Cutting dimensions in the LLL attack for the ETRU post-quantum cryptosystem

Augusto Miguel Camillo Silva[1], **Thiago do Rêgo Sousa**[2], **Tertuliano Souza Neto**[2]

[1]Universidade Federal de Juiz de Fora (UFJF) - Juiz de Fora - MG

[2]CEPESC, Agência Brasileira de Inteligência, Brasília - DF

Abstract. *NTRU is one of the most important post-quantum cryptosystems nowadays, based on polynomial rings with coefficients in $\mathbb{Z}$. Among its variants, the ETRU cryptosystem utilizes Eisenstein integers $\mathbb{Z}[\omega]$, where ω is a primitive cube root of unity. We explore this cryptosystem and introduce a new lattice based on May's technique, which proposes reducing the original lattice dimension to enable attacks with increased complexity. This new lattice allowed us to recover the private key of the ETRU system for a dimension that was not yet possible using current lattice reduction techniques over the original lattice.*

1. Introduction

Public key cryptography, initially proposed by Diffie and Hellman [Diffie W 1976], led to the development of various cryptographic systems, which currently protect a significant portion of digital communication. With the advancement of quantum computing, these classical public key cryptography algorithms like RSA [Rivest RL. 1978], elliptic curve cryptography (ECC) [Neal 1987], and Digital Signature Algorithm (DSA) [NIST 2019], are at risk of being broken in polynomial time by quantum computers using algorithms like Shor's [Shor 1994]. Post-quantum cryptography aims to ensure cryptographic security in the era of quantum computing.

The Post-Quantum Cryptography Standardization [NIST] is an initiative led by the National Institute of Standards and Technology in the United States with the goal of developing cryptographic standards that are secure against attacks from quantum computers. As of the latest update, the process was in its third round, with a reduced list of candidates considered for final standardization, including algorithms like Kyber, NTRU, and Classic McEliece.

The NTRU system was introduced by Hoffstein, Pipher, and Silverman, is an efficient public key cryptosystem based on polynomial rings with integer coefficients [Hoffstein et al. 1998]. NTRU is notable for its arithmetic operations of quadratic complexity, being significantly faster than RSA and ECC. It is based on the difficulty of solving certain lattice problems, such as finding the shortest vector in a convolutional lattice [May A 2001], making it resistant to quantum attacks. However, NTRU may suffer from decryption failures, although proper parameter selection can mitigate this issue.

Several variants of NTRU have been proposed to improve its security and efficiency, such as GNTRU which uses Gaussian integers, CTRU which is based on binary fields, QTRU using quaternion algebra, ETRU based on Eisenstein integers, among others [Sonika Singh 2016].

The ETRU system was introduced in [Monica Nevins 2010] as an extension of the original NTRU. A subsequent work by [Katherine Jarvis 2015] highlighted its superior speed, smaller key sizes, and simplicity in binary messaging compared to NTRU. Khaterine's study also compared the efficiency and security of ETRU and NTRU against meet-in-the-middle attacks and lattice attacks. In [Karbasi and Atani 2015], a new system called ILTRU was introduced as an extension of ETRU, exploiting properties of structured lattices to achieve high efficiency and security based on ideal lattices, with the established hardness of R-SIS[Lyubashevsky and Micciancio 2009] and R-LWE[Regev 2009] problems.

Subsequently, [S. Lyu and Ling 2020] deepened the understanding of the characteristics of algebraic lattices, emphasizing the appropriate design and performance limits of reduction algorithms. [Zhu and Tian 2021] compare the performance and security of NTRU and ETRU signature algorithms and argue that ETRU is faster.

On the security side of the ETRU system, [Katherine Jarvis 2015] proposed a lattice base attack on the private key that requires using base reduction techniques over a basis of dimension $4n$, where $n-1$ is the degree of the ETRU polynomials used to construct the system. We extend the original attack by introducing a new lattice that has vectors of smaller dimensions as was done in [May 1999] for the original NTRU system. We observe through a simulation study that, even after reducing the original lattice dimension, it is still possible to find the system's private key. The proposed attack can, in fact, recover the private key for $n = 61$ on a personal computer, something that was currently not feasible using the original lattice from [Katherine Jarvis 2015]. This suggests that ETRU has a lower level of security than expected by using its original lattice, highlighting the need for further analysis when choosing its parameter to ensure adequate security.

