Univariate Ideal Membership Parameterized by Rank, Degree, and Number of Generators

V. Arvind
Institute of Mathematical Sciences (HBNI), Chennai, India
arvind@imsc.res.in

Abhranil Chatterjee
Institute of Mathematical Sciences (HBNI), Chennai, India
abhranilc@imsc.res.in

Rajit Datta
Chennai Mathematical Institute, Chennai, India
rajit@cmi.ac.in

Partha Mukhopadhyay
Chennai Mathematical Institute, Chennai, India
partham@cmi.ac.in

Abstract

Let $\mathbb{F}[\mathrm{X}]$ be the polynomial ring over the variables $\mathrm{X} = \{x_1, x_2, \ldots, x_n\}$. An ideal $I = \langle p_1(x_1), \ldots, p_n(x_n)\rangle$ generated by univariate polynomials $\{p_i(x_i)\}_{i=1}^n$ is a *univariate ideal*. We study the ideal membership problem for the univariate ideals and show the following results.

- Let $f(\mathrm{X}) \in \mathbb{F}[\ell_1, \ldots, \ell_r]$ be a (low rank) polynomial given by an arithmetic circuit where $\ell_i : 1 \leq i \leq r$ are linear forms, and $I = \langle p_1(x_1), \ldots, p_n(x_n)\rangle$ be a univariate ideal. Given $\vec{\alpha} \in \mathbb{F}^n$, the (unique) remainder $f(\mathrm{X}) \pmod I$ can be evaluated at $\vec{\alpha}$ in deterministic time $d^{O(r)} \cdot \mathrm{poly}(n)$, where $d = \max\{\deg(f), \deg(p_1) \ldots, \deg(p_n)\}$. This yields a randomized $n^{O(r)}$ algorithm for minimum vertex cover in graphs with rank-r adjacency matrices. It also yields an $n^{O(r)}$ algorithm for evaluating the permanent of a $n \times n$ matrix of rank r, over any field $\mathbb{F}$. Over $\mathbb{Q}$, an algorithm of similar run time for low rank permanent is due to Barvinok [5] via a different technique.
- Let $f(\mathrm{X}) \in \mathbb{F}[\mathrm{X}]$ be given by an arithmetic circuit of degree k (k treated as fixed parameter) and $I = \langle p_1(x_1), \ldots, p_n(x_n)\rangle$. We show that in the special case when $I = \langle x_1^{e_1}, \ldots, x_n^{e_n}\rangle$, we obtain a randomized $O^*(4.08^k)$ algorithm that uses $\mathrm{poly}(n, k)$ space.
- Given $f(\mathrm{X}) \in \mathbb{F}[\mathrm{X}]$ by an arithmetic circuit and $I = \langle p_1(x_1), \ldots, p_k(x_k)\rangle$, membership testing is W[1]-hard, parameterized by k. The problem is MINI[1]-hard in the special case when $I = \langle x_1^{e_1}, \ldots, x_k^{e_k}\rangle$.

2012 ACM Subject Classification Theory of computation → Design and analysis of algorithms

Keywords and phrases Combinatorial Nullstellensatz, Ideal Membership, Parametric Hardness, Low Rank Permanent

Digital Object Identifier 10.4230/LIPIcs.FSTTCS.2018.7

Acknowledgements We thank the anonymous reviewers for their useful comments. The third author acknowledges partial support from Infosys Foundation. The fourth author acknowledges partial support from Infosys Foundation and Tata Trust.

38th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science (FSTTCS 2018).
Editors: Sumit Ganguly and Paritosh Pandya; Article No. 7; pp. 7:1–7:18
Leibniz International Proceedings in Informatics
LIPICS Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

1 Introduction

Let $R = \mathbb{F}[x_1, x_2, \ldots, x_n]$[1] be the ring of polynomials over the variables $\mathrm{X} = \{x_1, x_2, \ldots, x_n\}$. A subring $I \subseteq R$ is an ideal if $IR \subseteq I$. Computationally, an ideal I is often given by generators: $I = \langle f_1, f_2, \ldots, f_\ell \rangle$. Given $f \in R$ and $I = \langle f_1, \ldots, f_\ell \rangle$, the *Ideal Membership problem* is to decide whether $f \in I$ or not. In general, this is computationally highly intractable. In fact, it is EXPSPACE-complete even if f and the generators $f_i, i \in [\ell]$ are given explicitly by sum of monomials [21]. Nevertheless, special cases of ideal membership problem have played important roles in several results in arithmetic complexity. For example, the polynomial identity testing algorithm for depth three $\Sigma\Pi\Sigma$ circuits with bounded top fan-in; the structure theorem for $\Sigma\Pi\Sigma(k, d)$ identities use ideal membership very crucially [4, 13, 24].

In this paper, our study of ideal membership is motivated by a basic result in algebraic complexity: the Combinatorial Nullstellensatz of Alon [1], and we recall a basic result in that paper.

▶ **Theorem 1.** *Let $\mathbb{F}$ be any field, and $f(\mathrm{X}) \in \mathbb{F}[\mathrm{X}]$. Define polynomials $g_i(x_i) = \prod_{s \in S_i}(x_i - s)$ for non-empty subsets $S_i, 1 \leq i \leq n$ of $\mathbb{F}$. If f vanishes on all the common zeros of $g_1, \ldots, g_n$, then there are polynomials $h_1, \ldots, h_n$ satisfying* $\deg(h_i) \leq \deg(f) - \deg(g_i)$ *such that $f = \sum_{i=1}^{n} h_i g_i$.*

The theorem can be restated in terms of ideal membership: Let $f(\mathrm{X}) \in \mathbb{F}[\mathrm{X}]$ be a given polynomial, and $I = \langle g_1(x_1), g_2(x_2), \ldots, g_n(x_n) \rangle$ be an ideal generated by univariate polynomials g_i *without repeated roots*. Let $Z(g_i)$ denote the zero set of $g_i, 1 \leq i \leq n$. By Theorem 1, if $f \notin I$ then there is a $\vec{\alpha} = (\alpha_1, \ldots, \alpha_n) \in Z(g_1) \times \cdots \times Z(g_n)$ such that $f(\vec{\alpha}) \neq 0$. Of course, if $f \in I$ then $f|_{Z(g_1) \times \cdots \times Z(g_n)} = 0$.

Ideals I generated by univariate polynomials are called *univariate ideals*. For any univariate ideal I and any polynomial f, by repeated application of the division algorithm, we can write $f(\mathrm{X}) = \sum_{i=1}^{n} h_i(\mathrm{X}) g_i(x_i) + R(\mathrm{X})$ where R is unique and for each $i \in [n]$: $\deg_{x_i}(R) < \deg(g_i(x_i))$. Since the remainder is unique, it is convenient to write $R = f \bmod I$. By Alon's theorem, if $f \notin I$ then there is a $\vec{\alpha} \in Z(g_1) \times \cdots \times Z(g_n)$ such that $R(\vec{\alpha}) \neq 0$.

As an application of the theorem, Alon and Tarsi showed that checking k-colorability of a graph G is polynomial-time equivalent to testing whether the graph polynomial f_G is in the ideal $\langle x_1^k - 1, \ldots, x_n^k - 1 \rangle$ [1]. It follows that univariate ideal membership problem coNP-hard.

Univariate ideal membership is further motivated by its connection with two well-studied problems. Computing the permanent of a $n \times n$ matrix over any field $\mathbb{F}$ can be cast in terms of univariate ideal membership. Given a matrix $A = (a_{i,j})_{1 \leq i,j \leq n} \in \mathbb{F}^{n \times n}$, consider the product of linear forms $P_A(\mathrm{X}) = \prod_{i=1}^{n}(\sum_{j=1}^{n} a_{ij} x_j)$. The following observation is well known.

▶ **Fact 2.** *The permanent of the matrix A is given by the coefficient of the monomial $x_1 x_2 \cdots x_n$ in P_A.*

It follows immediately that $P_A(\mathrm{X}) \pmod{\langle x_1^2, \ldots, x_n^2 \rangle} = \mathrm{Perm}(A)\ x_1 x_2 \cdots x_n$. I.e., the remainder $P_A \pmod{\langle x_1^2, \ldots, x_n^2 \rangle}$ evaluates to $\mathrm{Perm}(A)$ at the point $\vec{1} \in \mathbb{F}^n$.

[1] We often use the shorthand notation $\mathbb{F}[\mathrm{X}]$.

Next, we briefly mention the connection of univariate ideal membership with the multilinear monomial detection problem, a benchmark problem that is useful in designing fast parameterized algorithms for a host of problems [16, 17, 18, 28].

Notice that, given an arithmetic circuit C computing a polynomial $f \in \mathbb{F}[\mathrm{X}]$ of degree k, checking if f has a non-zero multilinear monomial of degree k is equivalent to checking if $f \pmod{\langle x_1^2, \ldots, x_n^2\rangle}$ is non-zero. Moreover, the constrained multilinear detection problem studied in [6, 17] can also be viewed as a problem of deciding membership in a univariate ideal.

Our Results. A contribution of this paper is to consider several parameterized problems in arithmetic complexity as instances of univariate ideal membership. One parameter of interest is the rank of a multivariate polynomial: We say $f \in \mathbb{F}[\mathrm{X}]$ is a *rank* r polynomial if $f \in \mathbb{F}[\ell_1, \ell_2, \ldots, \ell_r]$ for linear forms $\ell_j : 1 \leq j \leq r$. This concept has found application in algorithms for depth-3 polynomial identity testing [24]. Given a univariate ideal I, a point $\vec{\alpha} \in \mathbb{F}^n$, and an arithmetic circuit computing a polynomial f of rank r, we obtain an efficient algorithm to compute $f \pmod{I}$ at $\vec{\alpha}$.

