Simulation-Secure Threshold PKE from Standard (Ring-)LWE

Hiroki Okada[1,2] and Tsuyoshi Takagi[2]

[1] KDDI Research, Inc., Japan
[2] The University of Tokyo, Japan

Abstract. Threshold public key encryption (ThPKE) is PKE that can be decrypted by collecting "partial decryptions" from t ($\leq N$) out of N parties. ThPKE based on the learning with errors problem (LWE) is particularly important because it can be extended to threshold fully homomorphic encryption (ThFHE). ThPKE and ThFHE are fundamental tools for constructing multiparty computation (MPC) protocols: In 2023, NIST initiated a project (NIST IR 8214C) to establish guidelines for implementing threshold cryptosystems. Because MPC often requires simulation-security (SS), ThPKE schemes that satisfy SS (SS-ThPKE) are also important. Recently, Micciancio and Suhl (ePrint 2023/1728) presented an efficient SS-ThPKE scheme based on LWE with a polynomial modulus. However, the scheme requires to use a nonstandard problem called "known-norm LWE" for the security proof because the norm $\|\mathbf{e}\|$ of the error of the public key is leaked from the partial decryptions. This leads to the following two challenges: 1) The construction based on LWE incurs a security loss of approximately 13 bits for 128-bit security. 2) No construction based on (standard) Ring-LWE has been presented. In this paper, we address both of these challenges: we propose an efficient SS-ThPKE scheme whose security is (directly) reduced from standard (Ring-)LWE with a polynomial modulus. The core technique of our construction is what we call "error sharing". We distribute shares of a small error $\boldsymbol{\zeta}$ via secret sharing, and use them to prevent leakage of $\|\mathbf{e}\|$ from partial decryptions.

1 Introduction

Threshold public key encryption (ThPKE) is public key encryption (PKE) whose ciphertexts can be decrypted by collecting "partial decryptions" from t out of N parties, where N is the total number of parties and t is a threshold. One of the attractive applications of ThPKE is threshold fully homomorphic encryption (ThFHE), which can essentially be constructed by replacing the PKE of ThPKE with fully homomorphic encryption (FHE). ThPKE and ThFHE are fundamental tools for constructing multiparty computation (MPC) protocols: In 2023, the National Institute of Standards and Technology (NIST) initiated a project to establish guidelines and recommendations for implementing those threshold cryptosystems [14]. ThFHE is a particularly important cryptographic tool that can be used to construct round optimal MPCs [5, 20, 22] and the

universal thresholdizer [9], which can be used to add threshold functionality to many cryptosystems such as CCA-secure PKE, signature schemes, pseudorandom functions (PRF) and functional encryptions.

The construction of FHE was first realized by Gentry [25] using the ideal lattice. In particular, constructions based on the learning with errors problem (LWE, Definition 3.7) [40] and Ring-LWE [29] (Definition 5.1) are efficient, leading to active research in this field [12, 16, 17]. Thus, ThPKE based on (Ring-)LWE is especially important because it can be extended to efficient ThFHE schemes. Moreover, ThPKE schemes with simulation-based security (SS) [15], which we call SS-ThPKE, are crucial because MPC is often formulated with SS.

However, existing SS-ThPKE schemes based on (Ring-)LWE, e.g., [4, 7, 9], are not efficient because they are based on (Ring-)LWE with a *superpolynomially* large modulus q; thus, the public key and the ciphertexts are superpolynomially long. Recently, Boudgoust and Scholl [10] and Chowdhury et al. [18] proposed ThPKE based on (Ring-)LWE with a polynomial modulus q, but their security proofs are game-based; thus, SS is not shown. Furthermore, their reductions are not tight because they are based on the Rényi divergence technique [6,39]. Dahl et al. [19] also proposed an SS-ThPKE scheme based on (Ring-)LWE with a polynomial modulus q; however, this scheme is not efficient because its modulus is switched to be superpolynomially large during decryption.

Micciancio and Suhl [35] recently proposed an efficient SS-ThPKE scheme based on LWE with a polynomial modulus q. However, the security proof of the scheme uses a nonstandard assumption called "known-norm LWE" (or "known-covariance Ring-LWE"), which is a problem to find the secret vector $\mathbf{s}$ from given an LWE sample $(\mathbf{A}, \mathbf{b} := \mathbf{As} + \mathbf{e})$ and the norm $\|\mathbf{e}\|$ of the error $\mathbf{e}$, because $\|\mathbf{e}\|$ in the public key (=an LWE sample) is leaked from the partial decryptions. As a consequence of using "known-norm LWE"(or "known-covariance Ring-LWE"), two challenges remain:

Question 1. *The SS-ThPKE scheme based on LWE of [35] relies on a nontight reduction from LWE to "known-norm LWE", which incurs a security loss of approximately* 13 *bits in some typical parameters selected for* 128*-bit security. Can we construct SS-ThPKE based on LWE with a polynomial modulus q without this loss? (See the upper part of Fig. 1.)*

Question 2. *The authors of [35] also proposed an SS-ThPKE scheme based on a nonstandard assumption called "known-covariance Ring-LWE", to which no reduction from standard assumptions such as Ring-LWE has been shown. Can we construct SS-ThPKE based on Ring-LWE with a polynomial modulus q? (See the lower part of Fig. 1.)*

Question 1 results in an efficiency loss for the LWE-based scheme. Question 2 is crucial because most practical lattice-based PKE/FHE schemes (e.g., [12, 17, 28]) are constructed based on Ring-LWE.

