Security analysis of DBTRU cryptosystem

Alexandra Ciobanu
Faculty of Computer Science
"A. I. Cuza" University of Iasi
alexandra.ciobanu2398@gmail.com

Marina Stefiuc
Faculty of Computer Science
"A. I. Cuza" University of Iasi
stefiucmarina@gmail.com

Abstract

Proposed by Thang and Binh (*NICS, 2015*), DBTRU is a variant of NTRU, where the integer polynomial ring is replaced by two binary truncated polynomial rings $GF(2)[x]/(x^n+1)$. DBTRU has significant advantages over NTRU in terms of security and performance. NTRU is a probabilistic public key cryptosystem having security related to some hard problems in lattices. In this paper we will present a polynomial-time linear algebra attack on the DBTRU cryptosystem which can break DBTRU for all recommended parameter choices and the plaintext can be obtained in less than one second using a single PC and this specific attack.

1 Introduction

The Number Theory Research Unit (NTRU) cryptosystem as a public key cryptosystem was proposed by Hoffstein, Pipher, and Silverman in 1996 and published in 1998 [1]. It was standardized by IEEE in 2008 [2]. In 2020, NTRU entered the third round of submissions in the National Institute of Standards Technology (NIST) post-quantum cryptography standardization process. NTRU works on the integer polynomial ring $\mathbb{Z}[x]/\left(x^n-1\right)$.

The encryption and decryption procedures involve linear operations between ring elements. This characteristic gives NTRU a great advantage over Rivest, Shamir, Adleman (RSA) cryptosystem and elliptic curve cryptosystem (ECC) in terms of computational speed and key size. NTRU can be classified as post-quantum cryptography, and its security is based on the hardness of the shortest vector problem in a certain lattice. Compared with traditional public key algorithms, its research has been a hot spot in the field of public key cryptography. NTRU is widely used in e-commerce, communication, embedded systems, and portable devices [3, 4].

Since 2002, cryptographers have been exploring the optimization of NTRU from the underlying mathematical structure in order to achieve a higher level of security or better performance. Banks et al. gave the non-invertible version in 2002 [5]. This extension can overcome the problem of finding "enough" invertible polynomials in small sets. In 2002, Gaborit et al. proposed CTRU [6], a NTRU-like cryptosystem that runs on $F_2[T][X]/\left(x^n-1\right)$. CTRU can avoid the attacks based on the LLL algorithm, although Vats proved that it is insecure under linear algebra attack in 2008 [7].

1.1 CTRU cryptosystem background

A CTRU cryptosystem depends on an integer N and on two irreducible polynomials P, Q of $A := \mathbb{F}_2[T]$. We shall assume that P and Q are polynomials of respective degrees s and m with $2 \leq s \leq m$, and, last but not least $GCD(m, s) = 1$. We work in the ring $R := A[X]/\left(X^N-1\right)$, of "truncated polynomials with polynomial coefficients". The reader already familiar with NTRU might want to keep in mind the following dictionary.

NTRU	CTRU
$\mathbb{Z}$	A
p	P
q	Q
$\log_2(p)$	s
$\log_2(q)$	m
$\mathbb{Z}[X]/\left(X^N-1\right)$	$A[X]/\left(X^N-1\right)$

Observe that the quotients rings A_P and A_Q of A by the ideals (P) and (Q) respectively are the finite fields $\mathbb{F}_{2^s}$ and $\mathbb{F}_{2^m}$. We denote by R_P, R_Q the quotient rings of R by the ideals (P) and (Q) respectively. By the arithmetic constraint

$GCD(m,s)=1$ we see that $\mathbb{F}_{2^s} \cap \mathbb{F}_{2^m} = \mathbb{F}_2$. Like in NTRU independence of reduction $\mathrm{mod}(P)$ and (Q) is essential to avoid trivial attacks. By $\deg(F)$ we shall denote the degree of F as a polynomial in T.

