OUKA
The University of Osaka
Institutional Knowledge Archive

Title	Lattice-Based Public Key Cryptosystems Invoking Linear Mapping Mask
Author(s)	Wang, Yuntao; Ikematsu, Yasuhiko; Yasuda, Takanori
Citation	Lecture Notes in Computer Science. 2022, 13600, p. 88-104
Version Type	AM
URL	https://hdl.handle.net/11094/90001
rights	
Note	

The University of Osaka Institutional Knowledge Archive : OUKA

https://ir.library.osaka-u.ac.jp/

The University of Osaka

Lattice-based Public Key Cryptosystems invoking Linear Mapping Mask

Yuntao Wang[1⋆][0000−0002−2872−4508], Yasuhiko Ikematsu[2][0000−0002−9714−2675], and Takanori Yasuda[3][0000−0002−2536−7429]

[1] Graduate School of Engineering, Osaka University
wang@comm.eng.osaka-u.ac.jp
[2] Institute of Mathematics for Industry, Kyushu University
ikematsu@imi.kyushu-u.ac.jp
[3] Institute for the Advancement of Higher Education, Okayama University of Science
tyasuda@ous.ac.jp

Abstract. In ProvSec 2018, Yasuda proposed a multivariate public key cryptosystem using the pq-method, whose security is based on the constrained MQ problem. Afterward, in SCIS 2020, he improved the cryptosystem by adding noise elements and simultaneously considered the cryptanalysis using the NTRU method. This improved cryptosystem is the first one combining lattice and multivariate public-key cryptosystem. In this paper, we propose three variants of Yasuda's cryptosystem. The main improvement is that we invite the linear structures instead of the multivariate quadratic polynomials. In particular, we simplify the procedure in key generation mechanism by using a linear mapping mask which produces resistance against the key-recovery attack. Furthermore, we propose a ring version that is quite efficient compared to the standard versions. Finally, we adopt the ring-LWE method instead of the original NTRU method to give a more promising cryptanalysis.

Keywords: Post-Quantum Cryptography, Lattice, Public Key Cryptography

1 Introduction

Nowadays, the security of modern public-key cryptographic schemes, such as RSA, ECC, DSA, ElGamal, and Diffie-Hellman key exchange, are based on number theoretic hard problems such as integer factorization problem (IFP), discrete logarithm problem (DLP) and their elliptic curve variants in certain groups, etc. With elaborately chosen parameters and implementations, the above cryptographic schemes are temporarily secure against current computing resources. However, it is known that overwhelming computing power is available by a quantum computer compared to classic computers. By coordinating with Shor's quantum algorithm [19], most of the above crypto algorithms are vulnerable to being broken in polynomial time by a sufficiently powerful quantum computer in

⋆ corresponding author

the near future. Since the above cryptosystems are widely deployed in real-world applications (e.g., IoT, HTTPS, online banking, cryptocurrency, software, etc.), developing secure and practical next-generation cryptographic algorithms is urgent. The well-known "post-quantum cryptography" (PQC) is highly expected to withstand outstanding quantum attacks. Actually, some international standards organizations such as NIST, ISO, and IETF already started the PQC standardization projects several years ago. They mainly focus on three primitives: Public-Key Encryption algorithms (PKE), the Key Encapsulation Mechanism (KEM), and the digital signature schemes. Among the several categories, lattice-based cryptography is considered a promising contender for its robust security strength, comparative light communication cost, desirable efficiency, and excellent adaptation capabilities [1]. Indeed, three over four PKE/KEM/signature algorithms are lattice-based candidates in PQC standardization announced by NIST in 2022 [2]. Further, four PKE/KEM algorithms are selected for the fourth-round candidate finalists.

The Multivariate Public-Key Cryptography (MPKC) is also one important component in PQC, where there is one multivariate scheme among three digital signature candidates in the third-round finalists [2]. The security of MPKC is based on the hardness of solving Multivariate Quadratic polynomials (MQ) problems. On the one side, the MPKC signature scheme is efficient due to conducting in a small number field. On the other side, MPKC is generally not adaptable for PKE due to its larger key length compared to other categories. For instance, some encryption schemes such as Simple Matrix Scheme [22], EFC [21], and HFERP [13] have been proposed these years. All these cryptosystems are suffering from a common and critical shortcoming: a large number of variables are required for a relatively secure level but incur a higher cost for encryption and decryption. One method to overcome this shortcoming is constructing trapdoor one-way functions given by injective polynomial maps. However, it is observed that one can turn a polynomial map injective easily by adding a restriction on its definition range. Namely, by using a constrained polynomial map, it is easy to construct an injective trapdoor one-way function. As a result, this function can be used to construct secure MPKC encryption schemes whose security is based on the hardness of solving the constrained multivariate polynomial problem.

It is known that the key sizes or the ciphertext sizes of MPKC or lattice-based cryptography are usually larger than twice the sizes conducted in most classical public key crypto schemes in cases of AES-128 bit security. However, due to the security of the latter being based on some number of theoretical problems, the computation cost takes more than the former securely based on the algebraic problems. In ProvSec2018, Yasuda proposed a multivariate PKE using the so-called *pq*-method [27]. The security of *pq*-method is based on the difficulty of solving the constrained MQ problem which is a hard problem in MPKC. Substantially, the constrained MQ problem can be seen as a quadratic version of the Inhomogeneous Short Integer Solution (ISIS) problem in lattice theory as well. For this reason, the cryptosystem using *pq*-method is considered the first PKE combining lattice and MPKC. In order to reduce the size of the

public key, Yasuda further improved the *pq*-method by inviting an error term in the encryption phase [28]. This idea is from the classical Learning with Error (LWE) problem in lattice theory which has been widely used in lattice-based cryptography [17]. The improved version of *pq*-method is named by *pqe*-method.

1.1 Motivations and Contributions

Motivations. The combination of an MPKC and lattice-based cryptography is a novel idea that may derive some benefits from the aspects of both computational cost and security: on the one side, the linear algebraic structure in lattice may provide a desirable efficiency and comparative communication cost, while cryptanalysis is challenging to handle due to a lack of thoroughgoing grasp of lattice (reduction) algorithms; on the other side, MPKC holds robust security but requires more variables resulting in relatively lower efficiency. Particularly, solving quadratic polynomials in MPKC obviously takes more effort than dealing with linear polynomials in lattice. This circumstance occurs in both mentioned above *pq*-method and the *pqe*-method, where we consider proposed methods that enjoy the lattice's fast computation and low cost and further strengthen their security from MPKC.

Contributions. We list the following contributions in this paper. Here, n is the number of variables, and p, q are moduli in the schemes.

- First, we improve the *pq*-method and the *pqe*-method by using linear structures instead of the quadratic polynomials, which we call linear-*pq* method and linear-*pqe* method, respectively. Due to different cryptanalysis, it is difficult to directly compare the keysize and ciphertext size between the improved crypto schemes and the original ones. Nevertheless, we can see the results by comparing the inside parameters used in each scheme. We summarize the parameter size and the polynomial multiplication cost in the following table.

	original *pq* method	linear-*pq* method	original *pqe* method	linear-*pqe* method
q	$O(n^4p^6)$	$O(n^2p^4)$	$O(n^4p^6)$	$O(n^2p^5)$
computational cost	$O(n^4)$	$O(n^2 \log n)$	$O(n^4)$	$O(n^2 \log n)$

- Moreover, applying a linear mapping mask at the end of key generation can strengthen the security against the key-recovery attack using the property of MPKC.
- Additionally, we propose a ring version of the linear-*pqe* method. As a result, the key size and computational cost are substantially reduced by a factor of $1/n$, which makes it the most efficient with the smallest key size among the three proposals.
- The security parameters are evaluated by the ring-LWE method rather than the original NTRU method.

