mathematics

MDPI

Article

Flexible and Compact MLWE-Based KEM

Wenqi Liang [1], Zhaoman Liu [2], Xuyang Zhao [2], Yafang Yang [2,*] and Zhichuang Liang [2,*]

[1] School of Computer Engineering and Science, Shanghai University, Shanghai 200444, China
[2] School of Computer Science, Fudan University, Shanghai 200433, China
* Correspondence: yafangyang18@fudan.edu.cn (Y.Y.); zcliang21@m.fudan.edu.cn (Z.L.)

Abstract: In order to resist the security risks caused by quantum computing, post-quantum cryptography (PQC) has been a research focus. Constructing a key encapsulation mechanism (KEM) based on lattices is one of the promising PQC routines. The algebraically structured learning with errors (LWE) problem over power-of-two cyclotomics has been one of the most widely used hardness assumptions for lattice-based cryptographic schemes. However, power-of-two cyclotomic rings may be exploited in the inflexibility of selecting parameters. Recently, trinomial cyclotomic rings of the form $\mathbb{Z}_q[x]/(x^n - x^{n/2} + 1)$, where $n = 2^k 3^l$, $k \geq 1, l \geq 0$, have received widespread attention due to their flexible parameter selection. In this paper, we propose Tyber, a variant scheme of the NIST-standardized KEM candidate Kyber over trinomial cyclotomic rings. We provide three parameter sets, aiming at the quantum security of 128, 192, and 256 bits (actually achieving 129, 197, and 276 bits) with matching and negligible error probabilities. When compared to Kyber, our Tyber exhibits stronger quantum security, by 22, 31, and 44 bits, than Kyber for three security levels.

Keywords: lattice-based cryptography; post-quantum cryptography; module learning with errors; Kyber; trinomial cyclotomics

MSC: 94A60

Citation: Liang, W.; Liu, Z.; Zhao, X.; Yang, Y.; Liang, Z. Flexible and Compact MLWE-Based KEM. *Mathematics* **2024**, *12*, 1769. https://doi.org/10.3390/math12111769

Academic Editor: Jonathan Blackledge

Received: 16 May 2024
Revised: 31 May 2024
Accepted: 4 June 2024
Published: 6 June 2024

1. Introduction

If practical quantum computers are ever built, the current public-key cryptography, which relies heavily on the hardness assumptions of factoring integers and solving discrete logarithms, will be vulnerable to quantum attacks. Given the escalating risks posed by quantum computing in recent years, the crypto community has shifted its research focus towards post-quantum cryptography (PQC). Constructing cryptographic schemes based on lattices is one of the promising PQC routines. It has driven several nations to launch professional organizations to start the standardizations of PQC schemes.

In 2016, the US National Institute of Standards and Technology (NIST) initiated a standardization competition for post-quantum cryptography primitives, including public-key encryption (PKE), key encapsulation mechanisms (KEMs), and digital signatures. Notably, lattice-based schemes occupied a significant portion of the submissions, accounting for 26 out of 64 in the initial round [1], 12 out of 26 in the second round [2], and ultimately, 7 out of 15 in the third round [3]. In 2022, NIST finally selected lattice-based schemes named Kyber [4] (official name is ML-KEM [5]) and Dilithium [6] (official name is ML-DSA [7]) as the standardized candidates [8].

The Chinese Association for Cryptologic Research (CACR) also initiated a PQC competition to standardize PQC schemes between 2018 and 2019. In the second round of the Chinese National cryptographic algorithms design contest, lattice-based schemes accounted for 11 out of 14 among public-key schemes [9].

Most of these lattice-based schemes are "small lattice systems", which are based on algebraically structured lattices, such as ideal lattices and module lattices, with polynomial

rings as their underlying algebraic structures. The most common one is the cyclotomic ring $\mathbb{Z}[x]/(\Phi_m(x))$, where $\Phi_m(x)$ is defined as the m-th cyclotomic polynomial.

For the lattice-based schemes, the learning with error (LWE) problem [10] is one of the most common hardness assumptions to construct public-key encryption or key encapsulation mechanisms. But for those "small lattice systems", they are based on variants of LWE, which are over cyclotomic rings $\mathcal{R} = \mathbb{Z}[\xi_m] \cong \mathbb{Z}[x]/(\Phi_m(x))$, where $\xi_m = \exp(\frac{2\pi i}{m})$ is an m-th root of unity, e.g., a ring learning with error (RLWE) problem [11] or module learning with error (MLWE) problem [12]. The most popular cyclotomic polynomial used in lattice-based crypto is the power-of-two cyclotomic polynomial: $\Phi_m(x) = x^n + 1$, where $m = 2^{e+1}$ and $n = \phi(m) = 2^e$ are power-of-two integers, and ϕ is the Euler function. At this time, its corresponding cyclotomic ring is $\mathbb{Z}[x]/(x^n + 1)$. In fact, the analysis in [11,12] is mainly in the case of $\mathbb{Z}[x]/(x^n + 1)$. Through the NIST round 3, Kyber [4], Saber [13], and Dilithium [6] use $\mathbb{Z}[x]/(x^{256} + 1)$ as their underlying polynomial ring. There are some advantages of choosing power-of-two cyclotomic rings. (1) They are simple but useful: $x^n + 1$, where n is a power of two, is one of the simplest cyclotomic rings. And $\mathbb{Z}[x]/(x^n + 1)$ is one of the best understood and the most widely studied cyclotomic rings in algebraic number theory, and there are no improved attacks that have been proposed against the schemes based on {R,M}LWE over $\mathbb{Z}[x]/(x^n + 1)$. (2) Most {R,M}LWE-based schemes use suitable parameters such that number theoretic transform (NTT) can be utilized to compute the polynomial multiplication in $\mathbb{Z}_q[x]/(x^n + 1)$. As we know, NTT-based schemes are very efficient due to the remarkable memory efficiency and speed of NTT, outperforming any other algorithm for multiplication in polynomial rings.

However, some disadvantages cannot be ignored in their practical application. The main focus should be on the inflexibility of selecting parameters. Take RLWE-based schemes as an example. The security level is directly influenced by the ring dimension n of RLWE-based schemes. Since n is a power of two, to achieve a higher security level, it is inconvenient to find a polynomial of some particular degree up to the next power of two. To reach 128-bit security, the ring dimension n should be somewhere around 700 [14]. There are two power-of-two integers: 512 and 1024 which are close to 700, but the former integer leads to insufficient security and the latter one leads to redundant security.

A natural question to ask in this point is as follows.

Motivating question 1: Are there ever flexible ways to use other cyclotomic rings rather than power-of-two cyclotomic rings?

Considering 128-bit security in the post-quantum era, it is interesting but meaningful to be able to construct lattice-based schemes over other cyclotomic rings as alternatives. For motivating question 1, the answer to the question is affirmative. The work in [15] shows that for any cyclotomic polynomial $\Phi_m(x)$, RLWE can work entirely in the ring $\mathbb{Z}[x]/(\Phi_m(x))$. There also have been some schemes using trinomial cyclotomic rings. For example, Falcon Round 1 used $\mathbb{Z}[x]/(x^n - x^{n/2} + 1)$, where $n = 3 \cdot 2^e$ [16]. NewHope-Compact, an RLWE-based scheme [17], and NTTRU, an NTRU-like scheme [18], use $\mathbb{Z}_q[x]/(x^{768} - x^{384} + 1)$ with a prime q. Scabbard applies $\mathbb{Z}_q[x]/(x^{768} - x^{384} + 1)$ with a power-of-two q due to its hardness of ring learning with rounding (RLWR) [19]. Later, the work in [14] instantiated NTRU over some trinomial cyclotomic rings of the form $\mathbb{Z}_q[x]/(x^n - x^{n/2} + 1)$ with various n in order to select flexible parameters. The fact is that $x^n - x^{n/2} + 1$ is the $3n$-th cyclotomic ring if n is of the form $n = 2^k 3^l$, $k \geq 1, l \geq 0$.