The rest of the paper is organized as follows. In Section 2 we introduce the NTRU system and in Section 3 its ETRU variant. The proposed key recovery attack using a lattice with a smaller dimension is developed in Section 4 and Section 5 concludes giving directions for further research.

2. The NTRU Cryptosystem

The NTRU public key cryptosystem depends on three integer parameters (n, p, q) such that $n \geq 1$, $\gcd(n, q) = \gcd(p, q) = 1$ and q is much larger than p. The primary arithmetic operations in the NTRU cryptosystem involve computations over polynomials defined in the rings $\mathcal{R}$, $\mathcal{R}_p$, and $\mathcal{R}_q$ as follows:

$$\mathcal{R} = \frac{\mathbb{Z}[x]}{(x^n - 1)}, \mathcal{R}_p = \frac{(\mathbb{Z}/p\mathbb{Z})[x]}{(x^n - 1)}, \mathcal{R}_q = \frac{(\mathbb{Z}/q\mathbb{Z})[x]}{(x^n - 1)}.$$

It can be observed that the ring $\mathcal{R}$ is associated with the other two rings. Specifically, for any polynomial $a(x)$ in $\mathcal{R}$, it can be associated with an element in $\mathcal{R}_p$ or $\mathcal{R}_q$ by reducing its coefficients modulo p or q, respectively.

A polynomial $a(x) \in \mathcal{R}$ is termed a ternary polynomial if its coefficients belong to the set $\{-1, 0, 1\}$. In addition, $a(x)$ can be associated with an element in $\mathcal{R}_p$ or $\mathcal{R}_q$ by reducing its coefficients modulo p or q, respectively

Given d_1 and d_2 positive integers. We define $\mathcal{T}(d_1, d_2)$ as the subset of ternary polynomial in $\mathcal{R}$ as follows:

$$\mathcal{T}(d_1, d_2) = \left\{ \begin{array}{l} a(x) \in \mathcal{R}, \\ a(x) \text{ has } d_1 \text{ coefficients equal to } 1, \\ a(x) \text{ has } d_2 \text{ coefficients equal to } -1, \\ \text{remaining coefficients of } a(x) \text{ are } 0. \end{array} \right\}$$

In the NTRU cryptosystem, operating with parameters (n, p, q, d), key generation, encryption and decryption are defined as follows:

1. **Key Generation:** Generate two ternary polynomials at random, $f(x) \in \mathcal{T}(d+1, d)$ and $g(x) \in \mathcal{T}(d, d)$ such that there exist two polynomials $f_p(x) \in \mathcal{R}_p$ and $f_q(x) \in \mathcal{R}_q$ satisfying $f(x)f_p(x) = 1 \in \mathcal{R}_q$ and $f(x)f_q(x) = 1 \in \mathcal{R}_q$. Then compute the polynomial:

   $$h(x) = f_q(x) * g(x) \in R_q,$$

   where $*$ denotes polynomial multiplication in R_q, i.e., a cyclic convolution product as defined in [Hoffstein et al. 1998, Section 1.1]. The polynomial $h(x)$ is the public key and the pair $(f(x), f_p(x))$ is the private key.

2. **Encryption:** Let $m(x) \in \mathcal{R}_p$ be a plaintext and choose, at random, a ternary polynomial $r(x) \in \mathcal{T}(d, d)$. The encrypted message is:

   $$e(x) \equiv ph(x) * r(x) + m(x) \mod q.$$

   Notice that the ciphertext $e(x)$ belongs to the ring $\mathcal{R}_q$.

