IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received December 22, 2020, accepted January 8, 2021, date of publication January 12, 2021, date of current version January 25, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3051047

Chosen-Ciphertext Secure Key Encapsulation Mechanism in the Standard Model

SHENGFENG XU[1] **AND XIANGXUE LI**[2,3,4]

[1]Department of Computer Science and Technology, East China Normal University, Shanghai 200062, China
[2]School of Software Engineering, East China Normal University, Shanghai 200062, China
[3]Shanghai Key Laboratory of Trustworthy Computing, East China Normal University, Shanghai 200062, China
[4]Westone Cryptologic Research Center, Beijing 100070, China

Corresponding author: Xiangxue Li (xxli@cs.ecnu.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61971192 and Grant 6191101004, and in part by the National Cryptography Development Fund under Grant MMJJ20180106.

ABSTRACT Key Encapsulation Mechanism (KEM) is a foundational cryptography primitive, which can provide secure symmetric cryptographic key material for transmission by using public key algorithms. Until now, many Chosen-Ciphertext (IND-CCA) secure KEM schemes are constructed from Chosen-Plaintext (IND-CPA) or One-Way (OW-CPA) secure PKE via the generic Fujisaki-Okamoto (FO) transformations (TCC 2017). However, the security relies on the Random Oracle Model (ROM). To the best of our knowledge, there are no IND-CCA secure KEM schemes based on Learning Parity with Noise (LPN) assumption that can against post quantum attacks in the standard model. In this work, we propose the first direct construction of LPN-based KEM, which is secure in the standard model. In particular, we use double-trapdoor technique to answer adversary's decryption queries correctly and a Target Collision Resistant (TCR) hash function to check the validity of the ciphertext. The encapsulated key is determined by a special LPN problem (with no random oracle required). The scheme is IND-CCA secure against post-quantum attacks under the low-noise LPN assumptions by a series of games and the security reduction is tight. Compared with previous schemes on 128-bit security level, our CCA-secure scheme only holds 50.78MB public keys, 62.50MB secret keys and 4.54KB ciphertexts, which is more efficient than the schemes of Döttling *et al.* (ASIACRYPT 2012), Kiltz *et al.* (PKC 2014) and Yu *et al.* (CRYPTO 2016) ((7.27GB, 7.24GB, 7.03KB), (80.89MB, 46.23MB, 6.80KB) and (70.95MB, 70.65MB, 86.50KB) respectively).

INDEX TERMS Key encapsulation mechanism, learning parity with noise, standard model, FO-like transformations.

I. INTRODUCTION

Learning Parity with Noise (LPN) problem is a significant researching area in cryptography academia. The computational version of LPN assumption with secret size $n \in \mathbb{N}$ and noise rate $0 < \mu < \frac{1}{2}$ postulates that given any $m = \Theta(n)$ (number of queries), it is computationally infeasible for any probabilistic polynomial-time (PPT) algorithm to recover the random secret $x \leftarrow \{0,1\}^n$ given $(A, A \cdot x + e)$, where $A \in \{0,1\}^{m \times n}$ is chosen uniformly at random, $e \in \{0,1\}^m$ is distributed to $\mathcal{B}_\mu^m$ (the concatenation of m independent copies of $\mathcal{B}_\mu$), $\mathcal{B}_\mu$ denotes the Bernoulli distribution with parameter μ (i.e., $\Pr[\mathcal{B}_\mu = 1] = \mu, \Pr[\mathcal{B}_\mu = 0] = 1 - \mu$), "$\cdot$" denotes matrix vector multiplication over GF(2) and "$+$" denotes the XOR operation. The decisional version of LPN simply assumes that $(A, A \cdot x + e)$ is pseudorandom. The two versions of LPN have been pointed out to be polynomially equivalent in [1]–[4]. LPN enjoys simplicity and suitability for weak-power devices (e.g.RFID tags) than other quantum-secure candidates such as Learning with Errors (LWE) [5]. The first cryptographic application of LPN is lightweight authentication [6]–[8].

LPN Hardness LPN has also been extensively studied in learning theory [9]–[11]. Under a constant noise rate, the most famous algorithms [7], [12] for solving LPN problem require $2^{\Theta(n/\log n)}$ time and samples. If one is restricted to $m = \Theta(n)$ samples, then the most famous attack [13] requires $2^{\Theta(n)}$ time. The best known solvers [14] for solving LPN problem need to use only $m = n^{1+\varepsilon}$ LPN samples when needed only

The associate editor coordinating the review of this manuscript and approving it for publication was S. K. Hafizul Islam.

$2^{\Theta(n/\log\log n)}$ time. Under a low noise rate $\mu = \Theta(\frac{1}{\sqrt{n}})$, if one is restricted to $m = \Theta(n)$ samples, then the best attack has exponential complexity $2^{\Theta(\sqrt{n})}$ [2], [15]–[17].

Key Encapsulation Mechanism From Low-Noise LPN: In Lepton, a Round-1 algorithm, Yu *et al.* [18] introduced two variants (the compact learning parity with noise problem and the ring variant of compact learning parity with noise problem) which are hard as standard LPN problem and proposed an IND-CPA secure key encapsulation mechanism from the ring variant of compact learning parity with noise problem. Yu *et al.* [18] constructed an IND-CCA secure key encapsulation mechanism by using FO transformation [19]–[21] to the IND-CPA secure key encapsulation mechanism. In [22], Cheng *et al.* constructed a multiple-recipient Key-Encapsulation Mechanism (mKEM) scheme in the Random Oracle Model (ROM) from low-noise LPN.

Key Encapsulation Mechanism With CCA Security: Intuitively, the KEM scheme can be simply regarded as the PKE scheme encrypting a random, except that the security models of the two schemes are different. There are many transformations (CHK transformation [23]–[25], FO transformation [19], [20] and so on) to construct an IND-CCA secure PKE scheme or an IND-CCA secure KEM scheme. In MP [23], the IND-CCA secure PKE scheme can be constructed via relatively generic transformations using either strongly unforgeable one-time signatures, or a message authentication code and weak form of commitment (BCHK [24]). In [25], Canetti *et al.* constructed an IND-CCA secure PKE scheme from any IND-CPA secure identity-based encryption (IBE) scheme by using CHK transformation. The CHK-transformation needs one-time signature increasing the ciphertext length.

For FO transformation, Fujisaki and Okamoto [19] proposed this transformation which turns any IND-CPA secure public-key encryption scheme into IND-CCA secure scheme in the random oracle model. Unfortunately, there are several drawbacks (e.g., non-tight security reduction and the need for a perfectly correct scheme) for the FO transformation. In [20], Hofheinz *et al.* provided a fine-grained and modular toolkit of FO transformations for turning IND-CPA secure schemes or One-Way (OW-CPA) secure schemes into IND-CCA secure public-key encryption schemes. Hofheinz *et al.* analyzed six different transformations $\mathrm{U}^{\perp}$, $\mathrm{U}^{\perp}_m$, $\mathrm{U}^{\not\perp}$, $\mathrm{U}^{\not\perp}_m$, $\mathrm{QU}^{\not\perp}_m$ and $\mathrm{QU}^{\perp}_m$, where $\perp$ ($\not\perp$) means explicit (implicit) rejection, m (without m) means $K = H(m)$ ($K = H(m, c)$) and Q means adding an additional hash value $H'(m)$ to the ciphertext. $\mathrm{QU}^{\not\perp}_m$ and $\mathrm{QU}^{\perp}_m$ ($\mathrm{U}^{\not\perp}$, $\mathrm{U}^{\not\perp}_m$, $\mathrm{U}^{\perp}$ and $\mathrm{U}^{\perp}_m$) all are in quantum random oracle model (in random oracle model (ROM)). These six transformations in [20] are robust against schemes with correctness errors and some transformations which are added an additional hash value to the ciphertext be analyzed in the Quantum Random Oracle Model (QROM). For two widely used generic transformations $\mathrm{FO}^{\not\perp}$ and $\mathrm{FO}^{\not\perp}_m$, Jiang *et al.* showed QROM security reductions without adding any ciphertext overhead in [26].

TABLE 1. Transformation and correctness error of IND-CCA secure schemes.