▶ **Theorem 3.** *Let $\mathbb{F}$ be an arbitrary field where the field arithmetic can be done efficiently, and C be a polynomial-size arithmetic circuit computing a polynomial f in $\mathbb{F}[\ell_1, \ell_2, \ldots, \ell_r]$, where $\ell_1, \ell_2, \ldots, \ell_r$ are given linear forms in $\{x_1, x_2, \ldots, x_n\}$. Let $I = \langle p_1, \ldots, p_n\rangle$ be a univariate ideal generated by $p_i(x_i) \in \mathbb{F}[x_i], 1 \leq i \leq n$. Given $\vec{\alpha} \in \mathbb{F}^n$, we can evaluate the remainder $f \pmod{I}$ at the point $\vec{\alpha}$ in time $d^{O(r)}\mathrm{poly}(n)$, where $d = \max\{\deg(f), \deg(p_i) : 1 \leq i \leq n\}$.*

This also allows us to check whether $f \in I$ by picking a point $\vec{\alpha}$ at random and checking whether $f \pmod{I}$ evaluated at $\vec{\alpha}$ is zero or not. The intuitive idea behind the proof of Theorem 3 is as follows. Given a polynomial $f(\mathrm{X}) \in \mathbb{F}[\ell_1, \ldots, \ell_r]$, a univariate ideal $I = \langle p_1(x_1), \ldots, p_n(x_n)\rangle$, and a point $\vec{\alpha} \in \mathbb{F}^n$, we first find an invertible linear transformation T such that the polynomial $T(f)$ becomes a polynomial over at most $2r$ variables. Additionally T has the property that T fixes the variables $x_1, \ldots, x_r$. Then we recover the polynomial (call it $\tilde{f}$) over at most $2r$ variables explicitly and perform division algorithm with respect to the ideal $I_{[r]} = \langle p_1(x_1), \ldots, p_r(x_r)\rangle$. For notational convenience, call $\tilde{f}$ be the polynomial obtained over at most $2r$ variables. It turns out $T^{-1}(\tilde{f})$ is the *true remainder* $f \pmod{I_{[r]}}$. Since the variables $x_1, \ldots, x_r$ do not play role in the subsequent stages of division, we can eliminate them by substituting $x_i \leftarrow \alpha_i$ for each $1 \leq i \leq r$. Then we apply the division algorithm on $T^{-1}(\tilde{f})|_{x_i \leftarrow \alpha_i : 1 \leq i \leq r}$ recursively with respect to the ideal $I_{[n]\setminus[r]}$ to compute the final remainder at the point $\vec{\alpha}$.

Our next result is an efficient algorithm to detect vertex cover in low rank graphs. A graph G is said to be of rank r if the rank of the adjacency matrix A_G is of rank r. Graphs of low rank were studied by Lovasz and Kotlov [2, 15] in the context of graph coloring. Our idea is to construct a low rank polynomial from the graph and check its membership in an appropriate univariate ideal.

▶ **Theorem 4.** *Given a graph $G = (V, E)$ on n vertices such that the rank of the adjacency matrix A_G is at most r, and a parameter k, there is a randomized $n^{O(r)}$ algorithm to decide if the graph G has vertex cover of size k or not.*

Theorem 3 also yields an $n^{O(r)}$ algorithm to compute the permanent of rank-r matrices over any field. Barvinok had given [5] an algorithm of same running time for the permanent of low rank matrices (over $\mathbb{Q}$) using apolar bilinear forms. By Fact 2, if matrix A is rank r then P_A is a rank-r polynomial, and for the univariate ideal $I = \langle x_1^2, \ldots, x_n^2\rangle$ computing

$P_A \pmod I$ at the point $\vec{1}$ yields the permanent. Theorem 3 works more generally for all univariate ideals. In particular, the ideal in the proof of Theorem 4 is generated by polynomials that are not powers of variables. Thus, Theorem 3 can potentially have more algorithmic consequences than the technique in [5].

If k is the degree of the input polynomial and the ideal is given by the powers of variables as generators, we have a randomized FPT algorithm for the problem.

▶ **Theorem 5.** *Given an arithmetic circuit C computing a polynomial $f(\mathrm{X}) \in \mathbb{Z}[\mathrm{X}]$ of degree k and integers $e_1, e_2, \ldots, e_n$, there is a randomized algorithm to decide whether $f \in \langle x_1^{e_1}, x_2^{e_2}, \ldots, x_n^{e_n}\rangle$ in $O^*(4.08^k)$ time.*

Note that this generalizes the well-known problem of *multilinear monomial detection* for which the ideal of interest would be $I = \langle x_1^2, x_2^2, \ldots, x_n^2\rangle$. Surprisingly, the run time of the algorithm in Theorem 5 is independent of the e_i. Brand et al. have given the first FPT algorithm for multilinear monomial detection in the case of general circuit with run time randomized $O^*(4.32^k)$ [7]. Recently, this problem has also been studied using the Hadamard product [3] of the given polynomial with the elementary symmetric polynomial (and differently using apolar bilinear forms [22]). When the number of generators in the ideal is treated as the fixed parameter, the problem is W[1]-hard.

▶ **Theorem 6.** *Given a polynomial $f(\mathrm{X}) \in \mathbb{F}[\mathrm{X}]$ by an arithmetic circuit C and univariate polynomials $p_1(x_1), p_2(x_2), \ldots, p_k(x_k)$, checking if $f \notin \langle p_1(x_1), p_2(x_2), \ldots, p_k(x_k)\rangle$ is* W[1]*-hard with k as the parameter.*

Theorem 6 is shown by a suitable reduction from independent set problem to ideal membership. To find an independent set of size k, the reduction produces an ideal with k univariates and the polynomial created from the graph has k variables. Unlike Theorem 5, the above parameterization of the problem remains MINI[1]-hard even if the ideal is generated by powers of variables. More precisely, we show the following result.

▶ **Theorem 7.** *Let C be a polynomial-size arithmetic circuit computing a polynomial $f \in \mathbb{F}[\mathrm{X}]$. Let $I = \langle {x_1}^{e_1}, {x_2}^{e_2}, \ldots, {x_k}^{e_k}\rangle$ be the given ideal where $e_1, \ldots, e_k$ are given in unary, checking if $f \notin I$ is* MINI[1]*-hard with k as parameter.*

It turns out that the complement of the ideal membership problem can be easily reduced from k-Lin-Eq problem which asks if there is a $\vec{x} \in \{0,1\}^n$ satisfying $A\vec{x} = \vec{b}$, where $A \in \mathbb{F}^{k\times n}$ and $\vec{b} \in \mathbb{F}^k$.

We can show k-Lin-Eq is hard for the parameterized complexity class MINI[1] by reducing the miniature version of 1-in-3 POSITIVE 3-SAT to it.

As already mentioned, the result of Alon and Tarsi [1] shows that the membership of f_G in $\langle x_1^k - 1, \ldots, x_n^k - 1\rangle$ is coNP-hard and the proof crucially uses the fact that the roots of the generator polynomials are all distinct. This naturally raises the question if univariate ideal membership is in coNP when each generator polynomial has distinct roots. We show membership in coNP.

▶ **Theorem 8.** *Let $f \in \mathbb{Q}[\mathrm{X}]$ be a polynomial of degree at most d given by a black-box. Let $I = \langle p_1(x_1), \ldots, p_n(x_n)\rangle$ be an ideal given explicitly by a set of univariate polynomials $p_1, p_2, \ldots, p_n$ as generators of maximum degree bounded by d. Let L be the bit-size upper bound for any coefficient in $f, p_1, p_2, \ldots, p_n$. Moreover, assume that p_is have distinct roots over $\mathbb{C}$. Then there is a non-deterministic algorithm running in time* $\mathrm{poly}(n, d, L)$ *that decides the non-membership of f in the ideal I.*

▶ Remark. The distinct roots case discussed in Theorem 8 is in stark contrast to the complexity of testing membership of $P_A(\mathrm{X})$ in the ideal $\langle x_1^2, \ldots, x_n^2\rangle$. That problem is equivalent to checking if $\mathrm{Perm}(A)$ is nonzero for a rational matrix A, which is hard for the exact counting class $\mathrm{C_{=}P}$. Hence it cannot be in coNP unless the polynomial-time hierarchy collapses.

Recall from Alon's Nullstellensatz that if $f \notin I$, then there is always a point $\vec{\alpha} \in Z(p_1) \times \ldots \times Z(p_n)$ such that $f(\vec{\alpha}) \neq 0$. Notice that in general the roots $\alpha_i \in \mathbb{C}$ and in the standard *Turing Machine* model the NP machine can not guess the roots directly with only finite precision. But we are able to prove that the NP machine can guess the tuple of roots $\vec{\tilde{\alpha}} \in \mathbb{Q}^n$ using only polynomial bits of precision and still can decide the non-membership. The main technical idea is to compute efficiently a parameter M only from the input parameters such that $|f(\vec{\tilde{\alpha}})| \leq M$ if $f \in I$, and $|f(\vec{\tilde{\alpha}})| \geq 2M$ if $f \notin I$. The NP machine decides the non-membership according to the final value of $|f(\vec{\tilde{\alpha}})|$. We remark that Koiran has considered the weak version of Hilbert Nullstellensatz (HN) problem [14]. The input is a set of multivariate polynomials $f_1, f_2, \ldots, f_m \in \mathbb{Z}[\mathrm{X}]$ and the problem is to decide whether $1 \in \langle f_1, \ldots, f_m\rangle$. The result of Koiran shows that $\overline{\mathrm{HN}} \in \mathrm{AM}$ (under GRH), and it is an outstanding open problem problem to decide whether $\overline{\mathrm{HN}} \in \mathrm{NP}$.

Organization. In Section 2 we give some background results. We prove Theorem 3 and Theorem 4 in Section 3. In Section 4, we explore the parameterized complexity of univariate ideal membership. In the first subsection, we prove 5, and in the second subsection we prove Theorems 6 and 7. Finally, in Section 5, we prove Theorem 8.

2 Preliminaries

Basics of Ideal Membership. Let $\mathbb{F}[\mathrm{X}]$ be the ring of polynomials $\mathbb{F}[x_1, x_2, \ldots, x_n]$. Let $I \subseteq \mathbb{F}[\mathrm{X}]$ be an ideal given by a set of generators $I = \langle g_1, \ldots, g_\ell\rangle$. Then for any polynomial $f \in \mathbb{F}[\mathrm{X}]$, it is a member of the ideal if and only if $f = \sum_{i=1}^{\ell} h_i g_i$ where $\forall i : h_i \in \mathbb{F}[\mathrm{X}]$. Dividing f by the g_i by applying the standard division algorithm does not work in general to check if $f \in I$. Indeed, the remainder is not even uniquely defined. However, if the leading monomials of the generators are already pairwise relatively prime, then we can apply the division algorithm to compute the unique remainder.

▶ **Theorem 9** (See[9], Theorem 3, proposition 4, pp.101). *Let I be a polynomial ideal given by a basis $G = \{g_1, g_2, \cdots, g_s\}$ such that all pairs $i \neq j$ $LM(g_i)$ and $LM(g_j)$ are relatively prime. Then G is a Gröbner basis for I.*

In particular, if the ideal I is a univariate ideal given by $I = \langle p_1(x_1), \ldots, p_n(x_n)\rangle$, we can apply the division algorithm to compute the unique remainder $f \pmod{I}$. To bound the run time of this procedure we note the following: Let $\bar{p}$ denote the ordered list $\{p_1, p_2, \ldots, p_n\}$. Let $\mathrm{Divide}(f; \bar{p})$ be the procedure that divides f by p_1 to obtain remainder f_1, then divides f_1 by p_2 to obtain remainder f_2, and so on to obtain the final remainder f_n after dividing by p_n. We note the following time bound for $\mathrm{Divide}(f; \bar{p})$.

