

Computing Asymptotic Bounds for Small Roots in Coppersmith's Method via Sumset Theory

Yansong Feng^{1,2}, Hengyi Luo^{1,2}, Qiyuan Chen^{1,2}, Abderrahmane Nitaj³, and Yanbin Pan^{1,2}

¹ State Key Laboratory of Mathematical Sciences, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, Beijing, China

² School of Mathematical Sciences, University of Chinese Academy of Sciences, Beijing, China
{fengyansong, luohengyi, chenqiyuan, panyanbin}@amss.ac.cn

³ Normandie Univ, UNICAEN, CNRS, LMNO, 14000 Caen, France
abderrahmane.nitaj@unicaen.fr

Abstract. Coppersmith's method is a well-known and practical method for solving polynomial modular equations involved in some cryptosystems such as RSA. An important and tedious task in this method consists in computing the asymptotic bounds. In this work, we address the challenge of computing such asymptotic bounds by introducing the Sumsets theory from Additive Combinatorics as a new analytical tool, which significantly streamlines manual calculations. More precisely, we develop the first provable algorithm for determining these asymptotic bounds, whereas the recent methods based on simple Lagrange interpolation are heuristic. Moreover, the experiments showed that our method is much more efficient than the previous method in practice. We also employ our method to improve the cryptanalytic results for the Commutative Isogeny Hidden Number Problem. Our approach may deepen the understanding of Coppersmith's method and inspire more security analysis methodologies.

Keywords: Additive combinatorics, Coppersmith's method, automated cryptanalysis, lattice-based cryptanalysis, asymptotic bounds

1 Introduction

In 1996, Coppersmith [Cop96] introduced methods for finding small roots of a single univariate modular polynomial, which were later extended to multivariate polynomials, integer polynomials, and modular polynomials by works such as [Cop97, HG01, May03, BM05, JM06, Cor04, Cor07]. These methods have since found wide applications in cryptanalysis [BHHG01, BV96, HR23, May02, MNS22, DMH20, TLP17]. The core paradigm of Coppersmith's method begins with constructing a set of polynomials sharing roots with the target polynomial. The coefficients of these polynomials form a lattice to be reduced. Crucially, the construction of polynomials determines all aspects of the method's performance - including root bounds and time complexity - making polynomial set construction the key optimization lever.

To find small integer roots of a single polynomial equation $f(x_1, \dots, x_k) \equiv 0 \pmod{M}$ in modular arithmetic, the Jochemsz-May Strategy [JM06] remains the state-of-the-art approach for general single polynomial equations. The method systematically constructs polynomial families G_m for a specific integer m , where each polynomial $g \in G_m$ satisfies $g(x_1, \dots, x_k) \equiv 0 \pmod{M^m}$. This generalizes Blömer-May's bivariate optimization [BM05] to multivariate cases. Recent work by Meers-Nowakowski [MN23] extended this to polynomial systems, with subsequent variants [Rya24, Sar24].

The Bound Determination Challenge. The bound of Coppersmith's method, $\mathbf{X} = (X_1, \dots, X_k)$, determines the maximal solution size $|x_i| < X_i$ that is recoverable via Coppersmith's method. The goal to improve Coppersmith's method is obviously to enlarge the bound, since a better bound leads to better attack in cryptanalysis. However, determining the bound usually has to be performed manually each time when a specific polynomial family G_m is considered. It is typically very time-consuming and tedious. Hence, fast automated computation of the bound allows us to efficiently determine in advance whether a specific polynomial family G_m can improve the previous results and makes the automated selection of better polynomial families in Coppersmith's method possible. Note that in Coppersmith's method, the bound arises from an inequality of the form

$$\det(\mathcal{L}) = X_1^{p_1(m)} \dots X_k^{p_k(m)} M^{p_{\mathcal{F}}(m)} < M^{m \dim(\mathcal{L})},$$

where \mathcal{L} is a lattice, $\det(\mathcal{L})$ is its determinant, $\dim(\mathcal{L})$ is its dimension, and $p_1(m), p_2(m), \dots, p_k(m)$, and $p_{\mathcal{F}}(m)$ are functions in m . To optimize the solution, we need to analyze the asymptotic bound as $m \rightarrow \infty$.

Flawed Automation Attempts. At Asiacrypt '23, Meers and Nowakowski [MN23] proposed a heuristic approach to compute the asymptotic bounds, where they assumed $p_{\mathcal{F}}$ is a degree- $(k+1)$ polynomial in m and employed Lagrange interpolation at $k+2$ points. Below, we expose critical flaws and limitations in their method:

- **A counterexample of heuristic in [MN23]:** For k -variable polynomials, the claim that $p_{\mathcal{F}}(m)$ is a polynomial in m of degree- $(k+1)$ fails for $f(x) = x^5 + x + 1$: the sequence $\{p_{\mathcal{F}}(m)\}_{m=1}^{\infty} = \{2, 8, 20, 40, 65, 95, \dots\}$ satisfies $\frac{5}{2}m^2 + \frac{5}{2}m - 10$ only when $m \geq 3$. We establish a corrected version of the heuristic: For any k -variable polynomial f , there exists $N \in \mathbb{N}$ such that for any $m > N$, $p_{\mathcal{F}}$ is a degree- $(k+1)$ polynomial in m . The proof can be found in Section 4. Moreover, their assertion that $m \dim(\mathcal{L})$ and the exponents p_1, \dots, p_k are degree- $(k+1)$ polynomials in m suffers from the same fundamental flaw.
- **Infeasible Computation:** While our corrected heuristic suggests Lagrange interpolation to recover $p_{\mathcal{F}}$ when $m > N$, state-of-the-art additive combinatorics results [GSW23] reveal fundamental limitations: For some 4-variable polynomials, the required threshold becomes $N > 2^{300}$. This implies we must compute the number of monomials in f^m for huge $m > 2^{300}$ to ensure correctness, which is computationally infeasible in practice. Furthermore, the local convergence phenomenon in Lagrange interpolation may yield incorrect results, which means that the output of their methods is unreliable. Details can be found in Appendix A.

As a consequence, the following natural question arises:

Can we compute $\dim(\mathcal{L})$ and $\det(\mathcal{L})$ more efficiently?

1.1 Our Contribution

We provide a positive answer to the above question and present a novel approach that avoids the tedious computation of asymptotic bounds in Coppersmith's method. This enables us to handle more complicated polynomial equations and achieve better results. We list our contributions as follows:

New Tools from Additive Combinatorics. We introduce fruitful results from additive combinatorics—including sumset theory and integer point counting in convex hull—into Coppersmith's method. These tools offer new perspectives for analyzing lattices in Coppersmith's method.

Beyond providing an efficient methodology for computing $\dim(\mathcal{L})$ and $\det(\mathcal{L})$, we re-examine prior heuristics through an additive combinatorial lens. While Meers *et al.* [MN23] and Sarkar [Sar24] assumed that $m \dim(\mathcal{L})$, p_1, \dots, p_k , and $p_{\mathcal{F}}$ are degree- $(k+1)$ polynomials in m (or jointly in m and t), our counterexample demonstrates that these heuristics require the critical precondition that m (or m and t jointly) must exceed a sufficient threshold. After establishing the necessary conditions, we formally prove that:

- Meers *et al.*'s heuristic holds for sufficiently large m .
- Sarkar's heuristic holds when both m and t are sufficiently large.

We believe these tools from additive combinatorics not only strengthen our theoretical understanding but also have the potential to inspire novel cryptanalytic techniques.

Formulas for Asymptotic Bounds in Coppersmith's Method. As $m \rightarrow \infty$, we derive elegant formulas for the bounds computation in Coppersmith's method. Taking the Jochemsz–May Basic Strategy as an example, we present the following formulas, for any arbitrarily small constant $\varepsilon > 0$:

$$\prod_{i=1}^k X_i^{\int_{N(f)} x_i dV} \cdot M^{\frac{k}{k+1} \int_{N(f)} 1 dV} < M^{\int_{N(f)} 1 dV - \varepsilon},$$

where $N(f)$ denotes the Newton polytope of f – the convex hull of exponent vectors $A(f) = \{(i_1, \dots, i_k) \mid x_1^{i_1} \cdots x_k^{i_k} \text{ is a monomial of } f\}$.

The Jochemsz–May Extended Strategy generalizes the Jochemsz–May Basic Strategy by introducing additional shift polynomials and guarantees results at least as strong as those of the latter. We also provide related formulas for the Jochemsz–May Extended Strategy. Recently, the Jochemsz–May Basic Strategy was generalized from a single polynomial equation to a system of polynomial equations [MN23, Sar24]. By adding additional shift polynomials, we further generalize these variants and derive the corresponding formulas. Since the approaches of Meers–Nowakowski [MN23] and Sarkar [Sar24] can be regarded as special cases of our generalization, our formulas naturally encompass their results.

Moreover, we point out that we also provide methods for computing the integrals $\int_{N(f)} x_i dV$ and $\int_{N(f)} 1 dV$ in the formulas or equivalently the bounds. Taking the formulas of the Jochemsz–May Basic strategy as an example, the computation of $\dim(\mathcal{L})$ is based on calculating the volume of the convex hull of the subset $A(f) \subset \mathbb{Z}^k$. For the integrals used in the computation of p_1, \dots, p_k and $p_{\mathcal{F}}$, we construct $(k+1)$ -dimensional subsets of \mathbb{Z}^{k+1} . The integral is equivalent to the volume of the convex hull of the subset, and the number of points in these subsets is bounded by at most twice that of $A(f)$. This approach makes our formulas more computer-friendly, as they can be computed directly from the input $A(f)$. For other generalized strategies, we apply a similar treatment. We have implemented our algorithms, and most of them completed within one second; our approach offers significant advantages over both manual computation and heuristic methods in [MN23, Sar24]. Further details can be found in Section 5.1.

We provide an efficient open source implementation of our algorithm for computing asymptotic bounds in SageMath. The source code is available at:

<https://github.com/fffmth/AsymptoticBounds>.

Our formulas enable *efficient* and *provable* computation of asymptotic root bounds X_i , advancing toward fully automated implementations of Coppersmith’s method.

Cryptanalytic Advances for CI-HNP. Building on the Commutative Isogeny Hidden Number Problem (CI-HNP) introduced at Asiacrypt ’23 [MN23], we achieve new *state-of-the-art* bounds for CI-HNP over CSURF (CSIDH on the surface [CD20]).

We compare our results with previous work in Table 1, where “Heuristic?” in this context means that a heuristic approach is adopted to compute the bound. We still need the Assumption 1 in Section 2 for Coppersmith’s method. Our framework enables precise security parameter selection for isogeny-based protocols by eliminating heuristic assumptions in bound computation and providing the first non-heuristic improvement over previous work.

Work	Required MSBs (%)	Heuristic?
[MN23]	75.61	Yes
[Rya24]	74.72 ⁴	Yes
[Sar24]	74.03	Yes
This work	74.01	No

Table 1: Comparison of CI-HNP Results (Required MSBs)

1.2 Technical Overview

Here we focus on the Jochemsz–May Basic Strategy to present our key ideas. Related formulas for the other cases are provided in Section 3 and 4.

⁴ An March, 2025 revision of Ryan’s work [Rya24] achieves 73.91% but still heuristic.

Starting Point: the Jochemsz-May Basic Strategy. Let $f \in \mathbb{Z}[x_1, \dots, x_k]$ be a monic polynomial with small root $\mathbf{u} = (u_1, \dots, u_k)$ satisfying $|u_i| < X_i$ for known bounds X_i , and modulus M . For a fixed integer m , the Jochemsz-May Basic Strategy defines the set:

$$J_\ell = \left\{ \lambda \mid \lambda \text{ is a monomial of } f^m \text{ and } \frac{\lambda}{\text{LM}(f)^\ell} \text{ is a monomial of } f^{m-\ell} \right\}, \quad (1)$$

for $0 \leq \ell \leq m$ and $J_{m+1} = \emptyset$, where $\text{LM}(f)$ denotes f 's leading monomial. The strategy constructs basis polynomials for every monomial $\lambda \in J_\ell \setminus J_{\ell+1}$:

$$g_\lambda(x_1, \dots, x_k) = \frac{\lambda}{\text{LM}(f)^\ell} f^\ell M^{m-\ell}, \quad \lambda \in J_\ell \setminus J_{\ell+1}. \quad (2)$$

All $g_\lambda(x_1, \dots, x_k)$'s form the set G_m . The lattice \mathcal{L} is constructed by taking the coefficient vectors of the polynomials $g_\lambda(x_1 X_1, \dots, x_k X_k)$ as a basis. Then the lattice \mathcal{L} constructed via Equation (1) and Equation (2) satisfies:

$$\dim(\mathcal{L}) = \left| \bigcup_{0 \leq \ell \leq m} J_\ell \right|, \quad (3)$$

$$\det(\mathcal{L}) = \prod_{g_\lambda(x_1, \dots, x_k) \in G_m} \text{LC}(g_\lambda(x_1 X_1, \dots, x_k X_k)) = \prod_{j=1}^k X_j^{p_j} M^{p_j}, \quad (4)$$

where $|\cdot|$ denotes the cardinality of a set and $\text{LC}(\cdot)$ denotes leading coefficient.

Example 1 (Univariate Case). For $m = 2$, $k = 1$, and $f(x_1) = x_1 + 1$, the lattice basis matrix constructed by the Jochemsz-May Basic Strategy is:

$$\begin{pmatrix} M^2 & 0 & 0 \\ M & MX_1 & 0 \\ 1 & 2X_1 & X_1^2 \end{pmatrix},$$

and we know that $\dim(\mathcal{L}) = 3$ and $\det(\mathcal{L}) = X_1^3 M^3$.

However, in the Jochemsz-May Basic Strategy paper [JM06], they assume that the monomials of f, f^1, \dots, f^{m-1} are all contained in the monomials of f^m . If this doesn't hold, the definition of J_ℓ in Equation (1) can be slightly changed such that J_ℓ contains all monomial λ of f^j for $j = 1, \dots, m$ for which $\lambda/\text{LM}(f)^\ell$ is a monomial of f^i for some $i = 0, \dots, m$.

To ensure the above assumption holds and to align with the sumset theory, we also modify Equation (1). Recall that for two sets A and B , their Minkowski sum is defined as $A + B = \{a + b \mid a \in A, b \in B\}$, and mA denotes the Minkowski sum of m copies of A .

For the leading monomial $x_1^{\alpha^{(1)}} \cdots x_k^{\alpha^{(k)}}$ of f , let $\alpha = (\alpha^{(1)}, \dots, \alpha^{(k)}) \in \mathbb{Z}^k$ and define $A(f) = \{(i_1, \dots, i_k) \mid x_1^{i_1} \cdots x_k^{i_k} \text{ is a monomial of } f\}$. We replace J_ℓ in Equation (1) as follows:

$$\begin{aligned} J_\ell &= \{x_1^{i_1} \cdots x_k^{i_k} \mid (i_1, \dots, i_k) \in mA(f) \text{ and } (i_1, \dots, i_k) \in (m - \ell)A(f) + \ell\alpha\} \\ &= \{x_1^{i_1} \cdots x_k^{i_k} \mid (i_1, \dots, i_k) \in (m - \ell)A(f) + \ell\alpha\}, \end{aligned}$$

ensuring that the monomials of f, f^1, \dots, f^{m-1} are all contained among the monomials of f^m .

Our formulas. In this section, we require $f \in \mathbb{Z}[x_1, \dots, x_k]$ to be a *well-defined k -variate* polynomial, that is, it satisfies the following conditions:

1. $f(0, \dots, 0) \neq 0$.
2. The lattice generated by $A(f)$ has finite index as a sublattice of \mathbb{Z}^k .

For simplicity, in this section, we assume that the index of the lattice generated by $A(f)$ is 1. For the general case, see Section 3.

For any *well-defined k -variate* polynomial f , our formulas show that the inequality $\det(L) < M^{m \dim(L)}$ can be asymptotically written as, for any arbitrarily small constant $\varepsilon > 0$:

$$\prod_{i=1}^k X_i^{\int_{N(f)} x_i dV} \cdot M^{\frac{k}{k+1} \int_{N(f)} 1 dV} < M^{\int_{N(f)} 1 dV - \varepsilon},$$

where $N(f)$ is the Newton polytope of f , that is the convex hull of $A(f)$. Below we will show how to prove these formulas.

Computing Dimension via Additive Combinatorics. We first need to rewrite $\dim(\mathcal{L})$ to match the sumset theory. As $J_\ell = \{x_1^{i_1} \cdots x_k^{i_k} \mid (i_1, \dots, i_k) \in (m - \ell)A(f) + \ell\alpha\}$, we have

$$J_{m+1} \subset \cdots \subset J_0,$$

which allows us to rewrite Equation (3) as follows:

$$\dim(\mathcal{L}) = \left| \bigcup_{0 \leq \ell \leq m} J_\ell \right| = |J_0| = |mA(f)|. \quad (5)$$

The equivalence between Equation (3) and Equation (5) is powerful and makes the computation of $\dim(\mathcal{L})$ match well with additive combinations.