There is a gap for schemes based on module lattices, especially MLWE-based schemes. One exception is that the work in [20] provided a variant scheme of Kyber; however, over power-of-three cyclotomic rings. Actually, no one has applied trinomial cyclotomics to MLWE-based schemes. Undoubtedly, MLWE-based schemes take into account the security of LWE-based schemes and the efficiency of RLWE-based schemes. Therefore, there will be a balance between security and efficiency by adjusting the parameters. Changing the sampling number k is a major way to achieve different security levels for MLWE-based schemes. But, the increase in k will lead to a more complex implementation. In addition, $\mathbb{Z}[x]/(x^n + 1)$ is still widely used in MLWE-based schemes. For example, Kyber,

an outstanding representative of MLWE-based schemes, and the only NIST-standardized KEM candidate, is based on the power-of-two cyclotomic ring $\mathbb{Z}_{3329}[x]/(x^{256}+1)$. Kyber's supporting documentation has mentioned that "One could consider using Kyber with a ring that is not $\mathbb{Z}[x]/(x^n+1)$", as $\mathbb{Z}[x]/(x^n+1)$ may be exploited in the inflexibility of selecting parameters. Such a sentence is also applicable to other MLWE-based schemes. Hence, it leads to the following question.

Motivating question 2: Could we extend the known power-of-two MLWE-based schemes (e.g., Kyber) to the cases over trinomial cyclotomic rings, with appropriate selection of parameters so as to achieve a practical security level and matching error probabilities?

We answer motivating question 2 in the affirmative by proposing a variant scheme of Kyber, named Tyber, which is constructed over trinomial cyclotomic rings $\mathbb{Z}_q[x]/(x^n-x^{n/2}+1)$, where n is a positive integer of the form $n=2^k3^l$, with $k\geq 1, l\geq 0$ in this paper. The modulus q is chosen as a prime number, in order to be suitable for NTT. The security level of our Tyber is aimed at NIST security levels I, III, and V, while it can also achieve negligible error probabilities.

1.1. Related Works

There is a line of recent works that use trinomial cyclotomic rings of the form $\mathbb{Z}[x]/(x^n-x^{n/2}+1)$. Table 1 shows their detailed descriptions.

Table 1. Details of related works.

References	Polynomial Rings	Cryptographic Primitives	Hardness Assumption
[16]	$\mathbb{Z}_{18433}[x]/(x^{768}-x^{384}+1)$	Digital signature	NTRU
[21,22]	$\mathbb{Z}_q[x]/(x^n-x^{n/2}+1), n\in\{648,768,864,972\}$	Digital signature	NTRU
[18]	$\mathbb{Z}_{7681}[x]/(x^{768}-x^{384}+1)$	PKE/KEM	NTRU
[14]	$\mathbb{Z}_q[x]/(x^n-x^{n/2}+1), n=2^k3^{l-1}, k\geq 1, l\geq 0$	PKE/KEM	NTRU
[23]	$\mathbb{Z}_q[x]/(x^n-x^{n/2}+1), n=2^k3^{l-1}, k\geq 1, l\geq 0$	PKE/KEM	NTRU
[24]	$\mathbb{Z}_q[x]/(x^n-x^{n/2}+1), n=2^k3^{l-1}, k\geq 1, l\geq 0$	PKE/KEM	module-NTRU
[17]	$\mathbb{Z}_{3457}[x]/(x^{768}-x^{384}+1)$	PKE/KEM	RLWE
[25]	$\mathbb{Z}_{7681}[x]/(x^{768}-x^{384}+1)$	PKE/KEM	RLWE
[19]	$\mathbb{Z}_{1024}[x]/(x^{768}-x^{384}+1)$	PKE/KEM	RLWR
This work	$\mathbb{Z}_q[x]/(x^n-x^{n/2}+1), n=2^k3^{l-1}, k\geq 1, l\geq 0$	PKE/KEM	MLWE

The early version of Falcon, i.e., Falcon Round 1 [16], used $\mathbb{Z}_{18433}[x]/(x^{768}-x^{384}+1)$ for its parameter set of $n=768$. Later, Espitau et al. [21] proposed Mitaka, which is a simpler, parallelizable and maskable variant of Falcon, and its underlying polynomial rings include trinomial cyclotomic rings. Then, the Gaussian sampling and smoothing parameters of Mitaka were studied and optimized in subsequent work [22]. Lyubashevsky and Seiler [18] proposed a variant of NTRU, named NTTRU, by offering a new ring structure $\mathbb{Z}_{7681}[x]/(x^{768}-x^{384}+1)$. There have even been further improvements since then. Duman et al. [14] extended the rings $\mathbb{Z}_q[x]/(x^n-x^{n/2}+1)$ with various n in order to select flexible parameter sets. Additionally, Liang et al. [23] proposed compact and efficient NTRU-based KEMs over trinomial cyclotomic rings with the aid of lattice-based error correction codes. Recently, Bai et al. [24] designed compact PKEs based on the module-NTRU hardness assumption over trinomial cyclotomic rings. As for RLWE-based schemes, Alkim et al. [17] improved NewHope and presented NewHope-Compact by offering a parameter set for NIST security level III, over the trinomial cyclotomic ring $\mathbb{Z}_{3457}[x]/(x^{768}-x^{384}+1)$. Similarly, Liang et al. [25] proposed NewHope-Unified, which used $\mathbb{Z}_{7681}[x]/(x^{768}-x^{384}+1)$ as its underlying ring for $n=768$. This can be extended to the case of RLWR-based schemes. For example, Bermudo Mera et al. [19] introduced a suite of post-quantum KEMs, named Scabbard, and it contained an RLWR-based KEM applying $\mathbb{Z}_{1024}[x]/(x^{768}-x^{384}+1)$.

1.2. Our Contributions

We propose Tyber, a variant scheme of Kyber over trinomial cyclotomic rings of the form $\mathbb{Z}_q[x]/(x^n - x^{n/2} + 1)$, where $n = 2^k 3^l$, $k \geq 1, l \geq 0$. Our Tyber includes an IND-CPA secure public key encryption and an IND-CCA secure key encapsulation mechanism. The parameter sets of Tyber are provided, featuring quantum security of 128, 192, and 256 bits (actually achieving 129, 197, and 276 bits) with matching and negligible error probabilities. When compared to Kyber, our Tyber exhibits stronger quantum security, by 22, 31, and 44 bits, than Kyber for three security levels. All analysis and conclusions in this paper can be extended to any other power-of-two MLWE-based schemes.

2. Preliminaries

2.1. Notation and Definitions

Let $\mathbb{Z}$ represent the ring of rational integers, with n and q being positive integers. We define $\mathbb{Z}_q$ as the quotient ring $\mathbb{Z}/q\mathbb{Z}$ and it comprises the set $\{0, 1, \ldots, q-1\}$. Furthermore, we denote $\mathbb{Z}_q^\times$ as the group of invertible elements within $\mathbb{Z}_q$. For a given real number x, we use the notation $\lceil x \rfloor$ to represent the integer closest to x. Additionally, we introduce the notation $\mathcal{R}$ for the ring $\mathbb{Z}[x]/(x^n - x^{n/2} + 1)$ and $\mathcal{R}_q$ for the quotient ring $\mathbb{Z}_q[x]/(x^n - x^{n/2} + 1)$. Elements in $\mathcal{R}$ or $\mathcal{R}_q$ are polynomials, denoted by regular font letters, such as f, g, v. All the vectors in this paper are column vectors by default. Bold lowercase letters represent polynomial vectors over $\mathcal{R}$ or $\mathcal{R}_q$ while bold uppercase letters are polynomial matrices. For example, $\mathbf{v}$ and $\mathbf{A}$, whose transposes are denoted by $\mathbf{v}^T$ and $\mathbf{A}^T$, respectively. A polynomial f in $\mathcal{R}$ (or $\mathcal{R}_q$) has two equivalent representations: a power series form $f = \sum_{i=0}^{n-1} f_i x^i$ and a column vector form $f = (f_0, f_1, \ldots, f_{n-1})^T$, where $f_i \in \mathbb{Z}$ (or $f_i \in \mathbb{Z}_q$) for $i = 0, 1, \ldots, n-1$. A function $\epsilon : \mathbb{N} \to [0, 1]$ is said to be negligible if it satisfies $\epsilon(\lambda) < 1/\lambda^c$ for any positive c and sufficiently large λ. Such a function is denoted by *negl*.