3. **Decryption:** To decrypt the ciphertext first compute:

   $$a(x) \equiv f(x) * e(x) \mod q.$$

   Then the reduction modulo p gives the desired plaintext

   $$b(x) \equiv f_p(x) * a(x) \mod p.$$

Due to the randomness of the polynomial $r(x)$, NTRU operates as a probabilistic cryptosystem. This means that a message $m(x)$ can be encrypted into multiple ciphertexts $ph(x) \cdot r(x) + m(x)$, each depending on the particular instance of $r(x)$. However, this introduces a potential vulnerability in the NTRU cryptosystem, as certain ciphertexts may fail to decrypt correctly back to the original message, a scenario referred to as decryption failure. Attacks documented in the literature exploit such decryption failures [Howgrave-Graham et al. 2003, Gama and Nguyen 2007, Jaulmes and Joux 2000], underscoring the necessity for careful parameter selection.

3. The ETRU Cryptosystem

ETRU is a lattice-based cryptosystem which is a variant of NTRU, constructed using truncated polynomials with coefficients in the ring of Eisenstein integers $\mathbb{Z}[\omega]$. The ring of Eisenstein integers is the set of complex numbers of the form $a + b\omega$ with $a, b \in \mathbb{Z}$, where ω is a primitive cube root of unity.

3.1. Describing ETRU

Let q be a nonzero element of $\mathbb{Z}[\omega]$. The set $(q) = \{rq \mid r \in \mathbb{Z}[\omega]\}$ forms the ideal in $\mathbb{Z}[\omega]$ generated by q. Denote by $\mathbb{Z}_q[\omega]$ the set of residue classes of the quotient ring $\mathbb{Z}[\omega]/\langle q \rangle$. For instance, $\mathbb{Z}_2[\omega]$ represents a field with four elements, namely, $\{0, 1, \omega, \omega + 1\}$. For any $z = a + b\omega \in \mathbb{Z}[\omega]$ we can define its norm by

$$|z| = a^2 - ab + b^2.$$

To reduce the probability of decryption failure, we shall represent the set of residues centered around 0, i.e., for an integer n, we have

$$\mathbb{Z}_n = \begin{cases} \left\{-\dfrac{n-1}{2}, \cdots, \dfrac{n-1}{2}\right\} & \text{if } n \text{ is odd,} \\ \left\{-\dfrac{n}{2} + 1, \cdots, \dfrac{n}{2}\right\} & \text{if } n \text{ is even.} \end{cases}$$

To set the ETRU parameters, we initially select two relatively prime elements in $\mathbb{Z}[\omega]$, p and q such that $|q|$ is much larger than $|p|$. This is necessary to make the polynomial inversion algorithms modulo p and q more efficient. It is preferable to choose both elements as primes or powers of primes. We choose a positive integer n (preferably prime) and set:

$$\mathcal{R} = \frac{\mathbb{Z}[\omega][x]}{(x^n - 1)}; \qquad \mathcal{R}_p = \frac{\mathbb{Z}_p[\omega][x]}{(x^n - 1)}; \qquad \mathcal{R}_q = \frac{\mathbb{Z}_q[\omega][x]}{(x^n - 1)} \tag{1}$$

Note that an element $f \in \mathcal{R}$ is a polynomial $f_0 + f_1 x + ... + f_{n-1} x^{n-1}$ where each coefficient f_i is an Eisenstein integer $f_i = a_i + b_i\omega$. Similarly, a polynomial $f \in \mathcal{R}_p$ (or $\mathcal{R}_q$) if and only if each coefficient $f_i \in \mathbb{Z}_p[\omega]$ (or $\mathbb{Z}_q[\omega]$). We define a rotation of $f \in \mathcal{R}$ as the polynomial

$$x^k f(x) = f_0 x^k + f_1 x^{k+1} + ... + f_{n-1} x^{n+k-1} \in \mathcal{R},$$

for an integer $k \in \mathbb{Z}$.

We've chosen $p = 2$ throughout the process, which offers numerous advantages in encoding binary messages into elements of $\mathcal{R}_p$, as mentioned in [Katherine Jarvis 2015].

