolation to get monomial access to g in time $\text{poly}(d^t)$ using Lemma 3.6.

We set $l_a := a_3x_3 + \ldots + a_tx_t$ and $l_b := b_3x_3 + \ldots + b_tx_t$ for variables $a_3, \ldots, a_t, b_3, \ldots, b_t$. Substituting $x_1 = l_a, x_2 = l_b$ into the monomial form, we obtain a system of $d^{O(t)}$ equations of degree at most d in $2(t-2)$ variables by equating the coefficients of monomials in the variables $x_3, \ldots, x_t$ to 0. Solutions to this would correspond to codimension 2 spaces that g vanishes on.

To remove the codimension 2 spaces that are contained in $\mathcal{S}_1(f)$, we first compute the set of all linear factors, $\mathcal{S}_1(g)$, by using a blackbox factoring algorithm as in Lemma 3.7, and then obtain monomial access to all the linear factors by interpolating them in $\text{poly}(d, t)$-time.

Then, we need to ensure that the solution to our system of equations $x_1 - l_a, x_2 - l_b$ is such that $\forall l$ such that $\mathbb{V}(l) \in \mathcal{S}_1(g)$ we have that $\dim(\text{span}(x_1 - l_a, x_2 - l_b, l)) = 3$. This is the same as saying that the $t \times 3$ matrix A_l with $x_1 - l_a, x_2 - l_b, l$ as rows has rank 3, which means at least one of the 3×3 minors is full-rank and has a non-zero determinant. Let the number of such minors be $k = \binom{t}{3}$. To handle these constraints, we introduce new variables $y_1, \ldots, y_k$, and for each relevant l we consider the inequality $sum_l = \sum_{j=1}^{k} y_j M_j \neq 0$, where the M_j are the determinants of the 3×3 minors of A_l. The inequality has solutions if and only if there exists a solution for which at least one of the $M_j's$ is non-zero. So now, we have $|\mathcal{S}_1(g)|$ (which is $\leq d$) inequalities in our system along with the previous equations. We note that we use the same variables $y_1, \ldots, y_k$ in all of the inequalities. Let $sum_l = \sum_{j=1}^{k} y_j M_j$ for all $\mathbb{V}(l) \in \mathcal{S}_1(g)$. Observe that the set of inequalities $\forall\, \mathbb{V}(l) \in \mathcal{S}_1(g)\ sum_l \neq 0$ is the same as having a single inequality $(\prod_{\mathbb{V}(l) \in \mathcal{S}_1(g)} sum_l) \neq 0$, which is same as requiring that $(\prod_{\mathbb{V}(l) \in \mathcal{S}_1(g)} sum_l) \cdot z = 1$ has a solution for a new variable z.

Thus, we can handle the condition of the solution not lying in any $\mathcal{S}_1(g)$ space, by simply adding one extra equation of degree $4|\mathcal{S}_1(g)| + 1 = 4d + 1$ and $k + 1$ additional variables to the system of equations we had earlier.

Running Time Analysis The sampling of the random $\alpha_{i,j}$ can be done in randomized $\text{poly}(t, \log d)$ time. From Lemma 3.7, we can get black-box access to the factors in time randomized $\text{poly}(t, d)$. We can do interpolation and get monomial access to g in time $d^{O(t)}$ using Lemma 3.6. As $|\mathcal{S}_1(g)| \leq d$, the loop on line 7 runs $O(d)$ times and does $\text{poly}(t)$ computation. Finally, the system of equations has $d^{O(t)}$ equations with degree $O(d)$ in $\text{poly}(t)$ variables. Therefore, we can find l_a, l_b by solving the system of equations in time $\text{poly}(d^{O(t^t)})$ using Theorem 3.8. Thus the entire algorithm works in $\text{poly}(d^{O(t^t)})$ time. □

Algorithm 1 Computing Vanishing $\mathcal{S}_2(f)$ for constant variate polynomials

Input: Blackbox access to circuit C of form $\Sigma\Pi\Sigma(3)$ computing polynomial $f \in \mathbb{F}[x_1, \ldots, x_t]$, $t \geq c_2$ and $\text{rank}(\text{sim}(C)) = t$

1: **function** $\mathcal{S}_2(f)$
2: Sample t^2 random values $\alpha_{ij}; i, j \in [t]$ uniformly from $\{1, \ldots, d^t\}$, and use them to define t linear forms $l'_i = \sum_{j=1}^{t} \alpha_{ij} x_j$. Check if they are independent, otherwise output error. Define isomorphism Φ such that for all $i \in [t]$, $\Phi(x_i) := l'_i$. Let $g = \Phi(f) = f(\Phi(x))$.
3: Using randomized black-box factoring from Lemma 3.7 and get access to the linear factors of g
4: Interpolate g to get monomial access to it and interpolate the linear factors to obtain $\mathcal{S}_1(g) := \{\mathbb{V}(l) : l | g\}$
5: Substitute $x_1 = a_3 x_3 + \ldots + a_{t-1} x_{t-1} + a_t x_t$ for linear form $x_1 - l_a$, $x_2 = b_3 x_3 + \ldots + b_{t-1} x_{t-1} + b_t x_t$ for linear form $x_2 - l_b$ in g and obtain equations in $a_3, \ldots, a_t, b_3, \ldots, b_t$ by equating coefficients of monomials in the variables $x_3, x_4, \ldots, x_{t-1}, x_t$ to 0.
6: Let $k = \binom{t}{3}$. Introduce new variables $z, y_1, \ldots, y_k$
7: **for** $\mathbb{V}(l) \leftarrow \mathcal{S}_1(g)$ **do**
8: Consider the $3 \times t$ matrix A_l formed by $l, x_1 - l_a, x_2 - l_b$.
9: Compute determinant of each 3×3 minor M_j of A_l. Compute and store $sum_l := \sum_{j=1}^{k} y_j M_j$
10: Add Equation $(\prod_{\mathbb{V}(l) \in \mathcal{S}_1(g)} sum_l) \cdot z = 1$ to the system of equations in Step 5.
11: Solve the system of equations in $a_3, \ldots, a_t, b_3, \ldots, b_t, y_1, \ldots, y_k, z$ using Theorem 3.8 to obtain a set of $(x_1 - l_a, x_2 - l_b)$.
12: Verify for each $(x_1 - l_a, x_2 - l_b)$ if f vanishes on $\mathbb{V}(\Phi^{-1}(x_1 - l_a), \Phi^{-1}(x_2 - l_b)) = \mathbb{V}(l_1, l_2)$ using Lemma 3.2, then add $\mathbb{V}(l_1, l_2)$ to $\mathcal{S}_2(f)$.
13: **Output** $\mathcal{S}_2(f)$.

5.1.2 Computing $\mathcal{S}_2(f)$ general case

We will now discuss how we can use the solution for the constant variate case to compute $\mathcal{S}_2(f)$ in the general case.

We will start by using a random linear isomorphism Φ on f such that $\Phi(x_i) = \sum_{j=1}^{n} \alpha_{ij} x_j$, where α_{ij} are chosen randomly from $[d^n]$, and define $g = \Phi(f) = f(\Phi(x))$. Let $t = c_2 + 1$. We will then consider the t variate polynomials g_i (for $i \geq t$) which are obtained from g by setting all variables x_j for $j > t - 1$ to zero, except x_i.

Thus

$$g_i = g|_{x_t = \ldots = x_{i-1} = x_{i+1} = \ldots = x_n = 0}$$

We will then find $\mathcal{S}_2(g_i)$ spaces using the constant variate algorithm and then show how to glue the learned spaces to get $\mathcal{S}_2(g)$ and then $\mathcal{S}_2(f)$.

We will need the following collection of simpler properties about g_i's summed up into Lemma 5.4 to prove the correctness of the algorithm computing $\mathcal{S}_2(f)$.

Lemma 5.4. *With probability $1 - o(1)$, the following hold.*

1. *For each $i \in \{t, \ldots, n\}$, the polynomials g_i can be computed by $\Sigma\Pi\Sigma(3)$ circuits C_i with $\text{rank}(\text{sim}(C_i)) = t = c_2 + 1$*
2. *For each $i \in \{t, \ldots, n\}$, $\mathcal{S}_1(g_i) = \{\Phi(l) : l \in \text{Lin}(f)\}|_{x_t = \ldots = x_{i-1} = x_{i+1} = \ldots = x_n = 0}$. In other words, no new linear factors arise after setting some of the variables to 0.*

3. *If* $\mathbb{V}(x_1 - l_1, x_2 - l_2) \in \mathcal{S}_2(g)$*, where* $l_1, l_2 \in \mathbb{F}[x_3, \ldots, x_n]$ *then* $\mathbb{V}(x_1 - l_{1i}, x_2 - l_{2i}) \in \mathcal{S}_2(g_i)$*, where* $l_{ji} = l_j|_{x_t=0,\ldots,x_{i-1}=0,x_{i+1}=0,\ldots,x_n=0}$*. In other words, the projected codimension 2 space continues to be a vanishing space as also continues to not lie in a codimension 1 vanishing space.*

Proof. The proof of this lemma is very similar to the proof of Lemma 5.3 in [Sin22]. We prove all three items below.

1. The circuit C after applying the isomorphism Φ will be of the form

$$\Phi(C) = \Phi(G) \times (\Phi(T_1) + \Phi(T_2) + \Phi(T_3))$$

Since Φ is an isomorphism, we have $\gcd(\Phi(T_1), \Phi(T_2), \Phi(T_3)) = 1$. We denote Γ_i as the homomorphism from $\mathbb{F}[x_1 \ldots, x_n]$ to $\mathbb{F}[x_1, \ldots, x_{t-1}, x_i]$ mapping $x_j \to 0$ for $j \in \{t, \ldots, i-1\} \cup \{i+1, \ldots, n\}$. Then for each i, g_i is computable by the following circuit

$$g_i = \Gamma_i(g) = \Gamma_i(\Phi(G)) \times (\Gamma_i(\Phi(T_1)) + \Gamma_i(\Phi(T_2)) + \Gamma_i(\Phi(T_3)))$$

We will first argue that with high probability $\gcd(\Gamma_i(\Phi(T_1), \Phi(T_2), \Phi(T_3))) = 1$. Consider any two linear forms $l = \sum_{i=1}^n c_i x_i$ and $l' = \sum_{i=1}^n c'_i x_i$ that $\text{span}(l) \neq \text{span}(l')$ that appear in the circuit. After applying Φ, the coefficients of x_1, x_2 in $\Phi(l), \Phi(l')$ will be $\sum_{i=1}^n \alpha_{1,i} c_i x_i$ and $\sum_{i=1}^n \alpha_{2,i} c_i x_i$ for $\Phi(l)$ and $\sum_{i=1}^n \alpha_{1,i} c'_i x_i$ and $\sum_{i=1}^n \alpha_{2,i} c'_i x_i$ for $\Phi(l')$. As $\text{span}(l) \neq \text{span}(l')$, if these two linear forms were in distinct gates, they would become equal and move to the gcd part only if the determinant of the 2×2 matrix with these coefficients as entries becomes identically 0. This happens with vanishingly small probability (by Lemma 3.2) as the determinant is a non-zero polynomial in $\alpha_{1,1}, \ldots, \alpha_{2,n}$ and we we choose $\alpha_{i,j}$ from a large set of size d^n.

Also, as we assumed that $\text{rank}(\text{sim}(C)) \geq t$, thus $\dim(\text{span}(\{l : l | T_1 \times T_2 \times T_3\})) \geq t$. Therefore, for each i we conclude that $\dim(\text{span}(\{\Gamma_i(\Phi(l)) : l | T_1 \times T_2 \times T_3\}))) = t$, which is the same set as $\{l : l | \Gamma_i(\Phi(T_1)) \times \Gamma_i(\Phi(T_2)) \times \Gamma_i(\Phi(T_3))\}$ as nothing moved to gcd. Therefore, $\text{rank}(\text{sim}(g_i)) = t$.

2. From Theorem 3.5 (Effective Hilbert Irreducibility), we have that for random projections that keep at least 3 variables alive, the irreducible factors of f remain irreducible with high probability, and hence no new linear factors are introduced. We make a random isomorphism and then reduce it to $t = c_2 > 3$ variables using Φ, Γ_i, and hence the set of linear factors of g_i should be precisely $\{\Gamma_i(\Phi(l)) : l | f\}$. Thus with high probability $\mathcal{S}_1(g_i) = \{\mathbb{V}(\Gamma_i(\Phi(l)) : \mathbb{V}(l) \in \mathcal{S}_1(f)\}$.

3. As we already saw in proof of Lemma 5.3, for any $\mathbb{V}(l_1, l_2) \in \mathcal{S}_2(g)$, it can be represented in the form $\mathbb{V}(x_1 - l'_1, x_2 - l'_2)$ where $l'_1, l'_2 \in \mathbb{F}[x_3, \ldots, x_n]$. It is fairly straightforward to see that if g vanishes on $\mathbb{V}(x_1 - l'_1, x_2 - l'_2)$ then g_i vanishes on $\mathbb{V}(x_1 - l'_{1i}, x_2 - l'_{2i})$ where $l'_{ji} = l'_j|_{x_t=0,\ldots,x_{i-1}=0,x_{i+1}=0,\ldots,x_n=0}$.

Also from part 2 of this lemma, we have that $\mathcal{S}_1(g_i) = \{\mathbb{V}(\Gamma_i(l)) : \mathbb{V}(l) \in \mathcal{S}_1(g)\}$, which means if $\mathbb{V}(x_1 - l'_1, x_2 - l'_2)$ weren't contained in any $\mathcal{S}_1(g)$ space, then $\mathbb{V}(x_1 - l'_{1i}, x_2 - l'_{2i})$ is also not contained in any $\mathcal{S}_1(g_i)$ space. Both of the above combine to give us that $\mathbb{V}(x_1 - l'_{1i}, x_2 - l'_{2i}) \in \mathcal{S}_2(g_i)$.

□

Now, provide the proof of Lemma 5.1, which we restate here for clarity.

Lemma 5.1 restated. *Let $\mathbb{F}$ be a field that is $\mathbb{R}$ or $\mathbb{C}$. Let f be a n-variate, degree d polynomial in $\mathbb{F}[x_1, \ldots, x_n]$ that is computed by a $\Sigma\Pi\Sigma(3)$ circuit C with $\text{rank}(\text{sim}(C)) \geq c_2$, where c_2 is as in Lemma 4.3. Then, there exists an algorithm(Algorithm 2) that outputs $\mathcal{S}_2(f)$ in $\text{poly}(n, d)$-time with probability $1 - o(1)$.*

Proof of Lemma 5.1. We first apply a random linear isomorphism and obtain $g = \Phi(f)$. For $t = c_2 + 1$, we obtain $n - t$ polynomials $g_i \in \mathbb{F}[x_1, \ldots, x_{t-1}, x_i]$ by setting all except $x_1, \ldots, x_{t-1}, x_i$ to 0. Then we solve the constant variate cases using Algorithm 1 discussed above to recover $\mathcal{S}_2(g_i)$. We will then recover $\mathcal{S}_2(g)$ by gluing together spaces from $\mathcal{S}_2(g_i)$ (across different choices of i) when the spaces are consistent when restricted to $x_1, \ldots, x_{t-1}$.

By part 1 of Lemma 5.4, the random invertible linear isomorphism ensures that when we set some of the variables to 0, the rank of the linear forms present in the circuits computing the g_i's remains high (equal to t) with high probability. Also, from part 2 or Lemma 5.4, we know the set $\mathcal{S}_1(g_i)$ contains exactly the linear factors of g projected down to the t variables in g_i.

From part 3 of Lemma 5.4, we have for every $\mathbb{V}(x_1 - l_1, x_2 - l_2) \in \mathcal{S}_2(g)$, there is $\mathbb{V}(x_1 - l_{1i}, x_2 - l_{2i}) \in \mathcal{S}_2(g_i)$, where $l_{ji} = l_j|_{x_t=0,\ldots,x_{i-1}=0,x_{i+1}=0,\ldots,x_n=0}$. Moreover we can find $\mathcal{S}_2(g_i)$ in $\text{poly}(d)^{\text{poly}(t)}$ time as described in Lemma 5.3.

To obtain $\mathcal{S}_2(g)$ (wlog of the form $\mathbb{V}(x_1 - l_1, x_2 - l_2)$), we will show how to glue these spaces learned in the constant variate case. To do this, we will look at spaces in $\mathcal{S}_2(g_i)$ and $\mathcal{S}_2(g_j)$ (for $i \neq j$) and "glue" them if they are consistent in the first $t-1$ variables. For this to be efficient, it will be very useful to have the property that distinct spaces in $\mathcal{S}_2(g_i)$ are distinct when restricted to the first $t - 1$ coordinates. This is probably already true with high probability due to the randomness of Φ. However to make the argument simpler to analyze, we consider and apply another random linear isomorphism Ψ (we will apply these to the spaces in $\mathcal{S}_2(g_i)$ for each i) defined as follows: $\forall i < t$, $\Psi(x_i) = x_i$ and $\forall i \in [t, n], \Psi(x_i) = x_i + \beta_{i,3}x_3 + \ldots + \beta_{i,t-1}x_{t-1}$ where $\beta_{i,j}$ are sampled independently and uniformly from $[d^n]$. The goal of introducing the map Ψ is to ensure that distinct spaces in $\mathcal{S}_2(g_i)$ are distinct when restricted to the first $t-1$ coordinates, and we prove this formally in the claim below.

Claim 5.5. *For all $i \in \{t, \ldots, n\}$, let $\mathbb{V}(x_1 - l_1, x_2 - l_2)$ and $\mathbb{V}(x_1 - l'_1, x_2 - l'_2)$ be distinct spaces in $\mathcal{S}_2(g_i)$ such that l_1, l_2, l'_1, l'_2 only depend on $x_3, \ldots, x_{t-1}, x_i$. Then $\mathbb{V}(\Psi(x_1 - l_1)_{x_i=0}, \Psi(x_2 - l_2)_{x_i=0}) \neq \mathbb{V}(\Psi(x_1 - l'_1)_{x_i=0}, \Psi(x_2 - l'_2)_{x_i=0})$. In particular*

$$\langle \Psi(x_1 - l_1)|_{x_i=0}, \Psi(x_2 - l_2)|_{x_i=0} \rangle \neq \langle \Psi(x_1 - l'_1)|_{x_i=0}, \Psi(x_2 - l'_2)|_{x_i=0} \rangle$$

Proof. The proof of this claim is very similar to Lemma 5.4 of [Sin22]. Consider 2 distinct elements $\mathbb{V}(x_1 - l_1, x_2 - l_2)$ and $\mathbb{V}(x_1 - l'_1, x_2 - l'_2)$ in $\mathcal{S}_2(g_i)$, with $l_1, l'_1, l_2, l'_2 \in \mathbb{F}[x_3, \ldots, x_{t-1}, x_i]$. Let $l_1 = a_3x_3 + \ldots + a_{t-1}x_{t-1} + a_ix_i$, $l_2 = b_3x_3 + \ldots + b_{t-1}x_{t-1} + b_ix_i$, $l'_1 = a'_3x_3 + \ldots + a'_{t-1}x_{t-1} + a'_ix_i$ and $l'_2 = b'_3x_3 + \ldots + b'_{t-1}x_{t-1} + b'_ix_i$. Therefore, we have

$$\begin{aligned}
l_1 &= (a_3 + \beta_{i,3}a_i)x_3 + \ldots + (a_{t-1} + \beta_{i,t-1}a_i)x_{t-1} + a_ix_i \\
l_2 &= (b_3 + \beta_{i,3}b_i)x_3 + \ldots + (b_{t-1} + \beta_{i,t-1}b_i)x_{t-1} + b_ix_i \\
l'_1 &= (a'_3 + \beta_{i,3}a'_i)x_3 + \ldots + (a'_{t-1} + \beta_{i,t-1}a'_i)x_{t-1} + a'_ix_i \\
l'_2 &= (b'_3 + \beta_{i,3}b'_i)x_3 + \ldots + (b'_{t-1} + \beta_{i,t-1}b'_i)x_{t-1} + b'_ix_i
\end{aligned}$$

Now, if $\langle \Psi(x_1 - l_1)|_{x_i=0}, \Psi(x_2 - l_2)|_{x_i=0} \rangle = \langle \Psi(x_1 - l'_1)|_{x_i=0}, \Psi(x_2 - l'_2)|_{x_i=0} \rangle$, this gives rise to system of linear equations in $\beta_{i,3}, \ldots, \beta_{i,t-1}$ given by $\forall j \in \{3, \ldots, t-1\}$, $\beta_{i,j}(a_i - a'_i) = (a_j - a'_j)$ and $\forall j \in \{3, \ldots, t-1\}$, $\beta_{i,j}(b_i - b'_i) = (b_j - b'_j)$. Since $\langle x_1 - l_1, x_2 - l_2 \rangle$ and $\langle x_1 - l'_1, x_2 - l'_2 \rangle$ are distinct, there is some choice of j for which at least one of $a_j \neq a'_j$ or $b_j \neq b'_j$ must hold. This gives a nonzero linear equation in $\beta_{i,j}$ (when viewed as formal variables) which must become zero for the specific choice of sampled values. By using Lemma 3.2, the probability this can happen is $\frac{1}{d^n}$ as we choose the $\beta_{i,j}$ from $[d^n]$. From Lemma 4.3 we know the $|\mathcal{S}_2(g_i)| = d^{O(1)}$, and hence taking union of all pairs from $\mathcal{S}_2(g_i)$ and union over all i, this gives us that with probability $1 - o(1)$ no two spaces in any of the $\mathcal{S}_2(g_i)$ are equal after applying Ψ and setting x_i to zero. □

As described in Algorithm 2, for each $\mathbb{V}(x_1 - l_{t1}, x_2 - l_{t2}) \in \mathcal{S}_2(g_t)$ we "glue" or combine it with a corresponding $\mathbb{V}(x_1 - l_{i1}, x_2 - l_{i2}) \in \mathcal{S}_2(g_i)$ if $\langle \Psi(l_{i1})|_{x_i=0}, \Psi(l_{i2})|_{x_i=0} \rangle = \langle \Psi(l_{t1})|_{x_t=0}, \Psi(l_{t2})|_{x_t=0} \rangle$. We each fixed space in $\mathcal{S}_2(g_t)$ with high probability there is a unique space in $\mathcal{S}_2(g_i)$ where this happens by the above claim. Note that every space in $\mathcal{S}_2(g)$ corresponds to some unique space in $\mathcal{S}_2(g_t)$ (since the spaces in $\mathcal{S}_2(g_t)$ are distinct restricted to first $t-1$ coordinates). To recover the spaces of $\mathcal{S}_2(g)$ with information for all coordinates, for each i, we use the information present in the glued space in $\mathcal{S}_2(g_i)$ to recover the information for the coordinate corresponding to x_i. Thus we obtain all spaces $\mathbb{V}(l_1, l_2)$ on which g vanishes and use Φ^{-1} to obtain $\mathcal{S}_2(f)$. □

Algorithm 2 Computing Vanishing Codimension 2 Subspaces

Input: Blackbox access to circuit C of form $\Sigma\Pi\Sigma(3)$ computing polynomial $f \in \mathbb{F}[x_1, \ldots, x_n]$, $\text{rank}(\text{sim}(C)) \geq c_2$

function $\mathcal{S}_2(f)$

Sample n^2 random values $\alpha_{ij}; i, j \in [n]$ uniformly from $\{1, \ldots, d^n\}$, and use them to define n linear forms $l'_i = \sum_{j=1}^{n} \alpha_{ij} x_j$. Check if they are independent, otherwise repeat. Define isomorphism Φ such that for all $i \in [n]$, $\Phi(x_i) := l'_i$. Let $g = \Phi(f) = f(\Phi(x))$.

Set $t = c_2 + 1$. For $i \in [t, n]$, Obtain $g_i = g|_{x_t = \ldots = x_{i-1} = x_{i+1} = \ldots = x_n = 0}$

For each $g_i, i \in [t, n]$, Compute $\mathcal{S}_2(g_i)$ using Algorithm 1.

We now describe how to glue these spaces across g_i.

Consider an isomorphism Ψ obtained as follows. Sample $(n - t + 1) \times (t - 3)$ random values $\beta_{i,j}$ where $i \in [t, n], j \in [3, t-1]$ uniformly from $\{1, \ldots, d^n\}$. For all $i \in [1, t-1]$, let $\Psi(x_i) = x_i$ and for all $i \in [t, n]$, let $\Psi(x_i) = x_i + \beta_{i,3} x_3 + \ldots + \beta_{i,t-1} x_{t-1}$.

for $\mathbb{V}(x_1 - l_{t1}, x_2 - l_{t2}) \in \mathcal{S}_2(g_t)$ **do**

$l_a := l_{t1}, l_b := l_{t2}$

for $i \in \{t+1, \ldots, n\}$ **do**

Search for $\mathbb{V}(x_1 - l_{i1}, x_2 - l_{i2})$ such that

$$\langle \Psi(x_1 - l_{t1})|_{x_t=0}, \Psi(x_2 - l_{t2})|_{x_t=0} \rangle = \langle \Psi(x_1 - l_{i1})|_{x_i=0}, \Psi(x_2 - l_{i2})|_{x_i=0} \rangle$$

If multiple such spaces are found, break out of the loop, and go to the next space in the outer loop.

If only one such space is found then update $l_a = l_a - \alpha x_i$ and $l_b = l_b - \beta x_i$ where α, β are coefficients of x_i in l_{i1}, l_{i2} respectively.

Add $\mathbb{V}(x_1 - l_a, x_2 - l_b)$ to $\mathcal{S}_2(g)$

For each $\mathbb{V}(l_1, l_2) \in \mathcal{S}_2(g)$, verify f vanishes on $\mathbb{V}(\Phi^{-1}(l_1), \Phi^{-1}(l_2))$. Output the set thus obtained, as $\mathcal{S}_2(f)$.

5.2 Computing $\mathcal{S}_3(f)$

Recall that in Lemma 4.4, we have defined $\mathcal{S}_3(f)$ as

$$\mathcal{S}_3(f) = \{W \mid W \text{ is a codimension 3 subspace of } \mathbb{F}^n, f \text{ vanishes over } W \text{ and } W \not\subset W' \text{ for any } W' \in \mathcal{S}_1(f) \cup \mathcal{S}_2(f)\}$$

We will first discuss computing $\mathcal{S}_3(f)$ when f depends only on a constant number of variables t such that $t \geq c_3$. Then, solve the general case computation of $\mathcal{S}_3$ spaces by solving it on multiple instances of constant variate cases and gluing them together.

5.2.1 Computing $\mathcal{S}_3(f)$ for constant variate polynomials

Lemma 5.6. *Let $\mathbb{F}$ be a field that is $\mathbb{R}$ or $\mathbb{C}$. Let f be a t-variate, degree d polynomial in $\mathbb{F}[x_1, \ldots, x_t]$ that is computed by a $\Sigma\Pi\Sigma(3)$ circuit such that $\text{rank}(\text{sim}(C)) = t$ and there doesn't exist a linear form $l \in \mathbb{F}[x_1, \ldots, x_t]$ with $\text{rank}(\text{sim}(C \mod l)) < c_2$ (where c_2 is as in Lemma 4.3) and $C \mod l \neq 0$, then there exists a randomized algorithm (Algorithm 3) computes $\mathcal{S}_3(f)$ in time $\text{poly}(d^{O(t^t)})$ if $t \geq c_3$, where c_3 as in Lemma 4.4.*

Proof. This is mostly similar to Lemma 5.3.We try to learn all codimension 3 spaces on which the polynomial vanishes, while excluding those spaces contained within the codimension 1 and 2 spaces (that we can compute using the previous algorithms.

Let Φ be a random linear isomorphism on $\mathbb{F}[x_1, \ldots, x_t]$ such that $\forall i \in [t], \Phi(x_i) = \sum_{j=1}^{t} \alpha_{ij} x_j$ where α_{ij} are sampled randomly from $[d^t]$. We first observe that if f vanishes over a codimension 2 space $\mathbb{V}(l_1, l_2, l_3)$, then after a random linear isomorphism Φ on the variables, $g = \Phi(f) = f(\Phi(x))$ will vanish over a space $\mathbb{V}(\Phi(l_1), \Phi(l_2), \Phi(l_3))$ and moreover this space can be represented in the form $\mathbb{V}(x_1 - l_a, x_2 - l_b, x_3 - l_c)$ for linear forms $l_a, l_b, l_c \in \mathbb{F}[x_4, \ldots, x_n]$.

Let $l_a = a_4x_4 + \ldots + a_tx_t$,$l_b = b_4x_4 + \ldots + b_tx_t$ and $l_c = c_4x_4 + \ldots + c_tx_t$. The basic idea in the constant variate case is that we can substitute $x_1 = a_4x_4 + \ldots + a_tx_t = l_a$, $x_2 = b_4x_4 + \ldots + b_tx_t = l_b$ and $x_3 = c_4x_4 + \ldots + c_tx_t = l_c$ into the monomial representation of the polynomial, and obtain the polynomial $g \mod \langle x_1 - l_a, x_2 - l_b, x_3 - l_c \rangle$. Since we are interested in the case when f vanishes over codimension 3 spaces, we equate coefficients of monomials over the variables $x_4, \ldots, x_t$ to 0 to get a system of equations over $3t - 9$ variables. This system of polynomial equations might have infinitely many solutions unless we discard all those codimension 3 spaces that are contained in $\mathcal{S}_1$ and $\mathcal{S}_2$ spaces. We also know from Lemma 4.4 that this suffices as $t \geq c_3$ and there is no linear form l such that $\text{rank}(\text{sim}(C \mod l)) < c_2$ and $C \mod l \neq 0$. The challenge remains to remove $\mathcal{S}_1(f)$ and $\mathcal{S}_2(f)$. To do that, we add two additional polynomial equations to the system of polynomial equations, that ensure for any $\mathbb{V}(l) \in \mathcal{S}_1(g)$, $\dim(\text{span}(l, x_1 - l_a, x_2 - l_b, x_3 - l_c)) = 4$ and $\mathbb{V}(l, l') \in \mathcal{S}_2(g)$, $\dim(\text{span}(l, l', x_1 - l_a, x_2 - l_b, x_3 - l_c)) \geq 4$. Finally, having computed the spaces $\mathbb{V}(x_1 - l_a, x_2 - l_b, x_3 - l_c)$ on which g vanishes, we simply apply Φ^{-1} to get $\mathbb{V}(l_1, l_2, l_3)$.

We now give a more detailed analysis.

We first observe that in Step 1 of Algorithm 3, the random linear forms $l'_1, \ldots, l'_t$ will be independent with high probability (as otherwise it will correspond to a certain determinant evaluating to 0, which happens with probability at most $d^{-(t-1)}$ due to Lemma 3.2).

Thus will high probability Φ is a random isomorphism, and we obtain the polynomial $g = \Phi(f)$ which is also computable by a $\Sigma\Pi\Sigma(3)$ circuit over t variables, and the simple part of the circuit has rank t. From now onwards let us assume that Φ is an isomorphism.

g vanishes on spaces of the form $\mathbb{V}(x_1 - l_a, x_2 - l_b, x_3 - l_c)$ As Φ is an isomorphism, f vanishes on $\mathbb{V}(l_1, l_2, l_3)$ if and only if g vanishes on $\mathbb{V}(\Phi(l_1), \Phi(l_2), \Phi(l_3)$. We will first observe that with high probability, for any space $\mathbb{V}(\Phi(l_1), \Phi(l_2), \Phi(l_3)) \in \mathcal{S}_2(g)$, there are linear forms $l_a, l_b, l_c \in \mathbb{F}[x_4, \ldots, x_n]$ such that $\mathbb{V}(x_1 - l_a, x_2 - l_b, x_3 - l_c) = \mathbb{V}(\Phi(l_1), \Phi(l_2), \Phi(l_3))$. The reason is the following: Let $l_1 = u_1x_1 + \ldots + u_tx_t$, $l_2 = v_1x_1 + \ldots + v_tx_t$, and $l_3 = w_1x_1 + \ldots + w_tx_t$. As $\mathbb{V}(l_1, l_2, l_3)$ is a codimension 3 space, hence $\dim(\text{span}(l_1, l_2, l_3)) = 3$. After applying the isomorphism Φ, they remain independent with high probability and coefficients of x_i in $\Phi(l_1)$ as $\sum_{j=1}^{n} \alpha_{i,j}u_jx_j$ and similarly for $\Phi(l_2) = \sum_{j=1}^{n} \alpha_{i,j}v_jx_j, \Phi(l_3) = \sum_{j=1}^{n} \alpha_{i,j}w_jx_j$. As they were independent, the determinant of the 3×3 matrix formed by the coefficients of x_1, x_2, x_3 from $\Phi(l_1), \Phi(l_2), \Phi(l_3)$ will be a non-zero polynomial in $\alpha_{1,1}, \ldots, \alpha_{3,n}$ and will vanish with low probability due to Lemma 3.2. This means that for space $\mathbb{V}(l_1, l_2, l_3) \in \mathcal{S}_3(g)$ there is a space $\mathbb{V}(x_1 - l_a, x_2 - l_b, x_3 - l_c) \in \mathcal{S}_3(g)$ where $l_a, l_b, l_c \in \mathbb{F}[x_4, \ldots, x_n]$.

Setting up a system of equations Observe that we can use interpolation to get monomial access to g in time $\text{poly}(d^t)$ using Lemma 3.6.

We set $l_a := a_4x_4 + \ldots + a_tx_t$, $l_b := b_4x_4 + \ldots + b_tx_t$ and $l_c := c_4x_4 + \ldots + c_tx_t$ for variables $a_4, \ldots, a_t, b_4, \ldots, b_t, c_4, \ldots, c_t$. Substituting $x_1 = l_a, x_2 = l_b, x_3 = l_c$ into the monomial form, obtaining a system of $d^{O(t)}$ equations of degree at most d in $3(t-3)$ variables by equating the coefficients of monomials in the variables $x_4, \ldots, x_t$ to 0. Solutions to this would correspond to codimension 3 spaces that g vanishes on.

To remove the codimension 3 spaces that are contained in $\mathcal{S}_1(f)$ and $\mathcal{S}_2(f)$, we first compute the set of all linear factors, $\mathcal{S}_1(g)$ by using blackbox factoring algorithm Lemma 3.7, and obtain the monomial access to all of them by interpolating them in $\text{poly}(d, t)$-time. We also use Lemma 5.1(Algorithm 1) to get $\mathcal{S}_2(g)$ in $d^{O(t^t)}$ time.

Then, we need to ensure that the solution to our system of equations $x_1 - l_a, x_2 - l_b, x_3 - l_c$ is such that $\forall l$ such that $\mathbb{V}(l) \in \mathcal{S}_1(g)$, we have that $\dim(\text{span}(x_1 - l_a, x_2 - l_b, x_3 - l_c, l)) = 4$ and $\forall l, l'$ such that $\mathbb{V}(l, l') \in \mathcal{S}_2(g)$, we have $\dim(\text{span}(x_1 - l_a, x_2 - l_b, x_3 - l_c, l, l')) \geq 4$. This is the same as saying that the $t \times 4$ matrix A_l with $x_1 - l_a, x_2 - l_b, x_3 - l_c, l$ as rows and $t \times 5$ matrix $A_{l,l'}$ with $x_1 - l_a, x_2 - l_b, x_3 - l_c, l, l'$ as rows have rank at least 4. This means at least one of the 4×4 minors of A_l is full rank, with a non-zero determinant. Let the number of such minors be $k_1 := \binom{t}{4}$. Also, at least one of the 4×4 minors of $A_{l,l'}$ is full rank, with a non-zero determinant. Let the number of such minors be $k_2 = 5 \cdot \binom{t}{4}$. To handle these, we introduce new variables $y_1, \ldots, y_{k_2+k_1}$, and for each relevant l we consider the inequalities $sum_l = \sum_{j=1}^{k_1} y_jM_j \neq 0$, where M_j are the determinants of the 4×4 minors of A_l. For each relevant l, l', we also consider $sum_{l,l} = \sum_{j=1}^{k_2} y_{j+k_1}M'_j \neq 0$ where M_j are the determinants of the 4×4 minors of $A_{l,l'}$. The inequality has solutions if and only if there exists a solution for which at least one of the M_j, M'_j is non-zero. So now, we have from Lemma 4.3 $|\mathcal{S}_1(g)| + |\mathcal{S}_2(g)|$ (which is $\leq d + d^7$) inequalities in our system along with the previous equations. We note that we can use the same new variables $y_1, \ldots, y_{k_1+k_2}$ in all of the inequalities. Observe that the set of inequalities $\forall\ \mathbb{V}(l) \in \mathcal{S}_1(g)\ sum_l \neq 0$ and $\forall\ \mathbb{V}(l, l') \in \mathcal{S}_2(g)\ sum_{l,l'} \neq 0$ is the same as having a single inequality $(\prod_{\mathbb{V}(l)\in\mathcal{S}_1(g)} sum_l) \cdot (\prod_{\mathbb{V}(l,l')\in\mathcal{S}_2(g)} sum_{l,l'}) \neq 0$, which is same as requiring that $(\prod_{\mathbb{V}(l)\in\mathcal{S}_1(g)} sum_l) \cdot (\prod_{\mathbb{V}(l,l')\in\mathcal{S}_2(g)} sum_{l,l'}) \cdot z = 1$ has a solution for a new variable z.

Thus, we can handle the condition of the solution not lying in any $\mathcal{S}_1(g)$ and $\mathcal{S}_2(g)$ space, by simply adding one extra equation of degree $5|\mathcal{S}_1(g)| + 5|\mathcal{S}_2(g)| + 1 = O(d^7)$ and $k_1 + k_2 + 1 = \text{poly}(t)$ variables to the system of equations we had earlier.

Running Time Analysis The sampling of the random $\alpha_{i,j}$ can be done in randomized $\text{poly}(t, \log d)$ time. From Lemma 3.7, we can get black-box access to the factors in time randomized $\text{poly}(t, d)$. Computing $\mathcal{S}_2(g)$ also takes $\text{poly}(d^{O(t^t)})$ time from Lemma 5.3. We can do interpolation and get monomial access to g in time $d^{O(t)}$ using Lemma 3.6. As $|\mathcal{S}_1(g)| \leq d$ and $|\mathcal{S}_2(g)| \leq O(d^8)$ from Lemma 4.3, the loop on line 8 runs $O(d)$ times and loop on line 11 runs in $O(d^8)$ time, both doing $\text{poly}(t)$ computation. Finally, the system of equations has $d^{O(t)}$ equations with degree $O(d)$ in $\text{poly}(t)$ variables. Therefore, we can find l_a, l_b, l_c by solving the system of equations in time $\text{poly}(d^{O(t^t)})$ using Theorem 3.8. Thus, the entire algorithm works in $\text{poly}(d^{O(t^t)})$ time. □

Algorithm 3 Computing Vanishing $\mathcal{S}_3(f)$ for constant variate polynomials

Input: Blackbox access to circuit C of form $\Sigma\Pi\Sigma(3)$ computing polynomial $f \in \mathbb{F}[x_1, \ldots, x_t]$, $t \geq c_3$ and $\text{rank}(\text{sim}(C)) = t$

1: **function** $\mathcal{S}_3(f)$
2: Sample t^2 random values $\alpha_{ij}; i, j \in [t]$ uniformly from $\{1, \ldots, d^t\}$, and use them to define t linear forms $l'_i = \sum_{j=1}^{t} \alpha_{ij} x_j$. Check if they are independent, otherwise repeat. Define isomorphism Φ such that for all $i \in [t]$, $\Phi(x_i) := l'_i$. Let $g = \Phi(f) = f(\Phi(x))$.
3: Using randomized black-box factoring from Lemma 3.7 and get access to the linear factors of g
4: Interpolate g to get monomial access to it and interpolate the linear factors to obtain $\mathcal{S}_1(g) := \{\mathbb{V}(l) : l | g\}$
5: Use Algorithm 1 to get $\mathcal{S}_2(g)$
6: Substitute $x_1 = a_4 x_4 + \ldots + a_{t-1} x_{t-1} + a_i x_i$ for linear form $x_1 - l_a$, $x_2 = b_4 x_4 + \ldots + b_{t-1} x_{t-1} + b_i x_i$ for linear form $x_2 - l_b$ and $x_3 = c_4 x_4 + \ldots + c_{t-1} x_{t-1} + c_i x_i$ for linear form $x_3 - l_c$ in g and obtain equations in $a_4, \ldots, a_t, b_4, \ldots, b_t, c_4, \ldots, c_t$ by equating the coefficients of monomials in the variables $x_4, \ldots, x_{t-1}, x_i$ to 0.
7: Let $k_1 = \binom{t}{4}$ and $k_2 = 5\binom{t}{5}$. Introduce new variables $z, y_1, \ldots, y_{k_1+k_2}$
8: **for** $\mathbb{V}(l) \leftarrow \mathcal{S}_1(g)$ **do**
9: Consider the $4 \times t$ matrix A_l formed by l and $x_1 - l_a, x_2 - l_b, x_3 - l_c$.
10: Compute the determinant of each 4×4 minor M_j of A_l. Compute and store $sum_l := \sum_{j=1}^{k_1} y_j M_j$
11: **for** $\mathbb{V}(l, l') \leftarrow \mathcal{S}_2(g)$ **do**
12: Consider the $4 \times t$ matrix $A_{l,l'}$ formed by l, l' and $x_1 - l_1, x_2 - l_2, x_3 - l_3$.
13: Compute each 4×4 minor M_j of $A_{l,l'}$. Compute and store $sum_{l,l'} := \sum_{j=k_1}^{k_1+k_2} y_j M_j$
14: Add Equation $(\prod_{\mathbb{V}(l) \in \mathcal{S}_1(g)} sum_l) \cdot (\prod_{\mathbb{V}(l_1,l_2) \in \mathcal{S}_2(g)} sum_{l_1,l_2}) \cdot z = 1$ to the system of equations in Step 6.
15: Solve the system of equations in $\mathbf{a}, \mathbf{b}, \mathbf{c}$ to obtain a set of $(x_1 - l_1, x_2 - l_2, x_3 - l_3)$.
16: Solve the system of equations in $a_4, \ldots, a_t, b_4, \ldots, b_t, c_4, \ldots, c_t, y_1, \ldots, y_{k_1+k_2}, z$ using Theorem 3.8 to obtain a set of $(x_1 - l_a, x_2 - l_b, x_3 - l_c)$.
17: Verify for each $(x_1 - l_a, x_2 - l_b, x_3 - l_c)$ if f vanishes on $\mathbb{V}(\Phi^{-1}(x_1 - l_a), \Phi^{-1}(x_2 - l_b), \Phi^{-1}(x_3 - l_c)) = \mathbb{V}(l_1, l_2, l_3)$ using Lemma 3.2, then add $\mathbb{V}(l_1, l_2, l_3)$ to $\mathcal{S}_3(f)$
18: **Output** $\mathcal{S}_3(f)$

5.2.2 Computing $\mathcal{S}_3(f)$ general case

We will now discuss how we can use the solution for the constant variate case to compute $\mathcal{S}_3(f)$ in the general case.

