A Public-Key Encryption Scheme Based on Non-linear Indeterminate Equations

Koichiro Akiyama[1(✉)], Yasuhiro Goto[2], Shinya Okumura[3], Tsuyoshi Takagi[4], Koji Nuida[5], and Goichiro Hanaoka[5]

[1] Corporate Research and Development Center, Toshiba Corporation, Kawasaki, Japan
koichiro.akiyama@toshiba.co.jp
[2] Department of Mathematics, Hokkaido University of Education, Hakodate, Japan
goto.yasuhiro@h.hokkyodai.ac.jp
[3] Department of Information and Communications Technology, Osaka University, Suita, Japan
okumura@cy2sec.comm.eng.osaka-u.ac.jp
[4] Institute of Mathematics for Industry, Kyushu University, Fukuoka, Japan
takagi@imi.kyushu-u.ac.jp
[5] National Institute of Advanced Industrial Science and Technology, Tokyo, Japan
{k.nuida,hanaoka-goichiro}@aist.go.jp

Abstract. In this paper, we propose a post-quantum public-key encryption scheme whose security depends on a problem arising from a multivariate non-linear indeterminate equation. The security of lattice cryptosystems, which are considered to be the most promising candidate for a post-quantum cryptosystem, is based on the shortest vector problem or the closest vector problem in the discrete linear solution spaces of simultaneous equations. However, several improved attacks for the underlying problems have recently been developed by using approximation methods, which result in requiring longer key sizes. As a scheme to avoid such attacks, we propose a public-key encryption scheme based on the "smallest" solution problem in the *non-linear* solution spaces of multivariate *indeterminate equations* that was developed from the algebraic surface cryptosystem. Since no efficient algorithm to find such a smallest solution is currently known, we introduce a new computational assumption under which proposed scheme is proven to be secure in the sense of IND-CPA. Then, we perform computational experiments based on known attack methods and evaluate that the key size of our scheme is able to be much shorter than those of previous lattice cryptosystems.

Keywords: Public-key cryptosystem · Post-quantum cryptosystem
Indeterminate equation · Smallest solution problem

S. Okumura—Research conducted while at Institute of Mathematics for Industry, Kyushu University.

C. Adams and J. Camenisch (Eds.): SAC 2017, LNCS 10719, pp. 215–234, 2018.
https://doi.org/10.1007/978-3-319-72565-9_11

1 Introduction

In 1994, Shor proposed quantum algorithms that can solve the factorization problem and the discrete logarithm problem in polynomial time [30]. This implies that elliptic curve cryptosystems and the RSA cryptosystem will no longer be secure once a quantum computer is built. Due to this, the importance of "Post-quantum cryptosystems" (PQCs) that will still be secure after the development of quantum computers has been recognized. With the recent active studies to develop quantum computers, NIST announced that the process of PQC standardization will begin in the end of 2017 [25]. Possible candidates for a PQC include lattice-based encryptions, code-based encryptions, and multivariate encryptions.

First lattice-based encryption was proposed in 1997 by Ajtai and Dwork [1]. Its security depends on the unique shortest vector problem in lattices. Goldreich et al. proposed the GGH cryptosystem, whose security is based on the closest vector problem for an integer lattice [14]. However, According to Nguyen and Stern, these schemes are not practical since they require large size parameters for security reasons [23,24]. Hoffstein et al. proposed the NTRU cryptosystem, whose security depends on the shortest vector problem for polynomial ring lattices [15]. In 2009, Regev proposed an LWE cryptosystem, whose security depends on the "learning with error" (LWE) problem [28]. Currently, NTRU, LWE, and their variants are relatively efficient among lattice-based encryption schemes.

However, there are several efficient approximation algorithms for finding the (nearly) shortest/closest vectors, such as the LLL [19], BKZ [29], and BKZ2.0 [8] algorithms. Recently, several improved attacks for these underlying problems using these methods, such as lattice decoding attacks [6] and subfield lattice attacks [18] have been developed. In order to avoid these attacks, the public-key sizes of lattice-based cryptosystems must be enlarged. Encryption schemes with large key sizes require a large amount of memory in applications.

Code-based encryption was first proposed in 1978 by McEliece [22]. Its security depends on the decoding problem for random linear codes, for which only exponential algorithms are known. However, it requires a large public-key size, of more than 1M bits. The multivariate public-key cryptosystem (MPKC) was first introduced in 1989 by Matsumoto and Imai [16] and was improved by Patarin [26]. Its security depends on the problem of solving non-linear equations (called multivariate equations) over finite fields. While the problem is NP-hard in general, almost all proposed schemes have been broken due to the special structure of the equations that are used as public keys. Several schemes with resistance against known attacks on MPKC have been proposed, but they still have large public keys [27,32,33].

These candidates require large public-key sizes of more than 24 K bits (under 128-bit security) to avoid improved attacks that take advantage of the special structure of the schemes. Even though many PQC candidates have been proposed, none of them are efficient enough for practical use. This might be due to their large public-key sizes and the large amount of memory that is therefore required in applications. In an effort to find a more practical PQC, Akiyama et al.

proposed the algebraic surface cryptosystem (ASC) [3], whose security depends on the section-finding problem (the problem of solving some kind of indeterminate equation). Although they claimed that their proposed scheme necessitates much shorter public keys than the other candidates for PQC, the scheme was broken by Faugére et al. [11]. In this paper, we intend to improve ASC by modifying the underlying problem to make the scheme secure while keeping the public-key size small relative to that of other PQC candidates.

Our Contribution. This paper proposes a post-quantum public-key encryption scheme whose security is based on the smallest solution problem for non-linear solution spaces of indeterminate equations, to which attack algorithms based on approximation (e.g., LLL and BKZ) cannot be applied. Our scheme was developed from ASC, which is designed such that its security depends on the intractability of solving some non-linear indeterminate equation [3]. ASC was broken by the ideal decomposition attack proposed in PKC 2010 [11]. We revise the scheme to be secure against this attack by adding a noise term to the cipher polynomial. Our scheme is provably secure in regards to IND-CPA under the intermediate equation of LWE (IE-LWE) assumption, which is a new computational assumption coming from analogy to the LWE assumption. An IND-CCA2 secure scheme is obtained by using a well-known conversion technique [10].

The linear algebraic attack, one of the known attacks for ASC, can be applied to the IE-LWE problem. Through this attack, the IE-LWE problem can be reduced to a lattice problem, but the rank of the lattice is larger than that of present lattice-based cryptosystems due to the properties of multivariate polynomials. This suggests that the keys (both public and secret) can be expected to be much shorter than those of lattice-based cryptosystems. Our scheme is, in this sense, a light PQC constructed by combining the beneficial properties of multivariate cryptography and lattice-based cryptography. According to our computational experiment on attacks, our scheme requires a public key that is 3/4 the length of the public keys in LWE and 1/3 the length of the public keys in NTRU. Moreover, our scheme supports multi-bit homomorphism as well as NTRU.

This paper is organized as follows. Section 2 gives our notation and a short overview of algebraic surface encryptions, which our scheme was developed from. In Sect. 3, we define the smallest solution problem and propose our new encryption scheme. Section 4 defines the computational assumption that makes our scheme provably secure and discusses the complexity of this assumption against some considered attacks. In Sect. 5, we give a set of appropriate parameters that make our scheme secure. We summarize the results and discuss directions for future work in Sect. 6.

2 Preliminaries

2.1 Notation

We express a polynomial with two variables x, y as $\xi(x, y) = \sum_{(i,j)\in\Gamma_\xi} \tau_{i,j} x^i y^j$, where Γ_ξ denotes the set of pairs (i, j) of the exponents of non-zero monomials

$x^i y^j$ in a polynomial $\xi(x, y)$. We refer to Γ_ξ as the **term set** of $\xi(x, y)$. Note that the cardinality $\#\Gamma_\xi$ is equal to the number of monomials in $\xi(x, y)$. Hereinafter, we write ξ instead of $\xi(x, y)$ when ξ is clearly a polynomial in two variables x, y.