Taking $G = \mathbb{Z}^k$ and $B = \emptyset$ in Theorem 1 and Corollary 1 in [Kho92], we have the following result, which is also a special case of Lemma 18 with $\text{ind } A = 1$:

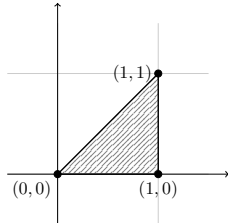
Lemma 2. *For any finite subset $A \subset \mathbb{Z}^k$ satisfying $\mathbb{Z}^k(A)$ (generated by $\{\sum n_i a_i \mid a_i \in A, n_i \in \mathbb{Z}, \sum n_i = 0\}$) is \mathbb{Z}^k , the cardinality of mA , i.e., $|mA|$, becomes polynomial in m with degree k for sufficiently large m , with leading coefficient equal to the volume of the convex hull of A .*

Direct application of this lemma completes the computation of $\dim(\mathcal{L})$.

Example 3. Taking the modular polynomial equation $f(x_1, x_2) = a_1 x_1 x_2 + a_2 x_1 + a_3 \equiv 0 \pmod{M}$ as an example. By manual calculation, the Jochemsz-May Basic Strategy yields a lattice \mathcal{L} and the corresponding $\dim(\mathcal{L})$ is

$$|\{\lambda \mid \lambda \text{ is a monomial of } f^m\}| = \sum_{i_1=0}^m \sum_{i_2=0}^{m-i_1} 1 = \frac{1}{2}m^2 + \frac{3}{2}m + 1 = \frac{1}{2}m^2 + o(m^2).$$

Using our formulas, we can compute the leading coefficient of $\dim(\mathcal{L})(m)$ by calculating the volume of the triangle with vertices $\{(0,0), (1,0), (1,1)\}$, which has an area of $1/2$. See Fig. 1.



$f(x_1, x_2) = a_1 x_1 x_2 + a_2 x_1 + a_3$,
 Δ is a triangle of $\{(0,0), (1,0), (1,1)\}$,
 The volume of Δ is $\frac{1}{2}$.

Fig. 1: Newton polytope corresponding to $f(x_1, x_2) = a_1 x_1 x_2 + a_2 x_1 + a_3$.

Computing Determinant via Additive Combinatorics. The determinant of the lattice \mathcal{L} satisfies:

$$\det(\mathcal{L}) = X_1^{p_1} \cdots X_k^{p_k} \cdot M^{p_{\mathcal{F}}}.$$

Thus, computing $\det(\mathcal{L})$ reduces to determining p_j for $j = 1, \dots, k$ and $p_{\mathcal{F}}$, where a key challenge lies in expressing $p_{\mathcal{F}}$ through sumsets.

Computation of p_j : From Equation (2), each monomial λ induces a polynomial $g_\lambda(x_1, \dots, x_k)$ whose leading monomial is indeed λ . Hence, the exponent of X_j in $\text{LC}(g_\lambda(x_1 X_1, \dots, x_k X_k))$ is equal to the exponent of x_j in monomial λ .

This enables us to formally express $p_j(m)$ using $A(f)$:

$$p_j(m) = \sum_{(i_1, \dots, i_k) \in mA(f)} i_j. \quad (6)$$

Considering the set $\mathcal{H}_j(A)$:

$$\mathcal{H}_j(A) = \bigcup_{(i_1, \dots, i_k) \in A} \{(i_1, \dots, i_k, 0), (i_1, \dots, i_k, 1), \dots, (i_1, \dots, i_k, i_j)\},$$

it enables the following identity:

$$\sum_{(i_1, \dots, i_k) \in mA(f)} i_j = |\mathcal{H}_j(mA(f))| - |mA(f)| = |m\mathcal{H}_j(A(f))| - |mA(f)|.$$

By Lemma 2, p_j is a polynomial with degree $k+1$ in m when m is large enough. Similar to above, the leading coefficient of p_j is exactly the volume of the convex hull of $\mathcal{H}_j(A(f))$, which also equals the integral of x_j over $N(f)$ (see Lemma 23).

Computation of $p_{\mathcal{F}}$: The central challenge is how to express $p_{\mathcal{F}}$ through sumsets. Applying Abel's summation formula yields:

$$p_{\mathcal{F}}(m) = \sum_{\ell=0}^m (m-\ell) |J_{\ell} \setminus J_{\ell+1}| = \sum_{\ell=0}^m (m-\ell) (|J_{\ell}| - |J_{\ell+1}|) = (m+1) |J_0| - \sum_{\ell=0}^m |J_{\ell}|.$$

To compute $\sum_{\ell=0}^m |J_{\ell}|$, we construct $(k+1)$ -dimensional polytopes:

$$\tilde{A}(f) = \{(i_1, \dots, i_k, 1) \mid (i_1, \dots, i_k) \in A(f)\} \cup \{(\alpha^{(1)}, \dots, \alpha^{(k)}, 0)\},$$

where $x_1^{\alpha^{(1)}} \cdots x_k^{\alpha^{(k)}}$ denotes f 's leading monomial. This gives:

$$\sum_{\ell=0}^m |J_{\ell}| = |m\tilde{A}(f)|,$$

enabling the reformulation:

$$p_{\mathcal{F}}(m) = (m+1) |mA(f)| - |m\tilde{A}(f)|.$$

Both $|mA(f)|$ and $|m\tilde{A}(f)|$ become polynomials in m asymptotically. Crucially, $\tilde{A}(f)$ forms a conical extension of $A(f)$ with volume scaling factor $\frac{1}{k+1}$, establishing the leading coefficient's fractional relationship.

Roadmap. Section 2 establishes foundational concepts including polynomial notation, lattice definitions, sumset theory, and Coppersmith's method. In Section 3 we present the formulas for a single polynomial equation with Jochensz-May Extended Strategy. In Section 4, we systematically generalize the Jochensz-May Extended Strategy from a single polynomial equation to a system of polynomial equations while deriving related formulas. Section 5 presents how to compute the asymptotic upper bounds and our improvements to the CI-HNP over CSURF. Moreover, in order to fully demonstrate the efficiency of computing asymptotic upper bounds, we have conducted sufficient experiments, which also can be found in Section 5. Finally, Section 6 states our conclusions.

2 Preliminaries

We use \mathbb{N} , \mathbb{Z} , \mathbb{Q} and \mathbb{R} to denote the set of all non-negative integers, the ring of integers, the field of rational numbers and the field of real numbers, respectively. We use little-o notation $o(\cdot)$ or big-O notation $O(\cdot)$ to represent the loose or tight upper bound for a function. For the vector $\mathbf{u} = (u_1, \dots, u_k)$, we use $\|\cdot\|$ to represent the Euclidean norm, defined as $\|\mathbf{u}\| = \left(\sum_{i=1}^k u_i^2\right)^{1/2}$. We also use $\|\cdot\|_{\infty}$ to denote the infinity norm. We denote by $|\cdot|$ the cardinality of a set.

2.1 Polynomials

For a k -variable polynomial $f \in \mathbb{Z}[x_1, \dots, x_k]$, it can be written as

$$f(x_1, \dots, x_k) = \sum_{i_1, \dots, i_k \in \mathbb{N}} \alpha_{i_1, \dots, i_k} \cdot x_1^{i_1} \cdot \dots \cdot x_k^{i_k},$$

where $\alpha_{i_1, \dots, i_k} \in \mathbb{Z}$.

A term $x_1^{i_1} \cdot \dots \cdot x_k^{i_k}$ is called a *monomial* of f if its coefficient $\alpha_{i_1, \dots, i_k} \neq 0$. The set of all monomials in f is denoted by $\text{supp}(f)$. The *total degree* $\deg(f)$ of f is defined as

$$\deg(f) = \max_{\alpha_{i_1, \dots, i_k} \neq 0} (i_1 + \dots + i_k).$$

Definition 4 (Monomial Order). Let \mathcal{M} be a set of monomials. A monomial order on \mathcal{M} is a total order \prec that satisfies the following properties:

1. For every $\lambda \in \mathcal{M}$, it holds that $1 \prec \lambda$.
2. If $\lambda_1 \prec \lambda_2$, then $\lambda \cdot \lambda_1 \prec \lambda \cdot \lambda_2$ for every monomial $\lambda \in \mathcal{M}$.

Using the lexicographic monomial order (\prec_{lex}) with $x_1 \prec_{\text{lex}} x_2 \prec_{\text{lex}} x_3$ as an example, this order first compares the exponents of x_3 in the monomials; if those are equal, it then compares the exponents of x_2 , and so on. Then we have $x_2^2 \prec_{\text{lex}} x_3$ and $x_1 \prec_{\text{lex}} x_2 \prec_{\text{lex}} x_2^2$.

If \prec is a monomial order, the *leading monomial* of a polynomial f is the unique monomial λ in f that satisfies $\lambda' \prec \lambda$ for every other monomial λ' in f . The *leading coefficient* is the coefficient of the leading monomial. We denote the leading monomial and the leading coefficient of f by $\text{LM}(f)$ and $\text{LC}(f)$, respectively. The *leading term* of f is given by

$$\text{LT}(f) = \text{LC}(f) \times \text{LM}(f).$$

A polynomial is *monic* if $\text{LC}(f) = 1$.

2.2 Lattices

A *lattice* is a discrete additive subgroup of \mathbb{R}^m where $m \in \mathbb{N}$. Another equivalent definition is presented as follows.

Definition 5 (Lattice). Let $\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n \in \mathbb{R}^m$ be n linearly independent vectors with $n \leq m$. The lattice \mathcal{L} spanned by $\{\mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n\}$ is the set of all integer linear combinations of these vectors, i.e.,

$$\mathcal{L} = \left\{ \mathbf{v} \in \mathbb{R}^m \mid \mathbf{v} = \sum_{i=1}^n a_i \mathbf{v}_i, a_i \in \mathbb{Z} \right\}.$$

The integer n is called the *rank* of the lattice \mathcal{L} , while m is referred to its *dimension*. The lattice \mathcal{L} is said to be *full-rank* if $n = m$.

A lattice can be represented using a basis matrix $\mathbf{B} \in \mathbb{R}^{n \times m}$, where each row corresponds to a basis vector \mathbf{v}_i . The *determinant* of \mathcal{L} is defined as

$$\det(\mathcal{L}) = \sqrt{\det(\mathbf{B}\mathbf{B}^t)},$$

where \mathbf{B}^t denotes the transpose of \mathbf{B} . If \mathcal{L} is full-rank, this simplifies to $\det(\mathcal{L}) = |\det(\mathbf{B})|$. In this paper, we consider full-rank lattices.

Definition 6 (Fundamental Parallelepiped). For any full-rank n -dimensional lattice \mathcal{L} with basis matrix \mathbf{B} , we define

$$\mathcal{P}(\mathbf{B}) = \{\mathbf{B}^t x \mid x \in \mathbb{R}^n, \forall i : 0 \leq x_i < 1\},$$

called the *fundamental parallelepiped* of \mathcal{L} . The *index* of \mathcal{L} as a sublattice of \mathbb{Z}^n is defined as the volume of $\mathcal{P}(\mathbf{B})$.

Definition 7 (Index of Sublattice of \mathbb{Z}^n). For any full-rank n -dimensional lattice $\mathcal{L} \subseteq \mathbb{Z}^n$ with basis matrix \mathbf{B} , the *index* of \mathcal{L} as a sublattice of \mathbb{Z}^n is defined as the volume of $\mathcal{P}(\mathbf{B})$, or equivalently, the determinant of \mathcal{L} .

Definition 8 (Shortest Vector Problem (SVP)). Given a lattice \mathcal{L} , the Shortest Vector Problem (SVP) asks to find a non-zero lattice vector $\mathbf{v} \in \mathcal{L}$ of minimum Euclidean norm, i.e., find $\mathbf{v} \in \mathcal{L} \setminus \{\mathbf{0}\}$ such that $\|\mathbf{v}\| \leq \|\mathbf{w}\|$ for all non-zero $\mathbf{w} \in \mathcal{L}$.

The Shortest Vector Problem (SVP) has been proven to be NP-hard under randomized reductions [Ajt98]. However, using the well-known LLL algorithm [LLL82], it is possible to efficiently compute a relatively short vector in polynomial time.

Lemma 9 (LLL Basis Reduction). Let $\mathcal{L} \subseteq \mathbb{Z}^n$ be a lattice. The LLL algorithm finds, in polynomial time, a basis $(\mathbf{v}_1, \dots, \mathbf{v}_n)$ of \mathcal{L} satisfying

$$\|\mathbf{v}_i\| \leq 2^{\frac{n(n-1)}{4(n+1-i)}} \det(\mathcal{L})^{\frac{1}{n+1-i}},$$

for all $i = 1, \dots, n$.

2.3 Sumset Theory for Polynomials

This section presents some results from additive combinatorics (sumset theory) and demonstrates their application to polynomials in $\mathbb{Z}[x_1, \dots, x_k]$. We also introduce some results concerning the problem of counting the number of lattice points contained within a convex bounded polyhedron, commonly referred to as a polytope.

Newton Polytope. For any polynomial $f \in \mathbb{Z}[x_1, \dots, x_k]$, define its monomial exponent set as:

$$A(f) = \{(i_1, \dots, i_k) \mid x_1^{i_1} \cdots x_k^{i_k} \text{ is a monomial of } f\}.$$

Definition 10 (Convex Hull). The convex hull of a set $S \subset \mathbb{R}^n$ is denoted as $N(S)$ and is defined as the smallest convex set containing S . Formally, it is given by:

$$N(S) = \left\{ \sum_{i=1}^m \lambda_i x_i \mid x_i \in S, \lambda_i \geq 0, \sum_{i=1}^m \lambda_i = 1 \right\}$$

where the sum is over some finite set of points $\{x_1, x_2, \dots, x_m\}$ from S , and λ_i are the convex coefficients.

Definition 11 (Newton Polytope). The Newton polytope of $f \in \mathbb{Z}[x_1, \dots, x_k]$, denoted by $N(f)$, is the convex hull of $A(f)$ in \mathbb{R}^k .

Example 12. For f with $\text{supp}(f) = \{1, x_1, x_1^2, x_2\}$, we have:

- $A(f) = \{(0, 0), (1, 0), (2, 0), (0, 1)\}$
- Newton polytope: Triangle with vertices $(0, 0)$, $(2, 0)$, and $(0, 1)$

Definition 13 (Minkowski Sum). For sets $A, B \subset \mathbb{Z}^k$, their Minkowski sum is:

$$A + B = \{a + b \mid a \in A, b \in B\}.$$

For $m \in \mathbb{N}$, denote by mA the sum of m identical sets A .

Property 14. For all nonzero $f_1, f_2 \in \mathbb{Z}[x_1, \dots, x_k]$:

$$N(f_1 \cdot f_2) = N(f_1) + N(f_2).$$

Well-Defined k -Variate Polynomials. To avoid cases like $f_1 = x_1 x_2 + 1$ reducible to $f_2 = z_1 + 1$ via substitution $z_1 = x_1 x_2$, where f_1^m exhibits univariate-like monomial growth despite its bivariate form, we need some requirements on f .

Definition 15 (Well-Defined k -Variate Polynomials). A polynomial $f \in \mathbb{Z}[x_1, \dots, x_k]$ is called well-defined k -variate if:

1. $f(0, \dots, 0) \neq 0$.
2. The lattice generated by $A(f)$ has finite index in \mathbb{Z}^k , denoted by $\text{ind } A(f)$.

We consider two types of summations. The first is the number of elements in $mA(f)$: $\sum_{\lambda \in mA(f)} 1$. The second is a weighted version: for homogeneous polynomial ϕ , $\sum_{\lambda \in mA(f)} \phi(\lambda)$. We also consider cases with multiple subsets like $m_1 A(f_1) + m_2 A(f_2)$.

Summation of $\sum_{\lambda \in mA(f)} 1$. For any finite subset $A \subset \mathbb{Z}^k$, Khovanskii [Kho92] first established polynomial growth of $\sum_{\lambda \in mA} 1$. We adapt Theorem 1 ($G = \mathbb{Z}^k$, $B = \emptyset$) and Corollary 2 from [Kho92]:

Lemma 16 ([Kho92], Theorem 1). *Let A be any finite subset of \mathbb{Z}^k . For all sufficiently large $m \in \mathbb{N}$, $|mA|$ is a polynomial in m with degree $< |A|$.*

Lemma 17 ([Kho92], Corollary 2). *For finite $A \subset \mathbb{Z}^k$ where $\mathbb{Z}^k(A)$ (generated by $\{\sum n_i a_i \mid a_i \in A, n_i \in \mathbb{Z}, \sum n_i = 0\}$) has finite index $\text{ind } A$ in \mathbb{Z}^k :*

$$\lim_{m \rightarrow \infty} \frac{|mA|}{m^k} = (\text{ind } A)^{-1} V(A),$$

where $V(A)$ denotes the volume of the convex hull of A .

