The ISC Int'l Journal of Information Security

January 2011, Volume 3, Number 1 (pp. 29–42)
http://www.isecure-journal.org

QTRU: Quaternionic Version of the NTRU Public-Key Cryptosystems

Ehsan Malekian [a], Ali Zakerolhosseini [a,*], and Atefeh Mashatan [b]

[a] *Faculty of Electrical and Computer Engineering, Shahid Beheshti University, Tehran, Iran.*
[b] *Security and Cryptography Laboratory, EPFL CH-1015, Lausanne, Switzerland.*

ARTICLE INFO.

Article history:
Received: 6 August 2010
Revised: 17 January 2011
Accepted: 20 January 2011
Published Online: 26 January 2011

Keywords:
QTRU, NTRU, Quaternion Algebra, Public-Key Cryptography, Encryption

ABSTRACT

In this paper we will construct a lattice-based public-key cryptosystem using non-commutative quaternion algebra, and since its lattice does not fully fit within Circular and Convolutional Modular Lattice (*CCML*), we prove it is arguably more secure than the existing lattice-based cryptosystems such as NTRU. As in NTRU, the proposed public-key cryptosystem relies for its inherent security on the intractability of finding the shortest vector in a certain non-convolutional modular lattice, yet it is efficient and cost effective, contrary to cryptosystems such as RSA or ECC. The detailed specification of the proposed cryptosystem, including the underlying algebraic structure, key generation, encryption and decryption process and also the issues regarding key security, message security, and probability of successful decryption are explained. We will further show, based on the existing results for lattice-reduction algorithms, that the proposed cryptosystem with a dimension of 41 will have a security equal to NTRU-167.

1 Introduction

Until 1996, most important public-key cryptosystems predominantly relied their security on the presumed difficulty of solving some number-theoretic problems such as the integer-factoring problem (IFP) or the discrete-logarithm problem (DLP) in a certain finite group [1]. Number-theoretic public-key cryptosystems suffer from the problems of slow and resource-demanding hardware or software implementation, as well as high-power consumption and more vulnerability to side-channel attacks. In addition to these drawbacks, it is not known whether IFP or DLP are solvable in polynomial time (on a conventional computer) or not [2].

During the past decade, it has been shown that the Closest-Vector Problem (CVP) is $\mathcal{NP}$-hard and the Shortest-Vector Problem (SVP) is also $\mathcal{NP}$-hard under randomized reduction [3–5]. These results led to new hopes for designing public-key cryptosystems based on worst-case hardness of the $\mathcal{NP}$-hard problems in a certain lattice [6]. Although most of lattice-based cryptosystems (for example Goldreich, Goldwasser, Halevi and Atjai-Dwork cryptosystems) were soon broken [6], the NTRU public-key cryptosystem, officially introduced in 1998 [7], managed to win public trust after numerous modifications and optimizations [8, 9] and eliminating some minor flaws [10]. It has now been fully standardized within IEEE P1363.1 [11].

Compared to more well-known cryptosystems such as RSA or ECC, the greatest advantage of NTRU

* Corresponding author.
Email addresses: e_malekian@sbu.ac.ir (E. Malekian), a-zaker@sbu.ac.ir (A. Zakerolhosseini), atefeh.mashatan@epfl.ch (A. Mashatan).
ISSN: 2008-2045

is that it works based on a class of basic arithmetic operations whose inherent complexity is rather low, amounting to $\mathcal{O}(N^2)$ in worst-case, where N is at most a 9-bit integer. Computational efficiency along with low cost of implementation have turned NTRU into a suitable choice for a large number of applications such as embedded systems, mobile phones, portable devices, and resource constrained devices [12, 13].

After the adoption of NTRU as a secure and safe scheme, several researches were conducted on generalization of the NTRU algebraic structure to different Euclidean rings, including $GF(2^k)[x]$ [14], the non-commutative ring of $k \times k$ matrices of polynomials in $\mathbb{Z}[x]/(x^N - 1)$ [15], the non-commutative ring $\mathrm{M} = M_k(\mathbb{Z})[X]/(X^n - I_{k\times k})$, where M is a matrix ring of $k \times k$ matrices of polynomials in $R = \mathbb{Z}[X]/(X^n - 1)$ [16], and Dedekind domains such as $\mathbb{Z}[i], \mathbb{Z}[\sqrt{-2}], \mathbb{Z}[\zeta_3]$ and $\mathbb{Z}[\zeta_5]$ [17–19]. Although generalization of NTRU to $GF(2^k)[x]$ in [14] never had a desirable result and was broken soon after [17], it resulted in a better understanding of the NTRU cryptosystem and suggested the idea of replacing NTRU algebraic structure with other rings and algebras.

In this paper we will extend the NTRU concept to non-commutative quaternion algebra, and will prove that the public-key cryptosystem based on this algebra is actually applicable and reasonable, and such a cryptosystem obtains its security from the hardness of the shortest-vector problem in a certain non-circulant lattice.

We considered the proposed scheme (hereafter called QTRU) exactly identical to NTRU in order to be able to cite the extensive and elaborate researches carried out regarding the security aspects of the cryptosystems in the past ten years. Hence, besides changing the underlying algebra, no other changes have been made. In particular, QTRU keeps the probabilistic properties of NTRU intact. Thus, QTRU inherits the main advantages of NTRU. Yet we argue that the proposed cryptosystem is not the end but the beginning of useful and constructive challenges for innovation of schemes different from NTRU, in a way that the best is made of the non-commutativity of this type of algebra.

The least advantage of QTRU is that its lattice is not fully classified under Convolutional Modular Lattice (*CML*), and hence the existing open problems with regard to the circular structure of this type of lattice and the probability of cryptosystem failure (with probable improvement in the Hermite factor of the future Lattice Reduction Algorithms) will fade. As a result of non-commutativity in the underlying algebraic structure, and bi-linearity of multiplication, many lattice-reduction algorithms work much slower. Consequently, we can reduce the dimension of the *module* considerably and, yet, obtain the same level of security.

In completely even circumstances, i.e. choosing the same parameters for both NTRU and QTRU, QTRU works four times slower than NTRU and the data are encrypted simultaneously as four vectors, but we claim that the dimension of QTRU can be reduced and the imposed speed reduction caused by quaternionic processing, can be compensated.

This paper is organized as follows: Section 2 provides an overview of the NTRU public-key cryptosystem. In Section 3 we briefly review some necessary background in quaternion algebras. We dedicate Section 4 to introducing the algebraic structure used in QTRU. Sections 5 and 6 are devoted to the description of QTRU and general analysis of the scheme. Last but not least, Section 7 discusses the security of QTRU against lattice-based attacks.

2 The NTRU Cryptosystem

The basic operations in NTRU take place in the ring $\mathbb{Z}[x]/(x^N - 1)$, which is known as the ring of convolution polynomials of rank N, where N is a prime [20, p. 392]. In the ring of convolution polynomials, addition and multiplication require only $\mathcal{O}(N)$ and $\mathcal{O}(N^2)$ operations, respectively. Let define the following three rings: $\mathcal{R} = \mathbb{Z}[x]/(x^N - 1)$, $\mathcal{R}_p = (\mathbb{Z}/p\mathbb{Z})[x]/(x^N - 1)$, and $\mathcal{R}_q = (\mathbb{Z}/q\mathbb{Z})[x]/(x^N - 1)$. An element f of the rings $\mathcal{R}$, $\mathcal{R}_p$, and $\mathcal{R}_q$ can be denoted interchangeably by a polynomial or its vector of coefficients: $f = \sum_{i=0}^{N-1} f_i.x^i \triangleq [f_0, f_1, ..., f_{N-1}]$. In the convolution rings, addition corresponds to the ordinary polynomial addition, i.e., component-wise, but multiplication, denoted by the symbol $\star$, is explicitly defined as follows

$$
\begin{aligned}
f(x) &:= \sum_{i=0}^{N-1} f_i x^i = [f_0, f_1, \ldots, f_{N-1}]_{1\times N},\ f_i \in \mathbb{Z} \\
g(x) &:= \sum_{i=0}^{N-1} g_i x^i = [g_0, g_1, \ldots, g_{N-1}]_{1\times N},\ g_i \in \mathbb{Z} \\
h(x) &:= \sum_{i=0}^{N-1} h_i x^i = [h_0, h_1, \ldots, h_{N-1}]_{1\times N},\ h_i \in \mathbb{Z} \\
h_k &:= \sum_{i=0}^{k} f_i.g_{k-i} + \sum_{i=k+1}^{N-1} f_i.g_{N+k-i} = \sum_{i+j \overset{N}{\equiv} k} f_i.g_j.
\end{aligned}
\tag{1}
$$

(Clearly, addition and multiplication in $\mathcal{R}_p$ or $\mathcal{R}_q$ are equivalent to performing the same operations in $\mathcal{R}$ and ultimately reducing the resulting coefficients mod p or mod q, respectively.)

Table 1. Definition of public parameters of NTRU.

Notation	Definition	Typical Value for $N = 167,\ p = 3,\ q = 128$
$\mathcal{L}_f$	$\{f \in \mathcal{R} \mid f$ has d_f coefficients equal to +1, $(d_f - 1)$ equal to -1, the rest 0$\}$	$d_f = 61$
$\mathcal{L}_g$	$\{g \in \mathcal{R} \mid g$ has d_g coefficients equal to +1, d_g equal to -1, the rest 0$\}$	$d_g = 20$
$\mathcal{L}_\phi$	$\{\phi \in \mathcal{R} \mid \phi$ has d_ϕ coefficients equal to +1, d_ϕ equal to -1, the rest 0$\}$	$d_\phi = 18$
$\mathcal{L}_m$	$\{m \in \mathcal{R} \mid$ coefficients of m are chosen modulo p, between $-p/2$ and $p/2\}$	-

Let d_f, d_g, d_ϕ, and d_m be constant positive integers less than N (typically $d_f = d_g = d_\phi = d_m = \mathbf{d} \approx N/3$) and let $\mathcal{L}_f$, $\mathcal{L}_m$, $\mathcal{L}_\phi$, $\mathcal{L}_g \subset \mathcal{R}$ be the subsets of *small polynomials* as defined in Table 1.

With the above notations and definitions, the NTRU public-key cryptosystem (known as NTRUEncrypt) can now be described as follows.

Public Parameters. The public parameters (N, p, q, d) in NTRU are assumed to be fixed and must be agreed upon by both the sender and the receiver. N and p are prime numbers and $\gcd(p, q) = \gcd(N, q) = 1$ and $q \gg p$, i.e., q is much bigger than p and N determines the structure of the ring $\mathbb{Z}[x]/(x^N - 1)$. (Typical values include $N = 167$ for moderate security, $N = 251$ for high security, and $N = 503$ for very high security along with $p = 3$, $q = 256$ and $d \approx N/3$.)

