mathematics

MDPI

Article

Highly Efficient and Light NTRU-Based Key Encapsulation Mechanisms with Small Moduli

Jing Fan [1,2], Bo-Yue Fang [3], Wei-Ze Wang [3], Neng-Hai Yu [1], Feng-Hua Li [4] and Long Wang [2,5,*]

[1] Department of Cyber Science and Technology, University of Science and Technology of China, Hefei 230026, China; fanjing@mail.ustc.edu.cn (J.F.); ynh@ustc.edu.cn (N.-H.Y.)
[2] The 6th Research Institute of China Electronics Corporation, Beijing 100083, China
[3] Department of Computer Science, Fudan University, Shanghai 200433, China; wzwang21@m.fudan.edu.cn (W.-Z.W.); byfang16@fudan.edu.cn (B.-Y.F.)
[4] Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100085, China; lifenghua@iie.ac.cn
[5] School of Cyber Science and Technology, Shandong University, Qingdao 266000, China
* Correspondence: peakwl@mail.sdu.edu.cn

Abstract

In this paper, we present CTRU-Light, an IND-CCA-secure key encapsulation mechanism (KEM) derived from NTRU and RLWE (and RLWR in variant) assumptions over power-of-two cyclotomic rings. Our CTRU-Light employs a compact NTT-compatible modulus $q = 641$ while maintaining minimal public key and ciphertext dimensions with negligible error probability. Specifically, the design yields public key and ciphertext sizes of 1206 bytes under an error probability bound of $\leq 2^{-110}$. When benchmarked against Kyber (NIST's sole standardized KEM), CTRU-Light demonstrates 23.0–30.0% lower bandwidth consumption, accelerates key generation by at least 6.0%, and achieves over 1.3× speed enhancement in both encapsulation and decapsulation procedures.

Keywords: lattice-based cryptography; post-quantum cryptography; key encapsulation mechanism; NTRU; compression strategy

MSC: 94A60

check for updates

Academic Editor: Jonathan Blackledge

Received: 2 September 2025
Revised: 6 October 2025
Accepted: 10 October 2025
Published: 24 October 2025

Citation: Fan, J.; Fang, B.-Y.; Wang, W.-Z.; Yu, N.-H.; Li, F.-H.; Wang, L. Highly Efficient and Light NTRU-Based Key Encapsulation Mechanisms with Small Moduli. *Mathematics* **2025**, *13*, 3388. https://doi.org/10.3390/math13213388

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Most contemporary public-key cryptosystems depend on the computational difficulty of integer factorization or discrete logarithm problems. These mathematical challenges, however, become vulnerable to efficient resolution through Shor's quantum algorithm, driving the advancement of post-quantum cryptography (PQC) [1].

The U.S. National Institute of Standards and Technology (NIST) PQC standardization initiative has categorized five principal classes of PQC primitives: code-based, hash-based, multivariate-based, isogeny-based, and lattice-based constructions [2]. Lattice-based approaches emerge as particularly advantageous due to their optimal balance between security assurances, communication economy, and operational efficiency [3].

Originally introduced in 1998, the NTRU cryptosystem [4] maintains its security integrity and continues to serve as a foundation for numerous cryptographic protocols. NTRU-based designs received consideration during the NIST PQC competition, exemplified by the Falcon signature scheme [5,6] and NTRUEncrypt KEMs [7]. While Kyber (an MLWE-based construction) ultimately secured standardization, NTRU persists within established standards including IEEE 1363.1 [8] and X9.98 [9], alongside implementations in protocols such as OpenSSH [10].

Developing an NTRU-based KEM that surpasses existing solutions in performance metrics presents an ongoing research challenge [11]. Prior solutions exhibit limitations in design efficiency despite supporting compact moduli and reduced bandwidth. Building upon foundational work in high-dimensional lattice constructions, this paper introduces CTRU-Light—a novel variant engineered to maintain robust security and operational efficiency while utilizing a significantly minimized modulus ($q = 641$). In the post-quantum era, ensuring the confidentiality and integrity of sensitive data in critical sectors like healthcare is a major challenge [12], for which our proposed algorithm offers an effective security mechanism.

Our Contributions

In detail, our contributions are summarized as follows:

- We introduce **CTRU-Light**, a cryptographic framework derived from the CTRU foundation [13]. This construction integrates an IND-CPA-secure public-key encryption scheme with an IND-CCA-secure key encapsulation mechanism, leveraging NTRU and RLWE assumptions over power-of-two cyclotomic rings.
- We establish a parameter configuration for CTRU-Light targeting NIST security level I. This design employs a compact NTT-compatible modulus $q = 641$, achieving minimized public key and ciphertext dimensions with negligible error probabilities. A compression methodology [14] further reduces key and ciphertext sizes by 5.7%.
- We develop C-based implementations of CTRU-Light and conduct comprehensive benchmarking against leading KEM schemes. Experimental validation confirms that CTRU-Light surpasses NEV [15] in security robustness, bandwidth efficiency, and error resilience. Compared to Kyber, CTRU-Light achieves 23.0–30.0% bandwidth reduction with computational speedups exceeding 1.3× in encapsulation/decapsulation and 6.0% in key generation.

