Lattice Codes for Lattice-Based PKE

Shanxiang Lyu, Ling Liu, Junzuo Lai, Cong Ling, Hao Chen

Abstract—The public key encryption (PKE) protocol in lattice-based cryptography (LBC) can be modeled as a noisy point-to-point communication system, where the communication channel is similar to the additive white Gaussian noise (AWGN) channel. To improve the error correction performance, this paper investigates lattice-based PKE from the perspective of lattice codes. We propose an efficient labeling function that converts between binary information bits and lattice codewords. The proposed labeling is feasible for a wide range of lattices, including Construction-A and Construction-D lattices. Based on Barnes-Wall lattices, a few improved parameter sets with either higher security or smaller ciphertext size are proposed for FrodoPKE.

Index Terms—public key encryption (PKE), lattice-based cryptography (LBC), lattice codes.

1 Introduction

The impending realization of scalable quantum computers has posed a great challenge for modern public key cryptosystems. As Shor's quantum algorithm [1] can solve the prime factorization and discrete logarithm problems in polynomial time, conventional public-key cryptosystems based on these problems are no longer secure. Although making a prophesy for when can we build a large quantum computer is hard, we should start preparing the next generation quantum-sate cryptosystem as soon as possible, because historical experiences show that deploying modern public key cryptography infrastructures takes a long time.

Reacting to this urgency, the subject of post-quantum cryptography (PQC) has been systematically developed in the last decade [2], [3]. PQC aims to design secure cryptosystems against quantum attacks, while being compatible to run on a classic computer. From 2016, the National Institute of Standards and Technology (NIST) has initiated a process to solicit, evaluate, and standardize one or more quantum-resistant public-key cryptographic algorithms. The process evolves around public key encryption/key encapsulation mechanism (PKE/KEM) and digital signature proposals.

Lattice-based proposals are conceived as the most promising quantum-safe candidates. NIST's standardization has come to round 3. Among the announced first track candidates [4], three out of four KEM proposals are based on lattices (Kyber, Saber, NTRU), and two out of three signature schemes are also based on lattices. Specifically, lattice-based cryptography (LBC) enjoys the following advantages. First, the LBC implementations offer security proofs based on NP-hard problems with average-case to worst-case hardness. Second, in addition to being quantum-age secure, they are notable for their efficiency, primarily due to their inherent linear algebra based matrix/ vector operations on integers. Finally, LBC constructions offer extended functionality for advanced security services such as identity-based encryption and fully-homomorphic encryption (FHE).

Considering lattice-based PKE/KEM, error correction techniques have been either implicitly or explicitly employed to achieve a small decryption failure rate (DFR). The encryption-decryption process of messages amounts to the transmission of messages through an additive noise channel. Since the decrypted messages cannot be 100% correct, the receiver can correct the errors by using error correction codes. Moreover, since the adversary may extract the secrets by taking advantages of high DFRs [5], [6], the DFR of a PKE/KEM scheme has to be extremely small (e.g., smaller than 2^{-128}). It is therefore promising to improve the error correction mechanism in lattice-based PKE/KEM, with the hope of obtaining better trade-off parameters:

- Security Strength: If the error correction mechanism allows to increase the noise variance while maintaining a small DFR, then the PKE/KEM scheme has a higher security level.
- Communication Bandwidth: If the error correction mechanism allows to reduce the modulo number while maintaining a small DFR, then the size of the ciphertext is reduced.

1.1 Related Works

KEMs can simultaneously output a session key together with a ciphertext that can be used to recover the session key. Two major approaches to design KEMs are PKEs (KEMs without reconciliation, see, e.g. [7]–[10]) and key exchanges (KEMs with reconciliation, see, e.g. [11]–[13]). PKEs enjoy the advantages of achieving IND-CCA security, so we focus on lattice-based PKEs in this work.

Most lattice-based PKEs have implicitly employed an error correction mechanism which is referred to as "modulation" in communication theory. It represents a mapping from a binary string to different positions in a constellation with symbols from $\{0, q\}$. If the noise amplitude is smaller than the error correction radius, then the decryption is correct. Thus a larger q enables higher error correction capability. Specifically, Regev's learning with errors (LWE) based PKE scheme [2] modulates 1-bit information $\{0, 1\}$

S. Lyu, J. Lai and H. Chen are with the College of Cyber Security, Jinan University, Guangzhou 510632, China (Email: lsx07@jnu.edu.cn, laijunzuo@gmail.com, chenhao@fudan.edu.cn), L. Liu is with the College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518060, China (Email: liulingcs@szu.edu.cn). C. Ling is with the department of EEE, Imperial College London, London, SW7 2AZ, United Kingdom (Email: c.ling@imperial.ac.uk).

to $\{0, q/2\}$. Kawachi et.al. [14] extends the PKE scheme to multi-bit modulation.

In recent years, researchers have realized that (digital) error correction codes (ECC) can be concatenated with modulations to obtain better error correction performance. For instance, the LAC [15] PKE employs BCH codes for error correction, which helps to reduce the modulo size q from 12289 to 251. The reason behind the small q is that, although the modulation level has minus error correction capability, the induced ECC helps to achieve a smaller DFR. Other examples can be found in the repetition codes based NewHope-Simple [8], and the Polar codes based NewHope-Simple [16]. The downside of an extra modern ECC is an increased complexity of the program code and a higher sensitivity to side-channel attacks [17] (information is obtained through physical channels such as power measurements, variable execution time of the decoding algorithm, etc).

More importantly, using ECC and modulation in a concatenated manner confines the overall performance of the system, whose deficiencies include less flexible parameter sets, and the independent decoding nature of modulation and ECC. Fortunately, the joint design of ECC and modulation (referred to as coded modulation) has been studied in information theory and wireless communications for a few decades. Ungerboeck's pioneering work [18] in the 1980s showed that coded modulation exhibits significant performance gains. Then, Forney [19] systematically studied the coded modulation from coset codes/lattice codes. In the language of coset codes, Ungerboeck's trellis coded modulation is the combination of trellis codes and modulation, while lattice codes represent the combination of linear codes and modulation [20]. A major breakthrough in information theory is that Erez and Zamir [21] shows high dimensional random lattice codes can achieve the capacity of additive white Gaussian noise (AWGN) channels. Recent years have also witnessed the use of Polar lattices [22] and LDPC lattices [23] in achieving the capacity of AWGN channels.

It is noteworthy that applying lattice codes in LBC is not straightforward, because previous lattice coding literature [20] was considering lattice codes for the physical layer (the transmission power of the codes matters), while the modulo q arithmetic in LBC represents a higher layer. In the past few years, there have been some works that employ lattice codes in PKEs. VanPoppelen designed a Leech lattice based PKE in 2016 [9], while Saliba et.al. [10] design an E_8-lattice-based PKE in 2021. The use of E_8 and Leech parallels the celebrated breakthrough in mathematics in recent years: proving the E_8 and Leech lattices offer the best sphere packing density in dimensions 8 and 24 [24], [25]. To actually deploy lattice codes, one may notice that a labeling technique from the input binary information bits to the set of lattice codewords is needed. For instance, with 2^{32} lattice codewords, defining the labeling through an exhaustive lookup table is too complicated, and the better solution should resort to a linear labeling function. Unfortunately, the labeling technique is missing in the Leech lattice based PKE [9], while the labeling technique for E_8 in [10] is nonlinear.

1.2 Contributions

To fully unleashed the power of lattice codes in LBC, this paper contributes in the following aspects.