1.1.1 Security Analysis of CTRU

In [1] four different kinds of attacks are given for CTRU cryptosystem. Given the similarity of structure of the two cryptosystems, the first three attacks are mainly identical and the fourth attack based on lattices is turned into an attack through the Popov normal form of a polynomial matrix.

1. Brute force attack
In the case of a brute force attack an attacker may want to try all possible choices for f and try to find if fh has entries of small degree. By analogy the same attack can also be done against a given message by testing all possible ϕ and searching for $e - \phi h \pmod{Q}$ has coefficients of small degree. Therefore the key security is $\# L_g$ and the message security is $\# L_\phi$. Hence as for NTRU using the meet-in-the-middle attack one has to take the square root.

2. Meet-in-the-middle attack
A meet-in-the-middle attack was proposed by Odlyzko for NTRU and developped by Silverman in [19]. This attack can also be used against this cryptosystem using the same argument on the degree of the polynomials. This attack needs a lot of storage capacity and cut the search time by the usual square root. Hence it means that the set of possible g and ϕ has to contain at least 2^{160} elements in order to obtain a security of 2^{80}.

3. Multiple transmission attacks
If Amanda sends a single message m with different ϕ 's but the same public key it is then possible to obtain information on the ϕ 's. Suppose she sends different encrypted messages e_i, then computing $(e_i - e_1)\,h \pmod{Q}$, one obtains exactly the value of $\phi_1 - \phi_i$, repeating this operation with the different e_i leads to sufficient information for some coordinates of ϕ_1 to allow a brute force attack on the remaining coordinates.

4. Popov Normal Form
Let F be a field and M an r by c matrix with entries in $F[T]$ where T is an undeterminate. We are interested in the $F[T]$-module L spanned by the rows of M. With every vector z of length c over $F[T]$ we attach its sup norm say $|z|$ defined as the largest degree in T of its entries. Formally,

$$|z| = \max\left\{\deg_T\left(z_i(T)\right) \mid i = 1, \cdots, c\right\}$$

There exists an effective algorithm of polynomial complexity to compute the minimum of the sup norm $|z|$ of $z \neq 0$ over $z \in L$. To describe this procedure we need the notion of (weak) Popov form for the matrix M. Define first the pivot index I_i attached to row i to be $=0$ if the row i of M is zero and as the rightmost column index j such that $m_{i,j}$ has the largest degree in T for $j \leq c$.

Next we say that M in weak Popov form if distinct rows are alloted distinct pivot indices. We can now quote [8, Lemma 8.1]

Lemma 1. *(Mulders & Storjohann) Let M be in weak Popov form and ρ the smallest sup norm of a row of M. All vectors in the $F[T]$-span of the rows of M have sup norm at least ρ*

The complexity of computing the weak Popov form is $O\left(rcRd^2\right)$ field operations, with R being the F-rank of M and d a best upper bound of the degrees of the entries of M.

1.2 Latest results

In 2005, Coglianese and Goi proposed MaTRU [9], which operates in the ring of k by k matrices $M_k(\mathbb{Z})[X]/\left(x^n - 1\right)$. Compared to NTRU, MaTRU further improves system operation efficiency. In 2011, Malekian et al. adopted the unique mathematical structure of quaternion algebra to design the QTRU cryptosystem [10], in which non-commutativeness plays a key role in the system, and which further enhances the security of QTRU. In 2015, Yasuda et al. proposed a general NTRU cryptosystem based on group ring, called GR-NTRU [11].

They investigated the security and performance of the cryptosystem under different instance group rings by combining group representation theory. In 2017, Thakur et al. designed NTRU over spit quaternion algebra; SQTRU can reduced the decryption failure due to a non-commutative algebraic structure. In 2018, Wang et al. presented a variant of NTRU with IND-CPA security named D-NTRU which has higher encryption and decryption efficiency than NTRU. In 2008, Karbasi et al. established PairTRU working in the $k \times k$ matrix ring with pairwise entries of k^2 distinct polynomials in $\mathbb{Z} \times \mathbb{Z}$. PairTRU is more secure than NTRU under lattice based attack. In 2020, Hajaje et al. proposed PMTRU by combining the advantages of NTRU with MATRU. PMTRU also improves the speed of encryption and decryption procedures.