The set of polynomials with two variables having the term set Γ over a ring R is denoted by $\mathfrak{F}_\Gamma/R$. This is defined as

$$\mathfrak{F}_\Gamma/R = \left\{ f \in R[x, y] \quad | \quad f = \sum_{(i,j)\in\Gamma} a_{ij} x^i y^j \right\}.$$

For simplicity, we write $\mathfrak{F}_\Gamma$ instead of $\mathfrak{F}_\Gamma/R$ when it is clearly over R.

In this paper, we take representative sets of $\mathbb{Z}_p$ and $\mathbb{Z}_q$ as $\mathbb{Z}_p^+ = \{0, 1, \cdots, p-1\}$ and $\mathbb{Z}_q^+ = \{0, 1, \cdots, q-1\}$, respectively. we refer to $\mathbb{Z}_q[t]/(t^n - 1)$ as R_q and denote the subset of R_q whose elements have restricted coefficients to the range of $\mathbb{Z}_p^+$ to R_p. Then, we can define the maximum coefficient of the polynomial ξ, which is denoted by $MC(\xi)$, as follows:

$$MC(\xi) = \max \left\{ \tau_{i,j} | \xi(x, y) = \sum_{(i,j)\in\Gamma_\xi} \tau_{i,j} x^i y^j \right\}, \tag{1}$$

where $\tau_{i,j}$ is regarded as an integer instead of a representative element in $\mathbb{Z}_p$ or $\mathbb{Z}_q$ to measure the size of the coefficients. Some properties of the maximum coefficient are described in Appendix B.

These concepts can be defined in the same manner for polynomials with one or three variables.

2.2 Algebraic Surface Cryptosystem

ASC was first introduced in 2006 by Akiyama and Goto [2]. The security of ASC depends on the section-finding problem, defined as follows.

Definition 1 (Section-finding Problem). If $X(x, y, t) = 0$ is an algebraic surface over field K, then the problem of finding a parameterized curve $(x, y, t) = (u_x(t), u_y(t), t)$ on X is called the *section-finding problem* on X.

A section can be considered as a solution of $X(x, y) = 0$, which is an indeterminate equation over the ring $K[t]$. In this paper, we write an algebraic surface $X(x, y) = 0$ over $F_p[t]$ instead of $X(x, y, t) = 0$ over F_p.

The problem of solving indeterminate equations over some rings or fields is known to be difficult. For example, the case of indeterminate equations over the integer ring $\mathbb{Z}$, a class of problems called Diophantine equations, is undecidable (Hilbert's 10th problem). "Undecidable" in this context means that there is no general algorithm to solve such indeterminate equations. The section-finding problem has also been proven to be undecidable [9].

To show the concept for the scheme we propose in this paper, we give an explanation of algebraic surface encryption. First, the simplest ASC can be described as

$$c(x, y) = m(x, y) + X(x, y)r(x, y), \tag{2}$$

where $X(x, y)$ is the public key, which defines an algebraic surface with a section. The polynomials $c(x, y)$ and $r(x, y)$ are a ciphertext polynomial and a random polynomial, respectively. The polynomial $m(x, y)$ is a plaintext polynomial in which plaintext is embedded. In the decryption phase, we substitute the secret key (a section of $X(x, y)$) into $c(x, y)$. Using the relation $X(u_x(t), u_y(t)) = 0$, we obtain $c(u_x(t), u_y(t)) = m(u_x(t), u_y(t))$. The plaintext can be recovered from the polynomial $m(u_x(t), u_y(t))$ as follows. First, we write $m(x, y)$ as $m(x, y) = \sum_{(i,j,k)\in\Gamma_m} m_{ijk}x^i y^j t^k$, where m_{ijk} are unknowns, and substitute the section into $m(x, y)$. Then, we obtain $m(u_x(t), u_y(t)) = \sum_{(i,j,k)\in\Gamma_m} m_{ijk}u_x(t)^i u_y(t)^j t^k$. The simultaneous linear equations in m_{ijk} are constructed by comparing the coefficients of t. When the number of variables is less than or equal to the rank of the coefficient matrix, we can recover the correct plaintext by solving the equations.

However, an attack that can break the scheme exists. We can expand the cipher polynomial $c(x, y)$ as

$$c(x,y) = \sum_{(i,j,k)\in\Gamma_m} m_{ijk}x^i y^j t^k + \left(\sum_{(i,j,k)\in\Gamma_X} a_{ijk}x^i y^j t^k\right)\left(\sum_{(i,j,k)\in\Gamma_r} r_{ijk}x^i y^j t^k\right), \quad (3)$$

where $\Gamma_m, \Gamma_X, and\ \Gamma_r$ are given as parameters and a_{ijk} are given coefficients of the public key X; and m_{ijk} and r_{ijk} are variables. By comparing the coefficients of the monomials, we obtain the simultaneous linear equations with the variables m_{ijk} and r_{ijk}. The relation $\#\Gamma_m + \#\Gamma_r < \#\Gamma_{Xr}$ is required for the decoding. However, in this case, the equations have unique solutions with high probability. We refer to the attacks of this type as **linear algebraic attacks**.

For avoiding this attack, Akiyama, Goto, and Miyake constructed the latest ASC scheme in 2009 [3]. From the cryptographic point of view, the ciphertext is equivalent to

$$c(x, y) = m(x, y)s(x, y) + X(x, y)r(x, y). \quad (4)$$

Here, $s(x, y)$ is employed as another random polynomial, and the term set $m(x, y)s(x, y)$ is equal to that of $X(x, y)r(x, y)$ ($\Gamma_{ms} = \Gamma_{Xr}$). In order to decrypt the ciphertext, we have to decompose $m(u_x(t), u_y(t))s(u_x(t), u_y(t))$ into $m(u_x(t), u_y(t))$ and $s(u_x(t), u_y(t))$. Since polynomial factorization (over F_p) is easy to compute by using the Berlekamp method, we can obtain $m(u_x(t), u_y(t))$ as a factor, and recover the plaintext from $m(u_x(t), u_y(t))$ in the same way as the previous scheme.

When applying the linear algebra attack to this scheme, $m(x, y)s(x, y)$ must be considered as a single polynomial $g(x, y)$ because the quadratic equations are derived from the variables m_{ijk} and s_{ijk}. (It is difficult to solve systems of quadratic equations in general.) Therefore, if the number of variables $\#\Gamma_r + \#\Gamma_{Xr}$ is greater than the number of equations $\#\Gamma_{Xr}$, then the linear algebra attack does not work.

Unfortunately, this scheme was also broken by the **ideal decomposition attack**, which was introduced by Faugere et al. [11]. They found that the ideal

(c, X) can be decomposed into (m, X) and (s, X) by calculating the resultant $Res_x(c, X)$ or $Res_y(c, X)$. Ultimately, they were able to recover the plaintext m by using this method to solve the linear equations.

3 Our Proposed Encryption Scheme

In this section, we propose a new ASC scheme that is resistant to the ideal decomposition attack. We accomplish this by changing the underlying ring of ASC to $\mathbb{Z}_q[t]/(t^n - 1)$ and adding a p divisible polynomial $p \cdot e(x, y)$ to the simplest ASC cipher polynomial (2) as noise. Our cipher polynomial is

$$c(x, y) = m(t) + X(x, y)r(x, y) + p \cdot e(x, y),$$

where $e(x, y)$ is a random polynomial with small coefficients, and p and m are a small prime and an element of $\mathbb{Z}_q[t]/(t^n - 1)$, respectively. The polynomial $e(x, y)$ works as a noise factor in the cipher, and the condition $\#\Gamma_e = \#\Gamma_{Xr}$ is required for resistance against the linear algebra attack. Also, a small solution of $X(x, y)$ is necessary in order to decrypt.