1.2 Organization

Section 2 recalls the notations and background, including some hard problems in lattice theory and multivariate polynomial theory. In particular, the predecessor of our proposals is introduced in this section. Then, we introduce our proposed public key encryption algorithms in Section 3. In Section 4, we estimate the security parameters with respect to AES-128 bit security using the 2016 estimate which is commonly used in cryptanalysis for lattice-based cryptosystems. We also evaluate the size of keys and ciphertext, and show the practical performance of each proposal. Finally, we conclude our work in Section 5.

2 Preliminaries

In this section, we prepare some mathematical notations used in the paper. Then we recall the computational problems associated with lattices and multivariate polynomials. At last, we review the *pq*-method [27] and the *pqe*-method [28] proposed by Yasuda, respectively.

Notations. Let m, n and l be positive integers ($\in \mathbb{Z}_{>0}$). The set $[m]$ means $\{1, \ldots, m\}$. Denote by $\mathbb{Z}_l$ the residue ring modulo l, i.e. the elements in $\mathbb{Z}_l$ are from 0 to $l-1$. For an element $a \in \mathbb{Z}_l$, we define a lift function of a by $\mathrm{lift}_l(a) \in I_l$ where $I_l := (-l/2, l/2] \cap \mathbb{Z}$. Simultaneously, we denote a finite field $\mathbb{F}_q$ with q a prime number. We represent n independent variables of $x_{i\in[n]}$ by a row vector of $\mathbf{x} = (x_1, \ldots, x_n)$. The set of polynomials with variables in $\mathbf{x}$ and coefficients in $\mathbb{F}_q$ is denoted by $\mathbb{F}_q[\mathbf{x}]$. Then, we prepare a sequence of m (upper triangular) matrices $A_1, \ldots, A_m \in \mathbb{F}_q^{n\times n}$, m row vectors $\mathbf{b}_1, \ldots, \mathbf{b}_m \in \mathbb{F}_q^n$, and m constants $c_1, \ldots, c_m \in \mathbb{F}_q$. By using the above notations, we define a quadratic polynomial system in $\mathbb{F}_q[\mathbf{x}]^m$ as $\mathcal{F}(\mathbf{x}) := \{f_i(\mathbf{x}) := \mathbf{x}A_i\mathbf{x}^T + \mathbf{b}_i\mathbf{x}^T + c_i := \sum_{j,k\in[n]} a_{ijk}x_jx_k + \sum_{j\in[n]} b_{ij}x_j + c_i \pmod q\}_{i\in[m]}$. For the sake of convenience, we write its vector form by $\mathcal{F}(\mathbf{x}) = (f_1(\mathbf{x}), \ldots, f_m(\mathbf{x})) \in \mathbb{F}_q[\mathbf{x}]^m$ in this paper. Moreover, we define $\texttt{NextPrime}(x)$ the first prime number no smaller than $x \in \mathbb{R}$.

2.1 Lattice

Lattice. A *lattice* L is generated by a *basis* B which is a set of linearly independent vectors $\mathbf{b}_1, \ldots, \mathbf{b}_n$ in $\mathbb{R}^m$: $L(\mathbf{b}_1, \ldots, \mathbf{b}_n) = \{\sum_{i=1}^n x_i\mathbf{b}_i, x_i \in \mathbb{Z}\}$. Note that in this paper we use integer lattices for convenience and we write the basis in a matrix form as $B = (\mathbf{b}_1, \ldots, \mathbf{b}_n) \in \mathbb{Z}^{m\times n}$. The integer n is the *rank* of the lattice, which equals to the *dimension* of the vector space spanned by L, i.e. $n = \dim(\mathrm{span}(L))$. It is called *full-rank* lattice when $m = n$.

The *Euclidean norm* of a lattice vector $\mathbf{v} \in \mathbb{R}^m$, also known as l_2-norm, is $\|\mathbf{v}\| := \sqrt{\mathbf{v}\cdot\mathbf{v}}$. There are at least two non-zero vectors with the same minimal Euclidean norm but contrary sign in a lattice L with basis $B = (\mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_n)$: this norm is called the 1-*st successive minimum* $\lambda_1(L)$ of $L(B)$. A *shortest vector* of L is of norm $\lambda_1(L)$.

The Shortest Vector Problem. Given a basis $B = (\mathbf{b}_1, \ldots, \mathbf{b}_n)$ of a lattice L, the *Shortest Vector Problem* asks to find a non-zero shortest vector in L. SVP is NP-hard for randomized reductions.

Unique Shortest Vector Problem. The *Unique SVP Problem* (uSVP) is for a given lattice L which satisfies $\lambda_1(L) \ll \lambda_2(L)$, to find the shortest nonzero vector in L. It is called γ *Unique SVP problem* if the gap of $\lambda_2(L)/\lambda_1(L) = \gamma$ is known.

Inhomogeneous Short Integer Solution Problem. Given an integer q, a matrix $\mathbf{A} \in \mathbb{Z}_q^{n \times m}$ and a vector $\mathbf{v} \in \mathbb{Z}_q^n$, *Inhomogeneous Short Integer Solution Problem* is to compute a short vector $\mathbf{y} \in \mathcal{B}$ s.t. $\mathbf{A}\mathbf{y} \equiv \mathbf{v} \pmod{q}$, where $\mathcal{B}$ is a set of short vectors with some Euclidean norm bound.

Learning with Errors (LWE) Problem [17]. There are four parameters in the LWE problem: the number of samples $m \in \mathbb{Z}$, the length $n \in \mathbb{Z}$ of secret vector, modulo $q \in \mathbb{Z}$ and the standard deviation $\sigma \in \mathbb{R}_{>0}$ for the discrete Gaussian distribution $D_{\mathbb{Z}^n,\sigma}$. Sample a matrix $\mathbf{A} \in \mathbb{Z}_q^{m \times n}$ and a secret vector $\mathbf{s} \in \mathbb{Z}_q^n$ uniformly at random, and randomly sample a relatively small perturbation vector $\mathbf{e} \in \mathbb{Z}_q^m$ from Gaussian distribution $D_{\mathbb{Z}^n,\sigma}$, i.e. $\mathbf{e} \xleftarrow{\$} D_{\mathbb{Z}^n,\sigma}$. The LWE distribution Ψ is constructed by pairs $(\mathbf{A}, \mathbf{b} \equiv \mathbf{A}\mathbf{s} + \mathbf{e} \pmod{q}) \in (\mathbb{Z}_q^{m \times n}, \mathbb{Z}_q^m)$ sampled as above. The search *learning with errors problem* (LWE problem) is for a given pair $(\mathbf{A}, \mathbf{b})$ sampled from LWE distribution Ψ, to compute the pair $(\mathbf{s}, \mathbf{e})$. The decision version of LWE problem asks to distinguish if the given pair $(\mathbf{A}, \mathbf{b})$ is sampled from LWE or uniform distribution. The proof of equivalent hardness between these two versions is given in the original LWE paper [17].

