A New Cryptanalytic Attack on an RSA Variant Arising from Cubic Pell Curves

Brahim Chnioune[a], Mohammed Rahmani[a,*], Abderrahmane Nitaj[b], Mhammed Ziane[a]

[a] *Mohammed I University, Faculty of Sciences, Department of Mathematics, ACSA Laboratory, Oujda, 60000, Morocco*
[b] *Normandie Univ, UNICAEN, CNRS, LMNO, Caen, 14000, France*

Abstract

In 2024, Nitaj and Seck introduced an RSA-like cryptosystem defined by a public key (N, e) and a corresponding private key (N, d) satisfying $ed-1 \equiv 0 \pmod{p^{2(a-1)}q^{2(b-1)}(p-1)^2(q-1)^2}$, where $N = p^a q^b$. More recently, Seck and Tchapgnouo (DAM 2026) proposed a partial key exposure attack against the same scheme for the case where $N = pq$, assuming partial knowledge of one prime factor. Using a lattice-based method, they showed that, under some constraints, the private parameters (d, p, q) can be recovered in polynomial time. In this paper, we further investigate the security of this construction by considering a dual-key scenario. Specifically, in the presence of two encryption exponents e_1 and e_2, with corresponding private exponents d_1 and d_2 satisfying $e_i d_i - 1 \equiv 0 \pmod{(p-1)^2(q-1)^2}$, for $i \in \{1, 2\}$, we prove that if the prime factors of N share a certain number of least significant bits, and if d_1 and d_2 share an amount of their most significant bits, then the private parameters (d_1, d_2, p, q) can be computed in polynomial time. Our method combines an extension of Coppersmith's small-root technique with lattice basis reduction, leading to improved bounds compared to existing single-key attacks.

Keywords:

*Corresponding author
Email addresses: brahim.chnioune.d24@ump.ac.ma (Brahim Chnioune), mohammed.rahmani@ump.ac.ma (Mohammed Rahmani), abderrahmane.nitaj@unicaen.fr (Abderrahmane Nitaj), m.ziane@ump.ac.ma (Mhammed Ziane)

RSA, Cubic Pell equation, Integer factorization, Coppersmith's method, LLL reduction algorithm

1. Introduction

Proposed by Rivest, Shamir, and Adleman in 1978 [21], the RSA cryptosystem is a foundational construction in public-key cryptography, whose security is rooted in the computational hardness of factoring large integers. The scheme is initialized by selecting two large primes p and q of comparable size and forming the modulus $N = pq$. A public exponent e is chosen such that it is invertible modulo $\varphi(N) = (p-1)(q-1)$. The corresponding private exponent d is then defined through the congruence $ed \equiv 1 \pmod{\varphi(N)}$, or equivalently via the relation $ed - k\varphi(N) = 1$. The pair (N, e) is made public, while d is kept secret. Encryption and decryption are performed through modular exponentiation, namely $c \equiv m^e \pmod{N}$ and $m \equiv c^d \pmod{N}$. These operations run in polynomial time, with complexities on the order of $O\left((\log N)^2 \log e\right)$ and $O\left((\log N)^2 \log d\right)$, respectively.

While choosing a small private exponent can accelerate RSA decryption, it introduces significant security risks. In 1990, Wiener [26] proved that RSA is insecure whenever $d < \frac{1}{3}N^{1/4}$. A major breakthrough came in 1996 with Coppersmith's method [4], which provides a polynomial-time algorithm for finding small roots of modular polynomial equations $f(x) \equiv 0 \pmod{M}$, even when the factorization of M is unknown. Building on these ideas, Boneh and Durfee [1] reformulated the key equation $ed - k\varphi(N) = 1$ into the congruence $xy - Nx - 1 \equiv 0 \pmod{e}$. Using Coppersmith's approach, they successfully recovered the solution $(x, y) = (k, p + q - 1)$, leading to a polynomial-time factorization of N and to improve Wiener's bound up to $d < N^{0.292}$.

To improve both the security and computational performance of RSA, several alternative constructions have been introduced. In some of these approaches, the modulus remains $N = pq$, while the classical Euler totient function $\varphi(N) = (p-1)(q-1)$ is replaced by other arithmetic functions. For instance, certain variants use $\psi(N) = (p^2-1)(q^2-1)$, as proposed in [11, 6, 2, 22]. A particular RSA variant was introduced by Murru and Saettone [15]. They proposed a modified RSA scheme based on a new operation $\odot$, derived from the cubic Pell equation

$$\mathcal{K}_\varsigma(N) : z_1^3 + \varsigma z_2^3 + \varsigma^2 z_3^3 - 3\varsigma z_1 z_2 z_3 \equiv 1 \pmod{N},$$

for some integer ς. The public exponent e is selected to satisfy $\gcd(e, \psi(N)) = 1$, where $\psi(N) = (p^2+p+1)(q^2+q+1)$. The corresponding private exponent d is defined by the congruence $ed \equiv 1 \pmod{\psi(N)}$. In this setting, encryption maps a message (m_1, m_2) to $(c_1, c_2) = (m_1, m_2)^{\odot e} \pmod{N}$, while decryption recovers the plaintext through $(m_1, m_2) = (c_1, c_2)^{\odot d} \pmod{N}$. Later, several cryptanalytic attacks have been proposed against the Murru–Saettone scheme, as shown in [27, 19].

A recent contribution by Nitaj and Seck [18] put forward an RSA-type construction that leverages encoding techniques in conjunction with a cubic Pell curve given by

$$\mathcal{K}_\varsigma(N) : z_1^3 + \varsigma z_2^3 + \varsigma^2 z_3^3 - 3\varsigma z_1 z_2 z_3 \equiv 1 \pmod{N}.$$

Here, the modulus N is defined as $N = p^a q^b$, and the associated exponents (e, d) are required to fulfill

$$ed - 1 \equiv 0 \pmod{p^{2(a-1)} q^{2(b-1)} (p-1)^2 (q-1)^2}.$$

In addition to the well-known attacks by Wiener and Boneh-Durfee, numerous other attacks have been developed targeting RSA and its variants. In 2010, Sarkar et al. [23] demonstrated that the RSA scheme becomes vulnerable when two public keys (N, e_1) and (N, e_2) are constructed using the same modulus N, with public exponents of magnitude comparable to N. More precisely, they established that if the associated private exponents d_1 and d_2, both bounded by N^δ, share a sufficient number of most significant bits (MSBs), then the modulus N can be factored in polynomial time provided that $|d_1 - d_2| < N^\varrho$ and $\delta < \frac{5}{8} - \frac{\varrho}{2}$. Subsequently, Nitaj et al. [16] refined this result. They showed that when the public exponents satisfy $e_1, e_2 \approx N^\alpha$ and the prime factors share ε least significant bits (LSBs) such that $2^\varepsilon = N^\omega$, the previous bound can be strengthened to

$$\delta < \frac{5}{2} - 2\omega - \varrho - \frac{1}{4}\sqrt{6(1-4\omega)(5+4\alpha-4\omega-4\varrho)}.$$

The use of two or more pairs of keys (e_i, d_i) with a common RSA modulus N is convenient for key management, especially in devices with limited computational capabilities. Moreover, such a setting can arise when the same RSA modulus is used by different entities. A typical example is when a server uses a single RSA modulus N, and distributes several pairs (e_i, d_i) to different users. This facilitates the computation on the server. More reasons

to use a common RSA modulus are studied in [14]. The scenario of using RSA with a common modulus and different pairs of keys has been analyzed in several works [25, 5, 9, 7, 23].

In 2024, Nitaj and Seck [18] introduced an attack on the variant based on the cubic Pell equation with a modulus of the form $N = p^a q^b$. They showed that the modulus can be factored in polynomial time whenever

$$d < N^{2-\frac{2(3a+b)}{(a+b)^2}}.$$

It is worth noting that the Nitaj–Seck attack leads to a weaker bound in the standard RSA configuration $N = pq$ (i.e., $a = b = 1$), since the bound approaches zero. This drawback was later addressed in 2025 by Rahmani and Nitaj [20], who improved the analysis using Coppersmith's lattice-based technique. Under the assumption $N = pq$, they proved that the scheme can be broken whenever $N < e < N^4$ and $d < N^{2-\sqrt{2}}$. More recently, Seck and Tchapgnouo [24] extended this attack to scenarios where an approximation of one of the RSA primes is known, subject to some additional constraints. Subsequently, Chnioune et al. [3] presented an attack on the same scheme. They showed that under the extra information that the RSA primes share an amount of their least significant bits, the factorization of N can be carried out in polynomial time under some constraints.

1.1. Our Contribution

In this work, we propose a novel lattice-based cryptanalytic technique for an RSA-type scheme based on the cubic Pell curve

$$\mathcal{K}_\varsigma(N) : z_1^3 + \varsigma z_2^3 + \varsigma^2 z_3^3 - 3\varsigma z_1 z_2 z_3 \equiv 1 \pmod{N},$$

involving two public keys (N, e_1) and (N, e_2) over a common modulus $N = pq$, with corresponding private keys (N, d_1) and (N, d_2). The method is effective when the prime factors of N share a subset of their least significant bits and the secret exponents d_1 and d_2 share a subset of their most significant bits.

We start with the fundamental relations $e_i d_i - k_i(p-1)^2(q-1)^2 = 1$ for $i = 1, 2$, from which we build a multivariate polynomial equation $h(s_1, s_2, s_3, s_4) = 0$. The small integer solutions of this polynomial are directly linked to the unknown secret values. By leveraging Coppersmith's framework for solving multivariate polynomial equations with small roots, we recover both the private exponents and the prime factors, i.e., (d_1, d_2, p, q).

More specifically, we establish that under the assumptions that the RSA primes share ε least significant bits with $2^{\varepsilon} = N^{\omega}$, that the secret exponents satisfy $|d_1 - d_2| < N^{\varrho}$ and $d_1, d_2 \leq N^{\delta}$, and that the public exponents satisfy $e_1, e_2 \approx N^{\lambda}$, the vulnerability of the scheme occurs under the following conditions

$$\varrho + 2\omega \leq 2.5 + \lambda,$$

and

$$\delta < 5 - 4\omega - \varrho - \frac{1}{2}\sqrt{6(5 - 4\omega + 2\lambda - 2\varrho)(1 - 4\omega)}.$$

In order to validate the proposed attack, we present a detailed numerical example and provide an implementation of the algorithm in SageMath. The source code is publicly available at:

`https://github.com/algebre96-hue/Cubic_RSA_attack`

1.2. The Organization of the Paper

The remainder of this paper is structured as follows. In Section 2, we present the necessary background and outline the principal tools used throughout this work. Section 3 describes the adaptation of Coppersmith's method to the cryptanalysis of the Nitaj–Seck RSA variant. In Section 4, we demonstrate the effectiveness of the attack through a detailed numerical example, and Section 5 concludes the paper.

2. Mathematical Background

This section is devoted to presenting auxiliary lemmas that play a key role in establishing our main result. We further recall basic concepts such as lattice basis reduction methods and the extended Coppersmith technique. We conclude by describing the main scheme that is vulnerable to our proposed attack.