3.1 Algorithms

Parameters. In this section, we introduce our scheme's parameters. Appropriate parameters are discussed in Sect. 5. The parameters are as follows.

1. p, q: The cardinality of $\mathbb{Z}_p, \mathbb{Z}_q$, where p, q are primes and $p \ll q$
2. n: The degree of the modulus polynomial of $R_q (= \mathbb{Z}_q[t]/(t^n - 1))$
3. Γ_X: The term set of the indeterminate equation $X(x, y) (= 0)$
4. Γ_r: The term set of the random polynomial $r(x, y)$

The total degrees of X and r are denoted by w_X and w_r, respectively. The relation between p and q is important to the decryption. The following condition must be fulfilled:

$$q > \#\Gamma_{Xr} \cdot p(p-1) \cdot (n(p-1))^{w_X + w_r}, \tag{5}$$

which reason is explained in Appendix B. It is evident that q is much greater than p.

Keys. The secret-key is a small (smallest is not necessary) solution of the indeterminate equation $X(x, y) = 0$, which is denoted by u:

$$u : (x, y) = (u_x(t), u_y(t)), \quad u_x(t), u_y(t) \in R_p, \tag{6}$$

where $\deg u_x(t) = \deg u_y(t) = n - 1$. Note that p is much smaller than q. Therefore, we call u a *small solution*. The public key is the indeterminate equation $X(x, y) = 0$ that has the smallest solution u:

$$X(x, y) = \sum_{(i,j) \in \Gamma_X} a_{ij} x^i y^j, \tag{7}$$

where $a_{ij} \in R_q$.

Key Generation. The key-generation algorithm, which accepts parameters $p, q, n, \Gamma_X, and \Gamma_r$ as input, can be described as follows. The secret key is generated as the random polynomials $u_x(t), u_y(t) (\in R_p)$, whose degrees are $n-1$. The indeterminate equation $X(x,y)=0$ is constructed according to the following procedure.

1. Choose a coefficient for each non-constant monomial as follows.
   (a) Set $X=0$.
   (b) For each (i,j) in Γ_X:
      i. Choose a coefficient $a_{ij}(t)$, with degree $n-1$ uniformly at random from the set R_q.
      ii. Set $X = X + a_{ij}(t)x^i y^j$.
2. Calculate the constant term $a_{00}(t)$ as
   $a_{00}(t) = -\sum_{(i,j)\in\Gamma_X-(0,0)} a_{ij}(t)u_x(t)^i u_y(t)^j \ (\in R_q)$.

Encryption

1. Embed a plaintext M into the coefficients of the plaintext polynomial $m(t) (\in R_p)$, whose degree is $n-1$.
2. Choose a random polynomial $r(x,y)$ in $\mathfrak{F}_{\Gamma_r}/R_q$ as follows.
   (a) Set $r=0$.
   (b) For each (i,j) in Γ_r:
      i. Choose a coefficient $r_{ij}(t)$, with degree $n-1$ uniformly at random from the set R_q.
      ii. Set $r = r + r_{ij}(t)x^i y^j$.
3. Choose a noise polynomial $e(x,y)$ for $\mathfrak{F}_{\Gamma_{Xr}}/R_p$ as follows.
   (a) Set $e=0$
   (b) For each (i,j) in Γ_{Xr}:
      i. Choose a coefficient $e_{ij}(t)$, with degree $n-1$ uniformly at random from the set R_p.
      ii. Set $e = e + e_{ij}(t)x^i y^j$.
4. Construct the cipher polynomial $c(x,y)$ as

$$c(x,y) = m(t) + X(x,y)r(x,y) + p \cdot e(x,y). \tag{8}$$

Decryption

1. Substitute the smallest solution u into $c(x,y)$ as a solution of X over $F_q[t]$:

$$c(u) = m(t) + p \cdot e(u), \tag{9}$$

   where $c(u)$ denotes $c(u_x(t), u_y(t))$. When the parameters p and q satisfy the relation described above (5), each coefficient of $m(t) + p \cdot e(u) \in \mathbb{Z}/(t^n - 1)$ is within the range of $\mathbb{Z}_q^+$. The proof for this is given in Appendix B
2. Extract $m(t)$ from $c(u)$ as $c(u) \ (mod\ p) = m(t)$, where we consider $c(u)$ as an element of $\mathbb{Z}[t]$.
3. Recover the plaintext M from the coefficients of $m(t)$.

From now on, we will refer to the public-key encryption scheme as the indeterminate equation cryptosystem (**IEC**) **encryption scheme**.

3.2 The Smallest-Solution Problem

Let us express the solution $u = (u_x(t), u_y(t))$ $(\in (\mathbb{Z}_q[t]/(t^n-1))^2)$ of an indeterminate equation as

$$u_x(t) = \sum_{i=0}^{n-1} \alpha_i t^i, \quad u_y(t) = \sum_{i=0}^{n-1} \beta_i t^i.$$

Then, the norm of the solution is defined as follows.

$$Norm(u) = \max\{\alpha_i, \beta_i \in \mathbb{Z}_q^+ \mid 0 \leq i \leq n-1\}$$

The security of our system depends on the smallest-solution problem, defined as follows.

Definition 2 (Smallest-solution Problem). If $X(x, y) = 0$ is an indeterminate equation over the ring $\mathbb{Z}_q[t]/(t^n - 1)$, then the problem of finding the solution $(x, y) = (u_x(t), u_y(t))$ on $\mathbb{Z}_q[t]/(t^n - 1)$ with the smallest norm is called the *smallest-solution problem* on X.

We are not able to apply the approximate lattice reduction algorithms directly to solving the problem because the solution space is non-linear.

4 Security

In this section, we introduce a computational assumption and discuss some possible attacks for the assumption, based on the attacks for ASCs.

4.1 Security Assumption

The polynomials over $\mathbb{Z}_q$ whose coefficients are in the range of 0 to $p-1$ are called size-p polynomials. If a polynomial is size p, this means that its coefficients are much smaller than those of an ordinary polynomial, since p is much smaller than q. We define the set of polynomials that have zero points in size p as follows:

$$\mathfrak{X}(\Gamma_X, p)/R_q = \{X \in \mathfrak{F}_{\Gamma_X}/R_q \mid \exists u_x(t), u_y(t) \in R_p \;\; X(u_x(t), u_y(t)) = 0\}.$$

When the sets of polynomials, such as $\mathfrak{X}(\Gamma_X, p)/R_q$, $\mathfrak{F}_{\Gamma_r}/R_q$, and $\mathfrak{F}_{\Gamma_{Xr}}/R_p$, that satisfy the condition

$$(0,0) \in \Gamma_X, (0,0) \in \Gamma_r$$

are given, we define the decisional problem as follows.

Definition 3 (IE-LWE problem). *When we write the set U_X, T_X as*

$$U_X = \mathfrak{X}(\Gamma_X, p)/R_q \times \mathfrak{F}_{\Gamma_{Xr}}/R_q, \tag{10}$$

$$T_X = \{(X, Xr + e) | X \in \mathfrak{X}(\Gamma_X, p)/R_q, r \in \mathfrak{F}_{\Gamma_r}/R_q, e \in \mathfrak{F}_{\Gamma_{Xr}}/R_p\}, \tag{11}$$

respectively, the IE-LWE problem is to distinguish the multivariate polynomials chosen from a 'noisy' set T_X of polynomials or from a set of $U_X - T_X$, where T_X is a subset of U_X.