Proposal	Transformation	Correctness error
CRYSTALS-Kyber	$\mathrm{U}^{\perp}$	✓
Classic McEliece	$\mathrm{U}^{\perp}$	✗
SABER	$\mathrm{U}^{\perp}$	✓
NTRU-HRSS-KEM	$\mathrm{QU}^{\perp}_m$	✗
FrodoKEM	$\mathrm{QU}^{\not\perp}_m$	✓
HQC	$\mathrm{QU}^{\perp}_m$	✓
SIKE [18]	$\mathrm{U}^{\perp}$	✗
MP [23]	DDN [28] or BCHK [24]	✓
PW [29]	—	✗
Ours	—	✓

In addition, there are many IND-CCA secure KEM schemes. Park *et al.* [27] constructed an IND-CCA secure KEM by using FO transformation [20], [26] from an IND-CCA secure PKE scheme. In July 2020, the National Institute of Standards and Technology (NIST) announced 15 Round-3 algorithms, which includes 7 Round-3 finalists and 8 Round-3 alternate candidates. Among these 15 Round-3 algorithms, there are 7 Round-3 algorithms (i.e., CRYSTALS-Kyber, Classic McEliece, SABER, NTRU-HRSS-KEM, FrodoKEM, HQC and SIKE [18]) for KEM constructions. From Table 1, we noticed that there are eight schemes using some conventional transformations (e.g., $\mathrm{U}^{\perp}$, $\mathrm{QU}^{\perp}_m$, $\mathrm{QU}^{\not\perp}_m$, DDN and BCHK) and six schemes have correctness errors. These 7 Round-3 algorithms are constructed by using generic transformations from IND-CPA secure schemes or OW-CPA secure schemes.

A. OUR CONTRIBUTION

In this work, we construct the first IND-CCA secure KEM scheme based on variants of low-noise LPN assumption [10], [30] in the standard model, the main techniques are a target collision resistant hash function and a double-trapdoor function. The target collision resistant hash function is used to check the validity of the ciphertext and the double-trapdoor technique is used to answer adversary's decryption queries correctly, so that our scheme can satisfy CCA security. Our construction does not rely on generic FO transformations, and the encapsulated key in our construction is determined by the LPN problem in [30] rather than a hash function.

The scheme is IND-CCA secure in the standard model under the low-noise LPN assumptions and the security reduction is tight. Compared with some lattice-based KEM schemes (e.g., CRYSTALS-Kyber, FrodoKEM [18] and so on), our scheme only involves operations over GF(2) (not GF(q)), which could be much faster when implemented using hardware. In addition, compared with the work [31], our construction based on low-noise LPN problem is secure against post-quantum attacks in standard model. Consider the performance on 128-bit security level, our CCA-secure scheme only holds 50.78MB public keys, 62.50MB secret keys and 4.54KB ciphertexts, which is more efficient than the schemes of Döttling *et al.* [10], Kiltz *et al.* [32] and Yu *et al.* [13] ((7.27GB, 7.24GB, 7.03KB), (80.89MB, 46.23MB, 6.80KB) and (70.95MB, 70.65MB, 86.50KB) respectively).

II. PRELIMINARIES

We denote that the column vector is s, and the row vector that can be regarded as the transpose of the column vector is represented by the lowercase letter s^{T}. The matrix is represented by capital letters A, and $|s|$ refers to the Hamming weight of the column vector s. $\mathcal{B}_\mu$ denotes the Bernoulli distribution with parameter μ (i.e., $\Pr[\mathcal{B}_\mu = 1] = \mu, \Pr[\mathcal{B}_\mu = 0] = 1 - \mu$) and $\mathcal{B}_\mu^m$ denotes the concatenation of m independent copies of $\mathcal{B}_\mu$. $\mathcal{B}_\mu^{q\times m}$ denotes a matrix distribution whose every column is i.i.d. to $\mathcal{B}_\mu^q$. $x \xleftarrow{\$} \chi$ means sampling x from distribution χ uniformly at random, and $x \longleftarrow \chi$ means sampling x according to distribution χ. We use U_m to denote a uniform distribution over $\{0, 1\}^m$. Let $negl(n)$ denotes a negligible function in n.

Definition 1 (Learning Parity with Noise): For $n \in \mathbb{N}$, $m = \Theta(n)$, $0 < \mu < \frac{1}{2}$, the decisional (n, μ)-LPN problem is hard if for every PPT algorithm $\mathcal{D}$ we have

$$|\Pr[\mathcal{D}(A, Ax+e)=1] - \Pr[\mathcal{D}(A, U_m)=1]| = \mathrm{negl}(n), \qquad (1)$$

where $A \xleftarrow{\$} \{0, 1\}^{m\times n}$, $e \xleftarrow{\$} \mathcal{B}_\mu^m$ and $x \xleftarrow{\$} \{0, 1\}^n$. The computational (n, μ)-LPN problem with the same n and μ is hard if for every PPT algorithm $\mathcal{D}$ we have

$$\Pr[\mathcal{D}(A, Ax + e) = x] = \mathrm{negl}(n), \qquad (2)$$

where $A \xleftarrow{\$} \{0, 1\}^{m\times n}$, $e \xleftarrow{\$} \mathcal{B}_\mu^m$ and $x \xleftarrow{\$} \{0, 1\}^n$. In some scenarios such as PKE, the problems further put a constraint on the number of samples, m, and thus might be written as (n, μ, m)-LPN problems.

Concrete Hardness: For $T := T(n)$, we say that the decisional/computational (n, μ, m)-LPN is T-hard if every probabilistic adversary of running time T the distinguishing (resp., inverting) advantage in (1) (resp., (2)) is upper bounded by $1/T$. For (n, μ)-LPN problem (whose m is unspecified), animplicit constraint is $m \leq T$.

Lemma 1 (Lemma 2.10 from [30]): For $l \in \mathbb{N}$, $n = 2l$, $c > 0$, $\mu = \sqrt{\frac{c}{n}}$, $(A, y^{\mathrm{T}}A)$ is computationally indistinguishable from (A, r) where $A \xleftarrow{\$} \{0, 1\}^{n\times(l+1)}$, $y \xleftarrow{\$} \mathcal{B}_\mu^n$ and $r \xleftarrow{\$} \{0, 1\}^{(l+1)}$.

Proof: Under the decisional (n, μ)-LPN assumption, $(A, y^{\mathrm{T}}A)$ is computationally indistinguishable from (A, r). This reduction can be refernced to [30].

Definition 2 (VLPN [10]): For $n \in \mathbb{N}$, $m = \Theta(n)$, $c > 0$, $\mu = \sqrt{\frac{c}{n}}$, the decisional (n, μ)-VLPN problem is hard if for every PPT algorithm $\mathcal{D}$ we have

$$|\Pr[\mathcal{D}(A, Ax + e) = 1] - \Pr[\mathcal{D}(A, U_m) = 1]| = \mathrm{negl}(n),$$

where $A \xleftarrow{\$} \{0, 1\}^{m\times n}$, $e \xleftarrow{\$} \mathcal{B}_\mu^m$ and $x \xleftarrow{\$} \mathcal{B}_\mu^n$. The computational (n, μ)-VLPN problem with the same n and μ is hard if for every PPT algorithm $\mathcal{D}$ we have

$$\Pr[\mathcal{D}(A, Ax + e) = x] = \mathrm{negl}(n),$$

where $A \xleftarrow{\$} \{0, 1\}^{m\times n}$, $e \xleftarrow{\$} \mathcal{B}_\mu^m$ and $x \xleftarrow{\$} \mathcal{B}_\mu^n$.

The *decisional* (n, μ)-VLPN problem to *decisional* (n, μ)-LPN problem reduction can be referenced to [10]. As on can see, a matrix-version of the *decisional* (n, μ)-VLPN problem is also hard by using a simple hybrid argument technique.

Definition 3 (KLPN [33]): For $n \in \mathbb{N}$, $c > 0$, $\mu = \sqrt{\frac{c}{n}}$ and $n < m$, the decisional (n, μ)-KLPN problem is hard if for every PPT algorithm $\mathcal{D}$ we have

$$|\Pr[\mathcal{D}(A, GA\ ) = 1] - \Pr[\mathcal{D}(A, U_{n\times m}) = 1]| = \mathrm{negl}(n),$$

where $A \xleftarrow{\$} \{0, 1\}^{m\times n}$ and $G \xleftarrow{\$} \mathcal{B}_\mu^{m\times m}$.
The decisional (n, μ)-KLPN problem is hard as the decisional (n, μ)-LPN problem in [32].