Cyclotomics. Additional information regarding cyclotomic polynomials is available in [26]. Given a positive integer m, the m-th root of unity is denoted by $\xi_m = \exp\left(\frac{2\pi i}{m}\right)$. The m-th cyclotomic polynomial, labeled $\Phi_m(x)$, is expressed as $\Phi_m(x) = \prod_{j=1,\gcd(j,m)=1}^{m} (x - \xi_m^j)$. This type of polynomial is monic, irreducible, and has a degree of $n = \varphi(m)$ over the polynomial ring $\mathbb{Z}[x]$, where φ represents the Euler function. The m-th cyclotomic field is $\mathbb{Q}(\xi_m) \cong \mathbb{Q}[x]/(\Phi_m(x))$, with its associated ring of integers being $\mathbb{Z}[\xi_m] \cong \mathbb{Z}[x]/(\Phi_m(x))$. Some important types of cyclotomic polynomials are mentioned in this paper: (1) Power-of-two cyclotomic polynomials $\Phi_m(x) = x^n + 1$ with m of the form $m = 2^k, k \geq 1$ and $n = \varphi(m) = m/2$; (2) Trinomial cyclotomic polynomials with m of the form $m = 2^k 3^l, k, l \geq 1$ and $n = \varphi(m) = m/3$.

Modular reductions. Let α be a positive integer. We define the modulo operation with signed remainder as follows. For even α, $r' = r \bmod^{\pm} \alpha$ represents the unique element in the range $-\frac{\alpha}{2} < r' \leq \frac{\alpha}{2}$ satisfying $r' \equiv r \pmod{\alpha}$. For odd α, $r' = r \bmod^{\pm} \alpha$ represents the unique element in the range $-\frac{\alpha-1}{2} \leq r' \leq \frac{\alpha-1}{2}$ satisfying $r' \equiv r \pmod{\alpha}$. For any α, $r' = r \bmod^{+} \alpha$ represents the unique element in the range $0 \leq r' < \alpha$ satisfying $r' \equiv r \pmod{\alpha}$. It is simply written as $r \bmod \alpha$ if the exact representation is not important.

Sizes of elements. For any element w in the ring $\mathbb{Z}_q$, $\|w\|_\infty$ represents $|w \bmod^{\pm} q|$. We define the ℓ_∞ norm and the ℓ_2 norm for any vector $w \in \mathcal{R}$ as follows: the ℓ_∞ norm is given by $\max_i |w_i|$, while the ℓ_2 norm is computed as $\sqrt{\sum_{i=0}^{n-1} \|w_i\|_\infty^2}$. Furthermore, for a vector $\mathbf{w} = (w_1, \ldots, w_k) \in \mathcal{R}^k$, we introduce the ℓ_∞ norm as $\max_i \|w_i\|_\infty$ and the ℓ_2 norm as $\sqrt{\sum_{i=1}^{k} \|w_i\|_\infty^2}$.

Sets and distributions. For a given set D, we utilize the notation $x \xleftarrow{\$} D$ to indicate that x is sampled uniformly from D. Furthermore, when referring to a probability distribution Ψ, the notation $x \leftarrow \Psi$ signifies that x is selected in accordance with the distribution Ψ. The centered binomial distribution B_η, parameterized by a positive integer η, is defined as follows: Sample $(a_1, \ldots, a_\eta, b_1, \ldots, b_\eta)$ uniformly from $\{0, 1\}^{2\eta}$ and output the

sum $\sum_{i=1}^{\eta}(a_i - b_i)$. The distribution $\bar{B}_\eta$ is defined as $B_\eta \bmod^{\pm} 3$. Sampling a polynomial $v \leftarrow \Psi$ or a polynomial vector $\mathbf{v} \leftarrow \Psi^k$ means sampling each coefficient according to Ψ individually.

Compression function. The compression function is formulated as $\mathsf{Compress}_q(x,d) = \left\lceil \frac{2^d}{q} \cdot x \right\rfloor \bmod^{+} 2^d$, while the decompression function is defined as $\mathsf{Decompress}_q(x,d) = \left\lceil \frac{q}{2^d} \cdot x \right\rfloor$. When they deal with a polynomial (vector), the procedure is applied to each coefficient individually. For any $x \in \mathbb{Z}_q$, $x' = \mathsf{Decompress}_q(\mathsf{Compress}_q(x,d),d)$ is an element close to x, i.e., $|x' - x \bmod^{\pm} q| \leq \lceil \frac{q}{2^{d+1}} \rfloor$.

Module learning with error (MLWE). Let n be a power of two. The underlying hardness assumption of Kyber [4,27] is module learning with error (MLWE) [12] over the ring $\mathcal{R}$. The hard problem module learning with errors (MLWE) over $\mathcal{R}$ is to distinguish uniform samples $(\mathbf{a}_i, b_i) \xleftarrow{\$} \mathcal{R}_q^k \times \mathcal{R}_q$ from the samples $(\mathbf{a}_i, b_i) \in \mathcal{R}_q^k \times \mathcal{R}_q$, where $\mathbf{a}_i \xleftarrow{\$} \mathcal{R}_q^k$ and $b_i = \mathbf{a}_i^T \mathbf{s} + e_i$ with $\mathbf{s} \leftarrow \Psi_1$ and $e_i \leftarrow \Psi_2$ for all i. The MLWE problem over $\mathcal{R}$ is hard if the advantage $\mathbf{Adv}^{\mathrm{mlwe}}_{m,k,\Psi_1,\Psi_2}(\mathsf{A})$ of any probabilistic polynomial time adversary A is negligible, where

$$\mathbf{Adv}^{\mathrm{mlwe}}_{m,k,\Psi_1,\Psi_2}(\mathsf{A}) = \left|\Pr\left[b' = 1 : \begin{array}{c} \mathbf{A} \xleftarrow{\$} \mathcal{R}_q^{l\times k}; (\mathbf{s},\mathbf{e}) \leftarrow \Psi_1^k \times \Psi_2^l; \\ \mathbf{b} = \mathbf{A}\mathbf{s} + \mathbf{e}; b' \leftarrow \mathsf{A}(\mathbf{A},\mathbf{b}) \end{array}\right] - \Pr\left[b' = 1 : \mathbf{A} \xleftarrow{\$} \mathcal{R}_q^{l\times k}; \mathbf{b} \leftarrow \mathcal{R}_q^l; b' \leftarrow \mathsf{A}(\mathbf{A},\mathbf{b})\right]\right|.$$

2.2. Cryptographic Primitives

A public-key encryption scheme contains PKE = (KeyGen, Enc, Dec), with a message space $\mathcal{M}$. The key generation algorithm KeyGen returns a pair of a public key and a secret key (pk, sk). The encryption algorithm Enc takes a public key pk and a message $m \in \mathcal{M}$ to produce a ciphertext c. The deterministic decryption algorithm Dec takes a secret key sk and a ciphertext c, and outputs either a message $m \in \mathcal{M}$ or a special symbol $\perp$ to indicate a rejection. The decryption error probability of PKE, which is denoted as δ, is defined as $\mathrm{E}[\max_{m\in\mathcal{M}}\Pr[\mathrm{Dec}(sk,\mathrm{Enc}(pk,m))] \neq m] < \delta$. The advantage of an adversary A against indistinguishability under chosen-plaintext attacks (IND-CPA) for public-key encryption is defined as

$$\mathbf{Adv}^{\mathrm{CPA}}_{\mathrm{PKE}}(\mathsf{A}) = \left|\Pr\left[b' = b : \begin{array}{c} (pk,sk) \leftarrow \mathrm{KeyGen}(); \\ (m_0,m_1) \leftarrow \mathsf{A}(pk); \\ b \xleftarrow{\$} \{0,1\}; c^* \leftarrow \mathrm{Enc}(pk,m_b); \\ b' \leftarrow \mathsf{A}(c^*) \end{array}\right] - \frac{1}{2}\right|.$$