We define the IE-LWE assumption.

Definition 4 (IE-LWE assumption). *The IE-LWE assumption is the assumption that the advantage*

$$Adv_{\mathfrak{B}}^{IE\text{-}LWE}(k) :=$$

$$\left| Pr\left[\mathfrak{B}(p,q,n,\Gamma_r,\Gamma_X,X,Y) \to 1 \;\middle|\; \begin{array}{l}(p,q,n,\Gamma_X,\Gamma_r,X) \xleftarrow{R} GenG(1^k);\\ r \xleftarrow{U} \mathfrak{F}_{\Gamma_r}/R_q; e \xleftarrow{U} \mathfrak{F}_{\Gamma_{Xr}}/R_p;\\ Y := Xr + e\end{array}\right] \right. \tag{12}$$
$$\left. -Pr\left[\mathfrak{B}(p,q,n,\Gamma_r,\Gamma_X,X,Y) \to 1 \;\middle|\; \begin{array}{l}(p,q,n,\Gamma_X,\Gamma_r,X) \xleftarrow{R} GenG(1^k);\\ \\ Y \xleftarrow{U} \mathfrak{F}_{\Gamma_{Xr}}/R_q\end{array}\right] \right|$$

is negligible. In other words,

$$Adv_{\mathfrak{B}}^{IE\text{-}LWE}(k) < \epsilon(k),$$

where $\epsilon(k)$ is a negligible function in the security parameter k.

IE-LWE is an extended variation of R-LWE$^{\times}_{\mathrm{HNF}}$, which is one of the variants of R-LWE defined by the polynomial ring R_q. This is claimed by a provably secure NTRU modification [31] and can be reduced to the shortest vector problem of the lattice derived from R_q. In this paper, we extend R-LWE$^{\times}_{\mathrm{HNF}}$ to the multivariate polynomial ring $R_q[x,y]$ so that the dimension of the lattice is larger than that of the lattice derived from R_q.

Theorem 1. *Under the IE-LWE assumption, the IEC encryption scheme $\Sigma = (Gen, Enc, Dec)$ is secure in the sense of IND-CPA. Specifically, if there is an adversary that runs in polynomial time and breaks the IEC encryption scheme Σ in the sense of IND-CPA, then there exists an algorithm $\mathfrak{B}$ that solves the IE-LWE problem in probabilistic polynomial time. Moreover, the following relation holds:*

$$Adv_{\Sigma,\mathfrak{A}}^{IND\text{-}CPA}(k) = 2 \cdot Adv_{\mathfrak{B}}^{IE\text{-}LWE}(k).$$

Proof. Due to space constraints, we omit the proof. We carried out the proof by using the same technique as in the proof of Lemma 13 in [31].

In addition, one can make the IEC encryption scheme IND-CCA2 secure by using well-known conversions such as those in [10]. However, the converted scheme is no longer a homomorphic one.

4.2 Possible for Attacks

In this subsection, we introduce two possible attacks for the IE-LWE assumption. Other attacks against ASC, which this scheme was developed from, cannot be applied to this problem. For example, the ideal decomposition attack described in Sect. 2.2 does not work on our scheme because our scheme does not have a multiple structure such as $m(x,y)s(x,y)$ in (4).

The Linear Algebra Attack. Given a pair of polynomials (X, Y), we can determine that (X, Y) is sampled from T_X if we find $r \in \mathfrak{F}_{\Gamma_r}/R_q$ and $e \in \mathfrak{F}_{\Gamma_{Xr}}/R_p$ such that $Y = Xr + e$. The problem of finding such polynomials r and e can be solved by using the linear algebra attack introduced in Sect. 2.2 as follows. We construct a system of linear equations by comparing the coefficients of $x^i y^j$ in the relation

$$\sum_{(i,j)\in\Gamma_{Xr}} d_{ij}x^i y^j = \left(\sum_{(i,j)\in\Gamma_X} a_{ij}x^i y^j\right)\left(\sum_{(i,j)\in\Gamma_r} r_{ij}x^i y^j\right) + \left(\sum_{(i,j)\in\Gamma_{Xr}} e_{ij}x^i y^j\right), \tag{13}$$

where r_{ij} and e_{ij} are R_q-valued and R_p-valued variables, respectively.

In the case of $\deg X = \deg r = 1$, we can set X, r, e, and Y in the following manner.

$$\begin{aligned} X(x,y) &= a_{10}x + a_{01}y + a_{00} \\ r(x,y) &= r_{10}x + r_{01}y + r_{00} \\ e(x,y) &= e_{20}x^2 + e_{11}xy + e_{02}y^2 + e_{10}x + e_{01}y + e_{00} \\ Y(x,y) &= d_{20}x^2 + d_{11}xy + d_{02}y^2 + d_{10}x + d_{01}y + d_{00} \end{aligned}$$

From the equation

$$\begin{aligned} X(x,y)r(x,y) = \; & a_{10}r_{10}x^2 + (a_{10}r_{01} + a_{01}r_{10})xy + a_{01}r_{01}y^2 + (a_{10}r_{00} + a_{00}r_{10})x \\ & + (a_{01}r_{00} + a_{00}r_{01})y + a_{00}r_{00}\,, \end{aligned}$$

we obtain a system of linear equations as follows:

$$\begin{aligned} a_{10}r_{10} + e_{20} &= d_{20} \\ a_{10}r_{01} + a_{01}r_{10} + e_{11} &= d_{11} \\ a_{01}r_{01} + e_{02} &= d_{02} \\ a_{10}r_{00} + a_{00}r_{10} + e_{10} &= d_{10} \\ a_{01}r_{00} + a_{00}r_{01} + e_{01} &= d_{01} \\ a_{00}r_{00} + e_{00} &= d_{00}\ . \end{aligned} \tag{14}$$

The system has a solution space with dimension at least three since the number of variables is more than the number of equations by three. In general, a linear system obtained with this attack has a solution space with a dimension at least $\#\Gamma_r$ since the system has $\#\Gamma_{Xr} + \#\Gamma_r$ variables and $\#\Gamma_{Xr}$ equations.

When we can find a solution such that e_{ij} are valued in R_p, we conclude that (X, Y) is in T_X. We may find it exactly with a brute force attack on the polynomial e, but this attack can be avoided by increasing $\#\Gamma_{Xr}$ to

$$((p-1)p^{n-1})^{\#\Gamma_{Xr}} > 2^k\,,$$

where k is a security parameter.

We employ a lattice-reduction attack to find such a small e_{ij}. Let us represent $a \in R_q$ as a vector $(a_0, a_1, \cdots, a_{n-2}, a_{n-1})$ for

$$a = a_0 + a_1 t + \cdots + a_{n-2}t^{n-2} + a_{n-1}t^{n-1}\,.$$

When the elements $b, c \in R_q$ are represented in the same manner as a, we can express $ab + c$ as

$$\begin{pmatrix} a_{n-1} & a_{n-2} & \cdots & a_1 & a_0 \\ a_{n-2} & a_{n-3} & \cdots & a_0 & a_{n-1} \\ a_{n-3} & a_{n-4} & \cdots & a_{n-1} & a_{n-2} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a_0 & a_{n-1} & \cdots & a_2 & a_1 \end{pmatrix} \begin{pmatrix} b_0 \\ b_1 \\ \vdots \\ b_{n-2} \\ b_{n-1} \end{pmatrix} + \begin{pmatrix} c_{n-1} \\ c_{n-2} \\ \vdots \\ c_1 \\ c_0 \end{pmatrix}.$$