Ring Learning With Errors (Ring-LWE) Problem [14] Let $m \geq 1$ be a power of 2 and $q \geq 2$ be an integer. Let $R_q = \mathbb{Z}_q[x]/\Phi_m(x)$, where $\Phi(x)$ is an irreducible polynomial with degree n. Let χ be a β-bounded distribution. For secret polynomial $\mathbf{s} \xleftarrow{\$} R_q$ and error polynomial $\mathbf{e} \xleftarrow{\$} \chi$, choosing $\mathbf{a} \in R_q$ uniformly at random, output $(\mathbf{a}, \mathbf{b} = \mathbf{a} \cdot \mathbf{s} + \mathbf{e}) \in (R_q, R_q)$. The search version of *ring learning with errors problem* (Ring-LWE problem) is: for $\mathbf{s} \xleftarrow{\$} R_q$, given $poly(n)$ number of samples of $(\mathbf{a}, \mathbf{b} = \mathbf{a} \cdot \mathbf{s} + \mathbf{e}) \in (R_q, R_q)$, find $\mathbf{s}$ (and $\mathbf{e}$ simultaneously).

2.2 Multivariate public key cryptography (MPKC)

In this subsection, we introduce the MP/MQ problems and their constrained variants used as security bases in MPKC.

Multivariate Polynomial Problem. Given a polynomial system of $\mathcal{F}(\mathbf{x}) \in \mathbb{F}_q[\mathbf{x}]^m$ with n variables and m polynomials, the *multivariate polynomial problem* (MP problem) is to find a solution of $\mathbf{x}_0 = (x_{01}, \ldots, x_{0n}) \in \mathbb{F}_q^n$ such that $\mathcal{F}(\mathbf{x}_0) = \mathbf{0}$. The hardness of MP problem is proven to be NP-complete [11].

Constrained Multivariate Polynomial Problem [27]. Given a bound parameter $L \in \mathbb{Z}_{>0}$ and a polynomial system of $\mathcal{F}(\mathbf{x}) \in \mathbb{F}_q[\mathbf{x}]^m$ with n variables and

m polynomials, the *constrained multivariate polynomial problem* (constrained-MP problem) asks to find a solution of $\mathbf{x}_0 = (x_{01}, \ldots, x_{0n}) \in I_L^n$ such that $\mathcal{F}(\mathbf{x}_0) = \mathbf{0}$.

When only quadratic polynomials are used in the MP problem (or the constrained MP problem), the problem is called the MQ problem (or the constrained MQ problem, respectively). Namely, the more specific MQ problem and constrained-MQ problem being used in multivariate public key cryptography are the versions invoking quadratic polynomials in the MP problem and Constrained-MP problem, respectively.

Common construction of quadratic MPKC based on MQ problem. Let n, m be two integers and q be a prime number. In a quadratic MPKC, the secret keys include an invertible quadratic map $F : \mathbb{F}_q^n[\mathbf{x}] \to \mathbb{F}_q^m[\mathbf{x}]$ and two affine maps of $S : \mathbb{F}_q^n[\mathbf{x}] \to \mathbb{F}_q^n[\mathbf{x}]$ and $T : \mathbb{F}_q^m[\mathbf{x}] \to \mathbb{F}_q^m[\mathbf{x}]$; the public key is bipolar structure with a composition $P = S \circ F \circ T : \mathbb{F}_q^n[\mathbf{x}] \to \mathbb{F}_q^m[\mathbf{x}]$. A plaintext $\mathbf{m} \in \mathbb{F}_q^n$ is encrypted by $\mathbf{c} = P(\mathbf{m})$; and $\mathbf{c}$ can be decrypted by $\mathbf{m} = T^{-1}(F^{-1}(S^{-1}(\mathbf{c}))$. The security of MPKC is based on the assumption that P is hard to invert without the secret keys. Matsumoto and Imai initially proposed the crypto scheme based on MP problem in EUROCRYPT'88 [15]. However, it was broken by Patarin in CRYPTO'95 [16].

2.3 MPKC using *pq*-method and *pqe*-method

First, we recap the cryptosystem using the *pq*-method proposed by Yasuda in [27]. Refer to the original paper for more details of the regime constructing the multivariate polynomial trapdoor function system $G(\mathbf{x})$.

- **Key Generation**:
  Let p be an odd prime number, n be a positive integer, and l_ψ be a positive odd integer.
  1) Randomly sample a multivariate quadratic polynomial system $\Phi(\mathbf{x}) \in \mathbb{Z}[\mathbf{x}]^n$. Here $\Phi(\mathbf{x}) \mod p$ is (almost) injective, and its inverse can be computed efficiently.
  2) Make a quadratic multivariate polynomial system $\Psi(\mathbf{x}) \in \mathbb{Z}[\mathbf{x}]^n$ with coefficients sampled from I_{l_ψ} uniformly at random. (Note that $I_{l_\psi} = (-l_\psi/2, l_\psi/2] \cap \mathbb{Z}$)
  3) Choose a prime number q satisfying $q > 4M_\psi M_\phi$, where
  $$\begin{aligned} M_\psi &\geq \max_{i\in[n]} \left\{ |\psi_i(\mathbf{d})| \middle| \mathbf{d} \in I_p^n \right\}, \\ M_\phi &\geq \max_{i\in[n]} \left\{ |\phi_i(\mathbf{d})| \middle| \mathbf{d} \in I_p^n \right\}. \end{aligned} \tag{1}$$
  4) Select a series of integers $r_1, \ldots, r_n$ in the range of (M_ϕ, q) satisfying $2M_\phi < \min_{k\in[2M_\psi]} |\mathrm{lift}_q(r_i k)|_{i\in[n]}$. Go back to Step 3 if it failed to sample such $r_1, \ldots, r_n$. Here we denote by $\Lambda_i = \{\mathrm{lift}_q(r_i k) | k = 0, \pm 1, \ldots, \pm M_\psi\}$.

5) Compute $\Psi_R(\mathbf{x}) = (r_1\psi_1(\mathbf{x}), \ldots, r_n\psi_n(\mathbf{x})) \in \mathbb{Z}[\mathbf{x}]^n$, and $G(\mathbf{x}) = (g_1(\mathbf{x}), \ldots, g_n(\mathbf{x})) = (\Phi(\mathbf{x}) + \Psi_R(\mathbf{x})) \mod q \in \mathbb{F}_q[\mathbf{x}]^n$.
6) Randomly sample an affine transformation T on $\mathbb{F}_q^n$; randomly sample a permutation matrix S of size n.
7) Compute $F = T \circ G \circ S : \mathbb{Z}^n \to \mathbb{F}_q^n$.
Secret key: $\Phi(\mathbf{x}) \mod p$, $\{r_i\}_{i\in[n]}$, T and S;
Public key: p and $F(\mathbf{x})$.

- **Encryption**:
Given a plaintext $\mathbf{m} \in I_p^n$, compute the ciphertext $\mathbf{c} = F(\mathbf{m}) \in \mathbb{F}_q^n$.

- **Decryption**:
1) Compute $\mathbf{c}' = (c'_1, \ldots, c'_n) = T^{-1}(\mathbf{c})$.
2) Find a (unique) $\lambda_i \in \Lambda_i$ such that $|\mathrm{lift}_q(c'_i - \lambda_i)| < M_\phi$ for all $i \in [n]$. Set $\tilde{c}_i = \mathrm{lift}_q(c'_i - \lambda_i) \in \mathbb{Z}$.
3) Calculate the solution $\tilde{\mathbf{b}} \in I_p^n$ of the equations $\Phi(\mathbf{x}) \equiv (\tilde{c}_1, \ldots, \tilde{c}_n) \mod p$.
4) Compute $\mathbf{m}' = S^{-1}(\tilde{\mathbf{b}})$ which matches with the plaintext $\mathbf{m}$.