A key encapsulation mechanism consists of three algorithms, which are defined as KEM = (KeyGen, Encaps, Decaps) with a key space $\mathcal{K}$. The key generation algorithm KeyGen returns a pair of a public key and a secret key (pk, sk). The encapsulation algorithm Encaps takes a public key pk to produce a ciphertext c and a key $K \in \mathcal{K}$. The deterministic decapsulation algorithm Decaps inputs a secret key sk and a ciphertext c, and outputs either a key $K \in \mathcal{K}$ or a special symbol $\perp$ to indicate a rejection. The correctness error δ of KEM is defined as $\Pr[\mathrm{Decaps}(sk,c) \neq K : (c,K) \leftarrow \mathrm{Encaps}(pk)] < \delta$. The advantage of an adversary A against indistinguishability under chosen-ciphertext attacks (IND-CCA) for the key encapsulation mechanism is defined as

$$\mathbf{Adv}^{\mathrm{CCA}}_{\mathrm{KEM}}(\mathsf{A}) = \left|\Pr\left[b' = b : \begin{array}{c} (pk,sk) \leftarrow \mathrm{KeyGen}(); \\ b \xleftarrow{\$} \{0,1\}; \\ (c^*,K_0^*) \leftarrow \mathrm{Encaps}(pk); \\ K_1^* \xleftarrow{\$} \mathcal{K}; \\ b' \leftarrow \mathsf{A}^{\mathrm{Decaps}(\cdot)}(pk,c^*,K_b^*) \end{array}\right] - \frac{1}{2}\right|.$$

2.3. Kyber

In 2017, Bos et al. [27] proposed a lattice-based cryptography suite called Cryptographic Suite for Algebraic Lattices (CRYSTALS for short). The algorithms of CRYSTALS are designed based on the MLWE problem over a module lattice, meaning that the algorithms take into account the security of LWE-based schemes and the efficiency of RLWE-based schemes. Among them, Kyber is an IND-CCA secure key encapsulation mechanism (KEM). Kyber follows a common construction framework. Specifically, it has two steps: the first step is to construct an IND-CPA secure public key encryption (Kyber.CPAPKE); The second step is to transform the IND-CPA secure PKE into an IND-CCA secure KEM (Kyber.CCAKEM) by using a variant of Fujisaki–Okamoto transform [28,29]. More precisely, Kyber is based on the MLWE problem over power-of-two cyclotomic ring $\mathbb{Z}[x]/(x^n+1)$, where n is a power of two. In the first round of the NIST PQC competition, Kyber's modulus was chosen to be 7681, but it was changed after the first round, and adjusted from 7681 to 3329 [4]. Additionally, Kyber's secret distribution has been different from the ciphertext noise distribution for Kyber512 since the third round. In 2022, NIST finally selected MLWE-based Kyber (official name is ML-KEM) as the only standardized KEM candidate [8].

3. Our Proposal: Tyber

In this section, we will propose Tyber, a variant scheme of Kyber [4] over trinomial cyclotomic rings $\mathbb{Z}_q[x]/(x^n - x^{n/2} + 1)$. The construction of our Tyber is based on [4], and also includes an IND-CPA secure public-key encryption (Tyber.CPAPKE) and an IND-CCA secure key encapsulation mechanism (Tyber.CCAKEM). There are some slight differences between our Tyber and that in [4].

3.1. Concrete Description

Firstly, the formal description of IND-CPA secure public key encryption (Tyber.CPAPKE) of our Tyber is presented in Algorithms 1–3. It can be transformed into its IND-CCA secure key encapsulation mechanism (Tyber.CCAKEM) by using a variant of the Fujisaki–Okamoto transform [28,29]. The detailed description of our Tyber.CCAKEM is presented in Algorithms A1–A3 in Appendix A.

Algorithm 1 Tyber.CPAPKE.KeyGen(): key generation

$\mathbf{A} \sim \mathcal{R}_q^{k \times k} := \mathsf{Sam}(\rho)$
$(\mathbf{s}, \mathbf{e}) \leftarrow \Psi_1^k \times \Psi_1^k$
$\mathbf{t} := \mathbf{A}\mathbf{s} + \mathbf{e}$
return $(pk := (\mathbf{t}, \rho), sk := \mathbf{s})$

Algorithm 2 Tyber.CPAPKE.Enc($pk = (\mathbf{t}, \rho), m \in \mathcal{M}$): encryption

$\mathbf{A} \sim \mathcal{R}_q^{k \times k} := \mathsf{Sam}(\rho)$
$(\mathbf{r}, \mathbf{e}_1, e_2) \leftarrow \Psi_1^k \times \Psi_2^k \times \Psi_2$
$\mathbf{u} := \mathsf{Compress}_q(\mathbf{A}^T\mathbf{r} + \mathbf{e}_1, d_u)$
$v := \mathsf{Compress}_q(\mathbf{t}^T\mathbf{r} + e_2 + \lceil \frac{q}{2} \rfloor \cdot m, d_v)$
return $c := (\mathbf{u}, v)$

Algorithm 3 Tyber.CPAPKE.Dec($sk = \mathbf{s}, c = (\mathbf{u}, v)$): decryption

$\mathbf{u} := \mathsf{Decompress}_q(\mathbf{u}, d_u)$
$v := \mathsf{Decompress}_q(v, d_v)$
return $m' := \mathsf{Compress}_q(v - \mathbf{s}^T\mathbf{u}, 1)$

We restate the definitions of $\mathcal{R}$ and $\mathcal{R}_q$: $\mathcal{R} = \mathbb{Z}[x]/(x^n - x^{n/2} + 1)$ and $\mathcal{R}_q = \mathbb{Z}_q[x]/(x^n - x^{n/2} + 1)$, respectively, where n is a positive integer of the form 2^k3^l with $k \geq 1$ and $l \geq 0$. We introduce $\mathcal{M}$ as the message space for Tyber.CPAPKE, consisting of binary strings of length n, which can be interpreted as polynomials in $\mathcal{R}$ with coefficients in $\{0,1\}$. Sam is an extendable output function, and takes as input an n-bit string ρ, and then, produces $\mathbf{A}$, uniformly random over $\mathcal{R}_q^{k\times k}$, in Algorithms 1 and 2. Ψ_1 and Ψ_2 are the distributions over $\mathcal{R}$. The definitions of $\mathsf{Compress}_q$ and $\mathsf{Compress}_q$ can be found in Section 2.1.

3.2. Parameter Sets

The parameter sets of Tyber are given in Table 2. We mainly provide parameter sets aimed at quantum security of 128, 192, and 256 bits. The polynomial dimension n is fixed to 324. Actually, n can be any integer of the form $2^k3^l, k \geq 1, l \geq 0$, like 256, 384, or 432. We use two moduli: $q = 2917$ for $k = 2$, and $q = 3889$ for $k \in \{3,4\}$. Both two moduli support very fast NTT-based polynomial multiplications when $n = 324$ according to the studies in [14,18]. $\Phi(x)$ means the underlying cyclotomic polynomial used in the schemes, and we use a trinomial cyclotomic polynomial of the form $x^n - x^{n/2} + 1$. Ψ_1 and Ψ_2 are the distributions over $\mathcal{R}$. We mainly consider the centered binomial distribution B_η and the distribution $\bar{B}_\eta$ with respect to a positive integer η, as described in Section 2.1. According to the studies in [30], the centered binomial distribution can guarantee a relatively strong theoretical security, while achieving easier and safer implementation. d_u and d_v are the compression parameters. The magnitudes of the public key ($|pk|$), ciphertext ($|ct|$), and bandwidth (B.W., i.e., $|pk| + |ct|$) are quantified in bytes. The column "(Sec.C,Sec.Q)" means the estimated security level with respect to the primal attack expressed in bits, where "Sec.C" denotes classical security and "Sec.Q" denotes quantum security. We follow the classical and the quantum core-SVP hardness methodology as in Kyber [4] and use the same Python script to calculate security levels. The last column δ gives the error probabilities, whose details can be found in Section 4.1.

Table 2. Parameter sets of Tyber.

Scheme		n	k	q	$\Phi(x)$	(Ψ_1, Ψ_2)	(d_u, d_v)	$\|pk\|$	$\|ct\|$	B.W.	(Sec.C,Sec.Q)	δ
Tyber	Tyber648	324	2	2917	$x^n - x^{n/2} + 1$	$(\bar{B}_2, \bar{B}_2)$	(9,5)	1004	932	1936	(142,129)	2^{-129}
	Tyber972	324	3	3889	$x^n - x^{n/2} + 1$	(B_1, B_1)	(10,3)	1490	1337	2827	(217,197)	2^{-204}
	Tyber1296	324	4	3889	$x^n - x^{n/2} + 1$	(B_1, B_1)	(10,5)	1976	1823	3799	(305,276)	2^{-256}

4. Analysis

In this section, we will present a correctness analysis, provable security reduction, and implementation analysis of our scheme.