The first equation of (14) is described as

$$A_{10}\boldsymbol{r_{10}} + \boldsymbol{e_{20}} = \boldsymbol{d_{20}}$$

when a_{10} is expressed as

$$A_{10} = \begin{pmatrix} a_{n-1} & a_{n-2} & \cdots & a_1 & a_0 \\ a_{n-2} & a_{n-3} & \cdots & a_0 & a_{n-1} \\ a_{n-3} & a_{n-4} & \cdots & a_{n-1} & a_{n-2} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a_0 & a_{n-1} & \cdots & a_2 & a_1 \end{pmatrix}$$

and $\boldsymbol{r_{10}}, \boldsymbol{e_{20}}, \boldsymbol{d_{20}}$ are denoted by

$$\begin{aligned} \boldsymbol{r_{10}} &= \begin{pmatrix} r_0 & r_1 & \cdots & r_{n-2} & r_{n-1} \end{pmatrix}^T, \\ \boldsymbol{e_{20}} &= \begin{pmatrix} e_{n-1} & e_{n-2} & \cdots & e_1 & e_0 \end{pmatrix}^T, \\ \boldsymbol{d_{20}} &= \begin{pmatrix} d_{n-1} & d_{n-2} & \cdots & d_1 & d_0 \end{pmatrix}^T, \end{aligned}$$

respectively. By adding the integer vector $\boldsymbol{u_{20}} = (u_{n-1}, \cdots, u_0)^T$, we obtain the equation over the integers, as follows.

$$A_{10}\boldsymbol{r_{10}} + q\boldsymbol{u_{20}} + \boldsymbol{e_{20}} = \boldsymbol{d_{20}}$$

Now, we can consider an integer lattice $\mathcal{L} = \begin{pmatrix} A_{10} & qI_n \end{pmatrix}$, where I_n denotes the $n \times n$ unit matrix. If we can find a point v closest to the $\boldsymbol{d_{20}}$ in the lattice $\mathcal{L}$, then we can detect $\pm\boldsymbol{e_{20}}$ from $\boldsymbol{v} - \boldsymbol{d_{20}}$ with high possibility. In the same way, $\pm\boldsymbol{e_{11}}$ can be detected from a point w closest to the $\boldsymbol{d_{11}}$ in the lattice $\begin{pmatrix} A_{10} & A_{01} & qI_n \end{pmatrix}$. However, we cannot distinguish whether the sample (X, Y) is sampled from T_X if the a_{ij}'s are invertible in R_q. For the equation $a_{10}r_{10} + e_{20} = d_{20}$, we can calculate $r_{10} \in R_q$ from any short vector e_{20} as $r_{10} = a_{10}^{-1}(d_{20} - e_{20})$. This implies that any sample $(X, Y) \in U_X$ satisfies the relation. This is true for any equation in (14).

Therefore, we need to simultaneously consider all equations in (14). Then, we see that the linear algebraic attack can be reduced to the closest vector problem (CVP) on the lattice

$$\begin{pmatrix} A_{10} & & qI_n & & & & \\ A_{01} & A_{10} & & qI_n & & & \\ & A_{01} & & & qI_n & & \\ A_{00} & & A_{10} & & & qI_n & \\ & A_{00} & A_{01} & & & & qI_n \\ & & A_{00} & & & & \end{pmatrix} \tag{15}$$

and the vector $(\boldsymbol{d_{20}}\, \boldsymbol{d_{11}}\, \boldsymbol{d_{02}}\, \boldsymbol{d_{10}}\, \boldsymbol{d_{01}}\, \boldsymbol{d_{00}})^T$. Here, blank entries are zero matrices.

Key-Recovery Attack. If a solution $\tilde{u} := (\tilde{u}_x(t), \tilde{u}_y(t)) \in R_q^2$ to $X(x, y) = 0$ (not necessarily the secret key) in which all coefficients are less than p is found, then the IE-LWE problem can be solved with high probability, as follows. For an IE-LWE instance (X, Y), if all coefficients of $p \cdot Y(\tilde{u})$ are multiples of p, then it can be concluded that (X, Y) is sampled from T_X. In fact, sampling (X, Y) from T_X implies that

$$p \cdot Y(\tilde{u}) = p(X(\tilde{u})r(\tilde{u}) + e(\tilde{u})) = p \cdot e(\tilde{u}),$$

and $MC(e(\tilde{u})) < q$ implies that all coefficients of $p \cdot e(\tilde{u})$ are multiples of p. On the other hand, if (X, Y) is sampled from U_X, then the probability that all coefficients of $p \cdot Y(\tilde{u})$ are multiples of p is about $1/p^n$. Therefore, if a small solution, such as $\tilde{u}$, can be found, then the IE-LWE problem can be solved with a probability higher than $1 - 1/p^n$ by checking whether all coefficients of $p \cdot Y(\tilde{u})$ are multiples of p. Since $n, p \geq 2$, the probability $1 - 1/p^n$ is at least $3/4$, which is non-negligible.

In the following, we consider the key-recovery attack on our encryption scheme (i.e., finding the smallest solution to $X(x, y) = 0$ over R_q by using lattice-reduction techniques). First, we consider the case of $\deg X = 1$. In this case, we need to find $u_x(t), u_y(t) \in R_p^2$ satisfying

$$a_{10}u_x(t) + a_{01}u_y(t) + a_{00} = 0. \tag{16}$$

We write this equation with a matrix and vectors in the same manner as the algebraic attack described above, as follows:

$$A \begin{pmatrix} \boldsymbol{u_x}\ \boldsymbol{u_y}\ \boldsymbol{u} \end{pmatrix}^T = \begin{pmatrix} -\boldsymbol{a_{00}} \end{pmatrix}, \tag{17}$$

where $\boldsymbol{u}$ is the vector corresponding to $u \in \mathbb{Z}[t]/(t^n - 1)$ and satisfying $a_{10}u_x(t) + a_{01}u_y(t) + qu + a_{00} = 0$ in $\mathbb{Z}[t]/(t^n - 1)$ and $A = \begin{pmatrix} A_{10}\ A_{01}\ qI_n \end{pmatrix}$. We consider the lattice $\mathcal{L}_A = \{\boldsymbol{x} | A\boldsymbol{x} = \boldsymbol{0}\}$ and let $\boldsymbol{v}$ be a solution to the system (17). Then, any solution of (17) can be written as $\boldsymbol{v} + \boldsymbol{w}$ ($\boldsymbol{w} \in \mathcal{L}_A$). Observe that our target solution $(\boldsymbol{u_x}, \boldsymbol{u_y}, \boldsymbol{u})$ of (17) is expected to be relatively short among the solutions of (17), because all of the coefficients of $u_x(t)$ and $u_y(t)$ are much smaller than q. This observation leads us to an approach to the key-recovery attack, as follows. First, we solve the system and find its solution space $\mathcal{L}_A$ and a solution $\boldsymbol{v}$. Second, we solve CVP to find the vector $\boldsymbol{w}$ closest to $\boldsymbol{v}$, and then $\boldsymbol{v} - \boldsymbol{w}$ is the smallest solution of (17) and is expected to be our target solution $(\boldsymbol{u_x}, \boldsymbol{u_y}, \boldsymbol{u})^T$.