Fixing $0 < r < 1$, we define the sets $\mathcal{L}_f$, $\mathcal{L}_q$, and $\mathcal{L}_\varphi$ of polynomials as the subsets of $\mathcal{R}$ containing approximately nr non-zero coefficients selected from the set $\mu_6 = \{\pm 1, \pm\omega, \pm\omega^2\}$ of units of $\mathbb{Z}[\omega]$ as follows.

Let k be the nearest integer to nr. The polynomials in $\mathcal{L}_f$ consist of all polynomials with k nonzero entries.

The polynomials in $\mathcal{L}_q$ and $\mathcal{L}_\varphi$ should be divisible by $x - 1$ modulo q. To achieve this, we select s, the nearest multiple of three to rn, and randomly pick s tuples of coefficients, each being $\pm\{1, \omega, \omega^2\}$. These tuples are then distributed across the coefficients while preserving the order of the chosen tuples.

For fixed n, p, q,r key generation, encryption and decryption in the ETRU system work as follows:

1. **Key Generation:** To generate the keys, choose two random polynomials $f(x) \in \mathcal{L}_f$ and $g(x) \in \mathcal{L}_g$. The polynomial $f(x)$ must have inverses mod p and mod q. Let $f_p(x) \in \mathcal{R}_p$ and $f_q(x) \in \mathcal{R}_q$ be the inverses of $f(x)$ under mod p and mod q, respectively. Thus, $f_p(x) \cdot f(x) \equiv 1 \mod p$ and $f_q(x) \cdot f(x) \equiv 1 \mod q$. Then, calculate $h(x) = f_q(x) \cdot g(x) \mod q$.
   The pair of polynomials $(f(x), f_p(x))$ is the private key and the polynomial $h(x)$ is the public key in the ETRU cryptosystem.

2. **Encryption:** To encrypt a message $m(x) \in \mathcal{R}_p$ we first choose a random polynomial $\varphi(x) \in \mathcal{L}_\varphi$ and then compute:
$$e(x) = p\varphi(x) \cdot h(x) + m(x) \mod q$$
   The polynomial $e(x) \in \mathcal{R}_q$ is the ciphertext.

3. **Decryption:** To decrypt the ciphertext $e(x)$ compute:
$$a(x) = f(x) \cdot e(x) \mod q$$
$$m(x) = f_p(x) \cdot a(x) \mod p$$
   We claim that the polynomial m is the original message that was encrypted above. In fact,
$$\begin{aligned} a(x) &= f(x) \cdot e(x) \mod q \\ &= f(x) \cdot (p\varphi(x) \cdot h(x) + m(x)) \mod q \\ &= pf(x) \cdot \varphi(x) \cdot h(x) + f(x) \cdot m(x) \mod q \end{aligned}$$
   The last equation holds if the coefficients of $pf \cdot \varphi \cdot h + f \cdot m$ are sufficiently small such that their values remain unchanged when reduced mod q. Now we compute
$$\begin{aligned} & f_p(x) \cdot a(x) \mod p \\ &= f_p(x) \cdot (pf(x) \cdot \varphi(x) \cdot h(x) + f(x) \cdot m(x)) \mod p \\ &= f_p(x) \cdot pf(x) \cdot \varphi(x) \cdot h(x) + f_p(x) \cdot f(x) \cdot m(x) \mod p \\ &= 0 + f_p(x) \cdot f(x) \cdot m(x) \mod p \\ &= m(x) \mod p \end{aligned}$$

In the next Section, we show how a key recovery attack is constructed for the ETRU system using lattices. After that, we introduced a new lattice that has a smaller dimension and can still be used for attacking the ETRU private key.

4. Key recovery attack

The coefficients of the public key $h(x)$ satisfy $f(x) * h(x) \equiv g(x) \mod q$. Thus, we can attack ETRU by solving this equivalence similarly to how we attack NTRU using lattices ([Hoffstein et al. 1998, Section 3.4]).