In SCIS 2020 [28], Yasuda further improved the *pq*-method by adding a noise polynomial into the encryption process called *pqe*-method. Given a matrix $A = (a_{ij}) \in \mathbb{Z}^{n\times n}$, we set

$$M_A = \max_{i\in[n]}\{\sum_{j=1}^{n} |a_{ij}|\}. \tag{2}$$

We show the *pqe*-method in the following algorithm.

- **Key Generation**:
Let p be an odd prime number and n be a positive integer. l_ψ, l_A, l_B be positive odd integers of size close to p.
1) Randomly sample a multivariate quadratic polynomial system $\Phi(\mathbf{x}) \in \mathbb{Z}[\mathbf{x}]^n$. Here $\Phi(\mathbf{x}) \mod p$ is (almost) injective and its inverse can be computed efficiently.
2) Make a multivariate polynomial system $\Psi(\mathbf{x}) \in \mathbb{Z}[\mathbf{x}]^n$ with coefficients sampled from I_{l_ψ} uniformly at random.
3) Randomly sample matrices $A \in I_{l_A}^{n\times n}$ and $B \in I_{l_B}^{n\times n}$.
4) Compute M_ϕ, M_ψ by (1) and M_A, M_B by (2). Then compute

$$\widetilde{M_\phi} = M_\phi + M_A \cdot \tfrac{l_A-1}{2}$$
$$\widetilde{M_\psi} = M_\psi + M_B \cdot \tfrac{l_B-1}{2}.$$

5) Choose a prime number q satisfying $q > 4\widetilde{M_\psi}\widetilde{M_\phi}$.
6) Select a series of integers $r_1, \ldots, r_n$ in the range of $(\widetilde{M_\phi}, q)$ satisfying $2\widetilde{M_\phi} < \min_{k\in[2\widetilde{M_\psi}]} |\mathrm{lift}_q(r_i k)|_{i\in[n]}$. Here we set $\Lambda_i = \{\mathrm{lift}_q(r_i k) | k = 0, \pm 1, \ldots, \pm\widetilde{M_\psi}\}$.
Return to Step 5 and choose a larger q, if it failed to sample such $r_1, \ldots, r_n$.

7) Compute $B_R(\mathbf{x}) = (r_j b_{ij})$ and $C = (pA + B_R) \mod q \in \mathbb{F}_q[\mathbf{x}]^{n\times n}$. Set $T = C^{-1}$ if C is non-singular, go back to Step 3.
8) Set $\Psi_R(\mathbf{x}) = (r_1\psi_1(\mathbf{x}), \ldots, r_n\psi_n(\mathbf{x})) \in \mathbb{Z}[\mathbf{x}]^n$ and $G(\mathbf{x}) = (g_1(\mathbf{x}), \ldots, g_n(\mathbf{x})) = (\Phi(\mathbf{x}) + \Psi_R(\mathbf{x})) \mod q \in \mathbb{F}_q[\mathbf{x}]^n$.
9) Randomly sample a permutation matrix S of size n and compute $F = T \circ G \circ S : \mathbb{Z}^n \to \mathbb{F}_q^n$.
   Secret key: $\Phi(\mathbf{x}) \mod p$, $\{r_i\}_{i\in[n]}$, T and S;
   Public key: p and $F(\mathbf{x})$.

- **Encryption**:
  The following process encrypts a plaintext $\mathbf{m} \in I_p^n$.
  1) Randomly sample a perturbation vector $\mathbf{e} \in I_p^n$.
  2) Compute the ciphertext $\mathbf{c} = F(\mathbf{m}) + \mathbf{e} \in \mathbb{F}_q^n$.

- **Decryption**:
  1) Compute $\mathbf{c}' = (c'_1, \ldots, c'_n) = T^{-1}(\mathbf{c})$.
  2) Find a (unique) $\lambda_i \in \Lambda_i$ such that $|\mathrm{lift}_q(c'_i - \lambda_i)| < \widetilde{M_\phi}$ for all $i \in [n]$. Set $\tilde{c}_i = \mathrm{lift}_q(c'_i - \lambda_i) \in \mathbb{Z}$.
  3) Calculate the solution $\tilde{\mathbf{b}} \in I_p^n$ of the equations $\Phi(\mathbf{x}) \equiv (\tilde{c}_1, \ldots, \tilde{c}_n) \mod p$.
  4) Compute $\mathbf{m}' = S^{-1}(\tilde{\mathbf{b}})$ which matches with the plaintext $\mathbf{m}$.

By invoking a perturbation in encryption, it can shrink the secret parameter n and reduce the key length accordingly. Overall, the performance of the cryptosystem based on *pqe*-method is improved compared to *pq*-method.

The security of both *pq*-method and *pqe*-method can be reduced to the constrained MQ problem, while *pqe*-method can also be reduced to a lattice problem NTRU [12]. The performance can be further improved by using a linear structure instead of a quadratic system. In that case, it mainly executes as a lattice-based PKC with linear polynomial multiplication and further strengthens its security by applying some properties from MPKC.

3 Our Proposals

In this section, we propose three variants based on the *pq*-method and *pqe*-method. Note that the original algorithms are both using quadratic polynomials, while our improved cryptosystems adopt linear polynomial systems. For a positive integer p and a matrix $L \in \mathbb{Z}_p^{n\times m}$, we define $\|L\|_p := \max_{\mathbf{a}\in I_p^n}\{|\mathbf{a} \cdot L|_\infty\}$.

Linear mapping mask. The elements of affine isomorphisms T and S are sampled from $\mathbb{F}_q$ randomly at uniform in the bipolar structure of *pq*-method. As a result, the security will not be reduced if we set the S as an identity map in computing the public key P [29]. Thus, we remain only one map of T where itself is secret. We call T a linear mapping mask to preserve the secret keys from a potential key-recovery attack. In addition, eliminating S may (slightly) reduce the cost of key generation and decryption algorithms.

3.1 A simplified linear version of *pq*-method

Firstly, we propose a variant of the *pq*-based public-key cryptosystem. For the sake of convenience, we call it linear-*pq* algorithm.

- **Key Generation**:
  Let p be a small odd prime number and n be a positive integer.
  1) Randomly sample a linear polynomial system $F(\mathbf{x}) = (f_1(\mathbf{x}), \ldots, f_n(\mathbf{x})) \in \mathbb{F}_p[\mathbf{x}]^n$ where $\mathbf{x} = (x_1, \ldots, x_n)$.
  2) Set $M = \|F\|_p$, and choose integer series of $r_1, \ldots, r_n$ larger than $2M$.
  3) Randomly sample another polynomial system $H(\mathbf{x}) = (h_1(\mathbf{x}), \ldots, h_n(\mathbf{x})) \in \mathbb{F}_p[\mathbf{x}]^n$ where the number of variables is $2n$, i.e. $\mathbf{x} = (x_1, \ldots, x_{2n})$.
  4) Set $L = \|H\|_p$, $r = \max_{i \in [n]}\{r_i\}$ and $q = \texttt{NextPrime}(2rL + 2M)$.
  5) Randomly sample an affine transformation T on $\mathbb{F}_q^n$ such that its inverse can be computed efficiently.
  6) Set
  $$G = [F \mid \mathbf{0}] + \begin{bmatrix} r_1 & & \\ & \ddots & \\ & & r_n \end{bmatrix} H.$$
  and $P = T \circ G : \mathbb{F}_q^{2n} \to \mathbb{F}_q^n$.
  Secret key: F, H, $\{r_i\}_{i \in [n]}$, T;
  Public key: p, q, P.