4.1. Correctness Analysis

The correctness analysis of Tyber.CPAPKE and Tyber.CCAKEM in our scheme is similar to that in [4,27]. Firstly, following the condition of decryption error in [4,27], we have the following theorem.

Theorem 1 (Derived from Theorem 1 in [27]). *Let $k, \Psi_1, \Psi_2, d_u, d_v$ be the values as in Table 2. Let $\mathbf{s}, \mathbf{e}, \mathbf{r}, \mathbf{e}_1, e_2$ be random variables according to the same distribution as in Algorithms 1–3. Let $\mathbf{c}_u \leftarrow \psi_{d_u}^k, c_v \leftarrow \psi_{d_v}$ be generated according to the distribution ψ_d, which is defined as follows: Sampling $y \xleftarrow{\$} \mathcal{R}$, and returning $(y - \mathsf{Decompress}_q(\mathsf{Compress}_q(y,d),d)) \bmod^{\pm} q$. Denote*

$$\delta = Pr\Big[\|\mathbf{e}^T\mathbf{r} - \mathbf{s}^T(\mathbf{e}_1 + \mathbf{c}_u) + c_v + e_2\|_\infty \geq \lceil q/4 \rfloor\Big], \tag{1}$$

then our Tyber.CCAKEM has an error probability of δ.

4.1.1. The Product in $\mathbb{Z}[x]/(x^n - x^{n/2} + 1)$

In order to calculate δ in Formula (1), the computations of $\mathbf{e}^T\mathbf{r} - \mathbf{s}^T(\mathbf{e}_1 + \mathbf{c}_u) + c_v + e_2$ have to be figured out. Note that all the computations in Formula (1) in Theorem 1 are performed in the rings $\mathcal{R}$ and $\mathcal{R}_q$. For example, the inner product $\mathbf{e}^T\mathbf{r}$ needs to be computed in the ring $\mathcal{R}_q = \mathbb{Z}_q[x]/(x^n - x^{n/2} + 1)$, where $\mathbf{e}, \mathbf{r} \leftarrow \Psi_1^k$.

Our way to calculate δ in Formula (1) is different from that in [4,27], since the form of the product $h = fg \in \mathbb{Z}[x]/(x^n + 1)$ is different from that of $h = fg \in \mathbb{Z}[x]/(x^n - x^{n/2} + 1)$. In the following, we take $\mathbb{Z}[x]/(x^4 + 1)$ as an example. The product of $f = \sum_{i=0}^{3} f_i x^i$ and $g = \sum_{i=0}^{3} g_i x^i$ can be represented as

$$h = \begin{bmatrix} h_0 \\ h_1 \\ h_2 \\ h_3 \end{bmatrix} = \begin{bmatrix} f_0 & -f_3 & -f_2 & -f_1 \\ f_1 & f_0 & -f_3 & -f_2 \\ f_2 & f_1 & f_0 & -f_3 \\ f_3 & f_2 & f_1 & f_0 \end{bmatrix} \cdot \begin{bmatrix} g_0 \\ g_1 \\ g_2 \\ g_3 \end{bmatrix}. \tag{2}$$

The main characteristic of h is that each coefficient of h is the sum of four numbers, each of which is in the form of $f_i g_j$. E.g., the third coefficient h_3 in Formula (2) is $h_3 = f_3 g_0 + f_2 g_1 + f_1 g_2 + f_0 g_3$. However, in the ring $\mathbb{Z}[x]/(x^4 - x^2 + 1)$, the product of f and g can be obtained from

$$h = \begin{bmatrix} h_0 \\ h_1 \\ h_2 \\ h_3 \end{bmatrix} = \begin{bmatrix} f_0 & -f_3 & -f_2 & -f_1 - f_3 \\ f_1 & f_0 & -f_3 & -f_2 \\ f_2 & f_1 + f_3 & f_0 + f_2 & f_1 \\ f_3 & f_2 & f_1 + f_3 & f_0 + f_2 \end{bmatrix} \cdot \begin{bmatrix} g_0 \\ g_1 \\ g_2 \\ g_3 \end{bmatrix}, \tag{3}$$

where the coefficient of h might contain some summands in the form of $f_i g_j + (f_i + f_{i'})g_{j'}$. E.g., the third coefficient h_3 in Formula (3) is $h_3 = (f_3 g_0 + (f_1 + f_3)g_2) + (f_2 g_1 + (f_0 + f_2)g_3)$.

Inspired by the methodology in [18], the general representation of the product between $f = \sum_{i=0}^{n-1} f_i x^i$ and $g = \sum_{i=0}^{n-1} g_i x^i$ in $\mathbb{Z}[x]/(x^n - x^{n/2} + 1)$ is achieved through a matrix–vector multiplication as follows:

$$h = \begin{bmatrix} h_0 \\ h_1 \\ \vdots \\ h_{n-1} \end{bmatrix} = \begin{bmatrix} \mathbf{L} - \mathbf{U} & -\mathbf{F} - \mathbf{U} \\ \mathbf{F} + \mathbf{U} & \mathbf{F} + \mathbf{L} \end{bmatrix} \cdot \begin{bmatrix} g_0 \\ g_1 \\ \vdots \\ g_{n-1} \end{bmatrix}, \tag{4}$$

where $\mathbf{F}, \mathbf{L}, \mathbf{U}$ are the Toeplitz matrices of dimension $\frac{n}{2}$, which are defined as follows:

$$\mathbf{F} = \begin{bmatrix} f_{n/2} & f_{n/2-1} & \cdots & f_1 \\ f_{n/2+1} & f_{n/2} & \cdots & f_2 \\ \vdots & \vdots & \ddots & \vdots \\ f_{n-1} & f_{n-2} & \cdots & f_{n/2} \end{bmatrix},$$

$$\mathbf{L} = \begin{bmatrix} f_0 & 0 & \cdots & 0 \\ f_1 & f_0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ f_{n/2-1} & f_{n/2-2} & \cdots & f_0 \end{bmatrix}, \quad \mathbf{U} = \begin{bmatrix} 0 & f_{n-1} & \cdots & f_{n/2+1} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & f_{n-1} \\ 0 & 0 & \cdots & 0 \end{bmatrix}.$$

The correctness error of Tyber is based on the general form of h. The whole product is divided into two parts through the form of partitioned matrices. As specified in Formula (4), the individual coefficients in the lower half of the resulting product, i.e.,

$$\begin{bmatrix} \mathbf{F}+\mathbf{U} & \mathbf{F}+\mathbf{L} \end{bmatrix} \cdot \begin{bmatrix} g_0 \\ g_1 \\ \vdots \\ g_{n-1} \end{bmatrix}, \tag{5}$$

are obtained from the sum of $n/2$ terms:

$$\sigma_{i,i',j,j'} = f_i g_j + (f_i + f_{i'}) g_{j'} \tag{6}$$

The third coefficient h_3 in Formula (3) is an example. The coefficient of the l-th row in the upper half, i.e.,

$$\begin{bmatrix} \mathbf{L}-\mathbf{U} & -\mathbf{F}-\mathbf{L} \end{bmatrix} \cdot \begin{bmatrix} g_0 \\ g_1 \\ \vdots \\ g_{n-1} \end{bmatrix}, \tag{7}$$

is the sum of $(n/2 - l)$ terms of the form $\sigma_{i,i',j,j'} = f_i g_j + (f_i + f_{i'}) g_{j'}$, as in Formula (6), and l terms of the form $\theta_{i,i',j,j'} = f_i g_j + f_{i'} g_{j'}$.

As suggested in [18], the first form has a "wider" distribution than the latter form from the random variance point of view. Therefore, our subsequent correctness analysis will be based on the first form for conservative estimation.