2.1. Auxiliary Lemmas

Based on the framework of [17], the following lemma provides explicit bounds on the RSA prime factors under the balanced-prime assumption.

Lemma 1. *For an RSA modulus $N = pq$ generated from primes of comparable magnitude, namely $2q > p > q$, the prime factors satisfy the following bounds:*

$$\sqrt{2N} > p > \sqrt{N} > q > \sqrt{\frac{N}{2}}.$$

In addition, we have $3\sqrt{N} > p + q$.

The lemma below, due to Coppersmith [4], provides a key result for RSA cryptanalysis and is commonly used to recover the factorization of an RSA modulus when partial information about one of its prime factors is known.

Lemma 2 (Coppersmith)**.** *Let* $N = pq$ *be an RSA modulus with* $q < p < 2q$. *Suppose that either the most significant half or the least significant half of the binary representation of one of the factors* p *or* q *is known. Then the prime factorization of* N *can be recovered in polynomial time.*

For $p - q = 2^{\varepsilon}x$ with unknown $x \in \mathbb{Z}$ and known $\varepsilon > 0$, the ε LSBs of p and q and the 2ε LSBs of $p + q$ can be obtained (cf. [16]). In this setting, we obtain the following result.

Lemma 3. *Let* $p - q = 2^{\varepsilon}x$ *and let* u_0 *satisfy* $y^2 \equiv N \pmod{2^{\varepsilon}}$. *Define*

$$v_0 \equiv 2u_0 + (N - u_0^2)\, u_0^{-1} \pmod{2^{2\varepsilon}}.$$

Then there exist integers p_0, q_0, *and* v *such that*

$$p = 2^{\varepsilon}p_0 + u_0, \quad q = 2^{\varepsilon}q_0 + u_0, \quad p + q = 2^{2\varepsilon}v + v_0.$$

The following lemma will be used in our main result.

Lemma 4. *Let* $n \in \mathbb{N}$. *Consider the polynomial* h *in four variables* (s_1, s_2, s_3, s_4) *defined by:*

$$h(s_1, s_2, s_3, s_4) = a_1 s_1 - a_2 s_2 P(s_4) + a_3 s_3 P(s_4) + a_0,$$

where $P(s_4) = \sum_{k=0}^{2} c_k s_4^k$ *is a quadratic polynomial with integer coefficients. The* n*-th power of* h *can be expressed as the following multi-summation:*

$$h^n(s_1, s_2, s_3, s_4) = \sum_{i_1=0}^{n} \sum_{i_2=0}^{n-i_1} \sum_{i_3=0}^{n-i_1-i_2} \sum_{i_4=0}^{2(i_2+i_3)} \mathcal{A}_{i_1,i_2,i_3,i_4} s_1^{i_1} s_2^{i_2} s_3^{i_3} s_4^{i_4},$$

where the coefficients $\mathcal{A}_{i_1,i_2,i_3,i_4}$ *are given by:*

$$\mathcal{A}_{i_1,i_2,i_3,i_4} = \binom{n}{i_1, i_2, i_3, n - i_1 - i_2 - i_3} (-1)^{i_2} a_0^{n-i_1-i_2-i_3} a_1^{i_1} a_2^{i_2} a_3^{i_3} a_{4,i_4}^{(i_2+i_3)},$$

and $a_{4,i_4}^{(m)}$ *denotes the coefficient of* $s_4^{i_4}$ *in the expansion of* $[P(s_4)]^m$.

Proof. To derive the expansion of h^n, we consider the function h as a multinomial expression in four variables (s_1, s_2, s_3, s_4). Let h be defined as:

$$h(s_1, s_2, s_3, s_4) = a_1 s_1 - a_2 s_2 P(s_4) + a_3 s_3 P(s_4) + a_0.$$

Applying the multinomial theorem for a power $n \in \mathbb{N}$, the expansion is given by:

$$h^n = \sum_{i_1+i_2+i_3+j=n} \frac{n!}{i_1! i_2! i_3! j!} (a_1 s_1)^{i_1} (-a_2 s_2 P(s_4))^{i_2} (a_3 s_3 P(s_4))^{i_3} a_0^j.$$

By substituting the index $j = n - i_1 - i_2 - i_3$ and introducing the multinomial coefficient notation $\binom{n}{i_1, i_2, i_3, j}$, we rearrange the terms as follows:

$$h^n = \sum_{i_1=0}^{n} \sum_{i_2=0}^{n-i_1} \sum_{i_3=0}^{n-i_1-i_2} \binom{n}{i_1, i_2, i_3, n-i_1-i_2-i_3} (-1)^{i_2} a_0^{n-i_1-i_2-i_3} a_1^{i_1} a_2^{i_2} a_3^{i_3} \times s_1^{i_1} s_2^{i_2} s_3^{i_3} [P(s_4)]^{i_2+i_3}.$$

The term $[P(s_4)]^{i_2+i_3}$ represents a polynomial in s_4 of degree $2(i_2 + i_3)$. Let $a_{4,i_4}^{(i_2+i_3)}$ denote the coefficient of $s_4^{i_4}$ in the expansion of this polynomial:

$$[P(s_4)]^{i_2+i_3} = \sum_{i_4=0}^{2(i_2+i_3)} a_{4,i_4}^{(i_2+i_3)} s_4^{i_4}.$$

Substituting this expression back into the triple summation, we obtain the finalized form:

$$h^n = \sum_{i_1=0}^{n} \sum_{i_2=0}^{n-i_1} \sum_{i_3=0}^{n-i_1-i_2} \sum_{i_4=0}^{2(i_2+i_3)} \mathcal{A}_{i_1,i_2,i_3,i_4} s_1^{i_1} s_2^{i_2} s_3^{i_3} s_4^{i_4},$$

where the global coefficient $\mathcal{A}_{i_1,i_2,i_3,i_4}$ is defined as:

$$\mathcal{A}_{i_1,i_2,i_3,i_4} = \binom{n}{i_1, i_2, i_3, n-i_1-i_2-i_3} (-1)^{i_2} a_0^{n-i_1-i_2-i_3} a_1^{i_1} a_2^{i_2} a_3^{i_3} a_{4,i_4}^{(i_2+i_3)}.$$

This completes the proof. □

2.2. LLL Lattice Reduction Algorithm

Consider a lattice $\mathcal{L} \subset \mathbb{R}^n$ of rank μ, generated by a set of linearly independent basis vectors $\{\xi_1, \xi_2, \ldots, \xi_\mu\}$. The lattice is defined as the set of all integer linear combinations of these vectors:

$$\mathcal{L} = \left\{ \sum_{i=1}^{\mu} x_i \xi_i \;\middle|\; x_i \in \mathbb{Z} \right\}.$$

In the case where the rank μ coincides with the ambient dimension n, the lattice is termed *full-rank*. For computational efficiency, it is standard to represent the basis as a matrix $\mathbf{B} \in \mathbb{R}^{\mu \times n}$, whose rows are the basis vectors:

$$\mathbf{B} = \begin{pmatrix} \xi_1 \\ \xi_2 \\ \vdots \\ \xi_\mu \end{pmatrix}.$$

Under this notation, any lattice vector $\mathbf{v} \in \mathcal{L}$ can be compactly expressed via the matrix-vector product $\mathbf{v} = \mathbf{x}\mathbf{B}$, where $\mathbf{x} \in \mathbb{Z}^\mu$ is an integer coefficient vector. The determinant of a lattice $\mathcal{L}$, also referred to as its volume, is defined as:

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

which simplifies to $|\det(\mathbf{B})|$ in the full-rank case. While any lattice of rank $\mu \geq 2$ admits an infinite set of bases, all such bases are related by unimodular transformations $\mathbf{B}' = \mathbf{U}\mathbf{B}$ (where $\det(\mathbf{U}) = \pm 1$). Consequently, they generate the same lattice and possess an invariant determinant. The identification of a reduced basis, formed by short and nearly orthogonal vectors, is a fundamental problem in computational number theory, and becomes increasingly difficult as the lattice dimension grows. To overcome this, the LLL algorithm, introduced by Lenstra, Lenstra, and Lovász [12], provides an efficient procedure to compute a reduced basis $\{\xi_1^*, \ldots, \xi_\mu^*\}$, where vectors are relatively short and nearly orthogonal. The resulting reduced basis is especially useful in cryptanalysis, offering a practical tool for lattice-based attacks on RSA and related schemes.

Among the key properties of the LLL algorithm, the following theorem plays a central role and underlies many lattice-based cryptanalytic methods [13].

Theorem 1. *Let $\mathcal{L}$ be a lattice of rank μ with basis $\{\xi_1, \ldots, \xi_\mu\}$. Applying the LLL algorithm produces a reduced basis $\{\xi_1^*, \ldots, \xi_\mu^*\}$ satisfying*

$$\|\xi_1^*\| \leq \ldots \leq \|\xi_l^*\| \leq 2^{\frac{\mu(\mu-1)}{4(\mu+1-l)}} \det(\mathcal{L})^{\frac{1}{\mu+1-l}}, \quad l = 1, \ldots, \mu.$$

2.3. Small Roots of Modular Equations

A fundamental result due to Coppersmith [4] (1996) provides a polynomial-time algorithm for recovering small integer roots of univariate congruences of the form $p(x) \equiv 0 \pmod{M}$, where M is known but its factorization is not assumed. This technique has been extended to multivariate polynomial systems

$$P(s_1, \ldots, s_n) = \sum_{j_1, \ldots, j_n} a_{j_1, \ldots, j_n} s_1^{j_1} \cdots s_n^{j_n},$$

with integer coefficients. The associated Euclidean norm is defined as

$$\|P\| = \sqrt{\sum_{j_1, \ldots, j_n} a_{j_1, \ldots, j_n}^2}.$$

In 1997, Howgrave–Graham [8] developed a variant of Coppersmith's method that establishes a sufficient condition guaranteeing that small modular roots of a multivariate polynomial lift to integer solutions. The corresponding formal result is stated as follows.

Theorem 2 (Howgrave-Graham)**.** *Let $P(s_1, \ldots, s_n)$ be a multivariate polynomial with integer coefficients and at most r nonzero terms, and let $M > 0$. Assume that a tuple $(s_1^\star, \ldots, s_n^\star) \in \mathbb{Z}^n$ satisfies*

$$\begin{cases} P(s_1^\star, \ldots, s_n^\star) \equiv 0 \pmod{M}, \\ |s_i^\star| < S_i, \quad 1 \leq i \leq n, \\ \|P(s_1 S_1, \ldots, s_n S_n)\| < \dfrac{M}{\sqrt{r}}. \end{cases}$$

Then $P(s_1^\star, \ldots, s_n^\star) = 0$.

For multivariate settings with at least three variables, Coppersmith-type algorithms are usually heuristic. As a result, the following assumption [1, 10] is often invoked in the literature.

Assumption 1. *Let $\{H_1, H_2, \ldots, H_\mu\}$ be the set of polynomials obtained after applying LLL reduction. We assume that these polynomials are algebraically independent. That is, for any polynomial $Q \in \mathbb{Z}[s_1, \ldots, s_\mu]$, the relation*

$$Q(H_1, H_2, \ldots, H_\mu) = 0,$$

implies that Q is the zero polynomial.