DBTRU was proposed by Thang and Binh in 2015 [12]. The name DBTRU indicates the use of number theory and two binary truncated polynomial rings $GF(2)[x]/\left(x^n + 1\right)$, $(n \in \mathbb{Z}^+)$. Because both algorithms for encryption and decryption of DBTRU are only simple modular polynomial operations, DBTRU is as fast as NTRU. Although the message-expansion factor in DBTRU is higher than that in NTRU, the keys of DBTRU are smaller under approximately the same level of security.

In this paper, we further analyze the **security of DBTRU** and propose a linear algebra attack that can break it for all recommended parameter choices to compare the security levels in NTRU. More precisely, we first explore a hidden linear relationship between the public keys and the secret keys and find the parameter constraints for plaintext and secret key security while guaranteeing correct decryption.

2 The DBTRU Cryptosystem

In this section we will describe the theoretical part of the DBTRU cryptosystem, more precisely the notations, with the help of which we will state key generation, encryption, decryption, as presented in [13].

2.1 Notations

This cryptosystem is based on two integer parameters: s, l and four sets: $\mathcal{B}_f$, $\mathcal{B}_g$, $\mathcal{B}_\phi$, $\mathcal{B}_m$ of polynomials with binary coefficients. In general, s is smaller than l and gcd(s, l) = 1. Let $\mathbb{R} = \mathbb{Z}$ [x]/(x^n-1). The polynomial ring $GF(2)[x]/(x^n+1)$ is denoted by $\mathcal{R}_n$[x]. DBTRU is working in $\mathcal{R}_s$[x] and $\mathcal{R}_l$ [x]. We write $*$ for polynomial multiplication in $\mathcal{R}_n$[x] and let deg(f) denote the degree of $f \in \mathcal{R}_n$[x]. Let d_f, d_g, d_ϕ and d_m denote the maximum degree and Hamming weight of f, g, ϕ and m respectively. We set the modular polynomials as $S = x^s + 1$ and $L = x^l + 1$. The definition $\mathcal{L}(d_1, d_2)$ in NTRU will be replaced with:

$$\mathcal{B}(d) = \{b \in \mathcal{R}_l[x] \mid deg(b \leq d\}.$$

2.2 Key Generation

For the key generation process, Bob chooses two arbitrary positive integers s and l s.t. $s < l$ and sets $d_f = s - 1$. More than that, Bob chooses a small positive integer $\mathcal{N}_f$ and arbitrary $\mathcal{N}_f$ polynomials $f_i \in \mathcal{B}_f$ (i $\in$ [1, $\mathcal{N}_f$]), which are invertible in both $\mathcal{R}_s$[x] and $\mathcal{R}_l$[x].

For each f_i, Bob computes $F_{i,s} \in \mathcal{R}_s$[x] and $F_{i,l} \in \mathcal{R}_l$[x], where $F_{i,s} * f_i \equiv 1$ mod S and $F_{i,l} * f_i \equiv 1$ mod L. Then Bob computes f and its two inverses, F_s and F_l:

$$f = \prod_{i=1}^{N_f} f_i \quad F_s = \prod_{i=1}^{N_f} F_{i,s} \quad F_l = \prod_{i=1}^{N_f} F_{i,l}.$$

Bob chooses a non-zero polynomial $g \in \mathcal{B}_g$ and computes:

$$h = g * F_l * S \text{ mod L}.$$

Private key: f, f_i, F_s.
Public key: h.

