An improved NTRU Cryptosystem via Commutative Quaternions Algebra

Nadia Alsaidi[1], Mustafa Saed[2], Ahmad Sadiq[3], Ali A. Majeed[1]
[1]Department of Applied Sciences, University of Technology, Iraq
[2]Hyundai-Kia America Technical Center, USA
[3]Department of Computer Science, University of Technology, Iraq
nadiamg08@gmail.com, msaed@hatci.com, drahmad_tark@uotechnology.edu.iq, ali_alany_91@gmail.com

***Abstract*—NTRU is a public key cryptosystem operating on the ring Z[*X*]/(X^N− 1), which is known as the ring of convolution polynomials of rank *N*, where *N* is a prime. Reducing the decryption failure probability is a big challenge associated with such type of cryptosystem and is related to the ring that *NTRU* is based on. In this paper, a new multidimensional public key cryptosystem is proposed using commutative ring of quaternions that is not fully fit within Circular and Convolutional Modular Lattice. The decryption failure of this new algebraic structure is reduced. Furthermore, its complexity is four times the complexity of the classical *NTRU*. This results in high secured system resistance to some well-known attacks. Despite this advantage, the computational time analysis shows that the proposed system is slower than the original *NTRU*.**

Keywords*— public key cryptography; NTRU; lattice; quaternion algebra; non-associative cryptosystem.

I. Introduction

CRYPTOGRAPHY is the science of protecting the privacy of information during communication under hostile conditions. Modern telecommunication networks, especially, the Internet and mobile-phone networks have tremendously extended the limits and possibilities of communications and information transmissions. Associated with this rapid development, there is a growing demand for cryptographic techniques, which have spurred a great deal of intensive research activities in the study of cryptography

In mid-1990, a software company needed a cryptosystem that deals with a few bits processors and small numbers. Three mathematicians, Jeffry Hoffstein, Jill Pipher and Joseph Silverman [1] suggested a new cryptosystem, *NTRU* (Number Theory Research Unit). This system is a public–key cryptosystem. The computational and space complexity problems motivated them to propose this system that was fully presented in 1998. It is not based on integer factorization and discrete logarithm problem, but, it is based on a class of arithmetic operations that are efficiently performed with insignificant storage and time complexity [2]. This property made NTRU very suitable choice for a large number of applications, such as mobile phones, portable devices, low-cost smart cards, and RFID devices [3].

Since the introduction of NTRU cryptosystem, many researchers tried to improve its performance during the past fifteen years. This was done through the development of its algebraic structure to some Dedekind domain and Euclidean rings such as $Z[i]$, and GF(2^k)[x]. The first generalization of NTRU to Euclidean integer was proposed by Gaborit, et al. [4]. Through his initiative suggestion of replacing NTRU algebraic structure with other rings, he referred to it CTRU. In 2005, Coglianese et al. [5] improved the *NTRU* cryptosystem by replacing its original ring with a $k\times k$ matrices ring of polynomial with order *n*, known as *MaTRU*. It has improved speed by a factor of $O(k)$ over NTRU. In 2009, Malekian et al. [6] presented the *QTRU* cryptosystem. It was a multi–dimensional public key using quaternion algebra extended ring, which is broader than Dedekind domain and Euclidean algebra. Their underlying algebraic structure was non–commutative. This implied keeping the positive points of NTRU, and making it more resistant to some lattice-based attacks [7]. Another framework based on the Eisenstein integers Z [w], was presented by Jarvis [8] in 2011. This ring is defined as a cube root of unity and the coefficients are integers from Z. They called it *ETRU*, and showed that *ETRU* had improved the *NTRU* security [9].

In this paper a new *NTRU* cryptosystem is proposed using commutative ring of quaternions *CQ*. It has the same structure of *QTRU* but depends on the polynomial algebra with coefficients in *CQ*. It will be referred to as *CQTRU*. Some conditions on the parameter selection are placed to alllow the proposed system high chance for successful decryption.

The text of this paper is organized in the following way: a brief summarization of the *NTRU* cryptosystem is presented in Section 2. Some mathematical description of the alternative *CQ*, as a base ring for the proposed system, is discussed in Section 3. In Section 4, the proposed *CQTRU* is introduced, whereas the implementation of *CQTRU* with the improvement of the decryption failure probability is presented in Section 5. The performance analysis is discussed in Section 6, and the conclusions are presented in Section 7.