Assuming the above conditions hold, the common root $(s_1^\star, \ldots, s_n^\star)$ of the polynomial system

$$H_i(s_1, \ldots, s_n) = 0, \quad i = 1, \ldots, \mu,$$

can be recovered using standard computational algebraic tools, such as Gröbner basis computations and resultant-based methods.

2.3.1. The strategy of Jochemsz and May for finding small integer roots

Let N be an integer whose factorization is hidden, and let $h(s_1, \ldots, s_n)$ be a multivariate polynomial such that

$$h(s_1^\star, \ldots, s_n^\star) = 0,$$

where at least one coordinate of the vector $(s_1^\star, \ldots, s_n^\star)$ is linked to a factor of N and can consequently be exploited to recover its factorization. Suppose further that the unknowns satisfy the bounds

$$|s_i^\star| < S_i, \qquad i = 1, \ldots, n.$$

The objective of the strategy of Jochemsz and May [10] is to recover the small integer root $(s_1^\star, \ldots, s_n^\star)$ in an efficient manner and subsequently use the recovered information to factor N.

To this end, we introduce two sets of monomials associated with the polynomial $h(s_1, \ldots, s_n)$, defined as follows:

$$\mathcal{N}_2 = \bigcup_{j=0}^{x} \left\{ s_1^{\alpha_1+j} s_2^{\alpha_2} \cdots s_n^{\alpha_n} \;\middle|\; s_1^{\alpha_1} s_2^{\alpha_2} \cdots s_n^{\alpha_n} \text{ is a monomial of } h^{\kappa-1} \right\},$$

$$\mathcal{N}_1 = \left\{ \text{monomials of } s_1^{\alpha_1} s_2^{\alpha_2} \cdots s_n^{\alpha_n} h \mid s_1^{\alpha_1} s_2^{\alpha_2} \cdots s_n^{\alpha_n} \in \mathcal{N}_2 \right\},$$

where x and κ are parameters that will be optimized later in the analysis.

Subsequently, we introduce the following family of shift polynomials over $\mathcal{N}_1$. For each monomial $s_1^{\alpha_1} s_2^{\alpha_2} \cdots s_n^{\alpha_n}$ belonging to the set $\mathcal{N}_2$, we define the associated shift polynomials as follows:

$$A_{\alpha_1,\ldots,\alpha_n} = s_1^{\alpha_1} s_2^{\alpha_2} \cdots s_n^{\alpha_n} h(s_1, \ldots, s_n) \prod_{j=1}^{n} S_j^{\ell_j - \alpha_j},$$

while for the monomials in the complement set $\mathcal{N}_1 \setminus \mathcal{N}_2$, we introduce the following auxiliary polynomials:

$$B_{\alpha_1,\ldots,\alpha_n} = s_1^{\alpha_1} s_2^{\alpha_2} \cdots s_n^{\alpha_n} \mathcal{R},$$

where ℓ_j is the largest exponent of s_j that appears in the monomials of $\mathcal{N}_2$, and

$$\mathcal{R} = \mathcal{W} \prod_{j=1}^{n} S_j^{\ell_j}, \quad \mathcal{W} = \|h(s_1 S_1, \ldots, s_n S_n)\|_\infty,$$

where $\|\mathbf{v}\|_\infty$ denotes the infinity norm of the vector $\mathbf{v}$.

Note that, in advance, we can assume without loss of generality that the constant term r_0 of h is 1. Indeed, any bound S_j satisfying $\gcd(r_0, S_j) \neq 1$ can be replaced by S_j' such that $S_j < S_j' < 2S_j$ and $\gcd(r_0, S_j') = 1$. The same argument applies to $\mathcal{W}$. This ensures that $\gcd(\mathcal{R}, r_0) = 1$, which allows us to define $H = h \cdot r_0^{-1} \pmod{\mathcal{R}}$, whose constant term is equal to 1.

Given the choice of $\mathcal{W}$ and $\mathcal{R}$, the coefficients of the resulting monomials in $A_{\alpha_1,\ldots,\alpha_n}(S_1 s_1, \ldots, S_n s_n)$ and $B_{\alpha_1,\ldots,\alpha_n}(S_1 s_1, \ldots, S_n s_n)$ form the row vectors of the basis matrix $\mathcal{M}$ of the lattice $\mathcal{L}$. The columns of the matrix correspond to the monomials $s_1^{\alpha_1} s_2^{\alpha_2} \cdots s_n^{\alpha_n} \in \mathcal{N}_1$. By arranging these monomials in lexicographical order, the matrix $\mathcal{M}$ can be put into upper triangular form. The LLL algorithm is subsequently employed to obtain a reduced basis $\mathcal{M}'$ from $\mathcal{M}$. The vectors of this reduced basis correspond to $|\mathcal{N}_1|$ polynomials $H_k(s_1, \ldots, s_n)$ satisfying

$$H_k(s_1^*, \ldots, s_n^*) \equiv 0 \pmod{\mathcal{R}}, \quad 1 \leq k \leq |\mathcal{N}_1|.$$

When the condition of Theorem 2 is fulfilled, these modular relations imply that

$$H_k(s_1^*, \ldots, s_n^*) = 0,$$

as equalities over $\mathbb{Z}$. Finally, under Assumption 1, the common solution $(s_1^*, \ldots, s_n^*)$ can be recovered by computing a Gröbner basis or by using resultants. Once the relevant component of this solution is obtained, it can be

exploited to recover a prime divisor of N, thereby completing the factorization.

Remark 1. *Note that, to facilitate the evaluation of the lattice determinant, it is convenient, in the multivariate extension of Coppersmith's method for finding small modular roots, to consider a monic polynomial, that is, a polynomial whose leading coefficient is equal to 1. This allows the construction of a lower triangular lattice basis matrix. In contrast, the strategy of Jochemsz and May [10] for finding small integer roots requires the polynomial under consideration to have a constant term equal to 1. This condition allows the construction of an upper triangular lattice basis matrix. Moreover, with the notation introduced above, the diagonal entries corresponding to the polynomials* $A_{\alpha_1,\ldots,\alpha_n}(s_1S_1,\ldots,s_nS_n)$ *are of the form*

$$\prod_{j=1}^{n} S_j^{\ell_j},$$

whereas those corresponding to $B_{\alpha_1,\ldots,\alpha_n}(s_1S_1,\ldots,s_nS_n)$ *are of the form*

$$\mathcal{W}\prod_{j=1}^{n} S_j^{\ell_j+\alpha_j}.$$

2.4. Description of the Nitaj-Seck Protocol

The encryption methodology introduced by Nitaj and Seck [18] is built upon a cubic form $\mathcal{K}_\varsigma(N)$ defined over the ring $\mathbb{Z}/N\mathbb{Z}$. The curve is represented by the following cubic equation:

$$\mathcal{K}_\varsigma(N) : z_1^3 + \varsigma z_2^3 + \varsigma^2 z_3^3 - 3\varsigma z_1 z_2 z_3 \equiv 1 \pmod{N}.$$

In this setting, ς is a cubic residue such that $\varsigma \equiv \theta^3 \pmod{N}$. A group operation $\odot$ can be defined on the set of points of the cubic Pell curve $\mathcal{K}_\varsigma(N)$, turning this set into an abelian group [15]. Given a point $P = (p_1, p_2, p_3)$ on $\mathcal{K}_\varsigma(N)$, we write $P^{\odot k}$ for the repeated application of the group operation to P, where k is any positive integer. More explicitly, this notation is defined by

$$P^{\odot k} = \underbrace{P \odot P \odot \cdots \odot P}_{k \text{ factors}}.$$

The scheme relies on two fundamental mapping operations:

- **Encoding ($\mathcal{E}$):** Transforms a message pair $(\mu_1, \mu_2) \in (\mathbb{Z}/N\mathbb{Z})^2$ into a point (z_1, z_2, z_3) on the curve $\mathcal{K}_\varsigma(N)$.
- **Decoding ($\mathcal{D}$):** Reverts a curve point (z_1, z_2, z_3) back to the original coordinates in the message space.

For the purpose of this paper, we are not concerned with the above functions; more details can be found in [18]. Our main focus is the key equation that characterizes their scheme. The setup phase involves the construction of the public and private parameters:

1. **Primes and Modulus:** Select two large primes p and q of equal bit-length ρ, with the constraint $p \equiv q \equiv 1 \pmod 3$. The modulus is computed as $N = p^a q^b$ for some small positive integers a, b.
2. **Cubic Totient:** The auxiliary function Ω for this modulus is defined as:
$$\Omega(a, b, N) = p^{2(a-1)} q^{2(b-1)} (p-1)^2 (q-1)^2. \tag{1}$$
3. **Residue Selection:** Pick a random $\theta \in (\mathbb{Z}/N\mathbb{Z})^*$ and derive the residue $\varsigma \equiv \theta^3 \pmod N$.
4. **Key Assignment:** Choose an encryption exponent e relatively prime to $pq(p-1)(q-1)$. The corresponding decryption exponent d is determined by:
$$ed - 1 \equiv 0 \pmod{\Omega(a, b, N)}. \tag{2}$$

The resulting public key is (N, θ, e), while the private key is (N, θ, d). Table 1 details the algebraic steps required for the encryption and decryption processes.

3. The Proposed Attack

This section is devoted to the description of a lattice-based attack against the Nitaj-Seck scheme [18]. The analysis relies on the assumption that the prime factors have partially shared LSBs, and that the corresponding secret exponents share common MSBs.

It is worth noting that Lemma 3 allows us to determine the ε of LSBs shared by the RSA primes. In particular, if $2^\varepsilon = N^\omega$ with $\omega \geq \frac{1}{4}$, then $\varepsilon = \omega \log_2(N) \geq \frac{1}{4} \log_2(N)$, which corresponds to at least half of the least significant bits of one of the prime factors. Therefore, Coppersmith's result (Lemma 2) yields a polynomial-time factorization of N, and consequently, we restrict our analysis to the case $\omega < \frac{1}{4}$.

Process	Mathematical Operations
Encryption	1. Represent the plaintext as $\mathbf{m} = (\mu_1, \mu_2) \in (\mathbb{Z}/N\mathbb{Z})^2$.
	2. Map $\mathbf{m}$ to the triplet $\mathbf{U} = (z_1, z_2, z_3) \in \mathcal{K}_\varsigma(N)$ via $\mathcal{E}(\mathbf{m}, \theta, N)$.
	3. Compute the point $\mathbf{C}$ using $\mathbf{C} = \mathbf{U}^{\odot e} \pmod N$.
	4. Extract the encrypted pair $(\gamma_1, \gamma_2) = \mathcal{D}(\mathbf{C}, \theta, N)$.
Decryption	1. Re-encode the ciphertext pair (γ_1, γ_2) to the point $\mathbf{C}$ on $\mathcal{K}_\varsigma(N)$.
	2. Retrieve the original triplet $\mathbf{U}$ using $\mathbf{U} = \mathbf{C}^{\odot d} \pmod N$.
	3. Apply the decoding function to recover $\mathbf{m} = \mathcal{D}(\mathbf{U}, \theta, N)$.