- **Encryption**:
  Given a plaintext $\mathbf{m} \in I_p^{2n}$, compute the ciphertext $\mathbf{c} = P(\mathbf{m}) \in \mathbb{F}_q^n$.

- **Decryption**:
  1) Compute $\mathbf{c}' = T^{-1}(\mathbf{c})$.
  2) For all $i \in [n]$, find a (unique) set of $\{\alpha_i\}$ $(-M \le \alpha_i \le M)$ and $\{\beta_i\}$ $(-L \le \beta_i \le L)$, such that $c'_i = \alpha_i + r_i\beta_i$.
  3) Calculate the solution $\mathbf{x}_0 = (x_{01}, \ldots, x_{0n})$ of equations $(f_1, \ldots, f_n) = (\alpha_1 \ldots, \alpha_n)$.
  4) Substitute $\mathbf{x}_0 = (x_{01}, \ldots, x_{0n})$ into $\mathbf{c}' = G(\mathbf{x})$ which remains variables of $i \in \{n+1, \ldots, 2n\}$, and calculate a solution $\mathbf{x}_1 = (x_{0,n+1}, \ldots, x_{0,2n})$ such that $c'_i = r_{i-n}\beta_i$. $\mathbf{m}' = (x_{01}, \ldots, x_{0n}, x_{0,n+1}, \ldots, x_{0,2n})$ coincides with the plaintext $\mathbf{m}$.

Correctness. For the sake of convenience, we denote by $[r_j]$ $(j \in [n])$ the matrix constructed with elements in $\{r_j\}$ at Step 6 of Key Generation. In addition, we further medially separate $G = [G_1 \mid G_2]$, $H = [H_1 \mid H_2]$ and $\mathbf{m} = [\mathbf{m}_1 \mid \mathbf{m}_2]$, respectively. Now, we substitute $\mathbf{c} = P(\mathbf{m}) \in \mathbb{F}_q^n$ into $\mathbf{c}' = T^{-1}(\mathbf{c})$. $q = \texttt{NextPrime}(2rL+2M)$ is set large enough to make sure that the elements in G will not change after the substitution, thus we can get $\mathbf{c}' = G(\mathbf{m}) = [G_1 \mid G_2](\mathbf{m}) = [F(\mathbf{m}_1) \mid \mathbf{0}] + [r_j][H_1(\mathbf{m}_1) \mid H_2(\mathbf{m}_2)]$. Since $r_j > 2M$ and $M = \|F\|_p$, there exists only one vector of $\boldsymbol{\alpha} = (\alpha_i)$ corresponding to $F(\mathbf{m}_1) = \boldsymbol{\alpha}$. Namely, at Step 3 of

Decryption, $\mathbf{x}_0 = (x_{01}, \ldots, x_{0n})$ coincides with $\mathbf{m}_1$. Then, we substitute $\mathbf{x}_0$ into $\mathbf{c}'$ which remains $c'_i = [r_j]H_2(\mathbf{x}_1)$ $(i \in \{n+1, \ldots, 2n\})$ to be recovered. Therefore, $\mathbf{x}_1 = (x_{0,n+1}, \ldots, x_{0,2n})$ corresponds to $\mathbf{m}_2$ by solving the linear functions. Finally, the message is correctly recovered by $\mathbf{m}' = (\mathbf{x}_0 \mid \mathbf{x}_1) = \mathbf{m}$.

Discussion. Now we explain the merits derived from linear-*pq* cryptosystem. In general, it is difficult to directly compare the improved proposal with the original one due to different security parameter evaluations. Despite this reality, we can estimate the size of $q = O(n^2p^4)$ in the linear-*pq* cryptosystem, which is attributed to the design of constructing a public key with two polynomial systems where the coefficients' sizes are significantly different. This is intuitively much smaller than $q = O(n^4p^6)$ in the *pq*-method.

Moreover, in the decryption of *pq*-method, we need to solve a quadratic polynomial system of n equations in n variables with integer coefficients. Accordingly, it requires $O(n^4)$ operations in key generation of the original *pq*-method by the state-of-the-art *pq*-TM method. Meanwhile, in the linear-*pq* method, the public key is designed by two linear polynomial systems. The computational cost in the key generation phase is $O(n^2 \log n)$ using the best-known Number-Theoretical Transform (NTT) algorithm for polynomial multiplications.

At the last step of key generation, the remaining T is a linear mapping mask to preserve the secret keys from a potential key-recovery attack. In addition, eliminating S may (slightly) reduce the cost of key generation and decryption algorithms.

The linear-*pq* method is modified obediently from the original *pq*-method. They are deterministic schemes, so they do not hold the security property of indistinguishability under a chosen-plaintext attack (IND-CPA). Namely, the adversary can distinguish the ciphertext c_b easily by re-encrypting the chosen plaintexts m_0 and m_1. Thus, the linear-*pq* method satisfies indistinguishability under onewayness attack (OW-CPA) under the hardness assumption of solving ISIS problems. (The ISIS reduction is explained in section 4.1)

3.2 A linear polynomial version of *pqe*-method

Secondly, we also propose a linear version for *pqe*-method, where the methodology is similar to the linear-*pq* algorithm. We call it linear-*pqe* method.

- **Key Generation**:
  Let p be an odd prime number and n be a positive integer.
  1) Sample matrices $L_{1,X}, L_{1,Y}, L_{r,X}, L_{r,Y} \in \mathbb{F}_p^{n\times n}$ randomly.
  2) Choose positive integers $M_{1,X}, M_{1,Y}, M_{r,X}, M_{r,Y}$ satisfying $\|L_{a,b}\|_p \leq M_{a,b}$ $(a \in \{1, r\}, b \in \{X, Y\})$. Set $M_1 = M_{1,X} + pM_{1,Y}$ and $M_r = M_{r,X} + M_{r,Y}$.
  3) Choose a prime number q satisfying $q > 4M_1M_r$.
  4) Select a series of integers $0 < r_1, \ldots, r_n < q$ and $k \in [2M_r]$ satisfying $2M_1 < \min\limits_{k\in[2M_r]} |\mathrm{lift}_q(r_ik)|_{i\in[n]}$. Return to Step 3 and choose a larger q, if it failed to sample such $r_1, \ldots, r_n$.

5) Compute

$$L_X = L_{1,X} + \begin{bmatrix} r_1 & & \\ & \ddots & \\ & & r_n \end{bmatrix} L_{r,X} \in \mathbb{Z}^{n\times n}$$

and

$$L_Y = pL_{1,Y} + \begin{bmatrix} r_1 & & \\ & \ddots & \\ & & r_n \end{bmatrix} L_{r,Y} \in \mathbb{Z}^{n\times n}.$$

Set $T = L_Y^{-1} \mod q$ if $L_Y \mod q \in \mathbb{F}_q^{n\times n}$ is non-singular, or go back to Step 1.

6) Compute $L_F = T \circ L_X \in \mathbb{F}_q^{n\times n}$ and $L_S = L_{1,X}^{-1} \mod p$.
Secret key: L_S, $\{r_i\}_{i\in[n]}$ and L_Y;
Public key: p, q and L_F.