II. The NTRU Cryptosystem

A simple description of the NTRU cryptosystem is summarized in this section. For more details, the reader is referred to [1, 10-14]. The NTRU system is principally based on the ring of the convolution polynomials of degree *N-1* denoted by $R=Z[x]/(x^N-1)$. It depends on three integer parameters *N*, *p* and *q*, such that, $(p, q)=1$. Before going through *NTRU* phases, there are four sets used for choosing *NTRU* polynomials with small positive integers denoted by L_m, L_f, L_g and $L_r \subseteq R$. It is like any other public key cryptosystem constructed through three phases: key generation, encryption and decryption.

A. Key Generation phase

To generate the keys, two polynomials *f and g* are chosen randomly from L_f and L_g respectively. The function *f* must be invertible. The inverses are denoted by F_p , $F_q \in R$, such that:

$$F_p * f \equiv 1 \ (mod\ p) \quad \text{and } F_q * f \equiv 1 (mod\ q)$$

The above parameters are private. The public key *h* is calculated by,

$$h = p\ F_q * g \ mod\ q \qquad (1)$$

Therefore; the public key is $\{h, p, q\}$, and the private key is: $\{f , F_p\}$.

B. Encryption phase

The encryption is done by converting the input message to a polynomial $m \in L_m$ and the coefficient of *m* is reduced *modulo p.* A random polynomial *r* is initially selected by the system, and the cipher text is calculated as follows,

$$e = r * h + m \quad mod\ q. \qquad (2)$$

C. Decryption phase

The decryption phase is performed as follows: the private key, *f,* is multiplied by the cipher text *e* such that,

$$\begin{aligned} f * e \ \ mod\ q &= f*(p.h*r+m)\ m\ od\ q \\ &= p.f*h*r+f*m \ \ mod\ q \\ &= p.f*F_p^{-1}*g*r+f*m \ \ mod\ q \\ &= p.g*r+f*m \quad mod\ q \end{aligned}$$

The last polynomial has coefficients most probably within the interval $(-q/2,q/2]$, which eliminates the need for reduction *mod q.* This equation is reduced also by *mod p* to give a term *f*m mod p,* after diminishing of the first term *p.g*r.* Finally, the message *m* is extracted after multiplying by $Fp^{-}1$, as well as adjusting the resulting coefficients via the interval $[-p/2, p/2)$.

III. Algebraic structure of CQTRU

The suggestion of replacing the original ring of *NTRU* with other rings Gaborit et al. [4], and based on *NTRU* structure, a new scheme for *NTRU* cryptosystem that depends on polynomial algebra with coefficients in the commutative ring of quaternions *CQ* is proposed to introduce a new cryptosystem called *CQTRU.* Prior to establishing the validity of the proposed system, The *CQ* ring should be defined with its addition and multiplication operations, and the existence of the multiplicative inverses [15-19].

A. Commutative Quaternions (CQ)

In a four-dimension vector space, a commutative quaternions set is denoted by *CQ,* and defined as:

$$CQ = \{ a = t + xi + yj + zk : t,x,y,z \in R \text{ and } i,j,k \notin R\}.$$

Where; *i, j, k* satisfy the following multiplication rules: $i^2 = k^2 = -1, j^2 = 1$ and $ij = k$.

In this paper, *i, j* and *k* are defined as $i^2 = a, j^2 = b$, $k^2 = ab$ and $ij = k$. By this definition, a general commutative algebraic system is defined. Assuming *F* is an arbitrary field, the commutative quaternion algebra *A* can be defined over *F* as:

$$A = \{a + bi + cj + dk \mid a, b, c, d \in F, i^2 = a, j^2 = b, i\,j = k\}.$$

Clearly, if we assume that $a = -1$, $b = 1$ and *F* be the field of real numbers *R,* then, based on the choices of *a* and *b* and the nature of the field *F,* the original definition of commutative quaternion is obtained.