Key Generation. To create a pair of public and private keys, first two small polynomials $g \in \mathcal{L}_g$ and $f \in \mathcal{L}_f$ are randomly generated. The polynomial f must be invertible in both $\mathcal{R}_p$ and $\mathcal{R}_q$. Let f_p^{-1} and f_q^{-1} denote the inverses in $\mathcal{R}_p$ and $\mathcal{R}_q$, respectively. Hence, by counting the number of irreducible polynomials in $\mathcal{R}_q$, the probability that a randomly generated polynomial is invertible in $\mathcal{R}_q$ will be greater than $(1 - p^{-n})^{(N-1)}/n$, where n is the smallest integer which satisfies $p^n \equiv 1 \bmod N$ [21]. However, in a rare event that f is not invertible, a new polynomial f can be easily generated.

While f, g, f_p^{-1}, and f_q^{-1} are kept secret, the public key h is computed and published as follows

$$h = f_q^{-1} \star g \,(\text{mod } q).$$

Encryption. The cryptosystem initially selects a random polynomial $\phi \in \mathcal{L}_\phi$, called the ephemeral key, and encodes the input message into a polynomial $m \in \mathcal{L}_m$. The ciphertext is computed as follows:

$$e = p.h \star \phi + m \ (mod\ q).$$

Neglecting the time required for ephemeral key generation and conversion time of the incoming message into the polynomial $m \in \mathcal{L}_m$, and by pre-computing and storing $p.h$, NTRU encryption takes $\mathcal{O}(N^2)$ steps.

Decryption. The first step of the decryption process starts by multiplying (convolving) the received polynomial e by the private key f

$$\begin{aligned} a :&= f \star e \,(\text{mod } q) = f \star (p.h \star \phi + m) \ (\text{mod } q) \\ &= p.f \star h \star \phi + f \star m \ \ (\text{mod } q) \\ &= p.f \star f_q^{-1} \star g \star \phi + f \star m \ \ (\text{mod } q) \\ &= p.g \star \phi + f \star m \ \ (\text{mod } q). \end{aligned} \tag{2}$$

In the second step, the coefficients of $a \in \mathcal{R}_q$ are identified with the equivalent representatives in $\mathcal{S} := \{-q/2 + 1, \ldots, +q/2\}$. Assuming that the public parameters have been chosen properly, the resulting polynomial is exactly equal to $p.g \star \phi + f \star m$ in $\mathcal{R}$. With this assumption, when we reduce the coefficients of a modulo p, the term $p.g \star \phi$ vanishes and $f \star m \ (\text{mod } \ p)$ remains. In order to extract the message m, it is enough to multiply $f \star m \ (\text{mod } \ p)$ by f_p^{-1}.

Successful Decryption. If the public parameters (N, p, q, d) are chosen to satisfy $q > (6d + 1).p$ (where $d := d_f = d_g = d_\phi$, as defined earlier), the decryption process will never fail. However, to have a better performance and also to reduce the size of the public key, the public parameter q can be chosen in such a way that the probability of decryption failure is very small (e.g., 2^{-80}) [20, p. 395]. Successful decryption depends on whether $|p.g \star \phi + f \star m|_\infty < q$ or not. With a few simple assumptions and probabilistic calculations [17, pp. 16], the upper-bound for the probability of successful decryption can be approximated as follows

$$\Pr(\text{successful decryption}) = \left(2\Phi(\frac{q-1}{2\sigma}) - 1\right)^N, \tag{3}$$

where $\Phi(\cdot)$ denotes the standard normal distribution function and $\sigma \approx \sqrt{\frac{36d_f.d_g}{N} + \frac{8d_f}{6}}$.

The NTRU Lattice. The underlying hard problem of NTRU is to find short vectors in Convolution Modular Lattices (CML), in which the lattice structure is cyclic and the entire lattice basis is specified by a single vector [22]. The idea of applying lattice-reduction techniques against NTRU was first introduced by Coppersmith and Shamir in [23]. Many articles are pub-

lished on the subject of lattice attacks against NTRU [24–27]. The main idea of the attacks is as follows.

Consider the public-key h as a vector $h = [h_0, h_1, ..., h_{N-1}]$. Then, the NTRU lattice $\mathcal{L}_{NTRU}$ is the lattice spanned by the rows of the following matrix of dimension $2.N$

$$\mathcal{L}_{NTRU} = \left[\begin{array}{c|c} \lambda.I_{N\times N} & \mathcal{H} \\ \hline \underset{N\times N}{0} & q.I_{N\times N} \end{array}\right] = \left[\begin{array}{c|c} & \begin{matrix} h_0 & h_1 & h_2 & \cdots & h_{N-1} \\ h_{N-1} & h_0 & h_1 & & h_{N-2} \\ h_{N-2} & h_{N-1} & \ddots & & h_{N-3} \\ \vdots & \vdots & & \ddots & \vdots \\ h_1 & h_2 & h_3 & \cdots & h_0 \end{matrix} \\ \lambda.I_{N\times N} & \\ & \\ \hline \underset{N\times N}{0} & q.I_{N\times N} \end{array}\right] \quad (4)$$

Assuming $f \star h = g \pmod{q}$, it is clear that the following vector is in the NTRU lattice $\mathcal{L}_{NTRU}$ with a relatively small L_2-norm:

$$v = (\lambda f_0, \lambda f_1, \cdots, \lambda f_{N-1}, g_0, g_1, \cdots g_{N-1}).$$

The constant $\lambda > 0$, which is known as the balancing constant, is chosen to maximize the efficiency of the lattice-reduction algorithms. According to [23] and [26], the best choice for λ is a value around $\|f\| / \|g\|$, where $\|\cdot\|$ denotes L_2-norm. The difference between CML and other types of lattices is that if the vector $v = (f, g)$ is in a CML lattice, then the entire N shifted vectors will also have the same norm and will be still in the lattice.

An adversary tries to find a short vector in $\mathcal{L}_{NTRU}$ by using a lattice-reduction algorithm such as the LLL or Deep Insertion Method. The length of the *target vector* (i.e., decryption key) in $\mathcal{L}_{NTRU}$ is $\mathcal{O}(1/\sqrt{N})$ times shorter than the one predicted by Gaussian heuristic [20, pp. 400–403] and with a high probability it may be one of the shortest vectors in $\mathcal{L}_{NTRU}$. If an adversary manages to find a short enough vector in the NTRU lattice $\mathcal{L}_{NTRU}$, then it may be used as a spurious key. Subsequently in Section 7, we will study that the lattice basis-reduction algorithms (e.g., *LLL* [28] and its variants) are able to solve SVP to within a factor of $2^{\mathcal{O}(N)}$. Hence, in order to ensure that the cryptosystem is secure, the lattice dimension (which is determined by the parameter N) must be large enough to provide a reasonable security level. However, using all peculiar properties of CML [29] and taking advantage of several tricks introduced in [30], as well as using the most efficient lattice-reduction algorithms, one may only be able to break NTRU-107 (N=107) [27].

3 A Brief Introduction to Quaternion Algebra

In this section, quaternion algebra is briefly introduced. Interested reader may refer to [31] or [32] for a more elaborate description. Throughout this paper, we take an *algebra* $\mathbb{A}$ to mean a finite-dimensional vector space $\mathbb{V}$ over a field $\mathbb{F}$ equipped with a bilinear and distributive multiplication (denoted by $\circ$). Similarly, by $\mathcal{R}$-algebra we mean a finite-dimensional $\mathcal{R}$-module with identity over the commutative ring $\mathcal{R}$ endowed with a bilinear multiplication.

Real quaternions, denoted by $\mathbb{H} := \{\alpha + \beta.i + \gamma.j + \delta.k \mid \alpha, \beta, \gamma, \delta \in \mathbb{R}\}$, is the second normed division algebra in the sense of Cayley-Dickson doubling method. A quaternion can also be shown by ordinary vector notations $q = \langle \alpha, \beta, \gamma, \delta \rangle$ over $\mathbb{R}^4$ or by $q = \langle \alpha, \beta \rangle$ over $\mathbb{C}^2$, as long as there is no ambiguity. As a vector space, addition and scalar multiplication are defined by ordinary component-wise vector addition and scalar multiplication. However, multiplication of two quaternions (which is not commutative) is defined according to the following rules:

$$i^2 = j^2 = k^2 = -1 \text{ and } ij = -ji = k.$$

According to the above rules defined for multiplication of the basis elements, the formula for multiplication of two quaternions $\boldsymbol{q}_0 := a + b.i + c.j + d.k$ and $\boldsymbol{q}_1 := \alpha + \beta.i + \gamma.j + \delta.k \in \mathbb{H}$ is as follows

$$\begin{aligned} \boldsymbol{q}_0 \circ \boldsymbol{q}_1 = \; & (a.\alpha - b.\beta - d.\delta - c.\gamma) + \\ & + (a.\beta + b.\alpha - d.\gamma + c.\delta)\,.i \\ & + (d.\beta + c.\alpha + a.\gamma - b.\delta)\,.j \\ & + (b.\gamma + a.\delta - c.\beta + d.\alpha)\,.k \end{aligned} \quad (5)$$

The set of real quaternions together with ordinary addition and multiplication defined as above, forms a *skew field*. For each quaternion $\alpha + \beta.i + \gamma.j + \delta.k$, the conjugate, denoted by $\bar{q}$, is given by $\bar{q} = \alpha - \beta.i - \gamma.j - \delta.k$, and the norm is defined by $N(q) = q \circ \bar{q} = \bar{q} \circ q = \alpha^2 + \beta^2 + \gamma^2 + \delta^2$. The inverse of the quaternion q is defined by $q^{-1} = \frac{\bar{q}}{N(q)}$, provided that $N(q) \neq 0$. The set of all real quaternions with norm 1 forms a non-commutative multiplicative group known as $SU(2)$, which is isomorphic to the multiplicative group of all 2×2 matrices of determinant 1 over $\mathbb{C}$.

Quaternion algebra can be generalized by replacing the field of real numbers $\mathbb{R}$ by any commutative ring $\mathcal{R}$ with identity. Moreover, instead of defining $i^2 = j^2 = k^2 = -1$ and $ij = -ji = k$, one can define i, j and k as $i^2 = a$, $j^2 = b$, $k^2 = -ab$ and $ij = -ji = k$, provided that the product ab is not zero. Assume $\mathcal{R}$ is an arbitrary commutative ring of characteristic not equal to 2. Then, the quaternion algebra $\mathbb{A}$ can be

defined over $\mathcal{R}$ as

$$\mathbb{A} := \left(\frac{a,b}{\mathcal{R}}\right) = \left\{\alpha+\beta.i+\gamma.j+\delta.k \middle| \alpha,\beta,\gamma,\delta\in\mathcal{R},\ i^2=a, j^2=b, ij=-ji=k\right\}.$$

Clearly, if we let a and b be -1 and $\mathcal{R}$ be the field of real numbers $\mathbb{R}$, we obtain the Hamiltonian quaternion, i.e., $\mathbb{H} = \left(\frac{-1,-1}{\mathbb{R}}\right)$. Based on the choice of a, b and the nature of the ring $\mathcal{R}$, we get two different isomorphic types for $\mathbb{A} = \left(\frac{a,b}{\mathcal{R}}\right)$:

(1) $\mathbb{A} = \left(\frac{a,b}{\mathcal{R}}\right)$ is a *Euclidean division ring* (i.e., a ring that is equipped with a degree function, see [33, pp. 151]) if and only if $\mathcal{R}$ is a field of characteristic 0 and for $q \in \left(\frac{a,b}{\mathbb{F}}\right)$, $N(q) = 0$ implies $q = 0$.