Table 1: Summary of the Nitaj-Seck cryptosystem.

Theorem 3. *Consider an RSA modulus $N = pq$, where p and q are balanced primes sharing ε least significant bits, with $2^\varepsilon = N^\omega$ for some $0 \leq \omega < \frac{1}{4}$. Let (e_i, d_i), for $i \in \{1, 2\}$, denote two encryption–decryption pairs satisfying the two key equations $e_i d_i - k_i(p-1)^2(q-1)^2 = 1$. Whenever $e_1, e_2 \approx N^\lambda$, $d_1, d_2 \leq N^\delta$, and $|d_1 - d_2| < N^\varrho$, the modulus N admits a polynomial-time factorization as long as*

$$\varrho + 2\omega \leq 2.5 + \lambda,$$

and

$$\delta < 5 - 4\omega - \varrho - \frac{1}{2}\sqrt{6(5 - 4\omega + 2\lambda - 2\varrho)(1 - 4\omega)}.$$

Proof. By Lemma 3, there exist an unknown parameter v and a known integer v_0 such that the sum of the primes is expressed as $p + q = 2^{2\varepsilon}v + v_0$. On the one hand, one has

$$(p-1)^2(q-1)^2 = (p+q)^2 - 2(N+1)(p+q) + N^2 + 2N + 1.$$

This leads to the expansion

$$\begin{aligned}(p-1)^2(q-1)^2 &= 2^{4\varepsilon}v^2 - 2^{2\varepsilon+1}(N - v_0 + 1)v + N^2 + 2N + 1 - 2Nv_0 - 2v_0 + v_0^2. \\ &= a_2v^2 + a_1v + a_0,\end{aligned}$$

where

$$a_2 = 2^{4\varepsilon}, \quad a_1 = -2^{2\varepsilon+1}(N - v_0 + 1),$$

and

$$a_0 = N^2 + 2N + 1 - 2Nv_0 - 2v_0 + v_0^2.$$

On the other hand, performing the multiplications of the first key equation by e_2 and the second by e_1, and then computing their difference, gives rise to the following identity:

$$e_1e_2(d_1 - d_2) + (p-1)^2(q-1)^2(e_1k_2 - e_2k_1) + e_1 - e_2 = 0. \tag{3}$$

Consequently, it follows that the quadruple $(s_1^\star, s_2^\star, s_3^\star, s_4^\star) = (d_1 - d_2, k_1, k_2, v)$ constitutes a root of the multivariate polynomial equation:

$$h(s_1, s_2, s_3, s_4) = 0,$$

where the functional form of h is defined as:

$$h(s_1, s_2, s_3, s_4) = e_1e_2s_1 - e_2s_2P(s_4) + e_1s_3P(s_4) + r_0, \tag{4}$$

with $P(s_4) = a_2s_4^2 + a_1s_4 + a_0$ and $r_0 = e_1 - e_2$.

To extract the small integer solutions of the previously introduced equation, we employ a variant of Coppersmith's method following the framework developed by Jochemsz and May [10]. This approach relies on constructing two separate families of monomials associated with the polynomial $h(s_1, s_2, s_3, s_4)$. To this end, we introduce two positive integer parameters, denoted by κ and x, where the latter will be carefully optimized in the forthcoming analysis. We then define the set of monomials $\mathcal{N}_2$ as follows:

$$\mathcal{N}_2 = \bigcup_{0 \le r \le x} \left\{ s_1^{\alpha_1}s_2^{\alpha_2}s_3^{\alpha_3}s_4^{\alpha_4+r} \mid s_1^{\alpha_1}s_2^{\alpha_2}s_3^{\alpha_3}s_4^{\alpha_4} \text{ is a monomial of } h^{\kappa-1}(s_1, s_2, s_3, s_4) \right\}.$$

Let $\mathcal{N}_1$ denote the collection of monomials $s_1^{\alpha_1}s_2^{\alpha_2}s_3^{\alpha_3}s_4^{\alpha_4}$ resulting from the product of the terms of $h(s_1, s_2, s_3, s_4)$ with those in $\mathcal{N}_2$. According to Lemma 4, the tuple $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)$ associated with the monomials appearing in $h^{\kappa-1}(s_1, s_2, s_3, s_4)$ is subject to the following conditions:

$$\begin{cases} \alpha_1 = 0, \ldots, \kappa - 1, \\ \alpha_2 = 0, \ldots, \kappa - 1 - \alpha_1, \\ \alpha_3 = 0, \ldots, \kappa - 1 - \alpha_1 - \alpha_2, \\ \alpha_4 = 0, \ldots, 2\alpha_2 + 2\alpha_3. \end{cases}$$

It follows that the monomials constituting the set $\mathcal{N}_2$ are specified as follows:

$$\begin{cases} \alpha_1 = 0, \dots, \kappa - 1, \\ \alpha_2 = 0, \dots, \kappa - 1 - \alpha_1, \\ \alpha_3 = 0, \dots, \kappa - 1 - \alpha_1 - \alpha_2, \\ \alpha_4 = 0, \dots, 2\alpha_2 + 2\alpha_3 + x. \end{cases}$$

Similarly, replacing $\kappa - 1$ with κ yields the elements of $\mathcal{N}_1$, whose exponents satisfy:

$$\begin{cases} \alpha_1 = 0, \dots, \kappa, \\ \alpha_2 = 0, \dots, \kappa - \alpha_1, \\ \alpha_3 = 0, \dots, \kappa - \alpha_1 - \alpha_2, \\ \alpha_4 = 0, \dots, 2\alpha_2 + 2\alpha_3 + x. \end{cases}$$

To proceed with the subsequent stages of the analysis, it is necessary to characterize the monomials belonging to the complement set $\mathcal{N}_1 \setminus \mathcal{N}_2$. It is straightforward to observe that this set can be partitioned into three disjoint subsets, denoted by $\mathcal{C}_0$, $\mathcal{C}_1$, and $\mathcal{C}_2$. Each subset is defined by specific boundary conditions on the exponents $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)$. The set $\mathcal{C}_0$ contains the monomials where the first exponent α_1 reaches the boundary value κ. The remaining exponents are constrained as follows:

$$\mathcal{C}_0 : \begin{cases} \alpha_1 = \kappa, \\ \alpha_2 = 0, \dots, \kappa - \alpha_1, \\ \alpha_3 = 0, \dots, \kappa - \alpha_1 - \alpha_2, \\ \alpha_4 = 0, \dots, 2\alpha_2 + 2\alpha_3 + x. \end{cases}$$

In the set $\mathcal{C}_1$, the second exponent α_2 is uniquely determined by

$$\mathcal{C}_1 : \begin{cases} \alpha_1 = 0, \dots, \kappa - 1, \\ \alpha_2 = \kappa - \alpha_1, \\ \alpha_3 = 0, \dots, \kappa - \alpha_1 - \alpha_2, \\ \alpha_4 = 0, \dots, 2\alpha_2 + 2\alpha_3 + x. \end{cases}$$

Finally, $\mathcal{C}_2$ corresponds to the cases in which the third exponent α_3 is char-

acterized by:

$$\mathcal{C}_2 : \begin{cases} \alpha_1 = 0, \ldots, \kappa - 1, \\ \alpha_2 = 0, \ldots, \kappa - 1 - \alpha_1, \\ \alpha_3 = \kappa - \alpha_1 - \alpha_2, \\ \alpha_4 = 0, \ldots, 2\alpha_2 + 2\alpha_3 + x. \end{cases}$$

A fundamental requirement for applying Coppersmith's framework is the determination of accurate upper bounds on the unknown integer roots. For the variable $s_1^\star$, since we have $|d_1 - d_2| < N^\varrho$, we set the bound $S_1 = N^\varrho$. To establish the bounds for $s_2^\star$ and $s_3^\star$, we note that $(p-1)^2(q-1)^2 > \frac{N^2}{4}$ and that for $i \in \{1, 2\}$, the following relation holds:

$$k_i = \frac{e_i d_i - 1}{(p-1)^2(q-1)^2}.$$

By considering $e_i \approx N^\lambda$ and $d_i \leq N^\delta$, it follows that:

$$s_i^\star < 4N^{\lambda+\delta-2}.$$

Accordingly, we set the bounds for $s_2^\star$ and $s_3^\star$ as $S_2 = S_3 = 4N^{\lambda+\delta-2}$.

Regarding $s_4^\star = v$, we utilize the LSB relation $2^\varepsilon = N^\omega$ and the sum of factors $p + q = 2^{2\varepsilon} v + v_0$. Given the inequality $p + q < 3N^{1/2}$ from Lemma 1, we can isolate $s_4^\star$ as follows:

$$s_4^\star = \frac{p + q - v_0}{2^{2\varepsilon}} < 3N^{\frac{1}{2}-2\omega}.$$

Since $\omega < \frac{1}{4}$, we can define the bound for $s_4^\star$ as:

$$S_4 = 3N^{\frac{1}{2}-2\omega}.$$

Next, let $\mathcal{W}$ denote the maximum absolute value of the coefficients of the polynomial function $h(s_1 S_1, s_2 S_2, s_3 S_3, s_4 S_4)$. This is formally defined by the L_∞ norm:

$$\mathcal{W} := \|h(s_1 S_1, s_2 S_2, s_3 S_3, s_4 S_4)\|_\infty.$$

From the structure of h as in (4), we have

$$h(s_1 S_1, s_2 S_2, s_3 S_3, s_4 S_4) = e_1 e_2 S_1 s_1 + (e_1 S_3 s_3 - e_2 S_2 s_2)(a_2 S_4^2 s_4^2 + a_1 S_4 s_4 + a_0) + r_0, \tag{5}$$

In particular, $-e_2 S_2 a_0$ is one of the coefficients of $h(s_1 S_1, s_2 S_2, s_3 S_3, s_4 S_4)$, and thus $\mathcal{W}$ can be bounded below by the term $e_2 S_2 a_0$. When N is taken to be sufficiently large, we observe that $a_0 \approx N^2$. Under the assumption that this approximation holds, we obtain the following lower bound

$$\mathcal{W} \geq e_2 S_2 a_0 \approx N^{\lambda} N^{\lambda+\delta-2} N^2 = N^{2\lambda+\delta}. \tag{6}$$

We then define the parameter $\mathcal{R}$, which is central to the lattice determinant calculation, as:

$$\mathcal{R} = \mathcal{W} \left(\prod_{j=1}^{3} S_j \right)^{\kappa-1} S_4^{2(\kappa-1)+x}.$$

Since the expression defining $\mathcal{R}$ may involve factors that are not necessarily integers, the value of $\mathcal{R}$ should be rounded to the nearest integer. We may, without loss of generality, assume that $\gcd(\mathcal{R}, r_0) = 1$. In practice, when this condition does not hold, the parameters S_j and $\mathcal{W}$ can be slightly adjusted until coprimality is achieved. This ensures the existence of the modular inverse $b_0 \equiv (r_0)^{-1} \pmod{\mathcal{R}}$. Utilizing this inverse, we define the transformed polynomial $H(s_1, s_2, s_3, s_4)$ as follows:

$$H(s_1, s_2, s_3, s_4) \equiv b_0 \cdot h(s_1, s_2, s_3, s_4) \pmod{\mathcal{R}}.$$

By construction, the constant term of H is normalized to 1.