Using Lemma 16 and Lemma 17, we have the following result:

Lemma 18 (Summation of $\sum_{\lambda \in mA(f)} 1$). *For any well-defined k -variate f , the number of points in $mA(f)$ is polynomial in m with degree k when m is large enough. Specifically, the leading coefficient is the volume of the Newton polytope corresponding to f divided by $\text{ind } A(f)$. That is,*

$$|mA(f)| = \frac{\int_{N(f)} 1 dV}{\text{ind } A(f)} m^k + o(m^k).$$

Proof. From Definition 15, $\mathbf{0} \in A(f)$ ensures, for any v in the lattice generated by $A(f)$:

$$\exists n_i \in \mathbb{Z}, a_i \in A(f) : v = \sum n_i a_i + (-\sum n_i) \mathbf{0},$$

with $\sum n_i + (-\sum n_i) = 0$. Thus the lattice generated by $A(f)$ is $\mathbb{Z}^k(A(f))$ with index $\text{ind } A(f)$. Lemma 17 then applies directly. \square

Granville et al. [GSW23] bounded the polynomialization threshold:

Lemma 19 ([GSW23], Theorem 1.1). *For finite $A \subset \mathbb{Z}^k$, $\exists p \in \mathbb{Q}[x]$ with $\deg p \leq k$ such that:*

$$|mA| = p(m) \quad \forall m \geq (2|A| \cdot w(A))^{(k+4)|A|}$$

where $w(A) = \max_{a,b \in A} \|a - b\|_\infty$.

For $f = x_1^3 + x_1 x_2 + x_1 x_3 + x_2 + x_3^2 x_4^2 + x_4^5 + 1$, this requires:

$$m > (2 \cdot 7 \cdot 5)^{8.7} \approx 2^{343}.$$

Lemma 16 was generalized to the multiple subsets setting by [HKN98, Nat00, NR02].

Lemma 20 ([NR02], Theorem 2). *For finite sets A_1, \dots, A_r in abelian semigroup S , there exists a polynomial $p(t_1, \dots, t_r)$, such that*

$$|h_1 A_1 + \dots + h_r A_r| = p(h_1, \dots, h_r)$$

for all sufficiently large integers h_1, \dots, h_r .

Ehrhart's theorem [Ehr62] establishes that for any integral polytope $N \subset \mathbb{R}^k$, the lattice point count

$$|mN \cap \mathbb{Z}^k|$$

is a polynomial in m . The leading coefficient of this polynomial is equal to the volume of N , as stated in Theorem 1 of [DL05].

Lemma 21. *Suppose $N \in \mathbb{R}^k$ is an integral polytope whose vertices have integral coordinates, then $|mN \cap \mathbb{Z}^k|$ is a polynomial in m with degree k and its leading coefficient is the volume of N .*

It is straightforward to derive the following result:

Lemma 22 (Summation of $\sum_{\lambda \in mN(f) \cap \mathbb{Z}^k} 1$). *For any well-defined k -variate f , the number of points in $mN(f)$ is polynomial in m with degree k . Specifically, the leading coefficient is the volume of the Newton polytope corresponding to f . That is,*

$$|mN(f) \cap \mathbb{Z}^k| = \int_{N(f)} 1 dV m^k + o(m^k).$$

Proof. Instantiate Lemma 21 with $N = N(f)$. \square

Summation of $\sum_{\lambda \in mA(f)} \phi(\lambda)$. Initial work by [BV97] analyzed

$$\sum_{\lambda \in mN \cap \mathbb{Z}^k} \phi(\lambda)$$

for integral polytopes N and homogeneous polynomials ϕ . Recently, further progress has been reported in this area [BDDL⁺24]. It is shown that this summation forms a polynomial in m of degree $k + \deg(\phi)$, with the leading coefficient given by $\int_{N(f)} \phi dV$. These integrals have been studied in [Bar92, Bar94, BBDL⁺11]. We specialize to $\phi = x_j$, the homogeneous coordinate function:

Lemma 23 (Summation of $\sum_{\lambda \in mA(f)} x_j(\lambda)$). *For any well-defined k -variate polynomial f , the sum $\sum_{(i_1, \dots, i_k) \in mA(f)} i_j$ is polynomial in m with degree $k + 1$ when m is large enough. Specifically, the leading coefficient is $\int_{N(f)} x_j dV$ divided by $\text{ind } A(f)$. That is,*

$$\sum_{(i_1, \dots, i_k) \in mA(f)} i_j = \frac{\int_{N(f)} x_j dV}{\text{ind } A(f)} m^{k+1} + o(m^{k+1}),$$

where $x_j : \mathbb{R}^k \rightarrow \mathbb{R}$ acts as $x_j((i_1, \dots, i_k)) = i_j$.

Next, we consider the setting with shift $E = \{(0, \dots, 0), (1, 0, \dots, 0)\} \subset \mathbb{Z}^k$:

Lemma 24 (Summation of $\sum_{\lambda \in m_1 A(f) + m_2 E} x_j(\lambda)$). *For any well-defined k -variate polynomials f , the summation*

$$\sum_{(i_1, \dots, i_k) \in m_1 A(f) + m_2 E} i_j$$

is a polynomial in m_1 and m_2 of degree $k + 1$ when m_1 and m_2 are sufficiently large. That is,

$$\sum_{\lambda \in m_1 A(f) + m_2 E} x_j(\lambda) = \underbrace{b_0^{(j)} m_1^{k+1} + b_1^{(j)} m_1^k m_2 + \dots + b_{k+1}^{(j)} m_2^{k+1}}_{\text{all monomials with degree } k+1} + \dots$$

Besides, for all $\delta \in \mathbb{N}$, $b_i^{(j)}$ satisfy

$$\sum_{i=0}^{k+1} b_i^{(j)} \delta^i = \frac{\int_{N(A(f) + \delta E)} x_j dV}{\text{ind } A(f) + \delta E}$$

where $x_j : \mathbb{R}^k \rightarrow \mathbb{R}$ acts as $x_j((i_1, \dots, i_k)) = i_j$.

Detailed proofs for both Lemma 23 and 24 can be found in Appendix C.

Remark 25. Using an analogous proof, this result can be extended to multiple subsets.

2.4 Coppersmith's Method

Suppose $f \in \mathbb{Z}[x_1, \dots, x_k]$ has a small root $\mathbf{u} = (u_1, \dots, u_k) \in \mathbb{Z}^k$ modulo some integer M . Here, a small root means that $|u_i| < X_i$ for known bounds X_i , for $i = 1, \dots, k$.

In Coppersmith's method [Cop96, Cop97, May03], one initially fixes an integer m and then constructs a set of polynomials that share a common root with the target polynomial modulo M^m . Subsequently, the coefficient vectors of these polynomials are used to construct a lattice \mathcal{L} serving as the lattice basis. Then, one searches for short vectors v_j ($j = 1, \dots, k$) in the lattice, such that the corresponding polynomials $h_j(x_1, \dots, x_k)$ satisfies the conditions of Lemma 26 [HG97].

Lemma 26. *Let $h(x_1, \dots, x_k) \in \mathbb{Z}[x_1, \dots, x_k]$ be a polynomial with at most ω monomials, and let M be a positive integer. If there exist integers (u_1, \dots, u_k) satisfying the following two conditions:*

1. $h(u_1, \dots, u_k) \equiv 0 \pmod{M}$,
2. there exist positive integers X_1, \dots, X_k such that $|u_i| < X_i$ for $i = 1, \dots, k$, and

$$\|h(x_1 X_1, \dots, x_k X_k)\| < \frac{M}{\sqrt{\omega}},$$

then $h(u_1, \dots, u_k) = 0$ holds over \mathbb{Z} .

Hence, for each h_j , both

$$h_j(\mathbf{u}) \equiv 0 \pmod{M^m} \quad \text{and} \quad h_j(\mathbf{u}) = 0 \quad \text{over } \mathbb{Z}$$

hold. Then the roots can be found by solving the equations $h_j(\mathbf{u}) = 0$.

Note that we aim to find short vectors that satisfy condition 2 of Lemma 26 in polynomial time. Using the LLL algorithm, one can efficiently approximate relatively short vectors. Combining Lemma 26 and Lemma 9, we require the lattice \mathcal{L} to satisfy

$$\det(\mathcal{L}) < M^{m \cdot \dim(\mathcal{L})}. \quad (7)$$

Since both $\det(\mathcal{L})$ and $\dim(\mathcal{L})$ depend on m, X_1, \dots, X_k , Equation (7) can be used to bound X_j for $j = 1, \dots, k$.

After constructing the lattice \mathcal{L} and applying the LLL algorithm, Coppersmith's method further requires the following assumption commonly required for the multivariate case [BD00, HM10, FNP24], and this heuristic holds for most instances encountered in practice:

Assumption 1. *The polynomials obtained from the LLL-reduced basis in Coppersmith's method generate an ideal corresponding to a zero-dimensional variety.*

3 Solving a Single Polynomial Equation

In this section, we review the Jochemsz-May Extended Strategy and present its corresponding formulas for the asymptotic bounds. Since the Jochemsz-May Basic Strategy can be seen as a special case of the Jochemsz-May Extended Strategy, all our results for the extended strategy works for the basic strategy.

3.1 The Jochemsz-May Extended Strategy

Let $f \in \mathbb{Z}[x_1, \dots, x_k]$ be monic with small root $\mathbf{u} = (u_1, \dots, u_k)$ satisfying $|u_i| < X_i$ for bounds X_i and the modulus is M . For integers m and t , the Jochemsz-May Extended Strategy introduces additional shifts in x_1 compared to the Jochemsz-May Basic Strategy. For the leading monomial $x_1^{\alpha^{(1)}} \cdots x_k^{\alpha^{(k)}}$ of f , let $\alpha = (\alpha^{(1)}, \dots, \alpha^{(k)}) \in \mathbb{Z}^k$ and define $A(f) = \{(i_1, \dots, i_k) \mid x_1^{i_1} \cdots x_k^{i_k} \text{ is a monomial of } f\}$. The only difference lies in the definition of J_ℓ , which is defined as:

$$J_\ell = \bigcup_{0 \leq j \leq t} \left\{ x_1^j \cdot \lambda \mid \lambda \in (m - \ell)A(f) + \ell\alpha \right\}. \quad (8)$$

Now all $g_\lambda(x_1, \dots, x_k)$'s form the set $G_{m,t}$. The lattice \mathcal{L} constructed via the Jochemsz-May Extended Strategy satisfies:

$$\dim(\mathcal{L}) = \left| \bigcup_{0 \leq \ell \leq m} J_\ell \right|, \quad (9)$$

$$\det(\mathcal{L}) = \prod_{g_\lambda(x_1, \dots, x_k) \in G_{m,t}} \text{LC}(g_\lambda(x_1 X_1, \dots, x_k X_k)). \quad (10)$$

Remark 27. Typically, t is chosen to be a constant multiple of m . Thus, in our asymptotic analysis, for convenience, we will abbreviate $\sum_{i+j < k} m^i t^j$ as $o(m^k)$. When t is set to 0, the approach reduces to the Jochemsz-May Basic Strategy.

Remark 28. To ensure that the assumption in the Jochemsz-May Extended Strategy—that the monomials of f, f^1, \dots, f^{m-1} are all contained among the monomials of f^m —holds, we modify the original definition of J_ℓ . Specifically, we replace

$$J_\ell = \bigcup_{0 \leq j \leq t} \left\{ x_1^j \cdot \lambda \mid \lambda \in \text{supp}(f^m) \text{ and } \frac{\lambda}{\text{LM}(f)^\ell} \in \text{supp}(f^{m-\ell}) \right\}$$

with

$$J_\ell = \bigcup_{0 \leq j \leq t} \left\{ x_1^j \cdot \lambda \mid \lambda \in (m - \ell)A(f) + \ell\alpha \right\}.$$

3.2 Our formulas.

For the leading monomial $x_1^{\alpha^{(1)}} \dots x_k^{\alpha^{(k)}}$ of f , denote $\alpha = (\alpha^{(1)}, \dots, \alpha^{(k)}) \in \mathbb{Z}^k$. Define $A(f)$ and $\tilde{A}(f)$ as follows

$$\begin{aligned} A(f) &= \{(i_1, \dots, i_k) \mid x_1^{i_1} \dots x_k^{i_k} \in \text{supp}(f)\}, \\ \tilde{A}(f) &= \{(i_1, \dots, i_k, 1) \mid (i_1, \dots, i_k) \in A(f)\} \cup \{(\alpha^{(1)}, \dots, \alpha^{(k)}, 0)\}, \end{aligned}$$

We also denote E and \tilde{E} as follows:

$$\begin{aligned} E &= \{(0, \dots, 0), \underbrace{(1, 0, \dots, 0)}_{\text{Only the 1-th position is 1}}\} \subset \mathbb{Z}^k, \\ \tilde{E} &= \{(0, \dots, 0, 0), \underbrace{(1, 0, \dots, 0, 0)}_{\text{Only the 1-th position is 1}}\} \subset \mathbb{Z}^{k+1}. \end{aligned}$$

Now we focus on the following inequality which derives the bound directly:

$$\det(\mathcal{L}) < M^{m \dim(\mathcal{L})}. \quad (11)$$

Computing $\dim(\mathcal{L})$ via Additive Combinatorics. For dimension, we obtain the following theorem:

Theorem 29. *For any well-defined k -variate f , the $\dim(\mathcal{L})$ obtained by the Jochemsz-May Extended Strategy is polynomial in t and m when t and m are large enough. More precisely,*

$$\dim(\mathcal{L})(m, t) = a_0 m^k + a_1 t m^{k-1} + \dots + a_k t^k + o(m^k),$$

when $t = O(m)$. Besides, denote $A(1 + x_1)$ as E . For all $\delta \in \mathbb{N}$, the coefficients a_i satisfy

$$\sum_{i=0}^k a_i \delta^i = \frac{\int_{N(A(f)+\delta E)} 1 dV}{\text{ind } A(f) + \delta E}$$

from which we can compute all a_i for $i = 0, \dots, k$.

Proof. We observe that

$$\left| \bigcup_{0 \leq \ell \leq m} J_\ell \right| = |mA(f) + tE|,$$

so we can rewrite Equation (9) as follows:

$$\dim(\mathcal{L}) = |mA(f) + tE|. \quad (12)$$

The equivalence between Equation (9) and Equation (12) is powerful and makes the computation of $\dim(\mathcal{L})$ match well with additive combinations.

From Lemma 20, we know $\dim(\mathcal{L})$ is a polynomial in m and t for large enough m and t . Substituting $t = m$, using Lemma 18, $\dim(\mathcal{L}) = |m(E + A(f))|$ is a polynomial of degree k in m for large enough m , implying that the total degree of $\dim(\mathcal{L})(m, t)$ is k , that is, there exists $\{a_i\}_{0 \leq i \leq k}$,

$$\dim(\mathcal{L})(m, t) = a_0 m^k + a_1 t m^{k-1} + \dots + a_k t^k + o(m^k).$$

When substituting $t = \delta m$ for $\delta = 0, \dots, k$, we have $\dim(\mathcal{L}) = |m(A(f) + \delta E)| = (\sum_{i=0}^k a_i \delta^i) m^k + o(m^k)$. Comparing the coefficients of m^k on both sides, we obtain

$$\sum_{i=0}^k a_i \delta^i = \frac{\int_{N(A(f)+\delta E)} 1 dV}{\text{ind } A(f) + \delta E}$$

from which can compute all a_i for $i = 0, \dots, k$. □

Computing $\det(\mathcal{L})$ via Additive Combinatorics. The determinant of the lattice \mathcal{L} satisfies:

$$\det(\mathcal{L}) = X_1^{p_1} \cdots X_k^{p_k} \cdot M^{p_{\mathcal{F}}}.$$

Thus, computing $\det(\mathcal{L})$ reduces to determining p_j for $j = 1, \dots, k$ and $p_{\mathcal{F}}$.

Computation of p_j : We establish the following result:

Theorem 30. *For any well-defined k -variate f , the p_j obtained by the Jochemsz-May Extended Strategy is polynomial in t and m when t and m are large enough for $j = 1, \dots, k$. More precisely,*

$$p_j(m, t) = b_0^{(j)} m^{k+1} + b_1^{(j)} t m^k + \dots + b_{k+1}^{(j)} t^{k+1} + o(m^{k+1}),$$

when $t = O(m)$. Besides, denote $A(1 + x_1)$ as E . For all $\delta \in \mathbb{N}$, the coefficients $b_i^{(j)}$ satisfy

$$\sum_{i=0}^{k+1} b_i^{(j)} \delta^i = \frac{\int_{N(A(f)+\delta E)} x_j dV}{\text{ind } A(f) + \delta E}$$

from which we can compute all $b_i^{(j)}$ for $i = 0, \dots, k+1$.