(2) $\mathbb{A} = \left(\frac{a,b}{\mathcal{R}}\right)$ is isomorphic to $M_2(\mathcal{R})$, the ring of all 2×2 matrices with entries from $\mathcal{R}$. Such an algebra is called a *split* algebra. In a split algebra, there are some nonzero elements $q \in \mathbb{A}$ which have no multiplicative inverses. Assuming $\mathcal{R} := GF(p)$ or $\mathcal{R} := GF(p^n)$, algebra $\mathbb{A} = \left(\frac{a,b}{\mathcal{R}}\right)$ is absolutely a split algebra [32, 34].

Definition 1. The set of all integral quaternions $\mathbb{L}$, i.e., the set of quaternions whose all components are in $\mathbb{Z}$, is known as Lipschitz quaternions and indeed forms a subring of the Hamiltonian quaternions $\mathbb{H}$.

$$\mathbb{L} = \{a + bi + cj + dk \mid a, b, c, d \in \mathbb{Z}\} \tag{6}$$

The set of all Lipschitz quaternions may be regarded as a lattice in $\mathbb{R}^4$ which is not *densely well-packed* (in the sense of Conway-Smith, see [31]) and so, factorization of a Lipschitz quaternion into primes (up to reordering and units) is far from unique.

4 Algebraic Structure of the Proposed Scheme

We begin this section by defining the following *algebras* over the finite fields $GF(p)$ and $GF(q)$, where p and q are prime numbers

$$\begin{aligned}\mathbb{L}_p &:= \{a + bi + cj + dk \mid a, b, c, d \in GF(p)\}\\ \mathbb{L}_q &:= \{a + bi + cj + dk \mid a, b, c, d \in GF(q)\}.\end{aligned} \tag{7}$$

Evidently, $\mathbb{L}_p$ and $\mathbb{L}_q$ are two *finite split algebras* which are isomorphic to $M_2(\mathbb{F}_p)$ and $M_2(\mathbb{F}_q)$, respectively, while the ring of Lipschitz quaternions $\mathbb{L}$ (6) is a normed division algebra. Let ω be the ring homomorphism from $\mathbb{Z}$ to $\mathbb{Z}_p$, given by $\omega(x) = x \bmod p$. We define the maps ψ_p and ψ_q from $\mathbb{L}$ to $\mathbb{L}_p$ and $\mathbb{L}_q$, respectively:

$$\begin{aligned}&\psi_p\left(a_0 + a_1 i + a_2 j + a_3 k\right)\\ &\quad= \omega(a_0) + \omega(a_1)i + \omega(a_2)j + \omega(a_3)k\\ &\quad= [a_0]_p + [a_1]_p i + [a_2]_p j + [a_3]_p k\end{aligned} \tag{8}$$

ψ_p will be defined in the same manner.

One can easily verify that the maps ψ_p and ψ_q are homomorphisms from $\mathbb{L}$ to $\mathbb{L}_p$ and $\mathbb{L}_q$, respectively. Thus, every element in the finite split algebras $\mathbb{L}_p$ and $\mathbb{L}_q$ can be represented by a coset representative in $\mathbb{L}$. Evidently, there are infinite coset representatives for each of the equivalence class in $\mathbb{L}$. Sometimes, certain conditions can be imposed on each coset representative to form a complete and unique set of representatives for the equivalence classes (or a set of possibly non-unique representatives that satisfies some specific conditions). For example, the representatives can be chosen from the equivalence classes such that each representative has a minimum Euclidean norm among its infinite alternatives.

Before we proceed, let us prove the following lemma which will be helpful in the proof of the main result of Section 7:

Lemma 1. *Given* $\boldsymbol{H} := \langle h_0, h_1, h_2, h_3\rangle \in \mathbb{L}_p$, *and assuming that the quaternionic equation* $\boldsymbol{F} \circ \boldsymbol{H} = \boldsymbol{G}$ *has at least one solution in* $\mathbb{L}_p$, *the set of all solutions* (*which are not all distinct*) *forms an integer lattice of dimension 8 in* $\mathbb{Z}^8$.

Proof. Let $\boldsymbol{\mathcal{F}} := \langle f_0, f_1, f_2, f_3\rangle$ and $\boldsymbol{\mathcal{G}} := \langle g_0, g_1, g_2, g_3\rangle \in \mathbb{L}_p$ be a pair of solutions for the quaternionic equation $\boldsymbol{F} \circ \boldsymbol{H} = \boldsymbol{G}$. According to the multiplication formula (5), the relation $\boldsymbol{F} \circ \boldsymbol{H} = \boldsymbol{G} \in \mathbb{L}_p$ leads to the following system of linear equations

$$\begin{cases} f_0.h_0 - f_1.h_1 - f_2.h_2 - f_3.h_3 = g_0 + k_0.p\\ f_0.h_1 + f_1.h_0 + f_2.h_3 - f_3.h_2 = g_1 + k_1.p\\ f_0.h_2 - f_1.h_3 + f_2.h_0 + f_3.h_1 = g_2 + k_2.p\\ f_0.h_3 + f_1.h_2 - f_2.h_1 + f_3.h_0 = g_3 + k_3.p \end{cases} \tag{9}$$

Now, consider the lattice $\mathcal{L}_h$ generated by the rows of the following matrix

$$M_h := \left[\begin{array}{cccc|cccc} 1&0&0&0 & h_0 & h_1 & h_2 & h_3\\ 0&1&0&0 & -h_1 & h_0 & -h_3 & h_2\\ 0&0&1&0 & -h_2 & h_3 & h_0 & -h_1\\ 0&0&0&1 & -h_3 & -h_2 & h_1 & h_0\\ \hline 0&0&0&0 & p & 0 & 0 & 0\\ 0&0&0&0 & 0 & p & 0 & 0\\ 0&0&0&0 & 0 & 0 & p & 0\\ 0&0&0&0 & 0 & 0 & 0 & p \end{array}\right] \tag{10}$$

As we can see, the rows of the above matrix are linearly independent and the lattice $\mathcal{L}_h$ is full rank. From (9), it is clear that

$$\langle f_0, f_1, f_2, f_3, -k_0, -k_1, -k_2, -k_3\rangle \cdot M_h = \langle f_0, f_1, f_2, f_3, g_0, g_1, g_2, g_3\rangle \tag{11}$$

and hence, $\langle f_0, f_1, f_2, f_3, g_0, g_1, g_2, g_3\rangle \in \mathcal{L}_h$. Conversely, by taking a linear combination of the rows of M_h, we can get the vectors of the form $\langle f_0, f_1, f_2, f_3, g_0, g_1, g_2, g_3\rangle$ that satisfy the system of equations (9). Thus, the lattice $\mathcal{L}_h$ of determinant p^4 comprises all the solutions to the equation $\boldsymbol{F} \circ \boldsymbol{H} = \boldsymbol{G}$ as one of the lattice points. □

Consider the rings $\mathbb{Z}_p[x]/(x^N-1)$ and $\mathbb{Z}_q[x]/(x^N-1)$ that are used in NTRU and let define the following quaternionic algebras $\mathbb{A}$, $\mathbb{A}_p$ and $\mathbb{A}_q$:

$$\mathbb{A} := \left(\frac{-1,-1}{\mathbb{Z}[x]/(x^N-1)}\right)$$

$$\mathbb{A}_p = \left(\frac{-1,-1}{\mathbb{Z}_p[x]/(x^N-1)}\right), \mathbb{A}_q = \left(\frac{-1,-1}{\mathbb{Z}_q[x]/(x^N-1)}\right). \tag{12}$$

For simplicity, p, q and N are all assumed to be prime numbers. Since $\mathbb{Z}_p[x]/(x^N-1)$ and $\mathbb{Z}_q[x]/(x^N-1)$ are finite rings of characteristics p and q, respectively, one can easily conclude that $\mathbb{A}_p$ and $\mathbb{A}_q$ are *finite split algebras*. In other words, $\mathbb{A}_p$ and $\mathbb{A}_q$ algebras possess all features of quaternion algebras, except that there are some nonzero elements whose norm is zero, and naturally such elements do not have a multiplicative inverse. Let us elaborate more on algebras $\mathbb{A}_p$ and $\mathbb{A}_q$

$$\begin{aligned}\mathbb{A}_p :&= \left(\frac{-1,-1}{\mathbb{Z}_p[x]/(x^N-1)}\right)\\ &= \{f_0(x)+f_1(x).i+f_2(x).j+f_3(x).k \mid\\ &\quad f_0,f_1,f_2,f_3 \in \mathbb{Z}_p[x]/(x^N-1)\}.\end{aligned}$$

$$\begin{aligned}\mathbb{A}_q :&= \left(\frac{-1,-1}{\mathbb{Z}_q[x]/(x^N-1)}\right)\\ &= \{g_0(x)+g_1(x).i+g_2(x).j+g_3(x).k \mid\\ &\quad g_0,g_1,g_2,g_3 \in \mathbb{Z}_q[x]/(x^N-1)\}.\end{aligned}$$

Assume that $q_0, q_1 \in \mathbb{A}_p$ (or $\mathbb{A}_q$), $q_0 = a(x)+b(x).i+c(x).j+d(x).k$ and $q_1 = \alpha(x)+\beta(x).i+\gamma(x).j+\delta(x).k$. Then, the addition and multiplication of two quaternions, norm and multiplicative inverse are defined as follows

- Addition: $q_0+q_1 = (a(x)+\alpha(x))+(b(x)+\beta(x)).i+(c(x)+\gamma(x)).j+(d(x)+\delta(x)).k$.
- Multiplication:
$$\begin{aligned}q_0 \circ q_1 = &(a(x)\star\alpha(x)-b(x)\star\beta(x)-d(x)\star\delta(x)-c(x)\star\gamma(x))\\ &+(a(x)\star\beta(x)+b(x)\star\alpha(x)-d(x)\star\gamma(x)+c(x)\star\delta(x)).i\\ &+(d(x)\star\beta(x)+c(x)\star\alpha(x)+a(x)\star\gamma(x)-b(x)\star\delta(x)).j\\ &+(b(x)\star\gamma(x)+a(x)\star\delta(x)-c(x)\star\beta(x)+d(x)\star\alpha(x)).k,\end{aligned}$$
where $\star$ denotes the convolution product, and addition and multiplication of the coefficients are performed modulo p and q.
- Conjugation: $\bar{q}_0 = a(x)-b(x).i-c(x).j-d(x).k$.
- Norm: By a slight abuse of the word *norm*, we define the squared norm of a quaternion as $N(q_0) = q_0 \circ \bar{q}_0 = a(x)^2+b(x)^2+c(x)^2+d(x)^2$.
- Multiplicative inverse
$$\begin{aligned}&N(q_0) \neq 0 \Rightarrow\\ &q_0^{-1} = \frac{\bar{q}_0}{N(q_0)}\\ &\quad = \left(a(x)^2+b(x)^2+c(x)^2+d(x)^2\right)^{-1}\cdot\\ &\quad (a(x)-b(x).i-c(x).j-d(x).k).\end{aligned}$$