▶ **Fact 10** (See [27], Section 6, pp.5-12). *Let $f \in \mathbb{F}[\mathrm{X}]$ be given by a size s arithmetic circuit and $p_i(x_i) \in \mathbb{F}[x_i]$ be given univariate polynomials. The running time of $\mathrm{Divide}(f; \bar{p})$ is bounded by $O(s \cdot \prod_{i=1}^{n}(d_i+1)^{O(1)})$, where $d_i = \max\{\deg_{x_i}(f), \deg(p_i(x_i))\}$.*

On Roots of Univariate Polynomials. The following lemma shows that the absolute value of any root of a univariate polynomial can be bounded in terms of the degree and the coefficients. The result is folklore.

▶ **Lemma 11.** *Let $f(x) = \sum_{i=0}^{d} a_i x^i \in \mathbb{Q}[x]$ be a univariate polynomial and α be a root of f. Then, either $\frac{|a_0|}{\sum_{i=1}^{d} |a_i|} \leq |\alpha| < 1$ or $1 \leq |\alpha| \leq d \cdot \frac{\max_i |a_i|}{|a_d|}$.*

Proof. Since α is a root of f, we have that, $0 = f(\alpha) = \sum_{i=0}^{d} a_i \alpha^i = 0$, and $\sum_{i=1}^{d} a_i \alpha^i = -a_0$. Then by an application of triangle inequality, we get that $\sum_{i=1}^{d} |a_i||\alpha|^i \geq |a_0|$. Now we analyse two different cases. In the first case assume that $|\alpha| < 1$. Observe that $|\alpha| \cdot (\sum_{i=1}^{d} |a_i|) \geq |a_0|$, and hence $|\alpha| \geq \frac{|a_0|}{\sum_{i=1}^{d} |a_i|}$. In the second case $|\alpha| \geq 1$. Observe that $-a_d \alpha^d = \sum_{i=0}^{d-1} a_i \alpha^i$. Then use triangle inequality to get that $|a_d||\alpha|^d \leq |\alpha|^{d-1} \cdot (\sum_{i=0}^{d-1} |a_i|)$. Now we get the following, $|\alpha| \leq \frac{\sum_{i=0}^{d-1} |a_i|}{|a_d|} \leq d \cdot \frac{\max_i |a_i|}{|a_d|}$. The lemma follows by combining the two cases. ◀

The next lemma shows that the separation between two distinct roots of any univariate polynomial can be lower bounded in terms of degree and the size of the coefficients. This was shown by Mahler [20].

▶ **Lemma 12.** *Let $g(x) = \sum_{i=0}^{d} a_i x^i \in \mathbb{Q}[x]$ and $2^{-L} \leq |a_i| \leq 2^L$ (if $a_i \neq 0$). Let α, β are two distinct roots of g. Then $|\alpha - \beta| \geq \frac{1}{2^{O(dL)}}$.*

The following lemma states that any univariate polynomial can not get a very small value (in absolute sense) on any point which is far from every root.

▶ **Lemma 13.** *Let $f = \sum_{i=1}^{d} a_i x^i$ be a univariate polynomial with $2^{-L} \leq |a_i| \leq 2^L$ (if $a_i \neq 0$). Let $\tilde{\alpha}$ be a point such that $|\tilde{\alpha} - \beta_i| \geq \delta$ for every root β_i of f then $|f(\tilde{\alpha})| \geq 2^{-L}\delta^d$.*

Proof. We observe that, $f(\tilde{\alpha}) = c\prod_{i=1}^{d}(\tilde{\alpha} - \beta_i)$. Since $|\tilde{\alpha} - \beta_i| \geq \delta$ we get, $|f(\tilde{\alpha})| = |c| \prod_{i=1}^{d} |\tilde{\alpha} - \beta_i| \geq 2^{-L}\delta^d$. This completes the proof. ◀

Parameterized Complexity Classes. We recall some standard definitions in parameterized Complexity [10, ch.1,pp. 7-14]. We only state them informally. For a parameterized input problem (x, k) with k be the parameter of interest, we say that the problem is in FPT if it has an algorithm with run time $f(k)|(x, k)|^{O(1)}$ for some computable function f. A parameterized reduction [10, def. 13.1] between two problems should be computable in time $f(k)|(x, k)|^{O(1)}$, and if the reduction outputs (x', k') then $k' \leq f(k)$.

The complexity class MINI[1] consists of parameterized problems that are miniature versions of NP problems: For $L \in$ NP, its miniature version mini(L) has instances of the form $(0^n, x)$, where $|x| \leq k \log n$, k is the fixed parameter, and x is an instance of L. Showing mini(L) to be MINI[1]-hard under parameterized reductions is evidence of its parameterized intractability, for it cannot be in FPT assuming the Exponential Time Hypothesis [12].

Hadamard Product. We recall the definition of Hadamard product of two polynomials.

▶ **Definition 14.** Given two polynomials $f, g \in \mathbb{F}[\mathrm{X}]$, the Hadamard product $f \circ g$ is defined as $f \circ g = \sum_m [m]f \cdot [m]g \cdot m$.

In this paper we adapt the notion of Hadamard product suitably and define a scaled version of Hadamard Product of two polynomials.

▶ **Definition 15.** Given two polynomials $f, g \in \mathbb{F}[X]$, their *scaled Hadamard Product* $f \circ^s g$, is defined as $f \circ^s g = \sum_m m! \cdot [m]f \cdot [m]g \cdot m$, where $m = x_{i_1}^{e_1} x_{i_2}^{e_2} \dots x_{i_r}^{e_r}$ and $m! = e_1! \cdot e_2! \cdots e_r!$ abusing the notation.

▶ Remark. Given two polynomials $f \in \mathbb{F}[X]$ and $g \in \mathbb{F}[X]$, if one of these two is a multilinear polynomial then scaled Hadamard product $f \circ^s g$ is same as Hadamard product $f \circ g$.

Symmetric Polynomial and Weakly Equivalence of Polynomials. The symmetric polynomial of degree k over n variables $\{x_1, x_2, \dots, x_n\}$, denoted by $S_{n,k}$, is defined as follows:

$$S_{n,k}(x_1, x_2, \dots, x_n) = \sum_{T \subseteq [n], |T|=k} \prod_{i \in T} x_i.$$

Notice that, $S_{n,k}$ contains all the degree k multillinear terms. A recent result of Lee gives the following homogeneous diagonal circuit for $S_{n,k}$ [19].

▶ **Lemma 16.** *The symmetric polynomial $S_{n,k}$ can be computed by a homogenous $\Sigma^{[s]} \wedge^{[k]} \Sigma$ circuit where $s \leq \sum_{i=0}^{k/2} \binom{n}{i}$.*

A polynomial $f \in \mathbb{F}[X]$ is said to be *weakly equivalent* to a polynomial $g \in \mathbb{F}[X]$, if the following is true. For each monomial m, $[m]f = 0$ if and only if $[m]g = 0$. In this paper, we will use polynomials weakly equivalent to $S_{n,k}$.

3 Ideal Membership for Low Rank Polynomials

In this section we prove Theorem 3. Given a r-rank polynomial f by an arithmetic circuit, a univariate ideal I, and a point $\vec{\alpha} \in \mathbb{F}^n$, we give an $d^{O(r)}$ time algorithm to evaluate the remainder polynomial $f \pmod I$ at $\vec{\alpha}$ where d is the degree of the polynomial f. As mentioned in Section 1, an application of our result yields an $n^{O(r)}$ time algorithm for computing the permanent of rank-r matrices over any field. Barvinok [5], via a different method, had obtained an $n^{O(r)}$ time algorithm for this problem over $\mathbb{Q}$. We also obtain a randomized $n^{O(r)}$ time algorithm for minimum vertex cover of low rank graphs. We first define the notion *rank* of a polynomial in $\mathbb{F}$[X].

▶ **Definition 17.** A polynomial $f(\mathrm{X}) \in \mathbb{F}[\mathrm{X}]$ is a *rank-r* polynomial if there are linear forms $\ell_1, \ell_2, \dots, \ell_r$ such that $f(\mathrm{X})$ is in the sub-algebra $\mathbb{F}[\ell_1, \dots, \ell_r]$.

For an unspecified fixed parameter r, we refer to rank-r polynomials as *low rank polynomials.*

Given $\vec{\alpha} \in \mathbb{F}^n$, a univariate ideal $I = \langle p_1(x_1), \dots, p_n(x_n) \rangle$, and a rank r polynomial $f(\ell_1, \dots, \ell_r)$ we show how to compute $f(\ell_1, \dots, \ell_r) \pmod I$ at $\vec{\alpha}$ using a recursive procedure $\mathrm{REM}(f(\ell_1, \dots, \ell_r), I, \vec{\alpha})$ efficiently. We introduce the following notation. For $S \subseteq [n]$, the ideal $I_S = \langle p_i(x_i) : i \in [S] \rangle$.

We first observe the following lemma which shows how to remove the redundant variables from a low rank polynomial.

▶ **Lemma 18.** *Given a polynomial $f(\ell_1, \dots, \ell_r)$ where $\ell_1, \dots, \ell_r$ are linear forms in $\mathbb{F}$[X], there is an invertible linear transform $T : \mathbb{F}^n \mapsto \mathbb{F}^n$ that fixes $x_1, \dots, x_r$ and the transformed polynomial $T(f)$ is over at most $2r$ variables.*

Proof. Write each linear form ℓ_i in two parts: $\ell_i = \ell_{i,1} + \ell_{i,2}$, where $\ell_{i,1}$ is the part over variables $x_1, \dots, x_r$ and $\ell_{i,2}$ is over variables $x_{r+1}, \dots, x_n$. W.l.o.g, assume that $\{\ell_{i,2}\}_{i=1}^{r'}$ is a maximum linearly independent subset of linear forms in $\{\ell_{i,2}\}_{i=1}^{r}$. Let $T : \mathbb{F}^n \to \mathbb{F}^n$ be the invertible linear map that fixes $x_1, \ldots, x_r$, maps the independent linear forms $\{\ell_{i,2}\}_{i=1}^{r'}$ to variables $x_{r+1}, \ldots, x_{r+r'}$, and suitably extends T to an invertible map. This completes the proof. ◀

The following lemma shows that the univariate division and evaluating the remainder at the end can be achieved by division and evaluation partially.