- **Encryption**:
The following process encrypts a plaintext $\mathbf{m} \in I_p^n$.
1) Randomly sample a perturbation vector $\mathbf{e} \in I_p^n$.
2) Compute the ciphertext $\mathbf{c} = L_F(\mathbf{m}) + \mathbf{e} \in \mathbb{F}_q^n$.

- **Decryption**:
1) Compute $\mathbf{b} = (b_1, \ldots, b_n) = L_Y \cdot \mathbf{c}$.
2) Find a (unique) k_i such that $|\mathrm{lift}_q(b_i - r_i k_i)| \leq M_1$ and $|k_i| \leq M_r$ for all $i \in [n]$. Set $\hat{b}_i = \mathrm{lift}_q(b_i - r_i k_i) \in \mathbb{Z}$.
3) Calculate $\mathbf{u} = \hat{\mathbf{b}} \mod p \in \mathbb{F}_p^n$ and compute $\mathbf{m}' = \mathrm{lift}_p(L_s \cdot \mathbf{u})$ which matches with the plaintext $\mathbf{m}$.

Correctness. First we take off the linear mapping mask by $\mathbf{b} = L_Y \cdot \mathbf{c} = L_Y \circ L_F(\mathbf{m}) + L_Y(\mathbf{e}) = L_Y \circ L_Y^{-1} \circ L_X(\mathbf{m}) + L_Y(\mathbf{e}) = L_X(\mathbf{m}) + L_Y(\mathbf{e}) = L_{1,X}(\mathbf{m}) + [r_i]L_{r,X}(\mathbf{m}) + pL_{1,Y}(\mathbf{e}) + [r_i]L_{r,Y}(\mathbf{e})$. Then, the computation at Step 2 in Decryption extracts items of $\hat{\mathbf{b}} = L_{1,X}(\mathbf{m}) + pL_{1,Y}(\mathbf{e})$ where the bounds of parameters ensure the items unchanged during the execution. Next, $\mathbf{u} = \hat{\mathbf{b}} \mod p \in \mathbb{F}_p^n = L_{1,X}(\mathbf{m})$ eliminates the item of $pL_{1,Y}(\mathbf{e})$. Consequently, message is correctly recovered by $\mathbf{m}' = \mathrm{lift}_p(L_s \cdot \mathbf{u}) = \mathrm{lift}_p(L_{1,X}^{-1} \circ L_{1,X}(\mathbf{m})) = \mathbf{m}$.

Discussion. Note that the boundary setting for q at Step 3 is because of $2M_1 \times 2M_r$ from step 4. At step 4, to make sure the decryption succeed by 100%, it requires $2M_1 < |\mathrm{lift}_q(r_i(k_a - k_b))|$ for $-M_r \leq k_a, k_b \leq M_r$, then we get $2M_1 < \min_{k\in[2M_r]} |\mathrm{lift}_q(r_i k)|_{i\in[n]}$ by setting $k = k_a - k_b \in [2M_r]$.

The advantages of linear-*pqe* algorithm comparing to *pqe*-method are analogous to that of the linear-*pq* method in Section 3.1. Due to the key construction in a linear polynomial system, a smaller modulus $q = O(n^2p^5)$ is available versus $q = O(n^4p^6)$ of the original *pqe*-method. Following the idea of *pqe*-method, we construct the linear mapping mask T in this scheme by computing the inverse of the secret key L_Y. It makes the computation lighter without sampling from $\mathbb{F}_q^n$ and tests its reversibility.

Furthermore, conducting the linear polynomial multiplications costs $O(n^2 \log n)$ operations which is more efficient than the $O(n^4)$ required in *pqe*-method. Besides, eliminating the matrix S may (slightly) speed up the composition of public key L_F in linear-*pqe* method.

3.3 A ring version of linear-*pqe* algorithm

Finally, we propose a ring version of linear-*pqe* algorithm, whose key size is $O(1/n)$ shorter than the other two algorithms. Thus, it derives a much better performance. We call it ring-*pqe* method.

We define a polynomial ring $R := \mathbb{Z}[x]/(x^n - 1)$. For a positive integer p and a polynomial $L \in R$, we define $\|L\|_p := \max_{\mathbf{a} \in I_p^n} \{|\mathbf{a} \cdot L|_\infty\}$. In this subsection, we let I_p^n represent a set of polynomials of $n-1$ degree and coefficients lie in I_p.

- **Key Generation**:
  Let p be an odd prime number and n be a positive integer.
  1) Sample matrices $L_{1,X}, L_{1,Y}, L_{r,X}, L_{r,Y} \in R/pR$ randomly.
  2) Choose positive integers $M_{1,X}, M_{1,Y}, M_{r,X}, M_{r,Y}$ satisfying $\|L_{a,b}\|_p \leq M_{a,b}$ ($a \in \{1, r\}, b \in \{X, Y\}$). Set $M_1 = M_{1,X} + pM_{1,Y}$ and $M_r = M_{r,X} + M_{r,Y}$.
  3) Choose a prime number q satisfying $q > 4M_1M_r$.
  4) Select an integer r in $(0, q)$ which satisfies $2M_1 < \min_{k \in [2M_r]} |\text{lift}_q(rk)|_{i \in [n]}$. Return to Step 3 and choose a larger q, if it fails to sample such r.
  5) Compute $L_X = L_{1,X} + rL_{r,X}$ and $L_Y = pL_{1,Y} + rL_{r,Y}$ Set $T = L_Y^{-1} \mod q$ if $L_Y \mod q \in R/qR$ is non-singular, or go back to Step 1.
  6) Compute $L_F = T \cdot L_X \in R/qR$ and $L_S = L_{1,X}^{-1} \mod p$.
  Secret key: L_S, r and L_Y;
  Public key: p, q and L_F.

- **Encryption**:
  The following process encrypts a plaintext $\mathbf{m} \in I_p^n$.
  1) Randomly sample a perturbation vector $\mathbf{e} \in I_p^n$.
  2) Compute the ciphertext $\mathbf{c} = L_F(\mathbf{m}) + \mathbf{e} \in R/qR$.

- **Decryption**:
  1) Compute $b(\mathbf{x}) = \sum_{i=0}^{n-1} b_i x^i = L_Y \cdot \mathbf{c}$.
  2) Find a (unique) k_i such that $|\text{lift}_q(b_i - rk_i)| \leq M_1$ and $|k_i| \leq M_r$ for all $i \in [n]$. Set $\hat{b}_i = \text{lift}_q(b_i - rk_i) \in \mathbb{Z}$.
  3) Calculate $\mathbf{u} = \hat{\mathbf{b}} \mod p \in \mathbb{F}_p^n$ and compute $\mathbf{m}' = \text{lift}_p(L_S \cdot \mathbf{u})$ which matches with the plaintext $\mathbf{m}$.

Correctness. We omit the proof here since it is similar to what for the above linear-*pqe* method.

Discussion. The ring structure results in an overwhelming reduction on both the public key size and the secret key size, by the multiplicative factor $1/n$. Thus the ring-*pqe* crypto scheme outperforms the linear-*pqe* method. Furthermore, as the same as linear-*pqe* method, we design the linear mapping mask T in ring-*pqe* by computing from part of the secret key L_Y inside.

4 Cryptanalysis and Performance

In this section, we first evaluate proper security parameters for linear-*pq* cryptosystem, linear-*pqe* cryptosystem, and ring-*pqe* cryptosystem, respectively. We consider the security levels of AES-128 in the NIST PQC standardization project [1] that the brute force attack on AES key search requires at least 2^{143} classical computing gates. Then we show the key and ciphertext sizes with respect to the above parameters for each scheme. Finally, we show the experimental results of the three cryptosystems.