Definition 4 (Target Collision Resistant Hash Functions): For $n \in \mathbb{N}$, $m = \Theta(n)$, $s = \Theta(n)$ and $m > n$, a family of functions $\mathcal{H} = \{h_k : \{0, 1\}^m \longrightarrow \{0, 1\}^n, k \in \{0, 1\}^s\}$ is a target collision resistant hash function (TCR) if for every PPT adversary $\mathcal{A}$, it holds that

$$\Pr\left[h_k(x') = h_k(x) \wedge x' \neq x : \begin{array}{r} k \xleftarrow{\$} \{0, 1\}^s \\ x \xleftarrow{\$} \{0, 1\}^m \\ x' \longleftarrow \mathcal{A}(k, x) \end{array} \right] = \mathrm{negl}(n).$$

Noted that target collision resistant hash function (TCR) is a weaker notion than collision resistant function, so that any practical collision resistant function can be used to construct TCR.

Lemma 2 (Chernoff Bound [10]): If $d \xleftarrow{\$} \mathcal{B}_\mu^m$ and $\delta > 0$, it holds that

$$\Pr[|d| > (1 + \delta)\mu m] \leq e^{\frac{-\min(\delta,\delta^2)}{3}\mu m}.$$

Lemma 3: For constants $0 < c < \frac{1}{12}$, $12c < \alpha < 1$ and $k \in \mathbb{N}$, let $n = \Theta(k^2)$, $m = 2n$, $\mu = \sqrt{c/n}$, $s \xleftarrow{\$} \mathcal{B}_\mu^n$, $e \xleftarrow{\$} \mathcal{B}_\mu^m$, $S \xleftarrow{\$} \mathcal{B}_\mu^{m\times m}$ and $E \xleftarrow{\$} \mathcal{B}_\mu^{n\times m}$, we have $\Pr[|S^{\mathrm{T}}e| > \frac{\alpha}{3}m \mid |e| \leq 2\mu m] < 2^{-\Theta(m)}$ and $\Pr[|E^{\mathrm{T}}s| > \frac{\alpha}{6}m \mid |s| \leq 2\mu n] < 2^{-\Theta(m)}$.

Proof: Assuming that the Hamming weight of e does not exceed $2\mu m$, we analyze the inner product $s^{\mathrm{T}}e$. It's not difficult to see that a necessary condition for $s^{\mathrm{T}}e = 1$ is that $s[i] = 1$ for at least one of the i's where $e[i] = 1$. We use this in the second step (eq. (4)), and the union bound in the third step (eq. (5)). So it holds that

$$\begin{aligned} \mu' : &= \Pr[s^{\mathrm{T}}e = 1 \mid |e| \leq 2\mu m] && (3)\\ &\leq \Pr[\exists i : (e[i] = 1) \wedge (s[i] = 1) \mid |e| \leq 2\mu m] && (4)\\ &\leq \mu \cdot 2\mu m = 4c < \frac{1}{3}\alpha. && (5) \end{aligned}$$

We have with $\delta := \frac{\alpha}{3\mu'} - 1$ (note that $\mu' \leq 4c < \frac{1}{3}\alpha$, $(1 + \delta)\mu' = \frac{1}{3}\alpha$)

$$\Pr[|S^{\mathrm{T}}e| > \frac{1}{3}\alpha m \mid |e| \leq 2\mu m] < e^{-\min(\delta,\delta^2)\mu' m/3}$$

by using the Chernoff bound.

Next, $\delta = \frac{\alpha}{3\mu'} - 1 \geq \frac{\alpha}{12c} - 1 > 0$ and $\delta\mu' = \frac{1}{3}\alpha - \mu' > 4c - \mu' \geq 0$ are lower bounded by constants and therefore

$$\Pr[|S^{\mathrm{T}}e| > \frac{1}{3}\alpha m \mid |e| \leq 2\mu m] < e^{-\min(\delta,\delta^2)\mu' m/3} = 2^{-\Theta(m)}.$$

Obviously, we can also prove that $\Pr[|E^{\mathrm{T}}s| > \frac{1}{6}\alpha m \mid |s| \leq 2\mu n] < 2^{-\Theta(m)}$ in a similar way. We omit these details.

III. KEY ENCAPSULATION MECHANISM

A key encapsulation mechanism consists of three algorithms KEM=(Gen,Encaps,Decaps). The key generation algorithm Gen(1^k) takes the security parameter 1^k as input and returns a pair of public key pk and private key sk. On input a public key pk, the key encapsulation algorithm Encaps(pk) returns a ciphertext C and a key k, where k is contained in the key space $\mathcal{K}$. On input a ciphertext C and the private key sk, the deterministic decapsulation algorithm Decaps(sk, C) returns a key $k :=$ Decaps(sk, C) or a failure symbol $\perp$. We say that the KEM scheme is correct, if for all public key pk, it holds that

$$\Pr\left[\mathrm{Decaps}(sk, C) \neq k : \begin{array}{c}(pk, sk) \longleftarrow \mathrm{Gen}(1^k) \\ (k, C) \longleftarrow \mathrm{Encaps}(pk)\end{array}\right] \leq \mathrm{negl}(k).$$

We recall the IND-CCA secure notions for KEM. Let $\mathcal{A}$ denotes a PPT adversary.

Definition 5 (IND-CCA KEM [20], [26]): A KEM scheme $\mathcal{KEM}$ is IND-CCA secure, if for any PPT adversary $\mathcal{A}$, the advantage of $\mathcal{A}$ is negligible in the experiment defined as below:

1. *Generate a pair of keys* $(pk, sk) \longleftarrow$ Gen(1^k). *The public key pk is given to* $\mathcal{A}$.
2. $b^* \in \{0, 1\}$ *is sampled randomly, and a pair of challenge ciphertext and key* $(C^*, k_1^*) \longleftarrow$ Encaps(pk) *are generated. The key* $k_0^* \in \mathcal{K}$ *is sampled at random.* $\mathcal{A}$ *gets* (C^*, k_b^*) *and computes its computation access to the decapsulation-oracle Decaps* $(sk, \cdot)$.
3. *Finally,* $\mathcal{A}$ *outputs a guessing* b'. *The advantage of* $\mathcal{A}$ *is defined as* $Adv_{KEM,\mathcal{A}}^{ind-cca}(1^k) := |\Pr[b' = b^*] - 1/2|$.

IV. CCA SECURE KEM FROM LOW-NOISE LPN

In this section, we directly construct an IND-CCA secure key encapsulation mechanism scheme from low-noise LPN.

A. OUR CONSTRUCTION

Our construction follows the general structure of previous works in [5], [32]. Let k be the security parameter, let $0 < c < \frac{1}{12}$, $l = \Theta(k^2)$, $n = 2l$, $m = 4l$ be any constants, let α be a constant with $12c < \alpha < 1$ and let $\mu = \sqrt{\frac{c}{n}}$ denotes the noise rate. An error correction code ECC : $\{0, 1\}^n \longrightarrow \{0, 1\}^m$ can encode n bit message and has an efficient decoding algorithm ECC^{-1}, which can correct up to αm bit errors. The error correction code can be constructed [33], [34]. Let $\mathcal{H} : \{0, 1\}^{3m+n} \longrightarrow \{0, 1\}^{(l+1)}$ is a family of target collision resistant hash functions. We show the construction of our $\mathcal{KEM}$ as follows:

Gen(1^k): On input a security parameter k, it uniformly chooses matrices $A \xleftarrow{\$} U_{m\times n}$, $S_0, S_1 \xleftarrow{\$} \mathcal{B}_{\mu}^{m\times m}$, $E_0, E_1 \xleftarrow{\$} \mathcal{B}_{\mu}^{n\times m}$ and $F \xleftarrow{\$} U_{(l+1)\times n}$, and computes $B = S_0^{\mathrm{T}}A + E_0^{\mathrm{T}}$, $D = S_1^{\mathrm{T}}A + E_1^{\mathrm{T}}$. It returns $(pk, sk) = ((A, B, D, F), (S_0, S_1))$.