4.1.2. Error Probability over $\mathbb{Z}[x]/(x^n - x^{n/2} + 1)$

The detailed procedure of calculating the error probability δ in Theorem 1 is given here. As for the term $\mathbf{e}^T\mathbf{r} - \mathbf{s}^T(\mathbf{e}_1 + \mathbf{c}_u) + c_v + e_2$ in Formula (1), each coefficient of the product $\mathbf{e}^T\mathbf{r}$ is distributed as the sum of $kn/2$ independent random variables of the form $\sigma_{i,i',j,j'} = e_i r_j + (e_i + e_{i'}) r_{j'}$, as in Formula (6), where $e_i, e_{i'}, r_j, r_{j'} \leftarrow \Psi_1$, since $\mathbf{e}^T\mathbf{r}$ is a polynomial inner product including k single polynomial multiplications.

The analysis is the same for the term $\mathbf{s}^T(\mathbf{e}_1 + \mathbf{c}_u)$, except that they are generated from different distribution $\mathbf{s} \leftarrow \Psi_1^k, \mathbf{e}_1 \leftarrow \Psi_2^k, \mathbf{c}_u \leftarrow \psi_{d_u}^k$, as in Theorem 1.

The sum of the random variances $\mathbf{e}^T\mathbf{r}$, $\mathbf{s}^T(\mathbf{e}_1 + \mathbf{c}_u)$, c_v, and e_2, is obtained by computing their convolutions, where it uses the symmetry of the centered binomial distribution. The probability that any coefficient of $\mathbf{e}^T\mathbf{r} - \mathbf{s}^T(\mathbf{e}_1 + \mathbf{c}_u) + c_v + e_2$ is greater than $\lceil q/4 \rfloor$ is its tail probability with the threshold $\lceil q/4 \rfloor$. Finally, the final correctness error δ is derived by applying the union bound.

As for the three parameter sets in Table 2, we obtain the corresponding error probabilities as 2^{-129}, 2^{-204}, and 2^{-256}, respectively, by using the reasonable but conservative methodology over trinomial cyclotomic rings mentioned above.

4.2. Provable Security Reduction

In the following, we will derive the provable security based on the MLWE assumption, which is similar to that of Kyber [4,27]. Formally, the following theorems guarantee its IND-CPA security and IND-CCA security.

Theorem 2. *Under the MLWE hardness assumption over trinomial cyclotomic rings, the public key encryption of Tyber is IND-CPA secure in the random oracle model.*

Proof. We complete our proof via a progression of games $\mathbf{G}_0$, $\mathbf{G}_1$, and $\mathbf{G}_2$. Consider an adversary A who challenges the IND-CPA security experiment. We define $\mathbf{Succ}_i$ as the

occurrence wherein A wins in the game $\mathbf{G}_i$, specifically, when A produces an output b' that matches the challenge bit b in $\mathbf{G}_i$.

Game $\mathbf{G}_0$. We define the initial security experiment as Game $\mathbf{G}_0$, which serves as the foundation for achieving original IND-CPA security. Thus, $\mathbf{Adv}^{\mathrm{CPA}}_{\mathrm{PKE}}(\mathsf{A}) = |\Pr[\mathbf{Succ}_0] - 1/2|$.

Game $\mathbf{G}_1$. This game is the same as $\mathbf{G}_0$, except replacing $\mathbf{t} := \mathbf{As} + \mathbf{e}$ used in KeyGen by $\mathbf{t} \xleftarrow{\$} \mathcal{R}_q^k$. To distinguish $\mathbf{G}_1$ from $\mathbf{G}_0$ is equivalent to solve an MLWE problem. More precisely, there exists an adversary B such that $|\Pr[\mathbf{Succ}_0] - \Pr[\mathbf{Succ}_1]| \leq \mathbf{Adv}^{\mathrm{mlwe}}_{k,k,\Psi_1,\Psi_1}(\mathsf{B})$.

Game $\mathbf{G}_2$. This game is identical to $\mathbf{G}_1$, except using uniformly random elements from $\mathcal{R}_q^k$ and $\mathcal{R}_q$ to replace $\mathbf{A}^T\mathbf{r} + \mathbf{e}_1$ and $\mathbf{t}^T\mathbf{r} + e_2$, respectively. Similarly, there exists an adversary C such that $|\Pr[\mathbf{Succ}_1] - \Pr[\mathbf{Succ}_2]| \leq \mathbf{Adv}^{\mathrm{mlwe}}_{k+1,k,\Psi_1,\Psi_2}(\mathsf{C})$.

Note that in $\mathbf{G}_2$ the information of m_b is perfectly hidden by uniformly random elements, so $\Pr[\mathbf{Succ}_2] = 1/2$.

Finally, we obtain $\mathbf{Adv}^{\mathrm{CPA}}_{\mathrm{PKE}}(\mathsf{A}) \leq \mathbf{Adv}^{\mathrm{mlwe}}_{k,k,\Psi_1,\Psi_1}(\mathsf{B}) + \mathbf{Adv}^{\mathrm{mlwe}}_{k+1,k,\Psi_1,\Psi_2}(\mathsf{C})$. Therefore, if the MLWE problem over trinomial cyclotomic ring is hard, our PKE is IND-CPA secure. □

If the underlying PKE is IND-CPA secure, the studies in [29,31] show us that the resulting KEM obtained by using a variant of the Fujisaki–Okamoto transform is IND-CCA secure in both the random oracle model and quantum random oracle model. According to [4,27,29,31], we have the following theorem.

Theorem 3. *Under the MLWE hardness assumption over the trinomial cyclotomic ring* $\mathbb{Z}[x]/(x^n - x^{n/2} + 1)$*, the key encapsulation mechanism of Tyber is IND-CCA secure in both the random oracle model and quantum random oracle model.*

4.3. Implementation Analysis

From an implementation point of view, the fundamental and time-consuming operation is the polynomial multiplication in algebraically structured lattice-based schemes, including Kyber and our Tyber. A more efficient polynomial multiplication algorithm can greatly accelerate the efficiency of the schemes. According to the studies in [14,18], our Tyber can achieve the same efficiency as Kyber.

As shown in Table 2, Tyber uses trinomial cyclotomic rings $\mathbb{Z}_q[x]/(x^n - x^{n/2} + 1)$, where $(n = 324, q = 2917)$ and $(n = 324, q = 3889)$. As for both parameter tuples, from the work in [18] we can know that there is the isomorphism $\mathbb{Z}_q[x]/(x^n - x^{n/2} + 1) \cong \mathbb{Z}_q[x]/(x^{n/2} - \zeta_1) \times \mathbb{Z}_q[x]/(x^{n/2} - \zeta_2)$, where ζ_1 and ζ_2 should satisfy $\zeta_1 + \zeta_2 = 1$ and $\zeta_1 \cdot \zeta_2 = 1$. We can choose $\zeta_1 = \zeta^{162}$ and $\zeta_2 = \zeta^{810}$, where ζ is the primitive $3n$-th (i.e., 972-th) root of unity in $\mathbb{Z}_q$. Then, we can utilize the efficient radix-2 NTT and radix-3 NTT techniques from [14]. The former corresponds to the isomorphism $\mathbb{Z}_q[x]/(x^{2s} - \zeta^{2\beta}) \cong \mathbb{Z}_q[x]/(x^s - \zeta^\beta) \times \mathbb{Z}_q[x]/(x^s + \zeta^\beta)$, and the latter corresponds to the isomorphism $\mathbb{Z}_q[x]/(x^{3s} - \zeta^{3\beta}) \cong \mathbb{Z}_q[x]/(x^s - \zeta^\beta) \times \mathbb{Z}_q[x]/(x^s - \rho\zeta^\beta) \times \mathbb{Z}_q[x]/(x^s - \rho^2\zeta^\beta)$, where s, β are positive integers and ρ is the third root of unity. In detail, the final isomorphism can be described as follows:

$$\mathbb{Z}_q[x]/(x^n - x^{n/2} + 1) \cong \prod_{i \in \mathbb{Z}_{3n}^\times} \mathbb{Z}_q[x]/(x - \zeta^i),$$

where $\mathbb{Z}_{3n}^\times$ is the group of invertible elements of $\mathbb{Z}_{3n}$.

According to the benchmark results in [14,18], the NTT technique mentioned above is as efficient as that of Kyber. Regarding the implementation analysis in this section, we present an implementation analysis that, while not exhaustive, aims to demonstrate the potential efficiency of our schemes in comparison to Kyber.