We will start by using a random linear isomorphism Φ on f such that $\Phi(x_i) = \sum_{j=1}^{n} \alpha_{ij} x_j$, where α_{ij} are chosen randomly from $[d^n]$, and define $g = \Phi(f) = f(\Phi(x))$. Let $t = c_3 + 1$. We will then consider the t variate polynomials g_i (for $i \geq t$) which are obtained from g by setting all variables x_j for $j > t-1$ to zero, except x_i.

Thus

$$g_i = g|_{x_t=\ldots=x_{i-1}=x_{i+1}=\ldots=x_n=0}$$

We will then find $\mathcal{S}_3(g_i)$ spaces using the constant variate algorithm and then show how to glue the learned spaces to get $\mathcal{S}_3(g)$ and then $\mathcal{S}_3(f)$.

But, we still need to argue that the removal of $\mathcal{S}_2$ spaces(similar to the removal of $\mathcal{S}_1$ spaces in part 2 of Lemma 5.4) after the number of variables is reduced corresponds exactly to the $\mathcal{S}_2$ spaces of g. From part 1 of Lemma 5.4, we have that all g_i can be computed by $\Sigma\Pi\Sigma(3)$ circuits C_i of rank t. Also, to be able to find the $\mathcal{S}_3$ spaces in the constant variate case, we need to argue that there is no linear form l such that $\text{rank}(\text{sim}(C_i \mod l)) < c_2$.

We will first argue that when we project down to t variables ($t = c_3 + 1$), there is no linear form l such that $\text{rank}(\text{sim}(C_i \mod l)) < c_2$ and $C_i \mod l_i \neq 0$.

Lemma 5.7. *If f is computed by a $\Sigma\Pi\Sigma(3)$ circuit C such that $\text{rank}(\text{sim}(C)) \geq c_3$ and there doesn't exist a linear form $l \in \mathbb{F}[x_1, \ldots, x_n]$ with $\text{rank}(\text{sim}(C \mod l)) < c_2$ and $C \mod l \neq 0$, then with probability $1 - o(1)$, for each $i \in \{t, \ldots, n\}$, g_i is also computed by a circuit C_i such that there doesn't exist a linear form $l_i \in \mathbb{F}[x_1, \ldots, x_{t-1}, x_i]$ with $\text{rank}(\text{sim}(C_i \mod l_i)) < c_2$ and $C_i \mod l_i \neq 0$.*

Proof. Observe that g_i is obtained by applying a random linear isomorphism and then setting a bunch of variables to zero. To prove the above lemma, observe that it suffices to prove it in the case where we apply a random linear isomorphism and then just set one variable to zero. If we can show that with a very high probability the resulting circuit will continue to have the desired property in this case, then we can recursively apply this procedure (alternately applying random linear isomorphisms and then setting a variable to zero) to eventually come down to only t variables and continue to show with high probability that still there is no l_i for which when we go mod l_i then the rank of the simple part drops. However alternately applying random linear isomorphisms and then setting a variable to zero is (essentially) equivalent to applying one single random isomorphism and then setting severable variables to zero at once. For simplicity, we will assume that these two distributions are the same for the rest of the argument. It is easy to adapt the argument to the original distribution or to change the definition of g_i to be sampled from the recursively defined distribution.

Let f be computed by a circuit of the form $G \times (T_1 + T_2 + T_3)$ such that $\gcd(T_1, T_2, T_3) = 1$. Note that g is then computed by a circuit of the form $\Phi(G) \times (\Phi(T_1) + \Phi(T_2) + \Phi(T_3))$ where $\gcd(\Phi(T_1), \Phi(T_2), \Phi(T_3)) = 1$. Thus $g \mod x_n$ is computed by a circuit C' of the form $\Phi(C) \mod x_n$. Note that we need to show that with very high probability the circuit C' computing $g \mod x_n$ is such that there is no $l \in \mathbb{F}[x_1, \ldots, x_{n-1}]$ with $\text{rank}(\text{sim}(C' \mod l)) < c_2$ and $C' \mod l \neq 0$.

Assume, this is not the case, i.e. $C' = \Phi(C) \mod x_n$ is such that for a linear form l, $\text{rank}(\text{sim}(C' \mod l)) < c_2$ and $C' \mod l \neq 0$. From the assumption in the lemma, we know $\text{rank}(\text{sim}(\Phi(C) \mod l)) \geq c_2$ as $C' \mod l = \Phi(C) \mod \langle l, x_n \rangle \neq 0$ implies $\Phi(C) \mod l \neq 0$.

Let $\mathcal{P}$ be the set of all 2-dimensional spaces such that for all $\text{span}(l_a, l_b) \in \mathcal{P}$ there exists $l_1 \in T_1, l_2 \in T_2, l_3 \in T_3$ such that $\text{span}(l_a, l_b) = \text{span}(l_1, l_2, l_3)$. Define a set of linear forms S as follows

$$S := \{l : l \in \text{Lin}(T_i) \text{ for some } i \in [3], \text{ or } \text{span}(l) = V_1 \cap V_2, V_1, V_2 \in \mathcal{P}\}$$

We have $|S| = d^6 + 3d$ as $|\text{Lin}(T_i)| \leq d$ for each $i \in [3]$ and the number of spaces in $\mathcal{P}$ is at most d^3.

Claim 5.8. *In the setting of the lemma, If there exists a linear form l such that* $\text{rank}(\text{sim}(C' \mod l)) < c_2$ *and* $C' \mod l \neq 0$*, then with probability* $1 - o(1)$ *there exists a linear form* $l' \in S$ *such that* $\text{span}(l, x_n) = \text{span}(l', x_n)$.

Proof. We know $\text{rank}(\text{sim}(\Phi(C) \mod \langle l, x_n \rangle)) < c_2$ while $\text{rank}(\text{sim}(\Phi(C) \mod l)) \geq c_2$ and $\text{rank}(\text{sim}(\Phi(C))) \geq c_3$. Therefore, the rank decrease from $\Phi(C)$ to $\Phi(C) \mod \langle l, x_n \rangle$ happens in 2 possible ways, either a linear form in one of the gates (say T_i) vanishes mod $\langle l, x_n \rangle$ or there are linear forms $l_1 \in \Phi(T_1), l_2 \in \Phi(T_2), l_3 \in \Phi(T_3)$ that become the same when we consider the circuit mod $\langle l, x_n \rangle$ and move to the gcd.

If it is the first case, and $\Phi(T_i) \mod \langle l, x_n \rangle = 0$, this means there is a linear form $l' \in \text{span}(l, x_n) \cap \text{Lin}(T_i)$. Therefore, l' is a linear form in S such that $\text{span}(l', x_n) = \text{span}(l, x_n)$ unless $x_n \in \text{span}(l')$. In this case, we have $\Phi^{-1}(x_n) \in \text{span}(l_i)$, or $\dim(\text{span}(l_i, \Phi^{-1}(x_n))) = 1$ for some $l_i \in T_i$. Since the coefficients of Φ are chosen randomly from $[d^n]$, the probability that this happens for a fixed l_i is $O(d^{-n})$. As there are at most $3d$ possibilities for l_i, after taking union bound over all l_i, we get this happens with probability $o(1)$. Therefore, with high probability, there is $l' \in S$ such that $\text{span}(l', x_n) = \text{span}(l, x_n)$.

In the second case, we have several set of linear forms $l_1 \in \Phi(T_1), l_2 \in \Phi(T_2), l_3 \in \Phi(T_3)$ that become the same linear form $\tilde{l} \neq 0 \mod \langle l, x_n \rangle$ when we consider the circuit mod $\langle l, x_n \rangle$ and move to the gcd. We break it into 2 cases based on $\dim(\text{span}(l_1, l_2, l_3))$:

- There is a l_1, l_2, l_3 such that $\dim(\text{span}(l_1, l_2, l_3)) = 3$: In this case, $\text{span}(l, x, \tilde{l}) = \text{span}(l_1, l_2, l_3)$ and therefore $x_n \in \text{span}(l_1, l_2, l_3)$ and after Φ^{-1}, we have $\Phi^{-1}(x_n) \in \text{span}(\Phi^{-1}(l_1), \Phi^{-1}(l_2), \Phi^{-1}(l_3))$, where $\Phi^{-1}(l_1), \Phi^{-1}(l_2), \Phi^{-1}(l_3)$ are linear forms in T_1, T_2, T_3 respectively. Again, this happens for fixed l_1, l_2, l_3 with probability $O(d^{-n})$. Taking a union bound over d^3 possibilities of l_1, l_2, l_3, it happens with probability $o(1)$.

- For all l_1, l_2, l_3 moving to gcd $\dim(\text{span}(l_1, l_2, l_3)) = 2$: In this case, we have $\text{span}(l_1, l_2, l_3)$ must intersect $\text{span}(l, x_n)$ in a line. Consider the case where there are 2 sets of linear forms going into the gcd (l_1, l_2, l_3 and l_1', l_2', l_3') when we consider the circuit mod $\langle l, x_n \rangle$, such that they intersect $\text{span}(l, x_n)$ in different lines. Then we have $x_n \in \text{span}(l_1, l_2, l_3, l_1', l_2', l_3')$. Similar to the earlier discussion, this happens with probability $O(d^{-n})$ for fixed $l_1, l_2, l_3, l_1', l_2', l_3'$, and taking union bound over d^6 choices, we get this happens with probability $o(1)$. Therefore, with a high probability, there are at least 2 sets of linear forms that move into the gcd that intersect $\text{span}(l, x_n)$ in the same space, let us say $\text{span}(l_0)$. From definition, we have these sets of linear forms in $\mathcal{P}$ and therefore $l_0 \in S$. Note, we already argued that $x_n \in \text{span}(l_1, l_2, l_3, l_1', l_2', l_3')$ happens with probability $o(1)$. Therefore, again in this case there is a linear form $l' = l_0$ in S such that with high probability, there is $l' \in S$ such that $\text{span}(l', x_n) = \text{span}(l, x_n)$.

□

From the above claim, as $\text{rank}(\text{sim}(C' \mod l)) < c_2$ and $C' \mod l \neq 0$, we have there must be a linear forms $l' \in S$ such that $\text{span}(l, x_n) = \text{span}(l', x_n)$. Let us fix a linear form l_0 in S. From the assumption, we have $\text{rank}(\text{sim}(\Phi(C) \mod l_0) \geq c_2$. If there exists an l such that $\text{rank}(\text{sim}(C' \mod l)) < c_2$ and $C' \mod l \neq 0$, then we also have $\text{rank}(\text{sim}(\Phi(C) \mod \langle l_0, x_n \rangle) < c_2$. In part 1

of Lemma 5.4, we showed that this happens with probability $O(d^{-n})$ and therefore for fixed l_0 this happens with probability $O(d^{-n})$. Taking a union bound for all l_0 in S which has size $O(d^6)$, we get that the probability such a l exists is $o(1)$.

□

The fact that there are no new $\mathcal{S}_1$ spaces was already argued in Part 2 of Lemma 5.4. Also in Part 3 of Lemma 5.4 we showed that the number of $\mathcal{S}_2$ spaces doesn't shrink. Now, we will argue no new $\mathcal{S}_2$ spaces are created for g_i's with high probability.

Lemma 5.9. *If f is computed by a $\Sigma\Pi\Sigma(3)$ circuit C such that $\text{rank}(\text{sim}(C)) \geq c_3$ and there doesn't exist a linear form $l \in \mathbb{F}[x_1, \ldots, x_n]$ with $\text{rank}(\text{sim}(C \mod l)) < c_2$ and $C \mod l \neq 0$, then with probability $1 - o(1)$,*

$$\mathcal{S}_2(g_i) = \{\Phi(\mathcal{S}_2(f))\}|_{x_t=\ldots=x_{i-1}=x_{i+1}=\ldots=x_n=0}$$

Proof. We want to argue that there is no new codimension 2 space over which g_i vanishes, which doesn't correspond to a space in $\mathcal{S}_2(g)$.

Similar to Lemma 5.7, we argue this recursively and therefore only need to show that no new $\mathcal{S}_2$ spaces are added when we consider $g \mod x_n$.

Consider a new space $\mathbb{V}(l_1, l_2) \in \mathcal{S}_2(g \mod x_n)$ such that it doesn't correspond to a space in $\mathcal{S}_2(g)$, i.e. $\mathbb{V}(l_1, l_2) \notin \{\mathcal{S}_2(g)\}|_{x_n=0}$. This means that the space $\mathbb{V}(l_1, l_2, x_n)$ is a codimension 3 space on which g vanishes. So $\mathbb{V}(l_1, l_2, x_n) \in \mathcal{S}_3(g)$ unless there is a space $\mathbb{V}(l_a, l_b) \in \mathcal{S}_2(g)$ such that $\text{span}(l_a, l_b) \subseteq \text{span}(l_1, l_2, x_n)$. But in this case for $g \mod x_n$, $\mathbb{V}(l_1, l_2)$ is not a new space, but just $\mathbb{V}(l_a, l_b) \mod x_n$ unless $x_n \in \text{span}(l_a, l_b)$ and therefore $\mathbb{V}(x_n, l') \in \mathcal{S}_2(g)$ such that $\text{span}(l_a, l_b) = \text{span}(x_n, l')$. Therefore, $\mathbb{V}(l_1, l_2, x_n) \in \mathcal{S}_3(g)$ or $\mathbb{V}(x_n, l') \in \mathcal{S}_2(g)$. Consider $\mathbb{V}(l_1, l_2, x_n) \in \mathcal{S}_3(g)$ first. Applying Φ^{-1}, we get that $\mathbb{V}(\Phi^{-1}(l_1), \Phi^{-1}(l_2), \Phi^{-1}(x_n))$ is in $\mathcal{S}_3(f)$. This means $\Phi^{-1}(x_n)$ must be in $\text{span}(l_1, l_2, l_3)$ for some $\mathbb{V}(l_1, l_2, l_3) \in \mathcal{S}_3(f)$. Since Φ was random, $\Phi^{-1}(x_n)$ must be random as well. We argue that a random linear form l doesn't lie in the kernel of a $\mathcal{S}_3$ space with high probability. From Lemma 4.4, we know as $\text{rank}(\text{sim}(f)) \geq c_3$ and there doesn't exist a linear form l, such that $\text{rank}(\text{sim}(f \mod l)) < c_2$, $|\mathcal{S}_3(f)| \leq O(d^{15})$. And, we pick l randomly, so the probability that the line lies in one of the spaces in $\mathcal{S}_3(f)$ is at most $|\mathcal{S}_3(f)| \cdot \mathbb{P}[l \in \text{span}(l_1, l_2, l_3)]$ for fixed l_1, l_2, l_3 over n variables by union bound. This means the rank of the matrix defined by l, l_1, l_2, l_3 as rows is 3, which happens with probability $O(d^{-n})$ as we pick coefficients of Φ from $[d^n]$.

Thus, the probability that a new space is introduced is $\frac{d^{15}}{d^n} \leq o(1)$. The same argument holds for $\Phi^{-1}(x_n)$ lying in the kernels of a $\mathcal{S}_2(f)$ space. □

Now, provide the proof of Lemma 5.2, which we restate here for clarity.

Lemma 5.2 restated. *Let $\mathbb{F}$ be a field that is $\mathbb{R}$ or $\mathbb{C}$. Let f be a n-variate, degree d polynomial in $\mathbb{F}[x_1, \ldots, x_n]$ that is computed by a $\Sigma\Pi\Sigma(3)$ circuit C such that $\text{rank}(\text{sim}(C)) \geq c_3$ and there doesn't exist a linear form $l \in \mathbb{F}[x_1, \ldots, x_n]$ with $\text{rank}(\text{sim}(C \mod l)) < c_2$, where c_2 is as in Lemma 4.3 and c_3 is as in Lemma 4.4. Then, there exists an algorithm (Algorithm 4) that outputs $\mathcal{S}_3(f)$, in $\text{poly}(n, d)$-time with probability $1 - o(1)$.*

Proof of Lemma 5.2. The idea is the same as Lemma 5.1, but we need to argue some more technicalities using the lemmas mentioned above. We first apply a random linear isomorphism and obtain $g = \Phi(f)$. For $t = c_3 + 1$, we obtain $n - t$ polynomials $g_i \in \mathbb{F}[x_1, \ldots, x_{t-1}, x_i]$ by setting all except $x_1, \ldots, x_{t-1}, x_i$ to 0. Then we solve the constant variate cases, using Algorithm 3 discussed

above to recover $\mathcal{S}_3(g_i)$. We will then recover $\mathcal{S}_3(g)$ by gluing together spaces from $\mathcal{S}_3(g_i)$ (across different choices of i) when the spaces are consistent when restricted to $x_1, \ldots, x_{t-1}$.

By part 1 of Lemma 5.4, the random invertible linear isomorphism ensures that when we set some of the variables to 0, the rank of the linear forms present in the circuits computing the g_i's remains high (equal to t) with high probability. Also, from part 2 or Lemma 5.4, we know the set $\mathcal{S}_1(g_i)$ contains exactly the linear factors of g projected down to the t variables in g_i.

Similar to part 3 of Lemma 5.4, we have for every $\mathbb{V}(x_1-l_1, x_2-l_2, x_3-l_3) \in \mathcal{S}_3(g)$, there is $\mathbb{V}(x_1-l_{1i}, x_2-l_{2i}, x_3-l_{3i}) \in \mathcal{S}_3(g_i)$, where $l_{ji} = l_j|_{x_t=0,\ldots,x_{i-1}=0,x_{i+1}=0,\ldots,x_n=0}$. As we already saw in proof of Lemma 5.6, for any $\mathbb{V}(l_1, l_2, l_3) \in \mathcal{S}_3(g)$, it can be represented in the form $\mathbb{V}(x_1-l'_1, x_2-l'_2, x_3-l'_3)$ where $l'_1, l'_2, l'_3 \in \mathbb{F}[x_4, \ldots, x_n]$. It is fairly straightforward to see that if g vanishes on $\mathbb{V}(x_1-l'_1, x_2-l'_2, x_3-l'_3)$ then g_i vanishes on $\mathbb{V}(x_1-l'_{1i}, x_2-l'_{2i})$ where $l'_{ji} = l'_j|_{x_t=0,\ldots,x_{i-1}=0,x_{i+1}=0,\ldots,x_n=0}$.

Also from part 2 of Lemma 5.4, we have that $\mathcal{S}_1(g_i) = \{\mathbb{V}(l|_{x_t=\ldots=x_{i-1}=x_{i+1}=\ldots=x_n=0}) : \mathbb{V}(l) \in \mathcal{S}_1(g)\}$ and from Lemma 5.9, we have $\mathcal{S}_2(g_i) = \{\mathcal{S}_2(g)\}|_{x_t=\ldots=x_{i-1}=x_{i+1}=\ldots=x_n=0}$, which means if $\mathbb{V}(x_1-l'_1, x_2-l'_2, x_3-l'_3)$ weren't contained in any $\mathcal{S}_1(g)$ or $\mathcal{S}_2(g)$ space, then $\mathbb{V}(x_1-l'_{1i}, x_2-l'_{2i})$ is also not contained in any $\mathcal{S}_1(g_i)$ or $\mathcal{S}_2(g'_i)$ space. Both of the above combine to give us that $\mathbb{V}(x_1-l'_{1i}, x_2-l'_{2i}, x_3-l'_{3i}) \in \mathcal{S}_2(g_i)$.

From Lemma 5.7, we have that with $1-o(1)$ probability none of the g_i's are such that there is a linear form l with $\text{rank}(\text{sim}(g_i \mod l)) < c_2$, and therefore from Lemma 4.4 the output list $\mathcal{S}_3(g_i)$ is also $\text{poly}(d)$ in size. We can find $\mathcal{S}_3(g_i)$ in $\text{poly}(d)$ time as described in Lemma 5.6. By Lemma 5.9, we also have that the codimension 2 spaces removed in Algorithm 3 correspond to $\mathcal{S}_2(g)$.

To obtain $\mathcal{S}_3(g)$ (wlog of the form $\mathbb{V}(x_1-l_1, x_2-l_2, x_3-l_3)$), we will show how to glue these spaces learned in the constant variate case. To do this, we will look at spaces in $\mathcal{S}_3(g_i)$ and $\mathcal{S}_3(g_j)$ (for $i \neq j$) and "glue" them if they are consistent in the first $t-1$ variables. For this to be efficient, it will be very useful to have the property that distinct spaces in $\mathcal{S}_3(g_i)$ are distinct when restricted to the first $t-1$ coordinates. This is probably already true with high probability due to the randomness of Φ. However to make the argument simpler to analyze, we consider and apply another random linear isomorphism Ψ (we will apply these to the spaces in $\mathcal{S}_3(g_i)$ for each i) defined as follows: $\forall i < t$, $\Psi(x_i) = x_i$ and $\forall i \in [t, n]$, $\Psi(x_i) = x_i + \beta_{i,4}x_4 + \ldots + \beta_{i,t-1}x_{t-1}$ where $\beta_{i,j}$ are sampled independently and uniformly from $[d^n]$. The goal of introducing the map Ψ is to ensure that distinct spaces in $\mathcal{S}_3(g_i)$ are distinct when restricted to the first $t-1$ coordinates, and we prove this formally in the claim below.

Claim 5.10. *For all $i \in \{t, \ldots, n\}$, let $\mathbb{V}(l_1, l_2, l_3)$ and $\mathbb{V}(l'_1, l'_2, l'_3)$ be distinct spaces in $\mathcal{S}_3(g_i)$ such that $l_1, l_2, l_3, l'_1, l'_2, l'_3$ only depend on $x_4, \ldots, x_{t-1}, x_i$. Then $\mathbb{V}(\Psi(x_1-l_1)_{x_i=0}, \Psi(x_2-l_2)_{x_i=0}, \Psi(x_3-l_3)_{x_i=0}) \neq \mathbb{V}(\Psi(x_1-l'_1)_{x_i=0}, \Psi(x_2-l'_2)_{x_i=0}, \Psi(x_3-l'_3)_{x_i=0})$. In particular*

$$\langle \Psi(l_1)|_{x_i=0}, \Psi(l_2)|_{x_i=0}, \Psi(l_3)|_{x_i=0} \rangle \neq \langle \Psi(l'_1)|_{x_i=0}, \Psi(l'_2)|_{x_i=0}, \Psi(l'_3)|_{x_i=0} \rangle$$

Proof. The proof of this claim is very similar to Claim 5.5and Lemma 5.4 of [Sin22]. Consider 2 distinct elements $\mathbb{V}(x_1-l_1, x_2-l_2, x_3-l_3)$ and $\mathbb{V}(x_1-l'_1, x_2-l'_2, x_3-l'_3)$ in $\mathcal{S}_3(g_i)$, with $l_1, l'_1, l_2, l'_2, l_3, l'_3 \in \mathbb{F}[x_4, \ldots, x_{t-1}, x_i]$. Let $l_1 = a_4x_4 + \ldots + a_{t-1}x_{t-1} + a_ix_i$, $l_2 = b_4x_4 + \ldots + b_{t-1}x_{t-1} + b_ix_i$, $l_3 = c_4x_4 + \ldots + c_{t-1}x_{t-1} + c_ix_i$, $l'_1 = a'_4x_4 + \ldots + a'_{t-1}x_{t-1} + a'_ix_i$, $l_2 = b'_4x_4 + \ldots + b'_{t-1}x_{t-1} + b'_ix_i$, and $l'_3 = c'_4x_4 + \ldots + c'_{t-1}x_{t-1} + c'_ix_i$. Therefore, we have

$$
\begin{aligned}
l_1 &= (a_4 + \beta_{i,4}a_i)x_4 + \ldots + (a_{t-1} + \beta_{i,t-1}a_i)x_{t-1} + a_i x_i \\
l_2 &= (b_4 + \beta_{i,4}b_i)x_4 + \ldots + (b_{t-1} + \beta_{i,t-1}b_i)x_{t-1} + b_i x_i \\
l_3 &= (c_4 + \beta_{i,4}c_i)x_4 + \ldots + (c_{t-1} + \beta_{i,t-1}c_i)x_{t-1} + c_i x_i \\
l'_1 &= (a'_4 + \beta_{i,4}a'_i)x_4 + \ldots + (a'_{t-1} + \beta_{i,t-1}a'_i)x_{t-1} + a'_i x_i \\
l'_2 &= (b'_4 + \beta_{i,4}b'_i)x_4 + \ldots + (b'_{t-1} + \beta_{i,t-1}b'_i)x_{t-1} + b'_i x_i \\
l'_3 &= (c'_4 + \beta_{i,4}c'_i)x_4 + \ldots + (c'_{t-1} + \beta_{i,t-1}c'_i)x_{t-1} + c'_i x_i
\end{aligned}
$$

Now, if $\langle \Psi(x_1 - l_1), \Psi(x_2 - l_2), \Psi(x_3 - l_3)\rangle \mid_{x_i=0} = \langle \Psi(x_1 - l'_1), \Psi(x_2 - l'_2), \Psi(x_3 - l'_3)\rangle \mid_{x_i=0}$, we have a system of linear equations in $\beta_{i,4}, \ldots, \beta_{i,t-1}$ given by $\forall j \in \{4, \ldots, t-1\}$ $\beta_{i,j}(a_i - a'_i) = (a_j - a'_j)$, $\beta_{i,j}(b_i - b'_i) = (b_j - b'_j)$, and $\beta_{i,j}(c_i - c'_i) = (c_j - c'_j)$. Since $\langle x_1 - l_1, x_2 - l_2, x_3 - l_3\rangle$ and $\langle x_1 - l'_1, x_2 - l'_2, x_3 - l'_3\rangle$ are distinct, there is some choice of j for which at least one of $a_j \neq a'_j$, $b_j \neq b'_j$ or $c_j \neq c'_j$ must hold. This gives a nonzero linear equation in $\beta_{i,j}$ (when viewed as formal variables) which must become zero for the specific choice of sampled values. By using Lemma 3.2,the probability this can happen is $\frac{1}{d^n}$ as we choose the $\beta_{i,j}$ from $[d^n]$. From Lemma 4.4 we know the $\mathcal{S}_3(g_i) = d^{O(1)}$, and taking the union of all pairs from $\mathcal{S}_3(g_i)$ gives us that with probability $1 - o(1)$ no two spaces in any of the $\mathcal{S}_3(g_i)$ are equal after applying Ψ and setting x_i to zero. □

As described in Algorithm 4, for each $\mathbb{V}(x_1 - l_{t1}, x_2 - l_{t2}, x_3 - l_{t3}) \in \mathcal{S}_2(g_t)$ we "glue" or combine it with a corresponding $\mathbb{V}(x_1 - l_{i1}, x_2 - l_{i2}, x_3 - l_{i3}) \in \mathcal{S}_2(g_i)$ if they are consistent in first t variables, i.e. $\langle \Psi(l_{i1})|_{x_i=0}, \Psi(l_{i2})|_{x_i=0}, \Psi(l_{i3})|_{x_i=0}\rangle = \langle \Psi(l_{t1})|_{x_t=0}, \Psi(l_{t2})|_{x_t=0}, \Psi(l_{t3})|_{x_t=0}\rangle$. We each fixed space in $\mathcal{S}_3(g_t)$ with high probability there is a unique space in $\mathcal{S}_3(g_i)$ where this happens by the above claim. Note that every space in $\mathcal{S}_3(g)$ corresponds to some unique space in $\mathcal{S}_3(g_t)$ (since the spaces in $\mathcal{S}_3(g_t)$ are distinct restricted to first $t-1$ coordinates). To recover the spaces of $\mathcal{S}_3(g)$ with information for all coordinates, for each i, we use the information present in the glued space in $\mathcal{S}_3(g_i)$ to recover the information for the coordinate corresponding to x_i. Thus we obtain all spaces $\mathbb{V}(l_1, l_2, l_3)$ on which g vanishes and use Φ^{-1} to obtain $\mathcal{S}_3(f)$. □

Algorithm 4 Computing Vanishing Codimension 3 Subspaces

Input: Blackbox access to circuit C of form $\Sigma\Pi\Sigma(3)$ computing polynomial $f \in \mathbb{F}[x_1, \ldots, x_n]$

function $\mathcal{S}_3(f)$

Sample n^2 random values $\alpha_{ij}; i, j \in [n]$ uniformly from $\{1, \ldots, d^n\}$, and use them to define n linear forms $l'_i = \sum_{j=1}^{n} \alpha_{ij} x_j$. Check if they are independent, otherwise repeat. Define isomorphism Φ such that for all $i \in [n]$, $\Phi(x_i) := l'_i$. Let $g = \Phi(f) = f(\Phi(x))$.

Set $t = c_3 + 1$. For $i \in [t, n]$, Obtain $g_i = g_{x_t=0,\ldots,x_{i-1}=0,x_{i-1}=0,\ldots,x_n=0}$

For each $g_i, i \in [t, n]$, Compute $\mathcal{S}_3(g_i)$ using Algorithm 3.

We now describe how to glue these spaces across g_i.

Consider an isomorphism Ψ obtained as follows. Sample $(n-t+1) \times (t-4)$ random values $\beta_{i,j}$ where $i \in [t, n], j \in [4, t-1]$ uniformly from $\{1, \ldots, d^n\}$. For all $i \in [1, t-1]$, let $\Psi(x_i) = x_i$ and for all $i \in [t, n]$, let $\Psi(x_i) = x_i + \beta_{i,4}x_4 + \ldots + \beta_{i,t-1}x_{t-1}$.

for $\mathbb{V}(x_1 - l_{t1}, x_2 - l_{t2}, x_3 - l_{t3}) \in \mathcal{S}_3(g_t)$ **do**

$l_a := l_{t1}, l_b := l_{t2}, l_c := l_{t3}$

for $i \in \{t+1, \ldots, n\}$ **do**

Search for $\mathbb{V}(x_1 - l_{i1}, x_2 - l_{i2}, x_3 - l_{i3})$ such that

$$\langle \Psi(x_1 - l_{t1})|_{x_t=0}, \Psi(x_2 - l_{t2})|_{x_t=0}, \Psi(x_3 - l_{t3})|_{x_t=0} \rangle = \langle \Psi(x_1 - l_{i1})|_{x_i=0}, \Psi(x_2 - l_{i2})|_{x_i=0}, \Psi(x_3 - l_{i3})|_{x_i=0} \rangle$$

If multiple such spaces are found, break out of the loop, and go to the next space in the outer loop.

If only one such space is found then update $l_a = l_a - \alpha x_i$, $l_b = l_b - \beta x_i$ and $l_c = l_c - \gamma x_i$ where α, β, γ are coefficients of x_i in l_{i1}, l_{i2}, l_{i3} respectively.

Add $\mathbb{V}(x_1 - l_a, x_2 - l_b, x_3 - l_c)$ to $\mathcal{S}_3(g)$

For each $\mathbb{V}(l_1, l_2, l_3) \in \mathcal{S}_3(g)$,Verify f vanishes on $\mathbb{V}(\Phi^{-1}(l_1), \Phi^{-1}(l_2), \Phi^{-1}(l_3)$. Output the set thus obtained, as $\mathcal{S}_3(f)$.

6 Using $\mathcal{S}_2(f)$ and $\mathcal{S}_3(f)$ to get some linear forms appearing in C

In this section, we show how to use the set of spaces in $\mathcal{S}_2, \mathcal{S}_3$ that were learnt in the previous section to actually learn a few linear forms that appear in the circuit. We do this by looking at intersections of the kernels of these spaces. The bulk of the section will be devoted to showing that these intersections enable us to learn many useful linear forms, and this is formalized in Theorem 6.1.

We classify the linear forms, which we can learn from $\mathcal{S}_2$ and $\mathcal{S}_3$, into the following 3 sets.

Consider 2 distinct spaces $\mathbb{V}(l_1, l_2), \mathbb{V}(l'_1, l'_2) \in \mathcal{S}_2(f)$. If the intersection of the two spaces $\text{span}(l_1, l_2)$ and $\text{span}(l'_1, l'_2)$ has dimension 1, then we add the linear form that corresponds to this dimension 1 space(we do this in a some canonical way, with leading coefficient 1, say) to the set $\mathcal{L}_2$.

Similarly, we consider 3 distinct spaces $\mathbb{V}(l_1, l_2, l_3), \mathbb{V}(l'_1, l'_2, l'_3), \mathbb{V}(l''_1, l''_2, l''_3) \in \mathcal{S}_3(f)$, and the intersection of the three spaces $\text{span}(l_1, l_2, l_3), \text{span}(l'_1, l'_2, l'_3), \text{span}(l''_1, l''_2, l''_3)$. If it has dimension 1, we add the linear form corresponding to the intersection to $\mathcal{L}_3$.

In the general case, where we can find $\mathcal{S}_2, \mathcal{S}_3$(satisfying conditions of Lemma 5.1 and Lemma 5.2), we will make our set of candidate linear forms using the union of the two sets

$$\mathcal{L}_{cand} = \mathcal{L}_2 \cup \mathcal{L}_3$$

From Lemma 5.1 and Lemma 5.2, we see that we can compute $\mathcal{S}_2(f)$ and $\mathcal{S}_3(f)$(except in certain special cases that we handle separately) in time $(nd)^{O(1)}$ time. We also showed in Lemma 4.3 and Lemma 4.4 that the size of $\mathcal{S}_2$ and $\mathcal{S}_3$ in these cases is $d^{O(1)}$ and thus we can go over all pairs or triplets and find their intersections in time $(nd)^{O(1)}$, and therefore find $\mathcal{L}_{cand}$ in $\text{poly}(n, d)$ time. This process is described in more detail in Algorithm 6.

This section aims to show that using these intersections we can obtain a significant number of linear forms from at least one of the gates in C. It is summarised as the following main theorem of the section.

Theorem 6.1. *Let $\mathbb{F}$ be a field that is $\mathbb{R}$ or $\mathbb{C}$. Let $f \in \mathbb{F}[x_1, \ldots, x_n]$ be a degree d polynomial computed by $\Sigma\Pi\Sigma(3)$ circuit of the form $C = G \times (T_1 + T_2 + T_3)$ such that $\gcd(T_1, T_2, T_3) = 1$. Let $\mathcal{R}(k)$ be as defined in Theorem 3.4. Let $c_{cand} > 36\mathcal{R}(3)$ be any constant. Suppose that C is such that $\text{rank}(\text{sim}(C)) \geq 15c_{cand} \log d$, then there exists an algorithm that runs in $(nd)^{O(\log d)}$ time and with $1 - o(1)$ probability outputs a set of linear forms $\mathcal{L}_{cand}$ such that $|\mathcal{L}_{cand}| = d^{O(1)}$ and there is a gate $T_i, i \in [3]$ such that $\dim(\text{span}(\text{Lin}(T_i) \cap \mathcal{L}_{cand})) \geq c_{cand} \log d$.*

Proof of Theorem 6.1. The analysis of obtain this set of linear forms becomes much easier if we have two additional assumptions namely:

- $\dim(\text{span}(\text{Lin}(G))) \leq 6\mathcal{R}(3) \log d$ and
- T_i is not of the form αl^d for any $\alpha \in \mathbb{F}$ and l that is a linear form in $\mathbb{F}[x_1, \ldots, x_n]$

We show in Theorem 6.2 how to obtain the set of linear forms $\mathcal{L}_{cand}$ in $\text{poly}(n, d)$ time with probability $1-o(1)$ such that $|\mathcal{L}_{cand}| = d^{O(1)}$ and there is a gate $T_i, i \in [3]$ such that $\dim(\text{span}(\text{Lin}(T_i) \cap \mathcal{L}_{cand})) \geq c_{cand} \log d$.

The case where there is a gate T_i that is of the form αl^d for some $\alpha \in \mathbb{F}$ and l that is a linear form in $\mathbb{F}[x_1, \ldots, x_n]$ is handled in Lemma 6.20, where we show how to obtain the required set in $\text{poly}(n, d)$ time with $1 - o(1)$ probability.

We will now argue that we can reduce the problem of finding $\mathcal{L}_{cand}$ for the general setting, with any G to finding $\mathcal{L}_{cand}$ when $\dim(\text{span}(\text{Lin}(G))) \leq 6\mathcal{R}(3) \log d$.

To do this, we first define $\mathcal{L}_{s1} := \{l : \mathbb{V}(l) \in \mathcal{S}_1(f)\} = \{l : l | f\}$ as the set of linear factors of f. We can find $\mathcal{L}_{s1}$ easily as we can get blackbox access to the factors of f in randomized $\text{poly}(n, d)$ time by Lemma 3.7 and then find the linear factors among them. We show in Lemma 6.4 that the set $\mathcal{L}_{s1} \setminus \text{Lin}(G)$ spans a space of dimension at most $6\mathcal{R}(3) \log d$. For all possible $S \subseteq \mathcal{L}_{s1}$ such that $\dim(\text{span}(S)) \leq 6\mathcal{R}(3) \log d$, we divide the circuit by linear forms in $\mathcal{L}_{s1} \setminus \text{span}(S)$ to get a new circuit C'. There will be one such S such that $\text{span}(S) = \text{span}(\mathcal{L}_{s1} \setminus \text{Lin}(G))$, and in this case all the linear forms in $\mathcal{L}_{s1} \setminus \text{span}(S)$ that we divide C by will be in $\text{Lin}(G)$. Therefore, for the new circuit $C' = G' \times (T_1 + T_2 + T_3)$ will still be computable by a $\Sigma\Pi\Sigma(3)$ circuit with $\text{sim}(C) = \text{sim}(C')$ and $\dim(\text{span}(\text{Lin}(G))) \leq 6\mathcal{R}(3) \log d$. As $\text{sim}(C) = \text{sim}(C')$, the set $\mathcal{L}_{cand}$ that satisfies the required properties for C' will also satisfy them for C. Since, we can find the set $\mathcal{L}_{cand}$ for input C' by Theorem 6.2 in $\text{poly}(n, d)$ time, we can find $\mathcal{L}_{cand}$ in $\text{poly}(n, d)$ for each choice of S. Clearly, $|\mathcal{L}_{s1}| \leq d$ and therefore the number of sets S will be at most $d^{O(\log d)}$. Thus, we can compute the set $\mathcal{L}_{s1}$ for general circuit with high rank in time $(nd)^{O(\log d)}$. □

Theorem 6.2. *Let $\mathbb{F}$ be a field that is $\mathbb{R}$ or $\mathbb{C}$. Let $f \in \mathbb{F}[x_1, \ldots, x_n]$ be a degree d polynomial computed by $\Sigma\Pi\Sigma(3, d)$ circuit of the form $C = G \times (T_1 + T_2 + T_3)$ such that $\gcd(T_1, T_2, T_3) = 1$. Let $\mathcal{R}(k)$ be as defined in Theorem 3.4. Let $c_{cand} > 36\mathcal{R}(3)$ be any constant. Suppose that C is such that*

1. $\text{rank}(\text{sim}(C)) \geq 15c_{cand} \log d$
2. $\dim(\text{span}(\text{Lin}(G))) \leq 6\mathcal{R}(3) \log d$ *and*
3. T_i *is not of the form* αl^d *for any* $\alpha \in \mathbb{F}$ *and* l *that is a linear form in* $\mathbb{F}[x_1, \ldots, x_n]$

then there exists an algorithm that runs in $\text{poly}(n, d)$ *time and with* $1 - o(1)$ *probability outputs a set of linear forms* $\mathcal{L}_{cand}$ *such that* $|\mathcal{L}_{cand}| = d^{O(1)}$ *and there is a gate* $T_i, i \in [3]$ *such that* $\dim(\text{span}(\text{Lin}(T_i) \cap \mathcal{L}_{cand})) \geq c_{cand} \log d$.

Proof. Let $\mathcal{L}_{s1}$ be the set of linear forms l that divide f. To show that we can obtain $c_{cand} \log d$ candidate linear forms, we divide the analysis into the following cases and then prove that we can get the candidate linear forms in separate lemmas.

- There is no linear form l such that $\text{rank}(\text{sim}(C \mod l)) \leq c_2$ and $C \mod l \neq 0$.
  - When there are at least 2 high-rank gates i.e. wlog T_1, T_2 are such that $\dim(\text{span}(\text{Lin}(T_1))) \geq 5c_{cand} \log d$ and $\dim(\text{span}(\text{Lin}(T_2))) \geq 5c_{cand} \log d$
    - The Third gate T_3 is such that $\dim(\text{span}(\text{Lin}(T_3) \setminus \mathcal{L}_{s1})) \geq 2$. (Lemma 6.10)
    - The Third gate T_3 is such that $\dim(\text{span}(\text{Lin}(T_3) \setminus \mathcal{L}_{s1})) \leq 1$. (Lemma 6.13)
  - There is exactly 1 gate, wlog T_1, with rank greater than $5c_{cand} \log d$. (Lemma 6.14)
- There is a linear form l such that $\text{rank}(\text{sim}(C \mod l)) \leq c_2$ and $C \mod l \neq 0$.
  - No T_i vanish mod l. (Lemma 6.15)
  - $\text{rank}(\text{sim}(T_2 + T_3)) < c_2$ and $l | T_1$. (Lemma 6.17)
  - $l | T_1$, $\text{rank}(\text{sim}(T_2 + T_3)) \geq c_2$ but $\text{rank}(\text{sim}((T_2 + T_3) \mod l)) < c_2$. (Lemma 6.18)

All the cases of this theorem are covered by the above division, and we will see the proofs for each of the cases in the following section.