Our Contributions. We address both Questions 1 and 2 above. We propose:

Fig. 1. Overview of the challenges (Questions 1 and 2) in ThPKE of [35], and our ThPKE schemes (Theorems 4.6 and 5.14).

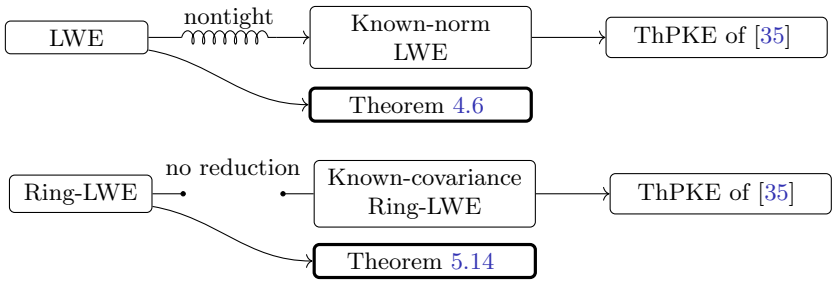

Table 1. Comparison of threshold PKE schemes from (Ring-)LWE

	$q = \mathrm{poly}(\lambda)$	Simulation Security	Tightness	Ring-LWE
[7]	✗	✓	✓	✓
[9]	✗[1]	✓	✓	✓
[10]	✓	✗	✗	✓
[35]	✓	✓	✗	✗
Theorem 4.6	✓	✓	✓	-
Theorem 5.14	✓	✓	✓	✓

[1] [9, Construction 5.6] (this is ThFHE, which subsumes ThPKE), requires a superpolynomial q to satisfy simulation-security. [8, Construction 8.29] is a generic construction of compact ThPKE based on the *universal thresholdizer*, which is constructed from noncompact ThFHE and NIZK.

- (Theorem 4.6) An SS-ThPKE scheme from LWE with a polynomial modulus q that does not use "known-norm LWE", and
- (Theorem 5.14) An SS-ThPKE scheme from Ring-LWE with a polynomial modulus q (that does not use "known-covariance Ring-LWE").

In Fig. 1, we illustrate the relations among Questions 1 and 2 and our ThPKE schemes. We also briefly summarize the differences between the existing schemes and ours in Table 1.

In addition to the main contributions mentioned above, we provide the following supplementary contributions, some of which may be of independent interest:

- *Reformulation of the Reused-A-LWE Problem*: Micciancio and Suhl [35] introduced a variant of LWE called the *Reused-A-LWE problem*, where two LWE samples $(\mathbf{A}, \mathbf{b}_1 = \mathbf{A}\mathbf{s} + \mathbf{e}_1)$ and $(\mathbf{A}, \mathbf{b}_2 = \mathbf{A}\mathbf{s} + \mathbf{e}_2)$ of a common (i.e., "reused") $\mathbf{A}$ with different error distributions are given. While [35] showed a reduction from LWE to Reused-A-LWE, the error distribution was limited to a continuous Gaussian distribution. In this paper, we generalize

the Reused-A-LWE problem (Definition 3.12) and show a reduction from LWE to Reused-A-LWE for any (continuous/discrete) error distribution (Theorem 3.15). Furthermore, we show that the loss of the error parameter in our reduction is smaller than that in [35] (Corollary 3.18).

- *Generalization of the Error Distribution of ThPKE*: In the ThPKE of [35], the error distribution for the LWE of the public key is limited to a *continuous* Gaussian distribution. However, *discrete* error distributions are desirable for practical implementation: If we implement continuous distributions with floating-point numbers, we must discuss the (negative) effects of rounding errors. In this paper, we construct our ThPKE scheme from LWE only with discrete error distributions. Specifically, we use (the standard) discrete Gaussian distribution over the integer lattice.
- *Generalization of Access Structures*: The ThPKE of [35] supports only (N, N)-threshold access structures (see Definition 2.24), where decryption is possible only when N out of N parties are involved. In contrast, our ThPKE supports all possible access structures achievable with binary coefficient linear secret sharing (BinLSS, see Definition 2.22), which include arbitrary (t, N)-access structures. This generalization is derived by simply applying the techniques presented in [9].