▶ **Lemma 19.** *Let $f(\mathrm{X}) \in \mathbb{F}[\mathrm{X}]$ and $I = \langle p_1(x_1), \ldots, p_n(x_n)\rangle$ be a univariate ideal. Let $R(\mathrm{X})$ be the unique remainder $f \pmod{I}$. Let $\vec{\alpha} \in \mathbb{F}^r, r \leq n$ and $R_r(\mathrm{X}) = f \pmod{I_{[r]}}$. Then $R(\alpha_1, \ldots, \alpha_r, x_{r+1}, \ldots, x_n) = R_r(\alpha_1, \ldots, \alpha_r, x_{r+1}, \ldots, x_n) \pmod{I_{[n]\setminus[r]}}$.*

Proof. From the uniqueness of the remainder for the univariate ideals, we get that $R(\mathrm{X}) = R_r(\mathrm{X}) \pmod{I_{[n]\setminus[r]}}$. Now we write explicitly the polynomial $R_r(\mathrm{X})$ as $R_r = \sum_{\bar{u}} r_{\bar{u}} \cdot x_{r+1}^{u_1} \ldots, x_n^{u_{n-r}}$ where $r_u \in \mathbb{F}[\mathrm{X}_{[r]}]$. So we get that,

$$R_r \pmod{I_{[n]\setminus[r]}} = \sum_{\bar{u}} r_{\bar{u}} \prod_{j=1}^{n-r} q(x_{r+j})$$

where $q(x_{r+j}) = x_{r+j}^{u_j} \pmod{p(x_{r+j})}$. Then the lemma follows by substituting $x_1 = \alpha_1, \ldots, x_r = \alpha_r$ in the relation $R = R_r \pmod{I_{[n]\setminus[r]}}$. ◀

We require the following lemma in the proof of the main result of this section.

▶ **Lemma 20.** *Let $f \in \mathbb{F}[\mathrm{X}]$, and $T : \mathbb{F}^n \to \mathbb{F}^n$ be an invertible linear transformation fixing $x_1, \ldots, x_r$ and mapping $x_{r+1}, \ldots, x_n$ to linearly independent linear forms over $x_{r+1}, \ldots, x_n$. Write $R = f \pmod{I_{[r]}}$ and $R' = T(f) \pmod{I_{[r]}}$. Then $R' = T(R)$.*

Proof. Let $f = \sum_{i=1}^{r} h_i(\mathrm{X}) \cdot p_i(x_i) + R(\mathrm{X})$ and $T(f) = \sum_{i=1}^{r} h'_i(\mathrm{X}) \cdot p_i(x_i) + R'(\mathrm{X})$. Note that $\deg_{x_i} R, \deg_{x_i} R' < \deg(p_i(x_i))$ for $1 \leq i \leq r$. Since T is invertible and also fixes $x_1, \ldots, x_r$, we can write $f = \sum_{i=1}^{r} T^{-1}(h'_i(\mathrm{X})) \cdot p_i(x_i) + T^{-1}(R'(\mathrm{X}))$. By the property of T it is clear that $\deg_{x_i}(T^{-1}(R'(\mathrm{X}))) < \deg(p_i(x_i))$ for $1 \leq i \leq r$. Combining two expression for f, we immediately conclude that $(R - T^{-1}(R')) = 0 \pmod{I_{[r]}}$ which forces that $R = T^{-1}(R')$. ◀

3.1 Proof of Theorem 3

Proof. We now describe a recursive procedure REM to solve the problem. The initial call to it is $\mathrm{REM}(f(\ell_1, \ldots, \ell_r), I_{[n]}, \vec{\alpha})$. We apply the invertible linear transformation obtained in Lemma 18 to get the polynomial $T(f)$ over the variables $x_1, \ldots, x_r, x_{r+1}, \ldots, x_{r+r'}$ where $r' \leq r$.[2] The polynomial $T(f)$ can be explicitly computed in time $\mathrm{poly}(L, s, n, d^{O(r)})$. Then we compute the remainder polynomial $f'(x_1, \ldots, x_{r+r'}) = T(f) \pmod{I_{[r]}}$ by applying the division algorithm which runs in time $\mathrm{poly}(L, s, n, d^{O(r)})$. Next we compute the polynomial $g = f'(\alpha_1, \ldots, \alpha_r, x_{r+1}, \ldots, x_{r+r'})$. Notice from Lemma 18 that $T^{-1}(x_{r+i}) = \ell_{i,2}$ for $1 \leq i \leq r'$, thus we are interested in the polynomial $g(\ell_{1,2}, \ldots, \ell_{r',2})$. Now we recursively compute $\mathrm{REM}(g(\ell_{1,2}, \ldots, \ell_{r',2}), I_{[n]\setminus[r]}, \vec{\alpha}')$ where $\vec{\alpha}' = (\alpha_{r+1}, \ldots, \alpha_n)$.

[2] We use f to denote $f(\ell_1, \ldots, \ell_r)$.

Correctness of the algorithm. Let $R(\mathrm{X}) = f \pmod{I_{[n]}}$ be the unique remainder polynomial. Let $R_r(\mathrm{X}) = f \pmod{I_{[r]}}$ and we know that $R_r \pmod{I_{[n]\setminus[r]}} = R$. So by Lemma 19, to show the correctness of the algorithm, it is enough to show that $g(\ell_{1,2}, \ldots, \ell_{r',2}) = R_r(\alpha_1, \ldots, \alpha_r, x_{r+1}, \ldots, x_n)$.

Following Lemma 20, write $R' = f'(x_1, \ldots, x_r, x_{r+1}, \ldots, x_n) = T(f) \pmod{I_{[r]}}$. Then, by Lemma 20 we conclude that $R' = T(R_r)$. It immediately follows that $R_r = T^{-1}(R') = f'(x_1, \ldots, x_r, T^{-1}(x_{r+1}), \ldots, T^{-1}(x_n))$. Now by definition the polynomial $g(\ell_{1,2}, \ldots, \ell_{r',2})$ is $f'(\alpha_1, \ldots, \alpha_r, T^{-1}(x_{r+1}), \ldots, T^{-1}(x_{r+r'}))$ which is simply $R_r(\alpha_1, \ldots, \alpha_r, x_{r+1}, \ldots, x_n)$.

Time complexity. First, suppose that the field arithmetic over $\mathbb{F}$ can be implemented using polynomial bits and L be the bit-size upper bound for any coefficient in $f, p_1, \ldots, p_n$. This covers all the finite fields where the field is given by an explicit irreducible polynomial. Also, over any such field the polynomial $T(f)$ can be explicitly computed from the input arithmetic circuit deterministically in time $\mathsf{poly}(L, s, n, d^{O(r)})$.

Notice that in each recursive application the number of generators in the ideal is reduced by at least one. Furthermore, in each recursive step we need time $\mathsf{poly}(L, s, n, d^{O(r)})$ to run the division algorithm. This gives us a recurrence of $t(n) \leq t(n-1) + \mathsf{poly}(L, s, n, d^{O(r)})$ which solves to $t(n) \leq \mathsf{poly}(L, s, n, d^{O(r)})$.

Bit-size growth over $\mathbb{Q}$: Over $\mathbb{Q}$, we only need to argue that the intermediate bit-size complexity growth is only polynomial in the input size.Let $\tilde{L}$ be the maximum bit size of any coefficient appearing in $f(z_1, \ldots, z_r)$, and let L be an upper bound on the bit sizes of the other inputs, i.e. bit sizes of coefficients of $\ell_1, \ldots, \ell_r, p_1, \ldots, p_n$ and $\alpha_1, \ldots, \alpha_n$. We will show that the circuit that we use in the next recursive step has coefficients of bit size at most $\tilde{L} + \mathsf{poly}(n, d, L)$.

Let $|c(h)|$ denote the maximum coefficient (in absolute value) appearing in any polynomial h. Then by direct expansion we can see that $|c(f(\ell_1, \ldots, \ell_r))| \leq 2^{\tilde{L}+\mathsf{poly}(n,d,L)}$. Also the linear transformation from lemma 18 can be implemented using poly-bit size entries. Together, we get that that $c(T(f(\ell_1, \ldots, \ell_r)) \leq 2^{\tilde{L}+\mathsf{poly}(n,d,L)}$. At this point, we expand the circuit and obtain $T(f)$ explicitly as a sum of $d^{O(r)}$ monomials. Then divide $T(f)$ by $p_1(x_1), \ldots, p_r(x_r)$ one-by-one, and substitute $x_1 = \alpha_1, \ldots, x_r = \alpha_r$ giving us the remainder $g(x_{r+1}, \ldots, x_{r+r'})$. We note that $|c(g)| \leq 2^{\tilde{L}+\mathsf{poly}(n,d,L)}$ [3]. Now the algorithm passes the $d^{O(r)}$ size $\Sigma\Pi\Sigma$ circuit $g(\ell_{1,2}, \ldots, \ell_{r',2})$ (We note that $T^{-1}(x_{r+1}) = \ell_{1,2}, \ldots, T^{-1}(x_{r+r'}) = \ell_{r',2}$), univariates $p_{r+1}(x_{r+1}), \ldots, p_n(x_n)$ and the point $(\alpha_{r+1}, \ldots, \alpha_n)$ for the next recursive call. We note that the bit-size upper bound L does not change for the input linear forms, and the coefficient bit-size of f grows from $\tilde{L}$ to $\tilde{L} + \mathsf{poly}(n, d, L)$ in one step of the recursion. This gives us the recurrence $S(n) \leq S(n-1) + \mathsf{poly}(n, d, L)$ with $S(1) = \tilde{L}$, which solves to $S(n) = O(\tilde{L} + \mathsf{poly}(n, d, L))$. ◀

▶ Remark. Given a rank r polynomial $f(\ell_1, \ldots, \ell_r)$ and a univariate ideal $I = \langle p_1(x_1), \ldots, p_n(x_n)\rangle$, we can decide the membership of f in I by testing identity of $f \pmod{I}$ i.e. by evaluating $f \pmod{I}$ at some $\alpha \in \mathbb{F}^n$ chosen randomly [11, 29, 26]. Hence, the membership can be decided in randomized $d^{O(r)} \cdot \mathsf{poly}(n)$ time where $d = \max\{\deg(f), \deg(p_i) : 1 \leq i \leq n\}$ using Theorem 3.

[3] We tackle a similar situation in Section 5, and Lemma 33 gives further explanation on the bit-complexity growth when we divide by univariate polynomials.

3.2 Vertex Cover Detection in Low Rank Graphs

In the Vertex Cover problem, we are given a graph $G = (V, E)$ on n vertices and an integer k and the question is to decide whether there is a vertex cover of size k in G. This is a classical NP-complete problem. In this section we show an efficient algorithm to detect vertex cover in a graph whose adjacency matrix is of low rank.

Proof of Theorem 4. We present a reduction from Vertex Cover problem to Univariate Ideal Membership problem that produces a polynomial whose rank is almost same as the rank of A_G. Consider the ideal $I = \langle x_1^2 - x_1, x_2^2 - x_2, \dots, x_n^2 - x_n \rangle$ and the polynomial

$$f = \prod_{s=1}^{\binom{n}{2}} (\vec{x} A_G \vec{x}^T - s) \cdot \prod_{t=0}^{n-k-1} \left(\sum_{i=1}^{n} x_i - t \right),$$

where A_G is the adjacency matrix of the graph G and $\vec{x} = (x_1, x_2, \dots, x_n)$ is row-vector.