Note that multiplication of two polynomials and inverse of a polynomial are taken in the underlying ring. The following operations will be needed for computing the inverse of an element in $\mathbb{A}_q$

(I) Calculation of $g(x) \leftarrow a(x)^2+b(x)^2+c(x)^2+d(x)^2$ in the underlying ring $\mathbb{Z}_q[x]/(x^N-1)$ (including $4.N^2$ multiplications and $3.N$ additions) with the worst-case complexity of $\mathcal{O}(N^2)$.
(II) Calculation of $g(x)^{-1}$ in the ring $\mathbb{Z}_q[x]/(x^N-1)$ with complexity of $\mathcal{O}(N^2 log(q^2))$, where q is the characteristic of the underlying field/ring([35]).
(III) Calculation of $g(x)^{-1}.(a(x)-b(x).i-c(x).j-d(x).k)$ (including $4N^2$ multiplications) with the worst-case complexity of $\mathcal{O}(N^2)$.

One can easily prove that the rings $\left(\frac{-1,-1}{\mathbb{Z}_p[x]/(x^N-1)}\right)$ and $\left(\frac{-1,-1}{\mathbb{Z}_q[x]/(x^N-1)}\right)$ are isomorphic to the ring of $N\times N$ circulant matrices with entries in $\mathbb{F}=\mathbb{Z}_p$ and $\mathbb{F}=\mathbb{Z}_q$, respectively. Thus, each of the isomorphic representation of these rings can be used without any ambiguity. Hence, we will use polynomial representation for the description of the proposed scheme and matrix representation for lattice analysis.

Definition 2. For a quaternion $\boldsymbol{F} := f_0(x)+f_1(x).i+f_2(x).j+f_3(x).k$ in $\mathbb{A}$, where $f_i(x) := \sum_{j=0}^{N-1} f_{i,j}.x^j := [f_{i,0}, f_{i,1}, ..., f_{i,N-1}]$, define the following notations

$$\|\boldsymbol{F}\|_\infty = \max_{0\leqslant i\leqslant 3,\ 0\leqslant j\leqslant N-1}(f_{i,j}) - \min_{0\leqslant i\leqslant 3,\ 0\leqslant j\leqslant N-1}(f_{i,j}) \tag{13}$$

$$\|\boldsymbol{F}\|_2 = \sqrt{\sum_{i=0}^{3}\sum_{j=0}^{N-1} f_{i,j}^2} \tag{14}$$

After setting up the required notation and algebras $\mathbb{A}$, $\mathbb{A}_p$ and $\mathbb{A}_q$, we describe the proposed scheme in next section in detail.

5 Proposed Scheme: QTRU

In the proposed cryptosystem, encryption, decryption and key generation are taken place in a *module*, and

similar to NTRU, the security of the cryptosystem depends on three parameters (N, p, q) and four subsets $\mathcal{L}_f, \mathcal{L}_m, \mathcal{L}_\phi, \mathcal{L}_g \subset \mathbb{A} = \left(\frac{-1,-1}{\mathbb{Z}[x]/(x^N-1)}\right)$. The constants N, p and q play a similar role as in NTRU except that for simplicity these constants are supposed to be all prime numbers. The constants d_f, d_g, d_ϕ, and d_m and the subsets $\mathcal{L}_f$, $\mathcal{L}_\phi$, $\mathcal{L}_g$, and $\mathcal{L}_m$ are defined exactly as in Table 1. The proposed cryptosystem operates as described below.

Key Generation. In order to generate a pair of public and private keys, two *small quaternions* $\boldsymbol{F}$ and $\boldsymbol{G}$ are randomly generated. By "small quaternion" we mean a quaternion with a small $\|\cdot\|_\infty$ norm (see Definition 2).

$$\boldsymbol{F} = f_0 + f_1.\mathbf{i} + f_2.\mathbf{j} + f_3.\mathbf{k}, \quad f_0, f_1, f_2, f_3 \in \mathcal{L}_f,$$

$$\boldsymbol{G} = g_0 + g_1.\mathbf{i} + g_2.\mathbf{j} + g_3.\mathbf{k}, \quad g_0, g_1, g_2, g_3 \in \mathcal{L}_g.$$

The quaternion $\boldsymbol{F}$ must be invertible in $\mathbb{A}_p = \left(\frac{-1,-1}{\mathbb{Z}_p[x]/(x^N-1)}\right)$ and $\mathbb{A}_q = \left(\frac{-1,-1}{\mathbb{Z}_q[x]/(x^N-1)}\right)$. As mentioned in the previous section, the necessary and sufficient condition for $\boldsymbol{F}$ to be invertible in $\mathbb{A}_p$ and $\mathbb{A}_q$ is that the polynomial $\|\boldsymbol{F}\| = (f_0^2 + f_1^2 + f_2^2 + f_3^2)$ be invertible in the rings $\mathbb{Z}_p[x]/(x^N-1)$ and $\mathbb{Z}_q[x]/(x^N-1)$. Given the fact that invertibility of quaternion $\boldsymbol{F}$ depends on the four polynomials f_0, f_1, f_2, f_3, there is much more freedom in choosing the polynomials. If the generated quaternion is not invertible in $\mathbb{A}_p$ and $\mathbb{A}_q$, a new quaternion can easily be generated.

In the second step, the inverses of $\boldsymbol{F}$ (denoted by $\boldsymbol{F}_p$ and $\boldsymbol{F}_q$) will be computed as follows

$$\boldsymbol{F}_p = \underbrace{\left(f_0^2 + f_1^2 + f_2^2 + f_3^2\right)^{-1}}_{\text{a scalar from } \mathbb{Z}_p[x]/(x^N-1)} \cdot \boldsymbol{F}^*_{=\mu_0+\mu_1.i+\mu_2.j+\mu_3.k}$$

$$\boldsymbol{F}_q = \underbrace{\left(f_0^2 + f_1^2 + f_2^2 + f_3^2\right)^{-1}}_{\text{a scalar from } \mathbb{Z}_q[x]/(x^N-1)} \cdot \boldsymbol{F}^*_{=\eta_0+\eta_1.i+\eta_2.j+\eta_3.k.}$$

Now, the public key, which is a quaternion, is calculated and then will be made public as follows.

$$\begin{aligned}\boldsymbol{H} = \boldsymbol{F}_q \circ \boldsymbol{G} = \\ &(\eta_0 \star g_0 - \eta_1 \star g_1 - \eta_3 \star g_3 - \eta_2 \star g_2) + \\ &(\eta_0 \star g_1 + \eta_1 \star g_0 - \eta_3 \star g_2 + \eta_2 \star g_3).i + \\ &(\eta_3 \star g_1 + \eta_2 \star g_0 + \eta_0 \star g_2 - \eta_1 \star g_3).j + \\ &(\eta_1 \star g_2 + \eta_0 \star g_3 - \eta_2 \star g_1 + \eta_3 \star g_0).k.\end{aligned} \tag{15}$$

The quaternions $\boldsymbol{F}$, $\boldsymbol{F}_p$ and $\boldsymbol{F}_q$ will be kept secret in order to be used in the decryption phase. It seems that the key generation in QTRU is 16 times slower than that of NTRU, when the same parameters (N, p, q) are used in both cryptosystems. *However, in QTRU, we can work with a smaller dimension N, without reducing the system security.*

We note that if the coefficients of i, j, and k in $\boldsymbol{F}$ and $\boldsymbol{G}$ are all set to zero, then QTRU is completely converted into NTRU.

Encryption. In the encryption process, the system first generates a random quaternion $\boldsymbol{\Phi} = \phi_0 + \phi_1.\mathbf{i} + \phi_2.\mathbf{j} + \phi_3.\mathbf{k}$, where $\phi_0, \phi_1, \phi_2, \phi_3 \in \mathcal{L}_\phi$. The plaintext must be converted into a quaternion $\boldsymbol{M} = m_0 + m_1.\mathbf{i} + m_2.\mathbf{j} + m_3.\mathbf{k}$, including four small polynomials $m_0, m_1, m_2, m_3 \in \mathcal{L}_m$. The messages could be generated from the same or four different sources but transformed into one quaternion based on a simple and pre-determined encoding scheme. The ciphertext will be computed as follows

$$\boldsymbol{E} = p.\boldsymbol{H} \circ \boldsymbol{\Phi} + \boldsymbol{M} \in \mathbb{A}_q. \tag{16}$$

Encryption needs one quaternionic multiplication including 16 convolution multiplications with the worst case running time $\mathcal{O}(N^2)$, and 4 polynomial additions which take fewer than $\mathcal{O}(N)$ steps. In the encryption phase, a total of four data vectors are encrypted at once.

Decryption. In the first step, the received ciphertext $\boldsymbol{E}$ is multiplied by the private key $\boldsymbol{F}$ on the left

$$\begin{aligned}\boldsymbol{B} := \boldsymbol{F} \circ \boldsymbol{E} &= \boldsymbol{F} \circ (p.\boldsymbol{H} \circ \boldsymbol{\Phi} + \boldsymbol{M}) \mod q \\ &= (\boldsymbol{F} \circ p.\boldsymbol{H} \circ \boldsymbol{\Phi} + \boldsymbol{F} \circ \boldsymbol{M}) \mod q \\ &= (p.\boldsymbol{F} \circ \boldsymbol{F}_q \circ \boldsymbol{G} \circ \boldsymbol{\Phi} + \boldsymbol{F} \circ \boldsymbol{M}) \mod q \\ &= (p.\boldsymbol{G} \circ \boldsymbol{\Phi} + \boldsymbol{F} \circ \boldsymbol{M}) \in \mathbb{A}_q.\end{aligned}$$

The coefficients of the four polynomials in the resulting quaternion $\boldsymbol{B}$ must be reduced mod q, and all of the coefficients are chosen in the interval $(-q/2, +q/2]$. In other words, the set of distinct representatives is chosen to be $\Omega = \{-q/2 + 1, \cdots, +q/2\}$. Assuming that $\boldsymbol{B} \in \mathbb{Z}_q[x]/(x^N - 1)$ is exactly equal to $p.\boldsymbol{G} \circ \boldsymbol{\Phi} + \boldsymbol{F} \circ \boldsymbol{M}$ in $\mathbb{A}$, when $\boldsymbol{B}$ is reduced mod p, the term $p.\boldsymbol{G} \circ \boldsymbol{\Phi}$ vanishes and $\boldsymbol{F} \circ \boldsymbol{M} \pmod p$ remains. In order to extract the original message $\boldsymbol{M}$, it will suffice to multiply $\boldsymbol{F} \circ \boldsymbol{M} \pmod p$ by $\boldsymbol{F}_p$ on the left and adjust the resulting coefficients within the interval $\Lambda = [-p/2, +p/2]$.