2. Preliminaries

2.1. Notations and Definitions

Let n and q be positive integers. Denote $\mathbb{Z}_q = \mathbb{Z}/q\mathbb{Z} \cong \{0, 1, \ldots, q-1\}$. Define the power-of-two cyclotomic rings $\mathcal{R} = \mathbb{Z}[x]/(x^n + 1)$ and $\mathcal{R}_q = \mathbb{Z}_q[x]/(x^n + 1)$, where $n = 2^k$ for some integer k. Elements in these rings are polynomials, denoted $f, g \in \mathcal{R}$ or $\mathcal{R}_q$.

Each polynomial $f \in \mathcal{R}$ (or $\mathcal{R}_q$) can be written as:

$$f(x) = \sum_{i=0}^{n-1} f_i x^i, \quad \text{where } f_i \in \mathbb{Z} \text{ (or } \mathbb{Z}_q).$$

We adopt notation from [13]. For a number $r \in \mathbb{R}$ and modulus q, define:

- $r' = r \bmod \pm q$: the representative of r in $[-\frac{q}{2}, \frac{q}{2})$
- $r' = r \bmod q$: the representative of r in $[0, q)$
- $\|w\|_{q,\infty} = |w \bmod \pm q|$: ℓ_∞ norm
- $\|w\|_{q,2} = \sqrt{\sum_i \|w_i\|_{q,\infty}^2}$: ℓ_2 norm for vector w

Random sampling notations.

- $x \leftarrow D$: sample x from distribution D.
- $x \xleftarrow{\$} D$: uniform random sample from finite set D.

Distributions.

- Centered Binomial Distribution $\mathcal{B}_\eta$: Sample $(a_1, \ldots, a_\eta, b_1, \ldots, b_\eta) \xleftarrow{\$} \{0,1\}^{2\eta}$ and output $\sum_{i=1}^{\eta}(a_i - b_i)$.

- Ternary Distribution $\mathcal{T}_\mu$: $x \in \{-1, 0, 1\}$ with $\Pr[x = \pm 1] = \mu$ and $\Pr[x = 0] = 1 - 2\mu$.

2.2. Cryptographic Primitives

Public-Key Encryption (PKE).

A PKE scheme is a triple $(\texttt{KeyGen}, \texttt{Enc}, \texttt{Dec})$.

- $(\texttt{pk}, \texttt{sk}) \leftarrow \texttt{KeyGen}()$
- $\texttt{ct} \leftarrow \texttt{Enc}(\texttt{pk}, m)$
- $m \leftarrow \texttt{Dec}(\texttt{sk}, \texttt{ct})$ or $\perp$

The error probability δ is:

$$\delta = \mathbb{E}_{(\texttt{pk},\texttt{sk})\leftarrow\texttt{KeyGen}}\left[\max_{m\in\mathcal{M}} \Pr[\texttt{Dec}(\texttt{sk}, \texttt{Enc}(\texttt{pk}, m)) \neq m]\right]$$

Key Encapsulation Mechanism (KEM).

A KEM scheme is $(\texttt{KeyGen}, \texttt{Encaps}, \texttt{Decaps})$ with key space $\mathcal{K}$.

$$(\texttt{pk}, \texttt{sk}) \leftarrow \texttt{KeyGen}(), \quad (\texttt{ct}, K) \leftarrow \texttt{Encaps}(\texttt{pk}),$$
$$K' \leftarrow \texttt{Decaps}(\texttt{sk}, \texttt{ct}) \in \mathcal{K} \cup \{\perp\}$$

The error probability is:

$$\delta = \Pr[\texttt{Decaps}(\texttt{sk}, \texttt{ct}) \neq K]$$

2.3. Hardness Assumptions

Definition 1 (NTRU Assumption [4]). *Let Ψ be a distribution over $\mathcal{R}$ and $p \in \mathcal{R}$ an invertible element. Sample $f', g \leftarrow \Psi$, set $f = pf' + 1$, and $h = g/f$. The decisional NTRU problem is to distinguish h from uniform in $\mathcal{R}$. The advantage of adversary $\mathcal{A}$ is:*

$$Adv^{\mathcal{R},\Psi}_{NTRU}(\mathcal{A}) = |\Pr[\mathcal{A}(h) = 1] - \Pr[\mathcal{A}(u) = 1]|,$$

where $u \xleftarrow{\$} \mathcal{R}$.