□

Before we go into the details of the cases outlined in Theorem 6.2, we will need some structural results about $\mathcal{S}_1, \mathcal{S}_2$ spaces which we will discuss in the next subsection.

6.1 Structural Results

Our first structural result is the following lemma, which bounds the dimension of the set of "rank-reducing" linear forms. The proof of Claim 4.8 in [DS05] implies this result. It can also be inferred as a special case of Lemma B.1 (restated with better notation in section B.1.2) in [KS09a], with $A = [k], \hat{r} = rk(sim(T_1 + T_2 + \ldots + T_k)), r_t = r', \chi = \left\lfloor \frac{\hat{r}}{2r' \log d} \right\rfloor$.

Lemma 6.3 (Implied from Claim 4.8,[DS05])**.** *Let* C *be a* $\Sigma\Pi\Sigma(k)$ *circuit of the form* $T_1 + T_2 + \ldots + T_k$ *such that* $\gcd(T_1, \ldots, T_k) = 1$. *Fix* $r' > 0$ *to be any constant such that* $\text{rank}(\text{sim}(T_1 + T_2 + \ldots + T_k)) > 2r' \log d + 2k$. *We define a linear form* l *to be rank-reducing if* $\forall i \in [k]$ $l \nmid T_i$ *and* $rank(sim(C \mod l)) \leq r'$. *If we define a set of rank-reducing linear forms for* C *as*

$$\mathcal{L} := \{l : l \text{ is rank-reducing for } C\}$$

then $\dim(\text{span}(\mathcal{L})) \leq max(r' \log d, 2k \log dk + 2k)$.

The next structural lemma bounds the dimension of linear forms dividing f that do not divide the gcd G.

Lemma 6.4. *Let $\mathcal{R}(k)$ be as defined in Theorem 3.4. Let $f \in \mathbb{F}[x_1, \ldots, x_n]$ be a degree d polynomial computed by $\Sigma\Pi\Sigma(3)$ circuit of the form $C = G \times (T_1 + T_2 + T_3)$ such that $\gcd(T_1, T_2, T_3) = 1$, we define*

$$\mathcal{L}_{s1} := \{l : \mathbb{V}(l) \in \mathcal{S}_1(f)\} = \{l : l | f\}$$

then, we have $\dim(\text{span}(\mathcal{L}_{s1} \setminus \text{Lin}(G))) \leq 5\mathcal{R}(3)\log d + 42$. In particular, as we consider polynomial families with increasing d, for large enough d, we have

$$\dim(\text{span}(\mathcal{L}_{s1} \setminus \text{Lin}(G))) \leq 6\mathcal{R}(3)\log d$$

Proof. The only interesting case is when $\text{rank}(T_1 + T_2 + T_3) \geq 5\mathcal{R}(3)\log d + 42$ as otherwise the contribution of linear forms dividing $T_1 + T_2 + T_3$ has to be less than $\text{rank}(T_1 + T_2 + T_3)$. So, we assume $rk(T_1 + T_2 + T_3) \geq 5\mathcal{R}(3)\log d + 42$. Let l be any linear form such that $l|(T_1 + T_2 + T_3)$. Such an l must be one of the following two kinds.

1. **Case 1:** $\forall i \in [3]\ \ l \nmid T_i$. As $f = 0 \mod l$, the circuit $(C \mod l)$ computes 0. Let the circuit $C \mod l$ be of the form $G' \times (T_1' + T_2' + T_3')$ with $\gcd(T_1', T_2', T_3') = 1$. Note that $(T_1' + T_2' + T_3')$ is a simple circuit computing 0, and hence by definition, $\text{rank}(T_1' + T_2' + T_3') < \mathcal{R}(3)$. Let $\mathcal{L}_{s1}'$ be the set of linear forms in $\mathcal{L}_{s1}$ which do not divide any T_i. We can use Lemma 6.3 with $r' = \mathcal{R}(3), k = 3$ and $\text{rank}(\text{sim}(T_1 + T_2 + T_3)) > 2\mathcal{R}(3)\log d + 6$, to obtain $dim(sp(\mathcal{L}_{s1}')) \leq max(\mathcal{R}(3)\log d, 6\log(3d) + 6) \leq 2\mathcal{R}(3)\log d + 18$.

2. **Case 2:** $\exists i \in [3]\ \ l | T_i$. If a linear form is such that it divides 2 T_i's, then for it to be in $\mathcal{L}_{s1}$, it also has to divide the third gate which means it contributes to G. Thus we are considering linear forms l that divide a gate (say T_1) and $\text{sim}(T_2 + T_3)$(but do not divide T_2 or T_3). If $\text{rank}(\text{sim}(T_2 + T_3)) < 2\log d$, then we are done as $\dim(\text{span}(\text{Lin}(\text{sim}(T_2 + T_3))) < \text{rank}(\text{sim}(T_2 + T_3)) < 2\log d$ and thus the linear forms in this case contribute to the dimension only by $2\log d$. Thus let us assume $\text{rank}(\text{sim}(T_2 + T_3)) \geq 2\log d$. Now if l divides $\text{sim}(T_2 + T_3)$, then when we go mod l, the rank reduces from greater than $2\log d$ to 1, as $T_2' + T_3' = 0$ only when $T_2' = -T_3'$. So, using Lemma 6.3 with $r' = 1$ and $\text{rank}(\text{sim}(T_2 + T_3)) > 2\log d$, the linear forms contributing in this case would lie in a $4\log 2d + 4$ dimensional space. Since i could be 1, 2, or 3, there are actually three such cases to consider. Thus, we have the linear forms in $\mathcal{L}_{s1}$ in this case lie in at most a $12\log d + 24 \leq 3\mathcal{R}(3)\log d + 24$ dimensional space.

Combining the two cases we get that $\dim(\text{span}(\mathcal{L}_{s1}) \setminus \text{Lin}(G)) \leq 5\mathcal{R}(3)\log d + 42$.

□

We see from the above proof that it is important that when the rank is high, for a codimension 1 vanishing space outside the gcd to exist, the gcd after we go mod the linear form needs to be high dimension and the remaining circuit needs to be low rank for it to be identity. In the case, where one of the gates T_i is such that $\dim(\text{span}(\text{Lin}(T_i)))$ is small, the rank of the new gcd will be upper bounded by the rank of the smallest rank gate, and hence there will be no linear forms in $\mathcal{S}_1 \setminus G$(except the ones that divide the low-rank gate and also $\text{sim}(C)$). We argue this idea more formally in the following lemma.

Lemma 6.5. *Let $\mathcal{R}(k)$ be as defined in Theorem 3.4. Let $f \in \mathbb{F}[x_1, \ldots, x_n]$ be a degree d polynomial computed by $\Sigma\Pi\Sigma(3)$ circuit of the form $C = G \times (T_1 + T_2 + T_3)$ such that $\gcd(T_1, T_2, T_3) = 1$. Moreover assume that $\text{rank}(\text{sim}(C)) \geq 2\mathcal{R}(3)\log d$, and $\exists\ i \in [3]$ such that $\text{rank}(\text{sim}(C)) - \dim(\text{span}(\text{Lin}(T_i))) \geq \mathcal{R}(3) + 3$. Then*

1. *Any* $l \in \mathcal{L}_{s1} \setminus \mathrm{Lin}(G)$ *will be such that* $l | T_j$ *for some* $j \in [3]$

2. $\mathcal{S}_2(f)$ *contains only spaces over which at least one of* T_1, T_2, T_3 *vanish.*

Proof. Let $l \in \mathcal{L}_{s1} \setminus \mathrm{Lin}(G)$. For the sake of contradiction, assume no multiplication gate vanishes when we go mod l. Let $C \mod l$ be of the form $G' \times (T_1' + T_2' + T_3')$ where $\gcd(T_1' + T_2' + T_3') = 1$. Then $dim(\mathrm{span}(\mathrm{Lin}(G'))) \leq \dim(\mathrm{span}(\mathrm{Lin}(G)) + \dim(\mathrm{span}(\mathrm{Lin}(T_i)))$, and therefore $\mathrm{rank}(\mathrm{sim}(C \mod l)) \geq \mathrm{rank}(\mathrm{sim}(C)) - \dim(\mathrm{span}(\mathrm{Lin}(T_i))) - 1 > \mathcal{R}(3)$. Thus $C \mod l \neq 0$. Hence, there are no linear forms $l \in \mathcal{L}_{s1} \setminus \mathrm{Lin}(G)$ that give rise to this case.

Now, we consider $\mathcal{S}_2(f)$ and want to show that for any $\mathbb{V}(l_1, l_2) \in \mathcal{S}_2(f)$ at least one of T_i's vanishes. Consider the case where none of the T_i's vanish on $\mathbb{V}(l_1, l_2) \in \mathcal{S}_2(f)$. Let $C \mod \langle l_1, l_2 \rangle = G' \times (T_1' + T_2' + T_3')$ where $\gcd(T_1', T_2', T_3') = 1$. Then, $\dim(\mathrm{span}(\mathrm{Lin}(G'))) \leq \dim(\mathrm{span}(\mathrm{Lin}(G)) + \dim(\mathrm{span}(\mathrm{Lin}(T_i)))$, and therefore we have

$$\mathrm{rank}(\mathrm{sim}(C \mod \langle l_1, l_2 \rangle)) \geq \mathrm{rank}(\mathrm{sim}(C)) - \dim(\mathrm{span}(\mathrm{Lin}(T_i))) - 2 > \mathcal{R}(3)$$

by assumption. Thus $C \mod \langle l_1, l_2 \rangle \neq 0$, but this cannot be since $\mathbb{V}(l_1, l_2) \in \mathcal{S}_2(f)$. Therefore, $\mathcal{S}_2(f)$ is empty in this case. □

The set $\mathcal{S}_2(f)$ contains all spaces of the form $\mathbb{V}(l_1, l_2, l_3)$ with $\dim(\mathrm{span}(l_1, l_2, l_3)) = 2$ and $l_1 \in \mathrm{Lin}(T_1), l_2 \in \mathrm{Lin}(T_2), l_3 \in \mathrm{Lin}(T_3)$. These spaces are useful as the intersection of the kernels of such spaces gives us linear forms in the circuit. At the same time, it also contains spaces that do not give us any information about the circuit and prevent us from learning codimension 3 spaces contained in them. To capture this usefulness, we partition $\mathcal{S}_2(f)$ into two subsets of spaces $\mathcal{S}_2^{reg}$ and $\mathcal{S}_2^{sp}$ as defined below:

$$\begin{aligned}\mathcal{S}_2^{reg}(f) := \{\mathbb{V}(l_1, l_2) : \mathbb{V}(l_1, l_2) \in \mathcal{S}_2,\ &\text{For a random } l \in \mathrm{span}(l_1, l_2) \text{ and} \\ &\mathrm{span}(l') := (\mathrm{span}(l_1, l_2) \mod l)\,,\ \text{with high probability } l' | \mathrm{sim}(C \mod l)\}\end{aligned}$$

$$\mathcal{S}_2^{sp}(f) := \mathcal{S}_2 \setminus \mathcal{S}_2^{reg}(f)$$

Now, we will work on bounding the structure of $\mathcal{S}_2^{reg}(f)$.

Definition 9 (Independent Vanishing Set)**.** *We define a subset* $W \subseteq \mathcal{S}_2^{reg}(f)$ *to be an independent vanishing set if there exists an integer* k *such that*

- $|W| = k$
- $\dim(\mathrm{span}(\{\mathrm{span}(l_1, l_2) : \mathbb{V}(l_1, l_2) \in W\})) = 2k$

i.e. the kernels of the spaces in W *are independent.*

We will argue in the following lemma that the size of any Independent Vanishing Set W is small.

Lemma 6.6. *Let* f *be a* n*-variate degree* d *polynomial over any infinite field*[4] *computed by circuit* $C = G \times (T_1 + T_2 + T_3)$ *such that* $\gcd(T_1, T_2, T_3) = 1$. *Let* $\mathcal{R}(k)$ *be as defined in Theorem 3.4. Then for any* $W \subseteq \mathcal{S}_2^{reg}(f)$ *that is an independent vanishing set for* f *we have*

$$|W| \leq 6\mathcal{R}(3) \log d$$

[4] in fact any large enough field would suffice

We first state and prove a simple claim that will be useful for the proof.

Claim 6.7. *Let $S = \{U_1, U_2, \dots U_r\}$ be any collection of 2-dimensional subspaces of $\mathbb{F}^n$ (where $\mathbb{F}$ is an infinite or large enough field) such that $\dim(\text{span}(S)) = 2r$. For $i \in [r]$, let l_i be the linear form corresponding to a uniformly random vector sampled from U_i. Let $f \in \mathbb{F}[x_1 \dots x_n]$ be a nonzero degree d polynomial. Then the probability that $f = 0 \mod \langle l_1, \dots, l_r \rangle$ is vanishingly small.*

Proof. First observe that it suffices to prove the above result for $r = 1$ and then use induction (since $U_1, U_2, \dots U_r$ are all independent). For $r = 1$, note that for any linear form l such that $f = 0 \mod \langle l \rangle$, l must be a linear factor of f. Since f can have at most d distinct linear factors (up to scaling), the result follows.

□

We now prove Lemma 6.6

Proof of Lemma 6.6. Let $k = |W|$. Let $W = \{W_1, W_2, \dots W_k\}$. Now each $W_i \in W$ is of the form $\mathbb{V}(l_{i1}, l_{i2})$. Let $V_i = \text{span}\{v_{i1}, v_{i2}\}$ where $v_{i1} \in \mathbb{F}^n$ is the vector corresponding to l_{i1} and v_{i2} is the vector corresponding to l_{i2}.

For each i let l_i be the linear form corresponding to a uniformly random vector sampled from V_i. Let l'_i be such that $\text{span}\{l_i, l'_i\} = \text{span}\{l_{i1}, l_{i2}\}$.

Let $A = \{l_i : i \in [k]\}$.

Now, we consider the circuit $C' = C \mod \langle A \rangle$. Let $f' = f \mod \langle A \rangle$.

Let C' be of the form $C' = G' \times (T'_1 + T'_2 + T'_3)$, where $\gcd(T'_1, T'_2, T'_3) = 1$. Observe that by Claim 6.7, $C' \neq 0$. Moreover each of G', T'_1, T'_2, T'_3 compute nonzero polynomials. We will show that $l'_1, l'_2, \dots, l'_k$ are all linear factors of f' (in the space $\mathbb{V}(l_1, l_2, \dots, l_k)$) that divide $(T'_1 + T'_2 + T'_3)$. Then, by Lemma 6.4, the bound on k follows.

Let $C \mod l_i$ be of the form $G_i \times (T_{1i} + T_{2i} + T_{3i})$ with $\gcd(T_{1i}, T_{2i}, T_{3i}) = 1$. By definition of $\mathcal{S}_2^{reg}$, we have l'_i divides $(T_{1i} + T_{2i} + T_{3i})$. We would like to show that l'_i continues to divide $T'_1 + T'_2 + T'_3$, i.e. it divides $\text{sim}(T_{1i} + T_{2i} + T_{3i} \mod \langle A \rangle)$.

Let T_{1i} be a gate such that $l'_i \nmid T_{1i}$. Such a gate exists as $\gcd(T_{1i}, T_{2i}, T_{3i}) = 1$. It suffices to show that no linear form dividing T_{1i} becomes equal to l'_i when we consider it over $\mathbb{V}(l_1, l_2, \dots, l_k)$. Consider any linear form l dividing T_{1i}. By assumption, $\text{span}\{l'_i\}$ cannot contain any factor of T_{1i}. Thus $l \notin \text{span}\{l'_i\}$ and hence it is of the form $\beta l'_i + l'$ where $l' \notin \text{span}\{l'_i\}$. By Claim 6.7 (applied to $\beta l'_i + l'$ but in the space $\mathbb{V}(l'_i)$), l' remains nonzero with high probability when we go mod $l_1, l_2, \dots, l_{i-1}, l_{i+1}, \dots l_k$, and moreover is still not in $\text{span}\{l'_i\}$.

□

We also define a set of codimension 3 spaces $\mathcal{S}_3^{sp}(f)$ for a polynomial f computed by a $\Sigma\Pi\Sigma(3)$ circuit as follows

$$\mathcal{S}_3^{sp}(f) = \{\mathbb{V}(l_1, l_2, l_3) : l_1 | T_1,\ l_2 | T_2,\ l_3 | T_3,\ \text{and}\ \dim(\text{span}(l_1, l_2, l_3)) = 3,\\ \mathbb{V}(l_1, l_2, l_3) \subset \mathbb{V}(l'_1, l'_2) \in \mathcal{S}_2^{sp}(f)\ \text{and}, l_1, l_2, l_3 \notin \text{span}(l'_1, l'_2)\}$$

We will below describe an algorithm that has knowledge of $\mathcal{S}_2(f)$ and it computes a set $\overline{\mathcal{S}_3^{sp}}$ such that $|\overline{\mathcal{S}_3^{sp}}| = d^{O(1)}$ and it contains all the spaces in $\mathcal{S}_3^{sp}(f)$.

Algorithm 5 Computing codimension 3 vanishing spaces contained in $\mathcal{S}_2^{sp}$

Input: Blackbox access to circuit C of form $\Sigma\Pi\Sigma(3,d)$ computing polynomial $f \in \mathbb{F}[x_1,\ldots,x_n]$

function SET-CONTAINING-$\mathcal{S}_3^{sp}(C)$
  Compute $\mathcal{S}_2(f)$ using Algorithm 2
  for $\mathbb{V}(l_1,l_2) \in \mathcal{S}_2(f)$ **do**
    Pick a random vector l in $\text{span}(l_1,l_2)$
    Let $C' := C \mod l$ and $l' := \text{span}(l_1,l_2) \mod l$.
    Let e be the multiplicity of l' in $\text{Lin}(C)$
    Let $\bar{C} := (C'/(l')^e) \mod l'$. Factorize $\bar{C}$ to get access to $\text{Lin}(\bar{C})$.
    for $l^* \in \text{Lin}(\bar{C})$ **do**
      Add $\mathbb{V}(l_1,l_2,l^*)$ to $\overline{\mathcal{S}_3^{sp}}$
  Output $\overline{\mathcal{S}_3^{sp}}$

Lemma 6.8. *Let f computed by a $\Sigma\Pi\Sigma(3)$ circuit C with $\text{rank}(\text{sim}(C)) \geq c_2$, where c_2 is as in Lemma 4.3. Then, there exists an algorithm(Algorithm 5) that outputs a set $\overline{\mathcal{S}_3^{sp}}$ with $|\overline{\mathcal{S}_3^{sp}}| = d^{O(1)}$ in $\text{poly}(n,d)$-time with probability $1-o(1)$ such that $\mathcal{S}_3^{sp}(f) \subseteq \overline{\mathcal{S}_3^{sp}}$.*

Proof. Consider any codimension 3 space $W = \mathbb{V}(l_1',l_2',l_3')$ such that $l_1'|T_1$, $l_2'|T_2$, $l_3'|T_3$ and $\mathbb{V}(l_1',l_2',l_3') \subseteq \mathbb{V}(l_1,l_2) \in \mathcal{S}_2^{sp}(f)$ and $l_1',l_2',l_3' \notin \text{span}(l_1,l_2)$. Recall from Algorithm 5, l is a random vector in $\text{span}(l_1,l_2)$ and $l' = \text{span}(l_1,l_2) \mod l$. As $\mathbb{V}(l_1',l_2',l_3') \subseteq \mathbb{V}(l_1,l_2)$, $\text{span}(l_1,l_2) \subseteq \text{span}(l_1',l_2',l_3')$ and $\text{span}(l_1',l_2',l_3') \mod \langle l,l'\rangle = \text{span}(l_1',l_2',l_3') \mod \langle l_1,l_2\rangle = \text{span}(l^*)$ for a linear form l^*, i.e. $\text{span}(l,l',l_*) = \text{span}(l_1',l_2',l_3')$. We want to argue that $l^*|\bar{C}$ where $\bar{C}$ is as defined in Algorithm 5. We first show that $\bar{C}$ is a $\Sigma\Pi\Sigma(3)$ circuit computing a non-zero polynomial. From Claim 6.7, C' is non-zero and $\bar{C}$ being non-zero follows. As $\mathbb{V}(l_1,l_2)$ is a space in $\mathcal{S}_2^{sp}(f)$, which means when we consider the circuit $C' = C \mod l$, each gate in the circuit is divisible by l'. By definition of $\mathcal{S}_2^{sp}$ $l' \nmid \text{sim}(C')$, as $\mathbb{V}(l_1,l_2)$ would be a $\mathcal{S}_2^{reg}$ space in that case. Therefore, all $l' \in \text{Lin}(C')$ occurrences will come from the gcd. Therefore after dividing by $(l')^e$ for e being the multiplicity of e, $sim(C')$ remains a $\Sigma\Pi\Sigma(3)$ circuit. Now, we have after going mod $\langle l,l'\rangle$ $\bar{C}$ is a $\Sigma\Pi\Sigma(3)$ circuit computing a non-zero polynomial. From assumption, none of the linear forms l_1',l_2',l_3' is zero mod $\langle l,l'\rangle$ and all the nonzero linear forms will be multiple of l^*. Therefore, l^* will divide the nonzero circuit $\bar{C}$. Hence, when Algorithm 5 iterates over $\mathbb{V}(l_1,l_2) \in \mathcal{S}_2^{sp}(f) \subseteq \mathcal{S}_2(f)$, one of the spaces added to $\overline{\mathcal{S}_3^{sp}}$ will be $\mathbb{V}(l_1',l_2',l_3')$.

As $|\mathcal{S}_2(f)| = d^{O(1)}$ from Lemma 4.3 as $\text{rank}(\text{sim}(C)) \geq c_2$, and $|\text{Lin}(\bar{C})| \leq d$, the number of spaces added into $\overline{\mathcal{S}_3^{sp}}$, will also be $d^{O(1)}$. From Lemma 5.1, we know $\mathcal{S}_2(f)$ can be computed in $\text{poly}(n,d)$ time, while the rest of the operations in the loop of Algorithm 5 are also $\text{poly}(n,d)$ time each repeated $|\mathcal{S}_2(f)| = d^{O(1)}$ so the entire algorithm runs in $\text{poly}(n,d)$ time. □

We make another observation as follows: if the rank of the simple part of the circuit of T_1+T_2 is large, then the number of essential variables is also large.

Lemma 6.9. *Let f be a homogeneous polynomial computed by a $\Sigma\Pi\Sigma(2)$ circuit $C = T_1+T_2$ such that $\gcd(T_1,T_2) = 1$. Let $t > 0$ be any nonnegative integer such that $\dim(\text{span}(\text{Lin}(T_1))) \geq t$ and $\dim(\text{span}(\text{Lin}(T_2))) \geq t$ and let $l,l_1,l_2 \in \mathbb{F}[x_1,\ldots,x_n]$ be arbitrary linear forms. $\mathcal{R}(k)$ as defined in Theorem 3.4. Then*

- *The number of essential variables(Definition 6) in T_1+T_2 is at least $\frac{t-\mathcal{R}(3)}{2}$.*
- *If $f \mod l$ has less than $\frac{t-\mathcal{R}(3)-1}{3}$ essential variables, then $f \mod l = 0$*

- *If* $f \mod \langle l_1, l_2\rangle$ *has less than* $\frac{t-\mathcal{R}(3)-2}{3}$ *essential variables, then* $f \mod \langle l_1, l_2\rangle = 0$

Proof. Let the number of essential variables in T_1+T_2 be c. Thus by definition $T_1+T_2 = g(l_1, \ldots, l_c)$ for some homogeneous polynomial g in $\mathbb{F}[x_1, \ldots, x_c]$ and linear forms $l_1, \ldots, l_c$ in $\mathbb{F}[x_1, \ldots, x_n]$. Let z be a new variable and consider a random linear isomorphism Φ which for each $i \in [c]$ maps $l_i \to \alpha_i z$ for a random $\alpha_i \in \mathbb{F}$. Then with high probability g is nonzero and is of the form αz^d for some constant $\alpha \in \mathbb{F}$. Therefore, we have $\Phi(T_1 + T_2) - \alpha z^d = 0$. Now we have a $\Sigma\Pi\Sigma(3)$ circuit equalling 0 and hence we can use rank bounds! By Lemma 3.4, $\text{rank}(\text{sim}(\Phi(T_1+T_2)-\alpha z^d)) \leq \mathcal{R}(3)$.

We will first show that the linear forms contributing to the GCD have rank at most c.

Consider any two linear forms $l_a \in T_1, l_b \in T_2$. As $\gcd(T_1, T_2) = 1$, we have $\text{span}(l_a) \neq \text{span}(l_b)$ and hence these linear forms do not contribute to the gcd. Suppose after applying Φ, two distinct linear forms got "collapsed" to the same and moved into the gcd. In other words $\text{span}(\Phi(l_a)) = \text{span}(\Phi(l_b))$. We will now show that the only way this can happen is if $l_a, l_b \in \text{span}(l_1, \ldots, l_c)$. This will then imply that the linear forms in the gcd have rank at most c.

Let $l_a = l'_a + \sum_{i=1}^{c} \beta_{a,i} l_i$ where $\beta_{a,i} \in \mathbb{F}$, and $l'_a = 0$ or $l'_a \notin \text{span}(l_1, \ldots, l_c)$. Similarly $l_b = l'_b + \sum_{i=1}^{c} \beta_{b,i} l_i$ where $\beta_{b,i} \in \mathbb{F}$, and $l'_b = 0$ or $l'_b \notin \text{span}(l_1, \ldots, l_c)$. We have $\Phi(l_a) = l'_a + \sum_{i=1}^{c} \beta_{a,i}\alpha_i z$ and $\Phi(l_b) = l'_b + \sum_{i=1}^{c} \beta_{b,i}\alpha_i z$.

Case 1: $\text{span}(l'_a) \neq \text{span}(l'_b)$. In this case $\Phi(l_a)$ and $\Phi(l_b)$ clearly remain independent.

Case 2: $\text{span}(l'_a) = \text{span}(l'_b)$. In case these spans are actually 0, then we are done. So let us assume the span is nonzero. In this case, without loss of generality assume the linear forms are scaled such that $l'_a = l'_b$. Then, since $\text{span}(l_a) \neq \text{span}(l_b)$, for some i, $\beta_{a,i} \neq \beta_{b,i}$. Hence with high probability $\sum_{i=1}^{c} \beta_{a,i}\alpha_i \neq \sum_{i=1}^{c} \beta_{b,i}\alpha_i$. Therefore $\Phi(l_a)$ and $\Phi(l_b)$ remain independent with high probability.

So, the only linear forms that move to the gcd of $\Phi(T_1), \Phi(T_2)$ are the ones that lie in $\text{span}(l_1, \ldots, l_c)$. Therefore, the linear forms that move into the gcd lie in a space of dimension at most c. Moreover, after applying Φ, the span of the linear forms from T_1 that do not move to the gcd can get shrunk by at most c. Therefore, $\text{rank}(\text{sim}(\Phi(T_1+T_2)-\alpha z^d)) \geq t-2c$. By the rank bound, thus $t-2c \leq \mathcal{R}(3)$, which gives us $c \geq \frac{t-\mathcal{R}(3)}{2}$. Thus finishes the first part of the lemma.

We now show that if $f \mod l$ is nonzero then it must have a large number of essential variables, and we will show how to deduce this either from the gcd or the simple part of the circuit.

$C \mod l$ is of the form $G' \times (T'_1 + T'_2)$ with $\gcd(T'_1, T'_2) = 1$. Now if $\dim(\text{span}(\text{Lin}(G'))) \geq \frac{t-\mathcal{R}(3)-1}{3}$, then clearly the number of essential variables of $C \mod l$ is greater than or equal to $\frac{t-\mathcal{R}(3)-1}{3}$ (unless $C \mod l = 0$) since G' is a product of linear forms which will continue to have high rank under any linear isomorphism. In the other case, when $\dim(\text{span}(\text{Lin}(G'))) < \frac{t-\mathcal{R}(3)-1}{3}$, then $\dim(\text{span}(\text{Lin}(T'_1))) \geq \frac{2t+\mathcal{R}(3)-2}{3}$ and $\dim(\text{span}(\text{Lin}(T'_2))) \geq \frac{2t+\mathcal{R}(3)-2}{3}$. Therefore by part 1 of the current lemma, we have $C \mod l$ has at least $\frac{t-\mathcal{R}(3)-1}{3}$ essential variables. Therefore, if $C \mod l$ has less than $\frac{t-\mathcal{R}(3)-1}{3}$ essential variables, then $C \mod l = 0$.

Similarly if $C \mod \langle l_1, l_2\rangle$ has less than $\frac{t-\mathcal{R}(3)-2}{3}$ essential variables, $C \mod \langle l_1, l_2\rangle = 0$. □

6.2 Candidate linear forms when there are at least two high rank gates

Wlog, we assume $\dim(\text{span}(\text{Lin}(T_1))) \geq 5c_{cand} \log d$ and $\dim(\text{span}(\text{Lin}(T_2))) \geq 5c_{cand} \log d$. Now, there could be two cases based on $\dim(\text{span}(\text{Lin}(T_3) \setminus \mathcal{L}_{s1}))$, which will be covered in the following subsections:

6.2.1 Obtaining candidate linear forms when $\dim(\text{span}(\text{Lin}(T_3) \setminus \mathcal{L}_{s1})) \geq 2$

In this case, we obtain the candidate linear forms using the sets $\mathcal{L}_2$ and $\mathcal{L}_3$, which we compute using Algorithm 6 (which runs in time $\text{poly}(n, d)$). Note, in $\mathcal{L}_3$, we also include linear forms that are the intersection of kernels of spaces in $\overline{\mathcal{S}_3^{sp}}$.

Algorithm 6 Computing Candidate Linear Forms

Input: Blackbox access to circuit C of form $\Sigma\Pi\Sigma(3)$ computing polynomial $f \in \mathbb{F}[x_1, \ldots, x_n]$ and $\mathcal{S}_3(f), \mathcal{S}_2(f)$

function $cand - \mathcal{L}(f)(C)$
  Compute $\mathcal{S}_2(f)$ and $\mathcal{S}_3(f)$ using Algorithms 2 and 4. Compute $\overline{\mathcal{S}_3^{sp}}$ using Algorithm 5. Set $\bar{\mathcal{S}}_3(f) := \mathcal{S}_3(f) \cup \overline{\mathcal{S}_3^{sp}}(f)$
  Initialize $\mathcal{L}_2, \mathcal{L}_3, \mathcal{L}_{cand} \leftarrow \phi$
  for $S_1, S_2, S_3 \in \bar{\mathcal{S}}_3(f)$ **do**
    Let $S_i = \mathbb{V}(l_{i1}, l_{i2}, l_{i3})$
    Check if $dim(\cap_{i\in[3]} sp(l_{i1}, l_{i2}, l_{i3})) = 1$, which means the three spaces intersect in a line. Find the line l that is the intersection of the three spaces and add it to $\mathcal{L}_3$
  $\mathcal{L}_{cand} \leftarrow \mathcal{L}_{cand} \cup \mathcal{L}_3$
  for $S_1, S_2 \in \mathcal{S}_2(f)$ **do**
    Let $S_i = \mathbb{V}(l_{i1}, l_{i2})$
    Check if $dim(\cap_{i\in[2]} sp(l_{i1}, l_{i2})) = 1$, which means the two spaces intersect in a line. Find the line l that is the intersection of the two spaces.
    Add l to $\mathcal{L}_2$
  $\mathcal{L}_{cand} \leftarrow \mathcal{L}_{cand} \cup \mathcal{L}_2$
  Output $\mathcal{L}_{cand}$

In the following lemma, we show that if there are 2 gates with at least $5c_{cand} \log d$ independent linear forms, then our output $\mathcal{L}_{cand}$ from either Algorithm 6 and Algorithm 7 will have at least $c_{cand} \log d$ linear forms from one of the T_i's. Recall $\mathcal{L}_{s1}$ is the set of linear forms l that divide f.

Lemma 6.10. *Let f be a polynomial computed by $\Sigma\Pi\Sigma(3)$ circuit $C = G \times (T_1 + T_2 + T_3)$ such that $\gcd(T_1, T_2, T_3) = 1$ and satisfies all properties from Theorem 6.2. Also, there doesn't exist linear form l such that $\text{rank}(\text{sim}(C \mod l)) \leq c_2$ and $C \mod l \neq 0$, where c_2 is as defined in Lemma 4.3, $\dim(\text{span}(\text{Lin}(T_1))) \geq 5c_{cand} \log d$ and $\dim(\text{span}(\text{Lin}(T_2))) \geq 5c_{cand} \log d$, and $\dim(\text{span}((\text{Lin}(T_3) \setminus \mathcal{L}_{s1}))) \geq 2$. Then, there exists an algorithm that computes a set of linear forms $\mathcal{L}_{cand}$ such that $|\mathcal{L}_{cand}| = d^{O(1)}$ and $\exists j \in [3]$ such that $\dim(\text{span}(\text{Lin}(T_j) \cap \mathcal{L}_{cand})) \geq c_{cand} \log d$.*

Proof. Observe that any linear form $l | T_1 \times T_2 \times T_3$ is in $\mathcal{L}_{cand}$ which is output of Algorithm 6, if it satisfies any of the following properties:

1. there exist three distinct spaces in $\bar{\mathcal{S}}_3(f) = \mathcal{S}_3(f) \cup \overline{\mathcal{S}_3^{sp}}$, the intersection of whose kernels is $\text{span}(l)$.

2. there exist two distinct spaces in $\mathcal{S}_2(f)$, the intersection of whose kernels is $\text{span}(l)$.

We need to show that, for at least $c_{cand} \log d$ independent linear forms in some T_j, at least one of the three above conditions is met. We will heavily use the structural results we obtained in the previous subsection. We will further break up the proof into the following cases:

- **Case 1:** $\forall\ i \in [3], dim(\text{span}(\text{Lin}(T_i))) \geq (2c_{cand} + 36\mathcal{R}(3)) \log d + 5$

  The set $\mathcal{L}_{cand}$ in this case will be the output of Algorithm 6. From Lemma 6.4, we know that the $dim(\text{span}(\mathcal{L}_{s1} \setminus \text{Lin}(G))) \leq 6\mathcal{R}(3) \log d$. Therefore, the dimension of the span of linear forms from any T_i in $\mathcal{L}_{s1}$ is at most $6\mathcal{R}(3) \log d$. Fix S to be any maximal set of independent spaces[5] of the form $\text{span}(l_1, l_2, l_3)$ with $l_1 \in T_1, l_2 \in T_2, l_3 \in T_3$ such that $\mathbb{V}(l_1, l_2, l_3)$ is contained in some space $\mathbb{V}(l_1', l_2') \in \mathcal{S}_2^{reg}(f)$, in particular $\text{span}(l_1', l_2') \subset \text{span}(l_1, l_2, l_3)$. As the spaces $\text{span}(l_1, l_2, l_3)$ are all independent, all the corresponding spaces in $\mathcal{S}_2^{reg}(f)$ will have independent kernels and hence will be an Independent Vanishing Set. Thus it follows from Lemma 6.6 that $|S| \leq 6\mathcal{R}(3) \log d$ and therefore $\dim(\text{span}(S)) \leq 18\mathcal{R}(3) \log d$. First, we observe the following. Let W be any codimension-3 space of the form $\mathbb{V}(l_1, l_2, l_3)$ on which f vanishes and such that $l_1 \in T_1, l_2 \in T_2, l_3 \in T_3$. Then if $W \subseteq \mathbb{V}(l_1', l_2')$ such that $\mathbb{V}(l_1', l_2') \in \mathcal{S}_2^{reg}(f)$, then $\text{span}(l_1, l_2, l_3)$ has to intersect $\text{span}(S)$. This follows from the maximality of S. Let $S' = \{S \cup (\mathcal{L}_{s1} \setminus \text{Lin}(G))\}$.

  Consider any $l_1 \in \text{Lin}(T_1)$ and $l_1 \notin \text{span}(S')$. Note that there will be at least $(5c_{cand} - 24\mathcal{R}(3)) \log d$ such independent linear forms in T_1.

  If we consider $T_1 + T_2 + T_3 \mod l_1$, it will be of the form $G' \times (T_2' + T_3')$ where $\gcd(T_2', T_3') = 1$. We consider two cases as follows

  Case(a) $l_1 \in \text{Lin}(T_1)$ is such that $l_1 \notin \text{span}(S')$ and $\dim(\text{span}(\text{Lin}(C \mod l_1))) \geq 12\mathcal{R}(3) \log d + 3$. In this case, we will show that l_1 will be in $\mathcal{L}_{cand}$.

  Observe that since $\dim(\mathcal{L}_{s1} \setminus \text{Lin}(G)) \leq 6\mathcal{R}(3) \log d$ (by Lemma 6.4) and by assumption, $\dim(\text{Lin}(G)) \leq 6\mathcal{R}(3) \log d$, thus $\dim(\mathcal{L}_{s1}) \leq 12\mathcal{R}(3) \log d$. It follows that there exist two independent linear forms l and l' dividing G' such that $\mathbb{V}(l_1, l)$ and $\mathbb{V}(l_1, l')$ are not contained within any space in $\mathcal{S}_1(f)$ and moreover f vanishes on them. Hence they lie in $\mathcal{S}_2(f)$, with their kernels intersecting in l_1. Hence all such $l_1 \in \text{Lin}(T_1)$ will be in $\mathcal{L}_{cand}$.

  Case(b) $l_1 \in \text{Lin}(T_1)$ is such that $l_1 \notin \text{span}(S')$ and $\dim(\text{span}(\text{Lin}(C \mod l_1))) \leq 12\mathcal{R}(3) \log d + 3$.

  We will show in most typical cases, any such l_1 will be in $\mathcal{L}_{cand}$. As we are showing this, there will arise one degenerate case where we fail, but then we will show that we can learn enough linear forms from T_2 or T_3.

  Pick any $l_1 \in \text{Lin}(T_1)$ such that $l_1 \notin \text{span}(S')$, $l_3 \in \text{Lin}(T_3')$ and $l_3 \notin \text{span}(S' \cup \{l_1\} \cup \text{Lin}(G'))$ and $l_2 \in \text{Lin}(T_2')$ and $l_2 \notin \text{span}(S' \cup \{l_1, l_3\} \cup \text{Lin}(G'))$. Consider $\mathbb{V}(l_1, l_2, l_3)$. Note that it does not intersect S' and hence is not contained in any space in $\mathcal{S}_1$ and $\mathcal{S}_2^{reg}$. Thus we will learn this space unless it is contained in a space in $\mathcal{S}_2^{sp}$, say $\mathbb{V}(l, l')$. We first observe that $l_1 \notin \text{span}(l, l')$. This because if it was the case then the space $\text{span}(l, l')$ would be of the form $\text{span}(l_1, l_1')$ where $l_1' \in \text{Lin}(G')$. In particular $l_1' \in \text{span}(l_1, l_2, l_3)$ but by choice of l_2 and l_3, this is not possible. Moreover observe that if l_2 and l_3 are also both not in $\text{span}(l, l')$, then we call any such $\mathbb{V}(l_1, l_2, l_3)$ a non-degenerate space for learning l_1. Moreover every such space gets learned as a space in $\mathcal{S}_3^{sp}$(contained in $\overline{\mathcal{S}_3^{sp}}$) or $\mathcal{S}_3$. It is not hard to see (we will formalize below) that enough non-degenerate spaces for learning l_1 will suffice in determining l_1.

  The issue arises in the degenerate case which is when either l_2 or l_3 is contained within $\mathbb{V}(l, l')$. We call any such $\mathbb{V}(l_1, l_2, l_3)$ a degenerate space for learning l_1. We will show that enough degenerate spaces will enable us to learn lots of linear forms from either T_2 or T_3, and hence in this case also we are done.

[5] where a set of spaces is independent if the dimension of the span of their union is a sum of dimensions of the individual spaces

As $\dim(\text{span}(\text{Lin}(C \mod l_1))) \leq 12\mathcal{R}(3)\log d + 3$, we have $\dim(\text{span}(\text{Lin}(T_2'))) \geq (5c_{cand} - 12\mathcal{R}(3))\log d - 3$ and $\dim(\text{span}(\text{Lin}(T_3'))) \geq (2c_{cand} + 24\mathcal{R}(3))\log d + 2$. We pick a set of independent linear forms $L_3 \subseteq \text{Lin}(T_3)$ such that for any $l_3 \in L_3$, $(l_3 \mod l_1) \in \text{Lin}(T_3')$ and $l_3 \notin \text{span}(S' \cup \{l_1\} \cup \text{Lin}(G'))$ and $\dim(\text{span}(L_3)) = 2c_{cand}\log d + 2$. It is not hard to see that such a set exists. Similarly, we pick $L_2 \subseteq \text{Lin}(T_2)$ such that for any $l_2 \in L_2$, $(l_2 \mod l_1) \in \text{Lin}(T_2') \notin \text{span}(S' \cup \{l_1\} \cup L_3 \cup \text{Lin}(G'))$ and $\dim(\text{span}(L_2)) = 2c_{cand}\log d + 2$. Again it is not hard to see that such a set exists. In fact there could have been as many as $3c_{cand} - 36\mathcal{R}(3))\log d - 3$ such linear forms since T_2 by assumption starts off being significantly high rank.