6 Analyzing QTRU

In this section, we analyze QTRU and discuss the probability of successful decryption, key security, message security, and the message expansion rate. Moreover, we suggest a set of parameters for the proposed scheme.

Successful Decryption. Probability of successful decryption in QTRU is calculated in the same way as NTRU and under the same assumptions considered in [21] and [17]. Moreover, for successful decryption in QTRU, all integer coefficients of $\boldsymbol{F} \circ \boldsymbol{E} =$

$(p.\boldsymbol{G}\circ\boldsymbol{\Phi}+\boldsymbol{F}\circ\boldsymbol{M})$ must lie in the interval $\left[\frac{-q+1}{2}, \frac{+q-1}{2}\right]$. Hence, we obtain

$$\boldsymbol{A} := \boldsymbol{F}\circ\boldsymbol{E} = (p.\boldsymbol{G}\circ\boldsymbol{\Phi}+\boldsymbol{F}\circ\boldsymbol{M}) = a_0+a_1.i+a_2.j+a_3.k, \tag{17}$$

where

$$\begin{aligned}
a_0 :&= p.g_0\star\phi_0 - p.g_1\star\phi_1 - p.g_3\star\phi_3 - p.g_2\star\phi_2 \\
&\quad + f_0\star m_0 - f_1\star m_1 - f_3\star m_3 - f_2\star m_2 \\
&= [a_{0,0}, a_{0,1}, ..., a_{0,N-1}], \\
a_1 :&= p.g_0\star\phi_1 + p.g_1\star\phi_0 - p.g_3\star\phi_2 + p.g_2\star\phi_3 \\
&\quad + f_0\star m_1 + f_1\star m_0 - f_3\star m_2 + f_2\star m_3 \\
&= [a_{1,0}, a_{1,1}, ..., a_{1,N-1}], \\
a_2 :&= p.g_3\star\phi_1 + p.g_2\star\phi_0 + p.g_0\star\phi_2 - p.g_1\star\phi_3 \\
&\quad + f_3\star m_1 + f_2\star m_0 + f_0\star m_2 - f_1\star m_3 \\
&= [a_{2,0}, a_{2,1}, ..., a_{2,N-1}], \\
a_3 :&= p.g_1\star\phi_2 + p.g_0\star\phi_3 - p.g_2\star\phi_1 + p.g_3\star\phi_0 \\
&\quad + f_1\star m_2 + f_0\star m_3 - f_2\star m_1 + f_3\star m_0 \\
&= [a_{3,0}, a_{3,1}, ..., a_{3,N-1}].
\end{aligned}$$

One can easily estimate that if we consider all NTRU assumptions, the expected values for all coefficients of a_0, a_1, a_2, a_3 remain equal to zero and their variance quadruples. We know that $f_i\star m_j (i,j=0,1,2,3)$ and $g_i\star\phi_j (i,j=0,1,2,3)$ are the products of two small polynomials and that the coefficients of f_i, g_i, and ϕ_i are assumed to be independent random variables that randomly take one of the values: -1, 0, and $+1$. Now, according to the definition of the subsets $\mathcal{L}_f$ and $\mathcal{L}_g$ from Table 1, we obtain

$$\begin{aligned}
f_i &= [f_{i,0}, f_{i,1}, ..., f_{i,N-1}] \quad i=0,1,2,3, \\
g_i &= [g_{i,0}, g_{i,1}, ..., g_{i,N-1}] \quad i=0,1,2,3, \\
\phi_i &= [\phi_{i,0}, \phi_{i,1}, ..., \phi_{i,N-1}] \quad i=0,1,2,3,
\end{aligned}$$

$$\Pr(f_{i,j}=1) = \frac{d_f}{N}, \quad \Pr(f_{i,j}=-1) = \frac{d_f-1}{N} \approx \frac{d_f}{N},$$

$$\Pr(f_{i,j}=0) = \frac{N-2d_f}{N},$$

$$\Pr(g_{i,j}=1) = \Pr(g_{i,j}=-1) = \frac{d_g}{N},$$

$$\Pr(g_{i,j}=0) = \frac{N-2d_g}{N},$$

$$\Pr(\phi_{i,j}=1) = \Pr(\phi_{i,j}=-1) = \frac{d_\phi}{N},$$

$$\Pr(\phi_{i,j}=0) = \frac{N-2d_\phi}{N},$$

$$\Pr(m_{i,j}=j) = \frac{1}{p}, \quad i=0,\cdots,3, \quad j=\frac{-p+1}{2}...\frac{+p-1}{2}.$$

Under the above assumptions, we get $E[f_{i,j}] \approx 0$, $E[g_{i,j}]=0$, $E[r_{i,j}]=0$, and $E[m_{i,j}]=0$. Therefore, we have

$$E[a_{i,j}] = 0 \quad i=0,1,2,3, \quad j=0,...,N-1.$$

In order to calculate $Var[a_{i,j}]$, analogous to NTRU, it is sufficient to write

$$Var[\phi_{i,k}.g_{j,l}] = \frac{4d_\phi.d_g}{N^2} \tag{18}$$

$$Var[f_{i,k}.m_{j,l}] = \frac{d_f(p-1).(p+1)}{6.N} \tag{19}$$

$$i,j=0,1,2,3, \quad k,l=0,...,N-1.$$

Thus we have,

$$\begin{aligned}
Var[a_{0,k}] = Var[&\sum_{i+j\overset{N}{\equiv}k} (p.g_{0,i}\star\phi_{0,j} - p.g_{1,i}\star\phi_{1,j} \\
&- p.g_{3,i}\star\phi_{3,j} - p.g_{2,i}\star\phi_{2,j} + f_{0,i}\star m_{0,j} - f_{1,i}\star m_{1,j} \\
&- f_{3,i}\star m_{3,j} - f_{2,i}\star m_{2,j})].
\end{aligned} \tag{20}$$

By substituting the values of $Var[\phi_{i,k}g_{j,l}]$ and $Var[f_{i,k}m_{j,l}]$ from (18) and (19), we obtain

$$Var[a_{0,k}] = \frac{16p^2d_\phi d_g}{N} + \frac{4d_f(p-1)(p+1)}{6}.$$

In a similar way, we have

$$Var[a_{1,k}] = Var[a_{2,k}] = Var[a_{3,k}] = \frac{16p^2d_\phi d_g}{N} + \frac{4d_f(p-1)(p+1)}{6}.$$

It is desirable to calculate the probability that $a_{i,k}$ lies within $\left[\frac{-q+1}{2}, \frac{+q-1}{2}\right]$, which implies successful decryption. With the assumption that $a_{i,k}$'s have normal distribution with zero mean and the variance calculated as above, we have

$$\begin{aligned}
\Pr\left(|a_{i,k}| \le \frac{q-1}{2}\right) &= \Pr\left(-\frac{q-1}{2} \le a_{i,k} \le \frac{q-1}{2}\right) \\
&= 2\Phi\left(\frac{q-1}{2\sigma}\right) - 1,
\end{aligned}$$

where Φ denotes the standard normal distribution function and

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

Assuming that $a_{i,k}$'s are independent random variables, the probability for successful decryption in QTRU can be calculated through the following two observations

- The probability for each of the messages m_0, m_1, m_2, or m_3 to be correctly decrypted is
$$\left(2\Phi(\frac{q-1}{2\sigma}) - 1\right)^N.$$
- The probability for all the messages m_0, m_1, m_2, and m_3 to be correctly decrypted is
$$\left(2\Phi(\frac{q-1}{2\sigma}) - 1\right)^{4.N}.$$

It is apparent that in QTRU, the variance of the coefficients $(p.\boldsymbol{G}\circ\boldsymbol{\Phi}+\boldsymbol{F}\circ\boldsymbol{M})$ increases by a factor

Table 2. The probability of successful decryption in QTRU, security level of the private key, and message security according to some generic parameters d_ϕ , d_g, d_f, $p = 3$, q, N.

Security Level	N	q	d_f	d_g	d_ϕ	Key Security	Message Security	Message Expansion	Pr(successful decryption)
Moderate	107	127	15	12	5	1.84×10^{60}	7.84×10^{31}	≈ 4.4	0.9997119974
Moderate	107	191	20	12	10	1.84×10^{60}	1.95×10^{53}	≈ 4.4	0.9999971752
High	149	191	20	12	10	7.78×10^{67}	3.38×10^{59}	≈ 4.6	0.9999998808
High	149	191	22	15	12	1.56×10^{79}	7.78×10^{67}	≈ 4.6	0.9999845041
High	149	255	50	20	15	1.986×10^{95}	1.56×10^{79}	≈ 4.6	0.9999563737
High	149	255	35	25	20	2.877×10^{108}	1.99×10^{95}	≈ 4.6	0.9994484943
High	167	255	40	20	18	7.11×10^{99}	1.87×10^{93}	≈ 4.7	0.9999808954
High	167	255	50	20	18	7.11×10^{99}	1.87×10^{93}	≈ 4.7	0.9999167707
High	167	255	40	24	22	4.91×10^{111}	9.60×10^{105}	≈ 4.7	0.9993964435
Highest	211	255	40	20	18	9.24×10^{108}	2.34×10^{101}	≈ 4.8	0.9999974680
Highest	211	255	30	24	22	8.37×10^{122}	1.38×10^{116}	≈ 4.8	0.9999782250
Highest	257	255	40	20	18	2.93×10^{116}	1.13×10^{108}	≈ 5.1	0.9999995888
Highest	257	255	30	24	24	1.29×10^{132}	1.29×10^{132}	≈ 5.1	0.9999923928

of 4 and, hence, the probability for decryption failure increases. In return, constant parameters of the system, including d_ϕ , d_g, d_f, q, and N, can be chosen in such a way that the decryption failure rate in QTRU remains equal to that of NTRU. The rightmost column of Table 2 shows the probability for successful decryption for some proposed values of d_ϕ , d_g, d_f, q, and N.