In the case of $\deg X = 2$, our approach to the key-recovery attack is similar to the approach in the case of $\deg X = 1$. Now, our goal is to find $u_x(t), u_y(t) \in R_p$ satisfying

$$A \left(\boldsymbol{u_x^2}\ \boldsymbol{u_x u_y}\ \boldsymbol{u_y^2}\ \boldsymbol{u_x}\ \boldsymbol{u_y}\ \boldsymbol{u} \right)^T = \left(-\boldsymbol{a_{00}} \right), \tag{18}$$

where $A = \left(A_{20}\ A_{11}\ A_{02}\ A_{10}\ A_{01}\ qI_n \right)$. $A = (A_{20}\, A_{11}\, A_{02}\, A_{10}\, A_{01}\, qI_n)$. Note that each entry of the vector $(\boldsymbol{u_x^2}, \boldsymbol{u_x u_y}, \boldsymbol{u_x^2})^T$ is in $\mathbb{Z}_{np^2}$. We observe that the key-recovery attack for $\deg X = 2$ is much more difficult than that for $\deg X = 1$ because the solution has the non-linear parts $\boldsymbol{u_x^2}$, $\boldsymbol{u_x u_y}$, and $\boldsymbol{u_x^2}$, which are hard to handle with lattice-reduction techniques. In fact, the key-recovery attack for $\deg X = 2$ did not succeed at all in our experiments, while the attack for $\deg X = 1$ succeeded for some n. Moreover, Babai's nearest-plane algorithm could not find closer vectors than the correct vector with $n \geq 20$. (See Table 2 in Sect. 4.3 for these results.)

We also considered the latest lattice attacks, such as the lattice-decoding attack and the subfield-lattice attack. As discussed in Appendix A, these are not applicable to our scheme.

4.3 Computational Experiment

In this subsection, we show our experimental results for the two attacks above in order to estimate the parameters that make the IE-LWE problem intractable. In our experiments, we used Babai's nearest-plane algorithm [5], which is a standard algorithm for solving CVP approximately. A lattice basis-reduction algorithm, such as the LLL algorithm [19] or BKZ [29] algorithm, is used in Babai's nearest-plane algorithm.

We use the **root of Hermite factor** (RHF) as an index to evaluate the quality of Babai's nearest plane algorithm. RHF is larger than or equal to 1 in general, and the quality improves as RHF decreases.

The LLL algorithm is expected to achieve RHF = 1.0219. In the case of the BKZ algorithm, RHF depends on the block sizes β. For example, $\beta = 20$ and $\beta = 28$ suggest RHF = 1.0128 and RHF = 1.0109, respectively. (See [12] for these values of RHF).

Our computing environment is as follows.

- CPU: AMD Opteron (TM) Processor 848
- Memory: 64 GB
- OS: Linux version 2.6.18-406.el5.centos.plus
- Software: Magma Ver2.21-5

Experimental Results for the Linear Algebra Attack. After choosing X, r, and e uniformly at random as in the encryption process in Sect. 3.1, we set $(Y, Z) = (X, Xr+e)$ and conducted experiments to determine whether the target e or a polynomial with small coefficients $< p$ could be found. Our experiments were conducted for the cases of $\deg X = \deg r = 1$ and $\deg X = \deg r = 2$, and we set $p = 3$ and increased n in each case. We generated three IE-LWE

instances for each parameter set and applied the linear algebra attack described in Sect. 4.2 against each instance.

In Table 1, we show our experimental results for the linear algebra attack, where "Time" is the average time that it took to conduct the linear algebra attack and q is the smallest prime number satisfying (5).

Table 1. Experimental results for the linear algebra attack

n	q	Degree of X	RHF		Rank	Results	Time (s)
			Target	Babai			
10	14401	1	0.9831	0.9831	60	Success	2.27
20	57601	1	0.9903	0.9903	120	Success	48.08
30	129607	1	0.9930	0.9930	180	Success	189.9
40	230431	1	0.9944	0.9944	240	Success	1023.97
50	360007	1	0.9954	1.016	300	Failure	4847.95
60	518411	1	0.9959	1.015	360	Failure	19233.13
10	14400011	2	0.9860	0.9860	150	Success	396.62
20	230400007	2	0.9913	0.9913	300	Success	11680.77
30	1166400007	2	0.9936	0.9936	450	Success	79429.53
40	3686400041	2	0.9948	0.9948	600	Success	223644.52

The experimental results show that the linear algebra attack for $\deg X = 1$ failed for $n \geq 50$ and the attack for $\deg X = 2$ succeeded for $n \leq 40$. In the case of $\deg X = 2$, it took too much time to complete the attack when n was more than 40, since the rank of the lattice (15) increases in proportion to the square of $\deg Xr$ ($3n \times 9n$ for $\deg X = 1$, $6n \times 21n$ for $\deg X = 2$). The linear algebra attack appears to fail for values of n large enough that RHF > 1.

Experimental Results for the Key-Recovery Attack. We conducted the key-recovery attack described in Sect. 4.2 for the same instances as the linear algebra attack. We consider the key-recovery attack as having succeeded even if we find two polynomials with small coefficients $< p$ that differ from the correct secret key $(u_x(t), u_y(t))$.

The experimental results described in Table 2 show that the key-recovery attack for $\deg X = 1$ failed for $n \geq 50$ and that the key recovery attack for $\deg X = 2$ did not succeed at all.

Moreover, in the case of $\deg X = 2$, Babai's nearest-plane algorithm could not find closer vectors than the correct vector when $n \geq 20$. This implies that the algorithm is not able to find the correct vector when $n \geq 20$.

Table 2. Experimental results for the key-recovery attack

n	q	Degree of X	RHF		Rank	Results	Time (s)
			Target	Babai			
10	14401	1	0.8541	0.8541	20	Success	0.08
20	57601	1	0.9143	0.9143	40	Success	1.62
30	129607	1	0.9374	0.9374	60	Success	9.37
40	230431	1	0.9508	0.9508	80	Success	35.84
50	360007	1	0.9589	0.9981	100	Failure	107.48
60	518411	1	0.9646	1.018	120	Failure	268.56
10	14400011	2	1.022	1.017	50	Failure	2.06
20	230400007	2	1.017	1.021	100	Failure	48.70
30	1166400007	2	1.014	1.021	150	Failure	391.84
40	3686400041	2	1.011	1.021	200	Failure	2182.66

5 Appropriate Parameter Values

In this section, we design appropriate parameter values using the experimental results in Sects. 4.3. Both the linear algebra attack and the key-recovery attack for $\deg X = 1$ failed when $n \geq 50$. However, a key-recovery attack could also be done by using a brute force method, as follows. Choose $\tilde{u}_x(t)$ randomly until the correct $u_y(t)$ (or a polynomial with sufficiently small coefficients) is found by solving the one-variable equation $X(\tilde{u}_x(t), y) = 0$ over R_q. In order to resist the brute force attack, the parameter n must be set such that the number of candidates for $u_x(t)$ is at least 2^k, where k is the security parameter. Therefore, we need to set $n \geq 80$ when we keep 128 bit security. Note that $n \geq 80$ is also required in the case of $\deg X = 2$ because the brute-force attack is independent of the degree of X. In addition, n is preferred to be prime since our scheme employs the same algebra as NTRU [13]. Using the above argument, we designed appropriate parameter values for our encryption scheme, shown in Table 3.

Table 3. Appropriate parameter values for our scheme

p	q (bit)	n	$\deg X$	$\deg r$	$\#\Gamma_{X^r}$	Secret Key (bit)	Public Key (bit)	Ciphertext (bit)
3	20	83	1	1	6	264	4980	9960
3	36	83	2	2	15	264	17928	44820

Using [7], we show a comparison of our encryption scheme with other lattice-based encryption schemes known as efficient ring-homomorphic encryption schemes, in Table 4. Table 4 shows that the size of the ciphertext in our scheme is larger than that in LWE, but the sizes of public and secret keys in our scheme are the smallest among those in the schemes in Table 4.