Following the approach introduced in [10], we construct two distinct families of polynomials that will form the basis of our lattice construction. For each monomial $s_1^{\alpha_1} s_2^{\alpha_2} s_3^{\alpha_3} s_4^{\alpha_4}$ belonging to the set $\mathcal{N}_2$, we define the associated shift polynomials as follows:

$$A_{\alpha_1,\alpha_2,\alpha_3,\alpha_4} = s_1^{\alpha_1} s_2^{\alpha_2} s_3^{\alpha_3} s_4^{\alpha_4} H(s_1, s_2, s_3, s_4) S_1^{\kappa-1-\alpha_1} S_2^{\kappa-1-\alpha_2} S_3^{\kappa-1-\alpha_3} S_4^{2(\kappa-1)+x-\alpha_4}.$$

Furthermore, for the monomials in the complement set $\mathcal{N}_1 \setminus \mathcal{N}_2$, we introduce the following auxiliary polynomials:

$$B_{\alpha_1,\alpha_2,\alpha_3,\alpha_4} = s_1^{\alpha_1} s_2^{\alpha_2} s_3^{\alpha_3} s_4^{\alpha_4} \mathcal{R}.$$

The coefficient vectors induced by the polynomials

$$A_{\alpha_1,\alpha_2,\alpha_3,\alpha_4}(s_1 S_1, s_2 S_2, s_3 S_3, s_4 S_4) \quad \text{and} \quad B_{\alpha_1,\alpha_2,\alpha_3,\alpha_4}(s_1 S_1, s_2 S_2, s_3 S_3, s_4 S_4),$$

generate a lattice $\mathcal{L}(\kappa, x)$ with dimension $\mu(\kappa, x)$. The dimension equals the total count of polynomials generated by the families $A_{\alpha_1,\alpha_2,\alpha_3,\alpha_4}$ and

$B_{\alpha_1,\alpha_2,\alpha_3,\alpha_4}$, which are parameterized by $\mathcal{N}_2$ and $\mathcal{N}_1 \setminus \mathcal{N}_2$, respectively. Consequently, computing the cardinality of $\mathcal{N}_1$ is sufficient to obtain the explicit expression for $\mu(\kappa, x)$, that is to say

$$\begin{aligned}\mu(\kappa,x) &= \sum_{\alpha_1=0}^{\kappa}\sum_{\alpha_2=0}^{\kappa-\alpha_1}\sum_{\alpha_3=0}^{\kappa-\alpha_1-\alpha_2}\sum_{\alpha_4=0}^{2\alpha_2+2\alpha_3+x} 1,\\ &= \frac{1}{6}\kappa^4 + \frac{1}{6}\kappa^3(x+7) + \frac{1}{6}\kappa^2(6x+17) + \frac{1}{6}\kappa(11x+17) + x + 1.\end{aligned}$$

To facilitate the evaluation of the determinant, the lattice basis matrix is arranged in triangular form. Following the approach described in [10], we impose a specific ordering on both rows and columns. The rows, corresponding to the polynomials $A_{\alpha_1,\alpha_2,\alpha_3,\alpha_4}$ and $B_{\alpha_1,\alpha_2,\alpha_3,\alpha_4}$, as well as the columns, associated with the monomials $s_1^{\alpha_1}s_2^{\alpha_2}s_3^{\alpha_3}s_4^{\alpha_4}$, are ordered lexicographically with respect to the index tuple $(\alpha_1, \alpha_2, \alpha_3, \alpha_4)$. Under this ordering, the resulting matrix is upper triangular. An illustrative example is provided in Table 2, where the symbol $\star$ designates the nonzero entries.

Polynomials \ Monomials	1	s_3	s_3s_4	$s_3s_4^2$	s_2	s_2s_4	$s_2s_4^2$	s_1
$A_{0,0,0,0}$	1	$\star$	$\star$	$\star$	$\star$	$\star$	$\star$	$\star$
$B_{0,0,1,0}$	0	$S_3\mathcal{R}$	0	0	0	0	0	0
$B_{0,0,1,1}$	0	0	$S_3S_4\mathcal{R}$	0	0	0	0	0
$B_{0,0,1,2}$	0	0	0	$S_3S_4^2\mathcal{R}$	0	0	0	0
$B_{0,1,0,0}$	0	0	0	0	$S_2\mathcal{R}$	0	0	0
$B_{0,1,0,1}$	0	0	0	0	0	$S_2S_4\mathcal{R}$	0	0
$B_{0,1,0,2}$	0	0	0	0	0	0	$S_2S_4^2\mathcal{R}$	0
$B_{1,0,0,0}$	0	0	0	0	0	0	0	$S_1\mathcal{R}$

Table 2: The lattice matrix in the case $(\kappa, x) = (1, 0)$.

Moreover, the diagonal entries of the obtained matrix are as follows:

- Regarding the polynomials $A_{\alpha_1,\alpha_2,\alpha_3,\alpha_4}$, the diagonal terms are:

$$(S_1S_2S_3)^{\kappa-1}S_4^{2(\kappa-1)+x}.$$

- Concerning the polynomials $B_{\alpha_1,\alpha_2,\alpha_3,\alpha_4}$, the diagonal terms are given by:

$$\mathcal{W}S_1^{\kappa-1+\alpha_1}S_2^{\kappa-1+\alpha_2}S_3^{\kappa-1+\alpha_3}S_4^{2(\kappa-1)+x+\alpha_4}.$$

To determine the total contribution to the determinant, we define the following partial sums for each index $i \in \{1, 2, 3, 4\}$ over the set $\mathcal{N}_1 \setminus \mathcal{N}_2$:

$$m_i = \sum_{s_1^{\alpha_1} s_2^{\alpha_2} s_3^{\alpha_3} s_4^{\alpha_4} \in \mathcal{N}_1 \setminus \mathcal{N}_2} \alpha_i. \tag{7}$$

An easy computation shows that the lattice determinant can be given as

$$\det(\mathcal{L}(\kappa, x)) = \left(\prod_{j=1}^{3} S_j^{(\kappa-1)|\mathcal{N}_1|+m_j} \right) \times S_4^{(2(\kappa-1)+x)|\mathcal{N}_1|+m_4} \times \mathcal{W}^{|\mathcal{N}_1 \setminus \mathcal{N}_2|}. \tag{8}$$

We next perform LLL reduction on the lattice basis matrix $\mathbf{B}(\kappa, x)$. This yields a reduced basis $\mathbf{B}'(\kappa, x)$, which induces a new family of polynomial relations:

$$H_i(s_1^\star, s_2^\star, s_3^\star, s_4^\star) \equiv 0 \pmod{\mathcal{R}}, \quad \text{for } i = 1, \ldots, \mu(\kappa, x).$$

The derivation of the target solution proceeds from the synthesis of Theorem 2 and the structural properties of the LLL-reduced basis established in Theorem 1. Specifically, for the case $l = 4$, the existence of the solution is ensured by satisfying the following condition:

$$2^{\frac{\mu(\kappa,x)(\mu(\kappa,x)-1)}{4(\mu(\kappa,x)-3)}} \det(\mathcal{L}(\kappa, x))^{\frac{1}{\mu(\kappa,x)-3}} < \frac{\mathcal{R}}{\sqrt{\mu(\kappa, x)}}.$$

By rearranging the terms in the determinant inequality and neglecting the lower-order LLL constants for sufficiently large N, the condition for root recovery simplifies to the following bound:

$$\det(\mathcal{L}(\kappa, x)) < \mathcal{R}^{\mu(\kappa,x)}.$$

In conjunction with (8), we obtain

$$\left(\prod_{j=1}^{3} S_j^{(\kappa-1)|\mathcal{N}_1|+m_j} \right) \times S_4^{(2(\kappa-1)+x)|\mathcal{N}_1|+m_4} \times \mathcal{W}^{|\mathcal{N}_1 \setminus \mathcal{N}_2|} < \left(\mathcal{W} \prod_{j=1}^{3} S_j^{\kappa-1} S_4^{2(\kappa-1)+x} \right)^{\mu(\kappa,x)}.$$

By substituting $\mu(\kappa, x) = |\mathcal{N}_1| = |\mathcal{N}_2| + |\mathcal{N}_1 \setminus \mathcal{N}_2|$ into the previous expression, we obtain

$$\prod_{j=1}^{4} S_j^{m_j} < \mathcal{W}^{|\mathcal{N}_2|}. \tag{9}$$

We facilitate the computation of m_j ($j \in \{1,2,3,4\}$) by considering the partition $\mathcal{N}_1 \setminus \mathcal{N}_2 = \mathcal{C}_0 \cup \mathcal{C}_1 \cup \mathcal{C}_2$. Under this setup, we define an auxiliary function $f(a_{\alpha_1,\alpha_2,\alpha_3,\alpha_4})$ across this decomposition as

$$\begin{aligned} f(a_{\alpha_1,\alpha_2,\alpha_3,\alpha_4}) = &\sum_{\alpha_1=\kappa}^{\kappa} \sum_{\alpha_2=0}^{\kappa-\alpha_1} \sum_{\alpha_3=0}^{\kappa-\alpha_1-\alpha_2} \sum_{\alpha_4=0}^{2\alpha_2+2\alpha_3+x} a_{\alpha_1,\alpha_2,\alpha_3,\alpha_4} \\ &+ \sum_{\alpha_1=0}^{\kappa-1} \sum_{\alpha_2=\kappa-\alpha_1}^{\kappa-\alpha_1} \sum_{\alpha_3=0}^{\kappa-\alpha_1-\alpha_2} \sum_{\alpha_4=0}^{2\alpha_2+2\alpha_3+x} a_{\alpha_1,\alpha_2,\alpha_3,\alpha_4} \\ &+ \sum_{\alpha_1=0}^{\kappa-1} \sum_{\alpha_2=0}^{\kappa-1-\alpha_1} \sum_{\alpha_3=\kappa-\alpha_1-\alpha_2}^{\kappa-\alpha_1-\alpha_2} \sum_{\alpha_4=0}^{2\alpha_2+2\alpha_3+x} a_{\alpha_1,\alpha_2,\alpha_3,\alpha_4}. \end{aligned}$$

Then, we can see that $m_j = f(\alpha_j)$. Moreover, the size of $\mathcal{N}_2$ is given by

$$|\mathcal{N}_2| = \sum_{\alpha_1=0}^{\kappa-1} \sum_{\alpha_2=0}^{\kappa-1-\alpha_1} \sum_{\alpha_3=0}^{\kappa-1-\alpha_1-\alpha_2} \sum_{\alpha_4=0}^{2\alpha_2+2\alpha_3+x} 1.$$