- We consider the plain-LWE scheme Frodo [7] and model it as a communication system, over which the communication channel is akin to the AWGN channel. Frodo was selected as an alternate candidate for the NIST PQC standardization Round 3, which may provide longer-term security guarantees since it is less susceptible to algebraic attacks. Unlike the ring-based or module-based schemes such as NewHope-Simple [8] and Kyber [26], conventional ECC cannot be easily applied to Frodo, as the number of symbols that the message bits are mapped to is very small (i.e., 64 symbols versus 256 or 1024 symbols). To fit into the FrodoPKE channel, optimal short lattice codes are needed.
- We present a universal and efficient labeling technique for cubic-shaping based lattice codes. Due to the modulo q arithmetic, lattice codes in LBC have to use hypercube shaping, which means the coarse lattice should be a simple integer lattice $q\mathbb{Z}^n$. Although the number of lattice codewords can be easily identified in hypercube shaping, there seems to no efficient labeling function available in the literature. In this regard, a labeling function is proposed to establish a one-to-one map between the binary information bits and the set of lattice vectors. For a fine lattice with large coding gains, we first rewrite its lattice basis to a rectangular form (the product of a unimodular matrix and a diagonal matrix). Then by further developing a non-uniform labeling function for the rectangular forms, the shaping lattice becomes an integer lattice. The proposed labeling is feasible for a wide range of lattices, such as D_4, E_8, BW_{16}, Λ_{24}, etc. In addition, we present a constant-time fast decoding algorithm for BW_{16}.
- Thanks to Hadamard's inequality, a unified DFR formula over AWGN channels is derived to serve the worst-case DFR analysis. Only the coding gain and the kissing number of lattices are needed in the DFR formula. Previously the DFR in lattice-code based PKE error correction were calculated by means of a computationally case-by-case intensive analysis. Subsequently better parameter sets for the FrodoPKE are provided, with either higher security levels or smaller ciphertext sizes. The recommended BW_{16}-based implementation has the following advantages: it has larger coding gain than E_8, while its dimension is also compatible with the 64-dimension requirement in FrodoPKE.

The rest of this paper is organized as follows. Backgrounds about lattice codes and PKE are reviewed in Section II. The proposed labeling is introduced and analyzed in Section III. Section IV presents a coset-based lattice decoding formulation, with a particular emphasize on BW_{16}. Section V presents the improved parameter sets for FrodoPKE. The last section concludes this paper.

2 Preliminaries

2.1 Lattice Codes and Hypercube Shaping

Definition 1 (Lattices)**.** An n-dimensional lattice Λ is a discrete additive subgroup of $\mathbb{R}^m$, $m \geq n$. Based on n

linearly independent vectors $\mathbf{b}_1, \ldots, \mathbf{b}_n$ in $\mathbb{R}^m$, Λ can be written as

$$\Lambda = \mathcal{L}(\mathbf{B}) = z_1\mathbf{b}_1 + z_2\mathbf{b}_2 + \cdots + z_n\mathbf{b}_n, \tag{1}$$

where $z_1, ..., z_n \in \mathbb{Z}$, and $\mathbf{B} = [\mathbf{b}_1, \ldots, \mathbf{b}_n]$ is referred to as a generator matrix of Λ.

Definition 2 (Closest Vector Problem)**.** Considering a target $\mathbf{t} \in \mathbb{R}^n$ and an n-dimensional lattice Λ, the closest vector problem ask to find the closest point $\mathbf{p} \in \Lambda$ to $\mathbf{t}$.

The nearest neighbor quantizer $Q_\Lambda(\cdot)$ denotes a function that solves CVP, i.e.,

$$Q_\Lambda(\mathbf{x}) = \arg\min_{\mathbf{v}\in\Lambda} \|\mathbf{x} - \mathbf{v}\|. \tag{2}$$

The Voronoi region $\mathcal{V}_\Lambda$ denotes set of query points that are closer to the origin than any other lattice points in Λ, i.e.,

$$\mathcal{V}_\Lambda = \{\mathbf{y} \mid \|\mathbf{y}\|^2 \leq \|\mathbf{y} - \mathbf{w}\|^2, \forall \mathbf{w} \in \Lambda\}.$$

Definition 3 (Modulo lattice)**.** $[\mathbf{x}] \mod \Lambda$ denotes the quantization error of $\mathbf{x}$ with respect to Λ:

$$[\mathbf{x}] \mod \Lambda = \mathbf{x} - Q_\Lambda(\mathbf{x}) \in \mathcal{V}_\Lambda. \tag{3}$$

Definition 4 (Nested lattices)**.** Two lattices Λ_f and Λ_c are nested if $\Lambda_c \subset \Lambda_f$. The denser lattice Λ_f is called the *fine/coding* lattice, and Λ_c is called the *coarse/shaping* lattice.

A typical and efficient method to build a set of points data transmission is through lattice codes. A lattice code $\mathcal{C}$ can be designed by properly chosen coset leaders of Λ_f/Λ_c:

$$\mathcal{C}(\Lambda_f, \Lambda_c) = \Lambda_f \cap \mathcal{V}_{\Lambda_c} = \{[\mathbf{w}] \mod \Lambda_c \mid \mathbf{w} \in \Lambda_f\}. \tag{4}$$

If $\Lambda_c = p\mathbb{Z}^n$, then (4) is called *hypercube shaping*, and the modulo lattice in (4) becomes the modulo of an integer number. The illustration of hypercube shaping in a 2-dimensional space is shown in Fig. 1. The purple points denote Λ_f, and those points enclosed with black circles denote Λ_c. The nesting relation is $\Lambda_c = 7\mathbb{Z}^2 \subset \Lambda_f \subset \mathbb{Z}^2$.

The coding gain of a lattice is defined as

$$\gamma(\Lambda) = \lambda_1(\Lambda)^2/\mathrm{Vol}(\Lambda)^{2/n} \tag{5}$$

where $\lambda_1(\Lambda)$ denotes the length of the shortest non-zero vector in Λ, and $\mathrm{Vol}(\Lambda) = |\mathcal{V}_\Lambda|$ denotes the volume of Λ.

2.2 PKE/KEM in LBC

FrodoKEM [7] is among the second round candidates of NIST standardization. The core of FrodoKEM is a public-key encryption scheme called FrodoPKE, whose IND-CPA security is tightly related to the hardness of a corresponding learning with errors problem. Without using algebraic structures, the security level of FrodoKEM is higher than other PKE/KEM schemes based on ring/module LWE.

A public key encryption scheme PKE is a tuple of algorithms (KeyGen , Enc , Dec) along with a message space $\mathcal{M}$.

In the key generation algorithm, by setting $\mathbf{S}, \mathbf{E} \sim \chi_\sigma^{n'\times\bar{n}}$, with χ_σ being a discrete Gaussian distribution with width σ, and $\mathbf{A}$ admits a uniform distribution in $\mathbb{Z}_q^{n'\times n'}$, it computes

$$\mathbf{B} = \mathbf{AS} + \mathbf{E} \in \mathbb{Z}_q^{n'\times\bar{n}}. \tag{6}$$

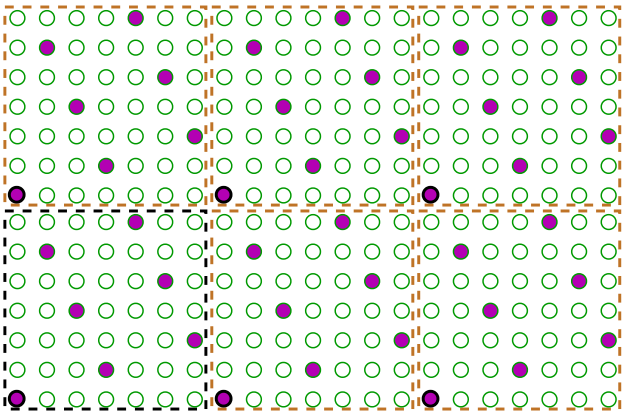

Fig. 1: Demonstration of the lattice code $\mathcal{C}(\Lambda_f, 7\mathbb{Z}^2) = \{(0,0),(1,5),(2,3),(3,1),(4,6),(5,4),(6,2)\}$ built from hypercube shaping (inside the black square).

The public key is $pk = (\mathbf{B}, \mathbf{A})$, and the secret key is $sk = \mathbf{S}$.