Proof. The exponent of X_j in $\text{LC}(g_\lambda(x_1 X_1, \dots, x_k X_k))$ equals to the exponent of x_j in monomial λ , which enables us to formally express $p_j(m)$ via $A(f)$:

$$p_j(m, t) = \sum_{(i_1, \dots, i_k) \in mA(f) + tE} i_j. \quad (13)$$

By substituting f and $1 + x_1$ into Lemma 24, we obtain for large enough m and t , there exist $b_i^{(j)}$ for $i = 0, \dots, k+1$ such that:

$$p_j(m, t) = b_0^{(j)} m^{k+1} + b_1^{(j)} t m^k + \dots + b_{k+1}^{(j)} t^{k+1} + o(m^{k+1}).$$

Besides, for all $\delta \in \mathbb{N}$, $b_i^{(j)}$ satisfy

$$\sum_{i=0}^{k+1} b_i^{(j)} \delta^i = \frac{\int_{N(A(f)+\delta E)} x_j dV}{\text{ind } A(f) + \delta E}.$$

□

Computation of $p_{\mathcal{F}}$: We establish the following result:

Theorem 31. *For any well-defined k -variate f , the $p_{\mathcal{F}}$ obtained by the Jochemsz-May Extended Strategy is polynomial in t and m when t and m are large enough. More precisely,*

$$p_{\mathcal{F}}(m, t) = c_0 m^{k+1} + c_1 t m^k + \dots + c_{k+1} t^{k+1} + o(m^{k+1}),$$

when $t = O(m)$. Besides, denote $A(1 + x_1)$ as E . For all $\delta \in \mathbb{N}$, the coefficients c_i satisfy

$$\sum_{i=0}^{k+1} c_i \delta^i = \frac{\int_{N(A(f)+\delta E)} 1 dV}{\text{ind } A(f) + \delta E} - \frac{\int_{N(\tilde{A}(f)+\delta \tilde{E})} 1 dV}{\text{ind } \tilde{A}(f) + \delta \tilde{E}}.$$

Proof. First, we use Abel's summation formula to rewrite $p_{\mathcal{F}}$.

$$p_{\mathcal{F}}(m, t) = \sum_{\ell=0}^m (m - \ell) |J_\ell \setminus J_{\ell+1}| = (m+1) |J_0| - \sum_{\ell=0}^m |J_\ell|. \quad (14)$$

Now we have

$$\sum_{\ell=0}^m |J_\ell| = |m \tilde{A}(f) + t \tilde{E}|,$$

so the computation of $p_{\mathcal{F}}$ is

$$p_{\mathcal{F}}(m, t) = (m+1) |mA(f) + tE| - |m \tilde{A}(f) + t \tilde{E}|.$$

For large enough m and t , as $|mA(f) + tE|$ and $|m\tilde{A}(f) + t\tilde{E}|$ both become polynomial in t and m by Lemma 20, we obtain there exist c_i for $i = 0, \dots, k+1$ such that

$$p_{\mathcal{F}}(m, t) = c_0 m^{k+1} + c_1 t m^k + \dots + c_{k+1} t^{k+1} + o(m^{k+1}).$$

Besides, by Lemma 20, for all $\delta \in \mathbb{N}$, c_i satisfy

$$\sum_{i=0}^{k+1} c_i \delta^i = \frac{\int_{N(A(f)+\delta E)} 1 dV}{\text{ind } A(f) + \delta E} - \frac{\int_{N(\tilde{A}(f)+\delta \tilde{E})} 1 dV}{\text{ind } \tilde{A}(f) + \delta \tilde{E}}.$$

□

Our final formulas. For any *well-defined* k -variate polynomial f , we can now provide formulas for the bound yielded by the Jochemsz-May Extended Strategy for Coppersmith's method, i.e., there exist $\{a_i\}_{0 \leq i \leq k}$, $\{b_i^{(j)}\}_{0 \leq i \leq k+1}$, and $\{c_i\}_{0 \leq i \leq k+1}$, such that $\det(\mathcal{L}) < M^{m \dim(\mathcal{L})}$ can be rewritten as follows, for any arbitrarily small constant $\varepsilon > 0$:

$$\prod_{j=1}^k X_j^{\sum_{i=0}^{k+1} b_i^{(j)} m^{k+1-i} t^i} \cdot M^{\sum_{i=0}^{k+1} c_i m^{k+1-i} t^i} < M^{\sum_{i=0}^k a_i m^{k+1-i} t^i - \varepsilon m^k} \quad (15)$$

These coefficients satisfy the following system of equations, which can be determined by solving the linear equation system for $\delta = 0, \dots, k+1$:

$$\begin{cases} \sum_{i=0}^k a_i \delta^i = \frac{\int_{N(A(f)+\delta E)} 1 dV}{\text{ind } A(f) + \delta E}, \\ \sum_{i=0}^{k+1} b_i^{(j)} \delta^i = \frac{\int_{N(A(f)+\delta E)} x_j dV}{\text{ind } A(f) + \delta E}, \\ \sum_{i=0}^{k+1} c_i \delta^i = \frac{\int_{N(A(f)+\delta E)} 1 dV}{\text{ind } A(f) + \delta E} - \frac{\int_{N(\tilde{A}(f)+\delta \tilde{E})} 1 dV}{\text{ind } \tilde{A}(f) + \delta \tilde{E}}. \end{cases}$$

Remark 32. The Jochemsz-May Extended Strategy serves as a general framework. However, in crypt-analysis, certain problems demand specialized techniques to achieve stronger results, as seen in the case of the Small Private Exponent RSA Problem [BD00, HM10]. In Appendix B, we further investigate the use of Newton polytopes to compute the dimension and determinant of these lattices involved with special techniques.

4 Solving a System of Polynomial Equations

In this section, we first generalize the Jochemsz-May Extended Strategy for a single polynomial equation to a system of polynomial equations, and then derive related formulas for the generalized case in Section 4.2. Furthermore, by introducing additional shift polynomials to the strategies proposed by Meers–Nowakowski [MN23] and Sarkar [Sar24] for solving systems of polynomial equations, we extend their methods to a more generalized framework and derive the corresponding formulas. Since the approaches of Meers–Nowakowski and Sarkar can be regarded as special cases of our generalization, our formulas naturally encompass their results.

Notations. In this section, we use the following notations. Suppose $\mathcal{F} = \{f_1, \dots, f_n\} \subseteq \mathbb{Z}[x_1, \dots, x_k]$ is a set of polynomials with a common small root $\mathbf{u} = (u_1, \dots, u_k) \in \mathbb{Z}^k$ modulo some integer M . Here, a small root means $|u_i| < X_i$ for known bound X_i , for $i = 1, \dots, k$. Let $\alpha_j \in \mathbb{Z}^k$ represent the integer points corresponding to the leading monomial of f_j for $j = 1, \dots, n$. That is, for $\alpha_j = (\alpha_j^{(1)}, \dots, \alpha_j^{(k)})$, we have $\text{LM}(f_j) = x_1^{\alpha_j^{(1)}} \cdots x_k^{\alpha_j^{(k)}}$. We denote B as $\{\alpha_1, \dots, \alpha_n\}$. And we define $\tilde{A}(f)$ as follows

$$\tilde{A}(f) = \{(i_1, \dots, i_k, 1) \mid (i_1, \dots, i_k) \in A(f)\} \cup \{(\alpha^{(1)}, \dots, \alpha^{(k)}, 0)\},$$

where $x_1^{\alpha^{(1)}} \cdots x_k^{\alpha^{(k)}}$ is the leading monomial of f .

We also denote E and \tilde{E} as follow:

$$E = \{(0, \dots, 0), \underbrace{(1, 0, \dots, 0)}_{\text{Only the 1-th position is 1}}\} \subset \mathbb{Z}^k,$$

$$\tilde{E} = \{(0, \dots, 0, 0), \underbrace{(1, 0, \dots, 0, 0)}_{\text{Only the 1-th position is 1}}\} \subset \mathbb{Z}^{k+1}.$$

4.1 Generalized Jochemsz-May Extended Strategy with Shift x_1

We first generalize the Jochemsz-May Extended Strategy to solve a system of polynomial equations with shift x_1 and then provide the corresponding formulas for $\dim(\mathcal{L})$ and $\det(\mathcal{L})$ later.

We denote $A(\mathcal{F})$ as follows:

$$A(\mathcal{F}) = \bigcup_{j=1}^n A(f_j). \quad (16)$$

Fix integers m and t . We define the sets

$$S_\ell = \{x_1^{i_1} \cdots x_k^{i_k} \mid (i_1, \dots, i_k) \in tE + (m - \ell)A(\mathcal{F}) + \bigcup_{\substack{\ell_1 + \dots + \ell_n = \ell \\ 0 \leq \ell_j \leq m}} \sum_{j=1}^n \ell_j \alpha_j\}$$

$$= \{x_1^{i_1} \cdots x_k^{i_k} \mid (i_1, \dots, i_k) \in tE + (m - \ell)A(\mathcal{F}) + \ell B\}, \quad (17)$$

for $0 \leq \ell \leq m$ and $S_{m+1} = \emptyset$, so that $S_m \subset \dots \subset S_0$. The strategy constructs basis polynomials for every monomial $\lambda = x_1^{i_1} \cdots x_k^{i_k} \in S_\ell \setminus S_{\ell+1}$:

Step 1: Compute ℓ_1, \dots, ℓ_n with $\sum_{j=1}^n \ell_j = \ell$, such that

$$(i_1, \dots, i_n) \in tE + (m - \ell)A(\mathcal{F}) + \sum_{j=1}^n \ell_j \alpha_j.$$

Step 2: Construct $g_\lambda(x_1, \dots, x_k)$ as follows:

$$g_\lambda(x_1, \dots, x_k) = \frac{\lambda}{\prod_{j=1}^n \text{LM}(f_j)^{\ell_j}} \prod_{j=1}^n f_j^{\ell_j} M^{m-\ell}. \quad (18)$$

Then we use the coefficient vectors of all $g_\lambda(x_1 X_1, \dots, x_k X_k)$ to form the basis of lattice \mathcal{L} . We provide Algorithm 1 to understand our strategy better.

4.2 Our formulas for $\dim(\mathcal{L})$ and $\det(\mathcal{L})$.

Next, we provide related formulas for computing the dimension and determinant for our Generalized Jochemsz-May Extended Strategy with shift x_1 .

Computing Dimension via Additive Combinatorics. For dimension, we obtain the following theorem:

Theorem 33. *For f_1, \dots, f_n such that $\sum_{j=1}^n f_j$ is well-defined k -variate, the $\dim(\mathcal{L})$ obtained by Algorithm 1 is polynomial in t and m when t and m are large enough. More precisely,*

$$\dim(\mathcal{L})(m, t) = a_0 m^k + a_1 t m^{k-1} + \dots + a_k t^k + o(m^k),$$

when $t = O(m)$. For all $\delta \in \mathbb{N}$, the coefficients a_i satisfy

$$\sum_{i=0}^k a_i \delta^i = \frac{\int_{N(A(\mathcal{F}) + \delta E)} 1 dV}{\text{ind } A(\mathcal{F}) + \delta E},$$

from which we can compute all a_i for $i = 0, \dots, k$.

Algorithm 1: Construction of the lattice with extra shifts of x_1

Input: $\mathcal{F} = \{f_1, \dots, f_n\} \subset \mathbb{Z}[x_1, \dots, x_k]$, integers t and m , modulus M , and bounds X_j for $j = 1, \dots, k$

Output: Set of polynomials, whose coefficient vector forms the basis of the lattice \mathcal{L}

- 1 Define $\alpha_j \in \mathbb{Z}^k$ is the integer point correspond to $\text{LM}(f_j)$ for $j = 1, \dots, n$;
- 2 Define $B = \{\alpha_1, \dots, \alpha_n\}$;
- 3 Compute $A(\mathcal{F}) = \bigcup_{j=1}^n A(f_j)$;
- 4 $G \leftarrow \emptyset$;
- 5 **for** $(i_1, \dots, i_k) \in tE + mA$ **do**
- 6 $\ell \leftarrow 0$;
- 7 **while** $(i_1, \dots, i_k) \in tE + (m - \ell)A(\mathcal{F}) + \ell B$ **do**
- 8 $\ell \leftarrow \ell + 1$;
- 9 **end**
- 10 Compute ℓ_1, \dots, ℓ_n with $\sum_{j=1}^n \ell_j = \ell$, such that $(i_1, \dots, i_n) \in tE + (m - \ell)A(\mathcal{F}) + \sum_{j=1}^n \ell_j \alpha_j$;
- 11 $g(x_1, \dots, x_k) = \frac{x_1^{i_1} \dots x_k^{i_k}}{\prod_{j=1}^n \text{LM}(f_j)^{\ell_j}} \prod_{j=1}^n f_j^{\ell_j} M^{m-\ell}$;
- 12 $G \leftarrow G \cup \{g(x_1 X_1, \dots, x_k X_k)\}$;
- 13 **end**
- 14 **return** G ;

Proof. We have

$$\dim(\mathcal{L}) = \left| \bigcup_{0 \leq \ell \leq m} S_\ell \right| = |mA(\mathcal{F}) + tE|.$$

As $\sum_{j=1}^n f_j$ is also a well-defined k -variate polynomial and $A(\mathcal{F}) = A(\sum_{j=1}^n f_j)$, Then from Lemma 20, we know $\dim(\mathcal{L})$ is a polynomial in m and t for large enough m and t . Substituting $t = m$, using Lemma 18, $\dim(\mathcal{L}) = |m(E + A(\mathcal{F}))|$ is a polynomial of degree k in m for large enough m , implying that the total degree of $\dim(\mathcal{L})(m, t)$ is k , that is, there exists $\{a_i\}_{0 \leq i \leq k}$,

$$\dim(\mathcal{L})(m, t) = a_0 m^k + a_1 t m^{k-1} + \dots + a_k t^k + o(m^k).$$

When substituting $t = \delta m$ for $\delta = 0, \dots, k$, we have $\dim(\mathcal{L}) = |m(A(\mathcal{F}) + \delta E)| = (\sum_{i=0}^k a_i \delta^i) m^k + o(m^k)$. Comparing the coefficients of m^k on both sides, we obtain

$$\sum_{i=0}^k a_i \delta^i = \frac{\int_{N(A(\mathcal{F}) + \delta E)} 1 dV}{\text{ind } A(\mathcal{F}) + \delta E},$$

from which we can compute all a_i for $i = 0, \dots, k$. □

Computing Determinant via Additive Combinatorics. The determinant of the lattice \mathcal{L} satisfies:

$$\det(\mathcal{L}) = X_1^{p_1} \dots X_k^{p_k} \cdot M^{p_{\mathcal{F}}}.$$

Thus, computing $\det(\mathcal{L})$ reduces to determining p_j for $j = 1, \dots, k$ and $p_{\mathcal{F}}$.

Computation of p_j : We establish the following result:

Theorem 34. For f_1, \dots, f_n such that $\sum_{j=1}^n f_j$ is well-defined k -variate, the p_j obtained by Algorithm 1 is polynomial in t and m when t and m are large enough. More precisely,

$$p_j(m, t) = b_0^{(j)} m^{k+1} + b_1^{(j)} t m^k + \dots + b_{k+1}^{(j)} t^{k+1} + o(m^{k+1}),$$

when $t = O(m)$. For all $\delta \in \mathbb{N}$, the coefficients $b_i^{(j)}$ satisfy

$$\sum_{i=0}^{k+1} b_i^{(j)} \delta^i = \frac{\int_{N(A(\mathcal{F}) + \delta E)} x_j dV}{\text{ind } A(\mathcal{F}) + \delta E},$$

from which we can compute all $b_i^{(j)}$ for $i = 0, \dots, k+1$.