Now, consider the set of spaces $\mathbb{S}(l_1)$ of the form $\mathbb{V}(l_1, l_2, l_3)$ where $l_2 \in L_2$ and $l_3 \in L_3$. Assume there is $l_2 \in L_2$ such that there are two linear forms l_3, l_3' in L_3 for which $\mathbb{V}(l_1, l_2, l_3)$ and $\mathbb{V}(l_1, l_2, l_3')$ are non-degenerate. Then for any $l_2' \in L_2$, if there is a $l_3'' \in L_3 (l_3''$ maybe equal to $l_3, l_3')$ such that $\mathbb{V}(l_1, l_2', l_3'')$ is non-degenerate, then by the intersection of the kernels of these non-degenerate spaces (which we observed we can learn) we have $l_1 \in \mathcal{L}_{cand}$. Also, if we are in the case where we have $l_2, l_2', l_2'' \in L_2$ and $l_3, l_3', l_3'' \in L_3$ such that all three of $\mathbb{V}(l_1, l_2, l_3), \mathbb{V}(l_1, l_2', l_3'), \mathbb{V}(l_1, l_2'', l_3'')$ are non-degenerate, then we again have $l_1 \in \mathcal{L}_{cand}$ by considering the intersections of kernels of these spaces.

So, there are only two cases where we did not manage to deduce that $l_1 \in \mathcal{L}_{cand}$. Either there is a linear form in $l_2 \in L_2$ such that for any other $l_2' \in L_2$, and for any $l_3 \in L_3$, $\mathbb{V}(l_1, l_2', l_3)$ is degenerate. Else, there are only two distinct spaces in $\mathbb{S}(l_1)$ of the form $\mathbb{V}(l_1, l_2, l_3)$ and $\mathbb{V}(l_1, l_2', l_3')$ with $l_2 \neq l_2'$ and $l_3 \neq l_3'$ which are non-degenerate.

In both cases, notice we have at least $2c_{cand}\log d$ independent linear forms from each of L_2, L_3 (call these sets L_2' and L_3') for which all the corresponding $4c_{cand}^2\log^2 d$ spaces are in the degenerate setting. By definition of degeneracy any such degenerate space $\mathbb{V}(l_1, l_2, l_3)$ is contained in an $\mathcal{S}_2^{sp}$ space of the form $\mathbb{V}(l, l')$ where either l_2 or l_3 is contained in $\text{span}(l, l')$. Recall that we have learnt all these $\mathcal{S}_2^{sp}$ spaces. Moreover all the $4c_{cand}^2\log^2 d$ $\mathcal{S}_2^{sp}$ spaces are distinct by choice of independence of linear forms that went into L_2 and L_3. To each such $\mathcal{S}_2^{sp}$ space, we can associate it with a choice of $l_2 \in L_2'$ or $l_3 \in L_3'$ that is contained in its kernel. If there are two distinct $\mathcal{S}_2^{sp}$ spaces that are associated with the same $l_2 \in L_2'$ or $l_3 \in L_3'$ then that choice of l_2 or l_3 will be learned in $\mathcal{L}_{cand}$. Since each choice of l_2 or l_3 can be associated with at most $2c_{cand}\log d$ of the $\mathcal{S}_2^{sp}$ spaces, thus by a simple averaging argument there are at least $2c_{cand}\log d$ independent linear forms from the union of L_2' and L_3' which are each associated with at least two distinct $\mathcal{S}_2^{sp}$ spaces and hence are in $\mathcal{L}_{cand}$. This means from at least one of $\text{Lin}(T_2), \text{Lin}(T_3)$ there are at least $c_{cand}\log d$ linear forms in $\mathcal{L}_{cand}$.

The running time of Algorithm 6, is $\text{poly}(n, d)$ as we have already seen that $|\mathcal{S}_2(f)|$, $|\mathcal{S}_3(f)|$, $|\overline{\mathcal{S}_3^{sp}}(f)|$ are all $d^{O(1)}$ in Lemma 4.3, Lemma 4.4 and Lemma 6.8. Therefore their intersections can be computed in $\text{poly}(n, d)$ time and the list of candidate linear forms is also $d^{O(1)}$. We also have seen how to compute $\mathcal{S}_2(f)$, $\mathcal{S}_3(f)$, and $\overline{\mathcal{S}_3^{sp}}$ in $\text{poly}(n, d)$ time in Lemma 5.1, Lemma 5.2 and Lemma 6.8.

- **Case 2:** $\exists i \in [3], \dim(\text{span}(\text{Lin}(T_i))) < (2c_{cand} + 36\mathcal{R}(3))\log d + 5$

  Wlog, let T_3 be the term such that $\dim(\text{span}(\text{Lin}(T_3))) \leq (2c_{cand} + 36\mathcal{R}(3))\log d + 5$. The $\mathcal{L}_{cand}$ for this case will be the output of Algorithm 7 which adds some more linear forms to the output of Algorithm 6. In this case, we first argue that the output of Algorithm 6, will have either $c_{cand}\log d$ independent linear forms from one of $\text{Lin}(T_1)$ and $\text{Lin}(T_2)$, or will have at least two independent linear forms l_3, l_3' from $\text{Lin}(T_3) \setminus \mathcal{L}_{s1}$. We then argue that we can reconstruct the circuit mod l_3 and l_3' using Theorem 3.10. These circuits that we reconstruct

may not look exactly like $C \mod l_3$, but using rank bound results in Theorem 3.4, the circuits we learn with be "close" to $C \mod l_3$ and hence will give us almost all linear forms from $T_1 \mod l_3$ as well as $T_1 \mod l_3'$. We can then consider gluings of these "projections" to obtain true linear forms from T_1. We include this set of possible gluings into $\mathcal{L}_{cand}$ and argue that we have got at least $c_{cand} \log d$ true linear forms from T_1.

Recall that by assumption $\dim(\text{span}(\text{Lin}(T_1)))$ and $\dim(\text{span}(\text{Lin}(T_2)))$ are both at least $5c_{cand} \log d$ and $\text{rank}(T_1 + T_2 + T_3) \geq 15c_{cand} \log d$, which means $\text{rank}(T_1 + T_2 + T_3) - \dim(\text{span}(\text{Lin}(T_3))) > \mathcal{R}(3) + 3$. From Lemma 6.5, we have that the only elements in $\mathcal{L}_{s1} \setminus \text{Lin}(G)$ will be linear forms that divide at least 1 gate, while $\mathcal{S}_2(f)$ will have spaces on which at least one of the T_i's vanish. If $l|T_1$ (or $l|T_2$), then $l|(T_2 + T_3)$, which is not possible as there is a rank difference between T_2 and T_3. Therefore, all $l \in \mathcal{L}_{s1} \setminus \text{Lin}(G)$ are such that $l|T_3$ and $l|\text{sim}(T_1 + T_2)$. Consider any $\mathcal{S}_2^{reg}$ space such that only T_1 vanishes over it. This means a linear form $l|T_1$ vanishes over this space, which means there is a linear form that divides $\text{sim}(T_2 + T_3) \mod l$ but this cannot happen as there is a difference between the rank of linear forms in the gates. Similarly, we can argue there is no space in $\mathcal{S}_2^{reg}$ such that only T_2 vanishes on it. This means all the spaces in $\mathcal{S}_2^{reg}(f)$ are such that T_3 and $T_1 + T_2$ vanish over it. So, the spaces left in $\mathcal{S}_2$ are those there either all three gates vanish over them or T_3 and $T_1 + T_2$ vanish over them.

The set of linear forms that divide $\text{sim}(T_1 + T_2)$ lie in a $6\mathcal{R}(3) \log d$ dimensional space using Lemma 6.4. Let this space be S. We define $S' = S \cup (\mathcal{L}_{s1} \setminus \text{Lin}(G))$.

Consider any $l_3 \in \text{Lin}(T_3) \setminus \mathcal{L}_{s1}$. We will show that either l_3 will be in $\mathcal{L}_{cand}$ which is the output of Algorithm 6 or else we will manage to find $c_{cand} \log d$ independent linear forms from either T_1 or T_2 that lie in $\mathcal{L}_{cand}$.

Fix any $l_3 \in \text{Lin}(T_3) \setminus \mathcal{L}_{s1}$ and look at $C \mod l_3$. Observe that $C \mod l_3$ will be nonzero and of the form $G' \cdot (T_1' + T_2')$. If $\dim(\text{span}(\text{Lin}(G') \setminus \mathcal{L}_{s1})) \geq 2$ then we have l_3 in kernel of at least two $\mathcal{S}_2$ spaces and hence will be in $\mathcal{L}_{cand}$ which is the output of Algorithm 6 and we are done.

Now suppose $\dim(\text{span}(\text{Lin}(G') \setminus \mathcal{L}_{s1})) \leq 1$.

We pick a set of independent linear forms $L_1 \subseteq \text{Lin}(T_1)$ such that $\dim(\text{span}(L_1)) = 2c_{cand} \log d + 2$ and for any $l_1 \in L_1$ we have $l_1 \notin \text{span}(S' \cup \{l_3\})$ and $(l_1 \mod l_3) \in T_1'$. Clearly such a set L_1 can be found. Similarly, we pick $L_2 \subseteq \text{Lin}(T_2)$ such that $\dim(\text{span}(L_2)) = 2c_{cand} \log d + 2$ and for any $l_2 \in L_2$ we have $(l_2 \mod l_3) \in \text{Lin}(T_2')$ and $l_2 \notin \text{span}(S' \cup \{l_3\} \cup L_1)$. Such an L_2 can be found (in fact even larger such sets can be found) because $\text{Lin}(T_2)$ contains at least $(3c_{cand} - 12\mathcal{R}(3)) \log d - 3$ independent linear forms after removing linear forms in $\text{span}(S' \cup \{l_3\} \cup L_1)$.

Now the argument proceeds almost identically to the argument in Case 1.b (with $l_3 \in \text{Lin}(T_3)$ playing the role of $l_1 \in \text{Lin}(T_1)$). By an identical argument we see that we have either enough non-degenerate spaces of form $\mathbb{V}(l_1, l_2, l_3)$ with $(l_1 \in L_1, l_2 \in L_2)$ such that we deduce $l_3 \in \mathcal{L}_{cand}$ from the intersection of kernels of $\mathcal{S}_3$ spaces, or most spaces are degenerate and we deduce that from one of L_1, L_2, we can find at least $c_{cand} \log d$ independent linear forms in $\mathcal{L}_{cand}$ (that arise from the intersection of kernels of $\mathcal{S}_2$ spaces).

Observe that since $\dim(\text{span}(\text{Lin}(T_3) \setminus \mathcal{L}_{s1})) \geq 2$, by the above argument, we would have learnt two distinct l_3 and l_3' in $\mathcal{L}_{cand} \cap (\text{Lin}(T_3) \setminus \mathcal{L}_{s1})$, or we would have learnt $c_{cand} \log d$ independent linear forms in $\mathcal{L}_{cand}$ from one of T_1 or T_2.

So we are left with the case where we have learnt at least two independent linear forms l_3, l_3'

in $\mathcal{L}_{cand} \cap (\text{Lin}(T_3) \setminus \mathcal{L}_{s1})$. We run the reconstruction algorithm of Theorem 3.10 to obtain a (possibly different) representation of the circuits $C \mod l_3$ and $C \mod l'_3$ if $\text{rank}(\text{sim}(C \mod l_3)) \geq \mathcal{R}(4)+1$ and $\text{rank}(\text{sim}(C \mod l'_3)) \geq \mathcal{R}(4)+1$. In case, the circuits are rank lower than $\mathcal{R}(4)+1$, we learn projections of the gates by just factoring $C \mod l_3$ and $C \mod l'_3$ using Lemma 3.7 giving us $\gcd(T_1, T_2) \bmod l_3, l'_3$ and will have rank at least $5c_{cand} \log d - \mathcal{R}(4)$ and the gluing will give us $c_{cand} \log d$ linear forms as required. In case, the circuits are high simple rank, the learned circuits will still be $\Sigma\Pi\Sigma(2)$ circuits, but as we will now observe, even a distinct $\Sigma\Pi\Sigma(2)$ representation of the same polynomial reveals enough information about the original representation. Let the original representation of $C \mod l_3$ be of the form $G' \times (T'_1 + T'_2)$ where $\gcd(T'_1, T'_2) = 1$. Then note that $(G \times T_1) \mod l_3 = G' \times T'_1$ and $(G \times T_2) \mod l_3 = G' \times T'_2$. The circuit reconstructed by Theorem 3.10 will be of the form $G'' \times (T''_1 + T''_2)$. As $G' \times (T'_1 + T'_2) - G'' \times (T''_1 + T''_2) = 0$, from Theorem 3.4, we have $\text{rank}(\text{sim}(G' \times (T'_1 + T'_2) - G'' \times (T''_1 + T''_2))) \leq \mathcal{R}(4)$ and hence either $\text{rank}(\text{sim}(G' \times T'_1 - G'' \times T''_1)) \leq \mathcal{R}(4)$ or $\text{rank}(\text{sim}(G' \times T'_1 - G'' \times T''_2)) \leq \mathcal{R}(4)$. Wlog, assume this happens with T''_1, i.e. $\text{rank}(\text{sim}(G' \times T'_1 - G'' \times T''_1)) \leq \mathcal{R}(4)$. Thus among the linear forms that appear in $G'' \times T''_1$, all except the linear forms that lie in a $\mathcal{R}(4)$-dimensional space must appear as linear forms in $(G \times T_1) \mod l_3$. Therefore, we have learned the projection mod l_3 of at least $5c_{cand} \log d - \mathcal{R}(4)$ independent linear forms from T_1 [6] and similarly there are $5c_{cand} \log d - \mathcal{R}(4)$ independent linear forms from T_1 for which we know the projection mod l'_3. We then "glue" these projections (see Algorithm 7 for a description of the gluing) to obtain at least $c_{cand} \log d$ independent linear forms from T_1 in $\mathcal{L}_{cand}$ output in Algorithm 7.

As described earlier, the output of Algorithm 6 can be computed in $\text{poly}(n, d)$ time and is $d^{O(1)}$-sized list of linear forms. In Algorithm 7, the loop on Line 3 iterates over these $d^{O(1)}$ possibilities while the reconstruction of $\Sigma\Pi\Sigma(2)$ circuits is done by Theorem 3.10 in $\text{poly}(n, d)$ time. The number of linear forms added to $proj$ is also $O(d)$ and hence $|proj| = d^{O(1)}$ as well. Therefore, Algorithm 7 runs in $\text{poly}(n, d)$-time and outputs a list $\mathcal{L}_{cand}$ of size $d^{O(1)}$.

□

[6] we actually learn a superset of these linear forms, but that suffices for us.

Algorithm 7 Computing Candidate Linear forms when 1 low rank term

Input: Blackbox access to circuit C of form $\Sigma\Pi\Sigma(3)$ computing polynomial $f \in \mathbb{F}[x_1, \ldots, x_n]$ with 1 term of low rank

function $cand - \mathcal{L}(f)(C)$
  Use Algorithm 6 to find $\mathcal{L}_{cand}(f)$, set $proj := \phi$
  for $l \leftarrow \mathcal{L}_{cand}(f)$ **do**
    Consider the circuit $C \mod l$, and use reconstruction algorithm for fan-in 2 gate in Theorem 3.10.
    Move to the next step if the algorithm outputs a circuit of form $G' \times (T_1' + T_2')$, else move to the next l
    For $l_i \in \text{Lin}(G') \cup \text{Lin}(T_1') \cup \text{Lin}(T_2')$, add (l_i, l) to $proj$.
  for $(l_a, l_1), (l_b, l_2) \in proj$ **do**
    if $l_a \mod l_2 = l_b \mod l_1$ **then**
      $l := l_a \mod l_2$
      Find α, β such that $l + \alpha l_1 = l_b$ and $l + \beta l_2 = l_a$
      Add $l + \alpha l_1 + \beta l_2$ to $\mathcal{L}_{cand}$
  Output $\mathcal{L}_{cand}$

6.2.2 Obtaining candidate linear forms when $\dim(\text{span}(\text{Lin}(T_3) \setminus \mathcal{L}_{s1})) < 2$

In this case, we will be handling the case when $\dim(\text{span}(\text{Lin}(T_3) \setminus \mathcal{L}_{s1})) \leq 1$, while the other two gates are high rank, i.e. $\dim(\text{span}(\text{Lin}(T_1))) \geq 5c_{cand} \log d$ and $\dim(\text{span}(\text{Lin}(T_2))) \geq 5c_{cand} \log d$. Observe that this is the only situation left to cover when there are two high rank gates.

Any linear form in $\text{Lin}(T_3) \cap (\mathcal{L}_{s1} \setminus \text{Lin}(G))$ will also divide $\text{sim}(T_1 + T_2)$. Using Lemma 6.4, one can see that $\dim(\text{span}(\text{Lin}(\text{sim}(T_1 + T_2)))) \leq 6\mathcal{R}(3) \log d$ which means that this case happens only when $\dim(\text{span}(\text{Lin}(T_3)) \leq 6\mathcal{R}(3) \log d + 1$. Below, we will prove a more general result as we do not use the condition that there doesn't exist any l such that $\text{rank}(\text{sim}(C \mod l)) \geq c_2$.

We will break the analysis of this case into two parts. First we will consider the case where $\dim(\text{span}(\text{Lin}(T_3))) < c$ for some constant c. In this case we will show something quite precise and strong about the linear forms that we can learn (see lemma statement below). Essentially we will learn almost all linear forms from the high rank gate (T_1).The other is is case when $c \leq \dim(\text{span}(T_3)) \leq 6\mathcal{R}(3) \log d + 1$. We will reduce the learning of linear forms in this case to learning multiple instances of the case when $\dim(\text{span}(\text{Lin}(T_3))) < c$ by taking suitable projections and then showing that we can glue the linear forms obtained in these cases to get back linear forms in T_1.

Lemma 6.11. *Let $\mathcal{R}(k)$ be as defined in Theorem 3.4. For a polynomial f computed by a $\Sigma\Pi\Sigma(3)$ circuit $C = G \times (T_1 + T_2 + T_3)$ satisfying conditions of Theorem 6.2 and $\dim(\text{span}(\text{Lin}(T_1))) \geq (5c_{cand} - 12\mathcal{R}(3)) \log d$ and $\dim(\text{span}(\text{Lin}(T_2))) \geq (5c_{cand} - 12\mathcal{R}(3)) \log d$ and $\dim(\text{span}(\text{Lin}(G \times T_3))) < c$, for $c = \mathcal{R}(3)$, then there exists an algorithm(Algorithm 8) that runs in time $\text{poly}(n, d)$ (assuming $\mathcal{R}(3)$ is constant) and computes a list of linear forms $\mathcal{L}_{cand}$ of size $d^{O(1)}$ such that the following hold:*

1. $\text{Lin}(\gcd(T_1, T_2)) \subseteq \mathcal{L}_{cand}$
2. *If* $\dim(\text{span}(\gcd(T_1, T_2))) < 5c_{cand} \log d - 18\mathcal{R}(3) \log d - c_2$ *(c_2 as defined in Lemma 4.3), then* $\text{Lin}(T_1) \subseteq \mathcal{L}_{cand}$

Proof. Note that in this lemma, we did not need to make the additional assumption that $\dim(\text{span}(\text{Lin}(T_3) \setminus \mathcal{L}_{s1})) \leq 1$ since our proof holds in the more general setting. However when we

apply it to prove Lemma 6.13 we will only care about the setting where the conditional holds that $\dim(\text{span}(\text{Lin}(T_3) \setminus \mathcal{L}_{s1})) \leq 1$. However, what makes this proof hard is precisely then the condition does hold, i.e. all except at most 1 linear form in T_3, also divide $T_1 + T_2$ and therefore, basically all codimension 2 and 3 spaces on which T_3 vanishes are contained in $\mathcal{S}_1$ spaces and hence are not learned by our algorithm.

The key observation is that there are many 2-dimensional spaces (and possibly also 1-dimensional spaces) spanned by linear forms in T_1 and T_2 such that when we go mod those spaces, T_1 and T_2 vanish, but T_3 doesn't and the whole circuit restricts to one that is constant dimensional. In particular, the resulting polynomial has only c essential variables (see Definition 6). Moreover we show that by combining techniques from Section 5 and Theorem 3.11, these 1-dim and most of 2-dim spaces can actually be learned. Once we have our hands on these spaces, we then use them to learn enough linear forms from T_1 and T_2.

We first observe that any codimension 1 or 2 space such that restricted to these the circuit C has at most c essential variables is an $\mathcal{S}_1$ or $\mathcal{S}_2$ space for $T_1 + T_2$. This follows immediately from parts 2 and 3 of Lemma 6.9, since if $T_1 + T_2$ didn't vanish, then it would have a large number of essential variables, and then adding on T_3 would keep the essential variables being high as T_3 only has low number of essential variables. .

We will show that these $\mathcal{S}_1$ and s_2 spaces can be efficiently learned. To learn these spaces, we will combine techniques from Section 5 and Theorem 3.11. In Lemma 5.1 from Section 5, we compute $\mathcal{S}_2$ spaces for polynomials of the form $T_1 + T_2 + T_3$ when $\text{rank}(T_1 + T_2 + T_3) \geq c_2$. In Lemma 5.1, to find the $\mathcal{S}_2$ spaces, we do a random projection to constantly many variables, reconstruct to get access to the monomial form of the projections, find the $\mathcal{S}_2$ spaces $\mathbb{V}(l_1, l_2)$ of the projected polynomial by solving a certain system of polynomial equations. To obtain this system, we treat the coefficients of the linear forms of l_1 and l_2 as variables. Since the projected polynomial must equal zero once $l_1 = l_2 = 0$, thus all its coefficients must become 0. Thus we obtain the system of equations by equating the coefficients of the projected polynomial to 0 after the substitution of $l_1 = l_2 = 0$. Once we learn the $\mathcal{S}_2$ spaces of the projected polynomial, we then glue them back together to get $\mathcal{S}_2$ spaces for the original circuit. In our current setup, the high level plan is the same. The problem in the current case is that when we try to find $\mathcal{S}_2(T_1 + T_2)$ is we do not have access to $T_1 + T_2$. However as we observed, in order to compute $\mathcal{S}_2(T_1 + T_2)$, it suffices to find codimension 2 spaces over which the polynomial restricts to few essential variables. We use this observation to come up with a different system of polynomial equations. In Lemma 3.12, we see that the rank of the partial derivative matrix of a polynomial f is equal to the number of essential variables of f. Thus the condition that the number of essential variables is at most c is equivalent to the condition that all the $c+1 \times c+1$ minors in the partial derivative matrix have determinant 0. We do have access to the entries of the partial derivative matrix of the projected polynomials, and hence can set up a system of equations by equating the determinants of the minors to 0. The rest of the idea is similar to Algorithm 2.

The algorithm for learning the $\mathcal{S}_1$ spaces of $T_1 + T_2$ is identical and just a simplification of the algorithm for learning the $\mathcal{S}_2$ spaces. Below we elaborate more on the details of both these algorithms.

Learning the $\mathcal{S}_1$ spaces of T_1+T_2 Similar to Algorithm 2, we take a random linear isomorphism Φ defined by $\Phi(x_i) = \sum_{j=1}^{n} \alpha_{i,j} x_j$(where $\alpha_{i,j}$ are chosen randomly from $[d^n]$) to get polynomial $g = \Phi(f) = f(\Phi(x))$. We then obtain polynomials g_i for $i \in [10c+1, n]$, by setting all but the first $10c$ variables and x_i to 0 in C. We can interpolate these (since they are only constant variate) to get white-box access to the monomial representation of the g_i's. Let $\Phi(C)|_{x_{10c+1}=\ldots=x_{i-1}=x_{i+1}=\ldots=x_n=0} =$

$G^{[i]} \times (T_1^{[i]} + T_2^{[i]} + T_3^{[i]})$. As $\dim(\text{span}(\text{Lin}(T_1)))$ and $\dim(\text{span}(\text{Lin}(T_2)))$ were $\Omega(\log d)$ (which is clearly more than $10c$), after projecting down the gates would still have rank at least $10c$ with high probability, similar to Lemma 5.1, while projected T_3 will have at most c essential variables. Now, any l such that $C \mod l$ has at most c essential variables must be such that $T_1 + T_2 \mod l$ has at most $2c$ essential variables. From part 2 of Lemma 6.9, any l such that $T_1 + T_2 \mod l$ has at most $2c$ essential variables must be such that $T_1 + T_2 \mod l = 0$ (Since we know that both T_1 and T_2 have rank at least $10c$ and c is large enough). Thus to learn $\mathcal{S}_1(T_1 + T_2)$, it suffices to learn the codimension 1 spaces on which $G \times (T_1 + T_2 + T_3)$ has at most c essential variables. We substitute $x_1 = \alpha_2 x_2 + \ldots + \alpha_{10c} x_{10c} + \alpha_i x_i$ into g_i which we have monomial access to. As seen in Lemma 3.12, the number of essential variables will be the rank of the partial derivative matrix. Using white-box access to the g_i's, we can get access to the partial derivative matrix and hence get a system of polynomial equations in $\alpha_2, \ldots, \alpha_{10c}, \alpha_i$ by equating all $(c+1) \times (c+1)$ minors of the matrix to 0, ensuring rank of the matrix is at most c. This ensures $C \mod l$ has at most c essential variables. The system of equations is in $10c$ variables and has $\text{poly}(d)$ equations (if c is constant) with a degree at most $c+1$ and hence can be solved in $\text{poly}(d)$ time using Theorem 3.8. We can then glue these solutions for g_i, similar to Lemma 2, by comparing coefficients in $x_2, \ldots, x_{10c}$, to get linear forms in $\text{Lin}(T_1 + T_2)$. From part 2 of Lemma 5.4, we have $\text{Lin}(T_1' + T_2') = \text{Lin}(\Phi(T_1) + \Phi(T_2))|_{x_{10c+1}=\ldots=x_{i-1}=x_{i+1}=\ldots=x_n=0}$. Therefore each $\mathcal{L}_i$, will contain linear forms from $\text{Lin}(T_1 + T_2)$ after Φ and $x_{10c+1} = \ldots = x_{i-1} = x_{i+1} = \ldots = x_n = 0$. We can then glue these projections naturally to get linear forms in $\text{Lin}(T_1 + T_2)$. Since we learn all linear forms in $\text{Lin}(T_1 + T_2)$, in particular, we learn all linear forms in $\text{Lin}(\gcd(T_1, T_2))$. Therefore, we are done with the requirement of the first condition of the current lemma.

Learning $\mathcal{S}_2$ spaces of $T_1 + T_2$ Recall that we only need to learn the $\mathcal{S}_2$ spaces when $\dim(\text{span}(\gcd(T_1, T_2))) < 5c_{cand} \log d - 18\mathcal{R}(3) \log d - c_2$. Otherwise, the $\mathcal{S}_1$ spaces suffice for the first part of the lemma. In Algorithm 8, we only find $\mathcal{S}_2$ spaces if the linear forms we found from $\mathcal{S}_1$ spaces lie in a space with dimension at most $5c_{cand} \log d - 12\mathcal{R}(3) \log d - c_2$, but since from Lemma 6.4, we have $\dim(\text{span}(\mathcal{L}_{s1} \setminus \text{Lin}(G))) \leq 6\mathcal{R}(3)$, means this case contains all instances of the case with $\dim(\text{span}(\gcd(T_1 + T_2))) < 5c_{cand} \log d - 18\mathcal{R}(3) \log d - c_2$.

Let $T_1 + T_2 = \gcd(T_1, T_2) \cdot (T_1' + T_2')$. We will find codimension 2 spaces on which C has at most c essential variables, and these codimension 2 spaces are not contained in codimension 1 (i.e. $\mathcal{S}_1$) spaces found earlier. Much of the algorithm is similar to what was done to find the $\mathcal{S}_1$ spaces. We substitute $x_1 = \alpha_3 x_3 + \ldots + \alpha_{10c} x_{10c} + \alpha_i x_i$ and $x_2 = \beta_3 x_3 + \ldots + \beta_{10c} x_{10c} + \beta_i x_i$ into g_i, and obtain the partial derivative matrix. Using it, we create a system of equations by setting the determinants of all $(c+1) \times (c+1)$ minors to zero, thus ensuring at most c essential variables. We ensure that the pair of linear forms obtained as solutions should form a space of dimension at least 3 with linear forms found in the previous step (i.e. the kernels of the $\mathcal{S}_1$ spaces), using the same approach as in Lemma 5.3. Since $\gcd(T_1, T_2) < 5c_{cand} \log d - 18\mathcal{R}(3) \log d - c_2$, we have $\text{rank}(\text{sim}(T_1 + T_2)) \geq 6\mathcal{R}(3) \log d + c_2 \geq 10c + 1$. Similar to Lemma 5.4, the projection will not create new linear factors, and therefore in the projected version, $\text{rank}(\text{sim}((T_1 + T_2)|_{x_{10c+1}=\ldots=x_{i-1}=x_{i+1}=\ldots=x_n})) = 10c + 1$ with high probability. As we have $10c + 1 > c_2$, from Lemma 4.3, the restriction of $T_1 + T_2$ has only $d^{O(1)}$ $\mathcal{S}_2$ spaces. Thus, we can find all these spaces for each g_i, and then glue them together similar to Algorithm 2. Similar to part 4 of Lemma 5.4, one can use a map Ψ such that for all $x_i \in [10c + 1, n]$ $\Psi(x_i) = x_i + \sum_{j=3}^{10c} \beta_{i,j} x_j$(where $\beta_{i,j}$ are chosen randomly from $[d^n]$) to show that with high probability, that distinct spaces remain distinct when you set x_i to 0. Therefore, each $\mathbb{V}(l_{1i}, l_{2i}) \in \mathcal{S}_2(g_i)$ map to unique $\mathbb{V}(l_{1j}, l_{2j}) \in \mathcal{S}_2(g_j)$ such that $\langle \Psi(l_{1i}), \Psi(l_{2i}) \rangle |_{x_i=0} = \langle \Psi(l_{1j}), \Psi(l_{2j}) \rangle |_{x_j=0}$ and all of which can be glued to get $\mathcal{S}_2$ spaces for $T_1 + T_2$. Thus, we have the set $\mathcal{S}_2(T_1 + T_2)$.

Using $\mathcal{S}_2(T_1+T_2)$ to learn linear forms from T_1: Using $\mathcal{S}_2(T_1+T_2)$, we form the list of candidate linear forms by including the linear forms that are the intersection of kernels of the spaces in $\mathcal{S}_2$. We now show that this list of candidate linear forms already contains all linear forms in T_1. Consider any $l_1 \in T_1$ and $l_1 \notin \gcd(T_1, T_2)$ (since the gcd was already learnt). The polynomial computed by $T_1 + T_2 \mod l_1 = T_2 \mod l_1$ will be a non-zero polynomial which is a product of linear forms. By Lemma 6.4, recall that the linear forms in $\mathrm{Lin}(\mathrm{sim}(T_1+T_2))$ lie in a $6\mathcal{R}(3)\log d$ dimensional space. Since, $\dim(\mathrm{span}(\mathrm{Lin}(\gcd(T_1,T_2)))) < 5c_{cand}\log d - 18\mathcal{R}(3)\log d - c_2$, we have $\dim(\mathrm{span}(\mathrm{Lin}(T_1+T_2))) < (5c_{cand} - 12\mathcal{R}(3))\log d - c_2$. From the assumption in the lemma, we have $\dim(\mathrm{span}(\mathrm{Lin}(T_2))) \geq (5c_{cand} - 12\mathcal{R}(3))\log d$. Therefore, there we can find two independent linear forms l_2, l_2' in T_2, such that $l_2, l_2' \notin \mathrm{span}(\{l_1\} \cup \mathrm{Lin}(T_1+T_2))$. This means there is no linear form $l \in \mathrm{Lin}(T_1+T_2)$ such that $l \in \mathrm{span}(l_1, l_2)$ or $l \in \mathrm{span}(l_1, l_2')$. Therefore, $\mathbb{V}(l_1, l_2)$ and $\mathbb{V}(l_1, l_2')$ are in $\mathcal{S}_2(T_1+T_2)$. Thus, for any $l_1 \in \mathrm{Lin}(T_1)$ and $l_1 \notin \gcd(T_1,T_2)$, we have $l_1 \in \mathcal{L}_{cand}$ from the intersection of kernels of spaces in $\mathcal{S}_2(T_1+T_2)$. We already showed $\gcd(T_1,T_2) \subseteq \mathcal{L}_{cand}$ and therefore, $\mathrm{Lin}(T_1) \subseteq \mathcal{L}_{cand}$.

Analysis of runtime: The system of equations we get in Line 9 will have $d^{O(c^c)}$ polynomial equations of degree $c+1$ in variables $O(c)$, and as $c = O(1)$, we can obtain the solutions to the system in $\mathrm{poly}(d)$ time. As described earlier each solution must be part of $\mathrm{Lin}(T_1+T_2)$ projected down to $c+1$ variables and therefore there can be at most $O(d)$ solutions for each g_i. Thus, the gluing step can also be done in $\mathrm{poly}(n,d)$ time, similar to Lemma 5.3. Algorithm 8 ensures that when we find the $\mathcal{S}_2$ spaces, the linear forms we found from codimension 1 spaces lie in a space with dimension less than $(5c_{cand} - 12\mathcal{R}(3))\log d - c_2$. As discussed above, the set of linear forms learnt $\mathcal{L}' = \mathrm{Lin}(\Phi(T_1+T_2))$, and $\gcd(\Phi(T_1), \Phi(T_2)) \subseteq \mathrm{Lin}(\Phi(T_1+T_2)$. Therefore, if $\dim(\mathrm{span}(\mathcal{L}')) < (5c_{cand} - 12\mathcal{R}(3))\log d - c_2$, then $\dim(\mathrm{span}(\mathrm{Lin}(\gcd(\Phi(T_1), \Phi(T_2))))) < (5c_{cand} - 12\mathcal{R}(3))\log d - c_2$ and $\mathrm{rank}(\mathrm{sim}(\Phi(T_1+T_2))) \geq c_2$. Therefore, the number of $\mathcal{S}_2$ spaces will be $d^{O(1)}$ as discussed in Lemma 4.3. Since, $\mathcal{L}_i = \mathrm{Lin}(\Phi(T_1+T_2)|_{x_{10c+1}=\ldots=x_{i-1}=x_{i+1}=\ldots=x_n=0})$, $|\mathcal{L}_i| \leq d$. Therefore, the system of equations we obtain for codimension 2 spaces, will have additional $O(d)$ equations similar to Algorithm 1. Therefore, we can solve the system of $\mathrm{poly}(d)$ polynomial equations with the degree at most $c+1$ over $O(c)$ variables in time $\mathrm{poly}(d)$. As the number of $\mathcal{S}_2$ spaces is $d^{O(1)}$, we can find the intersections of the codimension 2 spaces in time $\mathrm{poly}(n,d)$ as well, and the number of linear forms added to $\mathcal{L}_{cand}$ is also $d^{O(1)}$. □

Remark 6.12. *We observe that in Algorithm 8, if the output for polynomial f is $\mathcal{L}_{cand}$, then for an invertible linear isomorphism Ψ and input $\Psi(f)$, Algorithm 8 outputs $\Psi(\mathcal{L}_i) := \{\Psi(l) : l \in \mathcal{L}_{cand}\}$. To see this we observe that the linear forms in $\mathcal{L}_{cand}$ come from $\mathrm{Lin}(T_1+T_2)$ and intersection of kernels of spaces in $\mathcal{S}_2(T_1+T_2)$. Clearly for any $l \in \mathrm{Lin}(T_1+T_2)$, we have $\Psi(l) \in \mathrm{Lin}(\Psi(T_1)+\Psi(T_2))$. Also, for a space $\mathbb{V}(l_1,l_2) \in \mathcal{S}_2(T_1+T_2)$, we will have $\mathbb{V}(\Psi(l_1),\Psi(l_2)) \in \mathcal{S}_2(\Psi(T_1)+\Psi(T_2))$. This means for every linear form l in the intersections of kernels of $\mathcal{S}_2(T_1+T_2)$ spaces, there will $\Psi(l)$ in the intersections of kernels of $\mathcal{S}_2(\Psi(T_1)+\Psi(T_2))$ spaces.*

Algorithm 8 Computing Candidate Linear forms with two high dimensional gates and $\dim(\text{span}(\text{Lin}(T_3) \setminus \mathcal{L}_{s1})) < 2$ and constant dimension T_3

Input: Blackbox access to circuit C of form $\Sigma\Pi\Sigma(3)$ computing polynomial $f \in \mathbb{F}[x_1, \ldots, x_n]$ of Special form 4 with $\dim(\text{span}(\text{Lin}(T_3))) < c$

function $cand - \mathcal{L}(f)(C)$
  Sample n^2 random values $a_{ij}; i, j \in [n]$ uniformly from $S := \{1, \ldots, d^n\}$, and use them to define n linear forms $l'_i = \sum_{j=1}^{n} a_{ij} x_j$. Check if they are independent, otherwise repeat. Define isomorphism Φ with these linear forms $l'_1, \ldots, l'_n \in \mathbb{F}[x_1, \ldots, x_n]$ mapping $x_i \to l'_i$. Let $g = \Phi(f)$
  Set $t = 10c$. For $i \in \{t, \ldots, n\}$, obtain polynomials $g_i = g_{x_t = \ldots = x_{i-1} = x_{i+1} = \ldots = x_n = 0}$
  for $i \leftarrow \{t, \ldots, n\}$ **do**
    Interpolate to get monomial access to g_i
    Substitute $x_1 = \alpha_2 x_2 + \ldots + \alpha_{t-1} x_{t-1} + \alpha_i x_i$ into the g_i and get monomial access to $\frac{\partial g_i}{\partial x_j}$ for $j \in \{2, \ldots, t-1, i\}$
    Use these to access the partial derivative matrix M
    For each $(c+1) \times (c+1)$ minor A of M add an equation $det(A) = 0$
    Solve the system of equations to get solutions for $\alpha_2, \ldots, \alpha_{t-1}, \alpha_i$
    Add $x_1 - \alpha_2 x_2 - \ldots - \alpha_{t-1} x_{t-1} - \alpha_i x_i$ to $\mathcal{L}_i$
  Glue all linear forms in $\mathcal{L}_i$ together if they are consistent in the first t coordinates to obtain a set of linear forms $\mathcal{L}'$. Add $\Phi^{-1}(l)$ for all $l \in \mathcal{L}'$ to $\mathcal{L}_{cand}$
  if $\dim(\text{span}(\mathcal{L}')) < (5c_{cand} - 12\mathcal{R}(3)) \log d - c_2$ **then**
    for $i \leftarrow \{t, \ldots, n\}$ **do**
      Substitute $x_1 = \alpha_3 x_3 + \ldots + \alpha_{t-1} x_{t-1} + \alpha_i x_i = l_{1i}$ and $x_2 = \beta_3 x_3 + \ldots + \beta_{t-1} x_{t-1} + \beta_i x_i = l_{2i}$ into the g_i and get monomial access to $\frac{\partial g_i}{\partial x_j}$ for $j \in \{3, \ldots, t-1, i\}$
      Use these to access the partial derivative matrix M
      For each $(c+1) \times (c+1)$ minor A of M add an equation $det(A) = 0$
      for $l \in \mathcal{L}_i$ **do**
        Add an equation to the system of equations above which enforces $\dim(\text{span}(l, x_1 - l_{1i}, x_2 - l_{2i})) = 3$ (similar to Algorithm 1)
      Solve the system of equations in $\alpha_3, \ldots, \alpha_{t-1}, \alpha_i, \beta_3, \ldots, \beta_{t-1}, \beta_i$
      Add $\mathbb{V}(x_1 - l_{1i}, x_2 - l_{2i})$ to $\mathcal{S}_{2i}$
    Glue the spaces together if their projections to the first t coordinates is identical, to obtain $\mathcal{S}_2(T_1 + T_2)$
    If spaces $\mathbb{V}(l_1, l_2)$ and $\mathbb{V}(l'_1, l'_2)$ in $\mathcal{S}_2(T_1 + T_2)$ are such that $sp(l_1, l_2) \cap sp(l'_1, l'_2) = sp(l)$, add $\Phi^{-1}(l)$ to $\mathcal{L}_{cand}$
  Output $\mathcal{L}_{cand}$

Now, we are left with the only case where T_3 has a rank greater than constant but less than $6\mathcal{R}(3) \log d$ and $dim(\text{span}(\text{Lin}(T_3) \setminus \mathcal{L}_{s1})) \leq 1$.

Lemma 6.13. *Let $\mathcal{R}(k)$ be as defined in Theorem 3.4. For a polynomial f computed by circuit $C = G \times (T_1 + T_2 + T_3)$ satisfying conditions of Theorem 6.2 and $\dim(\text{span}(\text{Lin}(T_1))) \geq 5c_{cand} \log d$ and $\dim(\text{span}(\text{Lin}(T_2))) \geq 5c_{cand} \log d$, with $\dim(\text{span}(\text{Lin}(T_3) \setminus \mathcal{L}_{s1})) \leq 1$, there exists an algorithm that computes in $\text{poly}(n, d)$ time, assuming $\mathcal{R}(3)$ is constant, a list of linear forms $\mathcal{L}_{cand}$ such that $|\mathcal{L}_{cand}| = d^{O(1)}$ and $\dim(\text{span}(\mathcal{L}_{cand} \cap \text{Lin}(T_1)) \geq c_{cand} \log d$.*

Proof. We will reduce this case to multiple instances of the case when T_3 has rank at most $c = \mathcal{R}(3)$. To do this, we first observe that we can learn all but one of the linear forms (up to multiplicity) of

T_3 as part of the set $\mathcal{L}_{s1}$. This is because we are assuming that $\dim(\text{span}(\text{Lin}(T_3) \setminus \mathcal{L}_{s1})) \leq 1$, and we can compute $\mathcal{L}_{s1}$.