In the case of ETRU, a key $h(x)$ with parameters (n, q, p, r) generates a $4n-$dimensional lattice as we will show bellow. The vector corresponding to the pair of private keys (f, g) is a short vector in this lattice (in terms of the Euclidean norm), and therefore, we could discover the private key by finding a sufficiently short vector in the lattice. To achieve this, we use lattice basis reduction techniques, such as the LLL (Lenstra-Lenstra-Lovász) algorithm [Lenstra et al. 1982] or the BKZ (Block Korkin-Zolotarev) algorithm [Chen and Nguyen 2011].

4.1. ETRU lattice

Consider the isomorphism of additive groups $\varphi : \mathbb{Z}[\omega] \to \mathbb{Z}^2$ given by

$$\varphi(\alpha) = \varphi(a + b\omega) = (a, b).$$

For each element $\alpha \in \mathbb{Z}[\omega]$, we define the matrix $\langle\alpha\rangle$ that performs right multiplication in $\mathbb{Z}^2$

$$\langle\alpha\rangle = \begin{bmatrix} a & b \\ -b & a-b \end{bmatrix}$$

Let M be an $n \times n$ matrix with entries in $\mathbb{Z}[\omega]$. We define $\langle M\rangle$ as the $2n \times 2n$ matrix over $\mathbb{Z}$ by replacing each entry a_{ij} of M with $\langle a_{ij}\rangle$. Similarly, for any polynomial $f \in \mathcal{R}$, we define $\langle f\rangle$ as the application of the same operation over each coefficient of f.

For a given public key h for the ETRU system with parameters (n, q, p, r), let H represent the matrix formed by the coefficients of h and its $n-1$ rotations. Therefore,

$$\langle H\rangle = \begin{pmatrix} \langle h\rangle \\ \langle xh\rangle \\ \vdots \\ \langle x^{n-1}h\rangle \end{pmatrix}.$$

Then, the lattice of ETRU is defined as follows

$$L_{ETRU} = \begin{bmatrix} I_{2n} & \langle H\rangle \\ 0 & \langle qI_{2n}\rangle \end{bmatrix}, \tag{2}$$

where I_{2n} is the $2n$-dimensional identity matrix.

In ETRU, the private keys are associated with short vectors within L_{ETRU}. Indeed, the target vector (f, g) containing the private key associated with h, can be written as a linear combination of the rows of the matrix L_{ETRU} from (2)([Monica Nevins 2010, Section 8.1]). In other words, (f, g) belongs to the lattice generated by L_{ETRU}.

Using Gaussian heuristic [Nguyen 2010], the shortest vector expected from a lattice L of dimension N has length:

$$l = \sqrt{\frac{N}{2\pi e}} v^{1/N} \quad \text{where } v = det(L).$$

Thus, the shortest expected vector of the lattice L_{ETRU} of dimension $4n$ has length:

$$l_{ETRU} = \sqrt{\frac{4n}{2\pi e}} v^{1/4n} = \sqrt{\frac{4n}{2\pi e}} |q|^{2n/4n} = \sqrt{\frac{2n|q|}{\pi e}}.$$

The keys f and g each have rn non-zero entries in $\mu_6 = \{\pm 1, \pm\omega, \pm\omega^2\}$. In the lattice, each coefficient is viewed as $\{(\pm 1, 0), (0, \pm 1), (\pm 1, \pm 1)\}$. Thus, the norm of the target vector (f, g) lies between $\sqrt{2rn}$ and $\sqrt{4rn}$.

In cases where the norm of a vector (f, g) is maximized and the $|q|$ is sufficiently large, we obtain $\sqrt{4rn}$, which remains smaller than the expected shortest vector in the lattice. Hence, the likelihood of the pair (f, g) being found in the reduced basis lattice is high.

4.2. New ETRU lattice attack

The use of the BKZ (Block Korkin-Zolotarev) algorithm for lattice basis reduction becomes ineffective as the lattice dimension increases. This is because the execution time and complexity of BKZ grow exponentially with the dimension. To deal with this, we can apply May's idea [May 1999], which proposes reducing the lattice dimension. Specifically, May's idea involves cutting some coordinates of the vector g to reduce the problem's dimension for the NTRU system of [Hoffstein et al. 1998]. In what follows, we adapt this idea to attack the ETRU system in dimensions larger than those possible so far with the original lattice from (2).