To determine an optimized bound, we perform a standard asymptotic analysis by setting $x = u\kappa$. For a sufficiently large parameter κ, the dominant parts of the exponents m_j and the cardinality of the set $\mathcal{N}_2$ are evaluated as follows:

$$\begin{aligned} m_1 &= \frac{1+u}{6}\kappa^4 + o(\kappa^4), \\ m_2 &= \frac{3+2u}{12}\kappa^4 + o(\kappa^4), \\ m_3 &= \frac{3+2u}{12}\kappa^4 + o(\kappa^4), \\ m_4 &= \frac{6+8u+3u^2}{12}\kappa^4 + o(\kappa^4), \\ |\mathcal{N}_2| &= \frac{1+u}{6}\kappa^4 + o(\kappa^4). \end{aligned}$$

By focusing on the dominant terms, substituting $S_1 = N^{\varrho}$, $S_2 = S_3 = 4N^{\lambda+\delta-2}$, $S_4 = 3N^{\frac{1}{2}-2\omega}$, and the lower bound for $\mathcal{W}$ given in (6) into (9), and neglecting the smaller terms, we obtain

$$N^{\varrho\frac{1+u}{6}\kappa^4} \cdot N^{(\lambda+\delta-2)(\frac{3+2u}{12}\kappa^4+\frac{3+2u}{12}\kappa^4)} \cdot N^{(\frac{1}{2}-2\omega)\frac{6+8u+3u^2}{12}\kappa^4} < N^{(\frac{1+u}{6}\kappa^4)(2\lambda+\delta)}.$$

Subsequently, a direct simplification yields the expression:

$$-3(4\omega-1)u^2+4(\delta-8\omega+\rho-2)u+8\delta+4\lambda-24\omega+4\rho-18<0, \quad (10)$$

which reaches its optimal value at

$$\hat{u}=\frac{16\omega-2\varrho-2\delta+4}{3-12\omega}.$$

Upon substituting $\hat{u}$ into the above inequality, we obtain

$$-\delta^2-2(4\omega+\rho-5)\delta-12\lambda\omega+8\omega^2+4\omega\rho-\rho^2+3\lambda+4\omega+7\rho-\frac{35}{2}<0.$$

Solving the quadratic inequality yields the following valid range for δ

$$\delta<5-4\omega-\varrho-\frac{1}{2}\sqrt{6(5-4\omega+2\lambda-2\varrho)(1-4\omega)}.$$

Since $\omega<\frac{1}{4}$, then the former bound is well defined if

$$5-4\omega+2\lambda-2\varrho\geq 0,$$

or equivalently

$$\varrho+2\omega\leq 2.5+\lambda.$$

Under these constraints, and adhering to Assumption 1, we obtain a set of four polynomials $\{H_1,H_2,H_3,H_4\}$. By solving the system:

$$H_i(s_1,s_2,s_3,s_4)=0 \quad \text{for } i\in\{1,2,3,4\},$$

through the use of Gröbner basis techniques or resultant-based computations, we can uniquely determine the value of $s_4^\star=v$. Given the relationship $p+q=2^{2\varepsilon}v+v_0$ and the modulus $N=pq$, the prime factors (p,q) are efficiently recovered via the quadratic formula. This concludes the proof. □

Theorem 3 yields two primary implications: one in which the secret exponents share a substantial portion of their most significant bits, and another in which no such sharing occurs.

Corollary 1. *Let $N=pq$ be an RSA modulus with p and q of equal bit-length sharing ε least significant bits, where $2^\varepsilon=N^\omega$ and $0\leq\omega<\frac{1}{4}$. Suppose there exist (e_i,d_i), $i\in\{1,2\}$, such that*

$$e_id_i-k_i(p-1)^2(q-1)^2=1,$$

with $e_i \approx N^{\lambda}$ *and* $d_1 \approx d_2 \approx N^{\delta}$. *Then* (p, q) *can be recovered in polynomial time if*

$$2\omega < 2.5 + \lambda,$$

and

$$\delta < 5 - 4\omega - \frac{1}{2}\sqrt{6(5 - 4\omega + 2\lambda)(1 - 4\omega)}.$$

Proof. In the case where the secret exponents share a substantial number of their most significant bits, Theorem 3 ensures that the difference $|d_1 - d_2|$ is small, corresponding to $\varrho \approx 0$. By substituting $\varrho = 0$ into the bound established in the preceding theorem, we directly obtain the desired inequality. □

Corollary 2. *Let* $N = pq$ *be an RSA modulus with* p *and* q *of equal bit-length sharing* ε *least significant bits, where* $2^{\varepsilon} = N^{\omega}$ *and* $0 \leq \omega < \frac{1}{4}$. *Suppose there exist* (e_i, d_i), $i \in \{1, 2\}$, *such that*

$$e_i d_i - k_i(p-1)^2(q-1)^2 = 1,$$

with $e_i \approx N^{\lambda}$ *and* $d_i \leq N^{\delta}$. *Then* (p, q) *can be recovered in polynomial time if*

$$\delta < \min\left(2.5 + \lambda - 2\omega, \frac{17}{8} - \frac{1}{2}\omega - \frac{1}{8}\sqrt{3(16\lambda - 12\omega + 3)(1 - 4\omega)}\right).$$

Proof. In the absence of a significant overlap in the most significant bits of d_1 and d_2, the difference $|d_1 - d_2|$ is not necessarily small. A natural upper bound in this case is N^{δ}. Hence, Theorem 3 is applied by setting $\varrho = \delta$. Substituting this value into the quadratic inequality derived earlier and solving for δ leads to the desired bound. □

Remarks 1. *Note that the maximal theoretical bound for short secret exponent attacks on a single key in the Nitaj-Seck scheme [18], established by Rahmani and Nitaj [20], is given by* $d < N^{2-\sqrt{\lambda}}$ *when* $e = N^{\lambda}$. *This bound reaches its maximum when* $\lambda \approx 2$, *yielding* $d < N^{2-\sqrt{2}}$. *However, in practical scenarios, such attacks do not attain the bound* $N^{2-\sqrt{2}} \approx N^{0.585}$. *In the case of our attack involving two public keys* (N, e_1) *and* (N, e_2), *we compare this classical bound with the theoretical bounds on the secret exponents* d_1 *and* d_2 *derived from our method, particularly in light of the preceding corollaries.*

1. *The bound constructed in Corollary 1 for* $\lambda = 2$ *is reduced to*

$$\delta < 5 - 4\omega - \frac{1}{2}\sqrt{6(9-4\omega)(1-4\omega)}.$$

Let $B^{RN} = N^{2-\sqrt{2}}$ *denote the bound of Rahmani and Nitaj [20], and define*

$$B_1(\omega) = 5 - 4\omega - \frac{1}{2}\sqrt{6(9-4\omega)(1-4\omega)}.$$

Notice that in this case, the minimal bound provided by our method occurs when the prime factors share only a very small portion of their LSBs, that is, when $\omega \approx 0$. *In this case, our bound becomes*

$$B_1(0) = 5 - \frac{1}{2}\sqrt{54} \approx 1.3257.$$

Figure 1 illustrates that attacks leveraging two encryption exponents provide improved bounds compared to those based on a single exponent when the secret exponents share a large portion of their MSBs. It further shows that our bound increases significantly beyond that of Rahmani and Nitaj [20] as ω *varies over the interval* $[0, \frac{1}{4}]$.

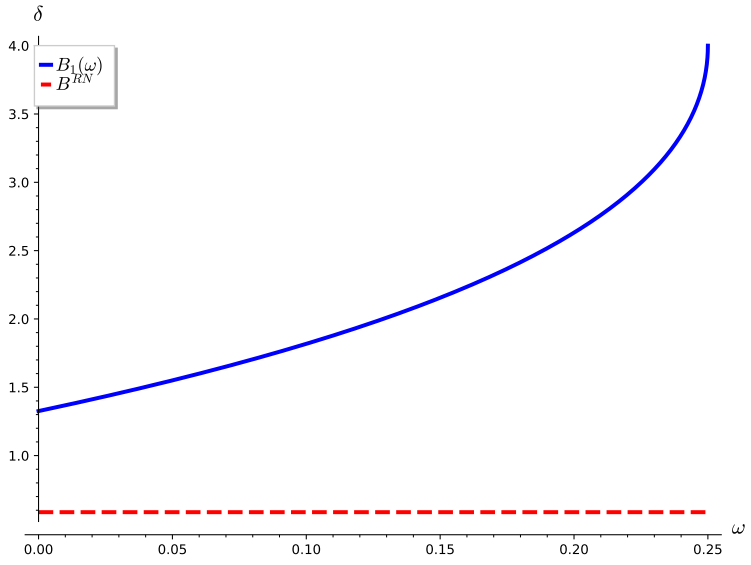

Figure 1: A graphical comparison between our bound and the bound obtained in [20].

2. *Let us consider the bound established in Corollary 2 with respect to δ for $\lambda = 2$, namely*

$$\delta < \frac{17}{8} - \frac{1}{2}\omega - \frac{1}{8}\sqrt{3(35 - 12\omega)(1 - 4\omega)}.$$

Let $B^{RN} = N^{2-\sqrt{2}}$ be the bound of Rahmani and Nitaj [20], and set

$$B_2(\omega) = \frac{17}{8} - \frac{1}{2}\omega - \frac{1}{8}\sqrt{3(35 - 12\omega)(1 - 4\omega)}.$$

Notice that in this case, the minimal bound provided by our method occurs when the prime factors share only a very small portion of their LSBs, that is, when $\omega \approx 0$. In this case, our bound becomes

$$B_2(0) = \frac{17}{8} - \frac{1}{8}\sqrt{105} \approx 0.8441.$$

Figure 2 illustrates that attacks exploiting two encryption exponents provide improved bounds compared to single-exponent attacks as the shared portion of the LSBs of the prime factors increases. It also shows that our bound remains significantly stronger than that of Rahmani and Nitaj [20] for $\omega \in [0, \frac{1}{4}]$.

3. *In 2026, Chnioune et al. [3] presented an attack on the same cryptosystem. They showed that, when the RSA primes p and q share their ε least significant bits, with $2^{\varepsilon} = N^{\omega}$ and $0 \leq \omega < \frac{1}{4}$, the factorization of N can be achieved in polynomial time for $e \approx N^2$ whenever δ satisfies*

$$\delta < B_3(\omega),$$

where

$$B_3(\omega) = \frac{1}{3}\left(7 - 4\omega - 2\sqrt{(1 - 4\omega)(7 - 4\omega)}\right).$$

We note that both the attack of Chnioune et al. [3] and our method are effective under the assumption that the primes share a sufficiently large number of LSBs. However, compared with the optimal bound $\delta < 2 - \sqrt{2} \approx 0.586$ established by Rahmani and Nitaj [20] for small private exponent attacks, their method provides a weaker bound when the primes share only a few LSBs, that is, when $\omega \approx 0$. In this case, their bound reduces to

$$B_3(0) = \frac{7 - 2\sqrt{7}}{3} \approx 0.569.$$

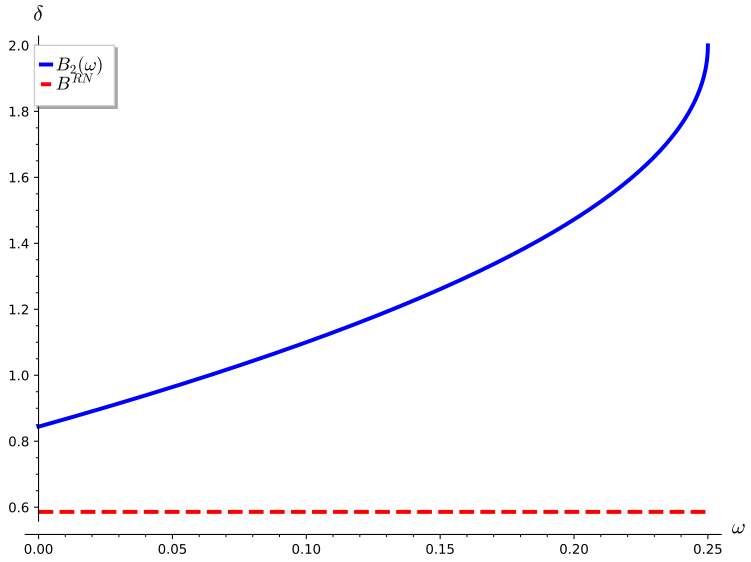

Figure 2: A graphical comparison between our bound and the bound obtained in [20].