In the part of public key encryption, it samples $\mathbf{S}', \mathbf{E}' \sim \chi_\sigma^{\bar{m}\times n'}$, $\mathbf{E}'' \sim \chi_\sigma^{\bar{m}\times\bar{n}}$, and computes

$$\mathbf{C}_1 = \mathbf{S}'\mathbf{A} + \mathbf{E}' \tag{7}$$

$$\mathbf{V} = \mathbf{S}'\mathbf{B} + \mathbf{E}''. \tag{8}$$

To encrypt $\mu \in \mathcal{M} = \{0,1\}^{\bar{m}\bar{n}B}$, the ciphertext is generated by

$$c = (\mathbf{C}_1,\ \mathbf{C}_2 = \mathbf{V} + \mathsf{Frodo.EncodeM}(\mu)) \tag{9}$$

The function Frodo.EncodeM represents a matrix encoding function of bit strings. In an element-wise manner, each B-bit value is transformed into the B most significant bits of the corresponding entry modulo q.

To decrypt, it employs the secret key $\mathbf{S}$ and the ciphertext $\mathbf{C}_1, \mathbf{C}_2$ to compute

$$\hat{\mu} = \mathsf{Frodo.DecodeM}(\mathbf{C}_2 - \mathbf{C}_1\mathbf{S}), \tag{10}$$

where Frodo.DecodeM standards for the demodulation function. At the security level of 145 bits and 210 bits, the recommended parameters are

Frodo640 :

$n' = 640, \bar{n} = 8, \bar{m} = 8, q = 2^{15}, \sigma = 2.75, \mathcal{M} = \{0,1\}^{128}$

Frodo976 :

$n' = 976, \bar{n} = 8, \bar{m} = 8, q = 2^{16}, \sigma = 2.3, \mathcal{M} = \{0,1\}^{192}$.

These levels correspond to the brute-force security of AES-128 and AES-192, respectively. The FrodoPKE algorithm is summarized in Fig. 2.

3 The Proposed Scheme

3.1 The Communication Model

Recall that the decryption algorithm of FrodoPKE computes

$$\begin{aligned}\mathbf{Y} &= \mathbf{C}_2 - \mathbf{C}_1\mathbf{S} \\ &= \mathsf{Frodo.EncodeM}(\mu) + \mathbf{S}'\mathbf{E} + \mathbf{E}'' - \mathbf{E}'\mathbf{S},\end{aligned} \tag{11}$$

Input Parameters: q, n', $\bar{n}$, $\bar{m}$, χ_σ.

Alice		**Bob**
$\mathbf{A} \leftarrow\!\!\$\ \mathbb{Z}_q^{n' \times n'}$		
$\mathbf{S}, \mathbf{E} \leftarrow\!\!\$\ \chi_\sigma^{n' \times \bar{n}}$		$\mathbf{S}', \mathbf{E}' \leftarrow\!\!\$\ \chi_\sigma^{\bar{m} \times n'}$
$\mathbf{B} = \mathbf{AS} + \mathbf{E}$	$\xrightarrow{\mathbf{A}, \mathbf{B}}$	$\mathbf{E}'' \leftarrow\!\!\$\ \chi_\sigma^{\bar{m} \times \bar{n}}$
		$\mathbf{C}_1 = \mathbf{S}'\mathbf{A} + \mathbf{E}'$
		$\mathbf{V} = \mathbf{S}'\mathbf{B} + \mathbf{E}''$
		$\mu \leftarrow\!\!\$\ \{0,1\}^{\bar{m}\bar{n}B}$
$\mathbf{Y} = \mathbf{C}_2 - \mathbf{C}_1\mathbf{S}$	$\xleftarrow{\mathbf{C}_1, \mathbf{C}_2}$	$\mathbf{C}_2 = \mathbf{V} +$
		$\mathsf{Frodo.EncodeM}(\mu)$
$\hat{\mu} = \mathsf{Frodo.DecodeM}(\mathbf{Y})$		

Fig. 2: FrodoPKE

whose addition is over the modulo q domain. From the perspective of communications, this amounts to transmitting the modulated μ through an additive noise channel. Specifically, Eq. (11) can be formulated as

$$\mathbf{y} = \mathbf{x} + \mathbf{n} \mod q, \tag{12}$$

where $\mathbf{x} = \mathsf{EncodeV}(\mu) \in \mathbb{R}^{\bar{m}\bar{n}}$ denotes a general error correction function, and $\mathbf{y}$, $\mathbf{n}$ represent the vector form of $\mathbf{Y}$ and $\mathbf{S}'\mathbf{E} + \mathbf{E}'' - \mathbf{E}'\mathbf{S}$, respectively. Since the element-wise modulo q is equivalent to modulo a lattice $q\mathbb{Z}^{\bar{m}\bar{n}}$, EncodeV can be designed from the perspective of lattice codes. As plotted in Fig. 3, the transmission system model consists of the following steps:

- Bit Mapper and Demapper. The former maps binary information bits to an information vector $\mathbf{z}$ defined over integers. The later performs the inverse operation. These operations are straightforward.
- Lattice Labeling and Delabeling: Given a message index $\mathbf{z}$, lattice labeling finds its corresponding lattice codeword $\mathbf{x} \in \mathcal{C}(\Lambda_f, \Lambda_c = q\mathbb{Z}^{\bar{m}\bar{n}})$. Delabeling denotes the inverse of labeling. Lattice labeling&delabeling will be examined in this section.
- CVP Algorithm: Find the closet lattice vector of $\mathbf{y}$ over Λ_f. The CVP algorithm of the employed fine lattice Λ_f will be designed in Section 4.

3.2 Lattice Labeling and Delabeling

Definition 5 (Rectangular Form)**.** A lattice basis $\mathbf{B}$ is in a rectangular form if

$$\mathbf{B} = \mathbf{U} \cdot \mathrm{diag}(\pi_1, \pi_2, ..., \pi_n), \tag{13}$$

where $\mathbf{U} \in \mathrm{GL}_n(\mathbb{Z})$ is a unimodular matrix, and $\pi_1, \pi_2, ..., \pi_n \in \mathbb{Q}^+$.

For any lattice with a rational basis, it has a rectangular form. Specifically, consider the Smith Normal Form factorization of a lattice basis $\mathbf{B}^* \in \mathbb{Q}^{n \times n}$, then we have

$$\mathbf{B}^* = \mathbf{U} \cdot \mathrm{diag}(\pi_1, \pi_2, ..., \pi_n) \cdot \mathbf{U}', \tag{14}$$

where $\mathbf{U}, \mathbf{U}' \in \mathrm{GL}_n(\mathbb{Z})$. As lattice bases are equivalent up to unimodular transforms, the term $\mathbf{U}'$ can be canceled out, and the rectangular form is derived.

For a lattice that features a rectangular form, we can design an efficient labeling scheme. The idea is that rectangular form pluses non-uniform labeling amounts to hypercube shaping. Specifically, let the fine lattice be

$$\Lambda_f = \mathcal{L}(\mathbf{B}_f) = \mathcal{L}(\mathbf{U} \cdot \mathrm{diag}(\pi_1, \pi_2, ..., \pi_n)). \tag{15}$$

Let $p \in \mathbb{Z}^+$ be a common multiplier of $\pi_1, \pi_2, ..., \pi_n$, and define

$$p_1 = p/\pi_1, p_2 = p/\pi_2, ..., p_n = p/\pi_n. \tag{16}$$

If $\mathbf{B}_c = \mathbf{B}_f \mathrm{diag}(p_1, p_2, ..., p_n)$, we have

$$\begin{aligned} \Lambda_c &= \mathcal{L}(\mathbf{U} \cdot \mathrm{diag}(\pi_1, \pi_2, ..., \pi_n) \cdot \mathrm{diag}(p_1, p_2, ..., p_n)) \\ &= \mathcal{L}(p\mathbf{U}) \\ &= p\mathbb{Z}^n. \end{aligned} \tag{17}$$

The last equality is due to the fact that a unimodular matrix can be regarded as a lattice basis of $\mathbb{Z}^n$. Hence modulo Λ_c becomes equivalent to modulo p. Then we arrive at the following theorem.