Given the public key h and the corresponding lattice L_{ETRU}, we can define a new lattice L'_{ETRU} by excluding $k < 2n$ columns from the submatrix $\langle H \rangle$. Let $\langle H \rangle_k$ be the columns of $\langle H \rangle$ that are kept. The new lattice L'_{ETRU} can be expressed as:

$$L'_{ETRU} = \begin{bmatrix} I_{2n} & \langle H \rangle_k \\ 0 & \langle qI_{2n-k} \rangle \end{bmatrix} \tag{3}$$

Thus, the shortest expected vector of the lattice L'_{ETRU} of dimension $4n - k$ has a length of:

$$l'_{ETRU} = \sqrt{\frac{4n-k}{2\pi e}} v^{1/(4n-k)} \quad \text{where } v = |q|^{\frac{2n-k}{4n-k}}$$

Let g_k be the polynomial g with k coordinates removed. If $k < rn$, the norm of the target vector (f, g_k) lies between $\sqrt{2rn}$ and $\sqrt{4rn}$. If $k > rn$, the shortest vector has a norm smaller than $\sqrt{4rn}$.

[May 1999]'s attack works since the truncated short vector (lets say by removing the last coefficients of the g polynomial) is still a linear combination of the lattice vectors (truncated at the same positions that we removed from g). In fact, one could simply choose k coefficients of the polynomial g and remove it to create a new lattice and solve the SVP problem. This approach speeds up the search for the shortest vectors since the computational time for running BKZ is roughly proportional to the dimension of the lattice.

Removing columns to reduce the dimension of the lattice L'_{ETRU} may introduce some inaccuracies in the lattice structure. However, for specific values of the parameters n and q, this approximation is still effective for identifying short vectors that correspond to the private keys (f, g_k). Therefore, despite the introduced inaccuracies, the reduced lattice remains sufficiently informative to enable the identification of the private keys. The dimension of the reduced lattice L'_{ETRU}, although this approximation introduces some inaccuracies, for certain values of n and q it is sufficient to find a short vector that corresponds to the private keys (f, g_k).

The removal of columns can be done randomly, but the effectiveness of this attack can vary significantly depending on which columns are removed. The goal of May's attack is to reduce the dimension of the lattice without losing the vectors that contain the necessary information for the attack. If the wrong columns are chosen, the resulting reduced lattice may not exhibit the properties needed for a successful attack. In this work, the exclusion of the columns was done on the right side of the matrix $\langle H \rangle$.

Algorithm to find the private key:

Input: Integers n, q, a public key h for the ETRU system and a cut parameter $k > 0$.

Output: A set of potential private keys f^* corresponding to h.

1. Use h, q, n and k to construct the corresponding lattice L'_{ETRU} from (3). This is a matrix where the upper right corner is formed by the coefficients of the polynomial h where every line is just a circular shift by one of the previous line.
2. Apply the BKZ algorithm to reduce the lattice basis and get a matrix L'_{reduced} of dimension $4n \times (4n - k)$.
3. Use the first $2n$ coordinates of every line of L'_{reduced} to construct a list of vectors.
3. For each vector in the list of vectors from step 3 and each value $r \in \{0, 1, 2, \ldots, 2n - 1\}$, create a list of potential keys by rotating the vector r positions.

4.3. Experimental results

In [Monica Nevins 2010], the BKZ algorithm was applied to find a vector with the same norm as the target vector (f, g) corresponding to the private key of the ETRU system, in order to assess its security against lattice attacks. The results reported in their paper showed the viability of the attack for a fixed $q = 383$ and $n \leq 57$. In the highest degree achieve, i.e., $n = 57$, the success rate of the attack is about 20% percent. The authors reported that after $n = 57$, the BKZ attack using the original ETRU lattice from (2) consistently fails and we also observed this phenomenon in our simulation studies when trying to run for $n > 57$.