2.3 Encryption and Decryption

If a second entity, let's say Alice, wnts to send a $s-$bit message m to Bob, firstly will randomly select a non-zero polynomial $\phi_0 \in \mathcal{B}_\phi$, a small positive integer N_ϕ and arbitrary N_ϕ polynomials $\phi_i \in \mathcal{B}_\phi \in [1, N_\phi]$. The ciphertext is given by the next formula:

$$e \equiv (\phi_0 * h + S * \sum_{i=1}^{N_\phi} \phi_i + m) \text{ mod } L. \tag{1}$$

Then, Alice sends the $l-$bit ciphertext e to Bob and after receiving e, Bob will compute:

$$a \equiv f * e \bmod L, \tag{2}$$

recovering the message m by computing:

$$m \equiv F_s * a \bmod S.$$

To ensure **successful** decryption, it is necessary that:

$$l > \max(\deg a) = N_f \cdot d_f + d_\phi + s.$$

3 Security Analysis

For the success of the attack, the authors discovered several vulnerabilities. First, they found that there is a hidden linear relationship between the public keys and the random non-zero polynomial in the encryption phase. Second, they constructed a linear system of equations with the unknown random non-zero polynomial and recovered the plaintext message after obtaining the random non-zero polynomial. In the last part of this section is presented the whole algorithm of the attack.

3.1 The Hidden Linear Relationship

Theorem 1. *As described in the DBTRU cryptosystem, let $S = x^s + 1$ and $L = x^l + 1$, where $s < l$. Let $\phi_i \in \mathcal{B}_\phi$ (i=0, 1, ..., N_ϕ) be some randomly chosen polynomials with $\phi_0 \neq 0$. For the ciphertext:*

$$e = (\phi_0 * h + S * \sum_{i=1}^{N_\phi} \phi_i + m) \bmod L, \tag{3}$$

if $l \geq s + 2d_\phi + 2$, then the part of the coefficients of e, namely, $e_{s+d_\phi+1}$, ..., e_{l-1} are equal to the coefficients of $\phi_0 * h$ mod L, with the same degree.

Proof. As calculated above (3), we can rewrite e as:

$$e = \sum_{i=0}^{l-1} e_i x^i,$$

where $e_i \in$ *GF(2) (i=0, 1, ..., l - 1).* We assume:

$$\phi_0 = \alpha_0 + \alpha_1 x + ... + \alpha_{d_\phi} x^{d_\phi},$$

where $\alpha_i \in$ *GF(2) (i=0, 1, ...,d_ϕ).* In addition,

$$h = h_0 + h_1 x + ... + h_{l-1} x^{l-1},$$

with $h_j \in GF(2)$ *(j=0, 1, ..., l-1)*. Next, we have:

$$deg(S * \sum_{i=1}^{N_\phi})\phi_i + m) \leq s + d_\phi.$$

Considering the maximum degree of components of $\phi_0 * h$, we have:

$$deg(\phi_0 * h) \leq d_\phi + d_h = d_\phi + l - 1.$$

From the precise analysis above, we have only part of the coefficients of e related to the $\phi_0 * h$, $S * \sum\limits_{i=1}^{N_\phi} \phi_i$ and m. More specifically, only the coefficients $e_0, e_1, ..., e_{d_\phi - 1}$ are affected by the modulo L, and $e_{s+d_\phi+1}, ..., e_{l-1}$ are just equal to the coefficients of $\phi_0 * h$ mod L with the same degree.

As a result of theorem 1, we can observe that the DBTRU break consists in the irrationality of the ciphertext structure. In each encryption process, we can build the following system of linear equations:

$$\begin{cases} h_{l-1}\alpha_0 + h_{l-2}\alpha_1 + ... + h_{l-d_\phi-1}\alpha_{d_\phi} = e_{l-1} \\ h_{l-2}\alpha_0 + h_{l-3}\alpha_1 + ... + h_{l-d_\phi-2}\alpha_{d_\phi} = e_{l-2} \\ \cdots \\ h_{s+d_\phi+1}\alpha_0 + h_{s+d_\phi}\alpha_1 + ... + h_{s+1}\alpha_{d_\phi} = e_{s+d_\phi+1} \end{cases} \tag{4}$$

with the partial coefficients $e_k = \sum_{i+j=k} \alpha_i \cdot h_j (s + d_\phi + 1 \leq k \leq l - 1)$ of the ciphertext e.