Let A_0 and A_1 be two commutative quaternion algebras such that:

$A_0 = \{f_0 + f_1.i + f_2\, j + f_3.k | f_0, f_1, f_2, f_3 \in R_p, i^2 = -1, j^2 = 1, ij = k\}$ and
$A_1 = \{g_0 + g_1.i + g_2.j + g_3.k | \ g_0, g_1, g_2, g_3 \in R_q , i^2 = -1, j^2 = 1, ij = k\}.$

Assume that $a_0, a_1 \in A_0$ (or A_1), such that, $a_0 = t_0 + x_0.i + y_0.j + z_0.k$ and $a_1 = t_1 + x_1.i + y_1.j + z_1 k$. Then, the operation on these two commutative quaternions; i.e. addition, multiplication and multiplicative inverse, will be given as:

$$a_0 + a_1 = (t_0 + t_1) + (x_0 + x_1)i + (y_0 + y_1)j + (z_0+z_1)k$$

$$a_0 \,.\, a_1 = (t_0\, t_1 - x_0\, x_1 + y_0\, y_1 - z_0\, z_1) + (x_0\, t_1 + t_0\, x_1 + z_0\, y_1 + y_0\, z_1)i + (t_0\, y_1 + y_0\, t_1 - x_0\, z_1 - z_0\, x_1)j + (z_0\, t_1 + t_0\, z_1 + x_0\, y_1 + y_0\, x_1)k.$$

B. Multiplicative invers in CQ Algebra

In *NTRU* public key cryptosystem scheme, the most important factor is the existence of the multiplicative inverse. For any element *a* in *CQ* to be used in *CQTRU*, the existence of its multiplicative inverse module *p* and *q* has to be checked.

For each $a \in CQ$, *a* can be represented by a 2×2 complex matrix, such that, if $a = a_0 + b_0 i + c_0 j + d_0 k \in CQ$, then *a* can be uniquely represented as $a = c_1 + jc_2$, where $c_1 = a_0 + b_0 i$, and $c_2 = c_0 + d_0 i$, $c_1, c_2 \in C$. Here C is the set of complex numbers [10].

Hence, for $a = c_1 + jc_2$, $\phi(a) = \begin{pmatrix} c_1 & c_2 \\ c_2 & c_1 \end{pmatrix}$, where ϕ is a bijective map.

Knowing $\phi(a)^{-1}$, the multiplicative inverse a^{-1} of $a \in CQ$ is calculated as follows:

If $(\alpha)^2 + (\beta)^2 \neq 0$, then $a^{-1} = \delta_0 + \delta_1 i + \delta_2 j + \delta_3 k$, where $\alpha = [a_0^2 + b_0^2 + c_0^2 + d_0^2]$, $\beta = [2a_0 * b_0 - 2c_0 * d_0]$.

Let $(\alpha^2+\beta^2)^{-1} = \partial$, then we have $[\delta_0 = \partial(\alpha * a_0 - \beta * b_0)$, $\delta_1 = \partial(\alpha * b_0 + \beta * a_0)$, $\delta_2 = \partial(\beta * d_0 - \alpha * c_0)$, and $\delta_3 = \partial(\alpha * d_0 + \beta * c_0)]$.

IV. THE PROPOSED CQTRU CRYPTOSYSTEM

In order to obtain a full understanding of how the *CQTRU* cryptosystem works, the algebraic structure for key generation, encryption and decryption, is designed as follows.

At the beginning, the parameters N, p, q have the property that N is an integer, p and q are relatively prime, and in all the algorithms, the parameter m represents either p or q depending upon which one is passed into the function.

A. Key Generation phase

To generate the public key, two small commutative quaternion F and G are randomly generated, such that

$$F = f_0 + f_1\, i + f_2\, j + f_3\, k \in L_f\,.$$
$$G = g_0 + g_1 i + g_2 j + g_3\, k \in L_g.$$

As it was mentioned above, F is invertible over A_0 and A_1 if $(\alpha^2+\beta^2)$ is invertible in Z_{cp} and Z_{cq}. Otherwise; a new commutative quaternion is generated. The inverses of F over Z_{cp} and Z_{cq} are denoted by F_p and F_q respectively.