To go beyond $n = 57$, our strategy is to use the new lattice developed in Section 4.2. With a $k > 0$, the lattice dimension drops from $4n$ to $4n - k$ and we can hope to successfully execute BKZ in order to find the private key pair.

The results are reported in Table 1 where for each value of $n \in \{41, 47, 57, 61\}$ and the BKZ block used to run the attack. We report the value k used for cutting the dimension of the lattice and the success rate of the attack in 100 experiments, usually for $k \in \{1, \ldots, 2n - 1\}$ The time to run the attack is closely related to the lattice dimension $4n - k$ and the BKZ block. For small values of n we can use a block of 10 in the BKZ algorithm and it works for finding the private key. On the other hand, for $n = 57$ and 61 we needed to increase the block to 20, since running BKZ with a block of 10 did not succeeded in finding the private key.

For $n = 41$, running the attack with a cut of 59 already returned a success rate of 1%, in which case the dimension of the lattice decrease from 164 to 105, which is a great improvement in the complexity of the attack and shows the usefulness of [May 1999]'s idea. When we decrease the cut to 50 the success rate drastically increases to 69% and it achieves 100% for a cut of 21.

For the case where $n = 61$ we experimented with $k \in \{30, 31, 2n - 1\}$, since running BKZ in the lattice L'_{ETRU} with $k < 30$ is very likely to fail. For $k = 41$ and in case BKZ runs without error, the average time taken per experiment is less than 2 minutes. For each experiment we generated a new key pair and applied the algorithm of Section 4.2, recording whether or not we found the correct key in the list of potential keys f^*. Out of 100 independent experiments, we found the correct key in 3 of them,

n	41	47	57	61
BKZ block	10	10	20	20
cut k (sucess)	54 (6%)	54 (3%)	59 (1%)	49 (1%)
	50 (69%)	50 (53%)	50 (51%)	45 (7%)
	43 (94%)	43 (88%)	45 (94%)	*
	21 (100%)	27 (96%)	*	*

Table 1. Matrix NTRU private key attack for varying n and fixed $q = 383$. For some cut values we reported the sucess rate of the attack over 100 experiments.

indicating the usefulness of the L'_{ETRU} lattice in finding the private key. To run the attack successfully we needed to use a BKZ block of 20, which slows the basis reduction process but has the advantage of returning a shorter basis for the lattice. The success rate of the algorithm decreases as we increase the value of n, and this is due to the fact that running BKZ would need more computational resources. On the other hand, the attack presented here shows that the dimension reduction can be used to improve BKZ in this setup. This means that we do not need to work with the complete ETRU lattice to find the private key and this should be taken into account when analyzing its real security against lattice attack in a similar way that was done for the NTRU NIST submission in [Daniel J. Bernstein et al. 2024].

5. Conclusion

The experimental results indicate that the approach using the new lattice developed based on May's technique is promising for extending the feasibility of key recovery attacks on the ETRU system to larger dimensions.

With this new lattice, we were able to find the private key in some of the simulations conducted for a dimension of $n = 61$, which was not possible using the current lattice reduction techniques on the original lattice. This suggests that the dimension reduction of the lattice can be a valid attack strategy, paving the way for future investigations and improvements.

However, it is important to note that the use of a larger BKZ block resulted in a significant reduction in the algorithm's running time, and despite the observed improvements, we encountered limitations imposed by the scalability of the BKZ algorithm in larger dimensions.

Therefore, for future directions, it would be interesting to explore how advanced basis reduction techniques, such as the BKZ algorithm, can be adapted to take full advantage of this modified lattice. Additionally, the development of an algebraic BKZ using (see [Lyu et al. 2020]) the new lattice could be a promising area for future research.

References

Chen, Y. and Nguyen, P. Q. (2011). Bkz 2.0: Better lattice security estimates. In *International Conference on the Theory and Application of Cryptology and Information Security*, pages 1–20. Springer.