Theorem 6 (Labeling Function)**.** *Let the message space be*

$$\mathcal{I} = \{0, 1, ..., p_1 - 1\} \times \cdots \times \{0, 1, ..., p_n - 1\}, \tag{18}$$

and define Λ_f, Λ_c as in (15) and (17). With $\mathbf{z} \in \mathcal{I}$, then the function

$$f(\mathbf{z}) = [\mathbf{B}_f \mathbf{z}] \mod p \tag{19}$$

is bijective.

Proof. It suffices to prove that f is both injective and surjective. "Injective" means no two elements in the domain of the function gets mapped to the same image, i.e., for $\mathbf{z}_1, \mathbf{z}_2 \in \mathcal{I}$,

$$\mathbf{z}_1 \neq \mathbf{z}_2 \rightarrow f(\mathbf{z}_1) \neq f(\mathbf{z}_2). \tag{20}$$

We prove this by using contradiction. If $f(\mathbf{z}_1) = f(\mathbf{z}_2)$, it implies that we can find $\mathbf{z}_1, \mathbf{z}_2 \in \mathcal{I}$, $\mathbf{z}_3 \in \mathbb{Z}^n$ such that

$$\begin{aligned} &\mathbf{B}_f(\mathbf{z}_1 - \mathbf{z}_2) = \mathbf{B}_f \cdot \mathrm{diag}(p_1, p_2, ..., p_n) \cdot \mathbf{z}_3 \\ &\rightarrow \mathbf{z}_1 - \mathbf{z}_2 = \mathrm{diag}(p_1, p_2, ..., p_n) \cdot \mathbf{z}_3. \end{aligned} \tag{21}$$

Then (21) has a solution only when $\mathbf{z}_3 = \mathbf{0}$, which leads to $\mathbf{z}_1 = \mathbf{z}_2$.

"Surjective" means that any element in the range of the function is hit by the function. Recall that the number of coset leaders is

$$|\det(\mathbf{B}_c)| / |\det(\mathbf{B}_f)| = p_1 p_2 \cdots p_n. \tag{22}$$

As $|\mathcal{I}| = p_1 p_2 \cdots p_n$, it follows from the injective property that all the coset leaders have been hit distinctively. So the surjection is proved. □

The inverse of f is given by $\hat{\mathbf{z}} = f^{-1}(\hat{\mathbf{x}})$, where

$$\hat{z}_i = \left(\mathbf{B}_f^{-1} \hat{\mathbf{x}}\right)_i \mod p_i,\ i = 1, ..., n. \tag{23}$$

In the transmission system model of Fig. 3, if the noise term $\mathbf{n}$ satisfies $f^{-1}\left(Q_{\Lambda_f}(\mathbf{n})\right) = \mathbf{0}$, then the estimated information integers $\hat{\mathbf{z}}$ are correct. A sufficient condition is to have $Q_{\Lambda_f}(\mathbf{n}) = \mathbf{0}$.

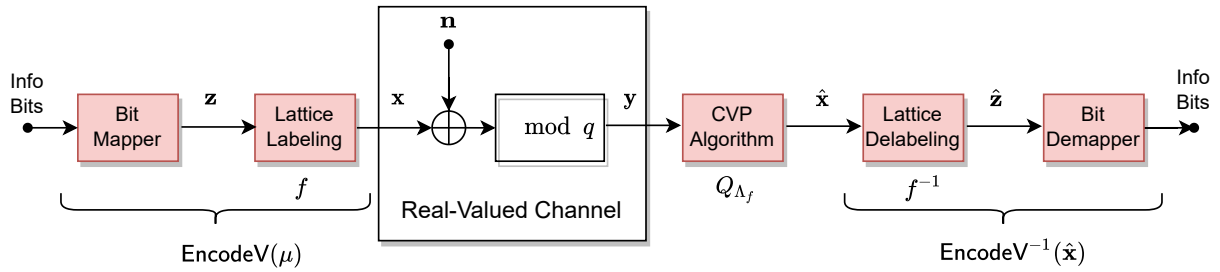

Fig. 3: The transmission system model.

Example: Consider the D_4 lattice, whose lattice basis can be represented as

$$\mathbf{B}_{D_4} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix} \cdot \mathrm{diag}(1,1,1,2). \tag{24}$$

To encode 7 bits over 4 dimensions, let the pair of nested lattices be $(\Lambda_f, \Lambda_c) = (D_4, 4\mathbb{Z}^4)$, and the message space be

$$\mathcal{I} = \{0,1,2,3\}^3 \times \{0,1\}. \tag{25}$$

Concrete steps over Fig. 3 are explained as follows. W.l.o.g, let the input binary string be $\{0,1,1,0,1,1,1\}$. Then the binary-to-integer transform Bin2Int yields

$$\mathbf{z} = [1,2,3,1]^\top.$$

By using lattice labeling in Eq. (19), we have

$$f(\mathbf{z}) = [1,2,3,8]^\top \mod 4 = [1,2,3,0]^\top.$$

If the additive noise is small enough, then at the receiver's side, we may assume that the quantization function Q_{Λ_f} outputs the same $f(\mathbf{z})$. With reference to delabeling in (23), the estimated integers are

$$\begin{aligned} \hat{\mathbf{z}} &= [1 \mod 4,\ 2 \mod 4,\ 3 \mod 4,\ -3 \mod 2]^\top \\ &= [1,2,3,1]^\top. \end{aligned}$$

Finally, the integer-to-binary transform Int2Bin yields the same binary string of the input.

3.3 Rectangular Forms of Code-Based Lattices

The proposed labeling is feasible for a wide range of lattices, such as low-dimensional optimal lattices D_2, D_4, E_8, Λ_{24}, and the general Construction-A and Construction-D lattices. Construction A and Construction D are popular techniques of rising linear codes to lattices, based on wich many remarkable lattices with large coding gains have been constructed(e.g., Barnes-Wall lattices and polar lattices).

Definition 7 (Construction A)**.** Let C be a (n,k,d) linear binary code. A vector $\mathbf{y}$ is a lattice vector of the Construction-A lattice over C if and only if $\mathbf{y}$ is congruent (modulo 2) to a codeword of C.

Let $\phi(\cdot)$ be a natural mapping function from $\mathbb{F}_2$ to $\{0,1\}$, and $\mathbf{G}$ be the generator matrix of a linear code C. By reformulating $\mathbf{G}$ as a Hermite normal form $\{\mathbf{I}, \mathbf{A}\}$, the Construction-A lattice of C can be written as

$$\Lambda_A = \mathcal{L}\left(\begin{bmatrix} \phi(\mathbf{I}) & \mathbf{0} \\ \phi(\mathbf{A}) & 2\mathbf{I} \end{bmatrix}\right). \tag{26}$$

The lattice basis of Λ_A is therefore a rectangular form. The volume of Λ_A is

$$V(\Lambda_A) = 2^{n-k}. \tag{27}$$

Definition 8 (Construction D)**.** Let $C_0 \subset C_1 \subset \cdots \subset C_a = \mathbb{F}_2^n$ be a family of nested binary linear codes, where C_i has parameters (n, k_i, d_i) and C_a is the trivial $(n,n,1)$ code. A vector $\mathbf{y}$ is a lattice vector of the Construction-D lattice over $C_0, ..., C_a$ if and only if $\mathbf{y}$ is congruent (modulo 2^a) to a vector in $C_0 + 2C_1 + \cdots + 2^{a-1}C_{a-1}$.