Encaps(pk): On input the public key $pk = (A, B, D, F)$, it samples matrices $S_0', S_1' \xleftarrow{\$} \mathcal{B}_{\mu}^{m\times m}$, $E_0', E_1' \xleftarrow{\$} \mathcal{B}_{\mu}^{n\times m}$, $s \xleftarrow{\$} \mathcal{B}_{\mu}^{n}$ and $e \xleftarrow{\$} \mathcal{B}_{\mu}^{m}$. Then, it outputs $(k, C := (c_0, c_1, c_2))$, where

$$\begin{aligned} c_0 &:= As + e \quad \in \{0, 1\}^m \\ c_1 &:= Bs + (S_0')^{\mathrm{T}}e - (E_0')^{\mathrm{T}}s + \mathrm{ECC}(s) \quad \in \{0, 1\}^m \\ c_2 &:= Ds + (S_1')^{\mathrm{T}}e - (E_1')^{\mathrm{T}}s + \mathrm{ECC}(s) \quad \in \{0, 1\}^m \\ t &:= \mathcal{H}(c_0, c_1, c_2, s) \quad \in \{0, 1\}^{(l+1)} \\ k &:= Fs + t \quad \in \{0, 1\}^{(l+1)}. \end{aligned}$$

Decaps(sk, C): On input the secret key $sk = (S_0, S_1)$ and a ciphertext (c_0, c_1, c_2), it computes $c_1 - S_0^{\mathrm{T}}c_0 = \mathrm{ECC}(s) + (S_0' - S_0)^{\mathrm{T}}e + (E_0 - E_0')^{\mathrm{T}}s$ and uses the decoding algorithm ECC^{-1} to reconstruct s. If it holds that

$$\begin{aligned} &|s| \leq 2\mu n \\ &\quad \wedge |c_0 - As| \leq 2\mu m \\ &\quad \wedge |c_1 - Bs - \mathrm{ECC}(s)| \leq \frac{1}{2}\alpha m \\ &\quad \wedge |c_2 - Ds - \mathrm{ECC}(s)| \leq \frac{1}{2}\alpha m, \end{aligned} \tag{6}$$

then it computes $t := \mathcal{H}(c_0, c_1, c_2, s)$ and sets $k := Fs + t$, else it lets $k :=\perp$. Finally, the output is k.

Remark 1: As one can see, the matrix S_1 in the secret key $sk = (S_0, S_1)$ can also be used to decrypt the ciphertext. we can also use the decoding algorithm ECC^{-1} *to reconstruct s from $c_2 - S_1^{\mathrm{T}}c_0$. if it holds that*

$$\begin{aligned} &|s| \leq 2\mu n \\ &\quad \wedge |c_0 - As| \leq 2\mu m \\ &\quad \wedge |c_1 - Bs - \mathrm{ECC}(s)| \leq \frac{1}{2}\alpha m \\ &\quad \wedge |c_2 - Ds - \mathrm{ECC}(s)| \leq \frac{1}{2}\alpha m, \end{aligned}$$

then it computes $t := \mathcal{H}(c_0, c_1, c_2, s)$ and lets $k := Fs + t$, else it sets $k =\perp$. At last, it returns k.

Theorem 1 (Correctness and Equivalence of the Secret Keys S_0, S_1): The $\mathcal{KEM}$ is correct since we choose appropriate choice of parameters. The secret keys S_0 and S_1 are equivalent during decapsulation phase.

Proof: For $e \xleftarrow{\$} \mathcal{B}_{\mu}^{m}$ and $s \xleftarrow{\$} \mathcal{B}_{\mu}^{n}$, it holds that

$$\Pr[|e| \leq 2\mu m] \geq 1 - 2^{-\Theta(\sqrt{m})} \tag{7}$$

$$\Pr[|s| \leq 2\mu n] \geq 1 - 2^{-\Theta(\sqrt{n})} \tag{8}$$

by using Chernoff bound.

As C is a properly generated ciphertext, it holds that

$$|s| \leq 2\mu n$$

$$
\begin{aligned}
&\wedge\ |e| \leq 2\mu m\\
&\wedge\ |E_1^{\mathrm{T}} s| \leq \tfrac{1}{6}\alpha m\\
&\wedge\ |E_0^{\mathrm{T}} s| \leq \tfrac{1}{6}\alpha m\\
&\wedge\ |S_0^{\mathrm{T}} e| \leq \tfrac{1}{3}\alpha m\\
&\wedge\ |S_1^{\mathrm{T}} e| \leq \tfrac{1}{3}\alpha m
\end{aligned}
$$

with overwhelming probability $1 - 2^{-\Theta(\sqrt{n})}$ by using Equations (7),(8), Lemma 3 and union bound, where $S_0, S_1 \xleftarrow{\$} \mathcal{B}_\mu^{m\times m}$, $E_0, E_1 \xleftarrow{\$} \mathcal{B}_\mu^{n\times m}$, $s \xleftarrow{\$} \mathcal{B}_\mu^{n}$ and $e \xleftarrow{\$} \mathcal{B}_\mu^{m}$. We can decode the correct s from $c_1 - S_0^{\mathrm{T}} c_0$ by using the decoding algorithm ECC^{-1}, where the error term satisfies $|(S_0' - S_0)^{\mathrm{T}} e + (E_0 - E_0')^{\mathrm{T}} s| \leq \alpha m$. The consistency check Equations (6) will pass. Then it computes $t := \mathcal{H}(c_0, c_1, c_2, s)$ and returns the correct key $k := Fs + t$. In the following lemma, we will show that S_0 and S_1 have the same decapsulation ability with overwhelming probability.

Lemma 4: Defined $\mathrm{Decaps}(sk, C)_{S_1}$ *as a decapsulation algorithm that decapsulation using the secret key* S_1 *to reconstruct s. Let* $\mathrm{Decaps}(sk, C)_{S_0} := \mathrm{Decaps}(sk, C)$. $\mathrm{Decaps}(sk, C)_{S_1}$ *and* $\mathrm{Decaps}(sk, C)_{S_0}$ *return the same k with overwhelming probability over the choice of parameters. Then, we get*

$$\Pr_{pk,sk}[\forall C : \mathrm{Decaps}(sk, C)_{S_1} \neq \mathrm{Decaps}(sk, C)_{S_0}] < 2^{-\Theta(m)}.$$

Proof: If $k =: \mathrm{Decaps}(sk, C)_{S_0}$, we can reconstruct s. By the consistency check Equations (6), we get

$$
\begin{aligned}
|s| &\leq 2\mu n\\
&\wedge\ |c_0 - As| \leq 2\mu m\\
&\wedge\ |c_1 - Bs - \mathrm{ECC}(s)| \leq \frac{1}{2}\alpha m\\
&\wedge\ |c_2 - Ds - \mathrm{ECC}(s)| \leq \frac{1}{2}\alpha m.
\end{aligned}
$$

We know that $\mathrm{Decaps}(sk, C)_{S_1}$ reconstructs the same s by using the decoding algorithm ECC^{-1}, where the error term satisfies $|(S_1' - S_1)^{\mathrm{T}} e + (E_1 - E_1')^{\mathrm{T}} s| \leq \alpha m$. We know that $|S_1'^{\mathrm{T}} e - E_1'^{\mathrm{T}} s| \leq \frac{1}{2}\alpha m$. If $|E_1^{\mathrm{T}} s - S_1^{\mathrm{T}} e| \leq \frac{1}{2}\alpha m$, then the decoding algorithm ECC^{-1} can construct the same s by using triangle inequality. If the same e and s satisfying $|e| \leq 2\mu m$, $|s| \leq 2\mu n$ at random, then we have

$$
\begin{aligned}
\Pr_{e,|e|\leq 2\mu m, S_1}[|S_1 e| \leq \frac{1}{3}\alpha m] &\geq 1 - 2^{-\Theta(m)}\\
\Pr_{s,|s|\leq 2\mu n, E_1}[|E_1 s| \leq \frac{1}{6}\alpha m] &\geq 1 - 2^{-\Theta(m)}
\end{aligned}
$$

by using Lemma 3. We can get $|E_1^{\mathrm{T}} s - S_1^{\mathrm{T}} e| \leq \frac{1}{2}\alpha m$ by using triangle inequality. We notice that e and s are not a randomly chosen one. Taking the union bound over all $2^{\log(m)\omega(\sqrt{m})}$ possible e satisfying $|e| \leq 2\mu m$, we have

$$\Pr_{S_1}[|S_1 e| \leq \frac{1}{3}\alpha m] \geq 1 - 2^{-\Theta(m)+\log(m)\omega(\sqrt{m})} = 1 - 2^{-\Theta(m)}.$$

In the similar way, taking the union bound over all $2^{\log(n)\omega(\sqrt{n})}$ possible s satisfying $|s| \leq 2\mu n$, we have

$$\Pr_{E_1}[|E_1 s| \leq \frac{1}{6}\alpha m] \geq 1 - 2^{-\Theta(m)+\log(n)\omega(\sqrt{n})} = 1 - 2^{-\Theta(m)}.$$

This shows that the same k is returned by $\mathrm{Decaps}(sk, C)_{S_1}$ with overwhelming probability over the choice of S_1 and E_1. $\mathrm{Decaps}(sk, C)_{S_0}$ return the same k with overwhelming probability over the choice of parameters.