4.1 Evaluating security parameters and key size

To evaluate a proper parameter set of (n, p, q), we consider that the security of proposed cryptosystems is based on relevant lattice problems. Namely, we can see the encryption procedure of linear-*pq* $\mathbf{c} = P(\mathbf{m}) = \mathbf{m} \cdot P \in \mathbb{F}_q^n$ as an ISIS instance since the norm of $\mathbf{m} \in I_p^{2n}$ is much smaller than that of the vector in the kernel space of $L(P)$ in $\mathbb{Z}^{2n}$ computed by the Gaussian heuristic. Simultaneously, we can regard the encryption in linear-*pqe* and ring-*pqe* as dealing with the LWE instance and the Ring-LWE instance, respectively. As discussed in the above proposals, a key recovery attack is not feasible owing to the subtle linear mapping mask. Hence, we apply the message recovery attacks using the lattice method against each problem. Simultaneously, we also consider the exhaustive search for the message recovery attack.

Message recovery attack. Typically the LWE problem and the Ring-LWE problem can be reduced to the SVP or uSVP using Bai-Galbraith's embedding technique [7]. Indeed, the cryptanalysis for linear-*pq*, linear-*pqe* and ring-*pqe* schemes are equivalent to evaluating the hardness of SVP in $(2n+1)$-dimensional lattices with volume of $q^{(n+1)}$. Refer to [23] for a detailed application and analysis of Bai-Galbraith's embedding technique.

Moreover, the lattice algorithms are also used in cryptanalysis. One of the best-known lattice algorithms is *BKZ algorithm* [18] and its variants [6,9,25,26], which processes the given basis until being almost β-reduced. In other words, the projected lengths of each basis vector are the shortest ones in the relative β-dimensional sub-lattice. BKZ costs exponentially in the blocksize β. In 2001, Ajtai et al. proposed a sieving algorithm to solve SVP [3]. It requires a running time of $2^{0.52n+o(n)}$ in dimension n and requires exponential storage of $2^{0.2n+o(n)}$ as well. For a β-dimensional sub-lattice the cost of sieving algorithm can be estimated in $2^{0.292\beta+o(\beta)}$ operations. If we take sieving as a subroutine in the β-dimensional sub-lattices inside of an n-dimensional lattice, the total BKZ-β cost can be estimated by $8n \cdot 2^{0.292\beta+12.31}$ operations [4]. We recall the following two definitions to evaluate the performance of lattice algorithms.

(a) The *root Hermite factor* [10] is defined as:

$$\delta = \mathrm{rHF}(\mathbf{b}_1, \ldots, \mathbf{b}_n) = (\|\mathbf{b}_1\|/\mathrm{Vol}(L)^{1/n})^{1/n}.$$

schemes	n	p	q
linear-pq	980	3	61473439 (26 bits)
linear-pqe	1022	3	133693951 (27 bits)
ring-pqe	1022	3	133693951 (27 bits)

Table 1. Parameter choice for AES-128 bit security.

The rHF of a BKZ-β reduced basis $\mathbf{B}$ of d-dimensional lattice $L(\mathbf{B})$ can be evaluated by

$$\delta = (((\pi\beta)^{1/\beta}\beta)/(2\pi e))^{\frac{1}{2(\beta-1)}}. \tag{3}$$

This is proposed and practically verified by Chen in [8].

(b) In order to estimate the hardness of LWE samples $(\mathbf{A}, \mathbf{b} \equiv \mathbf{As} + \mathbf{e} \pmod q) \in \mathbb{Z}_q^{m\times n} \times \mathbb{Z}_q^m$, the *2016 estimate* [5] states that if the Gaussian heuristic and the GSA hold for BKZ-β reduced basis and

$$\sqrt{\beta/d} \cdot \|(\mathbf{e}|1)\| \approx \sqrt{\beta}\sigma \leq \delta^{2\beta-d} \cdot \mathrm{Vol}(L_{(\mathbf{A},q)})^{1/d}, \tag{4}$$

then error $\mathbf{e}$ can be found by the BKZ-β reduction algorithm. It has been widely used in the cryptanalysis [24] for lattice-based cryptoschemes: given the dimension d, the modular q and the standard deviation σ of $\mathbf{e}_i$'s distribution, the 2016 estimate can output the optimal blocksize β in the attack by using equations (3) (4).

Exhaustive search.

For a ciphertext $\mathbf{c}$, the complexity of finding the solution of $P(\mathbf{m}) = \mathbf{c}$ in linear-pq and $L_F(\mathbf{m}) + \mathbf{e} = \mathbf{c}$ in the ring-pq by the exhaustive search is the same of $O(p^{2n})$. In the case of using Grover's quantum search algorithm, the complexity is $O(p^n)$.

Parameter suggestion and key sizes. Due to NIST's call for proposal in the PQC standardization project [1], any attack on AES-128 bit security requires at least 143 bits of classical gate operations. We evaluate the relevant parameters for AES-128 bit security in Table 1. There is no significant difference but just within one bit for the parameter sizes among different schemes.

schemes	pk(kB)	sk(kB)	ct(kB)
linear-pq	6242.6	3844.8	3.1
linear-pqe	3525.1	3789.7	3.5
ring-pqe	3.5	3.7	3.5

Table 2. The sizes of public key (pk), secret key (sk), and ciphertext (ct) for each scheme of AES-128 bit security.

Furthermore, we show the key sizes and ciphertext sizes in each proposed cryptosystem in Table 2. It shows that the ciphertext sizes are close to each other, while the key sizes of ring-*pq* are evidently $1/n$ of the other two schemes. This results in a comparatively high efficiency shown in the next section.

4.2 Implementation

schemes	`KeyGen(ms)`	`Enc(ms)`	`Dec(ms)`
linear-*pq*	21624.4	113.7	48596.6
linear-*pqe*	54408.5	64.1	161.9
ring-*pqe*	37.2	0.6	20.2

Table 3. The performance of each scheme with the unit of a millisecond (ms).

We implemented the three proposed crypto schemes in C++ language. In our implementation, we invite the number theory library (NTL) [20]. In particular, the number-theoretical transform (NTT) technique is conducted in NTL, which can speed up the polynomial multiplications over finite fields. Then we run 1,000 experiments on a computer with Intel Core i9 @ 3.6 GHz CPU, g++ version 7.4.0. We evaluate the average running time for each one in Table 3 with one decimal precision. It shows that the ring-*pqe* algorithm is overwhelmingly efficient compared to the other two schemes.

5 Conclusion

In this paper, we proposed three PKE algorithms based on linear-*pq*, linear-*pqe*, and ring-*pqe* methods, respectively. Compared to the original algorithms by Yasuda, our proposals use a much smaller modulus q and cost less by conducting linear polynomial multiplications. Besides, our schemes are secure against the key-recovery attack by invoking a linear mapping mask at the end of key generations. Furthermore, we evaluated the proper parameters for AES-128 bit security level and assessed the key size produced in our cryptosystems. In particular, the linear-*pqe* cryptosystem outperforms the other two algorithms regarding key size and practical efficiency.

Acknowledgement. We thank Dr. Atsushi Takayasu for his helpful comments on this work. This work was supported by JSPS KAKENHI Grant Number JP20K23322, JP21K11751, JP19K20266, JP20K03741, Japan. This work is based on the discussions at FY2019 IMI Joint Usage Research Program Short-term Joint Research "New Development of Constructing Next-Generation Cryptography via Unified Approaches of Mathematics Theory, Computation and Cryptology".