Denote the generator matrices of C_0, C_i, and C_a as

$$\mathbf{G}_0 = \begin{bmatrix} | & | & & | \\ \mathbf{g}_1 & \mathbf{g}_2 & \cdots & \mathbf{g}_{k_0} \\ | & | & & | \end{bmatrix} \tag{28}$$

$$\mathbf{G}_i = \begin{bmatrix} | & | & & | & & | \\ \mathbf{g}_1 & \mathbf{g}_2 & \cdots & \mathbf{g}_{k_0} & \cdots & \mathbf{g}_{k_i} \\ | & | & & | & & | \end{bmatrix} \tag{29}$$

$$\mathbf{G}_a = \begin{bmatrix} | & | & & | & & | & & | \\ \mathbf{g}_1 & \mathbf{g}_2 & \cdots & \mathbf{g}_{k_0} & \cdots & \mathbf{g}_{k_i} & \cdots & \mathbf{g}_{k_a} \\ | & | & & | & & | & & | \end{bmatrix}, \tag{30}$$

where $1 \le k_0 \le k_1 \le \cdots \le k_a = n$. Then the code formula of a Construction-D lattice is

$$\Lambda_D = \bigcup_{\mathbf{u}_i \in \{0,1\}^{k_i}} \left(\sum_{i=0}^{a-1} 2^i \phi(\mathbf{G}_i)\mathbf{u}_i\right) + 2^a \phi(\mathbf{G}_a)\mathbb{Z}^n \tag{31}$$

$$= \mathcal{L}(\phi(\mathbf{G}_a) \cdot \mathrm{diag}(2^0\mathbf{1}_{k_0}, ..., 2^a\mathbf{1}_{k_a - k_{a-1}})), \tag{32}$$

where $\mathbf{1}_{k_i}$ denotes an all-one vector of dimension k_i. Since $\mathbf{G}_a$ spans $\mathbb{F}_2^n$, $\mathbf{G}_a$ can be chosen as the column-wise permutation of a Hermite normal form to make $\phi(\mathbf{G}_a)$ a unimodular matrix. Thus Construction-D lattices have rectangular forms. Moreover, the volume of a Construction-D lattice is simply

$$V(\Lambda_D) = 2^{an - \sum_{i=0}^{a-1} k_i}. \tag{33}$$

Reed-Muller codes are a class of linear block codes over $\mathbb{F}_2^n$. With Construction D over Reed-Muller codes, the Barnes-Wall lattices can be obtained [19]. The rectangular

forms of Barnes-Wall lattices can be found by using Plotkin constructions:

$$\mathbf{G}_a = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}^{\otimes r} \cdot \mathbf{P}, \tag{34}$$

$$\phi(\mathbf{G}_a) = \phi\left(\begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}^{\otimes r} \right) \cdot \mathbf{P} \in \mathrm{GL}_n(\mathbb{Z}), \tag{35}$$

where $\mathbf{P}$ is a column permutation matrix to meet the nested relation of $\mathbf{G}_0$, ..., $\mathbf{G}_a$.

Algorithm 1: The closest vector algorithm $Q_{BW_{16}}$.

Input: A query vector $\mathbf{y}$.
Output: The closest vector $\hat{\mathbf{v}}$ of $\mathbf{y}$ in BW_{16}.
Define the codewords of $(16, 5, 8)$ as $\mathbf{d}_1, \ldots, \mathbf{d}_{32}$;
for $t = 1, \ldots 32$ **do**
  $\mathbf{y}_t = (\mathbf{y} - \mathbf{d}_t)/2$;
  $\hat{\mathbf{v}}_t = 2Q_{D_n}(\mathbf{y}_t) + \mathbf{d}_t$ ▷ Employ the CVP sub-routine of D_n;
  $\mathrm{Dist}_t = ||\mathbf{y} - \bar{\mathbf{v}}_t||$;
$t^* = \min_t \mathrm{Dist}_t$;
$\hat{\mathbf{v}} = \hat{\mathbf{v}}_{t^*}$.

4 CVP Decoding using unions of cosets

To resist timing attacks against lattice-based PKE/KEM, the decoding algorithms of lattice codes should not only feature low computational complexity, but a constant-time property as well, namely the implementation time is independent of the positions of the query vector in CVP. A natural and efficient way to design such decoding is to partition the lattices as unions of cosets.

4.1 Lattice Partition as Cosets

A closest lattice vector algorithm for Λ' can easily be applied to a coset $\mathbf{g} + \Lambda'$. If $Q_{\Lambda'}(\mathbf{t})$ is the closest point of Λ' to $\mathbf{t}$, then

$$Q_{\Lambda'+\mathbf{g}}(\mathbf{t}) = \mathbf{g} + Q_{\Lambda'}(\mathbf{t} - \mathbf{g}). \tag{36}$$

If $\Lambda' \subset \Lambda$, decoding $\mathbf{t}$ over Λ amounts to computing the closest vector in each coset using a decoding for Λ', and applying the union identity, i.e.,

$$\begin{aligned} Q_{\Lambda}(\mathbf{t}) &= Q_{\Lambda'+\mathbf{g}'}(\mathbf{t}), \\ \mathbf{g}' &= \mathrm{argmin}_{\mathbf{g}\in\Lambda/\Lambda'} \left\|\mathbf{t} - Q_{\Lambda'+\mathbf{g}}(\mathbf{t})\right\|^2. \end{aligned} \tag{37}$$

Thus for a lattice Λ with the sublattice Λ' of index $|G|$, the computational complexity $\mathrm{Comp}(\Lambda)$ of calculating $Q_{\mathbf{t}}(\Lambda)$ is bounded by $|G|(1 + \mathrm{Comp}(\Lambda'))$.

This strategy is frequently combined with the method of decoding direct-sum lattices, in which the sublattice in question is a direct sum of component lattices.

4.2 Decoding BW_{16}

Some low-dimensional Barnes-Wall lattices are

$$BW_8 = (8, 4, 4) + 2\mathbb{Z}^8 \approxeq E_8 \tag{38}$$

$$BW_{16} = (16, 5, 8) + 2(16, 15, 2) + 4\mathbb{Z}^{16} \approxeq \Lambda_{16} \tag{39}$$

$$BW_{32} = (32, 6, 16) + 2(36, 26, 4) + 4\mathbb{Z}^{32} \tag{40}$$

$$BW_{64} = (64, 7, 32) + 2(64, 42, 8) + 4(64, 63, 2) + 8\mathbb{Z}^{64}. \tag{41}$$

All these lattices admit a $\mathbb{Z}^n$ based coset partition, but such partition has a huge number of cosets in general. Whenever possible, partitioning the lattice as D_n based cosets helps to decode faster. We are particularly interested in the decoding of BW_{16}, whose lattice basis is

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 2 & 4 \\ 1 & 1 & 1 & 1 & 0 & 2 & 2 & 0 & 2 & 0 & 0 & 2 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 1 & 2 & 0 & 2 & 0 & 2 & 0 & 0 & 2 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 0 & 2 & 2 & 0 & 0 & 2 & 0 & 0 & 2 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 2 & 2 & 2 & 0 & 0 & 0 & 2 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

BW_{16} has a relatively large coding gain, while the dimension 16 is compatible to most PKE protocols. Moreover, a fast-constant time closest vector algorithm can be designed. To see this, we formulate BW_{16} as

$$BW_{16} = (16, 5, 8) + 2D_{16}, \tag{42}$$

It becomes evident that BW_{16} has a sublattice $2D_{16}$ of index 32, with coset representatives being the codewords of the $(16, 5, 8)$ first-order Reed-Muller code. With reference to Eq. (37), we have

$$\begin{aligned} Q_{BW_{16}}(\mathbf{t}) &= Q_{2D_{16}+\mathbf{g}'}(\mathbf{t}), \\ \mathbf{g}' &= \mathrm{argmin}_{\mathbf{g}\in(16,5,8)} \left\|\mathbf{t} - Q_{2D_{16}+\mathbf{g}}(\mathbf{t})\right\|^2. \end{aligned} \tag{43}$$

Thus by running the D_n lattice decoding algorithm for 32 times, the closest lattice vector of BW_{16} to a query vector can be found. The algorithms $Q_{BW_{16}}$ and Q_{D_n} are listed as Algorithm 1 and Algorithm 2, respectively.