Now, the public key is calculated as follows:

$$\begin{aligned} H &= F_q \,.\, G \mod q \\ &= (f_{q0} * g_0 - f_{q1} * g_1 - f_{q2} * g_2 - f_{q3} * g_3\,)+ \\ &\quad (\,f_{q1} * g_0 + f_{q0} * g_0 + f_{q2} * g_3 + f_{q3} * g_2\,)i+ \\ &\quad (\,f_{q0} * g_2 + f_{q2} * g_0 - f_{q1} * g_3 - f_{q3} * g_1\,)j+ \\ &\quad (\,f_{q0} * g_3 + f_{q3} * g_0 + f_{q2} * g_1 + f_{q1} * g_2\,)k\,. \end{aligned}$$

The commutative quaternions F, F_p and F_q will be kept secret in order to be used in the decryption phase. It is obvious that the estimated time to generate a key for the proposed scheme is 16 times slower than that of *NTRU*, when the same parameters (N, p *and* q) are selected for both cryptosystems. However, with a lower dimension N, we can achieve the original *NTRU* speed.

As mentioned previously, the new system is a 4-dimension space. Hence, if one chooses the coefficients of i, j and k to be zeros in the commutative quaternions F and G, then the system will be completely similar to *NTRU*. Moreover, this choice of zero coefficients for j and k will yield a cryptosystem based on complex numbers. Finally, if one of the coefficients of I, j or k is equal to zero, we obtain a tridimensional scheme.

B. Encryption phase

At the beginning of the encryption process, the cryptosystem must generate a random commutative quaternion called the blinding quaternion. The input message should be converted into a commutative quaternion. The cipher text will be computed and sent in the following way:

Let $M = m_0 + m_1 i + m_2\, j + m_3\, k$

where, $m_0, m_1, m_2, m_3 \in L_m$, generate a random quaternion $R = r_0 + r_1\, i + r_2\, j + r_3\, k$, and $r_0, r_1, r_2, r_3 \in L_r$.

Hence, the encryption function used is:

$$E = p.H\,.\,R + M \mod q \tag{3}$$

In this phase, a total of four data vectors are encrypted at the same time.

C. Decryption phase.

After receiving the cipher text E, the original message is constructed as follows.

The private key F is used to find B:

$$B = F\,.\,e \mod q \tag{4}$$

The coefficient of B should be reduced mod q into the interval $(-q/2, q/2]$.

The next step in the decryption process is to calculate the commutative quaternion D.

$$D = F_p\,.\,B \mod p. \tag{5}$$

The original message is obtained by reducing D in the interval $[-p/2, p/2]$.

D. How Decryption Works :

Since $B = F\,.\,E \mod q$

$$= (F\,.\,(\,p.H\,.R + M\,)) \mod q$$
$$= (F\,.\,p.H\,.R + F\,.\,M\,) \mod q,$$

the value of H is substituted to get,

$$B = (\,pF\,.\,F_q\,.\,G\,.\,R + F\,.\,M\,) \mod q$$
$$= (\,pG\,.\,R + F\,.\,M\,) \mod q\,.$$

Since $D = F_p\,.\,B \mod p$, then

$$D = F_p\,.(\,pG\,.\,R + F\,.\,M\,) \mod p$$
$$= (\,F_p\,.pG\,.\,R + F_p\,.F\,.\,M\,) \mod p$$

The term $(F_p\,.\,pG\,.\,R)$ will be disappear after reducing mod p, to obtain the term $(F_p\,.\,F\,.\,M\,)$.

Since $F_p\,.\,F = 1 \mod p$, normalizing the result into the interval $(-p/2, +p/2]$ yields the original message M. Therefore, the decryption speed is half the encryption speed because decryption includes 32 convolutions product. This is clearly analogous to the *NTRU* cryptosystem.

V. IMPLEMENTATION AND EXPREMENTS

Both CQRTU and NTRU are implemented in Matlab. The experiments were performed on a PC with 2.4 GHZ Intel Core 3, Quad processor and 4 MB Ram under windows 7, 32 bit operating system. For p=3, key generation, encryption and decryption speed with the probability of successful decryption are shown in Table 1. The probability of decryption failure depends on the choice of public parameters.

However, when N is fixed and the other parameters take larger values, the probability of decryption failure is decreased.