Let S be a basis of the linear forms in $\text{span}(\mathcal{L}_{s1})$. As $\dim(\text{span}(\text{Lin}(G))) \leq 6\mathcal{R}(3)\log d$ and $\dim(\text{span}(\text{Lin}(\mathcal{L}_{s1} \setminus \text{Lin}(G)))) \leq 6\mathcal{R}(3)\log d$, we have $|S| \leq 12\mathcal{R}(3)\log d$. From $\dim(\text{span}(\text{Lin}(T_3) \setminus \mathcal{L}_{s1})) \leq 1$ and Lemma 6.4, we know $\dim(\text{span}(T_3)) \leq 6\mathcal{R}(3)\log d + 1$.

Note that Lemma 6.5 implies that all linear forms in $\mathcal{L}_{s1} \setminus \text{Lin}(G)$ divide at least one gate. If $l \in \mathcal{L}_{s1} \setminus \text{Lin}(G)$ is such that $l|T_1$, then l also divides $T_2 + T_3$ (but doesn't individually divide T_2 or T_3), which is impossible as there is a rank difference in the linear forms of T_2 and T_3. Similarly it cannot be that l divides T_2. Therefore, all linear forms in $\mathcal{L}_{s1} \setminus \text{Lin}(G)$ are in $\text{Lin}(T_3)$.

The plan is to project the variables in the circuit in some random manner so that the linear forms from $G \times T_3$ only span a constant c-dimensional space, but at the same time, T_1 and T_2 continue to be high rank even after the projection. We will use the set S to find such a projection. At this point we can invoke Lemma 6.11 to learn some candidate linear forms from the projected circuit. We do this for several projections and then "glue" the candidate linear forms the we obtain. Note that the linear forms obtained from the projections will contain the projected versions of all the linear forms in $\gcd(T_1, T_2)$, and will contain the projected versions of all the linear in $\text{Lin}(T_1)$ if $\dim(\text{span}(\gcd(T_1 + T_2))) < 5c_{cand}\log d - 18\mathcal{R}(3)\log d - c_2$.

Without loss of generality, let $S := \{x_1, \ldots, x_{|S|}\}$. We can make this assumption for the following reason. Recall that S is a basis of $\text{span}(\mathcal{L}_{s1})$; therefore, we can compute S and consider an appropriate invertible linear transformation which makes $S := \{x_1, \ldots, x_{|S|}\}$. Now, consider a random linear isomorphism Φ such that $\forall\ x_i \in S, \Phi(x_i) = \sum_{j=1}^{|S|} \alpha_{i,j} x_j$ where $\alpha_{i,j}$ are picked randomly from $\{1, \ldots, d^n\}$, and let $g = \Phi(f) = f(\Phi(x))$. Now we consider for each $i \in [c-1, |S|]$ the polynomials g_i, obtained from g by setting $x_{c-1} = \ldots = x_{i-1} = x_{i+1} = \ldots = x_{|S|} = 0$. Observe, that for each g_i, one can view the way g_i is obtained from f as taking random linear subspace of $\text{span}(S)$ and considering the circuit C modulo the subspace. By Lemma 3.2 and similar to the analysis of Lemma 5.4 it follows that with high probability, under this transformation each gate of C remains nonzero and so does $T_1 + T_2$.

Thus for each i, the projected circuit g_i is of the form $T_1^{[i]} + T_2^{[i]} + T_3^{[i]}$ where $T_3^{[i]}$ has at most c essential variables and $T_1^{[i]}, T_2^{[i]}$ each have linear forms spanning a space with dimension at least $(5c_{cand} - 12\mathcal{R}(3))\log d$ (since we set at most $12\mathcal{R}(3)\log d$ variables to 0).

Now, we consider the following two cases on the original circuit: $\dim(\text{span}(\gcd(T_1, T_2))) \geq 5c_{cand}\log d - 18\mathcal{R}(3)\log d - c_2$ and $\dim(\text{span}(\gcd(T_1, T_2))) < 5c_{cand}\log d - 18\mathcal{R}(3)\log d - c_2$.

In the former case, for all g_i's, by Lemma 6.11 we can learn a set of linear forms $\mathcal{L}_i$ which contains $\gcd(T_1^{[i]}, T_2^{[i]})$. We will assume wlog that the coefficient of x_1 in all linear forms will be 1 since we only care about the linear forms up to scaling by a constant. Moreover due to the random map Φ, with high probability, for each $i \in [c-1, |S|]$, the coefficient of x_1 will be non-zero in all $l \in \mathcal{L}_i$. Consider a map Ψ such that for each $i \in [c-1, |S|]$, $\Psi(x_i) = \sum_{j=1}^{c-2} \beta_{i,j} x_j$ for $\beta_{i,j}$ chosen randomly from $[d^n]$. This map will allow us to argue that for each $i \in [c-1, |S|]$, distinct linear forms in $\mathcal{L}_i$ will remain distinct after applying Ψ and then setting x_i to 0. This distinctness holds due to remark 6.12 and a proof similar to part 4 of Lemma 5.4 (where a similar distinctness was proved). Thus we get that with probability $1 - o(1)$, for two distinct $l, l' \in \mathcal{L}_i$, $\Psi(l)|_{x_i=0} \neq \Psi(l')|_{x_i=0}$.

Once we have this distinctness property, we can glue these set of candidate linear forms across the different $\mathcal{L}_i$. To do this we glue $l_i \in \mathcal{L}_i$ with $l_j \in \mathcal{L}_j$ (in the natural way) if $\Psi(l_i)|_{x_i=0} = \Psi(l_j)|_{x_j=0}$.

Note that due to the distinctness property, there is at most one linear form each $l_i \in \mathcal{L}_i$ maps to in other $\mathcal{L}_j$'s and thus the gluing is efficient. Moreover the gluing will recover the linear forms in $\gcd(T_1, T_2)$. Since, $\dim(\text{span}(\gcd(T_1, T_2))) \geq 5c_{cand}\log d - 18\mathcal{R}(3)\log d - c_2$, we learn $c_{cand}\log d$ independent linear forms from T_1.

In the latter case, since $\dim(\text{span}(\gcd(T_1, T_2))) < 5c_{cand} \log d - 18\mathcal{R}(3) \log d - c_2$, after the projection we have $\dim(\text{span}(\gcd(T_1', T_2'))) < 5c_{cand} \log d - 18\mathcal{R}(3) \log d - c_2$ as no two linear forms become equal with high probability. We can learn a set of linear forms that contain the projections of all linear forms in $\text{Lin}(T_1)$ from lemma 6.11. Now, we can glue these set of candidate linear forms of g_i's similar to above case by using Ψ as defined above. We will glue $l_i \in \mathcal{L}_i$ with $l_j \in \mathcal{L}_j$ if $\Psi(l_i)|_{x_i=0} = \Psi(l_j)|_{x_j=0}$. This gives us linear forms from $\text{Lin}(T_1)$. Therefore, in this case we learn the entire $\text{Lin}(T_1)$ which we know from assumption has at least $5c_{cand} \log d$ independent linear forms.

The number of calls to Algorithm 8 will be at most $6\mathcal{R}(3) \log d - c$ and each call runs in time $\text{poly}(n, d)$ and outputs a list of size $d^{O(1)}$. Therefore, the gluing can also be done in time $\text{poly}(n, d)$. Thus, we can compute a $d^{O(1)}$ sized list of linear forms $\mathcal{L}_{cand}$ in $\text{poly}(n, d)$ time such that $\dim(\text{span}(\mathcal{L}_{cand} \cap \text{Lin}(T_1)) \geq c_{cand} \log d$. □

6.3 Candidate Linear forms when There is exactly 1 High Rank Gate

In this case, the circuit is of form $C = G \times (T_1 + T_2 + T_3)$ where $\gcd(T_1, T_2, T_3) = 1$, $\dim(\text{span}(\text{Lin}(T_1))) \geq 5c_{cand} \log d$ and $\dim(\text{span}(\text{Lin}(T_2))), \dim(\text{span}(\text{Lin}(T_3))) < 5c_{cand} \log d$. We also assume the circuit is such that there is no linear form l such that $\text{rank}(\text{sim}(C \mod l)) \leq c_2$ and $C \mod l \neq 0$.

Algorithm 9 Computing Candidate Linear forms when 1 high rank term(Case A)

Input: Blackbox access to circuit C of form $\Sigma\Pi\Sigma(3)$ computing polynomial $f \in \mathbb{F}[x_1, \ldots, x_n]$ with 1 term of high rank

function $cand - \mathcal{L}(f)(C)$
  Use Algorithm 6 to find $\mathcal{L}_{cand}(f)$, set $proj := \phi$
  for $l \leftarrow \mathcal{L}_{cand}(f)$ **do**
    Consider the circuit $C \mod l$, and use reconstruction algorithm for fan-in 2 gate in [Sin16a].
    Move to the next step if the algorithm outputs a circuit of form $G' \times (T_1' + T_2')$, else move to the next l
    For $l_i \in \text{Lin}(G') \cup \text{Lin}(T_1') \cup \text{Lin}(T_2')$, add (l_i, l) to $proj$.
  for $(l_a, l_1), (l_b, l_2) \in proj$ **do**
    if $l_a \mod l_2 = l_b \mod l_1$ **then**
      $l := l_a \mod l_2$
      Find α, β such that $l + \alpha l_1 = l_b$ and $l + \beta l_2 = l_a$
      Add $l + \alpha l_1 + \beta l_2$ to $\mathcal{L}_{cand}$
  Output $\mathcal{L}_{cand}$

Lemma 6.14. *Let f be a polynomial computed by $\Sigma\Pi\Sigma(3)$ circuit $C = G \times (T_1 + T_2 + T_3)$ such that $\gcd(T_1, T_2, T_3) = 1$ and satisfies all properties from Theorem 6.2. Let c_2 be as defined in Lemma 4.3. We assume there doesn't exist any linear form l such that $\text{rank}(\text{sim}(C \mod l)) \leq c_2$ and $C \mod l \neq 0$. Assume also that $\dim(\text{span}(\text{Lin}(T_2))) < 5c_{cand} \log d$ and $\dim(\text{span}(\text{Lin}(T_3))) < 5c_{cand} \log d$. Then, there exists an algorithm that computes a set of linear forms $\mathcal{L}_{cand}$ such that $|\mathcal{L}_{cand}| = d^{O(1)}$ and $\dim(\text{span}(\text{Lin}(T_1) \cap \mathcal{L}_{cand})) \geq c_{cand} \log d$.*

Proof. The idea in this lemma is pretty similar to Case 2 of Lemma 6.10 and Lemma 6.13. The analysis is more involved due to the two gates having small ranks.

From Lemma 6.5, we have that any linear form l in $\mathcal{L}_{s1} \setminus \text{Lin}(G)$ must divide a gate in the circuit. Consider the $l \in \mathcal{L}_{s1} \setminus \text{Lin}(G)$ such that it divides T_2(or T_3), then is must also divide $T_1 + T_3$(or $T_1 + T_2$), which cannot happen as there is a rank gap. Therefore, all linear forms l in

$\mathcal{L}_{s1} \setminus \mathrm{Lin}(G)$ are such that l divides T_1 and $T_2 + T_3$. Now, when we remove the codimension 2 and 3 spaces contained in $\mathcal{S}_1$ spaces, we might also remove all spaces containing linear forms from T_2 and T_3 in which case we don't learn any linear forms.

To handle all such cases, we divide this lemma into 2 major cases, where $\dim(\mathrm{span}(\mathrm{Lin}(T_2) \cup \mathrm{Lin}(T_3) \setminus \mathrm{span}(\mathcal{L}_{s1}))) \geq c_2$ and $\dim(\mathrm{span}(\mathrm{Lin}(T_2) \cup \mathrm{Lin}(T_3) \setminus \mathrm{span}(\mathcal{L}_{s1}))) < c_2$.

Case A: $\dim(\mathrm{span}(\mathrm{Lin}(T_2) \cup \mathrm{Lin}(T_3) \setminus \mathrm{span}(\mathcal{L}_{s1}))) \geq c_2$

In this case, we will first argue that by considering the intersection of the kernels of $\mathcal{S}_2$ and $\mathcal{S}_3(\bar{\mathcal{S}_3})$ spaces, we will either learn at least $c_{cand} \log d$ independent linear forms from T_1, or at least two independent linear forms from one of T_2 or T_3. If we learn two independent linear forms from one of T_2 or T_3 then we can use them to learn $c_{cand} \log d$ independent linear forms from T_1 as follows: we can reconstruct the circuit mod these two linear forms (one at a time), learn the resulting circuit of top fan-in 2 which is close to the original representation, and thus learn projections of enough linear forms in T_1, which we can glue back to get enough true linear forms from T_1 in $\mathcal{L}_{cand}$ output by Algorithm 9. The key difference in this from Case 2 of Lemma 6.10 is there might be two gates that are low rank instead of just one.

We also have from Lemma 6.5, that in this case, for all $\mathcal{S}_2$ spaces, at least one of the T_is must vanish, and hence also the sum of the other two. As the simple part of $T_1 + T_2$ or $T_1 + T_3$ cannot not have any divisors due to the rank difference in the gates, hence all spaces in $\mathcal{S}_2^{reg}$ will be of form $\mathbb{V}(l_1, l)$ such that $l_1 | T_1$ and $l | \mathrm{sim}((T_2 + T_3) \mod l_1)$.

Also, since $\mathrm{rank}(T_1+T_2+T_3) \geq 15 c_{cand} \log d$ and $\dim(\mathrm{span}(T_2)), \dim(\mathrm{span}(T_3)) < 5 c_{cand} \log d$, it follows that there exists a set of independent linear forms $L_1 \subseteq \mathrm{Lin}(T_1)$ such that $|L_1| = 5 c_{cand} \log d$ and for any $l_1 \in L_1$, it holds that $l_1 \notin \mathrm{span}(\mathrm{Lin}(T_2) \cup \mathrm{Lin}(T_3))$.

We will consider three cases. In each case we will first show that we can either learn at least $c_{cand} \log d$ independent linear forms from T_1, or at least two independent linear forms from one of T_2 or T_3.

Case A.1: The first case is when $\dim(\mathrm{span}(\mathrm{Lin}(T_2 + T_3) \setminus \mathrm{span}(\mathcal{L}_{s1}))) \geq 2$. This case is easy. Any space corresponding to a linear form in L_1 and any linear form in $\mathrm{Lin}(T_2 + T_3) \setminus \mathrm{span}(\mathcal{L}_{s1})$, will be learned in the set of $\mathcal{S}_2$ spaces as they are not contained in any $\mathcal{S}_1$ space. This we have for all $l_1 \in L_1$, at least two $\mathcal{S}_2$ spaces whose kernels contain l_1, and therefore $L_1 \subseteq \mathcal{L}_{cand}$ where $\mathcal{L}_{cand}$ is output of Algorithm 6.

Therefore, we only now need to consider the case where $\dim(\mathrm{span}(\mathrm{Lin}(T_2+T_3) \setminus \mathrm{span}(\mathcal{L}_{s1}))) \leq 1$. We will break this into two cases depending on whether $\dim(\mathrm{span}(\mathrm{Lin}(T_2+T_3) \setminus \mathrm{span}(\mathcal{L}_{s1})))$ equals 0 or 1.

Case A.2: First, consider the subcase where $\dim(\mathrm{span}(\mathrm{Lin}(T_2+T_3) \setminus \mathrm{span}(\mathcal{L}_{s1}))) = 0$, in other words $\mathrm{Lin}(T_2 + T_3)$ is empty. All $\mathcal{S}_2$ spaces whose kernels contain any $l_1 \in L_1$ will be of form $\mathbb{V}(l_1, l)$ where $l_1 \in L_1$ and $l \in \mathrm{Lin}(T_2 + T_3) \cap \mathrm{span}(\mathcal{L}_{s1})$. Moreover, from the assumption of this case we know $\dim(\mathrm{span}(\mathrm{Lin}(T_2) \cup \mathrm{Lin}(T_3) \setminus \mathrm{span}(\mathcal{L}_{s1}))) \geq c_2 \geq \mathcal{R}(3) + 10 \geq 14$. Then, one of $\dim(\mathrm{span}(\mathrm{Lin}(T_2) \setminus \mathrm{span}(\mathcal{L}_{s1})))$ or $\dim(\mathrm{span}(\mathrm{Lin}(T_3) \setminus \mathrm{span}(\mathcal{L}_{s1})))$ is at least 7. Wlog, let it be for T_2. Note that for any linear form l to be in $\mathcal{L}_{s1} \setminus \mathrm{Lin}(G)$, as discussed earlier, it needs to divide $T_2 + T_3$(but not T_2 or T_3). For that to happen, the linear forms in T_2 and T_3 become the same mod it, and therefore, both will have at least 6 independent linear forms outside $\mathrm{span}(\mathcal{L}_{s1})$. We can also ensure $\mathrm{span}(\mathrm{Lin}(G)) = \mathrm{span}(\mathrm{Lin}(T_1 + T_2 + T_3))$ by guessing(from a basis) at most $6\mathcal{R}(3) \log d$ subspace from $\mathrm{span}(\mathcal{L}_{s1})$ and dividing by linear forms not in the subspace as we did in Theorem 6.1 to decrease $\dim(\mathrm{span}(\mathrm{Lin}(G)))$. In case $\mathcal{L}_{s1}$ is empty, we still have $\dim(\mathrm{span}(\mathrm{Lin}(T_3))) \geq 2$ from assumption in Theorem 6.2. Let L_2 and L_3 be sets of independent linear forms such that $L_3 \subseteq \mathrm{Lin}(T_3) \setminus \mathrm{span}(\mathcal{L}_{s1})$, $\dim(\mathrm{span}(L_3)) = 2$, $L_2 \subseteq \mathrm{Lin}(T_2)$, $\dim(\mathrm{span}(L_2)) = 4$, and for any $l_2 \in L_2$ it holds that $l_2 \notin \mathrm{span}(L_3 \cup \mathcal{L}_{s1})$. Consider the set of spaces $\mathbb{V}(l_1, l_2, l_3)$ with $l_1 \in L_1, l_2 \in L_2, l_3 \in L_3$. Our choice of L_2, L_3 ensures that these spaces are not contained in $\mathcal{S}_1$ spaces. These spaces will

be in $\mathcal{S}_3(f)$ or non-degenerate $\mathcal{S}_3^{sp}(f)$(where non-degenerate case as defined in Lemma 6.10) unless they are contained in a $\mathcal{S}_2^{sp}$ space whose kernel contains l_2 or l_3 (since all $\mathcal{S}_2$ spaces whose kernels contain anything in L_1 will also have a linear form from $\mathcal{L}_{s1}$ in the kernel which is not possible due to our choice of L_2, L_3). Recall that every space in $\mathcal{S}_3(f)$ or non-degenerate $\mathcal{S}_3^{sp}(f)$ will be learned.

Let $L_3 = \{l_3, l_3'\}$ and consider the spaces $\mathbb{V}(l_1, l_2, l_3)$ for $l_1 \in L_1, l_2 \in L_2$. We will argue that either we learn l_3 in the output of Algorithm 6 or at least two independent linear forms from T_2. We break up the analysis of this case into the following subcases.

Case A.2a: These exists a choice of $l_2 \in L_2$ such that there are two linear forms l_1, l_1' in L_1 for which $\mathbb{V}(l_1, l_2, l_3)$ and $\mathbb{V}(l_1', l_2, l_3)$ are learnable. Moreover there is a choice of $l_2' \in L_2$, and a choice of $l_1'' \in L_1(l_1''$ maybe equal to $l_1, l_1')$ such that $\mathbb{V}(l_1'', l_2', l_3)$ is learnable. In this case, by the intersection of the kernels of these learnable spaces we have $l_3 \in \mathcal{L}_{cand}$.

Case A.2b: For at least three choices of l_2 (say l_2, l_2', l_2'') for which it it holds that for at least all except one choice of linear form $l_1 \in L_1$, $\mathbb{V}(l_1, l_2, l_3)$ is degenerate (and hence unlearnable). For each degenerate space, we associate it with a linear form l''(equal to one of l_1, l_2, l_3) such that $\mathbb{V}(l_1, l_2, l_3) \subseteq \mathbb{V}(l, l') \in \mathcal{S}_2^{sp}$ and $l'' \in \text{span}(l, l')$.

Fix l_2. We will show that either $\mathbb{V}(l_2, l_3)$ is an $\mathcal{S}_2^{sp}$ space or one of l_2, l_3 is in $\mathcal{L}_{cand}$.

For the fixed choice of l_2 and l_3, we consider the set of spaces $\mathbb{V}(l_1, l_2, l_3)$ as l_1 ranges over all linear forms in L_1. Since, all except one of them are degenerate and there are at least 4 of them (in fact there are $5c_{cand} \log d - 1$), one of l_2 or l_3 is associated twice with some degenerate space[7]. If any two of the degenerate spaces (say $\mathbb{V}(l_1, l_2, l_3), \mathbb{V}(l_1', l_2, l_3)$) are contained in the same $\mathcal{S}_2^{sp}$ space $\mathbb{V}(l, l')$, then $\text{span}(l, l') \subset \text{span}(l_1, l_2, l_3)$ and $\text{span}(l, l') \subset \text{span}(l_1', l_2, l_3)$. This means $\text{span}(l, l') \subset \text{span}(l_1, l_2, l_3) \cap \text{span}(l_1', l_2, l_3)$, and $\text{span}(l_1, l_2, l_3) \cap \text{span}(l_1', l_2, l_3) = \text{span}(l_2, l_3)$ which means $\mathbb{V}(l, l') = \mathbb{V}(l_2, l_3)$. In particular $\mathbb{V}(l_2, l_3)$ is an $\mathcal{S}_2^{sp}$ space. If this does not happen, this means that no two choices of $\mathbb{V}(l_1, l_2, l_3)$ and $\mathbb{V}(l_1', l_2, l_3)$ are contained in the same $\mathcal{S}_2^{sp}$ space, then there are distinct $\mathcal{S}_2^{sp}$ spaces for each choice of degenerate space. In this situation, since one of l_2, l_3 is associated with two degenerate spaces, it is also contained in the kernel of two distinct $\mathcal{S}_2^{sp}$ spaces, and hence obtained in $\mathcal{L}_{cand}$.

We can argue the same for l_2' and l_2''. If any two of $\mathbb{V}(l_2, l_3), \mathbb{V}(l_2', l_3), \mathbb{V}(l_2'', l_3)$ are in $\mathcal{S}_2^{sp}$, we again have two distinct $\mathcal{S}_2^{sp}$ spaces whose kernels contain l_3, and therefore $l_3 \in \mathcal{L}_{cand}$. Otherwise, we have at least two linear forms (say l_2, l_2') in L_2 such that either $l_3 \in \mathcal{L}_{cand}$ or $l_2, l_2' \in \mathcal{L}_{cand}$. Thus we have either l_3 in the output of Algorithm 6 or at least two independent linear forms from T_2.

We can repeat this argument for $l_3' \in L_3$, and therefore, we have either two independent linear forms from T_2 or T_3 in $\mathcal{L}_{cand}$ output in Algorithm 6.

We are now done with Case A.2. We are left with the case of $\dim(\text{span}(\text{Lin}(T_2 + T_3)) = 1$. Let l' be the single linear form which equals $\text{span}(\text{Lin}(T_2 + T_3))$. We consider two further cases where $l' \notin \gcd(T_2, T_3)$ and $l' \in \gcd(T_2, T_3)$.

Case A.3a: $\dim(\text{span}(\text{Lin}(T_2 + T_3) \setminus \mathcal{L}_{s1})) = 1$ and if l' is the single linear form which equals $\text{span}(\text{Lin}(T_2 + T_3) \setminus \mathcal{L}_{s1}))$ then $l' | \text{sim}(T_2 + T_3)$.

In this case, T_2 and T_3 must be close in rank(by almost 1), since when we go mod l', they become nonzero and equal. Recall that by assumption of this case, $\dim(\text{span}(\text{Lin}(T_2) \cup \text{Lin}(T_3) \setminus \text{span}(\mathcal{L}_{s1}))) \geq c_2$. Applying this to the linear forms in T_2 and T_3, it follows that $\dim(\text{span}(\text{Lin}(T_2) \setminus \text{span}(\mathcal{L}_{s1})))$ and $\dim(\text{span}(\text{Lin}(T_3) \setminus \text{span}(\mathcal{L}_{s1})))$ both are at least 7 (by the property of closeness of rank). So, we can pick a set of independent linear forms $L_2 \subseteq \text{Lin}(T_2) \setminus \text{span}(\mathcal{L}_{s1})$ such that for any $l_2 \in L_2$, $l_2 \notin \text{span}(L_1, l')$ and $\dim(\text{span}(L_2)) = 4$. Here we are using the fact that L_1 which was picked earlier had all its linear forms not lying in $\text{span}(\text{Lin}(T_2) \cup \text{Lin}(T_3))$. Similarly, we can pick a set of independent linear forms $L_3 \subseteq \text{Lin}(T_3)$ such that for any $l_3 \in L_3$, $l_3 \notin \text{span}(L_1, L_2, l', \mathcal{L}_{s1})$ and

[7]as observed before, the degenerate space cannot be associated with a linear form in L_1

$\dim(\text{span}(L_3)) = 2$. Now, we are in the same situation as Case A.2, i.e. when $\text{Lin}(T_2+T_3)\backslash\text{span}(\mathcal{L}_{s1})$ was empty as our choice of L_2 and L_3 ensures the presence of l' doesn't interfere with the spaces $\mathbb{V}(l_1, l_2, l_3)$. In particular since every $\mathcal{S}_2^{reg}$ space must contain l', the choice of L_1, L_2 and L_3 guarantees that no space of the form $\mathbb{V}(l_1, l_2, l_3)$ coming from them is contained in an $\mathcal{S}_2^{reg}$ space. Also, our choice of L_2 and L_3 ensures that $\mathbb{V}(l_1, l_2, l_3)$ are not contained in any $\mathcal{S}_1$ spaces. Thus, the exact same argument works and we learn at least two linear forms from T_2 or T_3 in the output of Algorithm 6.

Case A.3b: $\dim(\text{span}(\text{Lin}(T_2 + T_3) \setminus \text{span}(\mathcal{L}_{s1}))) = 1$ and if l' is the single linear form which equals $\text{span}(\text{Lin}(T_2 + T_3) \setminus \text{span}(\mathcal{L}_{s1}))$ then $l' | \gcd(T_2, T_3)$.

In this case, we first observe that we can learn l' by the intersection of kernels of $\mathcal{S}_2$ spaces $\mathbb{V}(l_1, l')$ for $l_1 \in L_1$. As $\dim(\text{span}(\text{Lin}(T_2) \cup \text{Lin}(T_3) \setminus \text{span}(\mathcal{L}_{s1}))) \geq c_2$, it means for at least one of T_2 or T_3(wlog T_2) $\dim(\text{span}(\text{Lin}(T_2) \setminus \text{span}(\mathcal{L}_{s1})))$ is at least 7. As discussed all linear forms l in $\mathcal{L}_{s1} \setminus \text{Lin}(G)$, will be such that $l | T_1$ and $l | T_2 + T_3$. If there is a linear form in $\mathcal{L}_{s1} \setminus \text{Lin}(G)$, then there is no rank gap in T_2 and T_3 and this case is the same as case 3a. In case there is rank gap between T_2 and T_3, $\mathcal{L}_{s1} \setminus \text{Lin}(G)$ is empty. We have $\dim(\text{span}(\text{Lin}(T_3))) \geq 2$, so there is at least 1 linear form $l_3 \notin \text{span}(L_1, l')$. Let $L_2 \subseteq T_2$ be a set of independent linear forms such that for any $l_2 \in L_2$, $l_2 \notin \text{span}(L_1, l', l_3)$ and $\dim(\text{span}(L_2))) = 4$. Now, consider the set of spaces $\mathbb{V}(l_1, l_2, l_3)$ with $l_1 \in L_1$ and $l_2 \in L_2$. Again, we are in the same situation as case A.2 when $\text{Lin}(T_2 + T_3)$ was empty as our choice of L_2 and L_3 ensures the existence of l' doesn't interfere with the spaces $\mathbb{V}(l_1, l_2, l_3)$. So, we have either $l_3 \in \mathcal{L}_{cand}$ or at least two independent linear forms from T_2 in $\mathcal{L}_{cand}$ output from Algorithm 6. As we already observed we can learn $l' \in \gcd(T_2, T_3)$ in $\mathcal{L}_{cand}$. Thus we again learn at least two linear forms from T_2 or T_3 in the output of Algorithm 6.

This completes Case A.3. In each case, we have shown that we can either learn at least $c_{cand} \log d$ independent linear forms from T_1, or at least two independent linear forms from one of T_2 or T_3. In the event that we learned two independent linear forms from one of T_2 or T_3, we now show how to proceed.

We assume we have learned at least two independent linear forms l_3, l_3' in $\mathcal{L}_{cand} \cap \text{Lin}(T_3)$ or two independent linear forms l_2, l_2' in $\mathcal{L}_{cand} \cap \text{Lin}(T_2)$. Wlog, assume it is the former. Since there is a rank gap between T_1 and T_2(or T_3), $\text{rank}(\text{sim}(C \mod l_3)) > \mathcal{R}(4)$(same for l_3', l_2, l_2'). Therefore, We can use Theorem 3.10 to obtain a (possibly different) representation of the circuits $C \mod l_3$ and $C \mod l_3'$. The learned circuits will still be $\Sigma\Pi\Sigma(2)$ circuits, but as we will now observe, even a distinct $\Sigma\Pi\Sigma(2)$ representation of the same polynomial reveals enough information about the original representation. Let the original representation of $C \mod l_3$ be of the form $G' \times (T_1' + T_2')$ where $\gcd(T_1', T_2') = 1$. Then note that $(G \times T_1) \mod l_3 = G' \times T_1'$ and $(G \times T_2) \mod l_3 = G' \times T_2'$. The circuit reconstructed by Theorem 3.10 will be of the form $G'' \times (T_1'' + T_2'')$. As $G' \times (T_1' + T_2') - G'' \times (T_1'' + T_2'') = 0$, from rank bounds in Theorem 3.4, we have $\text{rank}(\text{sim}(G' \times (T_1' + T_2') - G'' \times (T_1'' + T_2''))) \leq \mathcal{R}(4)$ and hence either $\text{rank}(\text{sim}(G' \times T_1' - G'' \times T_1'')) \leq \mathcal{R}(4)$ or $\text{rank}(\text{sim}(G' \times T_1' - G'' \times T_2'')) \leq \mathcal{R}(4)$. Wlog, assume this happens with T_1'', i.e. $\text{rank}(\text{sim}(G' \times T_1' - G'' \times T_1'')) \leq \mathcal{R}(4)$. Thus among the linear forms that appear in $G'' \times T_1''$, all except the linear forms that lie in a $\mathcal{R}(4)$-dimensional space must appear as linear forms in $(G \times T_1) \mod l_3$. Therefore, we have learned the projection mod l_3 of at least $5c_{cand} \log d - \mathcal{R}(4)$ independent linear forms from T_1 [8] and similarly there are $5c_{cand} \log d - \mathcal{R}(4)$ independent linear forms from T_1 for which we know the projection mod l_3'. We then "glue" these projections (see Algorithm 9 for a description of the gluing) to obtain at least $c_{cand} \log d$ independent linear forms from T_1 in $\mathcal{L}_{cand}$ output in Algorithm 9.

As described earlier, the output of Algorithm 6 can be computed in $\text{poly}(n, d)$ time and is $d^{O(1)}$-sized list of linear forms. In Algorithm 9, the loop on Line 3 iterates over these $d^{O(1)}$ possibilities

[8] we actually learn a superset of these linear forms, but that suffices for us.

while the reconstruction of $\Sigma\Pi\Sigma(2)$ circuits is done by [Sin16b] in $\text{poly}(n,d)$ time. The number of linear forms added to $proj$ is also $O(d)$ and hence $|proj| = d^{O(1)}$ as well. Therefore, Algorithm 9 runs in $\text{poly}(n,d)$-time and outputs a list $\mathcal{L}_{cand}$ of size $d^{O(1)}$.

Thus, we finish with Case A. Now, we consider Case B where we have $\dim(\text{span}(\text{Lin}(T_2) \cup \text{Lin}(T_3) \setminus \text{span}(\mathcal{L}_{s1}))) < c_2$.

Case B: $\dim(\text{span}(\text{Lin}(T_2) \cup \text{Lin}(T_3) \setminus \text{span}(\mathcal{L}_{s1}))) < c_2$ In this case, we will use an approach similar to Lemma 6.13. We present the outline of the argument here, but most details follow from proofs of Lemma 6.11 and Lemma 6.13. We will do a random linear transformation on $\mathcal{L}_{s1}$ and project it to a constant $c = \mathcal{R}(3)$ dimensional space. Then, we will have the number of essential variables in $T_2 + T_3$ will be at most $c + c_2$ as $\dim(\text{span}(\text{Lin}(T_2) \cup \text{Lin}(T_3) \setminus \text{span}(\mathcal{L}_{s1}))) < c_2$ while T_1 will have $\Omega(\log d)$ essential variables. Therefore, we can learn projected versions of linear forms in T_1 by finding linear forms l such that mod l the circuit has at most $c + c_2$ essential variables. Finally, we can glue all these projected versions to get linear forms in T_1.

Wlog let $S := \{x_1, \ldots, x_{|S|}\}$ be a basis for $\mathcal{L}_{s1}$. We know from Lemma 6.4 $\dim(\text{span}(\mathcal{L}_{s1} \setminus \text{Lin}(G))) \leq 6\mathcal{R}(3)\log d$ and the assumption in Theorem 6.2 that $\dim(\text{span}(\text{Lin}(G))) \leq 6\mathcal{R}(3)\log d$, which combined means $|S| \leq 12\mathcal{R}(3)\log d$. Now, consider a random linear isomorphism Φ such that $\forall\ x_i \in S, \Phi(x_i) = \sum_{j=1}^{|S|} \alpha_{i,j} x_j$ where $\alpha_{i,j}$ are picked randomly from $\{1, \ldots, d^n\}$ and let $g = \Phi(f)$. Now we consider for each $i \in [c-1, |S|]$ the polynomials g_i, obtained from g by setting $x_{c-1} = \ldots = x_{i-1} = x_{i+1} = \ldots = x_{|S|} = 0$. Observe, that for each g_i, one can view the way g_i is obtained as taking random linear subspace of $\text{span}(S)$ and considering the circuit C modulo the subspace. By Lemma 3.2 and similar to the analysis of Lemma 5.4 it follows that with high probability, under this transformation each gate of C remains nonzero.

Thus for each i, the projected circuit g_i is of the form $T_1^{[i]} + T_2^{[i]} + T_3^{[i]}$ where $T_2^{[i]} + T_3^{[i]}$ has at most $c + c_2$ essential variables and $T_1^{[i]}$ has linear forms spanning a space with dimension at least $(5c_{cand} - 12\mathcal{R}(3))\log d$.

We find the set of linear forms from $T_1^{[i]}$ similar to the first half of Lemma 6.11. For each g_i, we take a random invertible linear transformation Ψ and project the circuits to the first $10(c+c_2)-1$ variables and x_j to obtain polynomials h_j. In this, with high probability, the projected $T_1^{[i,j]}$ will be full rank, i.e. $\dim(\text{span}(\Psi(T_1^{[i]})|_{x_{10(c+c_2)}=\ldots=x_{j-1}=x_{j+1}=\ldots=x_n=0})) = 10(c+c_2)$, while $T_2^{[i]} + T_3^{[i]}$ will have at most $c + c_2$ essential variables. We can then interpolate this in time $\text{poly}(d)$ to get monomial access to these polynomials h_j's. We can form a system of equations to find linear forms l such that mod l, the polynomial has $c + c_2$ essential variables using the partial derivative matrix similar to Lemma 6.11. Solving this system of equations for each h_j and gluing these linear forms will give us linear forms in $T_1^{[i]}$. The gluing step is the same as Lemma 6.11 and therefore also efficient.

Once we have linear forms in $T_1^{[i]}$ for each g_i, we glue them to get linear forms in T_1 in $\mathcal{L}_{cand}$ similar to Lemma 6.13. Thus, we obtain all linear forms in T_1, while we know $\dim(\text{span}(\text{Lin}(T_1))) \geq 5c_{cand}\log d$. Therefore, we have in this case as well $\dim(\text{span}(\text{Lin}(T_1) \cap \mathcal{L}_{cand})) \geq c_{cand}\log d$

□

6.4 Candidate Linear Forms when there is a linear form l such that $\text{rank}(\text{sim}(C \mod l)) < c_2$ and $C \mod l \neq 0$

We saw in Lemma 4.4, that we were able to bound the number of $\mathcal{S}_3$ spaces when the rank of the circuit is greater than c_3 except when there exists a linear form l, such that $\text{rank}(\text{sim}(C \mod l)) < c_2$ and $C \mod l \neq 0$. We showed in the previous section how to learn candidate linear forms whenever we could bound the number of $\mathcal{S}_3$ spaces. In this section, we will show how to

learn linear forms in the case where were not able to bound the number of $\mathcal{S}_3$ spaces, i.e. where there exists a linear form l, such that $\text{rank}(\text{sim}(C \mod l)) < c_2$ and $C \mod l \neq 0$. This condition restricts the circuit structure to be of a few different special types (or forms), and in each of these different structural types we show how to learn the candidate linear forms.

We are still assuming in this section the assumptions that come with Theorem 6.2.

We get the following types of circuits when there exists l such that $\text{rank}(\text{sim}(C \mod l)) < c_2$ and $C \mod l \neq 0$.

- No T_i vanishes mod l. (We handle this case in Lemma 6.15, and we call such circuits as *Special Form 1*)
- l divides some gate, say T_1, and $\text{rank}(\text{sim}(T_2+T_3)) < c_2$. (We handle this case in Lemma 6.17 and we call such circuits as *Special Form 2*)
- l divides some gate, say T_1, $\text{rank}(\text{sim}(T_2 + T_3)) \geq c_2$ but $\text{rank}(\text{sim}((T_2 + T_3) \mod l)) < c_2$. (We handle this case in Lemma 6.18 and we call such circuits as *Special Form 3*)

The first case is when l does not divide any T_i, and in this, most linear forms from all 3 gates (except c_2 independent ones) must move to the gcd when we go mod l. This means the circuit will have the following structure

Definition 10 (**Special form 1**). *We define an input polynomial f to be of Special form 1 if it can be computed by a $\Sigma\Pi\Sigma(3)$ circuit C such that, for some linear form l and constants $\alpha_i, \beta_i, \gamma_i \in \mathbb{F}$, it is of form*

$$C = G \times \left(\left(\prod_{i=1}^{d'} (l_i + \alpha_i l) \right) T_1' + \left(\prod_{i=1}^{d'} (l_i + \beta_i l) \right) T_2' + \left(\prod_{i=1}^{d'} (l_i + \gamma_i l) \right) T_3' \right)$$

where T_1', T_2', T_3' are the product of linear forms such that $\gcd(T_1', T_2', T_3') = 1$, $\text{rank}(T_1'+T_2'+T_3') < c_2$ where c_2 is as defined in Lemma 4.3. Moreover, there is no l' such that for some $i \in [3]$ $l'|T_i$ and $\text{rank}(\text{sim}(C \mod l')) < c_2$ and $C \mod l' \neq 0$.

Now, we are left with cases, when there exists a gate (wlog say it is T_1) that vanishes mod some linear form l, and $\text{rank}(\text{sim}(T_2 + T_3 \mod l)) < c_2$. We further break this into two cases and first handle the case where $\text{rank}(\text{sim}(T_2 + T_3)) < c_2$, as defined below.

Definition 11 (**Special form 2**). *We define an input polynomial to be of Special form 2 if it can be computed by a $\Sigma\Pi\Sigma(3)$ circuit C such that, it is of form*

$$C = G \times \left(\left(\prod_{i=1}^{d} l_i \right) + T_2 + T_3 \right)$$

such that $\gcd(\prod_{i=1}^{d} l_i, T_2, T_3) = 1$ and $\text{rank}(\text{sim}(T_2+T_3)) < c_2$ where c_2 is as defined in Lemma 4.3.

The only case left is that $\text{rank}(\text{sim}(T_2 + T_3))$ was high initially but for some $l|T_1$, we have $\text{rank}(\text{sim}(T_2 + T_3 \mod l)) < c_2$.

Definition 12 (**Special form 3**). *We define an input polynomial f to be of Special form 3 if it can be computed by a $\Sigma\Pi\Sigma(3)$ circuit C such that it is of the form*

$$C = G \times (T_1 + T_2 + T_3)$$

where $\gcd(T_1, T_2, T_3) = 1$ *and for some* $l \in T_1$, $\text{rank}(\text{sim}(T_2 + T_3 \mod l)) < c_2$, *but* $\text{rank}(\text{sim}(T_2 + T_3)) \geq c_2$. *This circuit will be of the following form*

$$C = G \times \left(l \cdot T_1' + \prod_{i=1}^{d'} (l_i + \alpha_i l) T_2' + \prod_{i=1}^{d'} (l_i + \beta_i l) T_3' \right)$$

such that $\text{rank}(\text{sim}(T_2' + T_3')) < c_2$ *where* c_2 *is as defined in Lemma 4.3.*

In this section, we handle all three special forms and get a $d^{O(1)}$ list of linear forms such that it has at least $c_{cand} \log d$ independent linear forms from one of the multiplication gates. The main issue with all special forms is that we do not have a bound on the size of the set $\mathcal{S}_3(f)$ for these polynomials, but we utilize the structure in the circuits to compute the candidate linear forms using $\mathcal{S}_2(f)$.