Technical Overview. We explain how we improve the LWE-based ThPKE scheme proposed by Micciancio and Suhl [35]. Because we can improve the "known-covariance Ring-LWE"-based ThPKE scheme of [35] with essentially the same approach, we omit the detail here.

The fundamental issue in the LWE-based ThPKE of [35] is that the adversary conducting chosen plaintext attack (CPA) can derive the norm $\|\mathbf{e}\|$ of the error in the public key, which is an LWE sample $(\mathbf{A}, \mathbf{b} = \mathbf{A}\mathbf{s}+\mathbf{e})$. We address this issue in our scheme (Algorithm 1) by "error sharing" technique: we distribute the shares $(\mathsf{err}_1, \ldots, \mathsf{err}_N)$ of a short error $\mathsf{err} := \boldsymbol{\zeta}$ with secret sharing, in addition to the shares $(\mathsf{sk}_1, \ldots, \mathsf{sk}_N)$ of the secret key $\mathsf{sk} := \mathbf{s}$. This modification is made to secure the partial decryption process. The partial decryptions of our ThPKE ($\mathsf{PartDec}$: Line 9 in Algorithm 1) include a randomized value derived from err_i in addition to the conventional "smudging noise" e_{sm}. Note that although the standard deviation of e_{sm} was superpolynomially large in the conventional constructions such as [7,9] (and that is why it is called "smudging"), we set it as polynomially small as with [10,35].

By this construction, the information exposed to adversaries is changed: while the scheme of [35] discloses the error norm $\|\mathbf{e}\|$, our scheme discloses $\sqrt{\|\mathbf{e}\|^2 + \|\boldsymbol{\zeta}\|^2}$ (more generally, it discloses the distribution χ_{Sim} defined in Eq. (3)). In addition, we choose the short error $\boldsymbol{\zeta}$ conditioned the fixed $\mathbf{e}$ generated by KeyGen so that $\sqrt{\|\mathbf{e}\|^2 + \|\boldsymbol{\zeta}\|^2}$ becomes a public constant B that does not contain any information about $\|\mathbf{e}\|$ (Theorem 4.6). This technique can be described as applying "padding" to the value of $\|\mathbf{e}\|$, which varies depending on $\mathbf{e}$, to ensure it reaches a constant value B. In addition, due to the (information-theoretic) security of secret sharing (Definition 2.21), no information about $\boldsymbol{\zeta}$

is revealed to adversaries who do not have a valid set of shares that enables decryption. Therefore, our scheme does not leak information about $\|\mathbf{e}\|$ (or $\mathbf{e}$ itself) to the adversary. Thus, the security of our scheme is directly reduced from (standard) LWE with $q = \mathrm{poly}(n)$, without using "known-norm LWE" (Theorem 4.6).

Related Works. Some readers may think we can use *multihint extended-LWE* (mheLWE) [2] to improve the reduction from LWE to known-norm LWE, but this is not trivial. Note that mheLWE is a multihint generalization of *extended-LWE* (extLWE) [3, 13, 37], and it is quite different from multisecret generalization of extended-LWE, i.e., extLWE^t in [13]. $\mathsf{d\text{-}mheLWE}(n, m, q, \chi_{\mathsf{LWE}}, \chi_{\mathsf{hint}})$ is a problem to distinguish between

$$(\mathbf{A}, \mathbf{b}, \{y_i, \mathbf{z}_i\}_{i\in[t]}, \{y_i + \mathbf{z}_i^\intercal \mathbf{e}\}_{i\in[t]},) \text{ and } (\mathbf{A}, \mathbf{u}, \{y_i, \mathbf{z}_i\}_{i\in[t]}, \{y_i + \mathbf{z}_i^\intercal \mathbf{e}\}_{i\in[t]}),$$

where $(\mathbf{A}, \mathbf{b} := \mathbf{A}\mathbf{s} + \mathbf{e}) \leftarrow \mathsf{LWE}(n, m, q, \chi_{\mathsf{LWE}})$ (Definition 3.6), χ_{hint} is a (efficiently samplable) distribution specified by the adversary, and $(y_i, \mathbf{z}_i) \leftarrow \chi_{\mathsf{hint}}$. Note that the above mheLWE is more generalized than that of [3, 13] with the additional term y_i as in [37]. It is shown in [2, Theorem 4] that there is a reduction from $\mathsf{d\text{-}LWE}(n - t, m, q, \chi_{\mathsf{LWE}})$ to $\mathsf{d\text{-}mheLWE}(n, m, q, \chi_{\mathsf{LWE}}, t, \chi_{\mathsf{hint}})$ that loses the advantage by at most $2^{\Omega(t-n)}$. Note that the dimension of $\mathsf{d\text{-}LWE}$ is $n - t$. Thus, we require the number of hints $t < n$ to be a priori bounded. This condition on t is acceptable for applying mheLWE to some functional encryption schemes as demonstrated in [2]. In ThPKE of [35], the CPA adversary can obtain