▶ **Lemma 21.** *The rank of the polynomial f is at most $r + 1$.*

Proof. We note that A_G is symmetric since it encodes an undirected graph. Let Q be an invertible $n \times n$ matrix that diagonalizes A_G. So we have $Q A_G Q^T = D$ where D is a diagonal matrix with only the first r diagonal elements being non-zero. Let $\vec{y} = (y_1, y_2, \dots, y_n)$ be another row-vector of variables. Now, we show the effect of the transform $\vec{x} \mapsto \vec{y}Q$ on the polynomial $\vec{x} A_G \vec{x}^T$. Clearly, $\vec{y} Q A_G Q^T \vec{y}^T = \vec{y} D \vec{y}^T$ and since there are only r non-zero entries on the diagonal, the polynomial $\vec{y} D \vec{y}^T$ is over the variables $y_1, y_2, \dots, y_r$. Thus $g = \prod_{s=1}^{\binom{n}{2}} (\vec{x} A_G \vec{x}^T - s)$ is a rank r polynomial. Also $h = \prod_{t=0}^{n-k-1} (\sum_{i=1}^{n} x_i - t)$ is a rank 1 polynomial as there is only one linear form $\sum_{i=1}^{n} x_i$. Since $f = gh$, we conclude that f is a rank $r + 1$ polynomial. ◀

Now the proof of Theorem 4 follows from the next claim.

▶ **Claim 22.** *The graph G has a Vertex Cover of size k if and only if $f \notin I$.*

Proof. First, observe that the set of common zeroes of the generators of the ideal I is the set $\{0, 1\}^n$. Let S be a vertex cover in G such that $|S| \leq k$. We will exhibit a point $\vec{\alpha} \in \{0, 1\}^n$ such that $f(\vec{\alpha}) \neq 0$. This will imply that $f \notin I$. Identify the vertices of G with $\{1, 2, \dots, n\}$. Define $\vec{\alpha}(i) = 0$ if and only if $i \in S$. Since $\vec{x} A_G \vec{x}^T = \sum_{(i,j) \in E_G} x_i x_j$ and S is a vertex cover for G, it is clear that $\vec{x} A_G \vec{x}^T(\vec{\alpha}) = 0$. Also $(\sum_{i=1}^{n} x_i)(\vec{\alpha}) \geq n - k$. Then clearly $f(\vec{\alpha}) \neq 0$.

For the other direction, suppose that $f \notin I$. Then by Theorem 1, there exists $\vec{\alpha} \in \{0, 1\}^n$ such that $f(\vec{\alpha}) \neq 0$. Define the set $S \subseteq [n]$ as follows. Include $i \in S$ if and only if $\vec{\alpha}(i) = 0$. Since $f(\vec{\alpha}) \neq 0$, and the range of values that $\vec{x} A_G \vec{x}^T$ can take is $\{0, 1, \dots, |E|\}$, it must be the case that $\vec{x} A_G \vec{x}^T(\vec{\alpha}) = 0$. It implies that the set S is a vertex cover for G. Moreover, $\prod_{t=0}^{n-k-1} (\sum_{i=1}^{n} x_i - t)(\vec{\alpha}) \neq 0$ implies that $|S| \leq k$. ◀

The degree of the polynomial f is bounded by $n^2 + n$ and from Claim 22 we know that $f \pmod{I}$ is a non-zero polynomial if and only if G has a vertex cover of size k. By Schwartz-Zippel-Demillo-Lipton [11, 29, 26] lemma $(f \pmod{I})(\vec{\beta})$ is non-zero with high probability when $\vec{\beta}$ is chosen randomly from a small domain. Now, we need to just compute $(f \pmod{I})(\vec{\beta})$ where f is a rank $r + 1$ polynomial with $\ell_i = (\vec{x} Q^{-1})_i$ for each $1 \leq i \leq r$ and $\ell_{r+1} = \sum_{i=1}^{n} x_i$ which can be performed in $(n, k)^{O(r)}$ time using Theorem 3. ◀

4 Parameterized Complexity of Univariate Ideals

We have already mentioned in Fact 2, that checking if the integer permanent is zero is reducible to testing membership of a polynomial $f(\mathrm{X})$ in the ideal $\langle x_1^2, \ldots, x_n^2\rangle$. So univariate ideal membership is hard for the complexity class $\mathrm{C}_=\mathrm{P}$ even when the ideal is generated by powers of variables [23]. In this section we study the univariate ideal membership with the lens of parametrized complexity. The parameters we consider are either polynomial degree or number of the generators for the ideal.

4.1 Parameterized by the Degree of the Polynomial

We consider the following: Let I be a univarite ideal given by generators and $f \in \mathbb{F}[\mathrm{X}]$ a degree k polynomial. Is checking whether f is in I fixed parameter tractable (with k as the fixed parameter)?

We show that it admits an FPT algorithm for the special case when $I = \langle x_1^{e_1}, x_2^{e_2}, \ldots, x_n^{e_n}\rangle$ and we work over either $\mathbb{Z}$ or any finite field of large characteristic.

4.1.1 Proof of Theorem 5

Proof. The proof consists of following three lemmas. Firstly, given an input instance a degree-k $f(\mathrm{X})$ and ideal $I = \langle x_1^{e_1}, x_2^{e_2}, \ldots, x_n^{e_n}\rangle$ of ideal membership, we reduce it to computing the (scaled) Hadamard product of $f(\mathrm{X})$ and a polynomial $g(\mathrm{X})$, where $g(\mathrm{X})$ is a weighted sum of all degree k monomials that are not in I. Then we show that we can evaluate Hadamard product(defined in Section 2) of any two polynomials at a point in time roughly linear in the product of the size of the circuits when one of the polynomials is given by a diagonal circuit as input. Finally the last part of the proof is a randomized construction of a homogeneous degree k diagonal circuit of top fain-in roughly $O^*(4.08^k)$ that computes a polynomial weakly equivalent to the polynomial g with constant probability. Recall that, two polynomials f and g are said to be *weakly equivalent* if they share same set of monomials.

To define the polynomial $g(\mathrm{X})$, let $S_{m,k}$ be the elementary symmetric polynomial of degree k over m variables. Set $m = \sum_{i=1}^{n}(e_i - 1)$. Let $S_{m,k}$ is defined over the variable set $\{z_{1,1}, \ldots, z_{1,e_1-1}, \ldots, z_{n,1}, \ldots, z_{n,e_n-1}\}$. We define $g(\mathrm{X})$ as the polynomial obtained from $S_{m,k}$ replacing each $z_{i,j}$ by x_i.

▶ **Lemma 23.** *Given integers $e_1, e_2, \ldots, e_n$, and a polynomial $f(\mathrm{X})$ of degree k, $f \in \langle x_1^{e_1}, x_2^{e_2}, \ldots, x_n^{e_n}\rangle$ if and only if $f \circ^s g \equiv 0$.*

Proof. Suppose, $f \notin \langle x_1^{e_1}, x_2^{e_2}, \ldots, x_n^{e_n}\rangle$, then f must contain a degree k monomial $M = x_1^{f_1}x_2^{f_2}\ldots x_n^{f_n}$ such that $f_i < e_i$ for each $1 \le i \le n$. From the construction, it is clear that $g(\mathrm{X})$ contains M. Therefore, the polynomial $f \circ^s g$ is not identically zero. The converse is also true for the similar reason. ◀

▶ **Lemma 24.** *Given a circuit C of size s computing a polynomial $g \in \mathbb{F}[\mathrm{X}]$ and a homogeneous degree k diagonal circuit $\Sigma^{[s']}\wedge^{[k]}\Sigma$ circuit D of top fan-in s' computing $f \in \mathbb{Q}[\mathrm{X}]$ and $\vec{a} \in \mathbb{Q}^n$, we can evaluate $(f \circ^s g)(\vec{a})$ in deterministic $ss' \cdot \mathrm{poly}(n,k)$ time using $\mathrm{poly}(n,k)$ space.*

Proof. Let M be a degree d monomial over X in f and $M = x_1^{e_1}\cdots x_n^{e_n}$, it follows from the definition that

$$\left(M \circ^s (b_1x_1 + \ldots + b_nx_n)^d\right)(\vec{a}) = \left(M! \cdot \frac{d!}{M!} \cdot b_1^{e_1}\cdots b_n^{e_n} \cdot M\right)(\vec{a}) = d! \cdot M(a_1b_1, \ldots, a_nb_n).$$

Recall that, $M!$ is used for $e_1! \cdots e_n!$. As $\circ^s$ distributes over addition, we can write

$$\left(f \circ^s \sum_{i=1}^{s'} (b_{i1}x_1 + \ldots + b_{in}x_n)^d \right) (\vec{a}) = d! \cdot \sum_{i=1}^{s'} f(a_1 b_{i1}, \ldots, a_n b_{in}).$$

The computation can be done in deterministic $ss' \cdot \mathrm{poly}(n,k)$ time using $\mathrm{poly}(n,k)$ space. ◀

▶ **Lemma 25.** *There is an efficient randomized algorithm that constructs with constant probability a homogeneous degree k diagonal circuit D of top fan-in $O^*(4.08^k)$ which computes a polynomial weakly equivalent to the polynomial g (defined before Lemma 23).*

Proof. To construct such a diagonal circuit D, we use the idea of [22]. We pick a collection of colourings $\{\zeta : [m] \to [1.5 \cdot k]\}$ of size roughly $O^*((\frac{e}{\sqrt{3}})^k)$ uniformly at random. For each such colouring ζ_i, we define a $\Pi^{[1.5 \cdot k]}\Sigma$ formula $P_i = \prod_{j=1}^{1.5k}(L_j + 1)$, where $L_j = \sum_{\ell : \zeta_i(\ell) = j} x_\ell$. We say that a monomial is *covered* by a coloring ζ_i if the monomial is in P_i. It is easy to see that, given any multilinear monomial of degree k, the probability that a random coloring will cover the monomial is roughly $(\frac{\sqrt{3}}{e})^k$. Hence, going over such a collection of colorings of size $O^*((\frac{e}{\sqrt{3}})^k)$ chosen uniformly at random, with a constant probability all the multilinear terms of degree k will be covered. To take the Hadamard product with a polynomial of degree k, we need to extract out the degree k homogeneous part (say P_i') from each P_i. Notice that, using elementary symmetric polynomial over $1.5k$ many variables $S_{1.5k,k}$, we can write $P_i' = S_{1.5k,k}(L_1, \ldots, L_{1.5k})$. Now we use Lemma 16 to get a diagonal $\Sigma \wedge^{[k]} \Sigma$ circuit of top fan-in roughly $\binom{1.5k}{0.5k}$ for each P_i'. Define $D = \sum_{i=1}^{O^*((\frac{e}{\sqrt{3}})^k)} P_i'$. By a direct calculation, one can obtain a diagonal circuit D of top fan-in $O^*(4.08^k)$ which is weakly equivalent to the polynomial $S_{m,k}$. The construction of the polynomial $g(\mathrm{X})$ from $S_{m,k}$ is already explained before Lemma 23. ◀