Definition 2 (RLWE Assumption [16]). *Let Ψ be a distribution over $\mathcal{R}$. The decisional RLWE problem is to distinguish:*

$$(h, c = hr + e) \quad \textit{from} \quad (h, u), \quad \textit{where } r, e \leftarrow \Psi,\ u \xleftarrow{\$} \mathcal{R}$$

Definition 3 (RLWR Assumption [17]). *Let $q > p \geq 2$ and Ψ be a distribution over $\mathcal{R}$. Define $c = \left\lfloor \frac{p}{q} \cdot hr \right\rceil \bmod p$ for $r \leftarrow \Psi$. The RLWR problem is to distinguish:*

$$(h, c) \quad \textit{from} \quad (h, u), \quad \textit{where } h \xleftarrow{\$} \mathcal{R}_q,\ u \xleftarrow{\$} \mathcal{R}_p$$

2.4. Number Theoretic Transform (NTT)

The Number Theoretic Transform (NTT) is a special case of the Fast Fourier Transform (FFT) defined over a finite field. It is widely used for efficient polynomial multiplication due to its quasi-linear time complexity $\mathcal{O}(n \log n)$.

Given polynomials f and g in $\mathcal{R}_q$, their product can be efficiently computed using:

$$f \cdot g = \mathrm{INTT}(\mathrm{NTT}(f) \circ \mathrm{NTT}(g)),$$

where $\circ$ denotes point-wise multiplication, and INTT is the inverse transform.

2.5. Scalable E_8 Lattice Code

We use the scalable E_8 lattice code defined in [13], which is constructed from the 8-dimensional Extended Hamming Code. For a 4-bit input $k \in \{0,1\}^4$, the encoding algorithm outputs a lattice point $\lambda \cdot (kH \bmod 2)$, where H is a standard generator matrix. The decoding algorithm takes $v \in \mathbb{R}^8$, solves the closest vector problem in E_8, and returns k. From [13], Theorem 1: decoding succeeds if the Euclidean norm of the error vector, computed after reducing coefficients into $[-\lambda, \lambda)$, is less than λ.

2.6. The CTRU Framework

CTRU, introduced by Liang et al. [13], is a framework for constructing compact KEMs from NTRU lattices. Its design philosophy differs from module-based schemes like Kyber by building primitives directly over a single polynomial ring $\mathcal{R}_q$. This architectural choice is central to its efficiency and compactness.

Core Structure: A standard CTRU scheme consists of a public key $h = g/f \in \mathcal{R}_q$ and a ciphertext $c = \lfloor \frac{q_2}{q}(hr + e + M) \rceil \in \mathcal{R}_{q_2}$, where f, g, r, e are polynomials and M is the encoded message. The core advantage lies in this simplicity: both the public key and ciphertext are single ring elements, whereas Module-LWE schemes require vectors of polynomials, increasing bandwidth.

Advantages and Trade-offs: Compared to other NTRU/RLWR-based schemes, CTRU's primary advantage is its communication efficiency. The single-polynomial structure inherently leads to smaller public keys and ciphertexts. However, this simplicity presents a trade-off. While module-based schemes can easily scale to different security levels by adjusting the module's rank (i.e., the number of polynomials in vectors), CTRU's security scaling is primarily tied to the ring dimension n. This makes parameterization for a wide range of security levels less flexible. Our work, CTRU-Light, builds on this compact foundation, further optimizing it with a smaller modulus and refined techniques.

3. Our Scheme: CTRU-Light

We now introduce CTRU-Light, comprising an IND-CPA-secure public-key encryption scheme (`CTRU-Light.PKE`) and an IND-CCA-secure key encapsulation mechanism (`CTRU-Light.KEM`). Figure 1 provides a high-level overview of the scheme's operations.

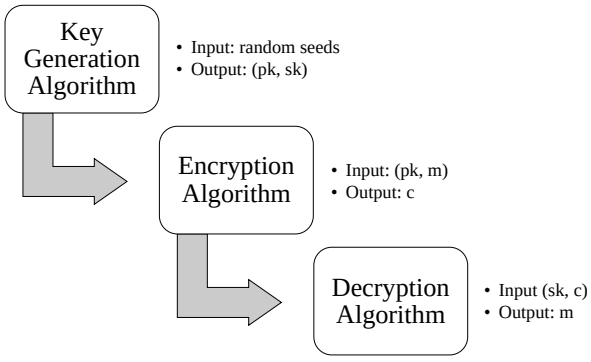


Figure 1. Flow chart of CTRU-Light PKE.

3.1. Proposal Description

The `CTRU-Light.PKE` scheme is defined in Algorithms 1–5. Let n denote a power of two and q an NTT-compatible prime. We define $p = 1 - x^{512}$. Secret polynomials f', g are sampled from distribution Ψ_1, while randomness r, e originate from Ψ_2. Messages $m \in \mathcal{M}$ are represented as binary polynomials of degree 255 ($|m| = 256$).