Next, we redefine equation (4) as:

$$A = \begin{bmatrix} h_{l-1} & h_{l-2} & \cdots & h_{l-d_\phi-1} \\ h_{l-2} & h_{l-3} & \cdots & h_{l-d_\phi-2} \\ \vdots & \vdots & \ddots & \vdots \\ h_{s+d_\phi+1} & h_{s+d_\phi} & \cdots & h_{s+1} \end{bmatrix},$$

where the elements of the matrix are tthe coefficients of the public key h.

The system of equations (4) has a unique solution, therefore the plaintext and secret polynomial ϕ_0 will be secure if $l < s + 2d_\phi + 2$.
In the next subsection will be presented how to recover the unique solution ϕ_0.

Remark 1. *Cao Minh Thang and Nguyen Binh [12] proposed an assessment of the algebraic attack on this scheme. Focusing on too many unknown polynomials was the real issue with their security analysis. With more attention, the researchers ([13]) managed to discover the hidden linear relationship between the public keys and the random non-zero polynomial.*

3.2 Recover the Non-Zero Polynomial ϕ_0

First, we have to analyze the solutions of Equation (4). If we know that the rank of matrix A defined above is equal to n, then Equation 4 should have only one solution, which is ϕ_0. This result (rank(A) is showed in [14], second Theorem:

Theorem 2. *Let N be a positive integer. Let $p_1, ..., p_l$ be the distinct prime factors of N. Consider the ring $x \times x$ matrices with entries in $\mathbb{Z}_{\mathbb{N}}$. Then, the proportion of invertible matrices (i.e., with determinant coprime to N) is equal to:*

$$\prod_{i=1}^{l} \prod_{k=1}^{n} (1 - p_i^{-k}).$$

Applying this theorem leads us to following Corollary:

Corollary 1. *Let p be a prime integer and $t \geq 0$ be an integer. Let $M_{(n+t)\times n}$ $(\mathbb{Z}_{\text{l}})$ denote the ring consisting of $(n + t) \times n$ matrices with entries in $\mathbb{Z}_{\text{l}}$. The probability of having at least one $n \times n$ invertible matrix in $M_{(n+t)\times n}$ $(\mathbb{Z}_{\text{l}}$ is:*

$$1 - \left(1 - \prod_{k=1}^{n} (1 - p^{-k})\right)^{\left(\begin{array}{c} n+t \\ n \end{array}\right)}$$

For more information, you can look at the proof of this corollary at [13].

Table 1. Probability of at least one $n \times n$ invertible matrix in $M_{(n+t)\times n}$ $(\mathbb{Z}_{\text{l}})$, with $p = 2$.

t n	t = 0	t = 1	t = 2	t = 3
n = 28	0.2879	0.99995	1.00000	1.00000
n = 45	0.28879	1.00000	1.00000	1.00000
n = 148	0.28879	1.00000	1.00000	1.00000

Remark 1. *From Table 1, we can see that even for $p = 2$, we only need to choose 3 times or more from $M_{(n+t)\times n}$ $(\mathbb{Z}_{\text{l}})$; then we can get a invertible $n \times n$ matrix with a probability close to 1.*

After obtaining ϕ_0, an attacker can recuperate the message m by computing:

$$m \equiv (e - \phi_0 * h) mod S.$$

Next, we display the proposed attack presented in [13]:

Algorithm 1 Algorithm 1: Main strategy of this attack

```
Input: $e_k = \sum_{i+j=k} \phi'_i \cdot h_j \, (s + d_\phi + 1 \leq k \leq l - 1)$.

1: Choose $d_\phi + 1$ equations from the input system of linear equations, and
   denote its coefficient matrix as $A$.
2: Determine whether $detA$ is equal to be zero.
3: If the $detA \neq 0$, apply Gaussian elimination to get the solution $a = (a_0, a_1, \cdots, a_{d_\phi})$
   of the selected systems of equations in Step 1.
4: Else, then reselect $d_\phi + 1$ equations, and go back to Step 2, until we find a
   system of equations for which its coefficient matrix is invertible.
5: For all equations entered, check if $a = (a_0, a_1, \cdots, a_{d_\phi})$ is a solution to each
   equation. If so, then we claim to have the target polynomial $\phi_0$.
6: Compute $(e - \phi_0 * h) \bmod S$.

Output: The $s - bit$ plaintext message $m$.
```

4 Experiments Results

In DBTRU, the authors concluded that as a variant of NTRU, DBTRU has advantages in both security and performance comparison with NTRU, as shown in Table 1 and Table 2.

Table 1: Comparison in moderate security mode of NTRU.

Moderate Security	NTRU	DBTRU
Basic parameters	$(N, p, q, d_f, d_g, d) = (107, 3, 64, 15, 12, 5)$	$(s, l, d_\phi, d_g, N_f, N_\phi) = (37, 197, 27, 105, 3, 4)$
S_m	$2^{26.5}$	$2^{51.21}$
S_k	2^{50}	$2^{51.71}$
Public key (bits)	642	197
Private key (bits)	340	222
Message-expansion	3.78	5.32

Table 2: Comparison in highest security mode of NTRU

Highest Security	NTRU	DBTRU
Basic parameters	$(N, p, q, d_f, d_g, d) = (503, 3, 256, 216, 72, 55)$	$(s, l, d_\phi, d_g, N_f, N_\phi) = (197, 1019, 147, 500, 3, 4)$
S_m	2^{170}	$2^{292.70}$
S_k	2^{285}	$2^{292.71}$
Public key (bits)	4024	1019
Private key (bits)	1595	1182
Message-expansion	5.05	5.17

Sage Math was used here to complete the experiments.

Table 3: The probability of having an invertible matrix

Parameters	Once	Twice	Three times
$(s, l, d_\phi, d_g, N_f, N_\phi) = (37, 197, 27, 105, 3, 4)$	0.2987	1	1
$(s, l, d_\phi, d_g, N_f, N_\phi) = (59, 293, 44, 120, 3, 4)$	0.2957	1	1
$(s, l, d_\phi, d_g, N_f, N_\phi) = (197, 1019, 147, 500, 3, 4)$	0.3033	1	1

From Table 3, the experiment data validate Remark 2.
Next, we give the total running time of breaking the DBTRU cryptosystem under 10,000 sets of data in Table 4: The running time for breaking DBTRU.

Table 4: The running time for breaking DBTRU

Parameters	The Number of Equations	The Number of Variables	Running Time(Sec)
$(s, l, d_\phi, d_g, N_f, N_\phi) = (37, 197, 27, 105, 3, 4)$	132	28	15.7352
$(s, l, d_\phi, d_g, N_f, N_\phi) = (59, 293, 44, 120, 3, 4)$	189	45	23.6364
$(s, l, d_\phi, d_g, N_f, N_\phi) = (197, 1019, 147, 500, 3, 4)$	674	148	128.0634

From Table 4 , the results show that for the three parameter choices recommended in the DBTRU cryptosystem, our proposed linear algebra attack can recover the plaintext within 1 s. □

5 Conclusions

In **CTRU**, the analogue of public key cryptosystem **NTRU**, the ring of integers is replaced by the ring of polynomials in one variable over a finite field. Attacks based on either the LLL algorithm or the Chinese Remainder Theorem was avoided. What we have to know is that an important tool of cryptanalys is the **Popov normal form** of matrices with polynomial entries. Also, the speed of encryption/decryption of **CTRU** is the same as **NTRU** for the same value of N.

Compared to NTRU, **DBTRU** has certain advantages regarding security and performance. At nearly the same level of security, **DBTRU** always has smaller keys. In this opaper, the focus was on breaking **DBTRU**, by a linear algebra attack, exploiting the secret linear relationship between public keys and secret keys. This attack is not only practical, but the plaintext can be recovered in less than 1s on a single PC.