5 Improving FrodoPKE with Barnes-Wall Lattices

5.1 DFR Analysis in the Worst Case

In FrodoPKE and most lattice-based PKE schemes, the discrete Gaussian distribution χ_σ is chosen to closely approximates the continuous Gaussian distribution. The Rényi divergence between this distribution and the continuous Gaussian is used in the security reduction. For simplicity,

Algorithm 2: The closest vector algorithm Q_{D_n}.

Input: A query vector $\mathbf{y}$.
Output: The closest vector $\hat{\mathbf{v}}$ of $\mathbf{y}$ in D_n.

$\mathbf{u} = \lfloor \mathbf{y} \rceil$;
$\delta = |\mathbf{y} - \mathbf{u}|$;
$t^* = \max_t |y_t - u_t|$;
$\mathbf{v} = \mathbf{u}$;
if $y_{t^*} - u_{t^*} > 0$ **then**
  $v_{t^*} \leftarrow v_{t^*} + 1$
else
  $v_{t^*} \leftarrow v_{t^*} - 1$
if $u_1 + \cdots + u_n \mod 2 = 0$ **then**
  $\hat{\mathbf{v}} = \mathbf{u}$
else
  $\hat{\mathbf{v}} = \mathbf{v}$

we will calculate the DFR ($\Pr(\hat{\mu} \neq \mu)$) by modeling χ_σ as a continuous Gaussian distribution $\mathcal{N}(0, \sigma^2)$.

Recall that the error term $\mathbf{n}$ has $\bar{m} \times \bar{n}$ entries, each entry has the form of $\mathbf{se} + \mathbf{e}'' - \mathbf{e}'\mathbf{s}$. Thus we have

$$\mathbb{E}(\mathbf{se} + \mathbf{e}'' - \mathbf{e}'\mathbf{s}) = 0 \tag{44}$$

$$\mathbb{E}\left(\|\mathbf{se} + \mathbf{e}'' - \mathbf{e}'\mathbf{s}\|^2\right) = 2n'\sigma^4 + \sigma^2. \tag{45}$$

Although the entries of $\mathbf{n}$ are not independent, we can use information theory to give a worst case analysis. The information entropy of $\mathbf{n}$ is no larger than that of the joint distribution of $\bar{m} \times \bar{n}$ i.i.d. $\mathcal{N}(0, 2n'\sigma^4 + \sigma^2)$ (also known as Hadamard's Inequality [27]). We adopt this "largest entropy" setting to analyze the DFR, which denotes an AWGN channel.

For a transmitted lattice vector $\mathbf{x}$ that has the same dimension as the channel, the DFR of the PKE protocol can be estimated by using the decoding error probability P_e of a lattice codeword. Let λ_1 and τ be the minimum Euclidean distance and the kissing number of Λ_f, respectively. With reference to [28], we have

$$P_e \triangleq \Pr\left(Q_{\Lambda_f}(\mathbf{n}) \neq \mathbf{0}\right) = \sum_{\mathbf{u} \neq \mathbf{x}, \mathbf{u} \in \Lambda_f} P(\mathbf{x} \to \mathbf{u}), \tag{46}$$

where $P(\mathbf{x} \to \mathbf{u})$ is the pairwise error probability. By considering i.i.d. Gaussian noise $\mathcal{N}(0, \bar{\sigma}^2)$ with $\bar{\sigma} = \sigma\sqrt{2n'\sigma^2 + 1}$, Eq. (46) becomes

$$P_e \leq \frac{\tau}{2}\mathrm{erfc}\left(\frac{\lambda_1}{2\sqrt{2}\bar{\sigma}}\right) \tag{47}$$

$$= \frac{\tau}{2}\mathrm{erfc}\left(\frac{\sqrt{\gamma}q}{2^{B+3/2}\bar{\sigma}}\right), \tag{48}$$

where the second equality is obtained by substituting $\lambda_1 = \sqrt{\gamma}\left(q^n/2^{Bn}\right)^{1/n}$, and B denotes the averaged number of bits encoded in each matrix entry. According to (48), the DFR is determined by a few factors: (i) The coding gain γ, which describes the density of lattice points packed in a unit volume for a given minimum Euclidean distance. (ii) The kissing number τ that measure the number of facets in the Voronoi region of a lattice. (iii) The modulo number q in LBC. (iv) The averaged number of encoded bits B. (v) The standard variance $\bar{\sigma}$ of the effective noise.

Finding the densest lattice structure is a well-studied topic, and the coding gain γ and kissing number τ of some low-dimensional optimal lattices can be found in [28]. Therefore, the key challenge is to appropriately design B, q, $\bar{\sigma}$ based on chosen γ and τ.

5.2 Lattice Parameter Settings

To define an infinite sequence of sphere packings in dimensions 2^r, $r = 1, 2, 3, ...$, which include the densest packings known in dimensions 2, 4, 8 and 16 [28], we adopt the Barnes-Wall lattices. Though being less dense than other known packings in dimensions 32 and higher, they enjoy the merits of doing explicit calculations. Notably, the kissing number [28][P. 151] is

$$\tau = (2+2)(2+2^2)\cdots(2+2^r), \tag{49}$$

where $r = \log_2(n)$. And the coding gain is

$$\gamma_r = 2^{(r-1)/2}, \tag{50}$$

which increases without limit.

Regarding the number of encoded bits B per dimension, by setting $(\Lambda_f, \Lambda_c) = (\Delta\Lambda, p\Delta\mathbb{Z}^n)$, where Δ denotes a scaling factor to enlarge the fine lattice, we have

$$B = 1/n \log_2 |\Delta\Lambda / p\Delta\mathbb{Z}^n| = 1/n \log_2 \frac{p^n}{V(\Lambda)}. \tag{51}$$

In addition, the fine lattice can be rotated to support more flexible rate control. Define the n-dimensional rotation matrix as

$$\mathbf{R}_n = \mathbf{I}_{n/2} \otimes \mathbf{R}_2, \tag{52}$$

where

$$\mathbf{R}_2 = \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}. \tag{53}$$

Then the following lattice partition chain can be obtained:

$$\Lambda / \mathbf{R}_n\Lambda / 2\Lambda / 2\mathbf{R}_n\Lambda / 4\Lambda / \cdots / p\mathbb{Z}^n. \tag{54}$$

By setting the pair of nested lattices as $(\Lambda_f, \Lambda_c) = (\Delta\mathbf{R}_n\Lambda, p\Delta\mathbb{Z}^n)$, then

$$B = 1/n \log_2 |\Delta\mathbf{R}_n\Lambda / p\Delta\mathbb{Z}^n| = 1/n \log_2 \frac{p^n}{2^{n/2}V(\Lambda)}. \tag{55}$$

Table 1 summarizes the parameters of some low-dimensional optimal lattices and the Barnes-Wall lattices. To concatenate low dimensional lattices, the Cartesian product is needed.

Definition 9 (Cartesian product)**.** The Cartesian product of two lattices Λ_1 and Λ_2 of dimensions n is an $2n$ dimensional lattice: $\Lambda_1 \times \Lambda_2 = \{(\mathbf{x}, \mathbf{y}) : \mathbf{x} \in \Lambda_1, \mathbf{y} \in \Lambda_2\}$.

The following lemma is immediate.

Lemma 10. *If Λ' is constructed from the k-fold Cartesian product of $\Lambda \subset \mathbb{R}^n$, i.e, $\Lambda' = \Lambda \times \cdots \times \Lambda \subset \mathbb{R}^{kn}$, then we have*

$$\tau(\Lambda') = k\tau(\Lambda) \tag{56}$$

$$\gamma_r(\Lambda') = \gamma_r(\Lambda). \tag{57}$$

TABLE 1: The properties of some popular lattices.