Table 1. Speed & probability of successful decryption, $p=3$

					Time in (ms)			
N	*q*	d_f	d_g	d_r	*Gen.*	*Encr.*	*Decr.*	*Pro(failure)*
73	128	10	8	5	65.3	10.9	18	0.000051782
73	128	12	10	6	67	12	18	0.0000003176
107	192	15	12	5	115	28	52	0.000288
107	192	20	12	10	116	27	50	0.0000028248
149	256	20	12	10	142	32	63	0.0000001192
149	256	35	25	20	145	33	60	0.0005515
167	256	40	20	18	186	36	68	0.00083229
167	256	50	21	19	186	39	70	0.00002532
211	256	40	20	18	275	53	93	0.000021775
211	256	30	24	22	278	53	94	0.000005822
257	256	40	20	18	350	71	126	0.0000004112
257	256	30	24	24	356	72	124	0.0000076072

A. *Decryption failure*

The probability of decryption failure is decreased if all commutative quaternion coefficients of $F \bullet E=(pG \bullet R+F \bullet M)$ lie in the interval $(\frac{-q}{2},\frac{q}{2}]$. For the *CQTRU*, this probability is computed as follows:

To calculate $var[a_{i,j}]$, it is sufficient to assume that $E[f_{i,k}]\approx 0$, $E[g_{i,k}]=E[r_{i,k}]=E[m_{i,k}]=0$, $E[a_{i,k}]=0$ where $i=0, 1, 2, 3$ and $k=0,\ldots,N-1$, and E is the mean function. Since each coefficient of quaternion element is a polynomial of degree N, then we have

$$Var\left[r_{i,k}, g_{j,i}\right]=\frac{4d_r \cdot d_g}{N^2}$$

$$Var\left[f_{i,k}, m_{j,i}\right]=\frac{d_f(p-1)(p+1)}{6N}$$

$$Var\left[a_{0,k}\right]=\frac{16p^2 d_r \cdot d_g}{N}+\frac{4d_f(p-1)(p+1)}{6}$$

$$\Pr(\left|a_{i,k}\right|<\frac{q}{2})=2\phi(\frac{q-1}{2\sigma})-1$$

Where; ϕ denotes the distribution of the standard normal variable, and

$$\sigma=\sqrt{\frac{16p^2 d_r d_g}{N}+\frac{4d_f(p-1)(p+1)}{6}}.$$

$a_{i,k}$'s are assumed to be independent random variables. The successful decryption probability in *CQRTU* can be calculated by the following two observations:

$$\left(2\phi\left(\frac{q-1}{2\sigma}\right)-1\right)^N,\ \left(2\phi\left(\frac{q-1}{2\sigma}\right)-1\right)^{4N} \qquad (6)$$

VI. PERFORMANCE ANALYSIS

After comparing *NTRU* to other cryptosystems, such as *RSA* and *ECC,* which are based on the number theoretic problem (e.g., factorization and discrete logarithm) [20], *NTRU* was found to have an advantage over them due to its fast and low space storage arithmetic operations. This turned *NTRU* into a very suitable choice for a large number of applications.

A. *Computational complexity*

For encryption, one commutative quaternion multiplication is needed in addition to 16 convolution multiplication and 4 polynomial addition; both with $O(N)$ complexity. In the encryption phase, any incoming data is converted into polynomial with coefficients between $-p/2$ and $p/2$. In other words, m_0, m_1, m_2 and m_3 are small polynomials modulo q.

B. *Security Attacks*

1- Alternate keys analysis in *CQTRU*

Compared to *NTRU*, any alternate of the private key f can be used to encrypt and decrypt the same messages as f. The attacker needs only to find one polynomial having the same properties of f. In *CQTRU,* to find the alternate private key F, the attacker needs to find four polynomials of the same properties of the privet key F. Hence, *CQTRU* is more robust to this attack than *NTRU*. Accordingly, it is considered to be more secure than *NTRU.*

2- Brute Force Attacks

Compared to *NTRU*, to recover the privet key f; an attacker has to try using all possible $f' \in L_f$ in an attempt to check if $f * h \bmod q$ has small polynomial coefficients or not. Another way is to try all possible $g' \in L_g$ and check if $g * h^{-1} \bmod q$ has small coefficients. In *CQTRU,* the attacker uses the same procedure, where he/she knows all the public parameters and constant d_r, d_g, d_f, q, p, and N. The attacker needs to look in the space of large order to be able to look in the spaces L_f and L_g, as follows:

$$\left|L_f\right|=\binom{N}{d_f}^N\binom{N-d_f+1}{d_f}=\frac{(N!)^4}{(d_f!)^8(N-2d_f)!^4}$$

$$\left|L_g\right|=\binom{N}{d_g}^N\binom{N-d_g+1}{d_g}=\frac{(N!)^4}{(d_g!)^8(N-2d_g)!^4}$$