Algorithm 1 $\texttt{CTRU-Light.PKE.KeyGen}(1^\kappa)$

repeat
  $f', g \leftarrow \Psi_1$
  $f := pf' + 1$
until f^{-1} exists in $\mathcal{R}_q$
$h := g/f$
return $(\texttt{pk} := h, \texttt{sk} := f)$

Algorithm 2 $\texttt{CTRU-Light.PKE.Enc}(\texttt{pk} = h, m \in \mathcal{M})$

$r, e \leftarrow \Psi_2$
$c := hr + e + \texttt{PolyEncode}(m)$
return c

Algorithm 3 $\texttt{CTRU-Light.PKE.Dec}(\texttt{sk} = f, c)$

$\tilde{m} := c \cdot f$
$m := \texttt{PolyDecode}(\tilde{m})$
return m

Algorithm 4 $\texttt{PolyEncode}(m = \sum_{i=0}^{|m|-1} m_i x^i \in \mathcal{M})$

for $i = 0$ to $|m|/4 - 1$ **do**
  $k_i := (m_{4i}, m_{4i+1}, m_{4i+2}, m_{4i+3}) \in \{0,1\}^4$
  $(v_{8i}, \ldots, v_{8i+7}) := \texttt{Encode}_{E_8}(k_i)$
end for
$v := \sum_{i=0}^{2|m|-1} v_i x^i$
return v

Algorithm 5 $\texttt{PolyDecode}(v = \sum_{i=0}^{n-1} v_i x^i \in \mathcal{R}_q)$

for $i = 0$ to $|m|/4 - 1$ **do**
  $x_i := (v_{8i}, \ldots, v_{8i+7}) \in \mathbb{R}^8$
  $(m_{4i}, m_{4i+1}, m_{4i+2}, m_{4i+3}) := \texttt{Decode}_{E_8}(x_i)$
end for
$m := \sum_{i=0}^{|m|-1} m_i x^i$
return m

3.2. CTRU-Light.KEM Construction

We employ the $\texttt{FO}_\perp^{\texttt{ID}(pk),m}$ transform [18] to derive IND-CCA security from our IND-CPA secure PKE. Let $\iota, \gamma \geq 256$ provide adequate security. Define $H : \{0,1\}^* \to \mathcal{K} \times \texttt{COINS}$, where $\mathcal{K}$ is the shared key domain and COINS is the PKE randomness domain. Let H_1 extract $\mathcal{K}$ from H's output, and $\texttt{ID} : \texttt{PK} \to \{0,1\}^\gamma$ as a deterministic identifier. To provide a clear, high-level overview of our KEM constructions, the operational flow is visualized in Figure 2. This flowchart details the three core procedures: key generation (KeyGen), encapsulation (Encaps), and decapsulation (Decaps). Given that the CTRU-Light and CTRU-Light-RLWR schemes are structurally identical aside from the handling of ciphertext, we present a unified diagram for conciseness. The steps marked in red are specific to the CTRU-Light-RLWR variant. All other steps in the process are common to both schemes.

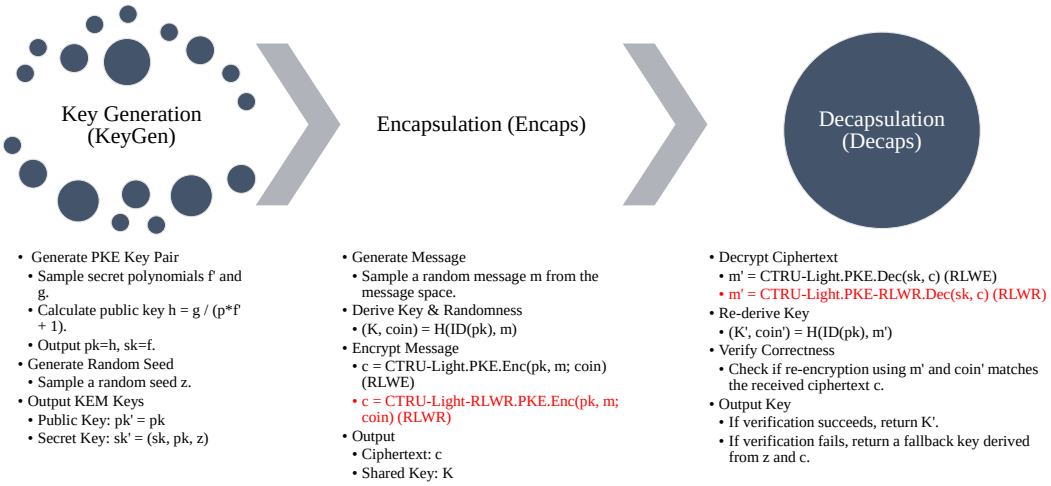


Figure 2. Unified flowchart of the KeyGen, Encaps, and Decaps steps for CTRU-Light.KEM, which are formally described in Algorithms 6–8. The steps highlighted in red are unique to the CTRU-Light-RLWR variant. The remaining steps are identical for both schemes.