6.4.1 Candidate Linear forms for Special form 1

A polynomial in special form 1 is of form

$$C = G \times \left(\left(\prod_{i=1}^{d'} (l_i + \alpha_i l) \right) T_1' + \left(\prod_{i=1}^{d'} (l_i + \beta_i l) \right) T_2' + \left(\prod_{i=1}^{d'} (l_i + \gamma_i l) \right) T_3' \right)$$

for $\text{rank}(T_1' + T_2' + T_3') < c_2$.

The high level plan to learn linear forms for such circuits will be the following. We will first show how to obtain the linear form l (note that there might be many different choices of l for which the structure arises), which defines the structure of the circuit, and we do this by considering the intersection of $\mathcal{S}_2$ spaces. We then learn the l_i's by factoring the circuit mod l. To obtain the required constants, we use the fact that f will vanish over the codimension 3 subspace $\mathbb{V}(l_i + \alpha_i l, l_j + \beta_j l, l_k + \gamma_k l)$, to get a system of equations in $\alpha_i, \beta_j, \gamma_k$.

Lemma 6.15. *Let* f *be a polynomial that can be computed by a circuit* $C = G \times (T_1 + T_2 + T_3)$ *in Special form 1 as defined in Definition 10 satisfying all assumptions from Theorem 6.2. Then there exists an algorithm that computes a list of linear forms* $\mathcal{L}_{cand}$ *in* $\text{poly}(n, d)$ *time such that* $|\mathcal{L}_{cand}| = d^{O(1)}$ *and* $\dim(\text{span}(\mathcal{L}_{cand} \cap \text{Lin}(T_1))) \geq c_{cand} \log d$.

Proof. We will first show that for a polynomial that is computed by a circuit in special form 1, the $\mathcal{S}_2(f)$ spaces will contain spaces of form $\mathbb{V}(l, l_i)$ for many choices of $i \in [d']$. In particular we will show that there are at least two distinct spaces of the form $\mathbb{V}(l, l_i)$ that are learnt in $\mathcal{S}_2(f)$. To see this, first note from the definition of special forms, we have $C \mod l \neq 0$ and therefore $\mathbb{V}(l)$ is not an $\mathcal{S}_1$ space. Also, note that $\dim(\text{span}(\{l_i\}_{i \in [d']}) \geq 15 c_{cand} \log d - c_2 - 1$ as we know from assumption in Theorem 6.2 that $\text{rank}(\text{sim}(C)) \geq 15 c_{cand} \log d$. As $\dim(\text{span}(\mathcal{L}_{s1} \setminus \text{Lin}(G))) \leq 6\mathcal{R}(3) \log d$ from Lemma 6.4 and $\dim(\text{span}(\{l_i\}_{i \in [d']}) \geq 15 c_{cand} \log d - c_2 - 1$, thus it follows that we have at least 2 distinct spaces $\mathbb{V}(l, l_i)$ not contained in $\mathcal{S}_1$ spaces. Thus we we look at the intersection of kernels of all $\mathcal{S}_2(f)$ spaces, l will lie in one of the intersections and thus we can recover a list which contains l. As shown in Lemma 4.3, $|\mathcal{S}_2(f)| = d^{O(1)}$ and therefore, possibilities for l are also $d^{O(1)}$.

Once, we have access to l, we consider $C \mod l$ and a factorization of it gives us access to the different l_i (note that $C \mod l \neq 0$). Note that l (which determines the special form 1 structure) might not be unique. We will consider two cases based on whether there is a unique choice of l or there are at least 2 such choices of l.

Case 1: Assume, we have 2 such special linear forms l and l' which both give rise to the special form 1 structure.

Let the circuit in special form 1 using the linear form l be of the form

$$C = G \times \left(\left(\prod_{i=1}^{d'} (l_i + \alpha_i l) \right) T_1' + \left(\prod_{i=1}^{d'} (l_i + \beta_i l) \right) T_2' + \left(\prod_{i=1}^{d'} (l_i + \gamma_i l) \right) T_3' \right).$$

Also, let the circuit in special form 1 with l' be of form

$$C' = G' \times \left(\left(\prod_{i=1}^{d'} (l_i' + \alpha_i' l') \right) T_1'' + \left(\prod_{i=1}^{d'} (l_i' + \beta_i' l') \right) T_2'' + \left(\prod_{i=1}^{d'} (l_i' + \gamma_i' l') \right) T_3'' \right).$$

Since they are computing the same polynomial, we have $C - C'$ computes an identity. By relabelling of gates and by rank bounds (Lemma 3.4), we can assume the first gate in both representations are "close" (i.e. the have common linear forms except for those lying in a $\mathcal{R}(6)$ dimensional space). Since by assumption G and G' both have rank at most $6\mathcal{R}(3)\log d$ and since T_1' and T_1'' both have rank at most c_2, thus, except for linear forms lying in a $2c_2 + \mathcal{R}(6) + 12\mathcal{R}(3)\log d$ dimensional space, we get that for each i, there is a j such that $l_i + \alpha_i l = l_j' + \alpha_j' l'$

When we go mod l, all the linear forms in $\prod_{i=1}^{d'}(l_i + \alpha_i l)$ move to the gcd and hence can be learnt by factorization. When we go mod l', all the linear forms in $\prod_{i=1}^{d'}(l_i' + \alpha_i' l')$ move to the gcd and hence can be learnt by factorization. Thus except for linear forms in a $2c_2 + \mathcal{R}(6) + 12\mathcal{R}(3)\log d$ dimensional space, for all other common linear forms that appear in $\prod_{i=1}^{d'}(l_i + \alpha_i l)$ as well as in $\prod_{i=1}^{d'}(l_i' + \alpha_i' l')$, we have learnt those linear forms mod l and well as mod l'.

Observe that there are at least $15c_{cand}\log d - 2c_2 - \mathcal{R}(6) - 12\mathcal{R}(3)\log d$ independent and common linear forms appearing in each of $\prod_{i=1}^{d'}(l_i + \alpha_i l)$ and $\prod_{i=1}^{d'}(l_i' + \alpha_i' l')$. Consider linear form $l_i + \alpha_i l$ that is common on both circuits. Thus $l_i + \alpha_i l = l_i' + \alpha_i' l'$. Since we can learn l, l', l_i, and l_i' thus this information is enough to recover α_i and α_i' by solving a suitable system of linear equations. Once we do this, we add $l_i + \alpha_i l$ to the set of candidate linear forms. Thus we obtain $15c_{cand}\log d - 2c_2 - \mathcal{R}(6) - 12\mathcal{R}(3)\log d$ independent linear forms in one of gates of C.

Case 2: We now consider the case when there is a unique choice of linear form l that gives rise to the special form structure for the underlying polynomial. After going mod l and factoring, we learn the different l_i, which as a set are high dimensional ($\dim(\mathrm{span}(\{l_i\}_{i\in[d']}) \geq 15c_{cand}\log d - c_2 - 1$). We will be interested in triples of linear forms of the following kind: the triple l_i, l_j, l_k from the list of linear factors that we will like to consider is such that $\dim(\mathrm{span}(l, l_i, l_j, l_k)) = 4$. Define the set $\mathcal{S}_3^*(f)$ to be the set of codimension 3 spaces of the form $\mathbb{V}(l_i + \alpha l, l_j + \beta l, l_k + \gamma l)$ with l_i, l_j, l_k satisfying properties above and $\alpha, \beta, \gamma \in \mathbb{F}$ such that $\mathbb{V}(l_i + \alpha l, l_j + \beta l, l_k + \gamma l) \in \mathcal{S}_3(f)$. We see that for $\alpha = \alpha_i, \beta = \beta_j, \gamma = \gamma_k$, $\mathbb{V}(l_i + \alpha l, l_j + \beta l, l_k + \gamma l)$ belongs to $\mathcal{S}_3^*(f)$. We would like to learn $(\alpha_i, \beta_j, \gamma_k)$ (and do this for many choices of i, j, k), and to do this, we will try to learn $\mathcal{S}_3^*(f)$. We then add $l_i + \alpha_i l, l_j + \beta_j l$ and $l_k + \gamma_k l$ to the set $\mathcal{L}_{cand}$.

Claim 6.16. *Assuming, we are in case 2 of Special form 1, i.e. there is a unique linear form l giving rise to the special form structure, then $\mathcal{S}_3^*(f)$ can be computed in randomized time* $\mathrm{poly}(n, d)$*, and $|\mathcal{S}_3^*(f)| = d^{O(1)}$.*

Proof. We will first prove that $|\mathcal{S}_3^*(f)| = d^{O(1)}$. From Lemma 4.4, we have that if rank of the circuit is at least c_3, and there does not exist a linear form l such that $\mathrm{rank}(\mathrm{sim}(C \mod l)) \leq c_2$, then $|\mathcal{S}_3(f)| = d^{O(1)}$. However in our case, we do have a linear form l such that $\mathrm{rank}(\mathrm{sim}(C$

mod l)) $\leq c_2$. However though we are not able to bound $|\mathcal{S}_3(f)|$, we will still be able to bound $|\mathcal{S}_3^*(f)|$. To see this, we inspect the proof of Lemma 4.4. Note that in our setting there is a single unique linear form l such that $\text{rank}(\text{sim}(C \mod l)) \leq c_2$. l does not divide any gate T_i of the circuit. Thus the only case where Lemma 4.4 does not allow us to prove a bound on $|\mathcal{S}_3(f)|$ is case 2.A. In this case, we consider triples of the form $\mathbb{V}(l_1, l_2, l_3)$, and if there is no $l' \in \text{span}(l_1, l_2, l_3)$ such that $\text{rank}(\text{sim}(C \mod l')) \leq c_2$ then the number of such triples is bounded. Now, of course in $\mathcal{S}_3(f)$ this condition does not always hold. However, it is easy to see that in $\mathcal{S}_3^*(f)$ it does hold! In every space $\mathbb{V}(l_i + \alpha l, l_j + \beta l, l_k + \gamma l)$ in $\mathcal{S}_3^*(f)$, clearly $l \notin \text{span}\{l_i + \alpha l, l_j + \beta l, l_k + \gamma l\}$ since $\dim(\text{span}(l, l_i, l_j, l_k)) = 4$. Also since l is the unique linear form such that $\text{rank}(\text{sim}(C \mod l)) \leq c_2$ thus there is no linear form in $\text{span}\{l_i + \alpha l, l_j + \beta l, l_k + \gamma l\}$ such that modulo it the rank crashes. Thus we are able to bound the number of codimension 3 spaces in $\mathcal{S}_3^*(f)$ even in Case 2.A and hence we are able to bound the number of these spaces overall.

We now show how to learn the set of spaces in $\mathcal{S}_3^*(f)$. The algorithm is even simpler than that for learning $\mathcal{S}_3(f)$ since we already know l and the various l_i. Thus after projecting to few variables and solving a system of polynomial equations, we can recover the values of all possible α, β, γ and we do not need to glue and lift. We provide the details below.

Pick l_i, l_j, l_k from $\{l_i\}_{i \in [d']}$ learnt from factoring mod l such that $\dim(\text{span}(l, l_i, l_j, l_k)) = 4$. Just like in Lemma 5.2 we consider a random invertible linear transformation Φ and set all but $t = c_3 = O(1)$ (c_3 is as in Lemma 4.4) variables $x_{t+1} = \ldots = x_n = 0$, to obtain g. As Φ is a random linear isomorphism, if f vanishes on $\mathbb{V}(l_1, l_2, l_3)$ then g vanishes on $\mathbb{V}(\Phi(l_1), \Phi(l_2), \Phi(l_3))|_{x_{t+1}=\ldots=x_n=0}$ with high probability.

Therefore, g vanishes on $\mathbb{V}(\Phi(l_i) + \alpha_i \Phi(l), \Phi(l_j) + \beta_j \Phi(l), \Phi(l_k) + \gamma_k \Phi(l))_{x_{t+1}=\ldots=x_n=0}$. Let α, β, γ be formal variables, and consider any invertible linear transformation Ψ on $x_1, \ldots, x_t$ such that it takes $x_1 \leftarrow \Phi(l_i + \alpha l), x_2 \leftarrow \Phi(l_j + \beta l), x_3 \leftarrow \Phi(l_k + \gamma l)$. Consider $\Psi(g)$ after setting $x_1 = x_2 = x_3 = 0$. Set up a system of polynomial equations in α, β, γ by equating the coefficients of monomials in $x_4, \ldots, x_t$ in $\Psi(g)|_{x_1=x_2=x_3=0}$ to zero. Also add equations so that
$\mathbb{V}(\Phi(l_i) + \alpha_i \Phi(l), \Phi(l_j) + \beta_j \Phi(l), \Phi(l_k) + \gamma_k \Phi(l))$ is not contained in a $\mathcal{S}_1$ or $\mathcal{S}_2$ space after the projection and Ψ similar to what was done in Lemma 5.6. So, the solutions of the system of equations will contain $\alpha = \alpha_i, \beta = \beta_j, \gamma = \gamma_k$ for all required values that determine the set $\mathcal{S}_3^{ast}(f)$.

So, all we need to argue is that the system of equations will have at most $d^{O(1)}$ solutions. As discussed in proof of Lemma 5.4, after a random invertible linear transformation and setting of all except constant variables, with high probability, the circuit will be full-rank, i.e.
$\text{rank}(\text{sim}(\Phi(C)|_{x_{t+1}=\ldots=x_n=0})) = t$. Moreover, by Lemma 5.7, there will remain a unique linear form such that modulo it the circuit has rank at most c_2. Thus even after the random linear transformation and projection, the number of $\mathcal{S}_3^*$ spaces of the new polynomial are still $d^{O(1)}$. It is easy to see that each solution of the system of equations we set up corresponds to a distinct $\mathcal{S}_3^*$ space of the projected polynomial. Therefore, the number of solutions of the system of equations will be $d^{O(1)}$, and hence we can efficiently find all solutions.

□

We now need to show that we can learn $c_{cand} \log d$ independent linear forms from a gate, and the analysis is basically identical to Case 1 of Lemma 6.10 except that we use $\mathcal{S}_3^*(f)$ which we computed instead of $\mathcal{S}_3(f)$ (which we do not know how to bound).

Now, we want to show we learn enough linear forms from our computation of $\mathcal{S}_3^*(f)$. Interestingly, if $\mathbb{V}(l_i + \alpha l, l_j + \beta l, l_k + \gamma l)$ is a $\mathcal{S}_3^*(f)$ space, then we learn three linear forms, one in each of the gates. We can also learn linear forms from the intersection of kernels of $\mathcal{S}_2$ spaces since $\mathcal{S}_2(f)$ can be computed in our setting. Observe that we can still also compute $\mathcal{S}_3^{sp}$ spaces just as they were defined and computed in Lemma 6.8, and so can also learn linear forms from the intersection

of kernels of spaces in $\overline{\mathcal{S}_3^{sp}}$. We will show that the union of linear forms computed by all these intersections and the computation of $\mathcal{S}_3^*(f)$ spaces will give us all the linear forms we need.

From Lemma 6.4, we know that the $dim(\text{span}(\mathcal{L}_{s1} \setminus \text{Lin}(G))) \leq 6\mathcal{R}(3)\log d$. Therefore, the dimension of the span of linear forms from any T_i in $\mathcal{L}_{s1}$ is at most $6\mathcal{R}(3)\log d$. Fix S to be any maximal set of independent spaces[9] of the form $\text{span}(l_1, l_2, l_3)$ with $l_1 \in T_1, l_2 \in T_2, l_3 \in T_3$ such that $\mathbb{V}(l_1, l_2, l_3)$ is contained in some space $\mathbb{V}(l_1', l_2') \in \mathcal{S}_2^{reg}(f)$, in particular $\text{span}(l_1', l_2') \subset \text{span}(l_1, l_2, l_3)$. As the spaces $\text{span}(l_1, l_2, l_3)$ are all independent, all the corresponding spaces in $\mathcal{S}_2^{reg}(f)$ will have independent kernels and hence will be an Independent Vanishing Set. Thus it follows from Lemma 6.6 that $|S| \leq 6\mathcal{R}(3)\log d$ and therefore $\dim(\text{span}(S)) \leq 18\mathcal{R}(3)\log d$. First, we observe the following. Let W be any codimension-3 space of the form $\mathbb{V}(l_1, l_2, l_3)$ on which f vanishes and such that $l_1 \in T_1, l_2 \in T_2, l_3 \in T_3$. Then if $W \subseteq \mathbb{V}(l_1', l_2')$ such that $\mathbb{V}(l_1', l_2') \in \mathcal{S}_2^{reg}(f)$, then $\text{span}(l_1, l_2, l_3)$ has to intersect $\text{span}(S)$. This follows from the maximality of S. Let $S' = \{S \cup (\mathcal{L}_{s1} \setminus \text{Lin}(G))\}$.

Consider any $(l_1 + \alpha_1 l) \in \text{Lin}(\prod_{i=1}^{d'}(l_i + \alpha_i l)) \notin \text{span}(S')$. Note that there will be at least $(15c_{cand} - 24\mathcal{R}(3))\log d - c_2 - 1$ such independent linear forms in $\text{Lin}(T_1)$.

If we consider $T_1 + T_2 + T_3 \mod (l_1 + \alpha_1 l)$, it will be of the form $G' \times (T_2'' + T_3'')$ where $\gcd(T_2'', T_3'') = 1$. We consider two cases as follows

Case(a) $(l_1 + \alpha_1 l) \in \text{Lin}(\prod_{i=1}^{d'}(l_i + \alpha_i l))$ is such that $(l_1 + \alpha_1 l) \notin \text{span}(S')$ and $\dim(\text{span}(\text{Lin}(C \mod (l_1 + \alpha_1 l)))) \geq 12\mathcal{R}(3)\log d + 3$. In this case, we will show that $(l_1 + \alpha_1 l)$ will be in $\mathcal{L}_{cand}$.

Observe that since $\dim(\mathcal{L}_{s1} \setminus \text{Lin}(G)) \leq 6\mathcal{R}(3)\log d$ (by Lemma 6.4) and by assumption, $\dim(\text{Lin}(G)) \leq 6\mathcal{R}(3)\log d$, thus $\dim(\mathcal{L}_{s1}) \leq 12\mathcal{R}(3)\log d$. It follows that there exist two independent linear forms l and l' dividing G' such that $\mathbb{V}((l_1 + \alpha_1 l), l)$ and $\mathbb{V}((l_1 + \alpha_1 l), l')$ are not contained within any space in $\mathcal{S}_1(f)$ and moreover f vanishes on them. Hence they lie in $\mathcal{S}_2(f)$, with their kernels intersecting in $(l_1 + \alpha_1 l)$. Hence all such $(l_1 + \alpha_1 l) \in \text{Lin}(\prod_{i=1}^{d'}(l_i + \alpha_i l))$ will be in $\mathcal{L}_{cand}$.

Case(b) $(l_1 + \alpha_1 l) \in \text{Lin}(\prod_{i=1}^{d'}(l_i + \alpha_i l))$ is such that $(l_1 + \alpha_1 l) \notin \text{span}(S')$ and $\dim(\text{span}(\text{Lin}(C \mod (l_1 + \alpha_1 l)))) \leq 12\mathcal{R}(3)\log d + 3$.

We will show in most typical cases, any such $(l_1 + \alpha_1 l)$ will be in $\mathcal{L}_{cand}$. As we are showing this, there will arise one degenerate case where we fail, but then we will show that we can learn enough linear forms from T_2 or T_3.

Pick any $(l_1 + \alpha_1 l) \in \text{Lin}(\prod_{i=1}^{d'}(l_i + \alpha_i l))$ such that $(l_1 + \alpha_1 l) \notin \text{span}(S')$, pick any $(l_3 + \gamma_3 l) \in \text{Lin}(\prod_{i=1}^{d'}(l_i + \gamma_i l))$ and $(l_3 + \gamma_3 l) \notin \text{span}(S' \cup \{l_1, l\} \cup \text{Lin}(G'))$ and pick any $(l_2 + \beta_2 l) \in \text{Lin}(\prod_{i=1}^{d'}(l_i + \beta_i l))$ and $(l_2 + \beta_2 l) \notin \text{span}(S' \cup \{l_1, l_3, l\} \cup \text{Lin}(G'))$. Consider $\mathbb{V}((l_1 + \alpha_1 l), (l_2 + \beta_2 l), (l_3 + \gamma_3 l))$. Note that it does not intersect S' and hence is not contained in any space in $\mathcal{S}_1$ and $\mathcal{S}_2^{reg}$. Thus we will learn this space in $\mathcal{S}_3^*(f)$ unless it is contained in a space in $\mathcal{S}_2^{sp}$, say $\mathbb{V}(l, l')$. We first observe that $(l_1 + \alpha_1 l) \notin \text{span}(l, l')$. This because if it was the case then the space $\text{span}(l, l')$ would be of the form $\text{span}((l_1 + \alpha_1 l), l_1')$ where $l_1' \in \text{Lin}(G')$. In particular $l_1' \in \text{span}((l_1 + \alpha_1 l), (l_2 + \beta_2 l), (l_3 + \gamma_3 l))$ but by choice of $(l_2 + \beta_2 l)$ and $(l_3 + \gamma_3 l)$, this is not possible. Moreover observe that if $(l_2 + \beta_2 l)$ and $(l_3 + \gamma_3 l)$ are also both not in $\text{span}(l, l')$, then we call any such $\mathbb{V}((l_1 + \alpha_1 l), (l_2 + \beta_2 l), (l_3 + \gamma_3 l))$ a non-degenerate space for learning $(l_1 + \alpha_1 l)$. Moreover every such space gets learned as a space in $\mathcal{S}_3^{sp}$(contained in $\overline{\mathcal{S}_3^{sp}}$) or $\mathcal{S}_3^*$. It is not hard to see (we will formalize below) that enough non-degenerate spaces for learning $(l_1 + \alpha_1 l)$ will suffice in determining $(l_1 + \alpha_1 l)$.

The issue arises in the degenerate case which is when either $(l_2 + \beta_2 l)$ or $(l_3 + \gamma_3 l)$ is contained within $\mathbb{V}(l, l')$. We call any such $\mathbb{V}((l_1 + \alpha_1 l), (l_2 + \beta_2 l), (l_3 + \gamma_3 l))$ a degenerate space for learning $(l_1 + \alpha_1 l)$. We will show that enough degenerate spaces will enable us to learn lots of linear forms

[9] where a set of spaces is independent if the dimension of the span of their union is a sum of dimensions of the individual spaces

from either T_2 or T_3, and hence in this case also we are done.

As $\dim(\text{span}(\text{Lin}(C \mod (l_1+\alpha_1 l))))) \leq 12\mathcal{R}(3)\log d+3$, we have $\dim(\text{span}(\text{Lin}(T_2'')))) \geq (15c_{cand}-12\mathcal{R}(3))\log d - c_2 - 3$ and $\dim(\text{span}(\text{Lin}(T_3'')))) \geq (15c_{cand}-12\mathcal{R}(3))\log d - c_2 - 3$. We pick a set of independent linear forms $L_3 \subseteq \text{Lin}\left(\prod_{i=1}^{d'}(l_i+\gamma_i l)\right)$ such that for any $(l_3+\gamma_3 l) \in L_3$, $((l_3+\gamma_3 l) \mod (l_1+\alpha_1 l) \in \text{Lin}(T_3'')$ and $l_3 \notin \text{span}(S' \cup \{l_1, l\} \cup \text{Lin}(G'))$ and $\dim(\text{span}(L_3)) = 2c_{cand}\log d + 2$. It is not hard to see that such a set exists. Similarly, we pick $L_2 \subseteq \text{Lin}\left(\prod_{i=1}^{d'}(l_i+\beta_i l)\right)$ such that for any $(l_2+\beta_2 l) \in L_2$, $((l_2+\beta_2 l) \mod (l_1+\alpha_1 l)) \in \text{Lin}(T_2'') \notin \text{span}(S' \cup \{l_1, l\} \cup L_3 \cup \text{Lin}(G'))$ and $\dim(\text{span}(L_2)) = 2c_{cand}\log d + 2$. Again it is not hard to see that such a set exists. In fact there could have been as many as $13c_{cand} - 36\mathcal{R}(3))\log d - 3$ such linear forms since $\{l_i\}_{i\in[d']}$ by assumption starts off being significantly high rank.

Now, consider the set of spaces $\mathbb{S}(l_1)$ of the form $\mathbb{V}((l_1+\alpha_1 l),(l_2+\beta_2 l),(l_3+\gamma_3 l))$ where $(l_2+\beta_2 l) \in L_2$ and $(l_3+\gamma_3 l) \in L_3$. Assume there is $(l_2+\beta_2 l) \in L_2$ such that there are two linear forms $(l_3+\gamma_3 l),(l_3'+\gamma_3' l)$ in L_3 for which $\mathbb{V}((l_1+\alpha_1 l),(l_2+\beta_2 l),(l_3+\gamma_3 l))$ and $\mathbb{V}((l_1+\alpha_1 l),(l_2+\beta_2 l),(l_3'+\gamma_3' l))$ are non-degenerate. Then for any $(l_2'+\beta_2' l) \in L_2$, if there is a $(l_3''+\gamma_3'' l) \in L_3((l_3''+\gamma_3'' l)$ maybe equal to $(l_3+\gamma_3 l),(l_3'+\gamma_3' l))$ such that $\mathbb{V}((l_1+\alpha_1 l),(l_2'+\beta_2' l),(l_3''+\gamma_3'' l))$ is non-degenerate, then by the intersection of the kernels of these non-degenerate spaces (which we observed we can learn) we have $(l_1+\alpha_1 l) \in \mathcal{L}_{cand}$. Also, if we are in the case where we have $(l_2+\beta_2 l),(l_2'+\beta_2' l),(l_2''+\beta_2'' l) \in L_2$ and $(l_3+\gamma_3 l),(l_3'+\gamma_3' l),(l_3''+\gamma_3'' l) \in L_3$ such that all three of $\mathbb{V}((l_1+\alpha_1 l),(l_2+\beta_2 l),(l_3+\gamma_3 l)),\mathbb{V}((l_1+\alpha_1 l),(l_2'+\beta_2' l),(l_3'+\gamma_3' l)),\mathbb{V}((l_1+\alpha_1 l),(l_2''+\beta_2'' l),(l_3''+\gamma_3'' l))$ are non-degenerate, then we again have $(l_1+\alpha_1 l) \in \mathcal{L}_{cand}$ by considering the intersections of kernels of these spaces.

So, there are only two cases where we did not manage to deduce that $(l_1+\alpha_1 l) \in \mathcal{L}_{cand}$. Either there is a linear form in $(l_2+\beta_2 l) \in L_2$ such that for any other $(l_2'+\beta_2' l) \in L_2$, and for any $(l_3+\gamma_3 l) \in L_3$, $\mathbb{V}((l_1+\alpha_1 l),(l_2'+\beta_2' l),(l_3+\gamma_3 l))$ is degenerate. Else, there are only two distinct spaces in $\mathbb{S}(l_1)$ of the form $\mathbb{V}((l_1+\alpha_1 l),(l_2+\beta_2 l),(l_3+\gamma_3 l))$ and $\mathbb{V}((l_1+\alpha_1 l),(l_2'+\beta_2' l),(l_3'+\gamma_3' l))$ with $(l_2+\beta_2 l) \neq (l_2'+\beta_2' l)$ and $(l_3+\gamma_3 l) \neq (l_3'+\gamma_3' l)$ which are non-degenerate.

In both cases, notice we have at least $2c_{cand}\log d$ independent linear forms from each of L_2, L_3 (call these sets L_2' and L_3') for which all the corresponding $4c_{cand}^2\log^2 d$ spaces are in the degenerate setting. By definition of degeneracy any such degenerate space $\mathbb{V}((l_1+\alpha_1 l),(l_2+\beta_2 l),(l_3+\gamma_3 l))$ is contained in an $\mathcal{S}_2^{sp}$ space of the form $\mathbb{V}(l,l')$ where either l_2 or l_3 is contained in $\text{span}(l,l')$. Recall that we have learnt all these $\mathcal{S}_2^{sp}$ spaces. Moreover all the $4c_{cand}^2\log^2 d$ $\mathcal{S}_2^{sp}$ spaces are distinct by choice of independence of linear forms that went into L_2 and L_3. To each such $\mathcal{S}_2^{sp}$ space, we can associate it with a choice of $(l_2+\beta_2 l) \in L_2'$ or $(l_3+\gamma_3 l) \in L_3'$ that is contained in its kernel. If there are two distinct $\mathcal{S}_2^{sp}$ spaces that are associated with the same $(l_2+\beta_2 l) \in L_2'$ or $(l_3+\gamma_3 l) \in L_3'$ then that choice of $(l_2+\beta_2 l)$ or $(l_3+\gamma_3 l)$ will be learned in $\mathcal{L}_{cand}$. Since each choice of $(l_2+\beta_2 l)$ or $(l_3+\gamma_3 l)$ can be associated with at most $2c_{cand}\log d$ of the $\mathcal{S}_2^{sp}$ spaces, thus by a simple averaging argument there are at least $2c_{cand}\log d$ independent linear forms from the union of L_2' and L_3' which are each associated with at least two distinct $\mathcal{S}_2^{sp}$ spaces and hence are in $\mathcal{L}_{cand}$. This means from at least one of T_2, T_3 there are at least $c_{cand}\log d$ linear forms in $\mathcal{L}_{cand}$. □

6.4.2 Candidate Linear forms for Special form 2

A polynomial in special form 2 has a circuit of the form

$$C = G \times \left(\left(\prod_{i=1}^{d} l_i\right) + T_2 + T_3\right)$$

such that $\gcd(\prod_{i=1}^{d} l_i, T_2, T_3) = 1$ and $\text{rank}(\text{sim}(T_2+T_3)) < c_2$ where c_2 is as defined in Lemma 4.3.

Lemma 6.17. *Let f be a polynomial that can be computed by a circuit C in special form 2 as defined in Definition 11 satisfying all assumptions from Theorem 6.2, then there exists an algorithm that computes a list of linear forms $\mathcal{L}_{cand}$ in $\text{poly}(n,d)$ time such that $|\mathcal{L}_{cand}| = d^{O(1)}$ and $\dim(\text{span}(\mathcal{L}_{cand} \cap \text{Lin}(T_1))) \geq c_{cand} \log d$.*

Proof. In this case, the input circuit is of form $C = G \times (T_1 + T_2 + T_3)$ with $\gcd(T_1, T_2, T_3) = 1$ such that $\text{rank}(T_1 + T_2 + T_3) \geq 15 c_{cand} \log d$ and $\text{rank}(\text{sim}(T_2 + T_3)) < c_2$. We will again break the analysis into several cases and argue that we have handled most of the cases already in previous arguments, and the only interesting case left for the lemma is when $\dim(\text{span}(\gcd(T_2, T_3))) < c_2 + 2$.

To do this, we consider the following division into cases for the problem

- **Case 1:** When $\dim(\text{span}(\gcd(T_2, T_3))) \geq 5 c_{cand} \log d$:

  We divide this further into 2 cases based on $\dim(\text{span}(\text{Lin}(T_1) \setminus \text{span}(\mathcal{L}_{s1})))$:

  - **Case 1.a** $\dim(\text{span}(\text{Lin}(T_1) \setminus \text{span}(\mathcal{L}_{s1}))) \geq 2$. From Lemma 6.4 and assumption on G from Theorem 6.2, we have that $\dim(\text{span}(\mathcal{L}_{s1})) \leq 12\mathcal{R}(3) \log d$. Therefore, $\dim(\text{span}(\gcd(T_2, T_3) \setminus \mathcal{L}_{s1})) \geq (5 c_{cand} - 12\mathcal{R}(3)) \log d > c_{cand} \log d$. Let l_1, l_2 be two independent linear forms that lie in $\text{Lin}(T_1) \setminus \text{span}(\mathcal{L}_{s1})$. Then for every linear form $l \in \gcd(T_2, T_3) \setminus \mathcal{L}_{s1}$, $\mathbb{V}(l, l_1)$ and $\mathbb{V}(l, l_2)$ are distinct spaces in $\mathcal{S}_2(f)$, and hence l can be learnt by the intersection of their kernels.

    Therefore, we learn $c_{cand} \log d$ linear forms in $\gcd(T_2, T_3)$ by intersection of kernels of $\mathcal{S}_2$ spaces.
  - **Case 1.b** $\dim(\text{span}(\text{Lin}(T_1) \setminus \text{span}(\mathcal{L}_{s1}))) \leq 1$. This case has been handled in Lemma 6.13 where we handle the case when there are 2 high-rank gates and for some gate $\dim(\text{span}(\text{Lin}(T_i) \setminus \text{span}(\mathcal{L}_{s1}))) \leq 1$. Note that in Lemma 6.13, we do not use $\mathcal{S}_3(f)$ and therefore the condition $\text{rank}(\text{sim}(C \mod l)) < c_2$ doesn't affect it.

- **Case 2:** When $c_2 + 2 \leq \dim(\text{span}(\gcd(T_2, T_3))) < 5 c_{cand} \log d$:

  In this case, we have from Lemma 6.5 that $\mathcal{L}_{s1}$ will only have linear forms that divide one of the gates in T_1, T_2, T_3 as combined rank of T_2, T_3 is low(since their gcd is low rank and their simple part has constant rank). This also means T_1 is high rank, i.e. $\dim(\text{span}(\text{Lin}(T_1))) \geq 10 c_{cand} \log d - c_2$. Any linear form that divides $\gcd(T_2, T_3)$ will not divide T_1 as $\gcd(T_1, T_2, T_3) = 1$. Moreover any linear form in $\mathcal{L}_{s1} \setminus \text{Lin}(G)$ will not divide T_2 or T_3 as if it did then it would also divide $T_1 + T_2$ or $T_1 + T_3$, but this cannot happen due to their rank difference. Therefore, all linear forms in $\mathcal{L}_{s1} \setminus \text{Lin}(G)$ must divide T_1 and $\text{sim}(T_2 + T_3)$. Since $\text{rank}(\text{sim}(T_2 + T_3)) < c_2$, $\dim(\text{span}(\mathcal{L}_{s1} \setminus \text{Lin}(G))) < c_2$. Now, we will argue that we can reduce it to the case when $\dim(\text{span}(\mathcal{L}_{s1})) < c_2$. We can guess (run for all choices) a $c_2 - 1$ dimensional subspace of $\mathcal{L}_{s1}$ and divide the circuit by the linear forms in $\mathcal{L}_{s1}$ not in the subspace. For correct choice of this space, we only divide by linear forms in $\text{Lin}(G)$. Thus the resulting circuit retains its $\Sigma\Pi\Sigma(3)$ structure and it suffices to learn the candidate linear forms from it. The new circuit clearly has $\dim(\text{span}(\mathcal{L}_{s1})) < c_2$. We also have $\dim(\text{span}(\gcd(T_2, T_3))) \geq c_2 + 2$, and therefore there are at least 2 independent linear forms in $\text{Lin}(\gcd(T_2, T_3)) \setminus \mathcal{L}_{s1}$. Let l_1, l_2 be two independent linear forms that lie in $\gcd(T_2, T_3) \setminus \mathcal{L}_{s1}$. Also, recall that $\dim(\text{span}(\text{Lin}(T_1))) \geq c_{cand} \log d + c_2 + 2$. Then for every linear form $l \in \text{Lin}(T_1)$ such that l is not in $\text{span}(\mathcal{L}_{s1}, l_1, l_2)$, it follows that $\mathbb{V}(l, l_1)$ and $\mathbb{V}(l, l_2)$ are distinct spaces in $\mathcal{S}_2(f)$, and hence l can be learnt by the intersection of their kernels.

  Therefore, we learn $c_{cand} \log d$ linear forms in $\text{Lin}(T_1) \setminus \mathcal{L}_{s1}$ by intersection of kernels of $\mathcal{S}_2$ spaces.

- **Case 3:** When $\dim(\text{span}(\gcd(T_2, T_3))) < c_2 + 2$:

  The discussion from the previous case follows in this case as well and we can reduce to the case where $\dim(\text{span}(\mathcal{L}_{s1})) < c_2$. Also, $\dim(\text{span}(\text{Lin}(T_1))) \geq 15c_{cand} \log d - 2c_2 - 2$. We will show how to learn linear forms in T_1 using the fact that in this case the number of essential variables in $T_2 + T_3 < 2c_2 + 2$, while the number of essential variables in T_1 is large. The solution is essentially the same as in the case B of Lemma 6.14, and in fact it is even simpler as we already have $\dim(\text{span}(\mathcal{L}_{s1})) \leq c_2$.

  To solve this case, we use Lemma 3.12, which shows that the number of essential variables is equal to rank of the partial derivative matrix. Using this, we are able to set up a system of polynomial equations to compute linear forms l such that $C \mod l$ has less than $2c_2 + 2$ essential variables. Moreover we show that these linear forms will be precisely the linear forms in T_1. The solution is similar to Lemma 6.11 and case B of Lemma 6.14.

  Clearly, if $l | T_1$, then $C \mod l = G \times (T_2 + T_3) \mod l$, and $C \mod l$ has less than $2c_2 + 2$ essential variables. Let Φ be a random linear isomorphism such that $\Phi(x_i) = \sum_{j=1}^{n} \alpha_{i,j} x_j$ for $\alpha_{i,j}$ chosen from $[d^n]$ and $g = \Phi(f) = f(\Phi(x))$. Also, for $t = 10c_2 + 1$ and $i \in [t, n]$ obtain polynomials $g_i = g|_{x_t=\ldots=x_{i-1}=x_{i+1}=\ldots=x_n=0}$. Let $a_2, \ldots, a_n$ be formal variables. Substitute $x_1 = a_2 x_2 + \ldots + a_t x_t + a_i x_i$ into each g_i. We can interpolate and obtain monomial access to the polynomials g_i and their partial derivatives. For each g_i we do the following. We know the rank of the partial derivative matrix is equal to the number of essential variables. We want to solve for linear forms which make this number smaller than $2c_2 + 2$. Therefore we add equations specifying that all $(2c_2 + 2) \times (2c_2 + 2)$ minors of the partial derivative matrix corresponding to g_i after substitution of $x_1 = a_2 x_2 + \ldots + a_t x_t + a_i x_i$ will be 0. This gives us a system of equations whose solutions (we argue in the next paragraph that the system can be solved) will contain the projection of l. We can then glue these projections learnt across the different g_i based on the first t coordinates similar to Lemma 6.11 (indeed this is the step for which we need the random isomorphism), to get linear forms in T_1.

  The main thing we need to argue still is that the set of solutions of each of the systems of equations is small as we can then use Theorem 3.8 to obtain the set of solutions in $\text{poly}(d)$ time. For this we observe that the only linear forms l for which $g_i \mod l$ has $< 2c_2 + 2$ essential variables are precisely the linear forms in T_1 projected down to $x_1, \ldots, x_{t-1}, x_i$ (and hence there are at most d of them). Since we fix the linear forms to have the coefficient of x_1 as 1, there is exactly 1 solution per linear form of T_1. If possible there is a linear form $l \nmid T_1$ for which $g_i \mod l$ has less than $2c_2 + 2$ essential variables. This cannot happen as T_1 after the random restriction, with high probability has t essential variables, and if $l \nmid T_1$, $T_1 \mod l \neq 0$ and hence after going mod l, the projected T_1 must have at least $t - 1$ essential variables. Moreover, $T_2 + T_3$ has less than $2c_2 + 2$ essential variables and after projection it will continue to have less than $2c_2 + 2$ essential variables. Since the sum of a polynomial with a high number of essential variables and small essential variables cannot have a small number of essential variables, therefore, only linear forms that divide T_1 after restriction will be part of the solutions.

□

6.4.3 Candidate Linear forms for Special form 3

In this case, the circuit is of form

$$C = G \times (T_1 + T_2 + T_3) = G \times \left(l \cdot T_1' + \prod_{i=1}^{d'} (l_i + \alpha_i l) T_2' + \prod_{i=1}^{d'} (l_i + \beta_i l) T_3' \right)$$

where $\operatorname{rank}(\operatorname{sim}(T_2' + T_3')) < c_2$, but $\operatorname{rank}(\operatorname{sim}(T_2 + T_3)) \geq c_2$.

Lemma 6.18. *Let f be a polynomial that can be computed by a circuit $C = G \times (T_1 + T_2 + T_3)$ in special form 3 as defined in Definition 12 satisfying all assumptions from Theorem 6.2, then there exists an algorithm that computes a list of linear forms $\mathcal{L}_{cand}$ in $\operatorname{poly}(n, d)$ time such that $|\mathcal{L}_{cand}| = d^{O(1)}$ and there exists an $i \in [3]$ such that $\dim(\operatorname{span}(\mathcal{L}_{cand} \cap \operatorname{Lin}(T_i))) \geq c_{cand} \log d$.*

Proof. We will break up our analysis into several cases. The analyses of many of the cases closely follow previous analyses of cases when the circuit was not in special form. The most involved case which we handle last will correspond to the setting where there is only one linear form l which results in the structure of the circuit being a special form circuit, and $\dim(\operatorname{span}(\operatorname{Lin}(T_1') \setminus \mathcal{L}_{s1})) \geq 2$ and $\dim(\operatorname{span}(\{l_i\}_{i \in [d']} \setminus \mathcal{L}_{s1})) \geq \mathcal{R}(6) + 2c_2 + 2$.