The space of L_f is a bigger than the space of L_g. For this reason, it is easier for the attacker to search in L_g. By using the brute force attack, an attacker can break a message encrypted by *CQTRU.* This can be done by searching in the space L_r because $E=H \bullet R+M$ ($mod\ q$) is known. If the attacker has an ability to find the random commutative quaternion R then he/she will be able to find the original message by calculating

$M=E\text{-}H\bullet R$ (*mod q*). It is obvious that in a brute force attack, the security of any message depends on how hard it is to find R. The order of the space L_r is calculated using the same approach of calculating the order of L_f and L_g,

$$|L_r| = \binom{N}{d_r}^N \binom{N-d_r+1}{d_r} = \frac{(N!)^4}{(d_r!)^8 (N-2d_r)!^4}$$

This comparison shows that *CQTRU* is more robust to this attack than *NTRU*.

3- Lattice based attacks

It is known that every commutative quaternion is isomorphism to a matrix called the fundamental matrix given in (7):

$$q = q_0 + q_1 i + q_2 j + q_3 k \cong \begin{pmatrix} q_0 & -q_1 & q_2 & -q_3 \\ q_1 & q_0 & q_3 & q_2 \\ q_2 & -q_3 & q_0 & -q_1 \\ q_3 & q_2 & q_1 & q_0 \end{pmatrix} \quad (7)$$

The system parameters (d_f, d_g, d_r, p, q, N) are known to the attacker as well as the public key $H=F_q\bullet G=h_0+h_1i+h_2j+h_3k$. When the attacker manages to find one of the commutative quaternions F or G, the *CQTRU* cryptosystem is broken. Note that, h_0, h_1, h_2 and h_3 are polynomials of order N over Z. These polynomials can be represented as vectors over Z^N as follows:

$H= h_0+h_1i+h_2j+h_3k\cong[h_0\ h_1\ h_2\ h_3]$, where

$h_0= h_{0,0} +h_{0,1}\, x+\dots\dots+h_{0,N-1}\, x^{N-1}$

$\cong[h_{0,0}\ \ h_{0,1} \dots\dots\dots h_{0,N-1}]$,

$h_1= h_{1,0} +h_{1,1}\, x+\dots\dots+h_{1,N-1}\, x^{N-1}$

$\cong[h_{1,0}\ \ h_{1,1} \dots\dots\dots h_{1,N-1}]$,

$h_2= h_{2,0} +h_{2,1}\, x+\dots\dots+h_{2,N-1}\, x^{N-1}$

$\cong[h_{2,0}\ \ h_{2,1} \dots\dots\dots h_{2,N-1}]$,

$h_3= h_{3,0} +h_{3,1}\, x+\dots\dots+h_{3,N-1}\, x^{N-1}$

$\cong[h_{3,0}\ \ h_{3,1} \dots\dots\dots h_{3,N-1}]$

Since the polynomial ring Z is isomorphic to the circulant matrix ring of order N over Z, the polynomials h_0, h_1, h_2 and h_3 can be represented in their isomorphic representation for lattice analysis as:

$$h(i)_{N\times N} = \begin{pmatrix} h_{i,0} & \cdots & h_{i,N-1} \\ h_{i,N-1} & \cdots & h_{i,N-2} \\ \vdots & \ddots & \vdots \\ h_{i,2} & \cdots & h_{i,1} \\ h_{i,1} & \cdots & h_{i,0} \end{pmatrix} \quad (8)$$

where $i=0, 1, 2, 3$.