	$\mathbb{Z}$	D_4	E_8	BW_{16}	Λ_{24}	BW_{32}	BW_{64}
Coding gain γ	1	$2^{1/2}$	2	$2^{3/2}$	4	4	$2^{5/2}$
Kissing number τ	2	24	240	4320	196560	146880	9694080
Volume $V(\Lambda)$	1	2	1	$2^{10} \cdot 4$	1	$2^{20} \cdot 4^6$	$2^{35} \cdot 4^{21} \cdot 8$
Constant-time CVP decoding	✓	✓ [29]	✓ [29]	✓ [This work]	✓ [9]	×	×

5.3 Improved Frodo Parameters

Frodo-640 and Frodo-976 satisfy Categories 1 and 3 for security levels in the NIST PQC Standardization, respectively. To satisfy Categories 1 and 3 for security levels, it is recommended to set the security level higher than 128 and 192 bits, respectively. In addition, because there is an attack method by using decryption failure [6], the DFR should be low. Therefore, it is desirable that the DFRs be less than 2^{-128} and 2^{-192} for Categories 1 or 3 for security levels, respectively.

Compared to the standard Frodo protocol, our scheme only modifies the labeling function, the corresponding CVP algorithm, and the choice of parameters σ, B, q. The security levels refer to the primal and dual attack via the FrodoKEM script pqsec.py [30]. The subscripts C, Q and P denote "classical", "quantum" and "paranoid" estimates on the concrete bit-security given by parameters (n', σ, q). We propose two sets of parameters: the first aims at improving the security level and the second at reducing the communication bandwidth.

5.3.1 Parameter set 1-Improved security strength

We increase σ while keeping n, q unchanged in Frodo-640 and Frodo-976. As shown in Table 2, error correction via E_8, BW_{16} and BW_{64} can improve the security level of the original Frodo-640 and Frodo-976 by 6 to 16 bits. Although the BW_{64} based parameter set offers the highest security enhancement, it only serves as the performance upper bound, as no efficient maximum likelihood decoding algorithm is available.

We recommend the BW_{16} based parameter set. It has about 6 to 8 bits of security advantage to the original Frodo. Moreover, Frodo-640-BW_{16} and Frodo-976-BW_{16} can reliably transmit 144 and 208 bits of information, respectively, outperforming the 128 and 192 bits of $\mathbb{Z}$ and E_8 based implementation [10], [13].

5.3.2 Parameter set 2-Reduced size of ciphertext

Recall that the size of ciphertext is $(\bar{m}n' + \bar{m}\bar{n})q/8$ bytes, so we can reduce q to achieve higher bandwidth efficiency. To keep the DFR small, we also reduce σ to various degrees, as long as the security level is no smaller.

As shown in Table 3, by reducing q from 2^{15} to 2^{14}, $|c|$ can be reduced from 9720 bytes to 9072 bytes in Frodo-640, and from 15744 bytes to 14760 bytes in Frodo-976. Again, the BW_{16} based parameter set is recommended.

5.4 IND-CCA Security

The lattice codes based PKE/KEM also features chosen ciphertext secure (IND-CCA) security. Similarly to the argument in [7], the IND-CPA security of FrodoPKE is upper bounded by the advantage of the decision-LWE problem for the same parameters and error distribution. To endow an IND-CPA encryption scheme with IND-CCA security, the post-quantum secure version of the Fujisaki-Okamoto transform [31], [32] can be applied.

6 Conclusions

Lattice codes can be viewed as analog ECCs, using the Euclidean metric rather than the Hamming metric. At a high level, structured codes are used for error correction, while random codes are used for security. For LBC the two tasks are mixed as both are needed. Since LBC and code-based cryptography have been separate, it becomes more tempting to use lattice codes for error correction in LBC, so as to make the scheme fully "lattice-based".

The bridge that connects lattice codes and LBC is the simple modulo q operation, which induces hypercube shaping. By presenting an efficient lattice labeling function, as well as a general formula to estimate the DFR, lattice codes based error correction becomes practical in LBC. By using some low-dimensional optimal lattices, a few improved parameter sets for FrodoPKE have been achieved, with either higher security or smaller ciphertext sizes. We would also like to remind that the lattice coding techniques in this work can also be applied to Ring/Module LWE-based encryption in a straightforward manner.

References

[1] P. W. Shor, "Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer," *SIAM J. Comput.*, vol. 26, no. 5, pp. 1484–1509, 1997.

[2] O. Regev, "On lattices, learning with errors, random linear codes, and cryptography," in *Proceedings of the 37th Annual ACM Symposium on Theory of Computing (STOC), Baltimore, MD, USA*, pp. 84–93, 2005.

[3] C. Peikert, "A decade of lattice cryptography," *Found. Trends Theor. Comput. Sci.*, vol. 10, no. 4, pp. 283–424, 2016.

[4] N. NIST, (2021). "Post-quantum cryptography. round 3 submissions."

[5] T. Fritzmann, T. Pöppelmann, and J. Sepúlveda, "Analysis of error-correcting codes for lattice-based key exchange," in *Selected Areas in Cryptography - SAC 2018 - 25th International Conference, Calgary, AB, Canada*, pp. 369–390, 2018.

[6] J. Anvers, F. Vercauteren, and I. Verbauwhede, "On the impact of decryption failures on the security of LWE/LWR based schemes," *IACR Cryptol. ePrint Arch.*, p. 1089, 2018.

[7] M. Naehrig, E. Alkim, J. Bos, L. Ducas, K. Easterbrook, B. LaMacchia, P. Longa, I. Mironov, V. Nikolaenko, C. Peikert *et al.*, "Frodokem," *Technical report, National Institute of Standards and Technology*, 2017.

[8] E. Alkim, L. Ducas, T. Pöppelmann, and P. Schwabe, "Newhope without reconciliation," *IACR Cryptol. ePrint Arch.*, p. 1157, 2016.

[9] A. van Poppelen, "Cryptographic decoding of the Leech lattice," *IACR Cryptol. ePrint Arch.*, p. 1050, 2016.

[10] C. Saliba, L. Luzzi, and C. Ling, "Error correction for FrodoKEM using the Gosset lattice," *CoRR*, vol. abs/2110.01740, 2021.

TABLE 2: The recommended parameter sets with higher security.

	Structure of lattice code		$n', \bar{n}, \bar{m}$	q	σ	B	DFR	c size (bytes)	Security		
	Λ_f	Λ_c							C	Q	P
Frodo-640	$\Delta \cdot \mathbb{Z}^{64}$	$\Delta \cdot 4\mathbb{Z}^{64}$	$640, 8, 8$	2^{15}	2.75	2	2^{-164}	9720	149	136	109
Frodo-640-E_8	$\Delta \cdot E_8^8$	$\Delta \cdot 4\mathbb{Z}^{64}$	$640, 8, 8$	2^{15}	3.25	2	2^{-164}	9720	156	142	113
Frodo-640-BW_{16}	$\Delta \cdot BW_{16}^4$	$\Delta \cdot 8\mathbb{Z}^{64}$	$640, 8, 8$	2^{15}	3.23	2.25	2^{-164}	9720	155	142	113
Frodo-640-BW_{64}	$\Delta \cdot 2\mathbf{R}_n^{-1} BW_{64}$	$\Delta \cdot 16\mathbb{Z}^{64}$	$640, 8, 8$	2^{15}	3.80	2.25	2^{-164}	9720	162	148	117
Frodo-976	$\Delta \cdot \mathbb{Z}^{64}$	$\Delta \cdot 8\mathbb{Z}^{64}$	$976, 8, 8$	2^{16}	2.3	3	2^{-220}	15744	216	196	156
Frodo-976-E_8	$\Delta \cdot E_8^8$	$\Delta \cdot 8\mathbb{Z}^{64}$	$976, 8, 8$	2^{16}	2.72	3	2^{-220}	15744	224	204	162
Frodo-976-BW_{16}	$\Delta \cdot BW_{16}^4$	$\Delta \cdot 16\mathbb{Z}^{64}$	$976, 8, 8$	2^{16}	2.71	3.25	2^{-220}	15744	224	204	161
Frodo-976-BW_{64}	$\Delta \cdot 2\mathbf{R}_n^{-1} BW_{64}$	$\Delta \cdot 32\mathbb{Z}^{64}$	$976, 8, 8$	2^{16}	3.20	3.25	2^{-220}	15744	232	211	167

TABLE 3: The recommended parameter sets with smaller size of ciphertext.