Theorem 2 (Security): Assume that $\mathcal{H}$ *is a target collision resistant hash function. If the decisional* (n, μ)*-VLPN problem is hard, our* $\mathcal{KEM}$ *is IND-CCA secure by using Lemma 1.*

Proof: Assumed that $\mathcal{A}$ is a PPT adversary against the IND-CCA security of our scheme. by defining a sequence of games and showing adjacent games indistinguishable, we can prove that the advantage $Adv_{KEM,\mathcal{A}}^{ind-cca}(1^k)$ is negligible in k. We use $\Pr[R_i]$ to denote the probability of a particular event that the adversary $\mathcal{A}$ wins in Game i.

Game 0: This is the IND-CCA security experiment for the KEM scheme. The challenger $\mathcal{C}$ honestly runs the adversary $\mathcal{A}$ with the security parameter k. The challenger $\mathcal{C}$ simulates the IND-CCA security game for $\mathcal{A}$ as follows:

Gen: The challenger $\mathcal{C}$ uniformly chooses matrices $A \xleftarrow{\$} U_{m\times n}$, $S_0, S_1 \xleftarrow{\$} \mathcal{B}_\mu^{m\times m}$, $E_0, E_1 \xleftarrow{\$} \mathcal{B}_\mu^{n\times m}$ and $F \xleftarrow{\$} U_{(l+1)\times n}$, and computes $B = S_0^{\mathrm{T}} A + E_0^{\mathrm{T}}, D = S_1^{\mathrm{T}} A + E_1^{\mathrm{T}}$. The challenger $\mathcal{C}$ sends $pk = (A, B, D, F)$ to the adversary $\mathcal{A}$, and keeps $sk = (S_0, S_1)$ to itself.

Challenge: The challenger $\mathcal{C}$ chooses matrices $S_0', S_1' \xleftarrow{\$} \mathcal{B}_\mu^{m\times m}$, $E_0', E_1' \xleftarrow{\$} \mathcal{B}_\mu^{n\times m}$, $s \xleftarrow{\$} \mathcal{B}_\mu^{n}$, $e \xleftarrow{\$} \mathcal{B}_\mu^{m}$ and $k_0^* \xleftarrow{\$} \mathcal{K}$, and chooses a bit $b^* \in \{0, 1\}$ at random. The challenger $\mathcal{C}$ sends $(k_{b^*}^*, (c_0^*, c_1^*, c_2^*))$ to the adversary $\mathcal{A}$, where

$$
\begin{aligned}
c_0^* &:= As + e \quad \in \{0, 1\}^m\\
c_1^* &:= Bs + (S_0')^{\mathrm{T}} e - (E_0')^{\mathrm{T}} s + \mathrm{ECC}(s) \quad \in \{0, 1\}^m\\
c_2^* &:= Ds + (S_1')^{\mathrm{T}} e - (E_1')^{\mathrm{T}} s + \mathrm{ECC}(s) \quad \in \{0, 1\}^m\\
t^* &:= \mathcal{H}(c_0^*, c_1^*, c_2^*, s) \quad \in \{0, 1\}^{(l+1)}\\
k_1^* &:= Fs + t^* \quad \in \{0, 1\}^{(l+1)}.
\end{aligned}
$$

Phase 1: When challenger gets a decapsulation query (c_0, c_1, c_2) from the adversary, the challenger returns $\perp$ to the adversary if $(c_0, c_1, c_2) = (c_0^*, c_1^*, c_2^*)$. Otherwise, the challenger firstly computes $c_1 - S_0^{\mathrm{T}} c_0 = \mathrm{ECC}(s) + (S_0' - S_0)^{\mathrm{T}} e + (E_0 - E_0')^{\mathrm{T}} s$ and uses the decoding algorithm ECC^{-1} to reconstruct s. Let $t := \mathcal{H}(c_0, c_1, c_2, s)$. If it holds that

$$
\begin{aligned}
|s| &\leq 2\mu n\\
&\wedge\ |c_0 - As| \leq 2\mu m\\
&\wedge\ |c_1 - Bs - \mathrm{ECC}(s)| \leq \tfrac{1}{2}\alpha m\\
&\wedge\ |c_2 - Ds - \mathrm{ECC}(s)| \leq \tfrac{1}{2}\alpha m,
\end{aligned}
$$

then there are two different cases as follows:

Case 1: $t = t^*$ *and* $(c_0, c_1, c_2, s) \neq (c_0^*, c_1^*, c_2^*, s)$*: In this case, the challenger* $\mathcal{C}$ *found a collision* $(c_0, c_1, c_2, s) \neq (c_0^*, c_1^*, c_2^*, s)$ *that satisfies* $\mathcal{H}(c_0, c_1, c_2, s) = \mathcal{H}(c_0^*, c_1^*, c_2^*, s)$. *The challenger* $\mathcal{C}$ *sends* $\perp$ *to the adversary* $\mathcal{A}$.

Case 2: $t \neq t^*$: In this case, the challenger $\mathcal{C}$ computes $k := Fs + t$ and returns k to the adversary $\mathcal{A}$.
Otherwise, it sets $k := \perp$. The challenger $\mathcal{C}$ sends k to the adversary $\mathcal{A}$.

Guess: Eventually, the adversary $\mathcal{A}$ outputs a guess b'. If $b' = b^*$, the challenger $\mathcal{C}$ outputs 1, otherwise it outputs 0.

Lemma 5: $|\Pr[R_0] - \frac{1}{2}| = Adv^{ind-cca}_{KEM,\mathcal{A}}(1^k)$.

Proof: The challenger $\mathcal{C}$ honestly runs the adversary $\mathcal{A}$, and outputs 1 if and only if $b' = b^*$.

Game 1: It is the same as Game 0, except that, the challenger $\mathcal{C}$ changes the key generation phase as follows:

KeyGen: The challenger $\mathcal{C}$ uniformly choose matrices $A \xleftarrow{\$} U_{m\times n}$, $S_0, S_1 \xleftarrow{\$} \mathcal{B}_\mu^{m\times m}$, $E_0, E_1 \xleftarrow{\$} \mathcal{B}_\mu^{n\times m}$, $F \xleftarrow{\$} U_{(l+1)\times n}$ and $D' \xleftarrow{\$} U_{m\times n}$, and computes $B = S_0^{\mathrm{T}}A + E_0^{\mathrm{T}}, D = S_1^{\mathrm{T}}A + E_1^{\mathrm{T}}$. The challenger $\mathcal{C}$ returns $pk = (A, B, D', F)$ to the adversary $\mathcal{A}$, and keeps $sk = (S_0, S_1)$ to itself.

Lemma 6: $|\Pr[R_1] - \Pr[R_0]| \leq \mathrm{negl}(n)$.

Proof: It is different that the challenger $\mathcal{C}$ replaces $D = S_1^{\mathrm{T}}A + E_1^{\mathrm{T}}$ in public key in Game 0 with $D' \xleftarrow{\$} U_{m\times n}$ in public key in Game 1. Under the matrix-version of the *decisional* (n, μ)-VLPN assumption, $pk = (A, B, D)$ in Game 0 is computationally indistinguishable from $pk = (A, B, D')$ in Game 1. Consequently, we have that $|\Pr[R_1] - \Pr[R_0]| \leq \mathrm{negl}(n)$.

Game 2: It is the same as game 1, except that, the challenger $\mathcal{C}$ changes the Challenge phase as follows:

Challenge: The challenger $\mathcal{C}$ chooses $S_0', S_1' \xleftarrow{\$} \mathcal{B}_\mu^{m\times m}$, $E_0', E_1' \xleftarrow{\$} \mathcal{B}_\mu^{n\times m}$, $s \xleftarrow{\$} \mathcal{B}_\mu^{n}$, $e \xleftarrow{\$} \mathcal{B}_\mu^{m}$ and $k_0^* \xleftarrow{\$} \mathcal{K}$, and samples a bit $b^* \in \{0, 1\}$ at random. Finally, the challenger $\mathcal{C}$ sends $(k_{b^*}^*, (c_0^*, c_1^*, c_2^*))$ to the adversary $\mathcal{A}$, where

$$\begin{aligned}
c_0^* &:= As + e && \in \{0,1\}^m\\
c_1^* &:= Bs + (S_0')^{\mathrm{T}}e - (E_0')^{\mathrm{T}}s + \mathrm{ECC}(s) && \in \{0,1\}^m\\
c_2^* &:= Ds + S_1^{\mathrm{T}}e - E_1^{\mathrm{T}}s + \mathrm{ECC}(s) && \in \{0,1\}^m\\
t^* &:= \mathcal{H}(c_0^*, c_1^*, c_2^*, s) && \in \{0,1\}^{(l+1)}\\
k_1^* &:= Fs + t^* && \in \{0,1\}^{(l+1)}.
\end{aligned}$$

Lemma 7: $\Pr[R_2] = \Pr[R_1]$.