Algorithm 6 $\texttt{CTRU-Light.KEM.KeyGen}(1^\kappa)$

$(\texttt{pk}, \texttt{sk}) \leftarrow \texttt{CTRU-Light.PKE.KeyGen}(1^\kappa)$
$z \xleftarrow{\$} \{0,1\}^\iota$
return $(\texttt{pk}', \texttt{sk}') := (\texttt{pk}, (\texttt{sk}, \texttt{pk}, z))$

Algorithm 7 $\texttt{CTRU-Light.KEM.Encaps}(\texttt{pk}')$

$m \xleftarrow{\$} \mathcal{M}$
$(K, \texttt{coin}) := H(\texttt{ID}(\texttt{pk}), m)$
$c := \texttt{CTRU-Light.PKE.Enc}(\texttt{pk}, m; \texttt{coin})$
return (c, K)

Algorithm 8 $\texttt{CTRU-Light.KEM.Decaps}(\texttt{sk}', c)$

$m' := \texttt{CTRU-Light.PKE.Dec}(\texttt{sk}, c)$
$(K', \texttt{coin}') := H(\texttt{ID}(\texttt{pk}), m')$
$\tilde{K} := H_1(\texttt{ID}(\texttt{pk}), z, c)$
if $m' \neq \perp$ **and** $c = \texttt{CTRU-Light.PKE.Enc}(\texttt{pk}, m'; \texttt{coin}')$ **then**
  return K'
else
  return $\tilde{K}$
end if

3.3. Correctness Analysis

Correctness hinges on the noise term $gr + ef + sf'$ remaining small enough for the E_8 decoder to function correctly. This is ensured with high probability by the small coefficients of the sampled polynomials. The detailed proof sketch can be found in the original manuscript.

4. Parameter Sets

Table 1 summarizes the cryptographic parameters for our CTRU-Light scheme, targeting NIST security level I. All size measurements are reported in bytes.

Table 1. Parameter configurations for CTRU-Light.

n	q	(Ψ_1, Ψ_2)	$\lvert pk \rvert$	$\lvert ct \rvert$	B.W.	NTRU-(C, Q)	RLWE-(C, Q)	δ
512	641	(B_1, B_1)	603	603	1206	(129,117)	(129,117)	2^{-113}

5. Provable Security Reduction

Our security analysis establishes that CTRU-Light.PKE achieves IND-CPA security under both the NTRU and RLWE assumptions.

Theorem 1 (IND-CPA security of CTRU-Light.PKE)**.** *For any probabilistic polynomial-time adversary* A*, there exist adversaries* B *and* C *with comparable running times satisfying:*

$$\mathbf{Adv}^{\mathrm{IND-CPA}}_{\mathrm{CTRU-Light.PKE}}(\mathsf{A}) \leq \mathbf{Adv}^{\mathrm{NTRU}}_{\mathcal{R}_q,\Psi_1}(\mathsf{B}) + \mathbf{Adv}^{\mathrm{RLWE}}_{\mathcal{R}_q,\Psi_2}(\mathsf{C}).$$

The proof follows a standard game-hopping argument. The IND-CCA security of CTRU-Light.KEM follows directly from the generic $\mathrm{FO}^{f}_{ID(pk),m}$ transformation [19] in the quantum random oracle model.

6. Concrete Security

We analyze the concrete security against the primal attack, which formulates cryptanalysis as a unique-Short Vector Problem (u-SVP) [20,21]. We use the core-SVP methodology [22] to estimate the costs of solving u-SVP using BKZ algorithms, with cost models of $2^{0.292b}$ (classical) and $2^{0.265b}$ (quantum) [23]. Our security estimates derive from adapted Python implementations of [13,24], with results in Table 1. We omit analysis of dual and overstretched NTRU attacks [25,26] as they are not practical for our parameter choices.

7. RLWR-Based Variant: CTRU-Light-RLWR

This section introduces CTRU-Light-RLWR, which uses rounding to further compress ciphertexts. The modified encryption and decryption procedures appear in Algorithms 9 and 10.

Algorithm 9 CTRU-Light-RLWR.PKE.Enc($pk = h, m \in \mathcal{M}$)

1: $r \leftarrow \Psi_2$
2: $c := \lfloor \frac{q_2}{q}(hr + \mathrm{PolyEncode}(m)) \rceil \bmod q_2$
3: **return** c

Algorithm 10 CTRU-Light-RLWR.PKE.Dec($sk = f, c$)

1: $\tilde{c} := \lfloor \frac{q}{q_2} c \rceil \bmod q$
2: $\tilde{m} := \tilde{c} f$
3: $m := \mathrm{PolyDecode}(\tilde{m})$
4: **return** m

The IND-CPA security of this variant reduces to the NTRU and RLWR assumptions. Table 2 presents its optimized parameter configuration.