- **Case A:** $\dim(\operatorname{span}(\gcd(T_2', T_3'))) \geq 5c_{cand} \log d$.
  - **Case A.1** $\dim(\operatorname{span}(\operatorname{Lin}(T_1) \setminus \mathcal{L}_{s1})) \geq 2$.
    From Lemma 6.4 and assumption on G from Theorem 6.2, we have that $\dim(\operatorname{span}(\mathcal{L}_{s1})) \leq 12\mathcal{R}(3) \log d$. Therefore, $\dim(\operatorname{span}(\gcd(T_2', T_3') \setminus \operatorname{span}(\mathcal{L}_{s1}))) \geq (5c_{cand} - 12\mathcal{R}(3)) \log d > c_{cand} \log d + 2$. Let l_1, l_2 be two independent linear forms that lie in $\operatorname{span}(\operatorname{Lin}(T_1) \setminus \mathcal{L}_{s1}$. Then for every linear form $l' \in \gcd(T_2', T_3') \setminus \operatorname{span}(\mathcal{L}_{s1}, l_1, l_2)$, there are at least 2 distinct codimension 2 spaces in $\mathcal{S}_2(f)$ $\mathbb{V}(l', l_1)$ and $\mathbb{V}(l', l_2)$. Therefore, we learn $c_{cand} \log d$ linear forms in $\gcd(T_2', T_3')$ by intersection of kernels of $\mathcal{S}_2$ spaces.
  - **Case A.2** $\dim(\operatorname{span}(\operatorname{Lin}(T_1) \setminus \mathcal{L}_{s1})) < 2$.
    This case has been handled in Lemma 6.13 where we handle the case when there are 2 high-rank gates and for some gate $\dim(\operatorname{span}(\operatorname{Lin}(T_i) \setminus \operatorname{span}(\mathcal{L}_{s1}))) \leq 1$. Note that in Lemma 6.13, we do not use $\mathcal{S}_3(f)$ and therefore the condition that $\operatorname{rank}(\operatorname{sim}(C \bmod l)) < c_2$ doesn't affect it.
- **Case B:** $\dim(\operatorname{span}(\gcd(T_2', T_3'))) < 5c_{cand} \log d$.
  In this case, at least one of $\operatorname{Lin}(T_1)$ or $\{l_i\}_{i \in [d']}$ must have dimension at least $5c_{cand} \log d$ (and in particular their sum must have dimension at least $10c_{cand} \log d$) as the simple part of the circuit C has rank at least $15c_{cand} \log d$.
  - **Case B.1:** $\dim(\operatorname{span}(\{l_i\}_{i \in [d']} \setminus \operatorname{span}(\mathcal{L}_{s1}))) < \mathcal{R}(6) + 2c_2 + 2$.
    In this case, we know using Lemma 6.4 that $\dim(\operatorname{span}(\{l_i\}_{i \in [d']})) < 12\mathcal{R}(3) \log d + \mathcal{R}(6) + 2c_2 + 2$, which means $\dim(\operatorname{span}(\operatorname{Lin}(T_1))) \geq 10c_{cand} \log d - 12\mathcal{R}(3) \log d - (\mathcal{R}(6) + 2c_2 + 2)$.
    - **Case B.1.a:** $\dim(\operatorname{span}(\gcd(T_2', T_3') \setminus \operatorname{span}(\mathcal{L}_{s1})) \geq 2$.
      From Lemma 6.4 and assumption on G from Theorem 6.2, we have that $\dim(\operatorname{span}(\mathcal{L}_{s1})) \leq 12\mathcal{R}(3) \log d$. Therefore, $\dim(\operatorname{span}(\operatorname{Lin}(T_1) \setminus \mathcal{L}_{s1})) \geq (10c_{cand} - 24\mathcal{R}(3)) \log d - (\mathcal{R}(6) + 2c_2 + 2) > c_{cand} \log d$. Let l_1, l_2 be two independent linear forms that lie in $\gcd(T_2', T_3') \setminus \mathcal{L}_{s1}$. Then for every linear form $l' \in \operatorname{Lin}(T_1) \setminus \operatorname{span}(\mathcal{L}_{s1}, l_1, l_2)$, there are at least 2 distinct codimension 2 spaces in $\mathcal{S}_2(f)$ $\mathbb{V}(l', l_1)$ and $\mathbb{V}(l', l_2)$. Therefore, we learn $c_{cand} \log d$ linear forms in $\operatorname{Lin}(T_1)$ by the intersection of kernels of $\mathcal{S}_2$ spaces.

* **Case B.1.b:** $\dim(\text{span}(\gcd(T_2', T_3') \setminus \text{span}(\mathcal{L}_{s1})) < 2$.
In this case, we have $\dim(\text{span}(\text{Lin}(T_2) \cup \text{Lin}(T_3) \setminus \text{span}(\mathcal{L}_{s1}))) < \mathcal{R}(6) + 3c_2 + 4$, which same as case B of Lemma 6.14. Note that in case B of Lemma 6.14, we do not use $\mathcal{S}_3(f)$ and therefore the condition $\text{rank}(\text{sim}(C \mod l)) < c_2$ doesn't affect it. We know from Lemma 3.4 that $\mathcal{R}(6) + 3c_2 + 4$ is a constant. Therefore, the constant in case B of Lemma 6.14 can be replaced with $\mathcal{R}(6) + 3c_2 + 4$ instead of c_2, as it doesn't affect the analysis because T_1 has $\Omega(\log d)$ rank.

– **Case B.2:** $\dim(\text{span}(\{l_i\}_{i\in[d']} \setminus \text{span}(\mathcal{L}_{s1}))) \geq \mathcal{R}(6) + 2c_2 + 2$.

* **Case B.2.a:** $\dim(\text{span}(\text{Lin}(T_1) \setminus \mathcal{L}_{s1})) < 2$.
From Lemma 6.4 and assumption on G from Theorem 6.2, we have that $\dim(\text{span}(\mathcal{L}_{s1})) \leq 12\mathcal{R}(3)\log d$. Therefore, $\dim(\text{span}(\text{Lin}(T_1))) < 12\mathcal{R}(3)\log d + 2$ and $\dim(\text{span}(\{l_i\}_{i\in[d']})) \geq 10c_{cand}\log d - (12\mathcal{R}(3)\log d + 2)$. Therefore, this case is handled by Lemma 6.13 as well.
* **Case B.2.b:** $\dim(\text{span}(\text{Lin}(T_1) \setminus \mathcal{L}_{s1})) \geq 2$.
This is last case left and the rest of the proof is devoted to understanding this case. Note that in this case we can learn linear forms l that give the circuit structure of special form 3 using intersection of kernels of $\mathcal{S}_2$ spaces. Fix one such l. To see this, let l_1, l_2 be two independent linear forms in $\{l_i\}_{i\in[d']} \setminus \text{span}(\mathcal{L}_{s1}, l)$ (where $\{l_i\}_{i\in[d']}$ are the linear forms as defined in the structure of special form 3) such that $\dim(\text{span}(l, l_1, l_2)) = 3$. Then $\mathbb{V}(l, l_1)$ and $\mathbb{V}(l, l_2)$ are two distinct codimension 2 spaces in $\mathcal{S}_2(f)$, whose kernels intersect in l and hence l can be learnt. Also, note that from the definition of Special form, it follows that $C \mod l \neq 0$. Thus once, we have l, we also can learn the set $\{l_i\}_{i\in[d']}$ by factoring $C \mod l$.

Case B.2.b: Let us again remind the reader the parameters of this case. We are in the setting where $\dim(\text{span}(\text{Lin}(T_1) \setminus \mathcal{L}_{s1})) < 2$ and $\dim(\text{span}(\{l_i\}_{i\in[d']} \setminus \text{span}(\mathcal{L}_{s1}))) \geq \mathcal{R}(6) + 2c_2 + 2$. Also $\dim(\text{span}(\gcd(T_2', T_3'))) < 5c_{cand}\log d$. We will first analyze the (easier) subcases where there are at least two distinct linear forms which give rise to the special form structure. Note that if any linear form gave rise to a Special form 2 structure, then we already handled this case when we handled Special form 2.

Case B.2.b with two linear forms giving special form structure Let l_a and l_b be two independent linear forms such that there are circuits C_a and C_b computing f with $\text{rank}(\text{sim}(C_a \mod l_a)) < c_2$, $C_a \mod l_a \neq 0$, and $\text{rank}(\text{sim}(C_b \mod l_b)) < c_2$, $C_b \mod l_b \neq 0$. We will divide it into two cases, one where both C_a and C_b are in Special form 3, and one where C_a is in special form 3 and C_b is in Special form 1. The case where there are only circuits for f in Special form 1, has already been handled in Lemma 6.15.

Case 1: There are two linear forms which both give rise to Special form 3 circuits.
Let

$$C_a = G_a \times (T_{1a} + T_{2a} + T_{3a}) = G_a \times \left(l_a \cdot T'_{1a} + \prod_{i=1}^{d'} (l_{ia} + \alpha_{ia} l_a) T'_{2a} + \prod_{i=1}^{d'} (l_{ia} + \beta_{ia} l_a) T'_{3a} \right)$$

and

$$C_b = G_b \times (T_{1b} + T_{2b} + T_{3b}) = G_b \times \left(l_b \cdot T'_{1b} + \prod_{i=1}^{d'} (l_{ib} + \alpha_{ib} l_b) T'_{2b} + \prod_{i=1}^{d'} (l_{ib} + \beta_{ib} l_b) T'_{3b} \right).$$

The main observation is that we can learn l_a and l_b (at least as part of a larger set) and also learn the l_{ia} and the l_{ib}. Moreoever by rank bounds, since the two circuits compute the same polynomial, the gates of C_a must be "close" to the gates of C_b. From this we can show that we can glue together to recover at least two of the linear forms from one of the gates of C_a. Once we have this, we can go mod these linear forms to get two different projections of another gate, which can again be glued to recover enough linear forms. We elaborate on this strategy below.

Since we know that C_a and C_b are computing the same polynomial, $C_a - C_b$ computes identity. By Lemma 3.4, we have that $\text{rank}(\text{sim}(C_a - C_b)) < \mathcal{R}(6)$ where $\mathcal{R}(6)$ is a constant for $\mathbb{F} = \mathbb{R}$ or $\mathbb{C}$. Then one of T_{2a} or T_{3a}, should be close to one T_{2b} or T_{3b}. Wlog let T_{2a} be close to T_{2b}, i.e. $\text{rank}(\text{sim}(G_a \times T_{2a} + G_b \times T_{2b})) < \mathcal{R}(6)$. Note $\mathcal{L}_{s1}$ is a set defined by the polynomial f and not the circuit representation. Therefore, $\text{Lin}(G_a) \subseteq \mathcal{L}_{s1}$ and $\text{Lin}(G_b) \subseteq \mathcal{L}_{s1}$. Thus, in $G_a \times T_{2a}$ and $G_b \times T_{2b}$ all linear forms, except those that lie in $\dim(\text{span}(\mathcal{L}_{s1})) + \mathcal{R}(6)$ dimensional space, will be the same. This means all $(l_{ia} + \alpha_{ia} l_a) \in \text{Lin}(T_{2a})$, except those in $\dim(\text{span}(\mathcal{L}_{s1})) + \mathcal{R}(6) + 2c_2$ dimensional space, are identical up to scaling to some $(l_{jb} + \alpha_{jb} l_b) \in \text{Lin}(T_{2b})$. We already know for both circuits $\dim(\text{span}(\{l_{ia}\}_{i\in[d']} \setminus \text{span}(\mathcal{L}_{s1}))) \geq \mathcal{R}(6) + 2c_2 + 2$ and $\dim(\text{span}(\{l_{ib}\}_{i\in[d']} \setminus \text{span}(\mathcal{L}_{s1}))) \geq \mathcal{R}(6) + 2c_2 + 2$ from discussion above. Now we consider two cases. The first one if when $\dim(\text{span}(\{l_{ia}\}_{i\in[d']} \setminus \text{span}(\mathcal{L}_{s1}))) \geq c_{cand} \log d + \mathcal{R}(6) + 2c_2 + 2$ and $\dim(\text{span}(\{l_{ib}\}_{i\in[d']} \setminus \text{span}(\mathcal{L}_{s1}))) \geq c_{cand} \log d + \mathcal{R}(6) + 2c_2 + 2$. The second case is when one of the two is smaller than $c_{cand} \log d + \mathcal{R}(6) + 2c_2 + 2$. In the first case for every pair of potential linear forms where $(l_{ia} + \alpha_{ia} l_a) = (l_{jb} + \alpha_{jb} l_b)$ we can learn these identical linear forms as follows. Note that we already have learnt l_{ia}, l_a, l_{jb}, l_b as part of a larger set. We can solve a system of linear equations to recover α_{ia} and α_{jb}. We do this for every possible choice of l_{ia}, l_a, l_{jb}, l_b to recover at least $c_{cand} \log d$ independent linear forms from T_{2a}.

In the second case, one of the circuits (wlog C_a) has a small rank $\{l_i\}_{i\in[d']}$. However we know the rank is still at least $\mathcal{R}(6) + 2c_2 + 2$, which means we will be able to learn at least 2 independent linear forms $(l_{ia} + \alpha_{ia} l_a)$ and $(l_{ja} + \alpha_{ja} l_a)$ from T_{2a}. Also, in this case $\text{span}(\text{Lin}(T'_{1a}))$ must have dimension at least $9c_{cand} \log d - \mathcal{R}(6) + 2c_2 + 2$ as the rank of the simple part of C_a is at least $15c_{cand} \log d$. In this case, we can reconstruct C_a mod $(l_{ia} + \alpha_{ia} l_a)$ and $(l_{ja} + \alpha_{ja} l_a)$ to get 2 independent projections of the linear forms in $\text{Lin}(T'_{1a})$(Algorithm 9), which we can glue back to get linear forms in $\text{Lin}(T'_{1a})$, similar to case A of Lemma 6.14. Therefore, we have $c_{cand} \log d$ independent linear forms from a gate.

Case 2: One linear form that makes the circuit of form Special form 3, and one linear form that makes the circuit into Special form 1

Let

$$C_a = G_a \times (T_{1a} + T_{2a} + T_{3a}) = G_a \times \left(l_a \cdot T'_{1a} + \prod_{i=1}^{d'} (l_{ia} + \alpha_{ia} l_a) T'_{2a} + \prod_{i=1}^{d'} (l_{ia} + \beta_{ia} l_a) T'_{3a} \right)$$

be the circuit that is in Special form 3, and

$$C_b = G_b \times (T_{1b} + T_{2b} + T_{3b}) = G_b \times \left(\left(\prod_{i=1}^{d'} (l_{ib} + \alpha_{ib} l_b) \right) T'_{1b} + \left(\prod_{i=1}^{d'} (l_{ib} + \beta_{ib} l_b) \right) T'_{2b} + \left(\prod_{i=1}^{d'} (l_{ib} + \gamma_{ib} l_b) \right) T'_{3b} \right)$$

be the circuit in Special form 1 computing the same polynomial f. Recall that in Lemma 6.15, we argued that $\{l_{ib}\}_{i\in[d']}$ spans a space of dimension at least $15c_{cand} \log d - c_2 - 1$. As the two circuits compute the same polynomial, $C_a - C_b$ computes identity and by rank bounds in Lemma 3.4, we have $\text{rank}(\text{sim}(C_a - C_b)) < \mathcal{R}(6)$. Then T_{2a} must be close to one of T_{1b}, T_{2b}, T_{3b}(wlog T_{1b}), i.e.

$\text{rank}(\text{sim}(G_a \times T_{2a} + G_b \times T_{1b})) < \mathcal{R}(6)$. Thus, in $G_a \times T_{2a}$ and $G_b \times T_{1b}$ all linear forms, except those that lie in $\dim(\text{span}(\mathcal{L}_{s1})) + \mathcal{R}(6)$ dimensional space, will be the same. The rest of the proof remains the same as Case 1, but is even easier to analyze as we know $\dim(\text{span}(\{l_{ib}\}_{i\in[d']})) \geq 15c_{cand}\log d - c_2 - 1$, and hence $\dim(\text{span}(\{l_{ia}\}_{i\in[d']})) \geq 10c_{cand}\log d - \mathcal{R}(6) - 2c_2 - 1$, and hence at least $c_{cand}\log d$ linear forms of the form $(l_{ia} + \alpha_{ia}l_a)$ can be learned from T_{2a}.

Case B.2.b with unique linear form giving special form structure. We are now only left with the case when we are in Case B.2.b and there is a single linear form l for which $\text{rank}(\text{sim}(C \mod l)) < c_2$ and $C \mod l \neq 0$ and $l|T_1$. Our approach in this case remains similar to our approach in Special form 1. Firstly, as discussed earlier we can find l in the intersection of $\mathcal{S}_2$ spaces of f and l_i's from factors of $C \mod l$.

Define $\mathcal{S}_3^*(f)$ to be the set of spaces of the form $\mathbb{V}(l', l_i + \alpha l, l_j + \beta l)$ such that l_i, l_j are in the set of linear factors $C \mod l$ and $\alpha, \beta \in \mathbb{F}$, $l \notin \text{span}(l', l_i + \alpha l, l_j + \beta l)$ and $\mathbb{V}(l', l_i + \alpha l, l_j + \beta l) \in \mathcal{S}_3(f)$.

We see that for $l' \in \text{Lin}(T_1')$ such that $l' \notin \text{span}(l)$ and l_i, l_j such that $\dim(\text{span}(l_i, l_j, l, l')) = 4$, $\mathbb{V}(l', l_i + \alpha_i l, l_j + \beta_j l)$ will be in $\mathcal{S}_3^*(f)$. Note that one such l' exists as $\dim(\text{span}(\text{Lin}(T_1) \setminus \text{span}(\mathcal{L}_{s1}))) \geq 2$ and many l_i, l_j exist (for each choice of l') as $\dim(\text{span}(\{l_i\}_{i\in[d']} \setminus \text{span}(\mathcal{L}_{s1}))) \geq \mathcal{R}(6) + 2c_2 + 2$.Our goal is to learn the α_i, β_j since then we can add $l_i + \alpha_i l, l_j + \beta_j l$ to the set $\mathcal{L}_{cand}$.

Claim 6.19. *Assuming, we are in case B.2.b of Special form 3 and there is a unique linear form l giving rise to the special form structure, then $\mathcal{S}_3^*(f)$ can be computed in randomized time* $\text{poly}(n, d)$, *and* $|\mathcal{S}_3^*(f)| = d^{O(1)}$.

Proof. The proof of this claim follows exactly the same idea as Claim 6.16. We will first prove that $|\mathcal{S}_3^*(f)| = d^{O(1)}$. From Lemma 4.4, we have that if rank of the circuit is at least c_3, and there does not exist a linear form l such that $\text{rank}(\text{sim}(C \mod l)) \leq c_2$, then $|\mathcal{S}_3(f)| = d^{O(1)}$. However, in our case, we do have a linear form l such that $\text{rank}(\text{sim}(C \mod l)) \leq c_2$. However though we are not able to bound $|\mathcal{S}_3(f)|$, we will still be able to bound $|\mathcal{S}_3^*(f)|$. To see this, we inspect the proof of Lemma 4.4. Note that in our setting there is a single unique linear form l such that $\text{rank}(\text{sim}(C \mod l)) \leq c_2$. l does divides a gate T_1 of the circuit. Thus the only case where Lemma 4.4 does not allow us to prove a bound on $|\mathcal{S}_3(f)|$ is case 1. We further observe in case 1, we are not able to upper bound the number of $\mathcal{S}_3$ spaces $\mathbb{V}(l_1, l_2, l_3)$ when for the linear form $l \in \text{span}(l_1, l_2, l_3)$ that divides a gate, rank of the simple part of the rest of the circuit mod l is small($< c_2$). In this case, we consider triples of the form $\mathbb{V}(l_1, l_2, l_3)$, and if there is no $l'' \in \text{span}(l_1, l_2, l_3)$ such that $\text{rank}(\text{sim}(C \mod l'')) \leq c_2$ then the number of such triples is bounded. Now, of course in $\mathcal{S}_3(f)$ this condition does not always hold. However, by definition, this condition holds in $\mathcal{S}_3^*(f)$! Also since l is the unique linear form such that $\text{rank}(\text{sim}(C \mod l)) \leq c_2$ thus there is no linear form in $\text{span}\{l', l_i + \alpha l, l_j + \beta l\}$ such that modulo it the rank crashes. Thus we are able to bound the number of codimension 3 spaces in $\mathcal{S}_3^*(f)$ when there is linear form such that $\text{rank}(\text{sim}(C \mod l')) \leq c_2$ and hence we are able to bound the number of these spaces overall.

We now show how to learn the set of spaces in $\mathcal{S}_3^*(f)$. The algorithm is even simpler than that for learning $\mathcal{S}_3(f)$ since we already know l and the various l_i. Thus after projecting to few variables and solving a system of polynomial equations, we can recover the values of all possible α, β and we do not need to glue and lift. We provide the details below.

Pick l_i, l_j from $\{l_i\}_{i\in[d']}$ learnt from factoring mod l such that $l_i, l_j \notin \text{span}(\mathcal{L}_{s1}, l)$. Just like in Lemma 5.2 we consider a random invertible linear transformation Φ and set all but $t = c_3 = O(1)$ (c_3 is as in Lemma 4.4) variables $x_{t+1} = \ldots = x_n = 0$, to obtain g. As Φ is a random linear isomorphism, if f vanishes on $\mathbb{V}(l_1, l_2, l_3)$ then g vanishes on $\mathbb{V}(\Phi(l_1), \Phi(l_2), \Phi(l_3))|_{x_{t+1}=\ldots=x_n=0}$ with high probability.

Therefore, g vanishes on $\mathbb{V}\left(\Phi(l'), \Phi(l_i) + \alpha\Phi(l), \Phi(l_j) + \beta\Phi(l)\right)_{x_{t+1}=\ldots=x_n=0}$. Consider new formal variables $a_1, \ldots, a_t$ and let $l_1 = a_1x_1 + \ldots + a_tx_t$. Let α, β be formal variables, and consider any invertible linear transformation Ψ on $x_1, \ldots, x_t$ such that it takes $x_1 \leftarrow l_1, x_2 \leftarrow \Phi(l_i + \alpha l), x_3 \leftarrow \Phi(l_j + \beta l)$. Consider $\Psi(g)$ after setting $x_1 = x_2 = x_3 = 0$. Set up a system of polynomial equations in $a_1, \ldots, a_t, \alpha, \beta$ by equating the coefficients of monomials in $x_4, \ldots, x_t$ in $\Psi(g)|_{x_1=x_2=x_3=0}$ to zero. Also add equations so that $\mathbb{V}\left(l_1, \Phi(l_i) + \alpha\Phi(l), \Phi(l_j) + \beta\Phi(l)\right)$ is not contained in a $\mathcal{S}_1$ or $\mathcal{S}_2$ space after the projection and Ψ similar to what was done in Lemma 5.6. Also add an equation that ensures $\dim(\text{span}(\Phi(l), l_1, \Phi(l_i) + \alpha\Phi(l), \Phi(l_j) + \beta\Phi(l))) = 4$, similar to adding an equation so that the space is not contained in $\mathbb{V}(l)$ in Lemma 5.6. So, the solutions of the system of equations will contain $l_1 = \Phi(l')|_{x_{t+1}=\ldots=x_n=0}, \alpha = \alpha_i, \beta = \beta_j$ for all required values that determine the set $\mathcal{S}_3^*(f)$.

So, all we need to argue is that the system of equations will have at most $d^{O(1)}$ solutions. As discussed in proof of Lemma 5.4, after a random invertible linear transformation and setting of all except constant variables, with high probability, the circuit will be full-rank, i.e. $\text{rank}(\text{sim}(\Phi(C)|_{x_{t+1}=\ldots=x_n=0})) = t$. Moreover, by Lemma 5.7, there will remain a unique linear form such that modulo it the circuit has rank at most c_2. Thus even after the random linear transformation and projection, the number of $\mathcal{S}_3^*$ spaces of the new polynomial are still $d^{O(1)}$. It is easy to see that each solution of the system of equations we set up corresponds to a distinct $\mathcal{S}_3^*$ space of the projected polynomial. Therefore, the number of solutions of the system of equations will be $d^{O(1)}$, and hence we can efficiently find all solutions.

□

We now need to show that we can learn $c_{cand} \log d$ independent linear forms from a gate, and the analysis is basically identical to the union of the analyses of Lemma 6.15, Case 2 of Lemma 6.10 and Case A of Lemma 6.14 except that we use $\mathcal{S}_3^*(f)$ which we computed instead of $\mathcal{S}_3(f)$ (which we do not know how to bound).

Now, we want to show that we can learn enough linear forms from some gate from our computation of $\mathcal{S}_3^*(f)$. Interestingly, if $\mathbb{V}(l', l_i + \alpha l, l_j + \beta l)$ is a $\mathcal{S}_3^*(f)$ space, then we immediately learn two linear forms, $(l_i + \alpha l)$ and $(l_j + \beta l)$, one from each of T_2 and T_3. We can also learn linear forms from the intersection of kernels of $\mathcal{S}_2$ spaces since $\mathcal{S}_2(f)$ can be computed in our setting. Observe that we can still also compute $\mathcal{S}_3^{sp}$ spaces just as they were defined and computed in Lemma 6.8, and so can also learn linear forms from the intersection of kernels of spaces in $\overline{\mathcal{S}_3^{sp}}$. We will show that the union of linear forms computed by all these intersections and the computation of $\mathcal{S}_3^*(f)$ spaces will give us all the linear forms we need.

We now break the analysis into 3 cases using $\dim(\text{span}(\{l_i\}_{i\in[d']}))$ and $\dim(\text{span}(\text{Lin}(T_1')))$. Note that both $\dim(\text{span}(\{l_i\}_{i\in[d']}))$ and $\dim(\text{span}(\text{Lin}(T_1')))$ cannot be smaller than $5c_{cand} \log d$ as we know $\dim(\text{span}(\gcd(T_2', T_3')) < 5c_{cand} \log d$ and $\text{rank}(T_1 + T_2 + T_3) \geq 15c_{cand} \log d$.

1. **Case 1:** $\dim(\text{span}(\{l_i\}_{i\in[d']})) \geq 5c_{cand} \log d$ and $\dim(\text{span}(\text{Lin}(T_1'))) \geq 5c_{cand} \log d$.

   This analysis goes exactly similar to the analysis for Special form 1. Fix S to be any maximal set of independent spaces[10] of the form $\text{span}(l_1, l_2, l_3)$ with $l_1 \in T_1, l_2 \in T_2, l_3 \in T_3$ such that $\mathbb{V}(l_1, l_2, l_3)$ is contained in some space $\mathbb{V}(l_1', l_2') \in \mathcal{S}_2^{reg}(f)$, in particular $\text{span}(l_1', l_2') \subset \text{span}(l_1, l_2, l_3)$. We discussed in Lemma 6.10 and Lemma 6.15 that $|S| \leq 12\mathcal{R}(3) \log d$. Let $S' = \{S \cup \mathcal{L}_{s1}\}$ which will have dimension $24\mathcal{R}(3) \log d$. Consider any $l_1 \in \text{Lin}(T_1')$ such that $l_1 \notin \text{span}(S')$. There will be $5c_{cand} \log d - 24\mathcal{R}(3) \log d - 1$ such independent linear forms. Consider $C \mod l_1$, and if $\dim(\text{span}(\text{Lin}(C \mod l_1))) \geq 12\mathcal{R}(3) \log d + 2$ then l_1 is learned by intersection of kernels of two distinct $\mathcal{S}_2(f)$. Therefore, we handled the case when there are $\mathcal{S}_2^{sp}$

[10] where a set of spaces is independent if the dimension of the span of their union is a sum of dimensions of the individual spaces

spaces on which l_1 vanishes. In case, we have $\dim(\text{span}(\text{Lin}(C \mod l_1))) < 12\mathcal{R}(3)\log d + 2$, we also have $\dim(\text{span}(\{l_i\}_{i\in[d']})) \geq 7c_{cand}\log d$ and $\dim(\text{span}(\text{Lin}(T_1))) \geq 7c_{cand}\log d$ as $\gcd(T_2', T_3') \subseteq \text{Lin}(C \mod l_1)$. We consider set $L_2 \subseteq \text{Lin}(\prod_{i=1}^{d'} l_i + \alpha_i l)$ such that any $(l_2 + \alpha_2 l) \in L_2$, $(l_2+\alpha_2 l) \notin \text{span}(\{l, l_1\} \cup S' \cup \text{Lin}(C \mod l_1))$ and $\dim(\text{span}(L_2)) = 3c_{cand}\log d + 2$. Similarly, pick $L_3 \subseteq \text{Lin}(\prod_{i=1}^{d'} l_i + \beta_i l)$ such that any $l_3 + \beta_3 l \in L_3$, $l_3 + \beta_3 l \notin \text{span}(\{l, l_1\} \cup S' \cup \text{Lin}(C \mod l_1) \cup L_2)$ and $\dim(\text{span}(L_3)) = 3c_{cand}\log d + 2$. Now because of our choice of L_2, L_3, we have every space in $\mathcal{S}_3^*$ unless it is contained in a space in $\mathcal{S}_2^{sp}$. If space is contained in $\mathcal{S}_3^*$ then we learn the corresponding linear forms in L_2 and L_3. Therefore the number of spaces l_1 in $\mathcal{S}_3^*$ is less than $c_{cand}\log d$, otherwise we have learned sufficient linear forms. Let the sets be L_2' and L_3' be subsets of L_2 and L_3 respectively such that none of the spaces corresponding to them are in $\mathcal{S}_3^*(f)$ and are contained in a codimension 2 space in $\mathcal{S}_2^{sp}$. We also have $\dim(\text{span}(L_2')) \geq 2c_{cand} + 2$ and $\dim(\text{span}(L_3')) \geq 2c_{cand} + 2$. Now we consider the set of spaces $\mathbb{S}(l_1)$ of the form $\mathbb{V}(l_1, l_2 + \alpha_2 l, l_3 + \beta_3 l)$ where $l_2 + \alpha_2 l \in L_2$ and $l_3 + \beta_3 l \in L_3$. If it is contained in a $\mathcal{S}_2^{sp}$ space then also we learned it in $\mathcal{S}_3^{sp}$ unless one of $l_2 + \alpha_2 l, l_3 + \beta_3 l$ is contained in the kernel of the $\mathcal{S}_2^{sp}$ space, and we call such spaces non-degenerate and we learn all non-degenerate spaces. Therefore, this case is now exactly like the analysis in Lemma 6.15, where we either learn l_1 from enough non-degenerate spaces or there are a lot of degenerate spaces and we learn $c_{cand}\log d$ linear forms from one of L_2' or L_3'. Since, we learn a general l_1(or we are already done), and there are $5c_{cand}\log d - 24\mathcal{R}(3)\log d - 1 > c_{cand}\log d$ such independent linear forms, we learn $c_{cand}\log d$ independent linear forms from T_1.

2. **Case 2:** $\dim(\text{span}(\{l_i\}_{i\in[d']})) < 5c_{cand}\log d$ and $\dim(\text{span}(\text{Lin}(T_1'))) \geq 5c_{cand}\log d$.

   The analysis in this case is similar to case A of Lemma 6.14. In this case, we observe we only need to learn 2 independent linear forms in $\text{Lin}(\prod_{i=1}^{d'}(l_i + \alpha_i l))$ and $\text{Lin}(\prod_{i=1}^{d'}(l_i + \beta_i l))$, then you can just reconstruct the circuit mod these linear forms to get projections of T_1', and therefore learn linear forms in T_1' by gluing these projections as done in case 2 of Lemma 6.10(Algorithm 7). The low rank means from Lemma 6.5 we can conclude that all $\mathcal{S}_2$ spaces are such that at least one of the gates vanishes over them. The gate has to be T_1 as if T_2(or T_3) vanished, then $T_1 + T_3$(or $T_1 + T_2$) will be divisible by a linear form which cannot happen due to the rank gap. Every space in $\mathcal{S}_2^{reg}$ will be of form $\mathbb{V}(l_1, l')$ where $l_1 \in \text{Lin}(T_1)$ and $l'|\text{sim}(T_2+T_3 \mod l_1)$. Since we know $\dim(\text{span}(\gcd(T_2', T_3'))) < 5c_{cand}\log d$ and $\dim(\text{span}(\{l_i\}_{i\in[d']})) < 5c_{cand}\log d$, then we can pick a set of linear forms L_1 such that for every $l_1 \in L_1$, $l_1 \notin \text{span}(\text{Lin}(T_2) \cup \text{Lin}(T_3))$. Consider the first case where $\dim(\text{span}(\text{Lin}(T_2 + T_3) \setminus \text{span}(\mathcal{L}_{s1})) \geq 2$, then we learn every linear form $l_1 \in L_1$ from the intersection of kernels of $\mathcal{S}_2$ spaces and hence have learned $c_{cand}\log d$ independent linear forms from a gate. Therefore, we are left with the case where $\dim(\text{span}(\text{Lin}(T_2 + T_3) \setminus \text{span}(\mathcal{L}_{s1})) < 1$. Let l' be the linear form such that $\text{span}(l') = \text{span}(\text{Lin}(T_2 + T_3) \setminus \text{span}(\mathcal{L}_{s1}))$. Recall we already showed we only need to consider the cases when $\dim(\text{span}(\{l_i\}_{i\in[d']} \setminus \text{span}(\mathcal{L}_{s1})) \geq \mathcal{R}(6) + 2c_2 + 2 \geq 36$. Pick $L_2 \subseteq \text{Lin}(\prod_{i=1}^{d'}(l_i + \alpha_i l))$ such that for any $(l_2 + \alpha_2 l) \in L_2$, $(l_2 + \alpha_2 l) \notin \text{span}(L_1, l', l)$ and $\dim(\text{span}(L_2)) = 4$. Here we are using the fact that L_1 which was picked earlier had all its linear forms not lying in $\text{span}(\text{Lin}(T_2) \cup \text{Lin}(T_3))$. Similarly, we can pick a set of independent linear forms $L_3 \subseteq \text{Lin}(T_3)$ such that for any $(l_3 + \beta_3 l) \in L_3$, $(l_3 + \beta_3 l) \notin \text{span}(L_1, L_2, l', l, \mathcal{L}_{s1})$ and $\dim(\text{span}(L_3)) = 2$. Now, consider the spaces $\mathbb{V}(l_1, (l_2 + \alpha_2 l), (l_3 + \beta_3 l))$ with $l_1 \in L_1, (l_2 + \alpha_2 l) \in L_2, (l_3 + \beta_3 l) \in L_3$. From our choices of L_1, L_2, L_3, these spaces will be in $\mathcal{S}_3^*(f)$ unless they are contained in a $\mathcal{S}_2^{sp}(f)$ spaces. If any of these is contained in $\mathcal{S}_3^*(f)$, then we learn the required 2 independent linear forms and are done. So, we are in the case when all these spaces will be contained in $\mathcal{S}_2^{sp}$ spaces. We will learn these spaces still in $\mathcal{S}_3^{sp}$

and can learn any $l_1 \in L_1$ with the intersections of the kernels(non-degenerate case), unless the linear forms from L_2, L_3 lie in the kernel of $\mathcal{S}_2^{sp}$ spaces(degenerate cases). Now, we are exactly in Case A.2 of Lemma 6.14, where we can learn 2 linear forms from L_2 or L_3 with enough degenerate spaces, or learn linear forms in L_1 from non-degenerate spaces. Thus, we can either learn $c_{cand} \log d$ linear forms from T_1' or learn 2 linear forms from T_2 or T_3, and then learn $c_{cand} \log d$ linear forms from T_1' by gluing the projection mod these linear forms which we obtain by reconstructing the top fan-in 2 circuit.

3. **Case 3:** $\dim(\text{span}(\{l_i\}_{i\in[d']})) \geq 5c_{cand} \log d$ and $\dim(\text{span}(\text{Lin}(T_1'))) < 5c_{cand} \log d$.

   The analysis in this case is similar to case 2 of Lemma 6.10. In this case, we only need to learn 2 linear forms from $\text{Lin}(T_1) \setminus \mathcal{L}_{s1}$, and then we can reconstruct the circuit mod these linear forms, and then glue these projections to get $\text{Lin}(\prod_{i=1}^{d'}(l_i + \alpha_i l))$. Note, we already know a linear form in $\text{Lin}(T_1) \setminus \mathcal{L}_{s1}$, which is l, so we only need 1 more linear form. Now as $\dim(\text{span}(\text{Lin}(T_1'))) < 5c_{cand} \log d$, from Lemma 6.5, we have all linear forms in $\mathcal{L}_{s1}$ will divide at least 1 gate, while $\mathcal{S}_2(f)$ will have spaces on which at least one of the T_i's vanish. If $l|T_2$ (or $l|T_3$), then $l|(T_1 + T_3)$, which is not possible as there is a rank difference between T_1 and T_3. Therefore, all $l \in \mathcal{L}_{s1} \setminus \text{Lin}(G)$ are such that $l|T_1$ and $l|(T_2 + T_3)$. Consider any $\mathcal{S}_2^{reg}$ space such that only T_2 vanishes over it. This means a linear form $l|T_2$ vanishes over this space, which means there is a linear form that divides $\text{sim}(T_1 + T_3) \mod l$ but this cannot happen as there is a difference between the rank of linear forms in the gates. Similarly, we can argue there is no space in $\mathcal{S}_2^{reg}$ such that only T_3 vanishes on it. This means all the spaces in $\mathcal{S}_2^{reg}(f)$ are such that T_1 and $T_2 + T_3$ vanish over it. So, the spaces left in $\mathcal{S}_2$ are those there either all three gates vanish over them or T_3 and $T_1 + T_2$ vanish over them.

   The set of linear forms that divide $\text{sim}(T_2 + T_3)$ lie in a $6\mathcal{R}(3) \log d$ dimensional space using Lemma 6.4. Let this space be S. We define $S' = S \cup (\mathcal{L}_{s1} \setminus \text{Lin}(G))$.

   Consider any $l' \in \text{Lin}(T_1') \setminus \mathcal{L}_{s1}$ such that $l' \notin \text{span}(l)$. We will show that either l' will be in $\mathcal{L}_{cand}$ or else we will manage to find $c_{cand} \log d$ independent linear forms from either $\prod_{i=1}^{d'}(l_i + \alpha_i l))$ or $\prod_{i=1}^{d'}(l_i + \beta_i l))$ that lie in $\mathcal{L}_{cand}$.

   Observe that $C \mod l'$ will be nonzero and of the form $G'' \cdot (T_2'' + T_3'')$ with $\gcd(T_2'', T_3'') = 1$. If $\dim(\text{span}(\text{Lin}(G') \setminus \mathcal{L}_{s1})) \geq 2$ then we have l' in kernel of at least two $\mathcal{S}_2$ spaces and hence will be in $\mathcal{L}_{cand}$ and we are done.

   Now suppose $dim(\text{span}(\text{Lin}(G') \setminus \mathcal{L}_{s1})) \leq 1$. In this case, $\dim(\text{span}(\gcd(T_2', T_3'))) \leq 1$, and therefore $\dim(\text{span}(\{l_i\}_{i\in[d']})) \geq 10c_{cand} \log d - 2 - c_2$.

   We pick a set of independent linear forms $L_2 \subseteq \text{Lin}(\prod_{i=1}^{d'}(l_i + \alpha_i l)))$ from $\text{Lin}(T_2)$ such that $\dim(\text{span}(L_2)) = 3c_{cand} \log d + 2$ and for any $(l_2 + \alpha_2 l) \in L_2$ we have $(l_2 + \alpha_2 l) \notin \text{span}(S' \cup \{l', l\})$ and $((l_2 + \alpha_2 l) \mod l') \in \text{Lin}(T_2'')$. Clearly such a set L_2 can be found. Similarly, we pick $L_3 \subseteq \text{Lin}(\prod_{i=1}^{d'}(l_i + \beta_i l))$ such that $\dim(\text{span}(L_2)) = 3c_{cand} \log d + 2$ and for any $(l_3 + \beta_3 l) \in L_3$ we have $((l_3 + \beta_3 l) \mod l') \in \text{Lin}(T_3'')$ and $(l_3 + \beta_3 l) \notin \text{span}(S' \cup \{l', l\} \cup L_2)$. Such an L_3 can be found (in fact even larger such sets can be found) because $\text{Lin}(\prod_{i=1}^{d'}(l_i + \beta_i l))$ contains at least $(7c_{cand} - 12\mathcal{R}(3)) \log d - 5 - c_2$ independent linear forms after removing linear forms in $\text{span}(S' \cup \{l', l\} \cup L_2)$. Now we are in the same case as Case 1, and we either learn $c_{cand} \log d$ linear forms from L_2 or L_3 from $\mathcal{S}_3^*$ or degenerate spaces or learn l' from non-degenrate spaces and intersection of kernels of $\mathcal{S}_3^{sp}$ spaces.

□

6.5 Candidate Linear forms when some $T_i = \alpha l^d$

Lemma 6.20. *Given polynomial f computed by a circuit $\Sigma\Pi\Sigma(3)$ C of form*

$$C = G \times (T_1 + T_2 + \alpha l^d)$$

such that $\text{rank}(\text{sim}(C)) \geq 15c_{cand} \log d$*, there exists an algorithm that computes a list of linear forms $\mathcal{L}_{cand}$ such that $|\mathcal{L}_{cand}| = d^{O(1)}$ and $\dim(\text{span}(\mathcal{L}_{cand} \cap \text{Lin}(T_1))) \geq c_{cand} \log d$ in time randomized* $\text{poly}(n, d)$ *time.*

Proof. Since $\text{rank}(\text{sim}(C)) \geq 15c_{cand} \log d$, there will be at least one gate(wlog T_1) such that $\dim(\text{span}(\text{Lin}(T_1)) \geq 7c_{cand} \log d$. If $\dim(\text{span}(\text{Lin}(T_2)) \leq c_2 - 1$, we have already handled that case in Special form 2 in Lemma 6.17. If $\dim(\text{span}(\text{Lin}(T_2)) \geq (5c_{cand} - 12\mathcal{R}(3)) \log d$, then we can use Lemma 6.11 to get the required set of candidate linear forms.