6 Acknowledgements

First of all, we would like to express our deepest appreciation for the well-put together work of **Xiaofei Tong**, **Jingguo Bi**, **Yufei Duan**, **Lixiang Li** and **Licheng Wang**, from which we got a clear view regarding DBTRU cryptosystem and, in particular, what the security of this system is based on.

We would like to acknoweledge and give major thanks to our Master's Programme Coordinator, Professor Ferucio Laurenţiu Ţiplea, Ph.D, who made this work possible. His guidance and advice carried us through all the stages of writing this paper. We would also like to thank our Course Supervisor, Professor Emil Simion, Ph.D, for his full cooperation, impeccable comments and suggestions, which has made the research study achieve its smooth completion.

References

1. Hoffstein, J., Pipher, J. & Silverman, J. H. *NTRU: A Ring Based Public Key Cryptosystem in Algorithmic Number Theory (ANTS III) Lecture Notes in Computer Science 1423* 1998.
2. Lieman, D. *et al.* Standard specification for public-key cryptographic techniques based on hard problems over lattices. *IEEE P1363* **1,** D2 (2001).
3. Committee, A. S. *et al.* Lattice-Based Polynomial Public Key Establishment Algorithm for the Financial Services Industry. *ANSI X9,* 98–2010.
4. Bailey, D. V., Coffin, D., Elbirt, A., Silverman, J. H. & Woodbury, A. D. *NTRU in constrained devices* in *Cryptographic Hardware and Embedded Systems—CHES 2001: Third International Workshop Paris, France, May 14–16, 2001 Proceedings 3* (2001), 262–272.
5. Banks, W. D. & Shparlinski, I. E. *A variant of NTRU with non-invertible polynomials* in *Indocrypt* **2** (2002), 62–70.
6. Gaborit, P., Ohler, J. & Soli, P. CTRU, a polynomial analogue of NTRU, INRIA. *Rapport de recherche* (2002).
7. Vats, N. *Algebraic cryptanalysis of CTRU cryptosystem* in *Computing and Combinatorics: 14th Annual International Conference, COCOON 2008 Dalian, China, June 27-29, 2008 Proceedings 14* (2008), 235–244.
8. Mulders, T. & Storjohann, A. On lattice reduction for polynomial matrices. *Journal of symbolic computation* **35,** 377–401 (2003).
9. Coglianese, M. & Goi, B.-M. *MaTRU: A new NTRU-based cryptosystem* in *Progress in Cryptology-INDOCRYPT 2005: 6th International Conference on Cryptology in India, Bangalore, India, December 10-12, 2005. Proceedings 6* (2005), 232–243.
10. Malekian, E., Zakerolhosseini, A. & Mashatan, A. QTRU: Quaternionic Version of the NTRU Public-Key Cryptosystems. *ISeCure* **3** (2011).
11. Yasuda, T., Dahan, X. & Sakurai, K. Characterizing NTRU-variants using group ring and evaluating their lattice security. *Cryptology ePrint Archive* (2015).
12. Thang, C. M. & Binh, N. DBTRU, a new NTRU-like cryptosystem based on dual binary truncated polynomial rings. *2015 2nd National Foundation for Science and Technology Development Conference on Information and Computer Science (NICS),* 11–16 (2015).
13. Tong, X., Bi, J., Duan, Y., Li, L. & Wang, L. Security Analysis of DBTRU Cryptosystem. *Entropy* **vol. 24,** 1349 (Sept. 2022).
14. Nguyen, P. Q. *Cryptanalysis of the Goldreich-Goldwasser-Halevi Cryptosystem from Crypto '97* in *Proceedings of the 19th Annual International Cryptology Conference on Advances in Cryptology* (Springer-Verlag, Berlin, Heidelberg, 1999), 288–304. ISBN: 3540663479.