Proof. By Equation (18), each monomial λ induces a polynomial $g_\lambda(x_1, \dots, x_k)$ whose leading monomial is λ . Hence the exponent of X_j in $\text{LC}(g_\lambda(x_1 X_1, \dots, x_k X_k))$ equals to the exponent of x_j in monomial λ , which enables us to formally express $p_j(m)$ via $A(\mathcal{F})$:

$$p_j(m, t) = \sum_{(i_1, \dots, i_k) \in mA(\mathcal{F}) + tE} i_j. \quad (19)$$

By Lemma 24, we obtain for large enough m and t , there exist $b_i^{(j)}$ for $i = 0, \dots, k+1$ such that:

$$p_j(m, t) = b_0^{(j)} m^{k+1} + b_1^{(j)} t m^k + \dots + b_{k+1}^{(j)} t^{k+1} + o(m^{k+1}).$$

Besides, for all $\delta \in \mathbb{N}$, $b_i^{(j)}$ satisfy

$$\sum_{i=0}^{k+1} b_i^{(j)} \delta^i = \frac{\int_{N(A(\mathcal{F}) + \delta E)} x_j dV}{\text{ind } A(\mathcal{F}) + \delta E}.$$

□

Computation of $p_{\mathcal{F}}$: We establish the following result:

Theorem 35. *For f_1, \dots, f_n such that $\sum_{j=1}^n f_j$ is well-defined k -variate, the $p_{\mathcal{F}}$ obtained by Algorithm 1 is polynomial in t and m when t and m are large enough. More precisely,*

$$p_{\mathcal{F}}(m, t) = c_0 m^{k+1} + c_1 t m^k + \dots + c_{k+1} t^{k+1} + o(m^{k+1}),$$

when $t = O(m)$. For all $\delta \in \mathbb{N}$, the coefficients c_i satisfy

$$\sum_{i=0}^{k+1} c_i \delta^i = \frac{\int_{N(A(\mathcal{F}) + \delta E)} 1 dV}{\text{ind } A(\mathcal{F}) + \delta E} - \frac{\int_{N(\tilde{A}(\mathcal{F}) + \delta \tilde{E})} 1 dV}{\text{ind } \tilde{A}(\mathcal{F}) + \delta \tilde{E}}.$$

Proof. First, we use Abel's summation formula to rewrite $p_{\mathcal{F}}$.

$$p_{\mathcal{F}}(m, t) = \sum_{\ell=0}^m (m - \ell) |S_\ell \setminus S_{\ell+1}| = (m+1) |S_0| - \sum_{\ell=0}^m |S_\ell|. \quad (20)$$

To compute $\sum_{\ell=0}^m |J_\ell|$ in Equation (20), we define $\tilde{A}(\mathcal{F})$ as follows:

$$\tilde{A}(\mathcal{F}) = \bigcup_{j=1}^n \tilde{A}(f_j) \quad (21)$$

Then we have

$$\sum_{\ell=0}^m |S_\ell| = |m\tilde{A}(\mathcal{F}) + t\tilde{E}|,$$

and

$$p_{\mathcal{F}}(m, t) = (m+1)|S_0| - \sum_{\ell=0}^m |S_\ell| = m|m\tilde{A}(\mathcal{F}) + t\tilde{E}| - |m\tilde{A}(\mathcal{F}) + t\tilde{E}|.$$

For large enough m and t , as $|m\tilde{A}(\mathcal{F}) + t\tilde{E}|$ and $|m\tilde{A}(\mathcal{F}) + t\tilde{E}|$ both become polynomial in t and m , we obtain there exist c_i for $i = 0, \dots, k+1$ such that:

$$p_{\mathcal{F}}(m, t) = c_0 m^{k+1} + c_1 t m^k + \dots + c_{k+1} t^{k+1} + o(m^{k+1}),$$

when $t = O(m)$. Besides, for all $\delta \in \mathbb{N}$, c_i satisfy

$$\sum_{i=0}^{k+1} c_i \delta^i = \frac{\int_{N(A(\mathcal{F}) + \delta E)} 1 dV}{\text{ind } A(\mathcal{F}) + \delta E} - \frac{\int_{N(\tilde{A}(\mathcal{F}) + \delta \tilde{E})} 1 dV}{\text{ind } \tilde{A}(\mathcal{F}) + \delta \tilde{E}}.$$

□

Our formulas. We can now compute the relevant formulas for the bounds obtained using our strategy in Coppersmith's method. There exist $\{a_i\}_{0 \leq i \leq k}$, $\{b_i^{(j)}\}_{0 \leq i \leq k+1}$, and $\{c_i\}_{0 \leq i \leq k+1}$, such that $\det(\mathcal{L}) < M^{m \dim(\mathcal{L})}$ can be rewritten as follows, for any arbitrarily small constant $\varepsilon > 0$:

$$\prod_{j=1}^k X_j^{\sum_{i=0}^{k+1} b_i^{(j)} m^{k+1-i} t^i} \cdot M^{\sum_{i=0}^{k+1} c_i m^{k+1-i} t^i} < M^{\sum_{i=0}^k a_i m^{k+1-i} t^i - \varepsilon m^k} \quad (22)$$

These coefficients satisfied the following equations, which can be determined by solving the linear equation system for $\delta = 0, \dots, k+1$:

$$\begin{cases} \sum_{i=0}^k a_i \delta^i = \frac{\int_{N(A(\mathcal{F})+\delta E)} 1 dV}{\text{ind } A(\mathcal{F}) + \delta E}, \\ \sum_{i=0}^{k+1} b_i^{(j)} \delta^i = \frac{\int_{N(A(\mathcal{F})+\delta E)} x_j dV}{\text{ind } A(\mathcal{F}) + \delta E}, \\ \sum_{i=0}^{k+1} c_i \delta^i = \frac{\int_{N(A(\mathcal{F})+\delta E)} 1 dV}{\text{ind } A(\mathcal{F}) + \delta E} - \frac{\int_{N(\tilde{A}(\mathcal{F})+\delta \tilde{E})} 1 dV}{\text{ind } \tilde{A}(\mathcal{F}) + \delta \tilde{E}}. \end{cases}$$

Remark 36. When considering multiple shifts, i.e., using $\sum_{j=1}^k t_j E_j$ instead of tE in Equation (17), the dimension of the lattice is given by $|mA(\mathcal{F}) + \sum_{j=1}^k t_j E_j|$, which is a polynomial in m, t_1, \dots, t_k for sufficiently large m, t_1, \dots, t_k . The formulas for $\dim(\mathcal{L})$ and $\det(\mathcal{L})$ are analogous to those presented in Section 4.2.

4.3 Some Variants

By incorporating additional shift polynomials, we further generalize the approaches of Meers–Nowakowski [MN23] and Sarkar [Sar24], referred to as Variant I and Variant II, respectively, and derive the corresponding formulas. Proofs for the related formulas are analogous to those presented in Section 4.2.

Variant I. We first define the sets of vertices as follows:

$$V_{\ell_1, \dots, \ell_n} = \sum_{j=1}^n \left(\frac{m}{n} - \ell_j \right) A(f_j) + \ell_j \alpha_j,$$

and

$$V_\ell = \bigcup_{\substack{\ell_1 + \dots + \ell_n = \ell \\ 0 \leq \ell_j \leq \frac{m}{n}}} V_{\ell_1, \dots, \ell_n}.$$

Now we can define the following monomial sets as

$$S_\ell = \left\{ x_1^{i_1} \cdots x_k^{i_k} \mid (i_1, \dots, i_k) \in tE + \bigcup_{\substack{\ell_1 + \dots + \ell_n = \ell \\ 0 \leq \ell_j \leq \frac{m}{n}}} \sum_{j=1}^n \left(\frac{m}{n} - \ell_j \right) A(f_j) + \ell_j \alpha_j \right\}.$$

for $0 \leq \ell \leq m$ and $S_{m+1} = \emptyset$, so that $S_m \subset \dots \subset S_0$. Variant I constructs basis polynomials for every monomial $\lambda = x_1^{i_1} \cdots x_k^{i_k} \in S_\ell \setminus S_{\ell+1}$:

Step 1: Compute ℓ_1, \dots, ℓ_n with $\sum_{j=1}^n \ell_j = \ell$ and $0 \leq \ell_j \leq \frac{m}{n}$, such that

$$(i_1, \dots, i_n) \in tE + V_{\ell_1, \dots, \ell_n}.$$

Step 2: Construct $g_\lambda(x_1, \dots, x_k)$ as follows:

$$g_\lambda(x_1, \dots, x_k) = \frac{\lambda}{\prod_{j=1}^n \text{LM}(f_j)^{\ell_j}} \prod_{j=1}^n f_j^{\ell_j} M^{m-\ell}.$$

We use the coefficient vectors of all $g_\lambda(x_1 X_1, \dots, x_k X_k)$ to form the basis of the lattice \mathcal{L} .

We define $A(\mathcal{F}) = \frac{1}{n} \sum_{j=1}^n A(f_j)$ and $\tilde{A}(\mathcal{F}) = \frac{1}{n} \sum_{j=1}^n \tilde{A}(f_j)$. The formulas yielded by Variant I are the same as those presented in Section 4.2 with the newly defined $A(\mathcal{F})$ and $\tilde{A}(\mathcal{F})$.

Remark 37. Here, we require $0 \leq \ell_j \leq \frac{m}{n}$ to simplify the calculations. However, when f_j has special structures, special structures, such as $f_1 = x_1 + 1$ and $f_2 = (x_1 + 1)(x_2 + 1)(x_3 + 1)$ where f_1 divides f_2 , allowing $\ell_j > \frac{m}{n}$ can lead to a better bound. In this case, it does not affect p_M and p_j for $j = 1, \dots, k$, but the computation of $p_{\mathcal{F}}$ requires additional manual computations based on our formulas.

Remark 38. The Meer–Nowakowski Strategy is a special case of Variant I with $t = 0$, so our proof can be viewed as a proof of their new heuristic.

Variant II. Fix integers t_1, t_2 and m . Define $A^{(1)}(\mathcal{F}) = \bigcup_{j=1}^n A(f_j)$ and $A^{(2)}(\mathcal{F}) = \sum_{j=1}^n A(f_j)$. We define the sets

$$\begin{aligned} S_\ell = & \bigcup_{\substack{\ell_1 + \ell_2 = \ell \\ \ell_1, \ell_2 \geq 0}} \{x_1^{i_1} \cdots x_k^{i_k} \mid (i_1, \dots, i_k) \in (m - \ell_1)A^{(1)}(\mathcal{F}) + \ell_1 B \\ & + \bigcup_{\substack{\ell_2^{(1)} + \dots + \ell_2^{(n)} = \ell_2 \\ 0 \leq \ell_2^{(j)} \leq t_1}} \sum_{j=1}^n (t_1 - \ell_2^{(j)})A(f_j) + \ell_2^{(j)}\alpha_j + t_2 E\} \end{aligned} \quad (23)$$

for $0 \leq \ell \leq m$ and $S_{m+1} = \emptyset$, so that $S_m \subset \dots \subset S_0$. Variant I constructs basis polynomials for every monomial $\lambda = x_1^{i_1} \cdots x_k^{i_k} \in S_\ell \setminus S_{\ell+1}$:

Step 1: Compute ℓ_1, \dots, ℓ_n with $\sum_{j=1}^n \ell_j = \ell$, such that

$$(i_1, \dots, i_n) \in (m - \ell)A^{(1)}(\mathcal{F}) + \ell B + t_1 A^{(2)}(\mathcal{F}) + t_2 E.$$

Step 2: Construct $g_\lambda(x_1, \dots, x_k)$ as follows:

$$g_\lambda(x_1, \dots, x_k) = \frac{\lambda}{\prod_{j=1}^n \text{LM}(f_j)^{\ell_j}} \prod_{j=1}^n f_j^{\ell_j} M^{m-\ell}.$$

We use the coefficient vectors of all $g_\lambda(x_1 X_1, \dots, x_k X_k)$ to form the basis of the lattice \mathcal{L} , and define $\tilde{A}^{(1)}(\mathcal{F}) = \bigcup_{j=1}^n \tilde{A}(f_j)$ and $\tilde{A}^{(2)}(\mathcal{F}) = \sum_{j=1}^n \tilde{A}(f_j)$. The formulas for the bound obtained by Variant II of Coppersmith's method are as follows:

There exist polynomials g_M, g_1, \dots, g_k and $g_{\mathcal{F}} \in \mathbb{Q}[x_1, x_2]$ with total degree $k + 1$, such that $\det(\mathcal{L}) < M^{m \dim(\mathcal{L})}$ can be rewritten as follows, for any arbitrarily small constant $\varepsilon > 0$:

$$\prod_{j=1}^k X_j^{m^{k+1} g_j(\frac{t_1}{m}, \frac{t_2}{m})} \cdot M^{m^{k+1} g_{\mathcal{F}}(\frac{t_1}{m}, \frac{t_2}{m})} < M^{m^{k+1} g_M(\frac{t_1}{m}, \frac{t_2}{m}) - \varepsilon m^k}.$$

These coefficients of polynomials g_M, g_1, \dots, g_k and $g_{\mathcal{F}}$ satisfied the following equations, which can be determined by solving the linear equation system for different values of δ_1 and δ_2 :

$$\begin{cases} g_M(\delta_1, \delta_2) = \frac{\int_{N(A^{(1)}(\mathcal{F}) + \delta_1 A^{(2)}(\mathcal{F}) + \delta_2 E)} 1 dV}{\text{ind } A^{(1)}(\mathcal{F}) + \delta_1 A^{(2)}(\mathcal{F}) + \delta_2 E}, \\ g_j(\delta_1, \delta_2) = \frac{\int_{N(A^{(1)}(\mathcal{F}) + \delta_1 A^{(2)}(\mathcal{F}) + \delta_2 E)} x_j dV}{\text{ind } A^{(1)}(\mathcal{F}) + \delta_1 A^{(2)}(\mathcal{F}) + \delta_2 E}, \\ g_{\mathcal{F}}(\delta_1, \delta_2) = \frac{\int_{N(A^{(1)}(\mathcal{F}) + \delta_1 A^{(2)}(\mathcal{F}) + \delta_2 E)} 1 dV}{\text{ind } A^{(1)}(\mathcal{F}) + \delta_1 A^{(2)}(\mathcal{F}) + \delta_2 E} - \frac{\int_{N(\tilde{A}^{(1)}(\mathcal{F}) + \delta_1 \tilde{A}^{(2)}(\mathcal{F}) + \delta_2 \tilde{E})} 1 dV}{\text{ind } \tilde{A}^{(1)}(\mathcal{F}) + \delta_1 \tilde{A}^{(2)}(\mathcal{F}) + \delta_2 \tilde{E}}. \end{cases}$$

Remark 39. Sarkar's strategy [Sar24] is a special case of Variant II with $t_2 = 0$, so our proof can be viewed as a proof of the new heuristic. Moreover, our strategy in Section 4.1 is also a special case of Variant II with $t_1 = 0$.

Variant III. We also analyze how to improve the lattice without using additional shift polynomials and propose Variant III as follows.

We first observe two ways to build the lattice for univariate polynomials. One is the Jochemsz–May Extended Strategy and the other chooses to replace $m - \ell$ by $\max(t - \ell, 0)$ with the parameter t

instead of introducing additional polynomials $\{x^j f^m \mid \text{for } j = 0, \dots, t\}$. Specifically, we will compare the different lattice constructions in Theorem 7 in [May03] and Theorem 3 in [LZPL15], but the results are the same. Here we generalize the latter one to the multivariate case to obtain variant III. Its advantage lies in not introducing additional shift polynomials, which results in a lower lattice dimension.

Lemma 40 (Theorem 7 in [May03]). *Let M be an unknown factorization integer with a divisor $b \geq M^\beta$. Furthermore, let $f(x)$ be a univariate, monic polynomial of degree δ . Then we can find all solutions x_0 for the equation $f(x) \equiv 0 \pmod{b}$ with*

$$|x_0| \leq M^{\frac{\beta^2}{\delta} - \epsilon}$$

in time polynomial in $(\log N, \delta, \frac{1}{\epsilon})$.

Proof. Fix m and t , we define

$$G_{m,t} = \{x^j M^{m-i} f^i \mid \text{for } i = 0, \dots, m \text{ and } j = 0, \dots, \delta - 1\} \cup \{x^j f^m \mid \text{for } j = 0, \dots, t\}.$$

We then construct the lattice \mathcal{L} , defined by the coefficient vectors of all polynomials in $G_{m,t}$. Details can be found in the proof of Theorem 6 in [May03]. Here we just focus on the polynomial used to construct \mathcal{L} . \square

Next, we introduce a special case of Theorem 3 in [LZPL15], where we just set $u = v = 1$.

Lemma 41 (Theorem 3 in [LZPL15]). *Let M be an unknown factorization integer with a divisor $b \geq M^\beta$. Furthermore, let $f(x)$ be a univariate, monic polynomial of degree δ . Then we can find all solutions x_0 for the equation $f(x) \equiv 0 \pmod{b}$ with*

$$|x_0| \leq M^{\frac{\beta^2}{\delta} - \epsilon}$$

in time polynomial in $(\log N, \delta, \frac{1}{\epsilon})$.

Proof. Fix m and t , we define

$$G_{m,t} = \{x^j M^{\max\{t-i, 0\}} f^i \mid \text{for } i = 0, \dots, m \text{ and } j = 0, \dots, \delta - 1\},$$

then construct the lattice \mathcal{L} , defined by the coefficient vectors of all polynomials in $G_{m,t}$. \square

Compared Lemma 41 with Lemma 40, there is no need to introduce $\{x^j f^m \mid \text{for } j = 0, \dots, t\}$ in the Jochemsz-May Extended Strategy, for the univariate polynomial equations. Next, we generalize the idea to the multivariate case as the Variant III.

Here we focus on modular unknown divisor case. Let M be an unknown factorization integer with a divisor $b \geq M^\beta$. Suppose $\mathcal{F} = \{f_1, \dots, f_n\} \subseteq \mathbb{Z}[x_1, \dots, x_k]$ is a set of polynomials with a common small root $\mathbf{u} = (u_1, \dots, u_k) \in \mathbb{Z}^k$ modulo some integer b .

Fix integers m and t . Consider $A(\mathcal{F}) = \bigcup_{j=1}^n A(f_j)$, we define the sets

$$S_\ell = \{x_1^{i_1} \cdots x_k^{i_k} \mid (i_1, \dots, i_k) \in (m - \ell)A(\mathcal{F}) + \ell B\}.$$

for $0 \leq \ell \leq m$ and $S_{m+1} = \emptyset$, so that $S_m \subset \dots \subset S_0$. The strategy constructs basis polynomials $g_\lambda(x_1, \dots, x_k)$ for every monomial $\lambda = x_1^{i_1} \cdots x_k^{i_k} \in S_\ell \setminus S_{\ell+1}$, satisfying $g_\lambda(\mathbf{u}) \equiv 0 \pmod{b^t}$:

Step 1: Compute ℓ_1, \dots, ℓ_n with $\sum_{j=1}^n \ell_j = \ell$, such that

$$(i_1, \dots, i_n) \in tE + (m - \ell)A(\mathcal{F}) + \sum_{j=1}^n \ell_j \alpha_j.$$

Step 2: Construct $g_\lambda(x_1, \dots, x_k)$ as follows:

$$g_\lambda(x_1, \dots, x_k) = \frac{\lambda}{\prod_{j=1}^n \text{LM}(f_j)^{\ell_j}} \prod_{j=1}^n f_j^{\ell_j} M^{\max(t-\ell, 0)}.$$

Use the coefficient vectors of all $g_\lambda(x_1 X_1, \dots, x_k X_k)$ to form the basis of the lattice \mathcal{L} .