Daniel J. Bernstein, Tanja Lange, Chitchanok Chuengsatiansup, and Peter Schwabe (Accessed: 2024). NTRU Prime. `https://ntruprime.cr.yp.to`. Website.

Diffie W, H. M. (1976). New directions in cryptography. *In IEEE Transactions on Information Theory*, 22:644–654.

Gama, N. and Nguyen, P. Q. (2007). New chosen-ciphertext attacks on ntru. In *International Workshop on Public Key Cryptography*, pages 89–106. Springer.

Hoffstein, J., Pipher, J., and Silverman, J. H. (1998). Ntru: A ring-based public key cryptosystem. In *International algorithmic number theory symposium*, pages 267–288. Springer.

Howgrave-Graham, N., Nguyen, P. Q., Pointcheval, D., Proos, J., Silverman, J. H., Singer, A., and Whyte, W. (2003). The impact of decryption failures on the security of ntru encryption. In *Annual International Cryptology Conference*, pages 226–246. Springer.

Jaulmes, É. and Joux, A. (2000). A chosen-ciphertext attack against ntru. In *Annual international cryptology conference*, pages 20–35. Springer.

Karbasi, A. H. and Atani, R. E. (2015). Iltru: An ntru-like public key cryptosystem over ideal lattices. *International Association for Cryptologic Research*, Cryptology ePrint Archive:549–558.

Katherine Jarvis, M. N. (2015). Etru: Ntru over the eisenstein integers. *Designs, Codes and Cryptography*, 74:219–242.

Lenstra, A. K., Lenstra, H. W., and Lovász, L. (1982). Factoring polynomials with rational coefficients. *Mathematische annalen*, 261:515–534.

Lyu, S., Porter, C., and Ling, C. (2020). Lattice reduction over imaginary quadratic fields. *IEEE Transactions on Signal Processing*, 68:6380–6393.

Lyubashevsky, V. and Micciancio, D. (2009). On bounded distance decoding, unique shortest vectors, and the minimum distance problem. *Lecture Notes in Computer Science*, 5677:450–461.

May, A. (1999). Cryptanalysis of ntru. *unpublished*.

May A, S. J. (2001). Dimension reduction methods for convolutional modular lattices. *Lecture Notes in Computer Science*, 2146:110–125.

Monica Nevins, Camelia KarimianPour, A. M. (2010). Ntru over rings beyond z. *Designs, Codes and Cryptography*, 56:65–78.

Neal, K. (1987). Elliptic curves cryptosystems. *Mathematics of Computation*, 48:203–209.

Nguyen, P. Q. (2010). Hermite's constant and lattice algorithms, the lll algorithm: Survey's and applications. *Information Security and Cryptography*, pp:16–69.

NIST. Post-Quantum Cryptography Standardization. https://csrc.nist.gov/Projects/post-quantum-cryptography/post-quantum-cryptography-standardization.

NIST (2019). Digital Signature Standard (DSS). https://csrc.nist.gov/pubs/fips/186-5/ipd.

Regev, O. (2009). On lattices, learning with errors, random linear codes, and cryptography. *Journal of ACM*, 56 no. 6:no. 6.

Rivest RL., Shamir A., A. L. (1978). A method for obtaining digital signatures and public key cryptosystem. *Communications of the ACM*, 21:120–126.

S. Lyu, C. P. and Ling, C. (2020). Lattice reduction over imaginary quadratic fields. *IEEE Transactions on Signal Processing*, 68:6380–6393.

Shor, P. (1994). Algorithms for quantum computation: Discrete logarithms and factoring. *Proceedings 35th Annual Symposium on Foundations of Computer Science*, page 124–134.

Sonika Singh, S. P. (2016). Generalizations of the ntru cryptosystem. 9:4823–6411.

Zhu, Z. and Tian, F. (2021). Comparison and intelligent analysis of ntru and etru signature algorithms for public key digital signature. *Journal of Physics: Conference Series*, 2083 no. 4:42009.