To compare this bound with our results, we consider the two upper bounds $B_1(\omega)$ and $B_2(\omega)$ introduced above. Figure 3 provides a graphical comparison of these bounds.

Remark 2. *Note that Seck and Tchapgnouo [24] extended the optimal bound $\delta < 2-\sqrt{\lambda}$ established by Rahmani and Nitaj [20] to scenarios in which some of the most significant bits of one of the RSA primes are known. Specifically, they studied the generalized key equation $ex - y(p-1)^2(q-1)^2 = z$ and showed that, given an approximation p_0 satisfying $|p - p_0| < N^{\zeta}$, together with $x < N^{\delta}$ and $z < N^{\gamma}$, the scheme can be broken provided that*

$$\delta < 2 - \frac{\gamma}{2} - \sqrt{2\zeta\lambda}.$$

Note that the bound $\delta < 2 - \sqrt{\lambda}$ established by Rahmani and Nitaj [20] is recovered when $\zeta = \frac{1}{2}$ and $\gamma = 0$. Furthermore, their attack achieves better bounds as the number of known most significant bits of one of the RSA primes increases, that is, when $\gamma < \frac{1}{2}$. However, exploiting this advantage requires a side-channel attack to obtain the corresponding bits. In contrast, our paper assumes that p and q share ε unknown least significant bits. The adversary can therefore consider successive candidate values of ε and use Lemma 3 to

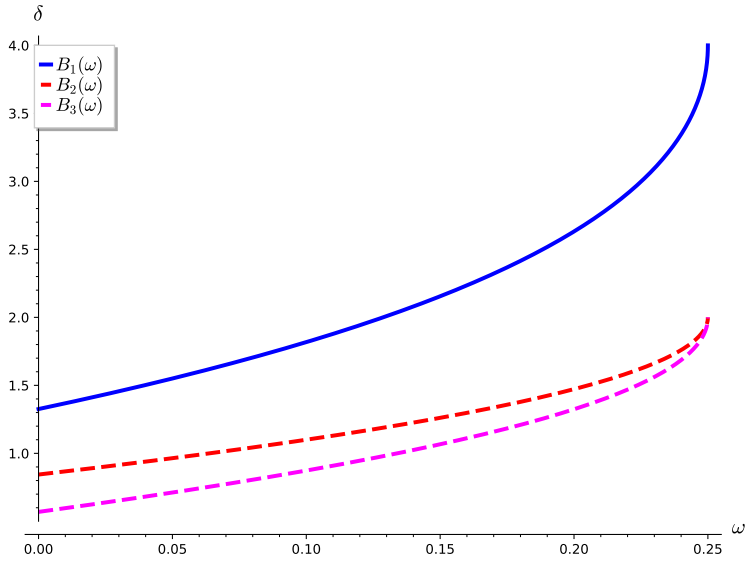

Figure 3: A graphical comparison between our bounds and the bound obtained in [3].

compute u_0 and subsequently v_0. Hence, unlike the approach of Seck and Tchapgnouo, our attack does not require any side-channel method to obtain or guess the shared ε LSBs.

4. A Detailed Experiment

In this section, we present a detailed numerical experiment to validate our theoretical analysis. In particular, we demonstrate that when the RSA primes share a certain number of least significant bits and the corresponding private exponents share a certain number of most significant bits, our method achieves improved bounds compared with existing attacks that target a single public key.

All simulations were conducted using SageMath 10.4 on a workstation running Ubuntu 22.04.3 LTS. The hardware configuration consists of an Intel® Core™ i5-4460 CPU (3.20 GHz), with 4 cores and 8 GB of RAM. The source code is publicly available at:

`https://github.com/algebre96-hue/Cubic_RSA_attack`

Our goal is to solve the key equations

$$e_1 d_i - k_i(p-1)^2(q-1)^2 = 1, \quad i \in \{1, 2\};$$

where the unknowns are in the tuple $(d_1, d_2, k_1, k_2, p, q)$. We take the following public parameters:

$$N = 526311336990840327896397942710355198201963093340120785697061735 7\backslash$$
$$275858669230457910083434 73,$$
$$e_1 = 930144098513925676360769122782984059263274292301245824844522385 8\backslash$$
$$327527284094015051501024936818245999621767820519000927977219378524 2\backslash$$
$$446339436270095292455198753763919338891490714019,$$
$$e_2 = 147816822399569415559874579563979329851621777411472253500204277 9\backslash$$
$$228138795112417330841588471289316834864869270415092882427461075762 3\backslash$$
$$533455720808134914080152837448366070089559806 79.$$

In this instance, the exponents relate to the modulus such that $e_1 \approx N^{\lambda_1}$ and $e_2 \approx N^{\lambda_2}$, with $\lambda_1 \approx 1.99471$ and $\lambda_2 \approx 1.97466$. We then define

$$\lambda = \max(\lambda_1, \lambda_2) \approx 1.9947.$$

Assume that the prime factors p and q share ε least significant bits. In practical scenarios, ε is determined by exhaustive search ($\varepsilon = 1, 2, \dots$). For this experiment, we set $\varepsilon = 37$, implying $2^\varepsilon = N^\omega$ with $\omega \approx 0.125$. Using Lemma 3, we solve $y^2 \equiv N \pmod{2^\varepsilon}$. Among the candidates, we identify $u_0 = 135629037383$. A subsequent calculation of the expression

$$v_0 \equiv 2u_0 + (N - u_0^2)u_0^{-1} \pmod{2^{2\varepsilon}},$$

gives that $v_0 = 11554141945947616034446$.

Following the algorithm outlined in Theorem 3, we aim to solve the equation

$$h(s_1, s_2, s_3, s_4) = e_1 e_2 s_1 - e_2 s_2 P(s_4) + e_1 s_3 P(s_4) + r_0 = 0,$$

where

$$r_0 = e_1 - e_2, \quad P(s_4) = a_2 s_4^2 + a_1 s_4 + a_0, \quad a_2 = 2^{4\varepsilon},$$

and

$$a_1 = -2^{2\varepsilon+1}(N - v_0 + 1), \quad a_0 = N^2 + 2N + 1 - 2Nv_0 - 2v_0 + v_0^2.$$

In alignment with the conditions of Theorem 3, we consider a scenario where d_1 and d_2 share 65 of their most significant bits (MSBs). For experimental

verification, we assume that the exponents satisfy $d_1, d_2 < N^{\delta}$ and that their difference is bounded by $|d_1 - d_2| < N^{\varrho}$. Setting $\delta = \varrho = 0.61$, we verify the admissibility of our attack. In particular, the theoretical upper bound for δ satisfies the requirements of Theorem 3, namely:

$$\varrho + 2\omega \approx 0.86 \leq 2.5 + \lambda \approx 4.4947,$$

and

$$\delta < 5 - 4\omega - \varrho - \frac{1}{2}\sqrt{6(5 - 4\omega + 2\lambda - 2\varrho)(1 - 4\omega)} \approx 1.555.$$

Next, we compute the corresponding bounds $S_1 = N^{\varrho}$, $S_2 = S_3 = 4N^{\lambda+\delta-2}$, and $S_4 = 3N^{\frac{1}{2}-2\omega}$, as well as the quantities

$$\mathcal{W} := \|h(s_1 S_1, s_2 S_2, s_3 S_3, s_4 S_4)\|_{\infty},$$

and

$$\mathcal{R} = \mathcal{W}\left(\prod_{j=1}^{3} S_j\right)^{\kappa-1} S_4^{2(\kappa-1)+x},$$

where we fix the parameters $(\kappa, x) = (2, 5)$. A first computation shows that $\gcd(r_0, \mathcal{R}) = 20$. We then progressively adjust the values of S_i and $\mathcal{W}$ until we obtain $\gcd(S_i, r_0) = \gcd(\mathcal{W}, r_0) = 1$. As a final result, we take the updated values

$$\begin{aligned}
S_1 =& \lfloor N^{\varrho} \rfloor \approx 536979940288198733640205173152759738233483484466459443 3,\\
S_2 =& S_3 = \lfloor 4N^{\lambda+\delta-2} \rfloor \approx 718613657973473378236734695537255073650890517104925081 9,\\
S_4 =& 3N^{\frac{1}{2}-2\omega} = 80803828083658451845121,\\
\mathcal{W} =& 257653285614420342645449903924946901214992632264545398924517518376737 9\backslash\\
& 150450010704300310273689201747887906114594965951276351523465460886710 7\backslash\\
& 543599359666449394802029568412864214661274261768779088464086319314799 3\backslash\\
& 656352064336399152607832336896635064770659797083834477963732964678036 7\backslash\\
& 527826489526664968013403770818350447424011740951329345276718405331945 3\backslash\\
& 331566899,
\end{aligned}$$

which ensures that $\gcd(r_0, \mathcal{R}) = 1$, as required in the proof of Theorem 3.