References

1. US Department of Commerce, National Institute of Standards and Technology. Post-Quantum Cryptography, 2020. `https://csrc.nist.gov/projects/post-quantum-cryptography/`.
2. PQC Standardization Process: Fourth Round Candidate Announcement, 2022. `https://csrc.nist.gov/News/2022/pqc-candidates-to-be-standardized-and-round-4`.
3. M. Ajtai, R. Kumar, and D. Sivakumar. A sieve algorithm for the shortest lattice vector problem. In *Proceedings of the Thirty-third Annual ACM Symposium on Theory of Computing*, pages 601–610, 2001.
4. M. R. Albrecht, R. Player, and S. Scott. On the concrete hardness of learning with errors. *Journal of Mathematical Cryptology*, 9(3):169–203, 2015.
5. E. Alkim, L. Ducas, T. Pöppelmann, and P. Schwabe. Post-quantum key exchange-a new hope. In *USENIX Security Symposium*, pages 327–343, 2016.
6. Y. Aono, Y. Wang, T. Hayashi, and T. Takagi. Improved progressive BKZ algorithms and their precise cost estimation by sharp simulator. In *Advances in Cryptology - EUROCRYPT 2016 - 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Proceedings, Part I*, pages 789–819, 2016.
7. S. Bai and S. D. Galbraith. Lattice decoding attacks on binary LWE. In *Information Security and Privacy - 19th Australasian Conference, ACISP 2014, Proceedings*, pages 322–337, 2014.
8. Y. Chen. Lattice reduction and concrete security of fully homomorphic encryption. *Dept. Informatique, ENS, Paris, France, PhD thesis*, 2013.
9. Y. Chen and P. Q. Nguyen. Bkz 2.0: Better lattice security estimates. In *Advances in Cryptology – ASIACRYPT 2011: 17th International Conference on the Theory and Application of Cryptology and Information Security, Proceedings*, pages 1–20, 2011.
10. N. Gama and P. Q. Nguyen. Predicting lattice reduction. In *Advances in Cryptology - EUROCRYPT 2008, 27th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Proceedings*, pages 31–51, 2008.
11. M. R. Garey and D. S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman, 1979.
12. J. Hoffstein, J. Pipher, and J. H. Silverman. NTRU: A ring-based public key cryptosystem. In *Algorithmic Number Theory, Third International Symposium, ANTS-III, Proceedings*, pages 267–288, 1998.
13. Y. Ikematsu, R. A. Perlner, D. Smith-Tone, T. Takagi, and J. Vates. HFERP - A new multivariate encryption scheme. In T. Lange and R. Steinwandt, editors, *Post-Quantum Cryptography - 9th International Conference, PQCrypto 2018, Fort Lauderdale, FL, USA, April 9-11, 2018, Proceedings*, volume 10786 of *Lecture Notes in Computer Science*, pages 396–416. Springer, 2018.
14. V. Lyubashevsky, C. Peikert, and O. Regev. On ideal lattices and learning with errors over rings. In *Advances in Cryptology - EUROCRYPT 2010, 29th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Proceedings*, pages 1–23, 2010.
15. T. Matsumoto and H. Imai. Public quadratic polynominal-tuples for efficient signature-verification and message-encryption. In C. G. Günther, editor, *Advances in Cryptology - EUROCRYPT '88, Workshop on the Theory and Application of of Cryptographic Techniques, Davos, Switzerland, May 25-27, 1988, Proceedings*, volume 330 of *Lecture Notes in Computer Science*, pages 419–453. Springer, 1988.

16. J. Patarin. Cryptanalysis of the matsumoto and imai public key scheme of eurocrypt'88. In D. Coppersmith, editor, *Advances in Cryptology - CRYPTO '95, 15th Annual International Cryptology Conference, Santa Barbara, California, USA, August 27-31, 1995, Proceedings*, volume 963 of *Lecture Notes in Computer Science*, pages 248–261. Springer, 1995.
17. O. Regev. On lattices, learning with errors, random linear codes, and cryptography. In *Proceedings of the 37th Annual ACM Symposium on Theory of Computing*, pages 84–93, 2005.
18. C. Schnorr and M. Euchner. Lattice basis reduction: Improved practical algorithms and solving subset sum problems. *Math. Program.*, 66:181–199, 1994.
19. P. W. Shor. Algorithms for quantum computation: Discrete logarithms and factoring. In *35th Annual Symposium on Foundations of Computer Science, Santa Fe, Proceeding*, pages 124–134, 1994.
20. V. Shoup. NTL, a library for doing number theory. Available at `http://www.shoup.net/ntl/`, 2017.
21. A. Szepieniec, J. Ding, and B. Preneel. Extension field cancellation: A new central trapdoor for multivariate quadratic systems. In T. Takagi, editor, *Post-Quantum Cryptography - 7th International Workshop, PQCrypto 2016, Fukuoka, Japan, February 24-26, 2016, Proceedings*, volume 9606 of *Lecture Notes in Computer Science*, pages 182–196. Springer, 2016.
22. C. Tao, A. Diene, S. Tang, and J. Ding. Simple matrix scheme for encryption. In P. Gaborit, editor, *Post-Quantum Cryptography - 5th International Workshop, PQCrypto 2013, Limoges, France, June 4-7, 2013. Proceedings*, volume 7932 of *Lecture Notes in Computer Science*, pages 231–242. Springer, 2013.
23. W. Wang, Y. Wang, A. Takayasu, and T. Takagi. Estimated cost for solving generalized learning with errors problem via embedding techniques. In *Advances in Information and Computer Security - 13th International Workshop on Security, IWSEC 2018, Sendai, Japan, September 3-5, 2018, Proceedings*, pages 87–103, 2018.
24. Y. Wang, Y. Aono, and T. Takagi. Hardness evaluation for search LWE problem using progressive BKZ simulator. *IEICE Transactions*, 101-A(12):2162–2170, 2018.
25. Y. Wang and T. Takagi. Studying lattice reduction algorithms improved by quick reordering technique. *Int. J. Inf. Sec.*, 20(2):257–268, 2021.
26. K. Yamamura, Y. Wang, and E. Fujisaki. Improved lattice enumeration algorithms by primal and dual reordering methods. In J. H. Park and S. Seo, editors, *Information Security and Cryptology - ICISC 2021 - 24th International Conference, Seoul, South Korea, December 1-3, 2021, Revised Selected Papers*, volume 13218 of *Lecture Notes in Computer Science*, pages 159–174. Springer, 2021.
27. T. Yasuda. Multivariate encryption schemes based on the constrained MQ problem. In J. Baek, W. Susilo, and J. Kim, editors, *Provable Security - 12th International Conference, ProvSec 2018, Jeju, South Korea, October 25-28, 2018, Proceedings*, volume 11192 of *Lecture Notes in Computer Science*, pages 129–146. Springer, 2018.
28. T. Yasuda. Multivariate public key system using noise. In *SCIS 2020*, 2020.
29. T. Yasuda, Y. Wang, and T. Takagi. Multivariate encryption schemes based on polynomial equations over real numbers. In J. Ding and J. Tillich, editors, *Post-Quantum Cryptography - 11th International Conference, PQCrypto 2020, Paris, France, April 15-17, 2020, Proceedings*, volume 12100 of *Lecture Notes in Computer Science*, pages 402–421. Springer, 2020.