$$(\mathbf{A}, \mathbf{b} := \mathbf{A}\mathbf{s} + \mathbf{e}, \{e_i^{\mathsf{sm}} + \mathbf{r}_i^\intercal \mathbf{e}\}_{i\in[t]}) \text{ for any (unbounded) } t = \mathrm{poly}(\lambda), \quad (1)$$

where $(\mathbf{A}, \mathbf{b}) \leftarrow \mathsf{LWE}(n, m, q, \chi_{\mathsf{LWE}})$, $e_i^{\mathsf{sm}} \leftarrow \mathcal{N}_{\sigma_{\mathsf{sm}}}$, and $\mathbf{r}_i \leftarrow \mathcal{N}_{\sigma_{\mathsf{enc}}}$. (Thus, the adversary can accurately estimate $\|\mathbf{e}\|$ because $e_i^{\mathsf{sm}} + \mathbf{r}_i^\intercal \mathbf{e} \sim \mathcal{N}_{\sqrt{\sigma_{\mathsf{sm}}^2 + \sigma_{\mathsf{enc}}^2 \|\mathbf{e}\|^2}}$.) We may simulate the distribution of Eq. (1) by $\mathsf{mheLWE}(n, m, q, \chi_{\mathsf{LWE}}, t, \chi_{\mathsf{hint}})$:

$$(\mathbf{A}, \mathbf{b} := \mathbf{A}\mathbf{s} + \mathbf{e}, \{y_i, \mathbf{z}_i\}_{i\in[t]}, \{y_i + \mathbf{z}_i^\intercal \mathbf{e}\}_{i\in[t]}),$$

where $y_i \leftarrow \mathcal{N}_{\sigma_{\mathsf{sm}}}$ and $\mathbf{z}_i \leftarrow \mathcal{N}_{\sigma_{\mathsf{enc}}}$. However, to obtain a (nontrivial) reduction from LWE, we require t to be a priori bounded and less than n, which does not meet the standard definition of the CPA adversary.

Organization. The remainder of this paper is organized as follows: In Section 2, we provide necessary definitions and lemmas, and describe the construction of linear secret sharing. In Section 3, we provide several lemmas related to the hardness of LWE, and then, we generalize the Reused-A-LWE problem and show the reduction from LWE. In Section 4 (resp. Section 5), we propose our LWE-based (resp. Ring-LWE-based) ThPKE scheme and prove its correctness and simulation-based security. Finally, we summarize this paper and discuss future work in Section 6.

2 Preliminaries

In Section 2.1, we provide the basic notations used in this paper. Then, we provide necessary definitions and lemmas related to statistics in Section 2.2 and Gaussian distribution in Section 2.3. Finally, we describe the construction of the linear secret sharing in Section 2.4.

2.1 Notations

We denote the base 2 logarithm by log. For any natural number $N \in \mathbb{N}$, the set of the first N natural numbers is denoted by $[N] = \{1, \ldots, N\}$. We denote the number of elements in a set S by $|S|$. When the set $\{x_i\}_{i \in S}$ is given, the index set S is also given. We define $\mathbb{Z}_q := \mathbb{Z}/q\mathbb{Z}$ and $\mathbb{R}_q := \mathbb{R}/q\mathbb{R}$ for a modulus $q \in \mathbb{N}$.

We use bold lower-case for vectors and bold upper-case for matrices. For a vector $\mathbf{x} = (x_1, \ldots, x_n)$, we denote the ith component x_i by $\mathbf{x}[i]$. The transpose of $\mathbf{x}$ is written as $\mathbf{x}^\intercal$. We denote the l_2-norm and l_∞-norm of $\mathbf{x}$ by $\|\mathbf{x}\|$ and $\|\mathbf{x}\|_\infty$, respectively. The identity matrix is denoted by $\mathbf{I}_n \in \mathbb{Z}^{n \times n}$. We write $\mathbf{\Sigma} \succ 0$ if $\mathbf{\Sigma}$ is positive definite. We say that $\mathbf{\Sigma}_1 \succ \mathbf{\Sigma}_2$ if $\mathbf{\Sigma}_1 - \mathbf{\Sigma}_2 \succ 0$. The largest and smallest singular values of a matrix $\mathbf{S}$ are denoted by $\sigma_{\max}(\mathbf{S})$ and $\sigma_{\min}(\mathbf{S})$. The Frobenius norm of a matrix $\mathbf{S}$ is $\|\mathbf{S}\|_F = \sqrt{\mathrm{tr}(\mathbf{S}^\intercal \mathbf{S})}$. Note that we have $\sigma_{\max}(\mathbf{S}) \leq \|\mathbf{S}\|_F \leq \sqrt{\mathrm{rank}(\mathbf{S})}\sigma_{\max}(\mathbf{S})$.