Brute-Force Attack. To conduct a brute-force attack against QTRU, an attacker who knows the public parameters, including the public key $\boldsymbol{H} = \boldsymbol{F}_q^{-1} \circ \boldsymbol{G}$, d_ϕ, d_g, d_f, q, p and N, must simply try each possible key in $\mathcal{L}_f$ (by multiplying on the left) until (s)he finds a short key for decryption. The size of the key space $\mathcal{L}_f (\approx \mathcal{L}_g)$ is calculated as follows

$$\#\mathcal{L}_f = \binom{N}{d_f}^4 \binom{N-d_f}{d_f}^4 = \frac{(N!)^4}{(d_f!)^8 (N-2d_f)!^4},$$

$$\#\mathcal{L}_g = \binom{N}{d_g}^4 \binom{N-d_g}{d_g}^4 = \frac{(N!)^4}{(d_g!)^8 (N-2d_g)!^4}.$$

Note that just like NTRU, $\boldsymbol{F}$ and all of its scalar rotations $(x^i \cdot \boldsymbol{F})$ can be served as decryption keys. Thus, the total state space which an attacker has to search for an encryption key is about $\#\mathcal{L}_f/N$ (see the values in the 8th column of Table 2). If enough memory is provided, the search time could be reduced to $\sqrt{\#\mathcal{L}_f/N}$ using Meet-In-The-Middle attack [36].

Similarly, in order to find the original message using brute-force attack, the attacker must search in $\mathcal{L}_\phi$. Thus, the message security is $\#\mathcal{L}_\phi/N$ for brute-force attack (shown at the 9th column of Table 2) and $\sqrt{\#\mathcal{L}_\phi/N}$ for Meet-In-The-Middle attack, where

$$\#\mathcal{L}_\phi = \frac{(N!)^4}{(d_\phi!)^8 (N-2d_\phi)!^4}.$$

For typical values of N, a brute-force attack appears to be practically impossible. Therefore, the QTRU cryptosystem seems to be completely secure against the brute-force attack.

Chosen Ciphertext Attacks. Since the basic scheme in QTRU is similar to NTRU, the security and survivability of the proposed cryptosystem against adaptively chosen ciphertext attacks [37] is exactly equivalent to NTRU, and therefore the techniques proposed for NTRU to prevent such attacks [8] (e.g., Padding techniques), can be used just as well for QTRU.

Message Expansion. Similar to NTRU, the length of the encrypted message in QTRU is more than the original message and that is part of the price one has to pay for gaining more speed in both cryptosystems. The expansion ratio can be easily calculated as $\frac{\log |C|}{\log |P|} = \frac{\log q^{4N}}{\log p^{4N}} = \frac{\log q}{\log p}$, where C is the state space for the encrypted message and P is the state space for plaintext; for NTRU and QTRU, this ratio depends merely on p and q. Table 2 shows the message expansion rate for some typical values of p and q. Message expansion rate for typical parameters in both NTRU and QTRU fluctuates between 4 and 5.

7 Analyzing Lattice Attacks Against QTRU

In this section we prove that the security of QTRU relies on the intractability of shortest-vector problem (SVP) in a certain type of lattice which is not fully circular. Obviously, Quaternionic matrices (and, more generally, all the matrices which are defined over a skew field) lack many properties of the matrices which

are defined over a field or commutative ring. For example, determinant is not generally well-defined for quaternionic matrices [1], and, many basic concepts of the lattice theory, such as *unimodular matrices* (i.e., matrices with $det(U) = \pm 1$ which preserve the structure of a lattice), and *fundamental parallelepiped volume* lose their meanings in the context of quaternionic matrices. Since the public key in QTRU is in the form $\boldsymbol{H} = \boldsymbol{F}_q \circ \boldsymbol{G} \in \mathbb{A}_q$, the only way which remains for attacking this special scheme and finding a suitable key for decryption is to expand $\boldsymbol{F} \circ \boldsymbol{H} = \boldsymbol{G} \pmod q$ as a system of linear equations and form a lattice of dimension $8.N$. In the following proposition we prove that the security of the proposed scheme relies on the intractability of SVP in a certain type of lattice.

Proposition 1. *Given the quaternion $\boldsymbol{H} \in \mathbb{A}_q$ and assuming that the quaternionic equation $\boldsymbol{F} \circ \boldsymbol{H} = \boldsymbol{G}$ has at least a pair of solutions $\boldsymbol{\mathcal{F}}, \boldsymbol{\mathcal{G}}$ in $\mathbb{A}_q$, then*

(a) *the set of all pairs of solutions (which are not all distinct), forms the integer lattice*

$$\mathcal{L}_{\mathcal{H}} := Row\ Span \begin{bmatrix} I_{4N\times 4N} & \mathcal{H}_{4N\times 4N}(\boldsymbol{H}) \\ 0_{4N\times 4N} & q.I_{4N\times 4N} \end{bmatrix}$$

of determinant $q^{4.N}$ and rank $8.N$ in $\mathbb{Z}^{8N}$.

(b) *Assume that $\|\boldsymbol{\mathcal{F}}\|_2 \leqslant \lambda$ and $\|\boldsymbol{\mathcal{G}}\|_2 \leqslant \lambda$. If $\lambda \ll \sqrt{\frac{2N.q}{\pi e}}$, then with a probability greater than $1 - \frac{\lambda}{\sqrt{\frac{2N.q}{\pi e}}}$, finding $\langle \boldsymbol{\mathcal{F}}, \boldsymbol{\mathcal{G}} \rangle$ will be turned into the shortest-vector problem in the lattice $\mathcal{L}_{\mathcal{H}}$.*

Proof. (a) Let $\boldsymbol{F} := \langle f_0(x), f_1(x), f_2(x), f_3(x) \rangle \in \mathbb{A}_q$ and $\boldsymbol{G} := \langle g_0(x), g_1(x), g_2(x), g_3(x) \rangle \in \mathbb{A}_q$ be a pair of solutions for the quaternionic equation $\boldsymbol{F} \circ \boldsymbol{H} = \boldsymbol{G}$. Similar to Lemma (1), we expand $\boldsymbol{F} \circ \boldsymbol{H} = \boldsymbol{G}$ as a system of linear equations (Note that for the sake of simplicity, the arguments x have been dropped)

$$\begin{cases} f_0 \star h_0 - f_1 \star h_1 - f_2 \star h_2 - f_3 \star h_3 = g_0 + p.k_0 \\ f_0 \star h_1 + f_1 \star h_0 + f_2 \star h_3 - f_3 \star h_2 = g_1 + p.k_1 \\ f_0 \star h_2 - f_1 \star h_3 + f_2 \star h_0 + f_3 \star h_1 = g_2 + p.k_2 \\ f_0 \star h_3 + f_1 \star h_2 - f_2 \star h_1 + f_3 \star h_0 = g_3 + p.k_3 \end{cases}$$
$$\text{for some } k_0, \ldots, k_3 \in \mathbb{Z}. \qquad (21)$$

where $\star$ denotes the convolution product as defined in (1). Since $\mathbb{Z}[x]/(x^N - 1)$ is isomorphic to the ring of circulant $N \times N$ matrices over $\mathbb{Z}$, let us replace h_0, h_1, h_2, and h_3 by matrices which correspond to them as follows

$$(\mathcal{H}_i)_{N\times N} \triangleq \begin{bmatrix} h_{i,0} & h_{i,1} & h_{i,2} & \cdots & h_{i,N-1} \\ h_{i,N-1} & h_{i,0} & h_{i,1} & & h_{i,N-2} \\ h_{i,N-2} & h_{i,N-1} & h_{i,0} & & h_{i,N-3} \\ \vdots & & & \ddots & \vdots \\ h_{i,2} & h_{i,3} & & & \\ h_{i,1} & h_{i,2} & & \cdots & h_{i,0} \end{bmatrix}$$
$$i := 0, 1, 2, 3.$$

Based on the above notations, we can form the lattice $\mathcal{L}_{\mathcal{H}}$ of dimension $8.N$ spanned by the rows of the following matrix

$$\mathcal{M}_{\mathcal{H}} := \left[\begin{array}{c|cccc} I_{4N\times 4N} & (\mathcal{H}_0) & (\mathcal{H}_1) & (\mathcal{H}_2) & (\mathcal{H}_3) \\ & (-\mathcal{H}_1) & (\mathcal{H}_0) & (-\mathcal{H}_3) & (\mathcal{H}_2) \\ & (-\mathcal{H}_2) & (\mathcal{H}_3) & (\mathcal{H}_0) & (-\mathcal{H}_1) \\ & (-\mathcal{H}_3) & (-\mathcal{H}_2) & (\mathcal{H}_1) & (\mathcal{H}_0) \\ \hline 0_{4N\times 4N} & \multicolumn{4}{c}{q.I_{4N\times 4N}} \end{array}\right]$$

As we can see from the system of linear equations (21) and the matrix $\mathcal{M}_{\mathcal{H}}$, it is clear that the vector $\langle f_0, f_1, f_2, f_3, g_0, g_1, g_2, g_3 \rangle_{1\times 8N}$ is in the lattice $\mathcal{L}_{\mathcal{H}}$, because we can get this vector as a $\mathbb{Z}$-linear combination of the rows of $\mathcal{M}_{\mathcal{H}}$ as follows

$$\underbrace{\langle f_{0,0},\ldots,f_{0,N-1}}_{\boldsymbol{f}_0},\underbrace{\ldots,f_{3,0},\ldots,f_{3,N-1}}_{\boldsymbol{f}_3},\underbrace{-k_{0,0},\ldots,-k_{0,N-1}}_{\boldsymbol{k}_0},\ldots,$$
$$\underbrace{-k_{3,0},\ldots,-k_{3,N-1}}_{\boldsymbol{k}_3}\rangle_{1\times 8.N} \cdot \mathcal{M}_{\mathcal{H}} = \underbrace{\langle f_{0,0},\ldots,f_{0,N-1}}_{\boldsymbol{f}_0},\ldots, \qquad (22)$$
$$\underbrace{f_{3,0},\ldots,f_{3,N-1}}_{\boldsymbol{f}_3},\underbrace{g_{0,0},\ldots,g_{0,N-1}}_{\boldsymbol{g}_0},\ldots\underbrace{g_{3,0},\ldots,g_{3,N-1}}_{\boldsymbol{g}_3}\rangle_{1\times 8.N}$$

Thus we have $\langle f_0, f_1, f_2, f_3, g_0, g_1, g_2, g_3 \rangle_{1\times 8N} \in \mathcal{L}_{\mathcal{H}}$.