Table 2. Parameter configuration for CTRU-Light-RLWR.

n	q	q_2	(Ψ_1, Ψ_2)	$\lvert pk \rvert$	$\lvert ct \rvert$	B.W.	NTRU-(C, Q)	RLWR-(C, Q)	δ
512	641	2^8	(B_1, B_1)	603	512	1115	(129,117)	(130,118)	2^{-102}

8. Implementation Details

8.1. Polynomial Compression Techniques

We employ space-efficient compression for public key polynomials h [14]. For coefficient pairs (h_i, h_{i+1}), we compute $h_i + q \cdot h_{i+1} \bmod q^2$. This technique leverages the 10-bit representation of $q = 641$ coefficients, saving 37 bytes (5.7% reduction) per polynomial for $n = 512$. The computational overhead of these compression and decompression operations is negligible. They primarily involve bitwise shifts and additions, which are orders of magnitude faster than the dominant operations in the scheme, such as NTT-based polynomial multiplication and SHA-3 hashing. Therefore, their impact on the overall performance benchmarks is minimal.

8.2. Optimized NTT Arithmetic

We implement incomplete NTT transformations for polynomial multiplication [27], as $q = 641$ precludes full NTT for $n = 512$. We exploit the property $q \equiv 1 \bmod 128$ to apply 6 levels of radix-2 FFT, decomposing the computation into products in 8-dimensional rings, which are then handled by optimized schoolbook and Karatsuba algorithms.

8.3. Cryptographic Primitives

We instantiate all hash functions using SHA-3 primitives (SHA3-512, SHAKE-256) for hashing, key derivation, and randomness expansion.

9. Performance Evaluation and Comparative Analysis

We present benchmarking results for our C implementations against prominent lattice-based KEMs. Table 3 provides a comparative analysis of the fundamental parameters of these schemes. The performance of these schemes is dominated by three main components: polynomial multiplication (typically via NTT), cryptographic hashing, and random sampling from specified distributions. The asymptotic complexity of the core multiplication operation is $O(n \log n)$. While a detailed breakdown of cycles per operation is highly platform-dependent, the total cycle counts presented in Table 4 provide a holistic and practical measure of the overall efficiency.

Benchmarks were conducted on an Intel Core i7-10510U @ 2.3GHz running Ubuntu 20.04 LTS with gcc 9.4.0 and flags `-O3 -march=native`.

Table 3. Comparative analysis of KEM schemes.

Scheme	Assumptions	n	q	$\|pk\|$	$\|ct\|$	B.W.	(C, Q)	δ
CTRU-Light	NTRU,RLWE	512	641	603	603	1206	(129,117)	2^{-113}
CTRU-Light-RLWR	NTRU,RLWR	512	641	603	512	1115	(129,117)	2^{-102}
NEV	NTRU,RLWE	512	769	615	615	1230	(120,109)	2^{-138}
Kyber-512	MLWE	512	3329	800	768	1568	(118,107)	2^{-139}

Table 4. Computational performance (kilo cycles).

Scheme	KeyGen	Encaps	Decaps
CTRU-Light-512	59.5	43.5	85.0
CTRU-Light-RLWR-512	60.0	41.9	77.8
Kyber-512	65.0	72.3	111.7
Saber-512 [28]	35.0	40.6	55.0

9.1. Comparison with NEV

When evaluating against NEV, CTRU-Light demonstrates measurable improvements. It reduces total bandwidth by approximately 2% due to the smaller modulus and enhanced compression. From a security perspective, CTRU-Light provides stronger theoretical guarantees, offering 9-bit and 8-bit improvements in classical and quantum security estimates, respectively, positioning it as a more robust alternative.

9.2. Comparison with Kyber

The comparative analysis with Kyber reveals several fundamental advantages of CTRU-Light. It reduces bandwidth requirements by 23%, primarily due to its single-polynomial structure compared to Kyber's matrix-based approach. This also translates to performance benefits, with 1.6× faster encapsulation and 1.3× faster decapsulation. CTRU-Light also provides 11-bit and 10-bit stronger security in classical and quantum settings, respectively.

This highlights a key design trade-off in lattice-based cryptography. Kyber's Module-LWE framework offers high flexibility, allowing security levels to be scaled by changing the module's dimensions (e.g., Kyber-512, Kyber-768, Kyber-1024). In contrast, CTRU-Light's NTRU approach over a single ring offers a simpler, more compact design, leading to superior bandwidth and performance for a given ring dimension. For applications where bandwidth and speed are paramount, CTRU-Light presents a compelling alternative.

A consolidated summary of these comparisons, highlighting the trade-offs between bandwidth, performance, and failure probability, is presented in Table 5.

Table 5. Comparative Analysis of KEMs at NIST Security Level I.

Scheme	Security Assumption	Bandwidth (Bytes) PK + CT = Total	Performance (k-Cycles) KeyGen/Encaps/Decaps	Decryption Failure (δ)
CTRU-Light-512	NTRU, RLWE	603 + 603 = **1206**	59.5/43.5/85.0	2^{-113}
Kyber-512	MLWE	800 + 768 = **1568**	65.0/72.3/111.7	2^{-139}
Saber-512	MLWR	672 + 736 = **1408**	**35.0/40.6/55.0**	2^{-120}
NEV-512	NTRU, RLWE	615 + 615 = **1230**	*(See Note)*	2^{-138}

Note on NEV Performance: The provided performance table does not include NEV. The NEV paper claims its KEM is 1.42-1.74X faster than Kyber. To avoid mixing data from different benchmarking environments, its specific values are omitted here.