Now, given a circuit C computing $f \in \mathbb{Z}[\mathrm{X}]$ and integers $e_1, \ldots, e_n$, to decide the membership of f in the ideal $I = \langle x_1^{e_1}, \ldots, x_n^{e_n} \rangle$, we construct a diagonal circuit D from Lemma 25. Following Lemma 23, we can decide the membership of f in the ideal checking the polynomial $C \circ^s D$ is identically zero or not which can be performed by randomly picking $\vec{a} \in \mathbb{Z}^n$ using Schwartz-Zippel-Demillo-Lipton Lemma [26, 29, 11] and evaluating $(C \circ^s D)(\vec{a})$ using Lemma 24. Over $\mathbb{Z}$ the given circuit can compute numbers as large as $2^{2^{n^{O(1)}}}$. To handle this, a standard idea is to evaluate the circuit modulo a random polynomial bit prime. ◀

4.2 Parameterized by Number of Generators

In this section, we consider the univariate ideal membership parameterized on the number of generators of the ideal. More precisely, given a polynomial $f(\mathrm{X})$, can we obtain an FPT algorithm for testing membership in the univariate ideal $\langle p_1(x_1), \ldots, p_k(x_k) \rangle$ parameterized by k? We show that the problem is W[1]-hard. Moreover, in contrast to the previous case, we obtain MINI[1]-hardness for a special case of the problem when the univariate generators are just power of variables.

Proof of Theorem 6. We show a reduction from *k-independent set*, a well known W[1]-hard problem [10], to this problem. Let $G = (V, E)$ be a graph on n vertices and k be the size of the independent set. We identify its vertex set with the numbers $\{1, 2, \ldots, n\}$ and the edges are tuples over $[n] \times [n]$. Define the univariate ideal $I = \langle p_1(x_1), \ldots, p_k(x_k) \rangle$ where for each

$1 \leq i \leq k$, we define $p_i(x_i) = \prod_{j=1}^{n}(x_i - j)$. Now we are going to define a polynomial f that uses only k variables which will be used for the ideal membership problem. First consider the polynomial $D = \prod_{1 \leq i \neq j \leq k}(x_i - x_j)$.

Now we define the polynomial,

$$f = \prod_{1 \leq i \neq j \leq k} \; \prod_{(u,v) \in E \subseteq [n] \times [n]} [(x_i - u)^2 + (x_j - v)^2] \cdot [(x_j - u)^2 + (x_i - v)^2].$$

The proof follows from the following claim.

▶ **Claim 26.** *$f \cdot D \notin \langle p_1(x_1), p_2(x_2), \ldots, p_k(x_k)\rangle$ if and only if G has an independent set of size k.*

Proof. We use Theorem 1 to prove the claim. Let $\{j_1, j_2, \ldots, j_k\}$ be an independent set in G. Notice that $(j_1, \ldots, j_k)$ is a common zero of the generators $p_1, \ldots, p_k$. Now notice that $f \cdot D$ does not vanish at the point $(j_1, \ldots, j_k)$ as all the edges $(j_\ell, j_{\ell'}) : 1 \leq \ell, \ell' \leq k$ are absent in the edge set E. Thus there is a common root of the ideal on which $f \cdot D$ does not vanish and hence $f \cdot D \notin \langle p_1(x_1), p_2(x_2), \ldots, p_k(x_k)\rangle$.

Now if $f \cdot D \notin \langle p_1(x_1), p_2(x_2), \ldots, p_k(x_k)\rangle$ then there is a common zero $(j_1, \ldots, j_k)$ of the ideal on which $f \cdot D$ does not vanish. Using the same argument one can easily see that $\{j_1, \ldots, j_k\}$ is an independent set in G. ◀

◀

4.2.1 Proof of Theorem 7

We first relate our univariate ideal membership problem with a linear algebraic problem k-LIN-EQ. It turns that k-LIN-EQ problem is more amenable to the MINI[1]-hardness proof. Finally we show a reduction from MINI-1-in-3 POSITIVE 3-SAT to k-LIN-EQ to complete the proof.

▶ **Definition 27.** k-LIN-EQ
Input: Integers k, n in unary, a $k \times n$ matrix A with all the entries given in unary and a k dimensional vector $\vec{b}$ with all entries in unary.
Parameter: k.
Question: Does there exist an $\vec{x} \in \{0,1\}^n$ such that $A\vec{x} = \vec{b}$?

▶ **Lemma 28.** *There is a parameterized reduction from k-LIN-EQ to the univariate ideal membership problem when the ideal is given by the powers of variables as generators.*

Proof. We introduce $2k$ variables $x_1, x_2, \ldots, x_k, y_1, y_2, \ldots, y_k$ where two variables will be used for each row. For each $i \in [n]$, let $\mu_i = \sum_{j=1}^{n} a_{ij}$. For each column $c_i = (a_{1i}, a_{2i}, \ldots, a_{ki})$ we construct the polynomial $P_i = ({y_1}^{a_{1i}}{y_2}^{a_{2i}} \ldots {y_k}^{a_{ki}} + {x_1}^{a_{1i}}{x_2}^{a_{2i}} \ldots {x_k}^{a_{ki}})$. We let $P_A = \prod_{i=1}^{n} P_i$ and we choose the ideal to be $\langle x_1^{b_1+1}, y_1^{\mu_1 - b_1 + 1}, \ldots, x_k^{b_k+1}, y_1^{\mu_k - b_k + 1}\rangle$. Notice that P_A has a small arithmetic circuit which is polynomial time computable.

▶ **Claim 29.** *An instance $(A, \vec{b})$ is an YES instance for k-LIN-EQ iff $P_A \notin \langle x_1^{b_1+1}, y_1^{\mu_1 - b_1 + 1}, \ldots, x_k^{b_k+1}, y_k^{\mu_k - b_k + 1}\rangle$.*

Proof of Claim. Suppose $(A, \vec{b})$ is an YES instance. Then there is an $\vec{x} \in \{0,1\}^n$ such that $A\vec{x} = \vec{b}$. Define $S := \{i \in [n] : \vec{x}_i = 1\}$ where $\mathbf{x_i}$ is the ith co-ordinate of $\vec{x}$. Think of the monomial where ${x_1}^{a_{1i}}{x_2}^{a_{2i}} \ldots {x_k}^{a_{ki}}$ is picked from P_i for each $i \in S$ and ${y_1}^{a_{1i}}{y_2}^{a_{2i}} \ldots {y_k}^{a_{ki}}$

is picked from reaming P_j's where $j \in \bar{S}$. This gives us the monomial $x_1^{b_1} y_1^{\mu_1 - b_1} \dots x_k^{b_k} y_1^{\mu_k - b_k}$ in the polynomial P_A. Thus $P_A \notin \langle x_1^{b_1+1}, y_1^{\mu_1-b_1+1}, \dots, x_k^{b_k+1}, y_k^{\mu_k-b_k+1} \rangle$.

Now we show the other direction. Now suppose $P_A \notin \langle x_1^{b_1+1}, y_1^{\mu_1-b_1+1}, \dots, x_k^{b_k+1}, y_k^{\mu_k-b_k+1} \rangle$. Let $S := \{i \in [n] : {x_1}^{a_{1i}} {x_2}^{a_{2i}} \dots {x_k}^{a_{ki}}$ is picked from $P_i\}$. There must be a monomial ${x_1}^{c_1} {x_2}^{c_2} \dots {x_k}^{c_k} {y_1}^{d_1} {y_2}^{d_2} \dots {y_k}^{d_k}$ in P_A such that for each i, $\sum_{j \in S} a_{ij} = c_i \leq b_i$, $\sum_{j \notin S} a_{ij} = d_i \leq (\mu_i - b_i)$. As, $\mu_i = \sum_{j \in S} a_{ij} + \sum_{i \notin S} a_{ij}$, we get $b_i \leq \sum_{j \in S} a_{ij}$. Hence, $\sum_{j \in S} a_{ij} = b_i$ for each i. Define $\vec{x} \in \{0,1\}^n$ where $\vec{x}_i = 1$ if $i \in S$ else $\vec{x}_i = 0$. This shows $(A, \vec{b})$ is an YES instance. ◀

◀

Before we prove the MINI[1]-hardness of k-LIN-EQ, we show that the following problem is MINI[1]-hard.

▶ **Definition 30.** MINI-1-in-3 POSITIVE 3-SAT
Input: Integers k, n in unary, a 3-SAT instance $\mathcal{E}$ consisting of only positive literals where $\mathcal{E}$ has at most $k \log n$ variables and atmost $k \log n$ clauses.
Parameter: k.
Question: Does there exist a satisfiable assignment for $\mathcal{E}$ such that every clause has exactly one TRUE literal?

▶ **Claim 31.** MINI-1-in-3 POSITIVE 3-SAT *is* MINI[1]*-hard.*

To prove the claim we only need to observe that the standard *Schaefer Reduction* [25] from 3-SAT to 1-in-3 POSITIVE 3-SAT is in fact a linear size reduction, that directly gives us an FPT reduction from MINI-3SAT to MINI-1-in-3 POSITIVE 3-SAT.

Proof of Theorem 7. Given a MINI-1-in-3 POSITIVE 3-SAT instance $\mathcal{E}$, order the variables $v_1, \dots, v_{k \log n}$ and the clauses $C_1, \dots, C_{k \log n}$. Construct the following $k \log n \times k \log n$ matrix M where the rows are indexed by the clauses and the columns are indexed by the variables. $M[i][j]$ is set to 1 if v_j appears in C_i, otherwise set it to 0. Make M a $2k \log n \times n$ matrix by adding an all zero row between every rows and appending all zero columns at the end. Now, define $\vec{e}$ as a $2k \log n$ dimensional vector where ith co-ordinate of e, $e_i = 1$ when i is odd and $e_i = 0$ when i is even. We want to find $\vec{y} \in \{0,1\}^n$ such that $M\vec{y} = \vec{e}$.

However this is not an instance of k-LIN-EQ. To make it so, we observe that M is a bit matrix and $\vec{e}$ is a bit vector, hence we can modify them to a $k \times n$ matrix A and k dimensional vector $\vec{b}$ in the following way. For each column j, think of the ith consecutive $2 \log n$ bits as the binary expansion of a single entry, call it N and set $A[i][j]$ to N. Similarly, we modify $\vec{e}$ to a k dimensional vector $\vec{b}$ by considering $2 \log n$ bits as a binary expansion of a single entry. Now the proof follows from the following claim.