Our formulas. Define $\tilde{A}(\mathcal{F}) = \bigcup_{j=1}^n \tilde{A}(f_j)$ and $A'(\mathcal{F}) = \{(i_1, \dots, i_k, 1) \mid (i_1, \dots, i_k) \in A(\mathcal{F})\}$. The formulas for the bound obtained by Variant III of Coppersmith's method are as follows:

$$\left\{ \begin{array}{l} \dim(\mathcal{L}) = \frac{\int_{N(A(\mathcal{F}))} 1 dV}{\text{ind } A(\mathcal{F})} m^k + o(m^k), \\ p_j = \frac{\int_{N(A(\mathcal{F}))} x_j dV}{\text{ind } A(\mathcal{F})} m^{k+1} + o(m^{k+1}), \\ p_{\mathcal{F}} = \sum_{i=0}^{k+1} c_i m^{k+1-i} t^i + o(m^{k+1}), \end{array} \right.$$

where c_i satisfied the following equations, which can be determined through interpolation at $\delta = 0, 1, \frac{1}{2}, \dots, \frac{1}{k+1}$:

$$\sum_{i=0}^{k+1} c_i \delta^i = \frac{\int_{N(A(\mathcal{F}))} 1 dV}{\text{ind } A(\mathcal{F})} - \frac{\int_{N(\delta \tilde{A}(\mathcal{F}) + (1-\delta)A'(\mathcal{F}))} 1 dV}{\text{ind } \delta \tilde{A}(\mathcal{F}) + (1-\delta)A'(\mathcal{F})}.$$

Now we will analyze the formulas of $\dim(\mathcal{L})$ and $\det(\mathcal{L})$. Note that the computations of $\dim(\mathcal{L})$ and p_j in $\det(\mathcal{L})$ are a special case of Theorem 34, where we need to set $t = 0$ in Theorem 34. Hence we just need to focus on the computation of $p_{\mathcal{F}}$ in $\det(\mathcal{L})$. We have:

Theorem 42. Suppose $\mathcal{F} = \{f_1, \dots, f_n\} \subseteq \mathbb{Z}[x_1, \dots, x_k]$ and m is an integer. Then $p_{\mathcal{F}}$ obtained is polynomial in t and m when t and m are large enough. That is, there exists c_i for $i = 0, \dots, k$ such that

$$\begin{aligned} p_{\mathcal{F}}(t, m) &= c_0 m^{k+1} + c_1 t m^k + \dots + c_{k+1} t^{k+1} + o(m^{k+1}) \\ &\approx c_0 m^{k+1} + c_1 t m^k + \dots + c_{k+1} t^{k+1}. \end{aligned}$$

Besides, for all $\delta \in \mathbb{Q}^+$, c_i satisfy

$$\sum_{i=0}^{k+1} c_i \delta^i = \frac{\int_{N(A(\mathcal{F}))} 1 dV}{\text{ind } A(\mathcal{F})} - \frac{\int_{N(\delta \tilde{A}(\mathcal{F}) + (1-\delta)A'(\mathcal{F}))} 1 dV}{\text{ind } \delta \tilde{A}(\mathcal{F}) + (1-\delta)A'(\mathcal{F})}.$$

Furthermore, we can determine the coefficients c_i through interpolation at $\delta = 1, \frac{1}{2}, \dots, \frac{1}{k+1}$.

Proof. Here we only need to prove the following Equation (24).

$$p_{\mathcal{F}}(t, m) = (t+1) |mA(\mathcal{F})| - \left| t\tilde{A}(\mathcal{F}) + (m-t)A'(\mathcal{F}) \right|. \quad (24)$$

By the definition of $p_{\mathcal{F}}$, we have

$$\begin{aligned} p_{\mathcal{F}}(t, m) &= \sum_{\ell=0}^m \max\{t-\ell, 0\} |S_{\ell} \setminus S_{\ell+1}| \\ &= \sum_{\ell=0}^t (t-\ell) |S_{\ell} \setminus S_{\ell+1}| \\ &= (t+1) |S_0| - \sum_{\ell=0}^t |S_{\ell}|. \end{aligned}$$

By definition, $|S_0|$ is simply $|mA(\mathcal{F})|$. Thus, we only need to compute $\sum_{\ell=0}^t |S_{\ell}|$.

Define $\tilde{S}_{\ell} = \{(i_1, \dots, i_k, m-\ell) \mid (i_1, \dots, i_k) \in S_{\ell}\}$. Then, $|\tilde{S}_{\ell}| = |S_{\ell}|$ and we have $\bigcup_{\ell=0}^t \tilde{S}_{\ell} = t\tilde{A}(\mathcal{F}) + (m-t)A'(\mathcal{F})$. By definition, the section of $t\tilde{A}(\mathcal{F}) + (m-t)A'(\mathcal{F})$ whose last component is $m-\ell$ is exactly \tilde{S}_{ℓ} . Therefore,

$$\sum_{\ell=0}^t |S_{\ell}| = \sum_{\ell=0}^t |\tilde{S}_{\ell}| = \left| \bigcup_{\ell=0}^t \tilde{S}_{\ell} \right| = \left| t\tilde{A}(\mathcal{F}) + (m-t)A'(\mathcal{F}) \right|.$$

This completes the proof. \square

One may ask if introducing t is meaningful, i.e., leads to a better result. We also prove that it works if and only when the modulus is an unknown divisor of a known integer.

Corollary 43. *Let M be an unknown factorization integer with a divisor $b \geq M^\beta$. Furthermore, let $f(x)$ be a monic polynomial. Then when we use the above strategy to find solutions of the equation $f(x) \equiv 0 \pmod{b}$, the extremum point of t is not $\frac{t}{m} = 1$, which means the introduction of t is useful.*

Proof. Suppose $b = M^\beta$ is an unknown divisor of M , now we consider the following inequality

$$X_1^{p_1} \cdot \dots \cdot X_k^{p_k} \cdot M^{p_{\mathcal{F}}} < b^{t \dim(\mathcal{L})}.$$

Only focus on terms that contain t , we have

$$m^k t - \frac{m^{k+1} - (m-t)^{k+1}}{k+1} < \beta t m^k.$$

Suppose $\frac{t}{m} = \delta$, the above inequality can be rewritten as

$$(k+1)\delta - (1 - (1-\delta)^{k+1}) - (k+1)\beta\delta < 0.$$

The optimal value for δ is $\delta_0 = 1 - (1-\beta)^{\frac{1}{k}}$. Hence, when $\beta \neq 1$, we have $\delta_0 \neq 1$. \square

5 Applications

We first provide methods for computing these integrals in our formulas, which yield the asymptotic bounds directly. Then, we use these formulas to present our improvements to the CI-HNP over CSURF.

5.1 How to Compute the Integral?

Since the Jochemsz–May Basic/Extended Strategy is a special case of our strategy in Section 4.1, we focus on this strategy in this section. For our formulas in Section 4.2, we first demonstrate that these integrals are, in fact, the volumes of the convex hulls of specific subsets and that the number of points in these subsets is bounded by $d = 2(k+2)|\sum_{j=1}^n A(f_j)|$. Consequently, the computation of the integrals is reduced to computing the volume of the convex hull for a subset containing a given number d of points. Next, we present existing results on computing the volume of the convex hull for a subset of d points and conduct experiments to show its efficiency.

Computing Dimension. We need to compute the integral $\int_{N(A(\mathcal{F})+\delta E)} 1 dV$ for $\delta = 0, \dots, k$. This integral is the volume of the convex hull for $A(\mathcal{F}) + \delta E$, and the number of points in this subset is bounded by $(\delta+1)|A(\mathcal{F})|$.

Computing Determinant. Computing $\det(\mathcal{L})$ reduces to computing p_j for $j = 1, \dots, k$ and $p_{\mathcal{F}}$.

Computation of p_j : We need to compute the integral $\int_{N(A(\mathcal{F})+\delta E)} x_j dV$ for $\delta = 0, \dots, k$. Denote \mathcal{H}_j as

$$\mathcal{H}_j(A) = \bigcup_{(i_1, \dots, i_k) \in A} \{(i_1, \dots, i_k, 0), (i_1, \dots, i_k, 1), \dots, (i_1, \dots, i_k, i_j)\}.$$

We know $\int_{N(A(\mathcal{F})+\delta E)} x_j dV$ can be written as $|\mathcal{H}_j(A(\mathcal{F})) + \delta \mathcal{H}_j(E)|$ from the proof of Lemma 24 in Appendix C. Now, we denote $\overline{\mathcal{H}}_j(A)$ as

$$\overline{\mathcal{H}}_j(A) = \bigcup_{(i_1, \dots, i_k) \in A} \{(i_1, \dots, i_k, 0), (i_1, \dots, i_k, i_j)\}.$$

It can be verified that the convex hull of $\mathcal{H}_j(A)$ is equivalent to the convex hull of $\overline{\mathcal{H}}_j(A)$. Then we have

$$\int_{N(A(\mathcal{F})+\delta E)} x_j dV = |\overline{\mathcal{H}}_j(A(\mathcal{F})) + \delta \overline{\mathcal{H}}_j(E)|,$$

and the number of points in this subset is bounded by $(\delta + 1)|\overline{\mathcal{H}}_j(A(\mathcal{F}))|$, which is no more than $2(\delta + 1)|A(\mathcal{F})|$.

Computation of $p_{\mathcal{F}}$: We need to compute the integral $\int_{N(\tilde{A}(\mathcal{F}) + \delta\tilde{E})} 1 dV$, which is the volume of the convex hull of $\tilde{A}(\mathcal{F}) + \delta\tilde{E}$. As the number of points in $\tilde{A}(\mathcal{F})$ is no more than $2|\tilde{A}(\mathcal{F})|$, the number of points in $\tilde{A}(\mathcal{F}) + \delta\tilde{E}$ is bounded by $2(\delta + 1)|A(\mathcal{F})|$.

Computing the Volume of a Polyhedron. Volume computation is #P-hard for a polytope $P \subseteq \mathbb{R}^d$ represented as the convex hull of vertices [DF88]. However, there exists an $O^*(n^4)$ polynomial-time approximation algorithm [LV06] (where the asterisk indicates that the dependence on error parameters and logarithmic factors in n is not shown). For example, in cryptanalysis—and even for more complex polynomial equations—the exact volume can be computed quickly, typically within one second.

Experiments. We implemented our algorithms using SageMath 10.3 on a MacBook Pro equipped with an M1 chip with a maximum CPU clock rate of 3.2 GHz. The source code for the experiments is open-sourced and available at

<https://github.com/ffmath/AsymptoticBounds>.

We further compared our method with the heuristic approach [MN23, Sar24]. Since the heuristic method does not always yield a polynomial of degree $k + 1$ when m is small, for computing $p_{\mathcal{F}}$, a natural idea is to interpolate the results of $p_{\mathcal{F}}(m)$ for $m = 0, \dots, k + 1$, then for $m = 1, \dots, k + 2$, and to terminate the heuristic method when the interpolated polynomials stabilize. We conducted experiments under different values of k as the following Table 2. Our method shows significant time advantages, as these polynomials require large values of m before becoming polynomial-like. For instance, in Exp. 5 - Exp. 8, the minimum required values of m were 12, 12, 14, and 18, respectively. Moreover, the bounds computed by our approach are both *provable* and *efficient*.

Experiment	k	Interpolation [MN23]	Ours
Exp. 1	3	10.9 s	0.2 s
Exp. 2	3	46.2 s	0.04 s
Exp. 3	3	2238.2 s	0.1 s
Exp. 4	3	828.3 s	0.1 s
Exp. 5	4	33112.4 s	0.2 s
Exp. 6	4	32451.7 s	0.1 s
Exp. 7	4	-	0.1 s
Exp. 8	4	-	0.2 s

Table 2: Running time for computing asymptotic upper bounds. " - " means longer than 24 hours. The details about the polynomials used in our experiments can be found in Table 3 in Appendix D.

5.2 Cryptanalytic Advances for CI-HNP

The Commutative Isogeny Hidden Number Problem (CI-HNP), introduced by Meers and Nowakowski [MN23], investigates the bit-security of isogeny-based key exchange protocols. The problem formulation concerns whether an adversary can recover a shared secret elliptic curve when given the most significant bits (MSBs) of the Diffie–Hellman style exchanged key. They focus on both CSIDH and CSURF key exchanges. Recently, the CSURF case has been continuously improved [Rya24, Sar24]. These results are all based on Coppersmith’s method. Notably, they all used a new heuristic when computing the bounds in Coppersmith’s method.

To minimize prerequisites on isogeny background, we present a self-contained and simplified definition for CI-HNP over CSURF; the original version appears in [MN23].

Definition 44 (CI-HNP (CSURF)). *Given a modulus $M \in \mathbb{N}$ and polynomials*

$$\begin{cases} f_1(x_1, x_2, x_3) := x_1^2 + a_1x_1x_2^2 + a_2x_1x_2 + a_3x_1 + a_4x_2^2 + a_5x_2 + a_6, \\ f_2(x_1, x_2, x_3) := x_3^2 + b_1x_1^2x_3 + b_2x_1x_3 + b_3x_3 + b_4x_1^2 + b_5x_1 + b_6, \end{cases}$$

for some constants $a_i, b_i \in \mathbb{Z}$, the CI-HNP over CSURF is to solve roots \mathbf{u} of f_1 and f_2 modular M . We say that α MSBs are required if the system of polynomial equations can be solved for $\|\mathbf{u}\|_\infty < M^{1-\alpha}$.

In [MN23], their strategy can solve all small roots u for $\|\mathbf{u}\|_\infty < M^{10/41}$, so their result implies that $31/41 \approx 75.61\%$ MSBs are required.

Next, we will show how to use our formulas to obtain improved results without introducing additional heuristics.

Theorem 45. *Given a modulus $M \in \mathbb{N}$ and polynomials*

$$\begin{cases} f_1(x_1, x_2, x_3) := x_1^2 + a_1x_1x_2^2 + a_2x_1x_2 + a_3x_1 + a_4x_2^2 + a_5x_2 + a_6, \\ f_2(x_1, x_2, x_3) := x_3^2 + b_1x_1^2x_3 + b_2x_1x_3 + b_3x_3 + b_4x_1^2 + b_5x_1 + b_6, \end{cases}$$

for some constants $a_i, b_i \in \mathbb{Z}$, a bound $X \in \mathbb{N}$, then all small roots $\mathbf{u} = (u_1, u_2, u_3)$ of f_1 and f_2 modulo M with $|u_i| < X$ can be found in polynomial time in $\log(M)$ under Assumption 1 if

$$X < M^{0.2599}.$$

Proof. Using our Variant II in Section 4.3, we choose shift as x_2 in , i.e., $E = A(1 + x_2)$ and the order as $x_2 \prec_{\text{lex}} x_1 \prec_{\text{lex}} x_3$ to construct lattice \mathcal{L} .

Using our formulas for $\det(\mathcal{L}) < M^{m \dim(\mathcal{L})}$, we have:

$$\begin{aligned} \dim(\mathcal{L}) &= 12t_1^3 + 7t_1^2t_2 + 28t_1^2m + 10t_1t_2m + 16t_1m^2 + 3t_2m^2 + \frac{8}{3}m^3 + o(m^3), \\ p_1 &= \frac{32}{3}t_1^4 + \frac{19}{3}t_1^3t_2 + \frac{104}{3}t_1^3m + 15t_1^2t_2m + 33t_1^2m^2 + 11t_1t_2m^2 \\ &\quad + 12t_1m^3 + \frac{7}{3}t_2m^3 + \frac{3}{2}m^4 + o(m^4), \\ p_2 &= 19t_1^4 + \frac{38}{3}t_1^3t_2 + 52t_1^3m + 24t_1^2t_2m + 45t_1^2m^2 + 14t_1t_2m^2 \\ &\quad + 16t_1m^3 + \frac{8}{3}t_2m^3 + 2m^4 + o(m^4), \\ p_3 &= \frac{34}{3}t_1^4 + 12t_1^3t_2 + \frac{7}{2}t_1^3t_2^2 + \frac{112}{3}t_1^3m + 28t_1^2t_2m + 5t_1t_2^2m \\ &\quad + 36t_1^2m^2 + 16t_1t_2m^2 + \frac{3}{2}t_2^2m^2 + \frac{40}{3}t_1m^3 + \frac{8}{3}t_2m^3 + \frac{5}{3}m^4 + o(m^4), \\ p_{\mathcal{F}} &= -10t_1^4 - 8t_1^3t_2 - 20t_1^3m - 11t_1^2t_2m - 2t_1^2m^2 - t_1t_2m^2 \\ &\quad + \frac{16}{3}t_1m^3 + t_2m^3 + \frac{4}{3}m^4 + o(m^4). \end{aligned}$$

Substituting $t_1 = 0.401m$ and $t_2 = 0.111m$, we obtain $X < M^{0.2599}$. □

We compare our results with previous work in Table 1. Our framework enables precise security parameter selection for isogeny-based protocols by eliminating heuristic assumptions in bound computation and providing the first non-heuristic improvement over previous work.