10. Conclusions

This work presents CTRU-Light, a high-performance NTRU-based KEM that advances the state-of-the-art. Through careful parameter optimization, we have demonstrated that NTRU-based constructions can surpass MLWE-based alternatives like Kyber in both performance and bandwidth efficiency. The scheme's compact size, efficient implementation, and strong security foundations position it as a viable candidate for deployment.

Future research directions should focus on expanding the scheme's applicability beyond power-of-two cyclotomic rings, potentially exploring other cyclotomic fields that could provide better flexibility for intermediate security levels. Additional investigations into side-channel resistant implementations would further strengthen the scheme's practical security profile. The extension of these techniques to other cryptographic primitives, particularly digital signatures, represents another promising avenue for research. These developments would help establish a more comprehensive NTRU-based cryptographic ecosystem, providing alternatives to current lattice-based standards. The results presented in this work suggest that NTRU-based constructions remain competitive in the

post-quantum landscape and warrant continued investigation as the field evolves toward standardization and widespread adoption.

Author Contributions: J.F. and L.W. Conceptualization; W.-Z.W. Methodology; B.-Y.F. Software; J.F., W.-Z.W. and B.-Y.F. Validation; N.-H.Y. Formal analysis; F.-H.L. Investigation; L.W. Resources; B.-Y.F. Data curation; J.F. Writing—original draft; J.F. Writing—review and editing; L.W. Supervision; N.-H.Y. Project administration; L.W. Funding acquisition. All authors have read and agreed to the published version of the manuscript.

Funding: The National Key R&D Program of China (No. 2022YFB3103800).

Data Availability Statement: No new data were created in this study.

Acknowledgments: The authors would like to acknowledge the developers of the open-source cryptographic libraries that were used for comparison and benchmarking.

Conflicts of Interest: Authors Jing Fan and Long Wang was employed by The 6th Research Institute of China Electronics Corporation. The remaining authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

References

1. Shor, P.W. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM J. Comput.* **1997**, *26*, 1484–1509. [CrossRef]
2. National Institute of Standards and Technology. PQC Standardization Process: Announcing Four Candidates to be Standardized, Plus Fourth Round Candidates. Available online: https://csrc.nist.gov/News/2022/pqc-candidates-to-be-standardized-and-round-4 (accessed on 27 August 2025).
3. Lu, X.; Liu, Y.; Jia, D.; Xue, H.; He, J.; Zhang, Z.; Liu, Z.; Yang, H.; Li, B.;Wang, K. LAC: Lattice-Based Cryptosystems. In *NIST Post-Quantum Cryptography Standardization Process*; NIST: Gaithersburg, MD, USA, 2019.
4. Hoffstein, J.; Pipher, J.; Silverman, J.H. NTRU: A ring-based public key cryptosystem. In Proceedings of the ANTS-III: International Symposium on Algorithmic Number Theory, Portland, OR, USA, 21–25 June 1998; pp. 267–288.
5. Fouque, P.A.; Hoffstein, J.; Kirchner, P.; Lyubashevsky, V.; Pornin, T.; Prest, T.; Ricosset, T.; Seiler, G.; Whyte, W.; Zhang, Z. Falcon: Fast-Fourier Lattice-Based Compact Signatures over NTRU. In *NIST Post-Quantum Cryptography Standardization Process*; NIST: Gaithersburg, MD, USA, 2020.
6. Fouque, P.A.; Kirchner, P.; Pornin, T.; Yu, Y. BAT: Small and Fast KEM over NTRU Lattices. *IACR Trans. Cryptogr. Hardw. Embed. Syst.* **2022**, *2022*, 240–265. [CrossRef]
7. Chen, C.; Danba, O.; Hoffstein, J.; Hülsing, A.; Rijneveld, J.; Schanck, J.M.; Saito, T. NTRU Submission. In *NIST Post-Quantum Cryptography Standardization Process*; NIST: Gaithersburg, MD, USA, 2020.
8. Jablon, D. IEEE P1363 Standard Specifications for Public-Key Cryptography. In Proceedings of the NIST Key Management Workshop, Gaithersburg, MD, USA, 1–2 October 2008.
9. Business Wire. Security Innovation's NTRUEncrypt Adopted as X9 Standard for Data Protection. Available online: https://www.businesswire.com/news/home/20110411005309/en/Security-Innovations-NTRUEncrypt-Adopted-X9-Standard-Data (accessed on 27 August 2025).
10. OpenSSH. OpenSSH Release Notes. Available online: https://www.openssh.com/releasenotes.html (accessed on 27 August 2025).
11. Hülsing, A.; Rijneveld, J.; Schanck, J.M.; Schwabe, P. High-Speed Key Encapsulation from NTRU. In Proceedings of the CHES 2017, Taipei, Taiwan, 25–28 September 2017; pp. 232–252.
12. Chinnasamy, P.; Dhavamani, L.; Ayyasamy, R.K.; Kumar, B.S.; Kiran, A. QuantumBlock health records: enhancing healthcare data security with quantum cryptography and blockchain technology. *Cluster Comput.* **2025**, *28*, 474. [CrossRef]
13. Liang, Z.; Fang, B.; Zheng, J.; Zhao, Y. Compact and efficient KEMs over NTRU lattices. *arXiv* **2022**, arXiv:2205.05413. [CrossRef]
14. Bernstein, D.J.; Brumley, B.; Chen, M.S.; Chuengsatiansup, C.; Lange, T.; Marotzke, A.; Peng, B.Y.; Tuveri, N.; van Vredendaal, C.; Yang, B.Y. NTRU Prime: Round 3. In *NIST Post-Quantum Cryptography Standardization Process*; NIST: Gaithersburg, MD, USA, 2020.
15. Zhang, J.; Feng, D.; Yan, D. NEV: Faster and Smaller NTRU Encryption Using Vector Decoding. In Proceedings of the ASIACRYPT 2023, Guangzhou, China, 4–8 December 2023; pp. 3–34.
16. Lyubashevsky, V.; Peikert, C.; Regev, O. On Ideal Lattices and Learning with Errors over Rings. In Proceedings of the EUROCRYPT 2010, French Riviera, France, 30 May–3 June 2010; pp. 1–23.