Therefore, we are interested in the case when $\dim(\text{span}(\text{Lin}(T_1)) \geq 7c_{cand} \log d$ and $c_2 \leq \dim(\text{span}(\text{Lin}(T_2)) < (5c_{cand} - 12\mathcal{R}(3)) \log d$. As there is a rank gap between any 2 gates, there is no linear form in $\mathcal{L}_{s1} \setminus \text{Lin}(G)$. Therefore, we can also assume $G = 1$ as we can divide by the set of linear factors of C after factoring. The approach in this is similar to Lemma 6.11, as we look for codimension 1 and codimension 2 spaces on which the polynomial has 1 essential variable. From Lemma 6.9, we have that the number of essential variables in $T_1 + T_2$ is at least $\frac{c_2 - \mathcal{R}(3)}{2} \geq 5$ while the number of essential variables in T_3 is 1. We use a random linear isomorphism Φ such that $\Phi(x_i) = \sum_{j=1}^{n} \alpha_{i,j} x_j$ for $\alpha_{i,j}$ chosen randomly from $[d^n]$ and let $g = \Phi(f) = f(\Phi(x))$. Let $c = c_2$ be a constant. For each $i \in [10c, n]$, we obtain polynomials g_i by setting $x_{10c} = \ldots = x_{i-1} = x_{i+1} = \ldots = x_n = 0$ in g. After this, $T_1^{[i]} + T_2^{[i]}$ will continue to be high rank, and will have at least 5 essential variables while $T_3^{[i]}$ have 1 essential variable. From part 2 of Lemma 6.9, we know the only way the polynomial has 1 essential variable is if $l | (T_1 + T_2)$. We can interpolate g_i's and get monomial access to them and the partial derivatives in $\text{poly}(d)$ time. Substitute $x_1 = a_2 x_2 + \ldots + a_{10c} x_{10c} + a_i x_i$ into the polynomials. Therefore, we can form a system of equation mod a linear form for which the polynomial has exactly 1 essential variable and therefore learn $\text{Lin}(T_1^{[i]} + T_2^{[i]})$ containing $\gcd(T_1^{[i]}, T_2^{[i]})$. We then glue these projections to learn $\text{Lin}(T_1 + T_2)$. Since there is a rank gap between T_1 and T_2 there is no linear factor of $T_1 + T_2$, and therefore $\text{Lin}(T_1 + T_2) = \gcd(T_1, T_2)$. Similarly, we also learn $\mathcal{S}_2(T_1 + T_2)$ as we find the $\mathcal{S}_2$ space for g_i's and glue them together as we did in Lemma 6.11. We can further learn linear forms by looking at the intersection of the kernels of the spaces in $\mathcal{S}_2(T_1 + T_2)$. The details of these computations are the same as Lemma 6.13.

We divide the analysis into two parts based on dimension of gcd, $\dim(\text{span}(\gcd(T_1, T_2))) \geq 2$ and $\dim(\text{span}(\gcd(T_1, T_2))) \leq 1$.

In case $\dim(\text{span}(\gcd(T_1, T_2))) \geq 2$, let $l_1, l_2 \in \gcd(T_1, T_2)$ be two independent linear forms. There will be two distinct $\mathcal{S}_2$ spaces $\mathbb{V}(l_1, l)$ and $\mathbb{V}(l_2, l)$ of f whose kernels intersect in l. Therefore, we can learn l by the intersection of kernels of spaces in $\mathcal{S}_2(f)$. Then we consider $C \mod l_1$ equal to $\alpha l^d \mod l$. Since we know l, we can find α and therefore know T_3. We can subtract αl^d from C, and then use the reconstruction algorithm of [Sin16b] for $\Sigma\Pi\Sigma(2)$ circuits to learn a circuit $T_1' + T_2'$ computing the same polynomial $T_1 + T_2$. Due to rank bounds we get that $\text{rank}(\text{sim}(T_1 + T_1')) \leq \mathcal{R}(4)$(or $\text{rank}(\text{sim}(T_1 + T_2')) \leq \mathcal{R}(4)$), and from Lemma 3.4, we have that we learnt in T_1', $7c_{cand} \log d - \mathcal{R}(4) \geq c_{cand} \log d$ independent linear forms in T_1.

In case $\dim(\text{span}(\gcd(T_1, T_2))) \leq 1$, let l_1 be any linear form such that $l_1 \in T_1$ and $l_1 \notin \gcd(T_1, T_2)$. Since $\dim(\text{span}(\text{Lin}(T_2))) \geq c_2$, we have at least two linear forms $l', l'' \in \text{Lin}(T_2)$ such that they are not in $\text{span}(l_1 \cup \gcd(T_1, T_2))$. This means there are two distinct codimension 2 spaces $\mathbb{V}(l_1, l')$ and $\mathbb{V}(l_1, l'')$ in $\mathcal{S}_2(T_1 + T_2)$ learnt above. Therefore we can learn any $l_1 \in T_1$ such that $l_1 \notin \gcd(T_1, T_2)$ through intersection of kernels of $\mathcal{S}_2(T_1 + T_2)$. We already learned the linear

forms in $\gcd(T_1, T_2)$, and therefore, in this case, we can learn the entire $\text{Lin}(T_1)$ which has at least $7c_{cand} \log d$ independent linear forms.

□

7 From a few linear forms to the reconstructing the entire circuit

We saw in Theorem 6.1 that if the polynomial can be computed by a $\Sigma\Pi\Sigma(3)$ circuit $C = G \times (T_1 + T_2 + T_3)$ such that $\text{rank}(\text{sim}(C)) \geq 15c_{cand} \log d$ for some constant $c_{cand} > 36\mathcal{R}(3)$, then we can compute a set of linear forms $\mathcal{L}_{cand}$ such that $|\mathcal{L}_{cand}| = d^{O(1)}$ and there exists a gate T_i, such that $\dim(\text{span}(\mathcal{L}_{cand} \cap \text{Lin}(T_i))) \geq c_{cand} \log d$. In this section, we will discuss how we can reconstruct the entire circuit if we have blackbox access to a $\Sigma\Pi\Sigma(3)$ circuit $C = G \times (T_1 + T_2 + T_3)$ using the set $\mathcal{L}_{cand}$.

We divide the analysis into two cases. The first case is the setting where we can compute $\mathcal{L}_{cand}$ (i.e. $\text{rank}(\text{sim}(C)) \geq 15c_{cand} \log d$), and the other case is where $\text{rank}(\text{sim}(C)) < 15c_{cand} \log d$.

We will divide the case where $\text{rank}(\text{sim}(C)) \geq 15c_{cand} \log d$ into two further subcases based on the distance(see Definition 4) between the gates of the circuit. The first subcase is when all gates are far apart, i.e. $\forall i, j \in [3], \Delta(T_i + T_j) = \text{rank}(\text{sim}(T_i + T_j)) \geq 2\mathcal{R}(4) \log d + 6$, and in this case we learn a $\Sigma\Pi\Sigma(3)$ computing the underlying polynomial (this representation will in fact be unique). The other subcase is when there are two distinct gates T_i, T_j such that $\Delta(T_i + T_j) = \text{rank}(\text{sim}(T_i + T_j)) < 2\mathcal{R}(4) \log d + 6$. In this case we will learn either a $\Sigma\Pi\Sigma(3)$ or a $\Sigma\Pi\Sigma(2, d, 2\mathcal{R}(4) \log d + 6)$ generalized circuit computing the polynomial as defined in Definition 7.

We will prove Theorem 7.1 in this section, from which the proof of Theorem 1.1 follows easily.

Theorem 7.1. *Let $\mathbb{F}$ be a field that is $\mathbb{R}$ or $\mathbb{C}$. Let $f \in \mathbb{F}[x_1, \ldots, x_n]$ be a degree d polynomial computed by $\Sigma\Pi\Sigma(3)$ circuit of the form $C = G \times (T_1 + T_2 + T_3)$ such that $\gcd(T_1, T_2, T_3) = 1$. Let $\mathcal{R}(k)$ be as defined in Theorem 3.4. Let $c_{cand} > max(36\mathcal{R}(3), 6\mathcal{R}(3) + 2\mathcal{R}(4) + 100)$ be any constant. Then, there exists a randomized algorithm that runs in $(nd)^{O(\log d)}$ time, and with probability $1 - o(1)$ does the following:*

1. *If $\text{rank}(\text{sim}(C)) < 15c_{cand} \log d$, then it outputs a $\Sigma\Pi\Sigma(1, d, 15c_{cand} \log d)$ generalized depth 3 circuit computing f.*
2. *If $\text{rank}(\text{sim}(C)) \geq 15c_{cand} \log d$, with $\forall i, j \in [3], i \neq j, \text{rank}(\text{sim}(T_i + T_j)) \geq 2\mathcal{R}(4) \log d + 6$ then it outputs a $\Sigma\Pi\Sigma(3)$ circuit computing f.*
3. *If $\text{rank}(\text{sim}(C)) \geq 15c_{cand} \log d$ and for some gates $T_i \neq T_j$, $\text{rank}(\text{sim}(T_i + T_j)) < 2\mathcal{R}(4) \log d + 6$ then it outputs a $\Sigma\Pi\Sigma(2, d, 2\mathcal{R}(4) \log d + 6)$ generalized depth 3 circuit computing f.*

Proof. In the first case, when $\text{rank}(\text{sim}(C)) < 15c_{cand} \log d$, we use Lemma 7.2 to obtain the $\Sigma\Pi\Sigma(1, d, 15c_{cand} \log d)$ generalized depth 3 circuit computing f. When $\text{rank}(\text{sim}(C)) \geq 15c_{cand} \log d$, we use Theorem 6.1 to get a list of linear forms $\mathcal{L}_{cand}$ in $(nd)^{O(\log d)}$ time such that $|\mathcal{L}_{cand}| = d^{O(1)}$ and $\dim(\text{span}(\text{Lin}(T_1) \cap \mathcal{L}_{cand})) \geq c_{cand} \log d$. If the gates all have a high distance with each other, i.e. $\forall i, j \in [3], i \neq j, \text{rank}(\text{sim}(T_i + T_j)) \geq 2\mathcal{R}(4) \log d + 6$ then use Lemma 7.3 to output a $\Sigma\Pi\Sigma(3)$ circuit computing f. If there are at least two gates that are close to each other, i.e. $\text{rank}(\text{sim}(T_i + T_j)) < 2\mathcal{R}(4) \log d + 6$ then use Lemma 7.4 to output a $\Sigma\Pi\Sigma(2, d, 2\mathcal{R}(4) \log d + 6)$ generalized depth 3 circuit computing f. □

Proof of Theorem 1.1. Let c_{cand} be as in THeorem 7.1. We fix $c > 15 \cdot c_{cand}$. When $\forall i, j \in [3], i \neq j, \text{rank}(\text{sim}(T_i + T_j)) \geq c \log d$, we also have $\text{rank}(\text{sim}(C)) \geq c \log d$,and therefore we reconstruct $\Sigma\Pi\Sigma(3)$ circuit from part 2 of Theorem 7.1. When for some $i, j \in [3], i \neq j, \text{rank}(\text{sim}(T_i + T_j)) <$

$c \log d$, we have two cases, $\text{rank}(\text{sim}(C)) \geq c \log d$ or $\text{rank}(\text{sim}(C)) < c \log d$. In the first case, from part 3 of Theorem 7.1, we learn a $\Sigma\Pi\Sigma(2, d, 2\mathcal{R}(4) \log d + 6)$ circuit which is contained in the class $\Sigma\Pi\Sigma(2, d, c \log d)$ as $c > 2\mathcal{R}(4)$. In the second case, from part 1 of Theorem 7.1, we learn a $\Sigma\Pi\Sigma(1, d, c \log d)$ circuit which is contained in class $\Sigma\Pi\Sigma(2, d, c \log d)$. □

7.1 Low Rank Reconstruction

In this section, we will give a reconstruction algorithm for the case when the $\text{rank}(\text{sim}(C)) < 15c_{cand} \log d$.

Lemma 7.2. *Given black-box access to a $\Sigma\Pi\Sigma(3)$ circuit $C = G \times (T_1 + T_2 + T_3)$ computing a degree d polynomial $f \in \mathbb{F}[x_1, \ldots, x_n]$ with $\text{rank}(\text{sim}(C)) < 15c_{cand} \log d$, there exists an algorithm that runs in time $(nd)^{O(\log d)}$ and with probability $1 - o(1)$ outputs a $\Sigma\Pi\Sigma(1, d, 15c_{cand} \log d)$ generalized depth 3 circuit as defined in Definition 7.*

Proof. The input circuit is of the form $C = G \times (T_1 + T_2 + T_3)$ computing f where $\dim(\text{span}(\{l : l|(T_1 \times T_2 \times T_3)\})) < 15c_{cand} \log d$. Clearly, the non-linear factor of f, $\text{NonLin}(f) = \frac{f}{\prod_{l \in \text{Lin}(f)} l}$ will divide $T_1 + T_2 + T_3$ and therefore, will have essential variables less than $\text{rank}(\text{sim}(C))$. So, we use Lemma 3.7 to get black-box access to $\text{NonLin}(f)$ and the linear factors $\text{Lin}(f)$ in randomized $\text{poly}(n, d)$ time. As $\text{NonLin}(f)$ has at most $\text{rank}(\text{sim}(C))$ essential variables, there exist a linear transformation A such that $\text{NonLin}(f)(A \cdot \overline{x})$ depends only on $\text{rank}(\text{sim}(C))$ variables. Using Theorem 3.11, we can compute A in randomized polynomial time. We can do polynomial interpolation in time $(nd)^{\text{rank}(\text{sim}(C))}$ from Lemma 3.6 to get monomial access to and hence learn $\text{NonLin}(f)(A \cdot \overline{x})$. We use A^{-1} to recover $\text{NonLin}(f)$, and then the circuit by multiplying it with $\text{Lin}(f)$. Notice that this would give us a $\Sigma\Pi\Sigma(1, d, 15c_{cand} \log d)$ circuit computing f. □

7.2 Large Distance Reconstruction

In this case, we assume the input circuit is of form $C = G \times (T_1 + T_2 + T_3)$ such that $\text{rank}(\text{sim}(C)) \geq 15c_{cand} \log d$ and $\forall i, j \in [3], \text{rank}(\text{sim}(T_i + T_j)) \geq 2\mathcal{R}(4) \log d + 6$. In this setting we will show that we can do proper reconstruction. In particular we will give an algorithm that outputs a $\Sigma\Pi\Sigma(3)$ circuit computing the underlying polynomial in $(nd)^{O(\log d)}$ time.

The outline of the algorithm is as follows. Notice that in this setting, using Theorem 6.1, we have a list of linear forms $\mathcal{L}_{cand}$ such that $\dim(\text{span}(\text{Lin}(T_1) \cap \mathcal{L}_{cand})) \geq c_{cand} \log d$ (wlog by relabelling we can assume the gate is T_1). Since, we have $|\mathcal{L}_{cand}| = d^{O(1)}$, we look at all subsets $\mathcal{L} \subset \mathcal{L}_{cand}$ such that $\dim(\text{span}(\mathcal{L})) = c_{cand} \log d$ and at least one of these subsets would be such that $\mathcal{L} \subseteq \text{Lin}(T_1)$. We will reconstruct the projections of $T_2 + T_3$ mod these linear forms and then glue them back to reconstruct T_2.

Algorithm 10 Reconstruction when distance is large

Input: Black-box access to Circuit $\Sigma\Pi\Sigma(3)$ with distance between gates $\geq 2\mathcal{R}(4)\log d + 6$, list of Linear forms $\mathcal{L}_{cand}$ from Theorem 6.1

function *Reconstruction*$(f)(C, \mathcal{L}_{cand})$
  for $\mathcal{L} \subseteq \mathcal{L}_{cand}$ with $\dim(\text{span}(\mathcal{L})) = c_{cand}\log d$ **do**
    Projections = [].
    for $l \in \mathcal{L}$ **do**
      Run the reconstruction Algorithm of Theorem 3.10 for $\Sigma\Pi\Sigma(2)$ circuits for $C \mod l$
      If the Algorithm returns a valid circuit, append it to Projections
    For all $G_i(T_{2i} + T_{3i}) \mod l_i$ guess which gate is projection of T_2 and run the following for all possible choices.
    Use gluing algorithm from Theorem 3.15 to reconstruct $G \times T_2$ up to a constant. Compare with projections to obtain T_2 exactly.
    Run the reconstruction algorithm of Theorem 3.10 on $f' = C - G \times T_2$.
    If the output is a $\Sigma\Pi\Sigma(2)$ circuit C', use PIT to check if $C' + G \times T_2 = C$. If yes, output $C' + G \times T_2$.

Lemma 7.3. *Let $\mathcal{R}(k)$ be as defined in Theorem 3.4. Given black-box access to a $\Sigma\Pi\Sigma(3)$ circuit $C = G \times (T_1 + T_2 + T_3)$ computing a degree d polynomial $f \in \mathbb{F}[x_1, \ldots, x_n]$ with $\text{rank}(\text{sim}(C)) \geq 15c_{cand}\log d$, with $\forall i, j \in [3], i \neq j, \text{rank}(\text{sim}(T_i + T_j)) \geq 2\mathcal{R}(4)\log d + 6$ and a list of linear forms $\mathcal{L}_{cand}$ such that $|\mathcal{L}_{cand}| = d^{O(1)}$, $\dim(\text{span}(\text{Lin}(T_1) \cap \mathcal{L}_{cand})) \geq c_{cand}\log d$, then there exists an algorithm(Algorithm 10) that runs in randomized $(nd)^{O(\log d)}$ time and with $1 - o(1)$ probability, outputs a $\Sigma\Pi\Sigma(3)$ circuit computing f.*

Proof. The main idea is that from the list of linear forms from Theorem 6.1, we can iterate over all subsets of $\mathcal{L}_{cand}$ size $c_{cand}\log d$ till we hit upon the correct set of linearly independent $c_{cand}\log d$ linear forms which we call $\mathcal{L}$ such that $\mathcal{L} \subseteq \text{Lin}(T_1)$. From assumption, we have that $\text{rank}(\text{sim}(T_2 + T_3)) \geq 2\mathcal{R}(4)\log d + 6$.

Using Lemma 6.3 which follows from [KS09a] it follows that the linear forms modulo which $T_2 + T_3$ become nonzero and low rank (less than $\mathcal{R}(4)$) will all lie in a low dimensional space$(2\mathcal{R}(4)\log d)$. Moreover the number of linear forms in $\mathcal{L}$ such that modulo it $T_2 + T_3$ becomes identically zero is at most $6\mathcal{R}(3)\log d$ from Lemma 6.4.

Thus we can conclude that for most linear forms l in $\mathcal{L}$ (all but $2\mathcal{R}(4)\log d + 6\mathcal{R}(3)\log d$), $T_2 + T_3$ stays high rank (rank at least $\mathcal{R}(4)$) after going modulo l. We can then reconstruct $G \times (T_2 + T_3)$ mod l as a $\Sigma\Pi\Sigma(2)$ circuit using Theorem 3.10. In fact, this reconstruction will be the unique representation using rank bounds Lemma 3.4. Since we can do this for most choices of $l \in \mathcal{L}$, this gives us enough $((c_{cand} - 2\mathcal{R}(4) - 6\mathcal{R}(3))\log d > 100\log d)$ projections of the gate $G \times T_2$, that we can glue using Theorem 3.15 to obtain $G \times T_2$ up to a constant. To figure out the constant, we compare the glued $G \times T_2$ with the projections learned mod l, to learn $G \times T_2$ exactly.

We subtract $G \times T_2$ from C and reconstruct $G \times (T_1 + T_3)$ as a $\Sigma\Pi\Sigma(2)$ circuit (note that from assumption $\text{rank}(\text{sim}(T_1 + T_3)) \geq 2\mathcal{R}(4)\log d + 6$) using Theorem 3.10. We then output $G \times (T_1 + T_2 + T_3)$, after checking it computes the same polynomial as C using PIT. As there are $d^{O(\log d)}$ choices of $\mathcal{L}$, the total running time is $(nd)^{O(\log d)}$

□

7.3 Low Distance Reconstruction

We now are left to handle the case where there are some two gates such that the distance between them is less than $2\mathcal{R}(4)\log d+6$. In this case, we will not do proper learning but instead learn the underlying polynomial as a generalized depth 2 circuit.

Algorithm 11 Reconstruction when distance is small

Input: Black-box access to Circuit $\Sigma\Pi\Sigma(3)$ with distance between terms less than $2\mathcal{R}(4)\log d+6$, list of Candidate Linear forms $\mathcal{L}_{cand}$ from Theorem 6.1

function *Reconstruction*$(f)(C,\mathcal{L}_{cand})$
  for $\mathcal{L}\subseteq\mathcal{L}_{cand}$ with $\dim(\text{span}(\mathcal{L}))=c_{cand}\log d$ **do**
    Projections $=[]$.
    for $l_i\in\mathcal{L}$ **do**
      Learn $C \bmod l_i$ using low rank reconstruction in Lemma 7.2 as $\Sigma\Pi\Sigma(1,d,2\mathcal{R}(4)\log d+6)$ circuit
      Let the learnt circuit be $C_i=\left(\prod_{j=1}^{d_i}l_{ij}\right)\cdot h_i(\overline{l_{i1}},\ldots,\overline{l_{ir}})$ with $r\le 2\mathcal{R}(4)\log d+6$
    Use Gluing algorithm from Theorem 3.15 to glue the $\prod_{j=1}^{d_i}l_{ij}$ parts and obtain $G\cdot\prod_{j=1}^{d}l_j$.
    Take the largest set of linear forms in $\mathcal{L}$ for which r is same. Fix a linear form $l_0\in\mathcal{L}$ and r.
    Find a l_i in the set such that $\overline{l_{01}},\ldots,\overline{l_{0r}},l_i$ are all independent.
    Glue h_0 and h_i using Lemma 3.16 to learn the $h(l_1,\ldots,l_r)$
    Factorize $C-\left(G\cdot\prod_{j=1}^{d}l_j\right)\cdot h(l_1,\ldots,l_r)$ to learn $G\times T_1$.
    Use PIT to check if $G\times(T_1+\left(\prod_{j=1}^{d}l_j\right)\cdot h(l_1,\ldots,l_r))=C$. If yes, output $G\times(T_1+\left(\prod_{j=1}^{d}l_j\right)\cdot h(l_1,\ldots,l_r))$.

Lemma 7.4. *Let $\mathcal{R}(k)$ be as defined in Theorem 3.4. Given black-box access to a $\Sigma\Pi\Sigma(3)$ circuit $C=G\times(T_1+T_2+T_3)$ computing polynomial f with $\text{rank}(\text{sim}(C))\ge 15c_{cand}\log d$, with two gates T_i,T_j such that $\text{rank}(\text{sim}(T_i+T_j))<2\mathcal{R}(4)\log d+6$ and a list of candidate linear forms $\mathcal{L}_{cand}$ such that $|\mathcal{L}_{cand}|=d^{O(1)}$, $\dim(\text{span}(\text{Lin}(T_1)\cap\mathcal{L}_{cand}))\ge c_{cand}\log d$, then there exists an algorithm(Algorithm 11) that runs in randomized time $(nd)^{O(\log d)}$ time and with $1-o(1)$ probability, outputs a $\Sigma\Pi\Sigma(2,d,2\mathcal{R}(4)\log d+6)$ circuit computing f .*

Proof. Similar to Lemma 7.3, we can iterate over all subsets $\mathcal{L}$ of $\mathcal{L}_{cand}$ with $c_{cand}\log d$ linear independent linear forms till we hit upon a set $\mathcal{L}\subseteq\text{Lin}(T_1)$. We divide the analysis into 2 parts.

First, the gates that are close are either T_1 and T_2 or T_1 and T_3. Wlog, let it be T_1 and T_2, i.e. $\text{rank}(\text{sim}(T_1+T_2))<2\mathcal{R}(4)\log d+6$. In this since, $\text{rank}(T_1+T_2+T_3)\ge 15c_{cand}\log d$, we have $\text{rank}(\text{sim}(T_2+T_3))\ge(15c_{cand}-4\mathcal{R}(4))\log d-6>2\mathcal{R}(4)\log d+6$. Therefore, in this case, we can learn $G\times T_3$ as in Lemma 7.3. Subtracting $G\times T_3$ gives us blackbox access to $G\times(T_1+T_2)$. Now, we learn $G\times(T_1+T_2)$ as a $\Sigma\Pi\Sigma(1,d,2\mathcal{R}(4)\log d+6)$ circuit from Lemma 7.2 in time $(nd)^{O(\log d)}$. Hence, we get a $\Sigma\Pi\Sigma(2,d,2\mathcal{R}(4)\log d+6)$ circuit computing f.

Now we consider the case where it is T_2 and T_3 that are close, i.e. $\text{rank}(\text{sim}(T_2+T_3))<2\mathcal{R}(4)\log d+6$. Hence, $G\times(T_2+T_3)$ can be written as a $\Sigma\Pi\Sigma(1,d,2\mathcal{R}(4)\log d+6)$ circuit $C'=\left(\prod_{j=1}^{d'}l_j\right)\cdot h(L_1,\ldots,L_r)$ where h does not have linear factors with $r<2\mathcal{R}(4)\log d+6$. We consider the circuit C' mod the linear forms $l_i\in\mathcal{L}$ and learn it as a $\Sigma\Pi\Sigma(1,d,2\mathcal{R}(4)\log d+6)$ circuit using Lemma 7.2 in time $(nd)^{\text{rank}(\text{sim}(T_2+T_3))}=(nd)^{O(\log d)}$. From Lemma 6.4, for at least $(c_{cand}-6\mathcal{R}(3))\log d$ linear forms in $\mathcal{L}$, we are able to learn non-zero $\Sigma\Pi\Sigma(1,d,2\mathcal{R}(4)\log d+6)$

circuits. Therefore, we learn the projections of circuit mod l_i as $(\prod_{j=1}^{d_i} l_{ij} \cdot h_i(L_{i1}, \ldots, L_{ir}))$, where h_i don't have linear factors and $r < 2\mathcal{R}(4)\log d + 6$. In the following claim, we show that for most of the projections $\prod_{j=1}^{d_i} l_{ij}$ will be projections of $\prod_{j=1}^{d'} l_j$ and $h_i(L_{i1}, \ldots, L_{ir}))$ are projections of $h(L_1, \ldots, L_r))$.

Claim 7.5. *Let f be a polynomial in $\mathbb{F}[x_1, \ldots, x_n]$ with no linear factors and k essential variables such that $f = h(l_1, \ldots, l_k)$ for a polynomial $h \in \mathbb{F}[y_1, \ldots, y_k]$ and $l_1, \ldots, l_k$ are linear forms in $\mathbb{F}[x_1, \ldots, x_n]$. Then for any linear forms $l \notin \text{span}(l_1, \ldots, l_k)$, $f \mod l$ will have k essential variables and no linear factors.*

Proof. The property of having linear factors is invariant under linear isomorphism, i.e. for a linear isomorphism Φ, f doesn't have linear factors iff $\Phi(f)$ doesn't have linear factors. Since $l \notin \text{span}(l_1, \ldots, l_k)$, we can consider an isomorphism Φ such that for $i \in [k]$, $x_i \leftarrow l_i$ and $x_{k+1} \leftarrow l$. Now, we have $\Phi(f) = h(x_1, \ldots, x_k)$ and $\Phi(f \mod l) = h(x_1, \ldots, x_k) \mod x_{k+1}$. Now, clearly $h(x_1, \ldots, x_k) \mod x_{k+1}$ will not have a linear factor if $h(x_1, \ldots, x_k)$ doesn't as it doesn't even depend on x_{k+1}. Also, from Lemma 3.13, we have that the representation $h(l_1, \ldots, l_k)$ of f is unique up to $\text{span}(l_1, \ldots, l_k)$. Therefore, $\Phi(f \mod l)$ has linear factors iff $\Phi(f)$ has linear factors. Since, we have from assumption f doesn't have linear factors, $f \mod l$ will also not have linear factors. □

From Claim 7.5, if $l_i \notin \text{span}(L_1, \ldots, L_r)$, then for projections modulo l_i, we have $\prod_{i=1}^{d_i} l_{ij} = \left(\prod_{j=1}^{d'} l_j\right) \mod l_i$ and $h_i(L_{i1}, \ldots, L_{ir}) = h(L_1, \ldots, L_r) \mod l_i$. There will be $(c_{cand} - 6\mathcal{R}(3) - 2\mathcal{R}(4))\log d - 6$ such independent linear forms l_i in $\mathcal{L}$. We can glue the $\prod_{i=1}^{d_i} l_{ij}$ projections to get $\left(\prod_{j=1}^{d'} l_j\right)$ as $(c_{cand} - 6\mathcal{R}(3) - 2\mathcal{R}(4))\log d - 6 > 100\log d$ from Theorem 3.15 in time $\text{poly}(d)$. From the $(c_{cand} - 6\mathcal{R}(3) - 2\mathcal{R}(4))\log d - 6$ set of linear forms, we can find two linear forms l_1, l_2 such that $l_1 \notin \text{span}(L_1, \ldots, L_r)$ and $l_2 \notin \text{span}(l_1, L_1, \ldots, L_r)$, and therefore $h \mod \langle l_1, l_2\rangle$ will have r essential variables. From Lemma 3.16, we can glue the h_i representations of $h \mod l_1$ and $h \mod l_2$, i.e. projections of the simple part mod l_1 and l_2 in time $(n \cdot d^{O(\log d)})$. Therefore, we obtain $T_2 + T_3$ as a $\Sigma\Pi\Sigma(1, d, 2\mathcal{R}(4)\log d + 6)$ circuit in time $O(n \cdot d^{O(\log d)})$. Finally, we subtract $T_2 + T_3$ from C and factorize it using Lemma 3.7 to get T_1. □

8 Future Work

The main open question would be to develop a reconstruction algorithm for depth 3 arithmetic circuits with arbitrary constant top fan-in. Several of the techniques developed in this paper, such as the algorithms for computing $\mathcal{S}_2, \mathcal{S}_3$ spaces (given that we know a way of bounding the number of them) and gluing projections from reconstructed gates, are generalizable to larger top fan-in. However, there are some significant challenges that remain. One main challenge is to prove a stronger structure theorem that bounds the number of $\mathcal{S}_i$ spaces (for all i that is at most the top fanin). Once we can bound the number of these spaces, we still do not know how to show that intersections of these spaces will recover the linear forms in the circuit. It is also a very interesting question to derandomize the current algorithm, as well as to generalize our algorithm to work in the setting of "generalized" depth-3 circuits. We believe the latter question to be not too difficult given the techniques developed in this paper.

References

[Ang88] D. Angluin. Queries and concept learning. *Machine Learning*, 2:319–342, 1988.

[AV08] M. Agrawal and V. Vinay. Arithmetic circuits: A chasm at depth four. In *Proceedings of the 49th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 67–75, 2008.

[BBB+00] A. Beimel, F. Bergadano, N. H. Bshouty, E. Kushilevitz, and S. Varricchio. Learning functions represented as multiplicity automata. *J. ACM*, 47(3):506–530, 2000.

[BDWY13] Boaz Barak, Zeev Dvir, Avi Wigderson, and Amir Yehudayoff. Fractional sylvester–gallai theorems. *Proceedings of the National Academy of Sciences*, 110(48):19213–19219, 2013.

[BE67] W Bonnice and MICHAEL Edelstein. Flats associated with finite sets in pd. *Niew. Arch. Wisk*, 15:11–14, 1967.

[BGKS22] Vishwas Bhargava, Ankit Garg, Neeraj Kayal, and Chandan Saha. Learning generalized depth three arithmetic circuits in the non-degenerate case. In *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2022)*. Schloss-Dagstuhl-Leibniz Zentrum für Informatik, 2022.

[BOT88] M. Ben-Or and P. Tiwari. A deterministic algorithm for sparse multivariate polynominal interpolation. In *Proceedings of the 20th Annual ACM Symposium on Theory of Computing (STOC)*, pages 301–309, 1988.

[BS24] Vishwas Bhargava and Devansh Shringi. Faster & deterministic FPT algorithm for worst-case tensor decomposition. *Electron. Colloquium Comput. Complex.*, TR24-123, 2024.

[BSV20] Vishwas Bhargava, Shubhangi Saraf, and Ilya Volkovich. Reconstruction of depth-4 multilinear circuits. *SODA 2020*, 2020.

[BSV21] Vishwas Bhargava, Shubhangi Saraf, and Ilya Volkovich. Reconstruction algorithms for low-rank tensors and depth-3 multilinear circuits. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 809–822, 2021.

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

[Car06] Enrico Carlini. Reducing the number of variables of a polynomial. In *Algebraic geometry and geometric modeling*, pages 237–247. Springer, 2006.

[DDS21] Pranjal Dutta, Prateek Dwivedi, and Nitin Saxena. Deterministic Identity Testing Paradigms for Bounded Top-Fanin Depth-4 Circuits. In Valentine Kabanets, editor, *36th Computational Complexity Conference (CCC 2021)*, volume 200 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 11:1–11:27, Dagstuhl, Germany, 2021. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

[DS05] Zeev Dvir and Amir Shpilka. Locally decodable codes with 2 queries and polynomial identity testing for depth 3 circuits. In *Proceedings of the Thirty-Seventh Annual ACM Symposium on Theory of Computing*, STOC '05, page 592–601, New York, NY, USA, 2005. Association for Computing Machinery.

[DSW14] Zeev Dvir, Shubhangi Saraf, and Avi Wigderson. Improved rank bounds for design matrices and a new proof of kelly's theorem. In *Forum of Mathematics, Sigma*, volume 2, page e4. Cambridge University Press, 2014.

[FS12] M. A. Forbes and A. Shpilka. Quasipolynomial-time identity testing of noncommutative and read-once oblivious algebraic branching programs. *Electronic Colloquium on Computational Complexity (ECCC)*, 19:115, 2012.

[GKKS13] A. Gupta, P. Kamath, N. Kayal, and R. Saptharishi. Arithmetic circuits: A chasm at depth three. In *Proceedings of the 54th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 578–587, 2013.

[GKL11] A. Gupta, N. Kayal, and S. V. Lokam. Efficient reconstruction of random multilinear formulas. In *IEEE 52nd Annual Symposium on Foundations of Computer Science, FOCS*, pages 778–787, 2011.

[GKL12] A. Gupta, N. Kayal, and S. V. Lokam. Reconstruction of depth-4 multilinear circuits with top fanin 2. In *Proceedings of the 44th Annual ACM Symposium on Theory of Computing (STOC)*, pages 625–642, 2012. Full version at https://eccc.weizmann.ac.il/report/2011/153.

[GKQ14] A. Gupta, N. Kayal, and Y. Qiao. Random arithmetic formulas can be reconstructed efficiently. *Computational Complexity*, 23(2):207–303, 2014.

[GKS20] Ankit Garg, Neeraj Kayal, and Chandan Saha. Learning sums of powers of low-degree polynomials in the non-degenerate case. In *2020 IEEE 61st Annual Symposium on Foundations of Computer Science (FOCS)*, pages 889–899. IEEE, 2020.

[GKST06] Oded Goldreich, Howard Karloff, Leonard J Schulman, and Luca Trevisan. Lower bounds for linear locally decodable codes and private information retrieval. *computational complexity*, 15:263–296, 2006.

[GOPS23] Abhibhav Garg, Rafael Oliveira, Shir Peleg, and Akash Kumar Sengupta. Radical sylvester-gallai theorem for tuples of quadratics. In *38th Computational Complexity Conference (CCC 2023)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2023.

[GOS22] Abhibhav Garg, Rafael Oliveira, and Akash Kumar Sengupta. Robust Radical Sylvester-Gallai Theorem for Quadratics. In Xavier Goaoc and Michael Kerber, editors, *38th International Symposium on Computational Geometry (SoCG 2022)*, volume 224 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 42:1–42:13, Dagstuhl, Germany, 2022. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

[GVJ88] D Yu Grigor'ev and Nicolai N Vorobjov Jr. Solving systems of polynomial inequalities in subexponential time. *Journal of symbolic computation*, 5(1-2):37–64, 1988.

[Han65] Sten Hansen. A generalization of a theorem of sylvester on the lines determined by a finite point set. *Mathematica Scandinavica*, 16(2):175–180, 1965.

[Ier89] D. Ierardi. Quantifier elimination in the theory of an algebraically-closed field. In *Proceedings of the Twenty-First Annual ACM Symposium on Theory of Computing*, STOC '89, page 138–147, New York, NY, USA, 1989. Association for Computing Machinery.

[Kay11] Neeraj Kayal. Efficient algorithms for some special cases of the polynomial equivalence problem. In *Proceedings of the twenty-second annual ACM-SIAM symposium on Discrete algorithms*, pages 1409–1421. SIAM, 2011.

[KNS19] Neeraj Kayal, Vineet Nair, and Chandan Saha. Average-case linear matrix factorization and reconstruction of low width algebraic branching programs. *computational complexity*, 28:749–828, 2019.

[KNST17] N. Kayal, V. Nair, C. Saha, and S. Tavenas. Reconstruction of full rank algebraic branching programs. In *32nd Computational Complexity Conference, CCC 2017.*, pages 21:1–21:61, 2017.

[Koi10] P. Koiran. Arithmetic circuits: the chasm at depth four gets wider. *CoRR*, abs/1006.4700, 2010.

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

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

[KS08] Zohar S Karnin and Amir Shpilka. Black box polynomial identity testing of generalized depth-3 arithmetic circuits with bounded top fan-in. In *2008 23rd Annual IEEE Conference on Computational Complexity*, pages 280–291. IEEE, 2008.

[KS09a] Zohar S Karnin and Amir Shpilka. Reconstruction of generalized depth-3 arithmetic circuits with bounded top fan-in. In *2009 24th Annual IEEE Conference on Computational Complexity*, pages 274–285. IEEE, 2009.

[KS09b] N. Kayal and S. Saraf. Blackbox polynomial identity testing for depth 3 circuits. In *Proceedings of the 50th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 198–207, 2009. Full version at https://eccc.weizmann.ac.il/report/2009/032.

[KS19] Neeraj Kayal and Chandan Saha. Reconstruction of non-degenerate homogeneous depth three circuits. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, pages 413–424, 2019.

[KSS14] Swastik Kopparty, Shubhangi Saraf, and Amir Shpilka. Equivalence of polynomial identity testing and deterministic multivariate polynomial factorization. In *2014 IEEE 29th Conference on Computational Complexity (CCC)*, pages 169–180. IEEE, 2014.

[KT90] Erich Kaltofen and Barry M Trager. Computing with polynomials given by black boxes for their evaluations: Greatest common divisors, factorization, separation of numerators and denominators. *Journal of Symbolic Computation*, 9(3):301–320, 1990.

[LST22] Nutan Limaye, Srikanth Srinivasan, and Sébastien Tavenas. Superpolynomial lower bounds against low-depth algebraic circuits. In *2021 IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 804–814, 2022.

[OS22] Rafael Oliveira and Akash Kumar Sengupta. Radical sylvester-gallai theorem for cubics. In *2022 IEEE 63rd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 212–220. IEEE, 2022.

[OS24] Rafael Oliveira and Akash Kumar Sengupta. Strong algebras and radical sylvester-gallai configurations. In *Proceedings of the 56th Annual ACM Symposium on Theory of Computing*, pages 95–105, 2024.

[PS21] Shir Peleg and Amir Shpilka. Polynomial time deterministic identity testing algorithm for $\Sigma^{[3]}\Pi\Sigma\Pi^{[2]}$ circuits via Edelstein–Kelly type theorem for quadratic polynomials. In *Proceedings of the 53rd Annual ACM SIGACT Symposium on Theory of Computing*, pages 259–271, 2021.

[PS22a] Shir Peleg and Amir Shpilka. A generalized sylvester–gallai-type theorem for quadratic polynomials. In *Forum of Mathematics, Sigma*, volume 10, page e112. Cambridge University Press, 2022.

[PS22b] Shir Peleg and Amir Shpilka. Robust Sylvester-Gallai Type Theorem for Quadratic Polynomials. In Xavier Goaoc and Michael Kerber, editors, *38th International Symposium on Computational Geometry (SoCG 2022)*, volume 224 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 43:1–43:15, Dagstuhl, Germany, 2022. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

[PSV24] Shir Peleg, Amir Shpilka, and Ben Lee Volk. Tensor Reconstruction Beyond Constant Rank. In Venkatesan Guruswami, editor, *15th Innovations in Theoretical Computer Science Conference (ITCS 2024)*, volume 287 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 87:1–87:20, Dagstuhl, Germany, 2024. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

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

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

[Shp19] Amir Shpilka. Sylvester-gallai type theorems for quadratic polynomials. In *Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing*, pages 1203–1214, 2019.

[Sin16a] Gaurav Sinha. *Blackbox Reconstruction of Depth Three Circuits with Top Fan-In Two*. PhD thesis, California Institute of Technology, 2016.

[Sin16b] Gaurav Sinha. Reconstruction of Real Depth-3 Circuits with Top Fan-In 2. In Ran Raz, editor, *31st Conference on Computational Complexity (CCC 2016)*, volume 50 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 31:1–31:53, Dagstuhl, Germany, 2016. Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

[Sin22] Gaurav Sinha. Efficient reconstruction of depth three arithmetic circuits with top fan-in two. In *13th Innovations in Theoretical Computer Science Conference (ITCS 2022)*. Schloss Dagstuhl-Leibniz-Zentrum für Informatik, 2022.

[SS11] Nitin Saxena and Comandur Seshadhri. Blackbox identity testing for bounded top fanin depth-3 circuits: the field doesn't matter. In *Proceedings of the forty-third annual ACM symposium on Theory of computing*, pages 431–440, 2011.

[SS13] Nitin Saxena and Comandur Seshadhri. From sylvester-gallai configurations to rank bounds: Improved blackbox identity test for depth-3 circuits. *Journal of the ACM (JACM)*, 60(5):1–33, 2013.

[Tav13] S. Tavenas. Improved bounds for reduction to depth 4 and depth 3. In *MFCS*, pages 813–824, 2013.

[Zip79] Richard Zippel. Probabilistic algorithms for sparse polynomials. In *International symposium on symbolic and algebraic manipulation*, pages 216–226. Springer, 1979.

ECCC ISSN 1433-8092

https://eccc.weizmann.ac.il