With respect to the above assumptions, to describe the partial lattice attack first, let the commutative quaternions F and G be represented by $F=[f_0\ \ f_1\ \ f_2\ \ f_3]$, and $G=[g_0\ \ g_1\ \ g_2\ \ g_3]$ where $f_0, f_1, f_2, f_3, g_0, g_1, g_2, g_3 \in Z[x]/(x^N\text{-}1)$. In order to form the lattice, the vectors $[u_0\ u_1\ u_2\ u_3\ v_0\ v_1\ v_2\ v_3]$ must belong to Z^{8N}. This lattice is denoted by $L_{partial}$ and defined by:

$$L_{partial} = \begin{pmatrix} I_{4N\times 4N} & 0_{4N\times 4N} \\ H_{4N\times 4N} & q_{4N\times 4N} \end{pmatrix} \in Z^{8N} \quad (9)$$

where, I refers to the identity matrix, 0 is the zero matrix, and H is the fundamental matrix of h_i's. $L_{partial}$ contains a vector in the form $[u_0\ u1\ u2\ u_3\ v_0\ v_1\ v_2\ v_3]\in Z^{8N}$, that satisfies $F\bullet H=G$. However, there is a major difference between *NTRU* and *CQTRU* lattices, such that all points spanned by the *CQTRU* lattice merely includes a partial subset of the total set of vectors satisfying $F \bullet H=G$. To see this, let $[u_0\ \ u_1\ \ u_2\ \ u_3\ \ v_0\ \ v_1\ \ v_2\ \ v_3]$ denote the vector satisfying $F\bullet H=G$, then $[-u_1\ u_0\ -u_3\ u_2\ -v_1\ v_0\ -v_3\ v_2]$ is the answer. Also, since $iF \bullet H=iG$, therefore, $L_{partial}$ will not necessarily contain such vector. The attacker may manage to use the lattice reduction algorithm [21-22] to find a short vector satisfying $F\bullet H=G$. However, even with such promising assumption, $L_{partial}$ has a dimension that is four times larger than the lattice dimension of *NTRU* with the same order N. Hence, the *CQTRU* with the parameters ($N=107$, p, q) offers the same level of security as *NTRU* with the parameters ($N=428$, p, q). Therefore, for any chosen parameters (N, p, q) to be used in *CQTRU*, the system will be four times slower than *NTRU* with the same parameters as it is shown by Tables (2) - (4), which demonstrate that for the three phases; key generating, encryption and decryption, *CQTRU* is also slower than *NTRU* under the same environments. However, the *CQTRU* security is four times as that offered by *NTRU* with the same parameters. On the other hand, *NTRU* with *4N* dimensions is sixteen times slower with respect to computational time than *NTRU* with N dimensions. Therefore, *CQTRU* has a security advantage over *NTRU*.

Table 2. Key generating time in ms for *NTRU* and *CQTRU*

N	q	d_f	d_g	d_r	*NTRU*	*CQTRU*
73	128	10	8	5	20	67
107	128	15	12	5	40	116
149	192	20	15	10	52	142
167	192	25	22	18	56	186
211	256	28	25	22	76	278
257	256	33	30	28	98	356

Table 3. Encryption time in ms for NTRU and CQTRU

N	q	d_f	d_g	d_r	*NTRU*	*CQTRU*
73	128	10	8	5	3.1	10.9
107	128	15	12	5	8	28
149	192	20	15	10	9.5	32
167	192	25	22	18	11	39
211	256	28	25	22	14	53
257	256	33	30	28	19	71

Table 4. Decryption time in ms for NTRU and CQTRU

N	q	d_f	d_g	d_r	*NTRU*	*CQTRU*
73	128	10	8	5	5.2	18
107	128	15	12	5	14	52
149	192	20	15	10	17	63
167	192	25	22	18	19	70
211	256	28	25	22	24	93
257	256	33	30	28	33	126

The partial lattice attacks do not always give successful results because $L_{partial}$ does not necessarily contain all solutions of $F{\bullet}H=G$ in such a way that f_0, f_1, f_2, f_3, g_0, g_1, g_2, or g_3 would be short vectors. Therefore, the attacker must find a lattice that contains all vectors which satisfy the congruence $F \bullet H=G$.

VII. CONCLUSIONS

By changing the underlying ring of *NTRU*, the *NTRU* cryptosystem has been improved through the introduction of a new *NTRU* like public key cryptosystem. This is constructed by replacing the base ring of *NTRU* with a commutative quaternions ring that resulted in the emergence of *CQTRU* cryptosystem. Despite the apparent increase in computational time, it is considered to be reasonable with consideration to its higher complexity. This generalization of the algebraic structure of the *NTRU* resulted also in an improved security level over *NTRU*, and a significant improvement in the reduction of the decryption failure probability.

REFERENCES

[1] J. Hoffstein, J. Pipher, and J. H. Silverman, "NTRU: A ring-based public key cryptosystem," in *Lecture Notes in Computer Science*. Springer-Verlag, vol. 1423, 1998, pp. 267–288.