(b) Since $\|\boldsymbol{F}\|_2$, and $\|\boldsymbol{G}\|_2$ are less than or equal to λ, it is clear that $\|\langle f_0, f_1, f_2, f_3, g_0, g_1, g_2, g_3 \rangle\|_2 \leqslant \sqrt{2}\lambda$, where

$$\|\langle f_0, \ldots, f_3, g_0, \ldots, g_3 \rangle\|_2 \triangleq \sqrt{\sum_{i=0}^{3}\sum_{j=0}^{N-1} f_{i,j}^2 + \sum_{i=0}^{3}\sum_{j=0}^{N-1} g_{i,j}^2}$$

Based on the Gaussian heuristic, the average length of the shortest nonzero vector in $\mathcal{L}_{\mathcal{H}}$ is (see [20, p. 377])

$$\mathrm{E}\{\|\mathbf{v}_{Shortest}\|_2\} = \sqrt{\frac{n}{2\pi e}}.\det(\mathcal{L}_{\mathcal{H}})^{\frac{1}{n}} \xrightarrow{n=8N,\ \det(\mathcal{L}_{\mathcal{H}})=q^{4N}} \approx \sqrt{\frac{4N.q}{\pi e}}. \qquad (23)$$

So, if $\boldsymbol{F}$ and $\boldsymbol{G}$ are taken short enough such that $\|\langle f_0, f_1, f_2, f_3, g_0, g_1, g_2, g_3 \rangle\|_2 \ll \sqrt{\frac{4N.q}{\pi e}}$, then based on the Markov inequality (which

[1] For quaternionic matrices, the determinant is defined in terms of the cosets modulo the commutator subgroup of the nonzero elements.

states that $\Pr\{Y > \alpha E[Y]\} < \frac{1}{\alpha}$), the vector $\langle f_0, f_1, f_2, f_3, g_0, g_1, g_2, g_3\rangle_{1\times 8.N}$ with a probability greater than $1 - \frac{\sqrt{2}\lambda}{\sqrt{\frac{4N.q}{\pi e}}} = 1 - \frac{\lambda}{\sqrt{\frac{2.N.q}{\pi e}}}$ will be one of the shortest vector in $\mathcal{L}_{\mathcal{H}}$. Consequently, finding a solution to the quaternionic equation $\boldsymbol{F} \circ \boldsymbol{H} = \boldsymbol{G}$ such that $\|\boldsymbol{F}\|_2, \|\boldsymbol{G}\|_2 \ll \sqrt{\frac{4N.q}{\pi e}}$, is transformed into an SVP in the lattice $\mathcal{L}_{\mathcal{H}}$ of dimension $8.N$. □

Here, let us briefly review the results which have been obtained so far with respect to solving SVP and lattice-reduction algorithms. In a lattice of relatively small dimension (e.g., ≤ 70), we can enumerate all short vectors using exhaustive search, but beyond dimension 100, exhaustive search is practically infeasible [38, 39]. Instead, we can use polynomial-time lattice-reduction algorithms to approximate the length of the shortest lattice vectors.

Let $\lambda_1(\mathcal{L}_{\mathcal{H}})$ denotes the length of the shortest non-zero vector in the lattice $\mathcal{L}_{\mathcal{H}}$ and let b_1 be the vector with minimum length returned by a lattice basis reduction algorithm such as LLL, BKZ-LLL or DEEP. Let define *approximation factor* as $\|b_1\| / \lambda_1(\mathcal{L}_{\mathcal{H}})$ and *Hermite factor* as $\|b_1\| / \det(\mathcal{L}_{\mathcal{H}})^{1/n}$. The LLL basis reduction algorithm [28] definitely achieves a Hermite factor $\lessapprox (4/3)^{(n-1)/4} \approx 1.07457^{n-1}$ and an approximation factor $\lessapprox (4/3)^{(n-1)/2} \approx 1.154^n$, but in practice, the LLL algorithm performs much better than the worst-case theoretical bounds. The authors of [38] state that based on extensive experiments performed on a large number of random lattices, the average Hermite factor is about $(1.0219)^n$ for the LLL algorithm, $(1.013)^n$ for BKZ-20 and $(1.011)^n$ for DEEP-50, which is much better than the theoretical bounds in high lattice dimension. A probabilistic analysis presented in [40] confirms that the average-case Hermite factor is close to the experimental results reported in [38].

Despite the fact that the NTRU lattice (of dimension $2.N$) has a special cyclic structure which may be exploited to improve the performance of the lattice-reduction algorithms [29], the best Hermite factor obtained for this kind of lattice is about $(1.01)^{2.N}$ [27], and with this Hermite factor, only NTRU-107 may be broken using current lattice-reduction algorithms.

Based upon the analytical and experimental results presented in [26, 41], the expected running time needed to find a suitable short vector in $\mathcal{L}_{NTRU}$ is exponential in N (assuming $q \approx 2.N$ and $d = d_f = d_g = d_\phi = N/3$). For example, the expected running time for $N = 251$ is estimated as 1.37×10^{13} MIPS-Years.

Now, let us turn our attention to the QTRU lattice $\mathcal{L}_{\mathcal{H}}$ as defined in Proposition (1). It is clear that the vector $\langle f_0, f_1, f_2, f_3, g_0, g_1, g_2, g_3\rangle_{1\times 8.N}$ is in $\mathcal{L}_{\mathcal{H}}$. Finding a short vector in this lattice may be used as the decryption (spurious) key. For the lattice $\mathcal{L}_{\mathcal{H}}$ we can readily find that:

- $\det(\mathcal{L}_{\mathcal{H}}) = q^{4.N}$.
- Assuming $d \approx N/3$, we have
  $\lambda = \|\langle f_0, f_1, f_2, f_3\rangle_{1\times 8.N}\|_2 = \|\langle g_0, g_1, g_2, g_3\rangle_{1\times 8.N}\|_2 \approx \sqrt{8.d} \approx 1.633\sqrt{N}$,
  because $\forall \boldsymbol{F} \in \mathcal{L}_f,\ \boldsymbol{G} \in \mathcal{L}_g \Rightarrow \lambda = \|\boldsymbol{F}\|_2 = \|\boldsymbol{G}\|_2 \approx \sqrt{8d} \approx \sqrt{8N/3} \approx 1.633\sqrt{N}$
- By the Gaussian heuristic (23), the target vectors in $\mathcal{L}_{\mathcal{H}}$ are about $\mathcal{O}(\sqrt{q})$ shorter than the Gaussian expected shortest length.
- Based on Proposition (1), with a probability greater than $1 - \frac{1.633.\sqrt{N}}{\sqrt{\frac{2.N.q}{\pi e}}} \approx 1 - \frac{3.37}{\sqrt{q}}$, the problem of finding a decryption (spurious) key for QTRU is equal to solving SVP to within a factor of approximately $\sqrt{q}$, which is believed to be intractable for lattice of dimension greater than 334 (2×167) [26, 41].

Putting together all of the preceding estimates and observations, yields the following corollary:

Corollary 1. *Assuming a Hermite factor $H = (1.01)^n$, and given the quaternion $\boldsymbol{H} \in \mathbb{A}_q$, solving the quaternionic equation $\boldsymbol{F} \circ \boldsymbol{H} = \boldsymbol{G}$ over the algebra $\mathbb{A}$ and finding a spurious key for QTRU is intractable for $N > 41$ ($8.N > 328$).*

Note that contrary to the NTRU lattice, $\mathcal{L}_{\mathcal{H}}$ is not completely circular and achieving a Hermite factor $\approx (1.01)^n$ seems to be too optimistic for this type of lattice. In addition, the open problems and doubts arisen with respect to the cyclic structure of the NTRU lattice may not exist in this case. As we mentioned earlier, the main open problem is "Is it possible that the cyclic structure of the convolutional lattices contribute to the improvement of the existing lattice-reduction algorithms and finding the shortest vector in polynomial time?"

A note on Coppersmith's Attack against Non-commutative NTRU. Soon after Coppersmith and Shamir suggested that lattice-based attack on NTRU might be useful for finding a spurious key, Hoffstein and Silverman proposed a new public-key cryptosystem called Non-commutative NTRU [42]. This cryptosystem is based on the group ring $\mathcal{R} = \mathbb{Z}[D_N]$, where D_N is the dihedral group of order $2N$, and it uses a commutative subring $\mathcal{R}_0 = \{\alpha \in \mathcal{R} | \alpha Y = Y\alpha\}$, where Y is an element of order two in D_N [43].

The Non-commutative NTRU was quickly broken by Coppersmith [44]. He exploited some properties of the subset $\mathcal{R}_1 = \{\alpha \in \mathcal{R} | \alpha Y = -Y\alpha\}$. Looking at $\mathcal{R}_0$ and $\mathcal{R}_1$, Coppersmith makes fake private keys. Then, he creates a linear map θ: $\mathcal{R}$ mod $q \to \mathcal{R}$ mod q

and breaks the cryptosystem. The linear map θ needs to have some specific properties for Coppersmith's attack to work. This map should be the identity when restricted to $\mathcal{R}_0 \pmod q$. Also, it should map $\mathcal{R}_1 \pmod q$ to itself. Moreover, $w = \theta(h)$, where h is the public key, should be a factor of p such that w/p has small coefficients mod q.

Naturally, one may think that the same idea could be applied to break the proposed scheme for QTRU. We now discuss why this attack will not work for QTRU. First and foremost, the underlying algebraic structure of QTRU is different from that of Non-commutative NTRU. On the other hand, if we want to use the idea of finding a linear map $\theta : \left(\frac{-1,-1}{\mathbb{Z}_q}\right) \to \left(\frac{-1,-1}{\mathbb{Z}_q}\right)$ with such properties, we would have to deal with a lattice of rank $4N$. In particular, when using a lattice-reduction algorithm, such as LLL, Coppersmith's attack will have the same complexity of the finding SVP in the QTRU lattice, which discussed in the previous part.

8 Conclusions

In this paper, we proved that using non-commutativity in a lattice-based cryptosystem is not only possible, but also if we design a noncommutative public-key cryptosystem similar to NTRU, it will remain both secure and efficient. Moreover, we claimed that this cryptosystem is more secure than NTRU, because its lattice structure does not completely fit into the category of Convolutional Modular Lattices. In addition, we introduced a public-key cryptosystem based on the non-commutative quaternion algebras.

We discussed the probability of successful decryption, message and key security, and message expansion ratio in the proposed cryptosystem. In addition, we compared the results to the NTRU, and a group of typical parameters for the proposed cryptosystem were introduced.

Although the proposed method seems to be four times slower than NTRU (under the same conditions, i.e., choosing the same parameters for both NTRU and QTRU cryptosystems), QTRU is much more resistant to lattice-based attacks when compared to NTRU. Hence, one can easily compensate for the speed loss by reducing the dimension and still obtain the same level of security. In addition, by using parallel algorithms, QTRU can be modified to a better one.

9 Further Work

This work was inspired by a desire to construct a Hadamard matrix of order 4×167 based on Williamson's construction method and Turyn's Sequences with small correlation. Over and above the discussion on cryptography, quaternionic lattice theory has valuable usages in coding theory, space-time coding, and quantum physics. Therefore, studying the nature of quaternionic lattices is of interest in continuation of this line of research. Furthermore, since NTRU and QTRU are based on a common concept that does not depend on a certain underlying algebra, this concept can be generalized to different types of rings, modules, and vector spaces, or different kinds of algebras in order to design new lattice-based cryptosystems and explore their possible advantages.