Remark 46. In our previous theorem, we implicitly assumed that t_1 , t_2 , and m are all integers. This is also consistent with practical attacks, where, for example, $t_1 = \lfloor 0.401m \rfloor$ and $t_2 = \lfloor 0.111m \rfloor$ are often chosen. However, when computing asymptotic bounds, the literature typically ignores this requirement in order to derive the bounds.

6 Conclusion

In this paper, we introduced novel techniques from additive combinatorics—namely, sumset theory and integer point counting in integral polytopes—into the analysis of Coppersmith’s method. Our approach establishes the first provable and efficient algorithm for computing the asymptotic bounds of Coppersmith’s method. As an application, we improve the cryptanalytic result for the Commutative Isogeny Hidden Number Problem.

Open Problems. Lastly, we state some open problems. The heuristic used for computing bounds in [Rya24] remains unresolved; perhaps the introduced tools from sumset theory may inspire further research. In Section 4.3, we propose several variants. Of course, some variants have not been mentioned. Given a system of polynomial equations, determining which variant yields better bounds before computing the bounds is also an interesting problem.

Acknowledgments

We thank Shaoshi Chen, Long Guo, Xingjian Li, and Jesús de Loera for the discussion on Additive Combinatorics. This work is partially supported by the National Natural Science Foundation of China (No. 62372445, 62032009) and the Innovation Program for Quantum Science and Technology (2021ZD0302902).

References

- Ajt98. M. Ajtai. The shortest vector problem in L_2 is NP-hard for randomized reductions. In *Proceedings of the thirtieth annual ACM symposium on Theory of computing*, pages 10–19, 1998.
- Bar92. A. I. Barvinok. Partition functions in optimization and computational problems. *Algebra i Analiz*, 4:3–53, 1992.
- Bar94. A. I. Barvinok. Computation of exponential integrals. *Journal of Mathematical Sciences*, 70:1934–1943, 1994.
- BBDL⁺11. V. Baldoni, N. Berline, J. De Loera, M. Köppe, and M. Vergne. How to integrate a polynomial over a simplex. *Mathematics of Computation*, 80(273):297–325, 2011.
- BD00. D. Boneh and G. Durfee. Cryptanalysis of rsa with private key d less than $n^{\sup 0.292}$. *IEEE transactions on Information Theory*, 46(4):1339–1349, 2000.
- BDDL⁺24. E. Bajo, R. Davis, J. A. De Loera, A. Garber, S. G. Mora, K. Jochemko, and J. Yu. Weighted ehrhart theory: Extending stanley’s nonnegativity theorem. *Advances in Mathematics*, 444:109627, 2024.
- BHHG01. D. Boneh, S. Halevi, and N. Howgrave-Graham. The modular inversion hidden number problem. In *Advances in Cryptology—ASIACRYPT 2001: 7th International Conference on the Theory and Application of Cryptology and Information Security Gold Coast, Australia, December 9–13, 2001 Proceedings 7*, pages 36–51. Springer, 2001.
- BM05. J. Blömer and A. May. A tool kit for finding small roots of bivariate polynomials over the integers. In *Advances in Cryptology—EUROCRYPT 2005: 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Aarhus, Denmark, May 22–26, 2005. Proceedings 24*, pages 251–267. Springer, 2005.
- BV96. D. Boneh and R. Venkatesan. Hardness of computing the most significant bits of secret keys in Diffie-Hellman and related schemes. In *Annual International Cryptology Conference*, pages 129–142. Springer, 1996.
- BV97. M. Brion and M. Vergne. Lattice points in simple polytopes. *Journal of the American Mathematical Society*, pages 371–392, 1997.
- CD20. W. Castryck and T. Decru. Csidh on the surface. In *International Conference on Post-Quantum Cryptography*, pages 111–129. Springer, 2020.
- Cop96. D. Coppersmith. Finding a small root of a univariate modular equation. In *International Conference on the Theory and Applications of Cryptographic Techniques*, pages 155–165. Springer, 1996.
- Cop97. D. Coppersmith. Small solutions to polynomial equations, and low exponent RSA vulnerabilities. *Journal of cryptology*, 10(4):233–260, 1997.
- Cor04. J.-S. Coron. Finding small roots of bivariate integer polynomial equations revisited. In *Advances in Cryptology—EUROCRYPT 2004: International Conference on the Theory and Applications of Cryptographic Techniques, Interlaken, Switzerland, May 2–6, 2004. Proceedings 23*, pages 492–505. Springer, 2004.

- Cor07. J.-S. Coron. Finding small roots of bivariate integer polynomial equations: A direct approach. In *Advances in Cryptology-CRYPTO 2007: 27th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 19-23, 2007. Proceedings 27*, pages 379–394. Springer, 2007.
- DF88. M. E. Dyer and A. M. Frieze. On the complexity of computing the volume of a polyhedron. *SIAM Journal on Computing*, 17(5):967–974, 1988.
- DL05. J. A. De Loera. The many aspects of counting lattice points in polytopes. *Mathematische Semesterberichte*, 52(2):175–195, 2005.
- DMH20. G. De Micheli and N. Heninger. Recovering cryptographic keys from partial information, by example. *Cryptology ePrint Archive*, 2020.
- Ehr62. E. Ehrhart. Sur les polyèdres rationnels homothétiques à n dimensions. *CR Acad. Sci. Paris*, 254:616, 1962.
- FNP24. Y. Feng, A. Nitaj, and Y. Pan. Partial prime factor exposure attacks on some RSA variants. *Theoretical Computer Science*, 999:114549, 2024.
- GSW23. A. Granville, G. Shakan, and A. Walker. Effective results on the size and structure of sumsets. *Combinatorica*, 43(6):1139–1178, 2023.
- HG97. N. Howgrave-Graham. Finding small roots of univariate modular equations revisited. In *IMA International Conference on Cryptography and Coding*, pages 131–142. Springer, 1997.
- HG01. N. Howgrave-Graham. Approximate integer common divisors. In *International cryptography and lattices conference*, pages 51–66. Springer, 2001.
- HKN98. S.-P. Han, C. Kirfel, and M. B. Nathanson. Linear forms in finite sets of integers. *The Ramanujan Journal*, 2:271–281, 1998.
- HM10. M. Herrmann and A. May. Maximizing small root bounds by linearization and applications to small secret exponent rsa. In *Public Key Cryptography-PKC 2010: 13th International Conference on Practice and Theory in Public Key Cryptography, Paris, France, May 26-28, 2010. Proceedings 13*, pages 53–69. Springer, 2010.
- HR23. N. Heninger and K. Ryan. The hidden number problem with small unknown multipliers: Cryptanalyzing mega in six queries and other applications. In *IACR International Conference on Public-Key Cryptography*, pages 147–176. Springer, 2023.
- JM06. E. Jochemsz and A. May. A strategy for finding roots of multivariate polynomials with new applications in attacking RSA variants. In *Advances in Cryptology-ASIACRYPT 2006: 12th International Conference on the Theory and Application of Cryptology and Information Security, Shanghai, China, December 3-7, 2006. Proceedings 12*, pages 267–282. Springer, 2006.
- Kho92. A. G. Khovanskii. Newton polyhedron, Hilbert polynomial, and sums of finite sets. *Functional Analysis and Its Applications*, 26(4):276–281, 1992.
- LLL82. A. K. Lenstra, H. W. Lenstra, and L. Lovász. Factoring polynomials with rational coefficients. *Mathematische annalen*, 261:515–534, 1982.
- LV06. L. Lovász and S. Vempala. Simulated annealing in convex bodies and an $o^*(n^4)$ volume algorithm. *Journal of Computer and System Sciences*, 72(2):392–417, 2006.
- LZPL15. Y. Lu, R. Zhang, L. Peng, and D. Lin. Solving linear equations modulo unknown divisors: revisited. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 189–213. Springer, 2015.
- May02. A. May. Cryptanalysis of unbalanced RSA with small CRT-exponent. In *Annual International Cryptology Conference*, pages 242–256. Springer, 2002.
- May03. A. May. *New RSA vulnerabilities using lattice reduction methods*. PhD thesis, Citeseer, 2003.
- MN23. J. Meers and J. Nowakowski. Solving the Hidden Number Problem for CSIDH and CSURF via Automated Coppersmith. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 39–71. Springer, 2023.
- MNS22. A. May, J. Nowakowski, and S. Sarkar. Approximate divisor multiples–factoring with only a third of the secret CRT-exponents. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 147–167. Springer, 2022.
- Nat00. M. B. Nathanson. Growth of sumsets in abelian semigroups. *arXiv preprint math/0002091*, 2000.
- NR02. M. B. Nathanson and I. Z. Ruzsa. Polynomial growth of sumsets in abelian semigroups. *Journal de théorie des nombres de Bordeaux*, 14(2):553–560, 2002.
- Rya24. K. Ryan. Solving multivariate coppersmith problems with known moduli. *Cryptology ePrint Archive*, 2024.
- Sar24. S. Sarkar. Enhanced bound for the commutative isogeny hidden number problem in csurf. In *International Conference on Cryptology in India*, pages 201–211. Springer, 2024.
- TLP17. A. Takayasu, Y. Lu, and L. Peng. Small CRT-exponent RSA revisited. In *Annual International Conference on the Theory and Applications of Cryptographic Techniques*, pages 130–159. Springer, 2017.
- Wie90. M. J. Wiener. Cryptanalysis of short rsa secret exponents. *IEEE Transactions on Information theory*, 36(3):553–558, 1990.

A Counterexample

For a polynomial f with k variables, Meers and Nowakowski [MN23] proposed a new heuristic that $p_{\mathcal{F}}(m)$ is a degree- $(k+1)$ polynomial, which fails in general. Moreover, their assertion that $m \dim(\mathcal{L})$ and the exponents p_1, \dots, p_k are degree- $(k+1)$ polynomials in m suffers from the same fundamental flaw. Since Sarkar's strategy [Sar24] generalizes the Meers–Nowakowski approach, any counterexample to their method also invalidates Sarkar's heuristic. Hence we focus on the counterexample in the Meers–Nowakowski approach.

Counterexample. Their central claim – that $m \dim(\mathcal{L})$ forms a degree- $k+1$ polynomial in m for k -variable polynomials – fails for small m . Consider $f = x^5 + x + 1$ where $\dim(\mathcal{L})$ (counting monomials in f^m) produces the sequence:

$$\{3, 12, 30, 60, 100, 150, \dots\} \quad \text{for } m = 1, 2, \dots$$

While $\dim(\mathcal{L}) = 5m^2 - 5m$ holds for $m \geq 3$, it fails completely for $m < 3$. This demonstrates the necessity of the **must condition**, i.e., large enough m . Similar issues affect $p_j(m)$ computations for $j = 1, \dots, k$.

Heuristic Validation Failure. Their proposed heuristic that $p_{\mathcal{F}}(m)$ is a degree- $(k+1)$ polynomial also requires correction. For $f = x^5 + x + 1$, the $p_{\mathcal{F}}(m)$ sequence:

$$\{2, 8, 20, 40, 65, 95, \dots\}$$

yields inconsistent interpolations:

$$\begin{aligned} m = 1, 2, 3: \quad & 3m^2 - 3m + 2 \\ m = 2, 3, 4: \quad & 4m^2 - 8m + 8 \\ m = 3, 4, 5: \quad & \frac{5}{2}m^2 + \frac{5}{2}m - 10 \end{aligned}$$

Only our Newton polytope analysis reveals the true asymptotic behavior:

$$\lim_{m \rightarrow \infty} \frac{p_{\mathcal{F}}(m)}{m^2} = \frac{k}{k+1} \cdot V(N(f)) = \frac{5}{2}.$$

Moreover, Additive combinatorics results [GSW23] reveal fundamental limitations: for 4-variable polynomials $f = x_1^3 + x_1x_2 + x_1x_3 + x_2 + x_3^2x_4^2 + x_4^5 + 1$, the required threshold becomes:

$$m > (2 \times 7 \times 5)^{8 \times 7} \approx 2^{343}.$$

This renders practical verification infeasible. While early termination at "stable" interpolants seems plausible, our local convergence example demonstrates pitfalls:

A.1 Local Convergence Phenomenon

Consider $f = x_1^3 + x_1x_2 + x_1x_3 + x_2 + x_3^2x_4^2 + x_4^5 + 1$ and track $p_{\mathcal{M}}(m) = m \dim(\mathcal{L})$ reveals:

Early termination (left of dashed line) suggests incorrect convergence to $25/12$, while full computation reveals the true limit 2. This highlights the risks of heuristic interpolation without theoretical guarantees and underscores our approach's advantages.

B A Toy Example (Small Private Exponent RSA Problem)

Here, we use the Small Private Exponent RSA Problem to illustrate our formulas. Moreover, in cryptanalysis, certain problems require specialized techniques to achieve better results, as seen in the case of the Small Private Exponent RSA Problem [BD00, HM10]. Next, we take this problem as an example to demonstrate how to integrate these specialized techniques with our strategy.

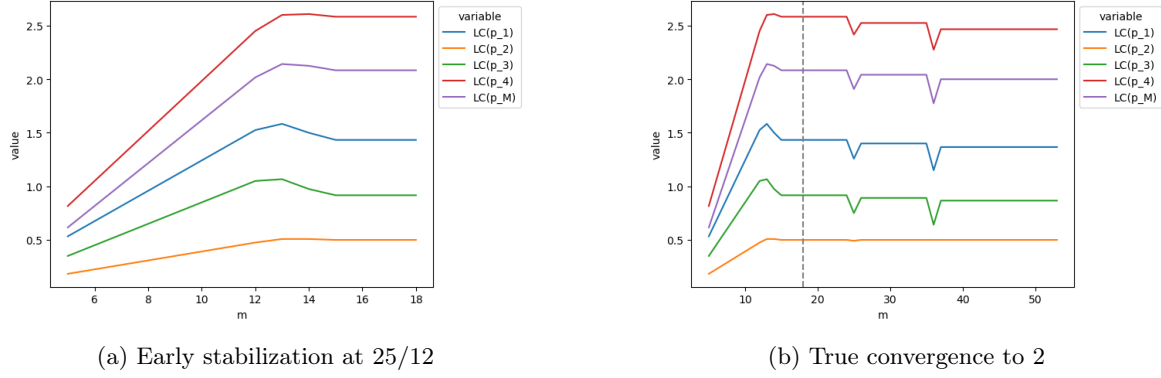


Fig. 2: Local vs global convergence in leading coefficient estimation (gray dashed line marks transition)

Definition 47 (Small Private Exponent RSA Problem). *Given an integer $N = pq$ with unknown factorization and $e \approx N$ satisfying $\gcd(e, (p-1)(q-1)) = 1$. The goal is to compute d such that*

$$ed \equiv 1 \pmod{(p-1)(q-1)}, \quad (25)$$

which can be reduced to solve the following equation:

$$f(x_1, x_2) = x_1(N+1+x_2) + 1 \equiv 0 \pmod{e}, \quad (26)$$

with desired root $(\frac{ed-1}{(p-1)(q-1)}, -p-q)$.

Previous results. Wiener [Wie90] showed that Equation (26) can be solved when $d < e^{0.25}$, which was later improved to $e^{0.284}$ and $e^{0.292}$ by Boneh and Durfee [BD00]. Their key idea was using Coppersmith's method to construct the lattice \mathcal{L} by leveraging the structure of Equation (26) and estimating the sublattice determinant, achieving the best result of $e^{0.292}$ with the *Geometrically Progressive Matrices* method. Herrmann and May [HM10] later attained the same bound using *Unravelled Linearization*.

B.1 Manual Calculation vs. Our Formulas in the Jochemsz-May Basic Strategy

Manual Calculation. We have

$$\left\{ \begin{array}{l} \dim(\mathcal{L}) = \sum_{i=0}^m \sum_{j=0}^i 1 = \frac{1}{2} m^2 + o(m^2), \\ p_1 = \sum_{i=0}^m \sum_{j=0}^i i = \frac{1}{3} m^3 + o(m^3), \\ p_2 = \sum_{i=0}^m \sum_{j=0}^i j = \frac{1}{6} m^3 + o(m^3), \\ p_{\mathcal{F}} = \sum_{i=0}^m \sum_{j=0}^i m - j = \frac{1}{3} m^3 + o(m^3). \end{array} \right.$$

Using our formulas. For computing $\dim(\mathcal{L})$, we have

$$\dim(\mathcal{L}) = \frac{\int_{N(f)} 1 dV}{\text{ind } A(f)} m^2 + o(m^2).$$

As $A(f) = \{(0,0), (0,1), (1,1)\}$, then $N(f)$ is a triangle with area $1/2$. And the lattice generated by $A(f)$ is \mathbb{Z}^2 , i.e., the area of the fundamental parallelepiped is 1. See Figure 3. Hence we have

$$\dim(\mathcal{L}) = \sum_{i=0}^m \sum_{j=0}^i 1 = \frac{1}{2} m^2 + o(m^2).$$



Fig. 3: $N(f)$ and \mathcal{P} : we can see that $N(f)$ is a triangle and the fundamental parallelepiped is a unit square.