Proof: It is different that S_1 and E_1 in the key generation phase are sampled from the same distribution as S_1' and E_1' in the challenge phase respectively. Noted that $D = S_1^{\mathrm{T}}A + E_1^{\mathrm{T}}$ is not included in the public key, so the adversary has not obtained any information about S_1 and E_1 before the challenge phase. The challenge ciphertext in Game 1 has the same distribution as the challenge ciphertext in Game 2. In all, Game 1 and Game 2 are identical from the adversary's view. So, we have that $\Pr[R_2] = \Pr[R_1]$.

Game 3: It is the same as Game 2, except that, the challenger $\mathcal{C}$ changes the key generation phase as follows:

KeyGen: The challenger $\mathcal{C}$ uniformly chooses matrices $A \xleftarrow{\$} U_{m\times n}$, $S_0, S_1 \xleftarrow{\$} \mathcal{B}_\mu^{m\times m}$, $E_0, E_1 \xleftarrow{\$} \mathcal{B}_\mu^{n\times m}$ and $F \xleftarrow{\$} U_{(l+1)\times n}$, and computes $B = S_0^{\mathrm{T}}A + E_0^{\mathrm{T}}, D = S_1^{\mathrm{T}}A + E_1^{\mathrm{T}}$. The challenger $\mathcal{C}$ sends $pk = (A, B, D, F)$ to the adversary $\mathcal{A}$, and keeps $sk = (S_0, S_1)$ to itself.

Lemma 8: $|\Pr[R_3] - \Pr[R_2]| \leq \mathrm{negl}(n)$.

Proof: It is different that the challenger $\mathcal{C}$ replaces $D' \xleftarrow{\$} U_{m\times n}$ in public key in Game 2 with $D = S_1^{\mathrm{T}}A + E_1^{\mathrm{T}}$ in public key in Game 3. Under the matrix-version of the *decisional* (n, μ)-VLPN assumption, $pk = (A, B, D')$ in Game 2 is computationally indistinguishable from $pk = (A, B, D)$ in Game 3. So, we have that $|\Pr[R_3] - \Pr[R_2]| \leq \mathrm{negl}(n)$.
Note that c_2^* in target ciphertext (c_0^*, c_1^*, c_2^*) is equal to $S_1^{\mathrm{T}}c_0^* + \mathrm{ECC}(s)$.

Game 4: It is the same as Game 3, except that, the challenger $\mathcal{C}$ answers decapsulation queries for any (c_0, c_1, c_2) by using S_1 instead of S_0.

Lemma 9: $|\Pr[R_4] - \Pr[R_3]| \leq \mathrm{negl}(n)$.

Proof: As one can see, S_1 and S_0 have equivalent decryption ability except with negligible probability by using Lemma 4.

Game 5: It is the same as Game 4, except that, the challenger $\mathcal{C}$ changes the challenge phase as follows:

Challenge: The challenger $\mathcal{C}$ chooses $s \xleftarrow{\$} \mathcal{B}_\mu^{n}$, $e \xleftarrow{\$} \mathcal{B}_\mu^{m}$ and $k_0^* \xleftarrow{\$} \mathcal{K}$. The challenger chooses a bit $b^* \in \{0, 1\}$ at random. At last, it returns $(k_{b^*}^*, (c_0^*, c_1^*, c_2^*))$ to the adversary $\mathcal{A}$, where

$$\begin{aligned}
c_0^* &:= As + e && \in \{0,1\}^m\\
c_1^* &:= Bs + S_0^{\mathrm{T}}e - E_0^{\mathrm{T}}s + \mathrm{ECC}(s) && \in \{0,1\}^m\\
c_2^* &:= Ds + S_1^{\mathrm{T}}e - E_1^{\mathrm{T}}s + \mathrm{ECC}(s) && \in \{0,1\}^m\\
t^* &:= \mathcal{H}(c_0^*, c_1^*, c_2^*, s) && \in \{0,1\}^{(l+1)}\\
k_1^* &:= Fs + t^* && \in \{0,1\}^{(l+1)}.
\end{aligned}$$

Note that c_1^* in target ciphertext (c_0^*, c_1^*, c_2^*) is equal to $S_0^{\mathrm{T}}c_0^* + \mathrm{ECC}(s)$.

Lemma 10: $|\Pr[R_5] - \Pr[R_4]| \leq \mathrm{negl}(n)$.

Proof: We prove this lemma by using similar proofs from Lemma 6 to Lemma 8. We omit these details.

Game 6: It is the same as Game 5, except that, the challenger $\mathcal{C}$ changes the challenge phase as follows:

Challenge: The challenger $\mathcal{C}$ chooses $v \xleftarrow{\$} U_m$, $s \xleftarrow{\$} \mathcal{B}_\mu^{n}$ and $k_0^* \xleftarrow{\$} \mathcal{K}$, and randomly samples a bit $b^* \in \{0, 1\}$. Finally, it returns $(k_{b^*}^*, (c_0^*, c_1^*, c_2^*))$ to the adversary $\mathcal{A}$, where

$$\begin{aligned}
c_0^* &:= v && \in \{0,1\}^m\\
c_1^* &:= S_0^{\mathrm{T}}v + \mathrm{ECC}(s) && \in \{0,1\}^m\\
c_2^* &:= S_1^{\mathrm{T}}v + \mathrm{ECC}(s) && \in \{0,1\}^m\\
t^* &:= \mathcal{H}(c_0^*, c_1^*, c_2^*, s) && \in \{0,1\}^{(l+1)}\\
k_1^* &:= Fs + t^* && \in \{0,1\}^{(l+1)}.
\end{aligned}$$

Lemma 11: Under the decisional (n, μ)-VLPN assumption, we have that $|\Pr[R_6] - \Pr[R_5]| \leq \mathrm{negl}(n)$.

Proof: It is different that the challenger $\mathcal{C}$ replaces $c^* = v$ in Game 6 with $c^* = As + e$ in Game 5, where $v \xleftarrow{\$} U_m$. Under the *decisional* (n, μ)-VLPN assumption, the challenge ciphertext (c_0^*, c_1^*, c_2^*) in Game 6 is computationally indistinguishable from the challenge ciphertext (c_0^*, c_1^*, c_2^*) in Game 5. So, we have that $|\Pr[R_6] - \Pr[R_5]| \leq \mathrm{negl}(n)$.

Game 7: It is the same as Game 6, except that, the challenger $\mathcal{C}$ changes the challenge phase as follows:

Challenge: The challenger $\mathcal{C}$ chooses $v \xleftarrow{\$} U_m$, $w \xleftarrow{\$} U_m$, $s \xleftarrow{\$} \mathcal{B}_\mu^n$, $w' \xleftarrow{\$} U_m$ and $k_0^* \xleftarrow{\$} \mathcal{K}$, and randomly samples a bit $b^* \in \{0, 1\}$. Finally, it returns $(k_{b^*}^*, (c_0^*, c_1^*, c_2^*))$ to the adversary $\mathcal{A}$, where

$$\begin{aligned} c_0^* &:= v && \in \{0,1\}^m \\ c_1^* &:= w + \mathrm{ECC}(s) && \in \{0,1\}^m \\ c_2^* &:= w' + \mathrm{ECC}(s) && \in \{0,1\}^m \\ t^* &:= \mathcal{H}(c_0^*, c_1^*, c_2^*, s) && \in \{0,1\}^{(l+1)} \\ k_1^* &:= Fs + t^* && \in \{0,1\}^{(l+1)}. \end{aligned}$$

Lemma 12: $|\Pr[R_7] - \Pr[R_6]| \leq \mathrm{negl}(n)$.

Proof: It is different that the challenger $\mathcal{C}$ replaces $c_1^* = w + \mathrm{ECC}(s)$ and $c_2^* = w' + \mathrm{ECC}(s)$ in Game 7 with $c_1^* = S_0^{\mathrm{T}} v + \mathrm{ECC}(s)$ and $c_2^* = S_1^{\mathrm{T}} v + \mathrm{ECC}(s)$ in Game 6, where $v \xleftarrow{\$} U_m$, $w \xleftarrow{\$} U_m$ and $w' \xleftarrow{\$} U_m$. Under the *decisional* (n, μ)-KLPN problem, the challenge ciphertext (c_0^*, c_1^*, c_2^*) in Game 7 is computationally indistinguishable from the challenge ciphertext (c_0^*, c_1^*, c_2^*) in Game 6. So, we have that $|\Pr[R_7] - \Pr[R_6]| \leq \mathrm{negl}(n)$.