Acknowledgements

The authors of this paper would like to thank the Iran Telecommunications Research Center for its support of this project. Furthermore, we are sincerely grateful for the generous guidance of Professor Hossein Hajiabolhassan, Faculty of Mathematics at Shahid Beheshti University. We thank the anonymous reviewers for their valuable comments and suggestions.

References

[1] Neal Koblitz and Alfred J. Menezes. A Survey of Public-Key Cryptosystems. *SIAM Review*, 46 (4):599–634, 2004.

[2] Neal Koblitz and Alfred Menezes. The Brave New World of Bodacious Assumptions in Cryptography. *Notices of the American Mathematical Society*, 57(3):357–365, 2010.

[3] Miklós Ajtai. The Shortest Vector Problem in L_2 Is NP-Hard for Randomized Reductions. In *Proceedings of the 30th Annual ACM Symposium on Theory of Computing* (*STOC'98*), pages 10–19, 1998.

[4] Daniele Micciancio. The Shortest Vector Problem Is NP-Hard to Approximate to within Some Constant. *SIAM Journal on Computing*, 30(6): 2008–2035, 2001. Preliminary version in FOCS 1998.

[5] Daniele Micciancio and Shafi Goldwasser. *Complexity of Lattice Problems: A Cryptographic Perspective*, volume 671 of *The Kluwer International Series in Engineering and Computer Science*. Kluwer Academic Publishers, Boston, MA, USA, 2002.

[6] Phong Q. Nguyen and Jacques Stern. The Two Faces of Lattices in Cryptology. In *Revised Papers from the International Conference on Cryptography and Lattices* (*CaLC'01*), pages 146–180, 2001.

[7] Jeffrey Hoffstein, Jill Pipher, and Joseph H. Silverman. NTRU: A Ring-Based Public Key Cryp-

tosystem. In *Proceedings of the 3rd International Symposium on Algorithmic Number Theory* (*ANTS-III*), pages 267–288, 1998.

[8] Jeffrey Hoffstein and Joseph Silverman. Optimizations for NTRU. Technical Report 015, NTRU Cryptosystems, 2000. Available at http://www.sisecure.com/cryptolab/pdf/TECH_ARTICLE_OPT.pdf.

[9] Philip S. Hirschhorn, Jeffrey Hoffstein, Nick Howgrave-Graham, and William Whyte. Choosing NTRUEncrypt Parameters in Light of Combined Lattice Reduction and MITM Approaches. In *Proceedings of the 7th International Conference on Applied Cryptography and Network Security* (*ACNS'09*), pages 437–455, 2009.

[10] Petros Mol and Moti Yung. Recovering NTRU Secret Key from Inversion Oracles. In *Proceedings of the 11th International Conference on Public Key Cryptography* (*PKC'08*), pages 18–36, 2008.

[11] *Standard Specifications for Public-Key Cryptographic Techniques Based on Hard Problems over Lattices*. IEEE P1363, 2008. Available at http://grouper.ieee.org/groups/1363/.

[12] Daniel V. Bailey, Daniel Coffin, Adam Elbirt, Joseph H. Silverman, and Adam D. Woodbury. NTRU in constrained devices. In *Proceedings of the 3rd International Workshop on Cryptographic Hardware and Embedded Systems* (*CHES'01*), pages 262–272, 2001.

[13] Colleen O'Rourke and Berk Sunar. Achieving NTRU with Montgomery Multiplication. *IEEE Transactions on Computers*, 52:440–448, 2003.

[14] Philippe Gaborit, Julien Ohler, and Patrick Solé. CTRU, a polynomial analogue of NTRU. Technical report, INRIA, France, 2002. Available at ftp://ftp.inria.fr/INRIA/publication/publi-pdf/RR/RR-4621.pdf.

[15] Michael Coglianese and Bok-Min Goi. MaTRU: A New NTRU-Based Cryptosystem. In *Proceedings of the 6th International Conference on Cryptology in India* (*INDOCRYPT*), pages 232–243, 2005.

[16] Nitin Vats. NNRU, a Noncommutative Analogue of NTRU. *The Computing Research Repository* (*CoRR*), abs/0902.1891, 2009. Available at http://arxiv.org/abs/0902.1891.

[17] R. Kouzmenko. Generalizations of the NTRU Cryptosystem. Master's thesis, Polytechnique Montreal, Canada, 2006.

[18] Camelia Karimianpour. Lattice-Based Cryptosystems. Master's thesis, University of Ottawa, Canada, 2007.

[19] Monica Nevins, Camelia Karimianpour, and Ali Miri. NTRU over rings beyond $\mathbb{Z}$. *Designs, Codes and Cryptography*, 56(1):65–78, 2010.

[20] Jeffrey Hoffstein, Jill Pipher, and J.H. Silverman. *An Introduction to Mathematical Cryptography*. Springer-Verlag, 1st edition, 2008.

[21] Jill Pipher. Lectures on the NTRU Encryption Algorithm and Digital Signature Scheme. Technical report, Brown University, Providence, Rhode Island, 2005. Available at http://www.math.brown.edu/~jpipher/grenoble.pdf.

[22] Alexander May and Joseph H. Silverman. Dimension Reduction Methods for Convolution Modular Lattices. In *Revised Papers from the International Conference on Cryptography and Lattices* (*CaLC'01*), pages 110–125, 2001.

[23] Don Coppersmith and Adi Shamir. Lattice Attacks on NTRU. In *International Conference on the Theory and Application of Cryptographic Techniques* (*EUROCRYPT*), pages 52–61, 1997.

[24] Nick Howgrave-Graham, Jeff Hoffstein, Jill Pipher, and William Whyte. On Estimating the Lattice Security of NTRU. Technical report, Brown University, Providence, Rhode Island, 2005. Available at http://www.math.brown.edu/~jpipher/NTRULattice-2005-1.pdf.

[25] Alexander May. Cryptanalysis of NTRU, 1999. Available at http://www.informatik.uni-frankfurt.de/~alex/ntru.ps.

[26] Joseph H. Silverman. Dimension-Reduced Lattices, Zero-Forced Lattices, and the NTRU Public Key Cryptosystem. Technical report, Security Innovation Inc., Boston, MA, USA, 1999. Available at http://securityinnovation.com/cryptolab/pdf/NTRUTech013.pdf.

[27] Nicolas Gama and Phong Q. Nguyen. Predicting Lattice Reduction. In *International Conference on the Theory and Application of Cryptographic Techniques* (*EUROCRYPT*), pages 31–51, 2008.

[28] A. K. Lenstra, H. W. Lenstra, and L. Lovász. Factoring Polynomials with Rational Coefficients. *Mathematische Annalen*, 261(4):515–534, 1982.

[29] Nicolas Gama, Nick Howgrave-Graham, and Phong Q. Nguyen. Symplectic Lattice Reduction and NTRU. In *International Conference on the Theory and Application of Cryptographic Techniques* (*EUROCRYPT*), pages 233–253, 2006.

[30] Tommi Meskanen. *On the NTRU Cryptosystem*. PhD thesis, University of Turku, Finland, 2005.

[31] John H. Conway and Derek A. Smith. *On Quaternions and Octonions: Their Geometry, Arithmetic, and Symmetry*. A. K. Peters, Ltd., 2003.

[32] David Lewis. Quaternion Algebras and the Algebraic Legacy of Hamilton. *Irish Mathematical Society Bulletin*, 57:41–64, 2006.

[33] Joseph J. Rotman. *Advanced Modern Algebra*. Prentice Hall, 2002.

[34] Zi Yang Sham. Quaternion Algebras and Quadratic Forms. Master's thesis, University of Waterloo, Ontario, Canada, 2008.

[35] Alfred J. Menezes, Paul C. van Oorschot, and

Scott A. Vanstone. *Handbook of Applied Cryptography*. CRC Press, 1996.

[36] Nick Howgrave-Graham, Joseph H. Silverman, and William Whyte. A Meet-In-The-Middle Attack on an NTRU Private Key. Technical report, Security Innovation Inc., Boston, MA, USA, 2002. Available at http://securityinnovation.com/cryptolab/pdf/NTRUTech004v2.pdf.

[37] Eliane Jaulmes and Antoine Joux. A Chosen-Ciphertext Attack against NTRU. In *Proceedings of the 20th Annual International Cryptology Conference on Advances in Cryptology* (*CRYPTO'00*), pages 20–36, 2000.

[38] Phong Q. Nguyen and Damien Stehlé. LLL on the Average. In *Proceedings of the 7th International Symposium on Algorithmic Number Theory* (*ANTS-VII*)., pages 238–256, 2006.

[39] Phong Q. Nguyen and Damien Stehlé. Low-Dimensional Lattice Basis Reduction Revisited. *ACM Transactions on Algorithms*, 5(4):1–48, 2009.

[40] Michael Schneider, Johannes Buchmann, and Richard Lindner. Probabilistic Analysis of LLL Reduced Bases. In *Western European Workshop on Research in Cryptology* (*WEWoRC*), volume 6429 of *LNCS*. Springer, 2009.

[41] Nick Howgrave-Graham, Jeff Hoffstein, Jill Pipher, and William Whyte. On Estimating the Lattice Security of NTRU. Cryptology ePrint Archive, Report 2005/104, 2005. Available at http://eprint.iacr.org/2005/104.

[42] Jeffrey Hoffstein and Joseph H. Silverman. A Non-Commutative Version of the NTRU Public Key Cryptosystem, 1997. Unpublished Paper.

[43] Kathryn Rendall Truman. *Analysis and Extension of Non-Commutative NTRU*. PhD thesis, University of Maryland, MD, USA, 2007.

[44] Don Coppersmith. Attacking Non-Commutative NTRU. Technical report, IBM, 1997.

Ehsan Malekian received the BSc degree in Computer Engineering from Isfahan University of Technology, Isfahan, Iran in 1993 and MSc degree from Shiraz University, Shiraz, Iran in 1995. He is now completing his PhD at the Shahid Beheshti University, Tehran, Iran. His research interests include Data Security, Computer Networks, Lattice-based Cryptography and Arithmetic Architectures for public-key cryptosystems.

Ali Zakerolhosseini received the BSc degree from the University of Coventry, UK, in 1985, MSc from the Bradford University, UK, in 1987, and PhD degree in Fast transforms from the University of Kent, UK, in 1998. He is currently been an associate professor in the Department of Electrical and Computer Engineering at Shahid Beheshti University, Tehran, Iran. His research focuses on Reconfigurable devices and Multi classifiers. His current research interests are data security, reconfigurable devices and parallel processing.

Atefeh Mashatan received a B. Math from Carleton University, Ottawa, Canada, in 2003. Subsequently, she obtained her M. Math in Combinatorics and Optimization, followed by her PhD (December 2008) in Cryptography from the University of Waterloo, Waterloo Canada. Her thesis is entitled "Message Authentication and Recognition Protocols Using Two-Channel Cryptography." She is now conducting research at the Swiss Federal Institute of Technology, Lausanne (EPFL), Switzerland.