▶ **Claim 32.** *$\mathcal{E}$ is an YES instance for* MINI-1-in-3 POSITIVE 3-SAT *if and only if there exists an $\vec{x} \in \{0,1\}^n$ such that $A\vec{x} = \vec{b}$.*

Proof. Suppose there is such a satisfiable assignment for $\mathcal{E}$. Define $S := \{j \in [k \log n] \mid v_j = \text{TRUE}\}$. Define $\vec{z} \in \{0,1\}^n$ such that $z_j = 1$ where $j \in S$ else $z_j = 0$. For each i, as C_i contains exactly one TRUE literal, hence $e_{2i+1} = \sum_{j=1}^{n} M[i][j] \cdot z_j = 1$ and $e_{2i} = 0$. Therefore $\vec{z}$ is a solution for $M\vec{y} = \vec{e}$. As every integer has a unique binary expansion, hence $\vec{z}$ is also a solution for $A\vec{x} = \vec{b}$.

Now we prove the other direction. Suppose $A\vec{z} = \vec{b}$ for some $\vec{z} \in \{0,1\}^n$. From the construction of the matrix M, it is sufficient to show that $\vec{z}$ is a satisfying assignment for $M\vec{y} = \vec{e}$. First we note that the numbers $A[i][j], b[i]$ in their binary expansion have bits 1 in the odd location and 0 in the even locations. Let $A[i][j] = \sum_{t=1}^{2\log n} a_{ijt}2^{t-1}$ and $b[i] = \sum_{t=1}^{2\log n} e_t 2^{t-1}$. Since $A\vec{z} = \vec{b}$ we have $\sum_{j=1}^{n} A[i][j] \cdot z_j = b[i]$. This shows that

$$\sum_{j=1}^{n} A[i][j] \cdot z_j = \sum_{j=1}^{n} \left(\sum_{t=1}^{2\log n} a_{ijt}2^{t-1} \right) \cdot z_j = \sum_{t=1}^{2\log n} \left(\sum_{j=1}^{n} a_{ijt} \cdot z_j \right) 2^{t-1}.$$

Since $\mathcal{E}$ is a 3-CNF formula we have $(\sum_{j=1}^{n} a_{ijt} \cdot z_j) \in \{0, 1, 2, 3\}$. Now we compare $(\sum_{j=1}^{n} a_{ijt} \cdot z_j)$ with the binary expansion of $b[i]$. When t is odd the bit e_t is 1 and so there must be a 1 in the corresponding bit of $(\sum_{j=1}^{n} a_{ijt} \cdot z_j)$. This shows that $(\sum_{j=1}^{n} a_{ijt} \cdot z_j) \neq 0$ when t is odd. Now if $(\sum_{j=1}^{n} a_{ijt} \cdot z_j) \in \{2, 3\}$ for any odd t then the term 2^{t+1} will be produced and this will not match the expansion of $b[i]$ as the $e_{t+1} = 0$. Thus by the uniqueness of binary expansion we conclude that $(\sum_{j=1}^{n} a_{ijt} \cdot z_j) = 1$ if t is odd and 0 otherwise. Thus $M\vec{y} = \vec{e}$ has a solution with $y_i = z_i$. ◀

◀

5 Non-deterministic Algorithm for Univariate Ideal Membership

In this section we prove Theorem 8. Given a polynomial $f(\mathrm{X}) \in \mathbb{Q}[\mathrm{X}]$ and a univariate ideal $I = \langle p_1(x_1), \ldots, p_n(x_n) \rangle$ where the generators are $p_1, \ldots, p_n$, we show a non-deterministic algorithm to decide the (non)-membership of f in I. By Theorem 1, it suffices to show that there is a common zero $\vec{\alpha}$ of the generators $p_1, p_2, \ldots, p_n$ such that $f(\alpha) \neq 0$. Since in general $\vec{\alpha} \in \mathbb{C}^n$, it is not immediately clear how to guess such a common zero by a NP machine. However, we are able to show that for the NP machine it suffices to guess such an $\vec{\alpha}$ upto polynomially many bits of approximation.

We begin by proving a few technical facts which are useful for the main proof. Write $f(\mathrm{X}) = \sum_{i=1}^{n} h_i(\mathrm{X})\ p_i(x_i) + R(\mathrm{X})$ where for all $i \in [n]$, $\deg_{x_i}(R) < \deg(p_i)$. For any polynomial g, let $|c(g)|$ be the maximum coefficient (in absolute value) appearing in g. The following lemma gives an estimate for the coefficients of the polynomials $h_1, \ldots, h_n, R$.

▶ **Lemma 33.** *Let $2^{-L} \leq |c(f)|, |c(p_i)| \leq 2^L$. Then there is $L' = \mathrm{poly}(L, d, n)$ such that $2^{-L'} \leq |c(h_i)|, |c(R)| \leq 2^{L'}$ where d is the degree upper bound for f, and $\{p_i : 1 \leq i \leq n\}$.*

Proof. The estimate on L' follows implicitly from the known results [8]. It can be also seen by direct computation. Write $f(\mathrm{X}) = \sum_i f_i(x_2, \ldots, x_n)\ x_1^i$ and then divide $x_1^i \pmod{p_1(x_1)}$ for each i. The modulo computation can be done by writing $x_1^i = q_1(x_1)p(x_1) + r_1(x_1)$ with the coefficients of q_1 and r_1 are unknown. We can then solve it using standard linear algebra. In particular, one can use the Cramer's rule for system of linear equation solution. The growth of the bit-size is only $\mathrm{poly}(L, d)$. More precisely, if $c_{\max}$ is the maximum among $|c(f)|, |c(p_1)|$, any final coefficient is at most $c_{\max} \cdot 2^{\mathrm{poly}(L,d)}$. We repeat the procedure for the other univariate polynomials one by one. The final growth on the coefficients size is at most $\mathrm{poly}(n, L, d)$. ◀

Let $\vec{\alpha} = (\alpha_1, \ldots, \alpha_n) \in \mathbb{C}^n$ be such that $p_i(\alpha_i) = 0$, $1 \leq i \leq n$. From Lemma 11, we get that $\frac{1}{2^{\hat{L}}} \leq |\alpha_i| \leq 2^{\hat{L}}$ where $\hat{L} = \mathrm{poly}(L, d)$. Let $\tilde{\alpha}_i \in \mathbb{Q}[i]$ be an ϵ-approximation of α_i, e.g. $|\alpha_i - \tilde{\alpha}_i| \leq \epsilon$. Then we show that the absolute value of $p_i(\tilde{\alpha}_i)$ is not too far from zero.

▶ **Observation 34.** *For $1 \leq i \leq n$ we have that $|p_i(\tilde{\alpha}_i)| \leq \epsilon \cdot 2^{(dL)^{O(1)}}$.*

Proof. Let $p_i(x_i) = c \cdot \prod_{j=1}^{d}(x_i - \beta_{i,j})$ and w.l.o.g assume that $\tilde{\alpha}_i$ is the approximation of the root $\beta_{i,1}$. Then $|p_i(\tilde{\alpha}_i)| \leq \epsilon \cdot |c| \cdot \prod_{j=2}^{d} |\tilde{\alpha}_i - \beta_{i,j}| \leq \epsilon \cdot |c| \cdot \prod_{j=2}^{d}(|\beta_{i,1} - \beta_{i,j}| + \epsilon) \leq \epsilon \cdot 2^{\mathrm{poly}(d,L)}$. The final bound follows from the bound on the roots given in Lemma 11. ◀

Since we have an upper bound on the coefficients of the polynomials $\{h_i : 1 \le i \le n\}$ from Lemma 33, it follows that for $1 \le i \le n$ we have that $|h_i(\tilde{\alpha})| \le 2^{(ndL)^{O(1)}}$. Here we use the fact that the approximate root α_i can be trivially bounded by $2^{\hat{L}+1}$.

5.1 Proof of Theorem 8

Proof. If f is not in the ideal I, by Alon's Nullstellensatz, we know that there exists a tuple $\vec{\alpha} = (\alpha_1, \ldots, \alpha_n) \in Z(p_1) \times \ldots \times Z(p_n)$ such that $R(\vec{\alpha}) \neq 0$. Suppose that the NP Machine guess the tuple $\vec{\tilde{\alpha}} = (\tilde{\alpha}_1, \ldots, \tilde{\alpha}_n)$ which is the ϵ-approximation of the tuple $\vec{\alpha} = (\alpha_1, \ldots, \alpha_n)$ [4]. Using the black-box for f, obtain the value for $f(\vec{\tilde{\alpha}})$. Next, we show that the value $|f(\vec{\tilde{\alpha}})|$ distinguishes between the cases $f \in I$ and $f \notin I$.

Case 1: *$f \in I$.* $|f(\vec{\tilde{\alpha}})| = |\sum_{i=1}^{n} h_i(\vec{\tilde{\alpha}})p_i(\tilde{\alpha_i})| \le (\sum_{i=1}^{n} |h_i(\vec{\tilde{\alpha}})|)\cdot\epsilon\cdot 2^{(dL)^{c_1}} \le \epsilon\cdot 2^{(ndL)^{c_2}}$. where the constant c_2 is fixed by Observation 34 and the bounds on $|h_i(\vec{\tilde{\alpha}})|$.

Case 2: *$f \notin I$.* Recall the inequality for complex numbers : $|Z_1 + Z_2| \ge |Z_2| - |Z_1|$. Using this write $|f(\vec{\tilde{\alpha}})| \ge |R(\vec{\tilde{\alpha}})| - \sum_{i=1}^{n} |h_i(\vec{\tilde{\alpha}})|\ |p_i(\vec{\tilde{\alpha}})|$. Notice that $|R(\vec{\tilde{\alpha}})| \ge |R(\vec{\alpha})| - |R(\vec{\tilde{\alpha}}) - R(\vec{\alpha})|$. Combining we get the following : $|f(\vec{\tilde{\alpha}}) \ge |R(\vec{\alpha})| - |R(\vec{\tilde{\alpha}}) - R(\vec{\alpha})| - \epsilon \cdot 2^{(ndL)^{c_2}}$.

Now to complete the proof, we show a lower bound on $|R(\vec{\alpha})|$ and an upper bound for $|R(\vec{\tilde{\alpha}}) - R(\vec{\alpha})|$.