Table 4. Comparison of our scheme with NTRU and LWE

Scheme	Secret Key		Public Key		Ciphertext	
	Theory	Actual (Kb)	Theory	Actual (Kb)	Theory	Actual (Kb)
NTRU [31]	$n\lceil\log_2 q\rceil$	≥ 70	$n\lceil\log_2 q\rceil$	≥ 70	$n\lceil\log_2 q\rceil$	≥ 70
LWE [20]	$n\lceil\log_2 q\rceil$	12	$2n\lceil\log_2 q\rceil$	24	$2n\lceil\log_2 q\rceil$	24
Our scheme	$2n\lceil\log_2 p\rceil$	0.3	$n\#\Delta_X\lceil\log_2 q\rceil$	18	$n\#\Delta_e\lceil\log_2 q\rceil$	45

From the point of view of solving indeterminate equations, the difference between the key-recovery attacks for our encryption scheme and the NTRU encryption scheme is the following. Our scheme for $\deg X = 2$ is based on the difficulty of finding a solution (a pair of univariate polynomials with small coefficients satisfying the non-linear indeterminate equation $X(x, y) = 0$. In contrast, the NTRU is based on the difficulty of finding polynomials f and g with small coefficients that satisfy the linear indeterminate equation $hx \equiv g \mod q$. Based on this difference, we conclude that the lattice basis-reduction in the NTRU is easier than that in our scheme. Moreover, this leads to the difference in the sizes of public and secret keys between our scheme and NTRU (and LWE).

6 Conclusion

In this study, we constructed a post-quantum encryption scheme whose security is based on an IE-LWE problem and related to the smallest-solution problem in non-linear spaces. This paper gave the algorithms for key generation, encryption/decryption, and the security proof in the sense of IND-CPA. Then, we discussed two attacks that can be applied to the IE-LWE problem and estimated the key size of our scheme according to the results of the computational experiment for these attacks. The sizes of the keys are estimated to be much smaller than those of lattice-based cryptosystems such as LWE and NTRU since no efficient approximation algorithms are known for non-linear spaces. Finally, we described our computational experiment to solve the problem using Babai's nearest-plane algorithm with LLL. In the future, we plan to conduct experiments using the lattice decoding attack and the subfield lattice attack to solve the problem.

Acknowledgments. The authors thank Keita Xagawa for suggesting us the attack [4,13] may work against our scheme when we choose the parameter n to be composite. The authors also thank anonymous referees for careful reading of our manuscript and for giving helpful comments.

A Further Discussion on Lattice Attacks

In this section, we discuss and analyze whether other lattice attacks, such as a lattice-decoding attack [6] and a subfield-lattice attack [18], can be applied to our scheme. The discussion and analysis of these attacks given here is rough. We plan to conduct more careful discussion and analysis in future work. In addition, analyzing the enumeration methods for CVP (e.g., [21]) is another important area for future study.

A.1 Lattice-Decoding Attack

The lattice decoding attack consists of three techniques: Kanan's embedding technique for reducing CVP to SVP [17], the BKZ algorithm for solving SVP, and the re-scaling of lattices. More precisely, the attack first reduces the search binary-LWE problem to the inhomogeneous short integer solution (ISIS) problem and then tries to solve the ISIS problem by reducing it to CVP. Kanan's embedding technique and the BKZ algorithm are used to solve the CVP. The re-scaling technique is required because some elements in the target vector are unbalanced in size. This approach seems to be applicable to the original search-LWE [28] as well as our scheme, but the shortness of the secret vector $\boldsymbol{s}$ is used in the analysis of the lattice decoding attack. However, for our scheme, the vector $\boldsymbol{r}$, which corresponds to $\boldsymbol{s}$ in the binary-LWE problem, is not short in general since the scheme requires that the vector $\boldsymbol{r}$ be chosen uniformly at random from $\mathbb{Z}_q$. Therefore, the lattice-decoding attack on the binary-LWE problem does not appear to be applicable to our scheme.

However, the embedding technique is applicable to the key-recovery and linear algebra attacks described in previous subsections. In fact, when we applied the technique to them, we obtained almost the same results as for our scheme.

A.2 Subfield-Lattice Attack

Here, we discuss the subfield-lattice attack on our scheme. This attack can be applied to homomorphic variants of NTRU. The attack reduces the lattice problem on certain number fields to the problem on their appropriate subfields by using norm maps from the original number fields to the subfields.

NTRU variants (i.e., the NTRU on $\mathbb{Z}_q[x]/(x^{2^k}+1)$ and $\mathbb{Z}_q[x]/(x^p-x-1)$ with prime numbers p and positive integers q) have been addressed in previous experiments by Kirchner et al. [18, Sect. 5]. There is no subfield of the number field $\mathbb{Q}[x]/(x^p-x-1)$, but the attack on $\mathbb{Z}_q[x]/(x^p-x-1)$ succeeds for many parameters. We infer that the size of the parameter q is strongly related to the success of the attack. As the size of q increases, the volume of the lattice becomes larger, and the SVP on the lattice becomes easier. In fact, the subfield attacks on NTRU with relatively small q fail in some cases (see [18, Figs. 1 and 2]). Moreover, the form $h = f/g$ of the public key for NTRU seems to have a positive effect on the attack, where f and g are secret polynomials with small coefficients and f is invertible in $\mathbb{Z}_q[x] = (x^{2^k}+1)$ or $\mathbb{Z}_q[x] = (x^p-x-1)$.

However, when comparing Table 3 in this paper with [18, Figs. 1 and 2], it is evident that the size of q in our scheme is much smaller than that of the NTRU variants. Moreover, there is a gap between the forms of the keys (public/secret-keys) in our scheme and those in the above NTRU variants. The data shows that the lattices derived from the two attacks on our scheme are very different from those derived from the subfield attacks on the above NTRU variants. Therefore, the subfield attack does not appear to be applicable to our scheme. In future work, we plan to consider a variant of the subfield attack on our scheme.

B Maximum Coefficient of Noise Term e

For our scheme, the condition $MC(p \cdot e(u)) < q$ is required in order to decrypt. In this section, we describe several properties of $MC(f(t))$ and use them to prove the condition (5).

For any a in $\mathbb{Z}_q^+$ and any $f(t), g(t)$ in R_q, the relation

$$\begin{aligned} &MC(af(t)) \le a \cdot MC(f(t)) \\ &MC(f(t)+g(t)) \le MC(f(t)) + MC(g(t)) \\ &MC(f(t)g(t)) \le n \cdot MC(f(t))MC(g(t)), \end{aligned} \tag{19}$$

are satisfied, where the equality is satisfied when all the coefficients of $f(t)$ and $g(t)$ are the same.

Considering the worst case gives us $u_x(t) = u_y(t) = \sum_{i=0}^{n-1}(p-1)t^i$. By applying (19) repeatedly, we obtain the following:

$$\begin{aligned} MC(e(u)) &= MC(\textstyle\sum_{(i,j)\in\Gamma_{Xr}} e_{ij}(t)u_x(t)^i u_y(t)^j) \\ &\le \textstyle\sum_{(i,j)\in\Gamma_{Xr}} MC(e_{ij}(t)u_x(t)^i u_y(t)^j) \\ &\le \textstyle\sum_{(i,j)\in\Gamma_{Xr}} n^{i+j} MC(e_{ij}(t))MC(u_x(t))^i MC(u_y(t))^j \\ &\le \textstyle\sum_{(i,j)\in\Gamma_{Xr}} (p-1)\cdot(n(p-1))^{i+j} \\ &\le \#\Gamma_{Xr}\cdot(p-1)\cdot(n(p-1))^{w_X+w_r}. \end{aligned}$$

The relation leads to the following condition:

$$q > \#\Gamma_{Xr}\cdot p(p-1)\cdot(n(p-1))^{w_X+w_r}. \tag{20}$$

This is the condition (5), so the condition (5) is proven.