[2] N. Smart, F. Vercauteren, J. H. Silverman, "An algebraic approach to NTRU (q = 2^n) via Witt vectors and overdetermined systems of nonlinear equations", SCN 2004, Amalfi, Italy, LNCS vol. 3352, Springer, 2004.

[3] J. Hoffstein and J. Silverman, "Optimizations for ntru," in *In Public Key Cryptography and Computational Number Theory*, 2000, pp. 11–15.

[4] P. Gaborit, J. Ohler, P. Sole, "CTRU, a polynomial analogue of NTRU", INRIA, Rapport de recherche 4621, INRIA 2002, ftp://ftp.inria.fr/INRIA/publication/publi-pdf/RR/RR-4621.pdf

[5] M. Coglianese and B.-M. Goi, "MaTRU: A new NTRU-based cryptosystem," in *INDOCRYPT*, 2005, pp. 232–243.

[6] E. Malekian, A. Zakerolhosseini, "NTRU-Like Public Key Cryptosystems beyond Dedekind Domain up to Alternative Algebra", Transactions on Computational Science X Lecture Notes in Computer Science Volume 6340, 2010, pp 25-41

[7] Malekian E., Zakerolhosseini A., Mashatan A.: QTRU: a lattice attack resistant version of NTRU PKCS based on quaternion algebra (preprint). Available from the Cryptology ePrint Archive: http://eprint.iacr. org/2009/386.pdf. Accessed Sep. 2012.

[8] K. Jarvis, "NTRU over the Eisenstein integers", Masters Thesis, University of Ottawa, 2011.

[9] K. Jarvis, M. Nevins, "ETRU: NTRU over the Eisenstein integers", Designs, Codes and Cryptography, vol.74, No.1, 2015, pp 219-242.

[10] D. Coppersmith and A. Shamir, "Lattice attacks on NTRU", in *EUROCRYPT*, 1997, pp. 52–61.

[11] M. Nevins, C. Karimianpour, and A. Miri, "Ntru over rings beyond z," Designs, Codes and Cryptography, vol. 56, No1, 2010, pp.65–78.

[12] R. Kouzmenko, "Generalizations of the NTRU cryptosystem," Master's thesis, Polytechnique, Montreal, Canada, 2006.

[13] J. Hoffstein, J. Pipher, and J. H. Silverman, *An Introduction to Mathematical Cryptography*, ser. Science+Business Media, LLC.2ed edition, Springer, 2014.

[14] J. Pipher, "Lectures on the NTRU encryption algorithm and digital signature scheme", Brown University, Providence RI 02912Grenoble, June 2002.

[15] F. Catoni, R. Cannata and P. Zampetti, "An Introduction to Commutative Quaternions", Adv. appl. Clifford alg. vol.16, No.1, 2006, pp.1–28.

[16] R. D. Schafer, *An introduction to non-associative algebras*. New York: Dover Publications Inc., 1996, corrected reprint of the 1966 original.

[17] W.D. Banks, I.E. Shparlinski, "A Variant of NTRU with Non-Invertible Polynomials", In: Menezes, A., Sarkar, P. (eds.) INDOCRYPT 2002, LNCS, Springer, Heidelberg vol. 2551, 2002, pp. 62–70.

[18] J. C. Baez, "The octonions," Bulletin of the American Mathematical Society, vol. 39, No. 2, 2002, pp. 145–205.

[19] J. H. Conway and D. A. Smith, *On Quaternions and Octonions: Their Geometry, Arithmetic, and Symmetry*. A. K. Peters, Ltd., 2003.

[20] A. J. Menezes, P. C. van Oorschot, and S. A. Vanstone, *Handbook of applied cryptography*. Boca Raton, Florida: CRC Press, 1996.

[21] J. Hoffstein, J. H. Silverman, and W. Whyte, "On estimating the lattice security of NTRU," Technical Report 104, Cryptology ePrint Archive (2005), http://eprint.iacr.org/2005/104/ 20, 2005.

[22] J. H. Silverman, "Dimension-reduced lattices, zero-forced lattices, and the NTRU public key cryptosystem," Technical Report No. 13 (1999). Available at (http://www.ntru.com), 1999.