Game 8: It is the same as Game 7, except that, the challenger $\mathcal{C}$ changes the challenge phase as follows:

Challenge: The challenger $\mathcal{C}$ chooses $v \xleftarrow{\$} U_m$, $w \xleftarrow{\$} U_m$, $s \xleftarrow{\$} \mathcal{B}_\mu^n$, $w' \xleftarrow{\$} U_m$, $z \xleftarrow{\$} U_{(l+1)}$ and $k_0^* \xleftarrow{\$} \mathcal{K}$, and randomly samples a bit $b^* \in \{0, 1\}$. At last, it returns $(k_{b^*}^*, (c_0^*, c_1^*, c_2^*))$ to the adversary $\mathcal{A}$, where

$$\begin{aligned} c_0^* &:= v && \in \{0,1\}^m \\ c_1^* &:= w + \mathrm{ECC}(s) && \in \{0,1\}^m \\ c_2^* &:= w' + \mathrm{ECC}(s) && \in \{0,1\}^m \\ t^* &:= \mathcal{H}(c_0^*, c_1^*, c_2^*, s) && \in \{0,1\}^{(l+1)} \\ k_1^* &:= z + t^* && \in \{0,1\}^{(l+1)}. \end{aligned}$$

Lemma 13: $|\Pr[R_8] - \Pr[R_7]| \leq \mathrm{negl}(n)$.

Proof: It is different that the challenger $\mathcal{C}$ replaces $k_1^* = z + t^*$ in Game 8 with $k_1^* = Fs + t^*$ in Game 7, where $z \xleftarrow{\$} U_{(l+1)}$. By using Lemma 1, the challenge ciphertext (c_0^*, c_1^*, c_2^*) in Game 8 is computationally indistinguishable from the challenge ciphertext (c_0^*, c_1^*, c_2^*) in Game 7. So, we have that $|\Pr[R_8] - \Pr[R_7]| \leq \mathrm{negl}(n)$.

Lemma 14: $|\Pr[R_8] - \frac{1}{2}|| = 0$.

Proof: Note that the challenge ciphertext (c_0^*, c_1^*, c_2^*) and k_1^* in Game 8 perfectly hide the information of s. The key k_1^* is perfectly indistinguishable from k_0^* in view of the adversary $\mathcal{A}$.

In all, we prove that our $\mathcal{KEM}$ is IND-CCA secure by using Lemma 6 - Lemma 14. In other words, we have that $Adv_{KEM,\mathcal{A}}^{ind-cca}(1^k) = |\Pr[R_0] - \frac{1}{2}|| \leq \mathrm{negl}(n)$.

TABLE 2. Assumption of schemes.

Scheme	Assumption
CRYSTALS-Kyber	R-LWE
Classic McEliece	LWE
SABER	Mod-LWR
NTRU-HRSS-KEM	R-lattice
FrodoKEM [18]	R-LWE
NewHope [35]	R-LWE
MP [23]	LWE
PW [29]	DDH or LWE
DMN [10]	VLPN
KMP [32]	KLPN
YZ [13]	LPN
Ours	VLPN

V. COMPARISONS

In this section, we provide some comparisons with several schemes in two tables. In Table 2, the schemes CRYSTALS-Kyber, Classic McEliece, SABER, NTRU-HRSS-KEM, FrodoKEM [18], MP [23] and NewHope [35] are based on lattices, while the schemes DMN [10], KMP [32] and YZ [13] are based on LPNs. The underlying assumption of PW [29] is DDH or LWE. In addition, as KEM can be simply regarded as the PKE scheme encrypting a random, we implement our IND-CCA secure KEM scheme from the VLPN, and compare several CCA secure PKE schemes (i.e., DMN [10], KMP [32] and YZ [13]) in Table 3.

We provide specific parameters for our $\mathcal{KEM}$ based on the VLPN assumption. We assume that the *decisional* (n, μ)-VLPN problem can be solved in time about $2^{4\mu n}$ and our $\mathcal{KEM}$ (from matrix version VLPN) has security $(2n + m + 1) \cdot 2^{4\mu n + 1}$. Let $n = 11449$, $c = \frac{1}{16}$, $\mu = 0.00234$, our $\mathcal{KEM}$ achieves about 128-bit security. We show the sizes of the public key, secret key and ciphertext with several IND-CCA secure PKE schemes and our $\mathcal{KEM}$ in Table 3. Our $\mathcal{KEM}$ is more efficient than DMN [10], KMP [32] and YZ [13] in terms of key sizes and ciphertext overhead on 128-bit security. Compared with CRYSTALS-Kyber, NTRU-HRSS-KEM and FrodoKEM [18], the sizes of key and ciphertext of our scheme have no advantage. However, these lattice-based schemes involve operations over GF(q). In contrast, our scheme only involves operations over GF(2), which has an obvious advantage when implemented using hardware.

TABLE 3. Specific parameters on 128-bit security.

Scheme	|Public key|	|Secret key|	|Ciphertext|
DMN [10]	7.27GB	7.24GB	7.03KB
KMP [32]	80.89MB	46.23MB	6.80KB
YZ [13]	70.95MB	70.65MB	86.50KB
Ours	50.78MB	62.50MB	4.54KB

VI. CONCLUSION AND FUTURE WORK

In this work, by using double-trapdoor technique and a target collision resistant hash function, we directly construct the first IND-CCA secure KEM scheme from the low-noise LPN. For the sizes of public key, secret key and ciphertext, our scheme is more efficient than DMN [10], KMP [32] and YZ [13]. Compared with CRYSTALS-Kyber, NTRU-HRSS-KEM and FrodoKEM [18], the sizes of key and ciphertext of our scheme (based on low-noise LPN) have no advantage. It is future work to improve the efficiency of the scheme (based on Ring-LPN).

REFERENCES

[1] A. Blum, M. Furst, M. Kearns, and R. J. Lipton, ''Cryptographic primitives based on hard learning problems,'' in *Advances in Cryptology—CRYPTO*. Berlin, Germany: Springer, 1994, pp. 278–291.

[2] J. Katz and J. S. Shin, ''Parallel and concurrent security of the hb and hb + protocols,'' in *Advances in Cryptology—EUROCRYPT*. Berlin, Germany: Springer, 2006, pp. 73–87.

[3] B. Applebaum, Y. Ishai, and E. Kushilevitz, ''Cryptography with constant input locality,'' *J. Cryptol.*, vol. 22, no. 4, pp. 429–469, Oct. 2009, doi: 10.1007/s00145-009-9039-0.

[4] J. Stern, ''A method for finding codewords of small weight,'' in *Coding Theory and Applications*. Berlin, Germany: Springer, 1989, pp. 106–113.

[5] G. Valiant, ''Finding correlations in subquadratic time, with applications to learning parities and juntas,'' in *Proc. IEEE 53rd Annu. Symp. Found. Comput. Sci.*, Oct. 2012, pp. 11–20.

[6] A. Blum, A. Kalai, and H. Wasserman, ''Noise-tolerant learning, the parity problem, and the statistical query model,'' *J. ACM*, vol. 50, no. 4, pp. 506–519, Jul. 2003, doi: 10.1145/792538.792543.

[7] A. Jain, S. Krenn, K. Pietrzak, and A. Tentes, ''Commitments and efficient zero-knowledge proofs from learning parity with noise,'' in *Advances in Cryptology—ASIACRYPT*. Berlin, Germany: Springer, 2012, pp. 663–680.

[8] A. Juels and S. A. Weis, ''Authenticating pervasive devices with human protocols,'' in *Proc. Annu. Int. Cryptol. Conf.* in Lecture Notes in Computer Science, vol. 3621, Santa Barbara, CA, USA: Springer, Aug. 2005, pp. 293–308, doi: 10.1007/11535218_18.

[9] E. Berlekamp, R. McEliece, and H. van Tilborg, ''On the inherent intractability of certain coding problems (Corresp.),'' *IEEE Trans. Inf. Theory*, vol. 24, no. 3, pp. 384–386, May 1978.

[10] N. Döttling, J. Müller-Quade, and A. C. A. Nascimento, ''IND-CCA secure cryptography based on a variant of the LPN problem,'' in *Advances in Cryptology—ASIACRYPT*. Berlin, Germany: Springer, 2012, pp. 485–503.