References

1. Ajtai, M., Dwork, C.: A public-key cryptosystem with worst-case/average-case equivalence. In: Proceedings of STOC 1997, pp. 284–293. ACM New York (1997)
2. Akiyama, K., Goto, Y.: A public-key cryptosystem using algebraic surfaces. In: Proceedings of PQCrypto 2006, pp. 119–138 (2006). http://postquantum.cr.yp.to/
3. Akiyama, K., Goto, Y., Miyake, H.: An algebraic surface cryptosystem. In: Jarecki, S., Tsudik, G. (eds.) PKC 2009. LNCS, vol. 5443, pp. 425–442. Springer, Heidelberg (2009). https://doi.org/10.1007/978-3-642-00468-1_24

4. Albrecht, M., Bai, S., Ducas, L.: A subfield lattice attack on overstretched NTRU assumptions. In: Robshaw, M., Katz, J. (eds.) CRYPTO 2016. LNCS, vol. 9814, pp. 153–178. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-53018-4_6
5. Babai, L.: On Lovász' lattice reduction and the nearest lattice point problem. Combinatorica **6**(1), 1–13 (1986). (Preliminary version in STACS 1985)
6. Bai, S., Galbraith, S.D.: Lattice decoding attacks on binary LWE. In: Susilo, W., Mu, Y. (eds.) ACISP 2014. LNCS, vol. 8544, pp. 322–337. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-08344-5_21
7. Bansarkhani, R.E., Cabarcas, D., Kuo, P.C., Schmidt, P., Schneider, M.: A selection of recent lattice-based signature and encryption schemes. Tatra Mt. Math. Publ. **53**(1), 81–102 (2012)
8. Chen, Y., Nguyen, P.Q.: BKZ 2.0: better lattice security estimates. In: Lee, D.H., Wang, X. (eds.) ASIACRYPT 2011. LNCS, vol. 7073, pp. 1–20. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-25385-0_1
9. Denef, J.: The Diophantine Problem for polynomial rings of positive characteristic. In: Proceedings of Logic Colloquium 1978, Studies in Logic and the Foundations of Mathematics, North Holland, Amsterdam-New York, vol. 97, pp. 131–145 (1979)
10. Fujisaki, E., Okamoto, T.: How to enhance the security of public-key encryption at minimum cost. In: Imai, H., Zheng, Y. (eds.) PKC 1999. LNCS, vol. 1560, pp. 53–68. Springer, Heidelberg (1999). https://doi.org/10.1007/3-540-49162-7_5
11. Faugère, J.-C., Spaenlehauer, P.-J.: Algebraic cryptanalysis of the PKC 2009 algebraic surface cryptosystem. In: Nguyen, P.Q., Pointcheval, D. (eds.) PKC 2010. LNCS, vol. 6056, pp. 35–52. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-13013-7_3
12. Gama, N., Nguyen, P.Q.: Predicting lattice reduction. In: Smart, N. (ed.) EUROCRYPT 2008. LNCS, vol. 4965, pp. 31–51. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-78967-3_3
13. Gentry, C.: Key recovery and message attacks on NTRU-composite. In: Pfitzmann, B. (ed.) EUROCRYPT 2001. LNCS, vol. 2045, pp. 182–194. Springer, Heidelberg (2001). https://doi.org/10.1007/3-540-44987-6_12
14. Goldreich, O., Goldwasser, S., Halevi, S.: Public-key cryptosystems from lattice reduction problems. In: Kaliski, B.S. (ed.) CRYPTO 1997. LNCS, vol. 1294, pp. 112–131. Springer, Heidelberg (1997). https://doi.org/10.1007/BFb0052231
15. Hoffstein, J., Pipher, J., Silverman, J.H.: NTRU: a ring-based public key cryptosystem. In: Buhler, J.P. (ed.) ANTS 1998. LNCS, vol. 1423, pp. 267–288. Springer, Heidelberg (1998). https://doi.org/10.1007/BFb0054868
16. Matsumoto, T., Imai, H.: Public quadratic polynomial-tuples for efficient signature-verification and message-encryption. In: Barstow, D., et al. (eds.) EUROCRYPT 1988. LNCS, vol. 330, pp. 419–453. Springer, Heidelberg (1988). https://doi.org/10.1007/3-540-45961-8_39
17. Kannan, R.: Minkowski's convex body theorem and integer programming. Math. Oper. Res. **12**(3), 415–440 (1987). INFORMS, Linthicum, Maryland, USA
18. Kirchner, P., Fouque, P.-A.: Comparison between Subfield and Straightforward Attacks on NTRU, IACR Cryptology ePrint Archive: Report 2016/717. http://eprint.iacr.org/2016/717
19. Lenstra, A.K., Lenstra Jr., H.W., Lovasz, L.: Factoring polynomials with rational coefficients. Math. Ann. **261**(4), 515–534 (1982). Springer
20. Lindner, R., Peikert, C.: Better key sizes (and attacks) for LWE-based encryption. In: Kiayias, A. (ed.) CT-RSA 2011. LNCS, vol. 6558, pp. 319–339. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-19074-2_21

21. Liu, M., Nguyen, P.Q.: Solving BDD by enumeration: an update. In: Dawson, E. (ed.) CT-RSA 2013. LNCS, vol. 7779, pp. 293–309. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-36095-4_19
22. McEliece, R.J.: A public-key cryptosystem based on algebraic coding theory. The Deep Space Network Progress Report, DSN PR 42–44, pp. 114–116 (1978)
23. Nguyen, P.: Cryptanalysis of the Goldreich-Goldwasser-Halevi cryptosystem from Crypto '97. In: Wiener, M. (ed.) CRYPTO 1999. LNCS, vol. 1666, pp. 288–304. Springer, Heidelberg (1999). https://doi.org/10.1007/3-540-48405-1_18
24. Nguyen, P., Stern, J.: Cryptanalysis of the Ajtai-Dwork cryptosystem. In: Krawczyk, H. (ed.) CRYPTO 1998. LNCS, vol. 1462, pp. 223–242. Springer, Heidelberg (1998). https://doi.org/10.1007/BFb0055731
25. https://www.nsa.gov/ia/programs/suiteb_cryptography/ (2015)
26. Patarin, J.: Hidden Fields Equations (HFE) and Isomorphisms of Polynomials (IP): two new families of asymmetric algorithms. In: Maurer, U. (ed.) EUROCRYPT 1996. LNCS, vol. 1070, pp. 33–48. Springer, Heidelberg (1996). https://doi.org/10.1007/3-540-68339-9_4
27. Porras, J., Baena, J., Ding, J.: ZHFE, a new multivariate public key encryption scheme. In: Mosca, M. (ed.) PQCrypto 2014. LNCS, vol. 8772, pp. 229–245. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-11659-4_14
28. Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. J. ACM **56**(6), 1–40 (2009). ACM, New York
29. Schnorr, C.R., Euchner, M.: Lattice basis reduction: improved algorithms and solving subset sum problems. Math. Program. **66**(1), 181–189 (1994). Springer
30. Shor, P.W.: Algorithms for quantum computation: discrete log and factoring. In: Proceedings of SFCS 1994, pp. 124–134. IEEE Computer Society Washington (1994)
31. Stehlé, D., Steinfeld, R.: Making `NTRU` as secure as worst-case problems over ideal lattices. In: Paterson, K.G. (ed.) EUROCRYPT 2011. LNCS, vol. 6632, pp. 27–47. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-20465-4_4
32. Tao, C., Diene, A., Tang, S., Ding, J.: Simple matrix scheme for encryption. In: Gaborit, P. (ed.) PQCrypto 2013. LNCS, vol. 7932, pp. 231–242. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-38616-9_16
33. Yasuda, T., Sakurai, K.: A multivariate encryption scheme with rainbow. In: Qing, S., Okamoto, E., Kim, K., Liu, D. (eds.) ICICS 2015. LNCS, vol. 9543, pp. 236–251. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-29814-6_19