For computing p_1 and p_2 , we consider the volume of the convex hull of $\mathcal{H}_1(A(f))$ and $\mathcal{H}_2(A(f))$, and we obtain the same result as the manual calculation.

Now we have

$$X_1^{\frac{1}{3}} X_2^{\frac{1}{6}} < M^{\frac{1}{6}},$$

substituting $X_1 = d$ and $X_2 = e^{1/2}$, reorganizing gives

$$\log_e d < 1/4,$$

which means we can break RSA using Coppersmith's method when $d < e^{1/4}$.

B.2 Combined with Some Specialized Techniques

If we directly use the formulas in Section 3, i.e., formulas for the Jochemsz-May Extended Strategy, we obtain

$$\begin{aligned} \dim(\mathcal{L}) &= \frac{1}{2}m^2 + mt, \\ \det(\mathcal{L}) &= X_1^{\frac{1}{3}m^3 + \frac{1}{2}tm^2} X_2^{\frac{1}{6}m^3 + \frac{1}{2}tm^2 + \frac{1}{2}t^2m} e^{\frac{1}{3}m^3 + \frac{1}{2}tm^2}. \end{aligned}$$

Substituting $t = \frac{1-2\log_e d}{2}m$, which yields $\log_e d < \frac{7}{6} - \frac{1}{3}\sqrt{7} \approx 0.284$. That is to say, when $d < e^{0.284}$, we can break RSA using Coppersmith's method.

Combined with Sublattice. Boneh and Durfee [BD00] introduced the definition of *geometrically progressive matrix* (see Definition 5.1 in [BD00]) and provided a property as follows:

Lemma 48. Suppose M be a $(a+1)b \times (a+1)b$ geometrically progressive matrix with parameters C , and let B be a real number. Define

$$S_B := \{(k, \ell) \in \{0, \dots, a\} \times \{1, \dots, b\} \mid M(k, \ell, k, \ell) \leq B\} \quad (27)$$

and set $w := |S_B|$. If \mathcal{L}_G is the lattice defined by the rows $(k, \ell) \in S_B$ of M , then

$$\det(\mathcal{L}_G) \leq ((a+1)b)^{w/2} (1+C)^{w^2} \prod_{(k, \ell) \in S_B} M(k, \ell, k, \ell). \quad (28)$$

They chose $B = e^m$ and selected the part of the original lattice L that is less than e^m as a sublattice like Equation (27), and used Lemma 48 to estimate the determinant of the sublattice. Therefore, they need to compute the right-hand side of the inequality in Equation (28).

This approach corresponds to taking a partial section of the Newton polytope. For example, in terms of lattice dimension (illustrated in Figure 4), the area of the left trapezoid is $\frac{1}{2}m^2 + mt$, representing the lattice dimension with the Jochemsz-May Extended Strategy. On the right, only part of the trapezoid is used, reducing the area to $\frac{1}{2}m^2 + \frac{1}{2}mt$.

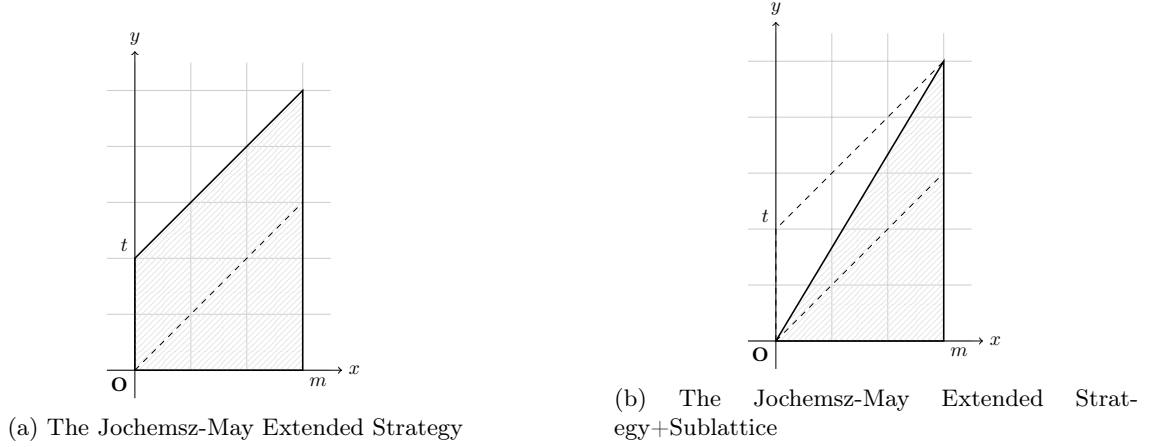


Fig. 4: Comparison of the Lattice Dimension with or without the Sublattice Approach.

Now for $j = 1, 2$, p_j in the sublattice setting equals the integral of x_j over the new Newton polytope, as shown in Figure 4b. Similarly, p_F is derived in this manner, resulting in the following:

$$\dim(\mathcal{L}_G) = \frac{1}{2}m^2 + \frac{1}{2}mt,$$

$$\det(\mathcal{L}_G) = X_1^{\frac{1}{3}m^3 + \frac{1}{3}tm^2} X_2^{\frac{1}{6}m^3 + \frac{1}{3}tm^2 + \frac{1}{6}t^2m} e^{\frac{1}{3}m^3 + \frac{1}{6}tm^2}.$$

Substituting $\det(\mathcal{L}_G) < e^{m \dim(\mathcal{L}_G)}$ and $t = (1 - 2 \log_e d)m$, which yields $\log_e d < 1 - \frac{\sqrt{2}}{2} \approx 0.292$.

Combined with Unravalled Linearization. In 2009, Herrmann and May [HM10] proposed *Unravalled Linearization*. They grouped terms in $f(x_1, x_2) = x_1x_2 + (N+1)x_1 + 1 \equiv 0 \pmod{e}$ by setting $u = x_1x_2 + 1$, which led to the relation $f'(x_1, x_2, u) = u + (N+1)x_1 \equiv 0 \pmod{e}$. They substituted $x_1x_2 \rightarrow u - 1$ back when they generated the lattice to eliminate terms with x_1x_2 . This decreased the lattice determinant, enhancing the attack. Another way to describe this is through the ideal $\langle u - x_1x_2 - 1 \rangle$, which is commonly encountered in code implementations. Recently, Ryan [Rya24] also pointed this out.

Unravalled Linearization essentially involves taking a partial section of the Newton polytope, see Figure 4b. Therefore, the calculations for the lattice dimension and its determinant align with those in the Jochemsz-May Extended Strategy with the Sublattice approach.

C Omitted Proofs from Section 2

C.1 Proof of Lemma 23

Lemma 23 (Summation of $\sum_{\lambda \in mA(f)} x_j(\lambda)$). *For any well-defined k -variate polynomial f , the sum $\sum_{(i_1, \dots, i_k) \in mA(f)} i_j$ is polynomial in m with degree $k+1$ when m is large enough. Specifically, the leading coefficient is $\int_{N(f)} x_j dV$ divided by $\text{ind } A(f)$. That is,*

$$\sum_{(i_1, \dots, i_k) \in mA(f)} i_j = \frac{\int_{N(f)} x_j dV}{\text{ind } A(f)} m^{k+1} + o(m^{k+1}),$$

where $x_j : \mathbb{R}^k \rightarrow \mathbb{R}$ acts as $x_j((i_1, \dots, i_k)) = i_j$.

Proof. Inspired by the geometric meaning of integrals, we first add a dimension and transfer the information of the integrand x_j to this new dimension. This gives a mapping from a family of subsets of \mathbb{R}^k to a family of subsets of \mathbb{R}^{k+1} :

$$\mathcal{K}_j(N) = \{(v_1, \dots, v_{k+1}) \in \mathbb{R}^{k+1} \mid (v_1, \dots, v_k) \in N, 0 \leq v_{k+1} \leq v_j\}.$$

The corresponding discrete version is:

$$\mathcal{H}_j(A) = \bigcup_{(i_1, \dots, i_k) \in A} \{(i_1, \dots, i_k, 0), (i_1, \dots, i_k, 1), \dots, (i_1, \dots, i_k, i_j)\}.$$

Both mappings are additive in the sense of set addition, i.e., $\mathcal{K}_j(N_1 + N_2) = \mathcal{K}_j(N_1) + \mathcal{K}_j(N_2)$ and $\mathcal{H}_j(A_1 + A_2) = \mathcal{H}_j(A_1) + \mathcal{H}_j(A_2)$. This formulation allows us to derive the identity

$$\sum_{(i_1, \dots, i_k) \in mA(f)} i_j = |\mathcal{H}_j(mA(f))| - |mA(f)| = |m\mathcal{H}_j(A(f))| - |mA(f)|.$$

For sufficiently large m , both terms are polynomials in m , implying that their difference is a polynomial of degree $k + 1$.

Now, consider its leading coefficient. Since the lattice generated by $\mathcal{H}_j(A(f))$ can be viewed as the direct sum of the lattice generated by $A(f)$ and \mathbb{Z} , the index of $\mathcal{H}_j(A(f))$ as a sublattice of \mathbb{Z}^{k+1} equals $\text{ind } A(f)$, i.e.,

$$\text{ind } \mathcal{H}_j(A(f)) = \text{ind } A(f).$$

It remains to prove that

$$\int_{N(\mathcal{H}_j(A(f)))} 1 \, dV = \int_{N(f)} x_j \, dV.$$

Since

$$\int_{N(f)} x_j \, dV = \int_{\mathcal{K}_j(N(f))} 1 \, dV,$$

it suffices to show that

$$N(\mathcal{H}_j(A(f))) = \mathcal{K}_j(N(f)).$$

Firstly, note that \mathcal{K}_j maps convex sets to convex sets (using the convexity of x_j) and that $\mathcal{H}_j(A(f)) \subseteq \mathcal{K}_j(N(f))$. Thus,

$$N(\mathcal{H}_j(A(f))) \subseteq \mathcal{K}_j(N(f)).$$

For the reverse inclusion, take any $v = (v' \mid v_{k+1}) \in \mathcal{K}_j(N(f))$, where $v' \in N(f)$. Then, we have

$$v' = \sum_{\delta \in A(f)} \lambda_\delta \delta, \quad \lambda_\delta \in [0, 1], \quad \sum_{\delta \in A(f)} \lambda_\delta = 1.$$

By definition,

$$0 \leq v_{k+1} \leq \sum_{\delta \in A(f)} \lambda_\delta x_j(\delta).$$

Additionally, observe that

$$\sum_{\delta \in A(f)} \lambda_\delta (\delta \mid x_j(\delta)) = (v' \mid \sum_{\delta \in A(f)} \lambda_\delta x_j(\delta)) \in N(\mathcal{H}_j(A(f)))$$

and

$$\sum_{\delta \in A(f)} \lambda_\delta (\delta \mid 0) = (v' \mid 0) \in N(\mathcal{H}_j(A(f))).$$

By convexity,

$$\begin{aligned} v = (v' \mid v_{k+1}) &= \frac{v_{k+1}}{\sum_{\delta \in A(f)} \lambda_\delta x_j(\delta)} \left(v' \mid \sum_{\delta \in A(f)} \lambda_\delta x_j(\delta) \right) \\ &\quad + \left(1 - \frac{v_{k+1}}{\sum_{\delta \in A(f)} \lambda_\delta x_j(\delta)} \right) (v' \mid 0) \\ &\in N(\mathcal{H}_j(A(f))). \end{aligned}$$

Thus,

$$\mathcal{K}_j(N(f)) \subseteq N(\mathcal{H}_j(A(f))).$$

□

C.2 Proof of Lemma 24

Lemma 24 (Summation of $\sum_{\lambda \in m_1 A(f) + m_2 E} x_j(\lambda)$). *For any well-defined k -variate polynomials f , the summation*

$$\sum_{(i_1, \dots, i_k) \in m_1 A(f) + m_2 E} i_j$$

is a polynomial in m_1 and m_2 of degree $k+1$ when m_1 and m_2 are sufficiently large. That is,

$$\sum_{\lambda \in m_1 A(f) + m_2 E} x_j(\lambda) = \underbrace{b_0^{(j)} m_1^{k+1} + b_1^{(j)} m_1^k m_2 + \dots + b_{k+1}^{(j)} m_2^{k+1}}_{\text{all monomials with degree } k+1} + \dots$$

Besides, for all $\delta \in \mathbb{N}$, $b_i^{(j)}$ satisfy

$$\sum_{i=0}^{k+1} b_i^{(j)} \delta^i = \frac{\int_{N(A(f)+\delta E)} x_j dV}{\text{ind } A(f) + \delta E}$$

where $x_j : \mathbb{R}^k \rightarrow \mathbb{R}$ acts as $x_j((i_1, \dots, i_k)) = i_j$.

Proof. The idea is also use \mathcal{H}_j to rewrite $\sum_{(i_1, \dots, i_k) \in m_1 A(f) + m_2 E} i_j$ as

$$\begin{aligned} \sum_{(i_1, \dots, i_k) \in m_1 A(f) + m_2 E} i_j &= |\mathcal{H}_j(m_1 A(f) + m_2 E)| - |m_1 A(f) + m_2 E| \\ &= |m_1 \mathcal{H}_j(A(f)) + m_2 \mathcal{H}_j(E)| - |m_1 A(f) + m_2 E|. \end{aligned}$$

Similarly, $\sum_{(i_1, \dots, i_k) \in m_1 A(f) + m_2 E} i_j$ is a polynomial of m_1, m_2 for sufficiently large m_1, m_2 by Lemma 20. That is, there exist $b_i^{(j)}$ for $i = 0, \dots, k+1$, such that:

$$\sum_{\lambda \in m_1 A(f) + m_2 E} x_j(\lambda) = \underbrace{b_0^{(j)} m_1^{k+1} + b_1^{(j)} m_1^k m_2 + \dots + b_{k+1}^{(j)} m_2^{k+1}}_{\text{all monomials with degree } k+1} + \dots$$

When substituting $m_2 = \delta m_1$ for $\delta = 0, \dots, k+1$, we have

$$\begin{aligned} \sum_{(i_1, \dots, i_k) \in m_1 A(f) + m_2 E} i_j &= |m_1 \mathcal{H}_j(A(f)) + m_2 \mathcal{H}_j(E)| - |m_1 A(f) + m_2 E| \\ &= |m_1 (\mathcal{H}_j(A(f)) + \delta \mathcal{H}_j(E))| - |m_1 (A(f) + \delta E)| \end{aligned}$$

Comparing the coefficients of m_1^{k+1} on both sides, we obtain

$$\sum_{i=0}^{k+1} b_i^{(j)} \delta^i = \frac{\int_{N(A(f)+\delta E)} x_j dV}{\text{ind } A(f) + \delta E}$$

from which can compute all $b_i^{(j)}$ for $i = 0, \dots, k+1$. □

D Details for f in Section 5

We also provide detailed information about the polynomials used in our experiments in Table 3.

	$\text{supp}(f)$
Exp. 1	$\text{supp}((x_1 * x_2 + x_1 + x_2 + 1) * (x_2 * x_3 + x_2 + x_3 + 1) * (x_1 * x_3 + x_1 + x_3 + 1))$
Exp. 2	$\text{supp}(x_1^3 + x_1 * x_2 + x_1 * x_3^2 + x_2^2 * x_3^3 + x_2^2 + x_2 + 2)$
Exp. 3	$\text{supp}((x_3^3 * x_2^2 + x_2^2 + x_1 * x_2 + 1) * (x_3^2 * x_1 + x_1^3 + x_2 + 1))$
Exp. 4	$\text{supp}((x_2^2 + x_1^3 * x_2 + 1) * (x_1^2 + x_2 * x_3 + x_3^4 + 1))$
Exp. 5	$\text{supp}((x_2^2 + x_1^3 * x_2 + x_2^2 * x_4^3 + 1) * (x_1^2 + x_2 * x_3 + x_3^4 + 1))$
Exp. 6	$\text{supp}((x_2^2 + x_1^3 * x_2 + x_2^2 * x_4^5 + 1) * (x_1^2 + x_2 * x_3 + x_3^4 + 1))$
Exp. 7	$\text{supp}((x_2^2 + x_1^3 * x_2 + x_2^2 * x_4^3 + 1) * (x_1^2 + x_2 * x_3 + x_3^2 * x_4 + 1))$
Exp. 8	$\text{supp}((x_2^2 + x_1^3 * x_2 + x_2^2 * x_4^5 + 1) * (x_1^2 + x_2 * x_3 + x_3^2 * x_4 + 1))$

Table 3: Details of f in Table 2.