▶ **Claim 35.** $|R(\vec{\alpha})| \ge \frac{1}{2^{(ndL)^{c_3}}}$ *for some constant c_3.*

Proof. Define the polynomial $\hat{R}(x_n) = R(\alpha_1, \ldots, \alpha_{n-1}, x_n) = c \cdot \prod_{j=1}^{d'}(x_n - \beta_j)$ where c is some constant and $d' \le d$. Note that α_n is not a zero for $\hat{R}(x_n)$. Consider the polynomial $Q(x_n) = p_n(x_n)\hat{R}(x_n)$. The set $\{\alpha_n, \beta_1, \ldots, \beta_{d'}\} \subseteq Z(Q)$ and $\alpha_n \neq \beta_j : 1 \le j \le d'$. Using the root separation bound for $|\alpha_n - \beta_j|$ obtained in Lemma 12, we can easily lower bound that $|\hat{R}(\alpha_n)| \ge \frac{1}{2^{(ndL)^{c_3}}}$. ◀

▶ **Claim 36.** $|R(\vec{\tilde{\alpha}}) - R(\vec{\alpha})| \le 2^{(ndL)^{c_4}}$ *for some constant c_4.*

Proof. Define $R^0(\vec{\tilde{\alpha}}) = R(\vec{\alpha})$ and $R^i(\vec{\tilde{\alpha}}) = R(\tilde{\alpha}_1, \ldots, \tilde{\alpha}_i, \alpha_{i+1}, \ldots, \alpha_n)$. Then we use triangle inequality to notice that $|R(\vec{\alpha}) - R(\vec{\tilde{\alpha}})| \le \sum_{i=1}^{n} |R^{i-1}(\vec{\tilde{\alpha}}) - R^i(\vec{\tilde{\alpha}})|$. Write explicitly $R^{i-1}(\vec{\tilde{\alpha}}) - R^i(\vec{\tilde{\alpha}}) = \sum_{\vec{e}} c_{\vec{e}}\tilde{\alpha}_1^{e_1} \ldots \tilde{\alpha}_{i-1}^{e_{i-1}}(\alpha_i^{e_i} - \tilde{\alpha}_i^{e_i})\alpha_i^{e_{i+1}} \ldots \alpha_n^{e_n}$. Notice the upper bounds on $|\alpha_i| \le 2^{(ndL)^{O(1)}}$, and $|\alpha_i - \tilde{\alpha}_i| \le \epsilon$. We apply these bounds and use triangle inequality to get that $|R(\vec{\tilde{\alpha}}) - R(\vec{\alpha})| \le \epsilon \cdot 2^{(ndL)^{c_4}}$. ◀

Combining Claim 35, and Claim 36, we get the lower bound $|f(\vec{\tilde{\alpha}})| \ge \frac{1}{2^{(ndL)^{c_3}}} - \epsilon \cdot (2^{(ndL)^{c_4}} + 2^{(ndL)^{c_2}})$. To make the calculation precise, let $3M = \frac{1}{2^{(ndL)^{c_3}}}$ and choose ϵ such that $\epsilon \cdot (2^{(ndL)^{c_4}} + 2^{(ndL)^{c_2}}) \le M$.

The final implication will be $|f(\vec{\tilde{\alpha}})| \le M$ when $f \in I$ and $|f(\vec{\tilde{\alpha}})| \ge 2M$ when $f \notin I$. It is important to note that the parameter M can be pre-computed from the input parameters efficiently.

Now we show how to verify that the guessed point $\vec{\tilde{\alpha}}$ is a good approximation of the roots for the univariate polynomials. We need to only verify that for each i, $\tilde{\alpha_i}$ is a good approximation for *some root* of the univariate polynomial $p_i(x_i)$. The fact that it is also a good approximation for the non-zero of R is already verified above. The NP machine, given

[4] Later we fix ϵ suitably and use Lemma 13 to verify in polynomial time that $\vec{\tilde{\alpha}}$ is indeed ϵ-approximation of $\vec{\alpha}$.

$p_1, \dots, p_n$ guesses $\tilde{\alpha_i}$ using b bits and verifies that $|p_i(\tilde{\alpha_i})| < 2^{-L}\epsilon^d$ which, by lemma 13, shows that the guessed $\tilde{\alpha_i}$ is ϵ-close to some root of p_i.

We note that such a guess always exists. Indeed, invoking Observation 34 with $|\alpha_i - \tilde{\alpha_i}| \le \delta$ we can conclude that $|p_i(\tilde{\alpha_i})| \le \delta \cdot 2^{(dL)^{O(1)}}$. Now, the NP machine can guess b bits such that $|\alpha_i - \tilde{\alpha_i}| \le 2^{-b}$. We require $2^{-b} \cdot 2^{(dL)^{O(1)}} < 2^{-L}\epsilon^d$, simplifying we get, $2^{-b} < 2^{-(dL)^{O(1)}} \cdot \epsilon^d$. Hence $b > (dL)^{O(1)} \log \frac{1}{\epsilon}$. Thus using $\text{poly}(d, L, \log \frac{1}{\epsilon})$ bits there is always a guess $\tilde{\alpha_i}$ for which $|p_i(\tilde{\alpha_i})| < 2^{-L}\epsilon^d$. ◀

References

1 Noga Alon. Combinatorial Nullstellensatz. *Comb. Probab. Comput.*, 8(1-2):7–29, January 1999. URL: `http://dl.acm.org/citation.cfm?id=971651.971653`.

2 Kotlov Andrew and Lovász László. The rank and size of graphs. *Journal of Graph Theory*, 23(2):185–189, 1996.

3 Vikraman Arvind, Abhranil Chatterjee, Rajit Datta, and Partha Mukhopadhyay. Fast Exact Algorithms Using Hadamard Product of Polynomials. *CoRR*, abs/1807.04496, 2018. `arXiv:1807.04496`.

4 Vikraman Arvind and Partha Mukhopadhyay. The ideal membership problem and polynomial identity testing. *Inf. Comput.*, 208(4):351–363, 2010. `doi:10.1016/j.ic.2009.06.003`.

5 Alexander I. Barvinok. Two Algorithmic Results for the Traveling Salesman Problem. *Math. Oper. Res.*, 21(1):65–84, February 1996. `doi:10.1287/moor.21.1.65`.

6 Andreas Björklund, Petteri Kaski, and Lukasz Kowalik. Constrained Multilinear Detection and Generalized Graph Motifs. *Algorithmica*, 74(2):947–967, 2016. `doi:10.1007/s00453-015-9981-1`.

7 Cornelius Brand, Holger Dell, and Thore Husfeldt. Extensor-coding. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2018, Los Angeles, CA, USA, June 25-29, 2018*, pages 151–164, 2018. `doi:10.1145/3188745.3188902`.

8 George E. Collins. Subresultants and Reduced Polynomial Remainder Sequences. *J. ACM*, 14(1):128–142, 1967. `doi:10.1145/321371.321381`.

9 David A. Cox, John Little, and Donal O'Shea. *Ideals, Varieties, and Algorithms: An Introduction to Computational Algebraic Geometry and Commutative Algebra, 3/e (Undergraduate Texts in Mathematics).* Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2007.

10 Marek Cygan, Fedor V. Fomin, Lukasz Kowalik, Daniel Lokshtanov, Dániel Marx, Marcin Pilipczuk, Michal Pilipczuk, and Saket Saurabh. *Parameterized Algorithms.* Springer, 2015. `doi:10.1007/978-3-319-21275-3`.

11 Richard A. Demillo and Richard J. Lipton. A probabilistic remark on algebraic program testing. *Information Processing Letters*, 7(4):193–195, 1978. `doi:10.1016/0020-0190(78)90067-4`.

12 Rodney G. Downey, Vladimir Estivill-Castro, Michael R. Fellows, Elena Prieto-Rodriguez, and Frances A. Rosamond. Cutting Up is Hard to Do: the Parameterized Complexity of k-Cut and Related Problems. *Electr. Notes Theor. Comput. Sci.*, 78:209–222, 2003. `doi:10.1016/S1571-0661(04)81014-4`.

13 Neeraj Kayal and Nitin Saxena. Polynomial Identity Testing for Depth 3 Circuits. *Computational Complexity*, 16(2):115–138, 2007. `doi:10.1007/s00037-007-0226-9`.

14 Pascal Koiran. Hilbert's Nullstellensatz Is in the Polynomial Hierarchy. *J. Complexity*, 12(4):273–286, 1996. `doi:10.1006/jcom.1996.0019`.

15 Andrevı Kotlov. Rank and Chromatic Number of a Graph. *J. Graph Theory*, 26(1):1–8, September 1997.

16 Ioannis Koutis. Faster Algebraic Algorithms for Path and Packing Problems. In *Automata, Languages and Programming, 35th International Colloquium, ICALP 2008, Reykjavik, Iceland, July 7-11, 2008, Proceedings, Part I: Tack A: Algorithms, Automata, Complexity, and Games*, pages 575–586, 2008. doi:10.1007/978-3-540-70575-8_47.

17 Ioannis Koutis. Constrained multilinear detection for faster functional motif discovery. *Inf. Process. Lett.*, 112(22):889–892, 2012. doi:10.1016/j.ipl.2012.08.008.

18 Ioannis Koutis and Ryan Williams. LIMITS and applications of group algebras for parameterized problems. *ACM Trans. Algorithms*, 12(3):31:1–31:18, 2016. doi:10.1145/2885499.

19 Hwangrae Lee. Power Sum Decompositions of Elementary Symmetric Polynomials. *Linear Algebra and its Applications*, 492:89 – 97, 2016. doi:10.1016/j.laa.2015.11.018.

20 K. Mahler. An inequality for the discriminant of a polynomial. *Michigan Math. J.*, 11(3):257–262, September 1964. doi:10.1307/mmj/1028999140.

21 E. Mayr and A. Meyer. The complexity of word problem for commutative semigroups and polynomial ideals. *Adv. Math*, 46:305–329, 1982.

22 Kevin Pratt. Faster Algorithms via Waring Decompositions. *CoRR*, abs/1807.06194, 2018. arXiv:1807.06194.

23 Sanjeev Saluja. A note on the permanent value problem. *Information Processing Letters*, 43(1):1–5, 1992. doi:10.1016/0020-0190(92)90021-M.

24 Nitin Saxena and C. Seshadhri. From sylvester-gallai configurations to rank bounds: Improved blackbox identity test for depth-3 circuits. *J. ACM*, 60(5):33:1–33:33, 2013. doi:10.1145/2528403.

25 Thomas J. Schaefer. The Complexity of Satisfiability Problems. In *Proceedings of the Tenth Annual ACM Symposium on Theory of Computing*, STOC '78, pages 216–226, New York, NY, USA, 1978. ACM.

26 Jacob T. Schwartz. Fast Probabilistic algorithm for verification of polynomial identities. *J. ACM.*, 27(4):701–717, 1980.

27 Madhu Sudan. Lectures on Algebra and Computation: Lecture Notes 6,12,13,14, 1998.

28 Ryan Williams. Finding paths of length k in $\mathrm{O}^{*}(2^{\mathrm{k}})$ time. *Inf. Process. Lett.*, 109(6):315–318, 2009. doi:10.1016/j.ipl.2008.11.004.

29 R. Zippel. Probabilistic algorithms for sparse polynomials. In *Proc. of the Int. Sym. on Symbolic and Algebraic Computation*, pages 216–226, 1979.