17. Banerjee, A.; Peikert, C.; Rosen, A. Pseudorandom Functions and Lattices. In Proceedings of the EUROCRYPT 2012, Cambridge, UK, 15–19 April 2012; pp. 719–737.
18. Hofheinz, D.; Hövelmanns, K.; Kiltz, E. A Modular Analysis of the Fujisaki-Okamoto Transformation. In Proceedings of the TCC 2017, Baltimore, MD, USA, 12–15 November 2017; pp. 341–371.
19. Duman, J.; Hövelmanns, K.; Kiltz, E.; Lyubashevsky, V.; Seiler, G. Faster Lattice-Based KEMs via a Generic Fujisaki-Okamoto Transform Using Prefix Hashing. In Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security, Virtual, 15–19 November 2021; pp. 2722–2737.
20. Micciancio, D.; Regev, O. Lattice-Based Cryptography. In *Post-Quantum Cryptography*; Bernstein, D.J., Buchmann, J., Dahmen, E., Eds.; Springer: Berlin/Heidelberg, Germany, 2008; pp. 147–191.
21. Langlois, A.; Stehlé, D. Worst-case to average-case reductions for module lattices. *Des. Codes Cryptogr.* **2015**, *75*, 565–599. [CrossRef]
22. Alkim, E.; Ducas, L.; Pöppelmann, T.; Schwabe, P. Post-Quantum Key Exchange—A New Hope. In Proceedings of the 25th USENIX Security Symposium, Austin, TX, USA, 10–12 August 2016; pp. 327–343.
23. Chen, Y.; Nguyen, P.Q. BKZ 2.0: Better Lattice Security Estimates. In Proceedings of the ASIACRYPT 2011, Seoul, Republic of Korea, 4–8 December 2011; pp. 1–20.
24. Avanzi, R.; Bos, J.; Ducas, L.; Kiltz, E.; Lepoint, T.; Lyubashevsky, V.; Schanck, J.M.; Schwabe, P.; Seiler, G.; Stehlé, D. CRYSTALS-Kyber: Algorithm Specifications and Supporting Documentation (Version 3.0). In *NIST Post-Quantum Cryptography Standardization Process*; NIST: Gaithersburg, MD, USA, 2020.
25. Kirchner, P.; Fouque, P.A. Revisiting lattice attacks on overstretched NTRU parameters. In Proceedings of the EUROCRYPT 2017, Paris, France, 30 April–4 May 2017; pp. 3–26.
26. Ducas, L.; van Woerden, W.P.J. NTRU Fatigue: How Stretched Is Overstretched? In Proceedings of the ASIACRYPT 2021, Singapore, 6–10 December 2021; pp. 3–32.
27. Lyubashevsky, V.; Seiler, G. NTTRU: Truly Fast NTRU Using NTT. *IACR Trans. Cryptogr. Hardw. Embed. Syst.* **2019**, *2019*, 180–201. [CrossRef]
28. Basso, A.; Mera, J.M.B.; D'Anvers, J.P.; Karmakar, A.; Roy, S.S.; Beirendonck, M.V.; Vercauteren, F. Supporting Documentation: Saber: Mod-LWR Based KEM (Round 3 Submission). In *NIST Post-Quantum Cryptography Standardization Process*; NIST: Gaithersburg, MD, USA, 2020.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.