[11] H. Cheng, X. Li, H. Qian, and D. Yan, ''Simpler CCA secure PKE from LPN problem without double-trapdoor,'' in *nformation and Communications Security*. Cham, Switzerland: Springer, 2018, pp. 756–766.

[12] E. Levieil and P. Fouque, ''An improved LPN algorithm,'' in *Proc. Int. Conf. Secur. Cryptogr. Netw.* in Lecture Notes in Computer Science, vol. 4116. Maiori, Italy: Springer, Sep. 2006, pp. 348–359, doi: 10.1007/11832072_24.

[13] Y. Yu and J. Zhang, ''Cryptography with auxiliary input and trapdoor from constant-noise LPN,'' in *Advances in Cryptology—CRYPTO*. Berlin, Germany: Springer, 2016, pp. 214–243.

[14] V. Lyubashevsky, ''The parity problem in the presence of noise, decoding random linear codes, and the subset sum problem,'' in *Proc. Int. Workshop Approximation Algorithms Combinat. Optim.* in Lecture Notes in Computer Science, vol. 3624. Berkeley, CA, USA: Springer, Aug. 2005, pp. 378–389, doi: 10.1007/11538462_32.

[15] A. Becker, A. Joux, A. May, and A. Meurer, ''Decoding random binary linear codes in 2n/20: How 1 + 1 = 0 improves information set decoding,'' in *Advances in Cryptology—EUROCRYPT*. Berlin, Germany: Springer, 2012, pp. 520–536.

[16] D. J. Bernstein, T. Lange, and C. Peters, ''Smaller decoding exponents: Ball-collision decoding,'' in *Proc. Annu. Cryptol. Conf.* in Lecture Notes in Computer Science, vol. 6841. Santa Barbara, CA, USA: Springer, Aug. 2011, pp. 743–760, doi: 10.1007/978-3-642-22792-9_42.

[17] P. Kirchner, ''Improved generalized birthday attack,'' *IACR Cryptol. ePrint Arch.*, vol. 2011, no. 2011/377, p. 377, 2011. [Online]. Available: http://eprint.iacr.org/2011/377

[18] NIST. (2017). *National Institute for Standards and Technology*. [Online]. Available: https://csrc.nist.gov/projects/post-quantum-cryptography/round-1-submis%sions

[19] E. Fujisaki and T. Okamoto, ''Secure integration of asymmetric and symmetric encryption schemes,'' in *Proc. 19th Annu. Int. Cryptol. Conf.* in Lecture Notes in Computer Science, vol. 1666. Santa Barbara, CA, USA: Springer, Aug. 1999, pp. 537–554, doi: 10.1007/3-540-48405-1_34.

[20] D. Hofheinz, K. Hövelmanns, and E. Kiltz, ''A modular analysis of the Fujisaki-Okamoto transformation,'' in *Proc. Theory Cryptogr. Conf.* in Lecture Notes in Computer Science, vol. 10677. Baltimore, MD, USA: Springer, Nov. 2017, pp. 341–371, doi: 10.1007/978-3-319-70500-2_12.

[21] E. E. Targhi and D. Unruh, ''Post-quantum security of the Fujisaki-Okamoto and OAEP transforms,'' in *Proc. Theory Cryptogr. Conf.* in Lecture Notes in Computer Science, vol. 9986. Beijing, China: Springer, Oct. 2016, pp. 192–216, doi: 10.1007/978-3-662-53644-5_8.

[22] H. Cheng, X. Li, H. Qian, and D. Yan, ''CCA secure multi-recipient KEM from LPN,'' in *Proc. Int. Conf. Inf. Commun. Secur.* in Lecture Notes in Computer Science, vol. 11149. Lille, France: Springer, Oct. 2018, pp. 513–529, doi: 10.1007/978-3-030-01950-1_30.

[23] D. Micciancio and C. Peikert, ''Trapdoors for lattices: Simpler, tighter, faster, smaller,'' in *Proc. 31st Annu. Int. Conf. Theory Appl. Cryptograph. Techn.* in Lecture Notes in Computer Science, vol. 7237, D. Pointcheval and T. Johansson, Eds. Cambridge, U.K.: Springer, Aug. 2012, pp. 700–718, doi: 10.1007/978-3-642-29011-4_41.

[24] D. Boneh, R. Canetti, S. Halevi, and J. Katz, ''Chosen-Ciphertext security from identity-based encryption,'' *SIAM J. Comput.*, vol. 36, no. 5, pp. 1301–1328, Jan. 2007, doi: 10.1137/S009753970544713X.

[25] R. Canetti, S. Halevi, and J. Katz, ''Chosen-ciphertext security from identity-based encryption,'' in *Proc. Int. Conf. Theory Appl. Cryptograph. Techn.* Berlin, Germany: Springer, 2004, pp. 207–222.

[26] H. Jiang, Z. Zhang, L. Chen, H. Wang, and Z. Ma, ''IND-CCA-secure key encapsulation mechanism in the quantum random oracle model, revisited,'' in *Proc. 38th Annu. Int. Cryptol. Conf.* in Lecture Notes in Computer Science, vol. 10993. Santa Barbara, CA, USA: Springer, Aug. 2018, pp. 96–125, doi: 10.1007/978-3-319-96878-0_4.

[27] S. H. Park, S. Kim, D. H. Lee, and J. H. Park, ''Improved ring LWR-based key encapsulation mechanism using cyclotomic trinomials,'' *IEEE Access*, vol. 8, pp. 112585–112597, 2020, doi: 10.1109/ACCESS.2020.3002223.

[28] D. Dolev, C. Dwork, and M. Naor, ''Nonmalleable cryptography,'' *SIAM J. Comput.*, vol. 30, no. 2, pp. 391–437, Jan. 2000, doi: 10.1137/S0097539795291562.

[29] C. Peikert and B. Waters, ''Lossy trapdoor functions and their applications,'' in *Proc. 14th Annu. ACM Symp. Theory Comput. - STOC*, 2008, pp. 187–196, doi: 10.1145/1374376.1374406.

[30] S. Park, ''How practical is public-key encryption based on LPN and ring-LPN,'' Cryptol. ePrint Arch., Tech. Rep. 2012/699, 2012.

[31] E. Kiltz, ''Chosen-ciphertext secure key-encapsulation based on gap hashed diffie-hellman,'' in *Proc. 10th Int. Conf. Pract. Theory Public-Key Cryptogr.* in Lecture Notes in Computer Science, vol. 4450. Beijing, China: Springer, Aug. 2007, pp. 282–297, doi: 10.1007/978-3-540-71677-8_19.

[32] E. Kiltz, D. Masny, and K. Pietrzak, ''Simple chosen-ciphertext security from low-noise LPN,'' in *Public-Key Cryptography—PKC*. Berlin, Germany: Springer, 2014, pp. 1–18.

[33] D. Micciancio and P. Mol, ''Pseudorandom knapsacks and the sample complexity of LWE search-to-decision reductions,'' in *Proc. 31st Annu. Cryptol. Conf.* in Lecture Notes in Computer Science, vol. 6841, P. Rogaway, Ed. Santa Barbara, CA, USA: Springer, Aug. 2011, pp. 465–484, doi: 10.1007/978-3-642-22792-9_26.

[34] G. Zemor, ''On expander codes,'' *IEEE Trans. Inf. Theory*, vol. 47, no. 2, pp. 835–837, Nov. 2001, doi: 10.1109/18.910593.

[35] E. Alkim, L. Ducas, T. Pöppelmann, and P. Schwabe, ''Post-quantum key exchange - A new hope,'' in *Proc. 25th USENIX Secur. Symp., USENIX Secur.*, T. Holz and S. Savage, Eds. Austin, TX, USA: USENIX Association, Aug. 2016, pp. 327–343. [Online]. Available: https://www.usenix.org/conference/usenixsecurity16/technical-sessions/p%resentation/alkim

SHENGFENG XU is currently pursuing the M.S. degree with the Department of Computer Science, East China Normal University, Shanghai, China. His research interests include post-quantum cryptography and learning parity with noise.

XIANGXUE LI received the Ph.D. degree from Shanghai Jiao Tong University, in 2006. He was with the School of Information Security Engineering, Shanghai Jiao Tong University. He is currently a Professor with the School of Software Engineering, East China Normal University. He has authored three books and has authored or coauthored more than 70 articles. His research interests include lightweight protocol design, anonymity, and pseudorandom sequence.