5. Comparisons

As illustrated in Table 3, we provide concise comparisons between our scheme and the NIST-standardized candidate Kyber [4]. n is the polynomial dimension. q is the modulus.

$\Phi(x)$ means the underlying cyclotomic polynomial used in the schemes. The magnitudes of the public key ($|pk|$), ciphertext ($|ct|$), and bandwidth (B.W., i.e., $|pk| + |ct|$) are quantified in bytes. "Sec.C" denotes classical security and "Sec.Q" denotes quantum security, both of which are expressed in bits. δ means the error probability.

Table 3. Comparison of schemes.

Scheme		n	k	q	$\Phi(x)$	$\|pk\|$	$\|ct\|$	B.W.	(Sec.C,Sec.Q)	δ
Tyber (Ours)	Tyber648	324	2	2917	$x^n - x^{n/2} + 1$	1004	932	1936	(142,129)	2^{-129}
	Tyber972	324	3	3889	$x^n - x^{n/2} + 1$	1490	1337	2827	(217,197)	2^{-204}
	Tyber1296	324	4	3889	$x^n - x^{n/2} + 1$	1976	1823	3799	(305,276)	2^{-256}
Kyber	Kyber512	256	2	3329	$x^n + 1$	800	768	1568	(118,107)	2^{-139}
	Kyber768	256	3	3329	$x^n + 1$	1184	1088	2272	(183,166)	2^{-164}
	Kyber1024	256	4	3329	$x^n + 1$	1568	1568	3136	(256,232)	2^{-174}

Upon comparison, our scheme utilizes trinomial cyclotomic rings, so there is more flexibility when selecting parameters. The dimension n in our scheme can take values of the form $2^k3^l, k \geq 1, l \geq 0$. However, Kyber suffers from the inflexibility of selecting parameters due to its underlying power-of-two cyclotomic rings, since n can only be $2^k, k \geq 1$.

Although Kyber has a more compact public key and ciphertext for the three security levels, Kyber actually achieves quantum security of 107, 166, and 232 bits, respectively, which is far less than 128, 192, and 256 bits, respectively. Note that Kyber768 has a quantum security of 166 bits, which has a very large margin for quantum security of 128 bits, resulting in larger security redundancy. Another important point is that the error probability of Kyber1024 is only 2^{-174}, which actually does not match its security requirement as 232-bit quantum security.

According to Table 3, our scheme stands out with the practical and reliable security guarantees, since our scheme achieves the target quantum security of 128, 192, and 256 bits (actually achieving 129, 197, and 276 bits). The error probabilities of our scheme are precisely calibrated to satisfy the targeted security level for each parameter set, making them negligible in comparison to the specified security level, as they are substantively lower than 2^{-129}, 2^{-204}, and 2^{-256}, respectively. When compared to Kyber, Tyber648, Tyber972, and Tyber1296 exhibit stronger quantum security, by 22, 31, and 44 bits, than Kyber512, Kyber768, and Kyber1024, respectively. In addition, Tyber972 and Tyber1296 demonstrate significantly lower error probabilities when compared to Kyber768 and Kyber1024, respectively.

Note that Tyber uses different moduli, $q = 2917$ and $q = 3889$, in order to achieve a balanced integrated performance for the three security levels. However, to adapt to different moduli we need two suites of NTT algorithms with different primitive roots of unity, resulting in more complicated implementation and more memory usage. In addition, according to the studies in Section 4.1, the trinomial cyclotomic rings used in Tyber lead to lower error probabilities due to their more complicated structures, but the error probabilities can be controlled in a negligible range by choosing parameter sets carefully.

6. Conclusions and Future Works

To overcome the inflexibility of selecting parameters with respect to MLWE-based schemes over power-of-two cyclotomic rings, in this paper we propose Tyber, a variant scheme of Kyber over trinomial cyclotomic rings, and provide three parameter sets which achieve the target quantum security of 128, 192, and 256 bits (actually achieving 129, 197, and 276 bits) with matching and negligible error probabilities. Tyber exhibits stronger quantum security by 22, 31, and 44 bits than Kyber for the three security levels, respectively. As for the limitation of this work, we only provide the concrete construction and theoretical analysis of Tyber. Therefore, the future works should consist of practical software or hardware implementations, such as C, Cortex-M4 and FPGA implementations.

Author Contributions: Conceptualization, methodology, writing—original draft preparation, W.L. and Z.L. (Zhichuang Liang); writing—review and editing, W.L., Z.L. (Zhaoman Liu), X.Z., Y.Y. and Z.L (Zhichuang Liang). All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Key Research and Development Program of China (2022YFB2701600), the General Project of State Key Laboratory of Cryptography (MMKFKT202227), the Technical Standard Project of Shanghai Scientific and Technological Committee (21DZ2200500), the Shanghai Collaborative Innovation Fund (XTCX-KJ-2023-54), and the Special Fund for Key Technologies in Blockchain of Shanghai Scientific and Technological Committee (23511100300).

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A. IND-CCA KEM from Fujisaki–Okamoto Transform

Algorithm A1 CCAKEM.KeyGen()

$(pk', sk') :=$ CPAPKE.KeyGen()
$pkh := \mathcal{F}(pk')$
$z \xleftarrow{\$} \{0,1\}^n$
return $(pk := pk', sk := (z, pkh, pk, sk'))$

Algorithm A2 CCAKEM.Encaps(pk)

$m \xleftarrow{\$} \{0,1\}^n$
$(\hat{K}, r) := \mathcal{G}(\mathcal{F}(pk), m)$
$c :=$ CPAPKE.Enc$(pk, m; r)$
$K := \mathcal{H}(\hat{K}, c)$
return (c, K)

Algorithm A3 CCAKEM.Decaps($sk = (z, pkh, pk, sk'), c$)

$m' :=$ CPAPKE.Dec(sk', c)
$(\hat{K}', r') := \mathcal{G}(pkh, m')$
$c' :=$ CPAPKE.Enc$(pk, m'; r')$
if $c = c'$ **then**
  return $K := \mathcal{H}(\hat{K}', c)$
else
  return $K := \mathcal{H}(z, c)$
end if

References

1. NIST. Post-Quantum Cryptography, Round 1 Submissions. 2016. Available online: https://csrc.nist.gov/Projects/post-quantum-cryptography/post-quantum-cryptography-standardization/round-1-submissions (accessed on 1 June 2024).
2. NIST. Post-Quantum Cryptography, Round 2 Submissions. 2019. Available online: https://csrc.nist.gov/Projects/post-quantum-cryptography/post-quantum-cryptography-standardization/round-2-submissions (accessed on 1 June 2024).
3. NIST. Post-Quantum Cryptography, Round 3 Submissions. 2020. Available online: https://csrc.nist.gov/Projects/post-quantum-cryptography/post-quantum-cryptography-standardization/round-3-submissions (accessed on 1 June 2024).
4. Avanzi, R.; Bos, J.; Ducas, L.; Kiltz, E.; Lepoint, T.; Lyubashevsky, V.; Schanck, J.M.; Schwabe, P.; Seiler, G.; Stehlé, D. CYRYSTALS-Kyber—Algorithm specifications and supporting documentation (version 3.01). *NIST Post-Quantum Cryptogr. Stand. Process.* **2020**, *2*, 1–43.
5. NIST. Module-Lattice-Based Key-Encapsulation Mechanism Standard. In *NIST Post-Quantum Cryptography Standardization Process*; NIST: Gaithersburg, MD, USA, 2023.
6. Bai, S.; Ducas, L.; Kiltz, E.; Lepoint, T. Supporting documentation: CRYSTALS-Dilithium: Algorithm Specifications and Supporting Documentation. In *NIST Post-Quantum Cryptography Standardization Process*; NIST: Gaithersburg, MD, USA, 2020.
7. *FIPS 204*; Module-Lattice-Based Digital Signature Standard; NIST Post-Quantum Cryptography Standardization Process. NIST: Gaithersburg, MD, USA, 2023.