Unless otherwise specified, we treat $\lambda \in \mathbb{N}$ or $n \in \mathbb{N}$ as a security parameter. We write $\mathrm{negl}(n) = n^{-\omega(1)}$ for the set of negligible functions and $\mathrm{poly}(n) = n^{O(1)}$ for the set of polynomial functions. We call the function $(1 - \mathrm{negl}(n))$ overwhelming. The term "probabilistic polynomial time" is abbreviated as PPT. For problems P_1 and P_2, we denote the PPT reduction from P_1 to P_2 by $\mathsf{P}_1 \leq \mathsf{P}_2$.

2.2 Statistics

We write $X_1, X_2 \overset{\text{iid}}{\sim} \chi$ when variables X_1 and X_2 are independent and identically distributed (i.i.d.) according to χ. For a distribution χ over $\mathbb{R}$, we denote by $\lfloor \chi \rceil$ the distribution of $\{\lfloor X \rceil \mid X \leftarrow \chi\}$. We denote the uniform distribution over a set S by $\mathcal{U}(S)$, and denote the random variable X sampled from $\mathcal{U}(S)$ by $X \xleftarrow{\$} S$.

We provide the necessary definitions, lemmas, and facts as follows:

Definition 2.1. *The statistical distance between χ_1 and χ_2 is defined as $\Delta(\chi_1, \chi_2) := \frac{1}{2}\sum_{x \in \Omega} |f_{\chi_1}(x) - f_{\chi_2}(x)|$, where $f_{\chi_1}(x)$ and $f_{\chi_2}(x)$ are the probability functions of χ_1 and χ_2, respectively, and $\Omega := \mathrm{Supp}(\chi_1) \cup \mathrm{Supp}(\chi_2)$. This definition is naturally extended to continuous distributions.*

Definition 2.2 ($\chi_1 \overset{\mathsf{stat}}{\approx} \chi_2$). *Distributions χ_1 and χ_2 are* statistically indistinguishable *and denote as $\chi_1 \overset{\mathsf{stat}}{\approx} \chi_2$ if we have: $\Delta(\chi_1, \chi_2) = \mathrm{negl}(\lambda)$*

Definition 2.3 ($\chi_1 \overset{\mathsf{comp}}{\approx} \chi_2$). *Distributions χ_1 and χ_2 over the set Ω are* computationally indistinguishable *and denoted as $\chi_1 \overset{\mathsf{comp}}{\approx} \chi_2$ if $|\Pr[\mathcal{A}(\chi_1) = 1] - \Pr[\mathcal{A}(\chi_2) = 1]| = \mathrm{negl}(\lambda)$ holds for any PPT algorithm $\mathcal{A} : \Omega \to \{0, 1\}$.*

Definition 2.4. *The min-entropy of a discrete distribution χ is defined as* $H_\infty(\chi) = \log \min_{x \in \mathrm{Supp}(\chi)} 1/\Pr_{X \leftarrow \chi}[X = x]$.

Lemma 2.5 (Leftover Hash Lemma [11, Lemma 2.1]). *Let q be prime and $m, n \in \mathbb{N}$. Let $\mathbf{r}$ be a random variable over $\mathbb{Z}_q^m$ and $\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{m \times n}$. Then, we have* $\Delta((\mathbf{A}, \mathbf{r}^\intercal \mathbf{A}), (\mathbf{A}, \mathcal{U}(\mathbb{Z}_q^n))) \leq \sqrt{q^n 2^{-H_\infty(\mathbf{r})}}$.

Fact 2.6. *For $m \geq n \log q + 2\lambda$ and $\mathbf{r} \sim \mathcal{U}(\{0,1\}^m)$, $\sqrt{q^n 2^{-H_\infty(\mathbf{r})}} \leq 2^{-\lambda}$ holds.*

Lemma 2.7 (In [30, Lemma 4.8]). *For any $m \geq n + \omega(\log n)$, $\mathbf{A} \leftarrow \mathcal{U}(\mathbb{Z}_q^{m \times n})$ is nonsingular, i.e., the rows of $\mathbf{A}$ generate $\mathbb{Z}^n$, with overwhelming probability.*

We say that χ is B-bounded if $\chi < B$ with overwhelming probability:

Definition 2.8. *Let χ be a (continuous / discrete) distribution over $\mathbb{R}$. We say χ is* B-bounded *if $\Pr_{X \leftarrow \chi}[|X| \geq B] = \mathrm{negl}(n)$ holds.*

2.3 Gaussians

The continuous Gaussian distribution with a mean of 0 and a standard deviation $\sigma > 0$ is denoted as $\mathcal{N}_\sigma$.