From the coefficient vectors of the polynomials $A_{\alpha_1,\alpha_2,\alpha_3,\alpha_4}(s_1 S_1, s_2 S_2, s_3 S_3, s_4 S_4)$ and $B_{\alpha_1,\alpha_2,\alpha_3,\alpha_4}(s_1 S_1, s_2 S_2, s_3 S_3, s_4 S_4)$ defined in the proof of Theorem 3, we

construct a lattice $\mathcal{L}(\kappa, x)$ of dimension $\mu(\kappa, x) = 80$. After applying LLL reduction, the reduced basis yields 80 polynomials. Computing a Gröbner basis allows us to extract the first four polynomials, and solving the resulting system over the integers gives

$$\begin{aligned}
s_1^\star &= -542451020224347005453339200685883946473596313535396734 0,\\
s_2^\star &= 260211972742639010457164605423204131712821314032309905,\\
s_3^\star &= 330819338832278000261437049570993206167675854812027 05,\\
s_4^\star &= 76812460878792620045822.
\end{aligned}$$

Once $v = s_4^\star$ is determined, one can solve the system formed by

$$p + q = 2^{2\varepsilon} v + v_0 = 145094636288298448995522901395162837362394689 4,$$

and $N = pq$. It follows that

$$\begin{aligned}
p &= 725473181441492244977614506975816179676798791,\\
q &= 725473181441492244977614506975812193947148103.
\end{aligned}$$

The LLL reduction phase, followed by the Gröebner basis computation, was completed in under 35 seconds. Subsequently, the modular inverses of e_1 and e_2 modulo $(p-1)^2(q-1)^2$ are computed, yielding

$$\begin{aligned}
d_1 &= 774930028891924293546992534956675173489841560422075579,\\
d_2 &= 619944023113539434808038454181551463822580469577604 2919.
\end{aligned}$$

Observe that d_1 and d_2 are of magnitude $N^{0.6006}$ and $N^{0.6107}$, respectively, which are greater than the maximal bound for small secret exponent attacks in the single-exponent setting, namely $N^{2-\sqrt{2}} \approx N^{0.58578}$.

Additional examples for large values of N and e are provided in Appendix A.

5. Conclusion

In this work, we developed a lattice-based cryptanalytic framework targeting the Nitaj–Seck cryptosystem defined by the relation $ed - k(p-1)^2(q-1)^2 = 1$, where the modulus was given by $N = pq$. Our analysis focused on a setting involving two public encryption exponents. We assumed that

the underlying prime factors shared a certain amount of their least significant bits, while the corresponding private exponents shared part of their most significant bits. Exploiting this structural information, we combined Coppersmith's small-root technique with lattice basis reduction methods to establish new bounds on the secret parameters that guaranteed the success of the attack. Our findings indicated that attacking two encryption exponents significantly extended the range of vulnerable parameters compared to the classical single-exponent setting.

Appendix A More Examples for Larger Values of N and e

We implement the attack presented in Corollary 2 for instances involving large public-key parameters (N, e). For each selected instance, we solve the polynomial equation

$$h(s_1, s_2, s_3, s_4) = 0,$$

defined in (4) following the procedure described in the Theorem 3. In all tested cases, the proposed method successfully factors the RSA modulus $N = pq$ within a practical amount of computational time. The obtained results further show that our method can recover private exponents greater than $N^{2-\sqrt{2}} \approx N^{0.586}$ under the extra information that the RSA primes share an amount of their LSBs. Consequently, these instances lie beyond the range covered by the bounds established for the previously known attack [20]. After performing the experiments, we observe that our method does not attain the upper bound for δ given in Corollary 2. However, it improves the optimal bound for the small private exponent attacks against the scheme of Nitaj and Seck [18] under the following settings. When the primes share only a few bits of their LSBs, our method can recover private exponents of magnitude up to $N^{0.61}$. As the number of shared bits increases, our method becomes more effective and achieves improved bounds up to $N^{1.23}$. This behavior can be explained by the nature of the small root

$$(s_1^\star, s_2^\star, s_3^\star, s_4^\star) = (d_1 - d_2, k_1, k_2, v),$$

where

$$v = \frac{p + q - v_0}{2^{2\varepsilon}}.$$

When the number ε of shared bits between the primes decreases, the size of v increases, which negatively affects the size of the root and limits the

applicability of Coppersmith's method. This explains why, in this setting, we obtain a bound of approximately $N^{0.61}$. Conversely, when the number of shared bits increases, the size of v decreases, making the root more suitable for applying Coppersmith's method to find small roots.

The numerical results are collected in Table 3. The notation used in the table is specified below:

- $\text{len}(n)$ denotes the binary bit-length of a positive integer n.
- $\omega = \log_N(2^{\varepsilon})$, where ε denotes the number of shared LSBs of the primes.
- δ_0 is defined by $d_1 \approx d_2 \approx N^{\delta_0}$.
- λ is defined by $e_1 \approx e_2 \approx N^{\lambda}$.
- κ and x are two parameters used to construct the lattice of dimension $\dim(\mathcal{L})$.
- $\mathcal{T}_1$ denotes the time required to perform the LLL reduction.
- $\mathcal{T}_2$ denotes the time required to perform Gröbner basis computation.

References

[1] Boneh, D., Durfee, G.: Cryptanalysis of RSA with private key d less than $N^{0.292}$, Advances in Cryptology-Eurocrypt'99, Lecture Notes in Computer Science Vol. 1592, Springer-Verlag, pp. 1–11 (1999)

[2] Castagnos, G.: An efficient probabilistic public-key cryptosystem over quadratic fields quotients. Finite Fields and Their Applications, **13**(3), pp. 563–576, Elsevier, (2007)

[3] Chnioune, B., Rahmani, M., Nitaj, A., Ziane, M.: A Novel Cryptanalysis of the Cubic Pell RSA Variant. Axioms, **15**(5), 347 (2026).

[4] Coppersmith, D.: Small solutions to polynomial equations, and low exponent RSA vulnerabilities. Journal of Cryptology, 10(4), pp. 233–260 (1997)

[5] Delaurentis, J. M. A further weakness in the common modulus protocol for the RSA cryptoalgorithm. Cryptologia, 8(3), 253-259, (1984)

len(N)	len(e_i)	ω	δ_0	λ	κ	x	dim($\mathcal{L}$)	$\mathcal{T}_1$	$\mathcal{T}_2$
400	798	0.062708	0.6028	1.9994	2	7	100	2.23 min	19.8 sec
500	999	0.070038	0.6195	1.9981	2	7	100	2.69 min	23.56 sec
600	1197	0.071840	0.6341	1.9991	2	6	90	2.38 min	31.57 sec
700	1396	0.07586	0.6439	1.9972	2	6	90	3.37 min	38.57 sec
800	1600	0.12507	0.7031	1.9999	2	6	90	5.32 min	3.09 min
900	1799	0.12230	0.7224	1.9999	2	6	90	7.48 min	3.91 min
1000	1999	0.15006	0.8027	1.9994	2	6	90	14.30 min	10.14 min
1100	2198	0.18194	0.9124	1.9991	2	6	90	27.11 min	22.38 min
1300	2596	0.21171	1.2310	1.9981	2	5	80	1.15 h	25.95 min

Table 3: Experimental results for various public keys (N, e_i), $i \in \{1, 2\}$.

[6] Elkamchouchi, H., Elshenawy, K., Shaban, H.: Extended RSA cryptosystem and digital signature schemes in the domain of Gaussian integers. In: The 8th International Conference on Communication Systems, 2002. ICCS 2002. vol. 1, pp. 91–95. IEEE (2002)

[7] Hinek, M.J., Lam, C.C.Y., Common modulus attacks on small private exponent RSA and some fast variants (in practice), Journal of Mathematical Cryptology (2009)

[8] Howgrave-Graham, N.: Finding small roots of univariate modular equations revisited, In Cryptography and Coding, LNCS 1355, pp. 131–142, Springer-Verlag (1997)

[9] Howgrave-Graham, N., Seifert, JP. Extending Wiener's Attack in the Presence of Many Decrypting Exponents. In: Secure Networking — CQRE [Secure]'99. CQRE 1999. Lecture Notes in Computer Science, vol 1740. Springer, Berlin, Heidelberg, (1999)

[10] Jochemsz, E., May, A.: A strategy for finding roots of multivariate

polynomials with new applications in attacking RSA variants, in: ASIACRYPT 2006, LNCS 4284, 2006, pp. 267–282, Springer-Verlag (2006)

[11] Kuwakado, H., Koyama, K., Tsuruoka, Y.: A New RSA-type scheme based on singular cubic curves with equation $y^2 \equiv x^3 + bx^2 \pmod{N}$. IEICE TRANSACTIONS on Fundamentals of Electronics, Communications and Computer Sciences, **78**(1), pp. 27–33, The Institute of Electronics, Information and Communication Engineers (1995)

[12] Lenstra, A.K., Lenstra, H.W., Lovász, L.: Factoring polynomials with rational coefficients, Mathematische Annalen, Vol. 261, pp. 513–534, (1982)

[13] May, A.: New RSA Vulnerabilities Using Lattice Reduction Methods. PhD thesis, University of Paderborn (2003).

[14] Moore, J.H., Protocol failures in cryptosystems, Contemporary Cryptology: The Science of Information Integrity, G. J. Simmons, ed., IEEE, New York, 1991, pp. 541-558 (1991)

[15] Murru N., Saettone F.M.: A Novel RSA-Like Cryptosystem Based on a Generalization of the Rédei Rational Functions. In: Kaczorowski J., Pieprzyk J., Pomykala J. (eds) Number-Theoretic Methods in Cryptology. NuTMiC 2017. Lecture Notes in Computer Science, vol 10737. Springer, Cham (2018)

[16] Nitaj, A., Ariffin, M.R.K., D.I. Nassr, Bahig, H.M.: New attacks on the RSA cryptosystem, in D. Pointcheval and D. Vergnaud (Eds.): AFRICACRYPT 2014, LNCS 8469, pp. 178198, 2014. Springer (2014)

[17] Nitaj, A.: Another generalization of Wiener's attack on RSA, In: Vaudenay, S. (Ed.) Africacrypt 2008. LNCS, vol. 5023, 1740. Springer, Heidelberg (2008)

[18] Nitaj, A., Seck, M.: *A New Public Key Encryption Scheme Based on the Cubic Pell Curve Using Encoding Functions,* Moroccan Journal of Algebra and Geometry with Applications (2024).

[19] Rahmani, M., Nitaj, A., Ziane, M.: Further cryptanalysis of some variants of the RSA cryptosystem. Journal of Applied Mathematics and Computing, 1-31 (2024).

[20] Rahmani, M., Nitaj, A.: Improved Cryptanalysis of an RSA Variant Based on Cubic Pell Curve. In International Conference on Cryptology in Africa. Cham: Springer Nature Switzerland, pp. 113-125, (2025).

[21] Rivest, R., Shamir, A., Adleman, L.: A Method for Obtaining digital signatures and public-key cryptosystems, Communications of the ACM, Vol. 21 (2), pp. 120–126 (1978)

[22] Said, M.R.M., Loxton, J.: A cubic analogue of the RSA cryptosystem. Bulletin of the Australian Mathematical Society, **68**(1), pp. 21–38, (2003)

[23] Sarkar, S., Maitra, S.: Cryptanalysis of RSA with two decryption exponents. Information Processing Letters, 110(5), 178-181 (2010).

[24] Seck, M., Tchapgnouo, H. B.: A generalized partial exposure lattice attack against an RSA variant based on cubic Pell curves. Discrete Applied Mathematics, 386, 334-344 (2026).

[25] Simmons, G. J. A 'weak' privacy protocol using the RSA cryptalgorithm. Cryptologia, 7, 180-182 (1983)

[26] Wiener, M.: Cryptanalysis of short RSA secret exponents, IEEE Transactions on Information Theory,**36**(3), 553–558 (1990)

[27] Zheng, M., Kunihiro, N., Yao, Y.: Cryptanalysis of the RSA variant based on cubic Pell equation. Theor. Comput. Sci. 889, pp. 135–144 (2021)