8. NIST. PQC Standardization Process: Announcing Four Candidates to be Standardized, Plus Fourth Round Candidates. 2022. Available online: https://csrc.nist.gov/News/2022/pqc-candidates-to-be-standardized-and-round-4 (accessed on 1 June 2024).
9. CACR. The Public-Key Algorithms in the Second Round of National Cryptographic Algorithm Design Competition. 2019. Available online: http://sfjs.cacrnet.org.cn/site/term/list_77_1.html (accessed on 1 June 2024).
10. Regev, O. On lattices, learning with errors, random linear codes, and cryptography. In Proceedings of the 37th Annual ACM Symposium on Theory of Computing, Baltimore, MD, USA, 22–24 May 2005; Gabow, H.N., Fagin, R., Eds.; ACM: New York, NY, USA, 2005; pp. 84–93. [CrossRef]
11. Lyubashevsky, V.; Peikert, C.; Regev, O. On Ideal Lattices and Learning with Errors over Rings. In Proceedings of the Advances in Cryptology—EUROCRYPT 2010, 29th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Monaco, French, 30 May–3 June 2010; Lecture Notes in Computer Science; Gilbert, H., Ed.; Springer: Berlin/Heidelberg, Germany, 2010; Volume 6110, pp. 1–23. [CrossRef]
12. Langlois, A.; Stehlé, D. Worst-case to average-case reductions for module lattices. *Des. Codes Cryptogr.* **2015**, *75*, 565–599. [CrossRef]
13. Basso, A.; Mera, J.M.B.; D'Anvers, J.P. Supporting documentation: SABER: Mod-LWR based KEM (Round 3 Submission). In *NIST Post-Quantum Cryptography Standardization Process*; NIST: Gaithersburg, MD, USA, 2020.
14. Duman, J.; Hövelmanns, K.; Kiltz, E.; Lyubashevsky, V.; Seiler, G.; Unruh, D. A Thorough Treatment of Highly-Efficient NTRU Instantiations. In Proceedings of the PKC 2023, Atlanta, GA, USA, 7–10 May 2023; Volume 13940, pp. 65–94.
15. Ducas, L.; Durmus, A. Ring-LWE in Polynomial Rings. In Proceedings of the Public Key Cryptography—PKC 2012—15th International Conference on Practice and Theory in Public Key Cryptography, Darmstadt, Germany, 21–23 May 2012; Fischlin, M., Buchmann, J., Manulis, M., Eds.; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2012; Volume 7293, pp. 34–51. [CrossRef]
16. Fouque, P.A.; Hoffstein, J.; Kirchner, P.; Lyubashevsky, V. Falcon: Fast-Fourier Lattice-based Compact Signatures over NTRU. In *NIST Post-Quantum Cryptography Standardization Process*; NIST: Gaithersburg, MD, USA, 2016.
17. Alkim, E.; Bilgin, Y.A.; Cenk, M. Compact and Simple RLWE Based Key Encapsulation Mechanism. In Proceedings of the Progress in Cryptology—LATINCRYPT 2019—6th International Conference on Cryptology and Information Security in Latin America, Santiago de Chile, Chile, 2–4 October 2019; Schwabe, P., Thériault, N., Eds.; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2019; Volume 11774, pp. 237–256. [CrossRef]
18. Lyubashevsky, V.; Seiler, G. NTTRU: Truly Fast NTRU Using NTT. *IACR Trans. Cryptogr. Hardw. Embed. Syst.* **2019**, *2019*, 180–201. [CrossRef]
19. Mera, J.M.B.; Karmakar, A.; Kundu, S.; Verbauwhede, I. Scabbard: A suite of efficient learning with rounding key-encapsulation mechanisms. *IACR Trans. Cryptogr. Hardw. Embed. Syst.* **2021**, *2021*, 474–509. [CrossRef]
20. Hassan, C.A.; Yayla, O. Radix-3 NTT-Based Polynomial Multiplication for Lattice-Based Cryptography. Master's Thesis, Middle East Technical University, Ankara, Turkey, 2022; p. 726.
21. Espitau, T.; Fouque, P.; Gérard, F.; Rossi, M.; Takahashi, A.; Tibouchi, M.; Wallet, A.; Yu, Y. Mitaka: A Simpler, Parallelizable, Maskable Variant of Falcon. In Proceedings of the Advances in Cryptology—EUROCRYPT 2022—41st Annual International Conference on the Theory and Applications of Cryptographic Techniques, Trondheim, Norway, 30 May–3 June 2022; Dunkelman, O., Dziembowski, S., Eds.; Proceedings, Part III; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2022; Volume 13277, pp. 222–253. [CrossRef]
22. Espitau, T.; Wallet, A.; Yu, Y. On Gaussian Sampling, Smoothing Parameter and Application to Signatures. In Proceedings of the Advances in Cryptology—ASIACRYPT 2023—29th International Conference on the Theory and Application of Cryptology and Information Security, Guangzhou, China, 4–8 December 2023; Guo, J., Steinfeld, R., Eds.; Proceedings, Part VII; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2023; Volume 14444, pp. 65–97. [CrossRef]
23. Liang, Z.; Fang, B.; Zheng, J.; Zhao, Y. Compact and Efficient KEMs over NTRU Lattices. *Comput. Stand. Interfaces* **2024**, *89*, 103828. [CrossRef]
24. Bai, S.; Jangir, H.; Lin, H.; Ngo, T.; Wen, W.; Zheng, J. Compact Encryption based on Module-NTRU problems. In Proceedings of the PQCrypto 2024, 2024, *to be appeared*.
25. Liang, Z.; Shen, S.; Shi, Y.; Sun, D.; Zhang, C.; Zhang, G.; Zhao, Y.; Zhao, Z. Number Theoretic Transform: Generalization, Optimization, Concrete Analysis and Applications. In Proceedings of the Information Security and Cryptology—16th International Conference, Inscrypt 2020, Guangzhou, China, 11–14 December 2020; Revised Selected Papers; Wu, Y., Yung, M., Eds.; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2020; Volume 12612, pp. 415–432. [CrossRef]
26. Washington, L.C. *Introduction to Cyclotomic Fields*; Graduate Texts in Mathematics 83; Springer: Berlin/Heidelberg, Germany, 1997.
27. Bos, J.W.; Ducas, L.; Kiltz, E.; Lepoint, T.; Lyubashevsky, V.; Schanck, J.M.; Schwabe, P.; Seiler, G.; Stehlé, D. CRYSTALS—Kyber: A CCA-Secure Module-Lattice-Based KEM. In Proceedings of the 2018 IEEE European Symposium on Security and Privacy, EuroS&P 2018, London, UK, 24–26 April 2018; pp. 353–367.

[CrossRef]
28. Fujisaki, E.; Okamoto, T. Secure Integration of Asymmetric and Symmetric Encryption Schemes. In Proceedings of the Advances in Cryptology—CRYPTO'99, 19th Annual International Cryptology Conference, Santa Barbara, CA, USA, 15–19 August 1999; Lecture Notes in Computer Science; Wiener, M.J., Ed.; Springer: Berlin/Heidelberg, Germany, 1999; Volume 1666, pp. 537–554. [CrossRef]
29. Hofheinz, D.; Hövelmanns, K.; Kiltz, E. A Modular Analysis of the Fujisaki-Okamoto Transformation. In Proceedings of the Theory of Cryptography—15th International Conference, TCC 2017, Baltimore, MD, USA, 12–15 November 2017; Proceedings, Part I; Kalai, Y., Reyzin, L., Eds.; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2017; Volume 10677, pp. 341–371. [CrossRef]
30. Alkim, E.; Ducas, L.; Pöppelmann, T.; Schwabe, P. Post-quantum Key Exchange—A New Hope. In Proceedings of the 25th USENIX Security Symposium, USENIX Security 16, Austin, TX, USA, 10–12 August 2016; Holz, T., Savage, S., Eds.; USENIX Association: Washington, DC, USA, 2016; pp. 327–343.
31. Jiang, H.; Zhang, Z.; Chen, L.; Wang, H.; Ma, Z. IND-CCA-Secure Key Encapsulation Mechanism in the Quantum Random Oracle Model, Revisited. In Proceedings of the Advances in Cryptology—CRYPTO 2018—38th Annual International Cryptology Conference, Santa Barbara, CA, USA, 19–23 August 2018; Proceedings, Part III; Shacham, H., Boldyreva, A., Eds.; Lecture Notes in Computer Science; Springer: Berlin/Heidelberg, Germany, 2018; Volume 10993, pp. 96–125. [CrossRef]