For a rank-n matrix $\mathbf{S} \in \mathbb{R}^{n \times m}$, the (centered) ellipsoid Gaussian function on $\mathbb{R}^n$ with the (scaled) covariance matrix $\mathbf{\Sigma} = \mathbf{S}\mathbf{S}^\intercal \in \mathbb{R}^{n \times n}$ is defined as: $\rho_{\mathbf{S}}(\mathbf{x}) := \exp(-\pi \mathbf{x}^\intercal (\mathbf{S}\mathbf{S}^\intercal)^{-1} \mathbf{x})$. The function $\rho_{\mathbf{S}}(\mathbf{x})$ is determined exactly by $\mathbf{\Sigma} \succ 0$, and there exist a unique $\sqrt{\mathbf{\Sigma}} \succ 0$ such that $\sqrt{\mathbf{\Sigma}}\sqrt{\mathbf{\Sigma}} = \mathbf{\Sigma}$ (see, e.g., [27, Theorem 7.2.6]). Thus, we have $\rho_{\mathbf{S}} = \rho_{\sqrt{\mathbf{\Sigma}}}$. When $\mathbf{S} = s\mathbf{I}_n$, we write $\rho_{\mathbf{S}}$ as ρ_s. For any set $A \subseteq \mathbb{R}^n$, we define $\rho_{\mathbf{S}}(A) := \sum_{\mathbf{x} \in A} \rho_{\mathbf{S}}(\mathbf{x})$.

A lattice $\mathcal{L}$ is the set of all integer linear combinations of linearly independent vectors $\mathbf{b}_1, \cdots, \mathbf{b}_n \in \mathbb{R}^m$, i.e., $\mathcal{L} = \{\sum_{i=1}^n z_i \mathbf{b}_i \mid \mathbf{z} \in \mathbb{Z}^n\}$. We say that the rank of this lattice is n and its dimension is m. If $n = m$, we call the lattice *full rank*. While the integer lattice $\mathcal{L} := \mathbb{Z}^n$ is the primary focus in this paper, we sometimes describe lemmas with the general lattice $\mathcal{L}$. We define the discrete Gaussian distribution over the lattice $\mathcal{L}$ as follows:

Definition 2.9. *For a rank-n matrix $\mathbf{S} \in \mathbb{R}^{n \times m}$, the (centered)* discrete Gaussian distribution *with the covariance matrix $\mathbf{\Sigma} := \mathbf{S}\mathbf{S}^\intercal$ is defined as*

$$\forall \mathbf{x} \in \mathbb{Z}^n, \mathcal{D}_{\mathbb{Z}^n, \mathbf{S}}(\mathbf{x}) = \rho_{\mathbf{S}}(\mathbf{x}) / \rho_{\mathbf{S}}(\mathbb{Z}^n).$$

In particular, when $\mathbf{S}\mathbf{S}^\intercal = s^2 \mathbf{I}_n$ for some $s > 0$, we write $\mathcal{D}_{\mathbb{Z}^n, \mathbf{S}}$ as $\mathcal{D}_{\mathbb{Z}^n, s}$ and call it as the spherical *discrete Gaussian distribution.*

Note that the n-dimensional vector whose elements are i.i.d 1-dimensional discrete Gaussian $D_{\mathbb{Z},s}$ follows $\mathcal{D}_{\mathbb{Z}^n,s}$:

Fact 2.10. *Let $x_1, \ldots, x_n \overset{\text{iid}}{\sim} \mathcal{D}_{\mathbb{Z},s}$ and $\mathbf{x} := (x_1, \ldots, x_n)^\intercal$; then $\mathbf{x} \sim \mathcal{D}_{\mathbb{Z}^n,s}$.*

Proof. The joint distribution function of $\mathbf{x}$ is $\prod_{i=1}^n (\rho_s(x_i) / \sum_{y_i \in \mathbb{Z}} \rho_s(y_i)) = \rho_s(\mathbf{x}) / \sum_{\mathbf{y} \in \mathbb{Z}^n} \rho_s(\mathbf{y}) = \mathcal{D}_{\mathbb{Z}^n,s}(\mathbf{x})$. □

Given a lattice $\mathcal{L}$ and $\epsilon > 0$, we define the smoothing parameter of $\mathcal{L}$ as $\eta_\epsilon(\mathcal{L}) = \min\{s \mid \rho_{1/s}(\widehat{\mathcal{L}}) \leq 1 + \epsilon\}$. In particular, for any $\epsilon > 0$, we have $\eta_\epsilon(\mathbb{Z}^n) \leq$