	Structure of lattice code		$n', \bar{n}, \bar{m}$	q	σ	B	DFR	c size (bytes)	Security		
	Λ_f	Λ_c							C	Q	P
Frodo-640	$\Delta \cdot \mathbb{Z}^{64}$	$\Delta \cdot 4\mathbb{Z}^{64}$	$640, 8, 8$	2^{15}	2.75	2	2^{-164}	9720	149	136	109
Frodo-640-E_8	$\Delta \cdot E_8^8$	$\Delta \cdot 4\mathbb{Z}^{64}$	$640, 8, 8$	2^{14}	2.3	2	2^{-163}	9072	156	143	114
Frodo-640-BW_{16}	$\Delta \cdot BW_{16}^4$	$\Delta \cdot 8\mathbb{Z}^{64}$	$640, 8, 8$	2^{14}	2.3	2.25	2^{-160}	9072	156	143	114
Frodo-640-BW_{64}	$\Delta \cdot 2\mathbf{R}_n^{-1} BW_{64}$	$\Delta \cdot 16\mathbb{Z}^{64}$	$640, 8, 8$	2^{14}	2.7	2.25	2^{-160}	9072	163	149	118
Frodo-976	$\Delta \cdot \mathbb{Z}^{64}$	$\Delta \cdot 8\mathbb{Z}^{64}$	$976, 8, 8$	2^{16}	2.3	3	2^{-220}	15744	216	196	156
Frodo-976-E_8	$\Delta \cdot E_8^8$	$\Delta \cdot 8\mathbb{Z}^{64}$	$976, 8, 8$	2^{15}	1.95	3	2^{-209}	14760	225	205	162
Frodo-976-BW_{16}	$\Delta \cdot BW_{16}^4$	$\Delta \cdot 16\mathbb{Z}^{64}$	$976, 8, 8$	2^{15}	1.95	3.25	2^{-206}	14760	225	205	162
Frodo-976-BW_{64}	$\Delta \cdot 2\mathbf{R}_n^{-1} BW_{64}$	$\Delta \cdot 32\mathbb{Z}^{64}$	$976, 8, 8$	2^{15}	2.3	3.25	2^{-204}	14760	234	213	169

[11] J. Ding, "A simple provably secure key exchange scheme based on the learning with errors problem," *IACR Cryptol. ePrint Arch.*, p. 688, 2012.

[12] E. Alkim, L. Ducas, T. Pöppelmann, and P. Schwabe, "Post-quantum key exchange - A new hope," in *25th USENIX Security Symposium, USENIX Security 16, Austin, TX, USA*, pp. 327–343, 2016.

[13] Z. Jin, S. Shen, and Y. Zhao, "Compact and flexible KEM from ideal lattice," *IEEE Transactions on Information Theory*, 2022.

[14] A. Kawachi, K. Tanaka, and K. Xagawa, "Multi-bit cryptosystems based on lattice problems," in *Public Key Cryptography - PKC 2007, 10th International Conference on Practice and Theory in Public-Key Cryptography, Beijing, China*, pp. 315–329, 2007.

[15] X. Lu, Y. Liu, Z. Zhang, D. Jia, H. Xue, J. He, and B. Li, "LAC: practical ring-lwe based public-key encryption with byte-level modulus," *IACR Cryptol. ePrint Arch.*, p. 1009, 2018.

[16] J. Wang and C. Ling, "How to construct polar codes for ring-lwe-based public key encryption," *Entropy*, vol. 23, no. 8, p. 938, 2021.

[17] J. D'Anvers, M. Tiepelt, F. Vercauteren, and I. Verbauwhede, "Timing attacks on error correcting codes in post-quantum schemes," in *Proceedings of ACM Workshop on Theory of Implementation Security, TIS@CCS 2019, London, UK, November 11, 2019*, pp. 2–9, 2019.

[18] G. Ungerboeck, "Channel coding with multilevel/phase signals," *IEEE Trans. Inf. Theory*, vol. 28, no. 1, pp. 55–66, 1982.

[19] G. D. F. Jr., "Coset codes-II: Binary lattices and related codes," *IEEE Trans. Inf. Theory*, vol. 34, no. 5, pp. 1152–1187, 1988.

[20] R. Zamir, *Lattice Coding for Signals and Networks*. Cambridge, UK: Cambridge University Press, 2014.

[21] U. Erez and R. Zamir, "Achieving 1/2 log (1+SNR) on the AWGN channel with lattice encoding and decoding," *IEEE Trans. Inf. Theory*, vol. 50, no. 10, pp. 2293–2314, Oct. 2004.

[22] L. Liu, Y. Yan, C. Ling, and X. Wu, "Construction of capacity-achieving lattice codes: Polar lattices," *IEEE Trans. Commun.*, vol. 67, no. 2, pp. 915–928, 2019.

[23] P. R. B. da Silva and D. Silva, "Multilevel LDPC lattices with efficient encoding and decoding and a generalization of Construction D," *IEEE Trans. Inf. Theory*, vol. 65, no. 5, pp. 3246–3260, 2019.

[24] M. S. Viazovska, "The sphere packing problem in dimension 8," *Annals of Mathematics*, pp. 991–1015, 2017.

[25] H. Cohn, A. Kumar, S. Miller, D. Radchenko, and M. Viazovska, "The sphere packing problem in dimension 24," *Annals of Mathematics*, vol. 185, no. 3, pp. 1017–1033, 2017.

[26] J. W. Bos, L. Ducas, E. Kiltz, T. Lepoint, V. Lyubashevsky, J. M. Schanck, P. Schwabe, G. Seiler, and D. Stehlé, "CRYSTALS - kyber: A cca-secure module-lattice-based KEM," in *2018 IEEE European Symposium on Security and Privacy, EuroS&P 2018, London, United Kingdom*, pp. 353–367, 2018.

[27] T. M. Cover, *Elements of information theory*. John Wiley & Sons, 1999.

[28] J. H. Conway and N. J. A. Sloane, *Sphere Packings, Lattices and Groups*, 3rd ed. Springer New York, 1999.

[29] ——, "Fast quantizing and decoding and algorithms for lattice quantizers and codes," *IEEE Trans. Inf. Theory*, vol. 28, no. 2, pp. 227–231, 1982.

[30] J. Bos, L. Ducas, I. Mironov, V. Nikolaenko, A. Raghunathan, and D. Stebila. (2016). Classical, quantum, and plausible (conservative) quantum cost estimates. [Online]. Available: https://github.com/lwe-frodo/parameter-selection/blob/master/pqsec.py

[31] E. Fujisaki and T. Okamoto, "Secure integration of asymmetric and symmetric encryption schemes," in *Advances in Cryptology - CRYPTO '99, 19th Annual International Cryptology Conference, Santa Barbara, California, USA, August 15-19, 1999, Proceedings*, pp. 537–554, 1999.

[32] D. Hofheinz, K. Hövelmanns, and E. Kiltz, "A modular analysis of the fujisaki-okamoto transformation," in *Theory of Cryptography - 15th International Conference, TCC 2017, Baltimore, MD, USA, November 12-15, 2017, Proceedings, Part I*, pp. 341–371, 2017.