$\eta_\epsilon^+(\mathbb{Z}^n) := \sqrt{\ln(2n(1+1/\epsilon))/\pi}$. We also define $\tilde{\eta}_\epsilon(\cdot) := \sqrt{2}\eta_\epsilon(\cdot)$ and $\tilde{\eta}_\epsilon^+(\mathbb{Z}^n) := \sqrt{2}\eta_\epsilon^+(\mathbb{Z}^n)$ for simplicity of notation. Unless otherwise specified, we set $\epsilon = \mathrm{negl}(\lambda)$. Note that if we set, e.g., $\epsilon = (-1+2^{\frac{\lambda}{c}-1}/n)^{-1} (\in \mathrm{negl}(\lambda))$ then we have $\eta_\epsilon^+(\mathbb{Z}^n) = \sqrt{\lambda/c\pi}$ for any $c = O(1)$. We also extend the smoothing parameter to positive definite matrices:

Definition 2.11 ([38, Definition 2.3]). *Let $\mathbf{\Sigma} \succ 0$ be any positive definite matrix. For any lattice $\mathcal{L}$, we say that $\sqrt{\mathbf{\Sigma}} \geq \eta_\epsilon(\mathcal{L})$ if $\eta_\epsilon(\sqrt{\mathbf{\Sigma}}^{-1}\mathcal{L}) \leq 1$.*

The denominator of $\mathcal{D}_{\mathbb{Z}^n,s}(\mathbf{x})$ can be approximated as follows:

Fact 2.12 (Special case of [40, Claim 3.8]). *For any $\epsilon > 0$ and $s \geq \eta_\epsilon^+(\mathbb{Z}^n)$, we have $\rho_s(\mathbb{Z}^n) \in (1 \pm \epsilon)s^n$.*

The linear sum of $\mathcal{D}_{\mathbb{Z},r}$ is statistically close to $\mathcal{D}_{\mathbb{Z}^n,r'}$ for some $r' > r$:

Lemma 2.13 ([32, Theorem 3.3]). *Let $\mathcal{L}$ be an n-dimensional lattice, $\mathbf{e} \in \mathbb{Z}^m$ a nonzero integer vector such that $\gcd(\mathbf{e}) = 1$, $s_i \geq \|\mathbf{e}\|_\infty \tilde{\eta}_\epsilon(\mathcal{L})$, $r_i \leftarrow D_{\mathbb{Z},s_i}$ independently for $i = 1, \ldots, m$, and define $\mathbf{r} := (r_1, \ldots, r_m)^\intercal$. Then, the distribution of $\tilde{e} = \mathbf{r}^\intercal \mathbf{e}$ is statistically close to $D_{\mathcal{L},\tilde{s}}$, where $\tilde{s} = \sqrt{\sum_{i=1}^m (s_i e_i)^2}$. In particular, if $s = s_1 = \cdots = s_m$, then $\tilde{s} = s\|\mathbf{e}\|$.*

In particular, if $m = 2$ and $\mathbf{e} = (1,1)^\intercal$, then the distribution of $r_1 + r_2$ is statistically close to $D_{\mathbb{Z},\sqrt{s_1^2+s_2^2}}$ for $s_1, s_2 \geq \tilde{\eta}_\epsilon^+(\mathbb{Z})$.

The linear transformation of a spherical discrete Gaussian is (statistically close to) a (ellipsoid) discrete Gaussian:

Lemma 2.14 ([21, Lemma 3]). *Let $\epsilon = \mathrm{negl}(\lambda)$ and $s \geq \eta_\epsilon^+(\mathbb{Z}^n)$. For any nonsingular matrix $\mathbf{T} \in \mathbb{Z}^{n\times n}$, we have $\mathbf{T} \cdot D_{\mathbb{Z}^n,s} \overset{\mathsf{stat}}{\approx} D_{\mathbb{Z}^n,s\mathbf{T}}$.*

It is also known that the sum of two ellipsoid discrete Gaussians is statistically close to an ellipsoid discrete Gaussian:

Lemma 2.15 (Special case of [24, Theorem 3]). *Let $\epsilon = \mathrm{negl}(\lambda)$. Let $\mathbf{S}_1, \mathbf{S}_2 \in \mathbb{Z}^{n\times n}$ be nonsingular matrices such that $\eta_\epsilon^+(\mathbb{Z}^n) \leq \mathbf{S}_1$ and $\eta_\epsilon^+(\mathbb{Z}^n) \leq \mathbf{S}_2$. Then, we have $D_{\mathbb{Z}^n,\mathbf{S}_1} + D_{\mathbb{Z}^n,\mathbf{S}_2} \overset{\mathsf{stat}}{\approx} D_{\mathbb{Z}^n,\sqrt{\mathbf{S}_1\mathbf{S}_1^\intercal + \mathbf{S}_2\mathbf{S}_2^\intercal}}$*

The tail bound of $\mathcal{D}_{\mathbb{Z}^n,\mathbf{S}}$, $\mathcal{D}_{\mathbb{Z}^n,s}$ and $\mathcal{D}_{\mathbb{Z},s}$ can be obtained as follows:

Lemma 2.16 ([1, Lemma 3]). *For a rank-n lattice $\mathcal{L}$, $\epsilon \in (0,1)$ and matrix $\mathbf{S}$ s.t. $\sigma_{\min}(\mathbf{S}) \geq \eta_\epsilon(\mathcal{L})$, we have $\Pr_{\mathbf{x}\leftarrow\mathcal{D}_{\mathcal{L},\mathbf{S}}}[\|\mathbf{x}\| > \sigma_{\max}(\mathbf{S})\sqrt{n}] \leq \frac{1+\epsilon}{1-\epsilon} \cdot 2^{-n}$, where $\sigma_{\max}(\mathbf{S})$ and $\sigma_{\min}(\mathbf{S})$ are the largest and smallest singular values of $\mathbf{S}$.*

Lemma 2.17 ([34, Lem. 4.4]). *For any n-dimensional lattice $\mathcal{L}$, $\epsilon \in (0,1)$ and $s \geq \eta_\epsilon(\mathcal{L})$, we have $\Pr_{\mathbf{x}\leftarrow\mathcal{D}_{\mathcal{L},s}}[\|\mathbf{x}\| > s\sqrt{n}] \leq \frac{1+\epsilon}{1-\epsilon} \cdot 2^{-n}$.*

Lemma 2.18. *For any $\epsilon \in (0,1)$, $s \geq \eta_\epsilon^+(\mathbb{Z})$ and $t \in \mathbb{N}$, we have $\Pr_{x\leftarrow\mathcal{D}_{\mathbb{Z},s}}[|x| \geq t] \leq \frac{s}{(1-\epsilon)\pi t} e^{-\pi t^2/s^2}$. In particular, for any $t = s \cdot \Omega(\sqrt{\lambda})$, we have $\Pr_{x\leftarrow\mathcal{D}_{\mathbb{Z},s}}[|x| \geq t] = \mathrm{negl}(\lambda)$, i.e., $\mathcal{D}_{\mathbb{Z},s}$ is $s \cdot \Omega(\sqrt{\lambda})$-bounded (Definition 2.8).*

Proof. We complete the proof by Fact 2.12 and routine calculation:

$$\begin{aligned}&\mathrm{Pr}_{x\leftarrow\mathcal{D}_{\mathbb{Z},s}}[|x|\geq t] = 2\mathrm{Pr}_{x\leftarrow\mathcal{D}_{\mathbb{Z},s}}[x\geq t] = 2\textstyle\sum_{y=t}^{\infty}\rho_s(y)/\sum_{y\in\mathbb{Z}}\rho_s(y)\\ &\leq \tfrac{2}{1-\epsilon}\textstyle\sum_{y=t}^{\infty}\tfrac{1}{s}e^{-\pi y^2/s^2} \leq \tfrac{2}{(1-\epsilon)t}\textstyle\sum_{y=t}^{\infty}\tfrac{y}{s}e^{-\pi y^2/s^2} \leq \tfrac{2}{(1-\epsilon)t}\textstyle\int_t^{\infty}\tfrac{y}{s}e^{-\pi y^2/s^2}dy \quad \square\end{aligned}$$

2.4 Linear Secret Sharing (LSS)

In this subsection, we describe the construction of linear secret sharing used in our proposed scheme based on the construction presented in [9], along with the related definitions.

Definition 2.19 (Monotone Access Structure). *The power set of a set S is $\mathbb{P}(S) = \{A \mid A \subseteq S\}$. Let $P = \{P_1, \ldots, P_N\}$ be a set of parties. A collection $\mathbb{A} \subseteq \mathbb{P}(P)$ is monotone if we have $B \in \mathbb{A} \implies C \in \mathbb{A}$ for any sets $B \subseteq C (\subseteq P)$. A monotone access structure on P is a monotone collection $\mathbb{A} \subseteq \mathbb{P}(P) \backslash \varnothing$.*

In this paper, we consider only monotone access structures. Hence, we simply refer to them as "access structures."

Definition 2.20 ((In)valid party set). *Let $P = \{P_1, \ldots, P_N\}$ be a set of parties and $\mathbb{A}$ be a monotone access structure on P. The sets $S \in \mathbb{A}$ are called the valid sets and the sets $S \in \mathbb{P}(P) \backslash \mathbb{A}$ are called the invalid sets. Furthermore, we define the maximal invalid party set as $\{S \notin \mathbb{A} \mid \forall P_i \in P \backslash S, S \cup \{P_i\} \in \mathbb{A}\}$, and the minimal valid party set as $\{S \in \mathbb{A} \mid \forall S' \subsetneq S, S' \notin \mathbb{A}\}$.*

We define the syntax of secret sharing and necessary conditions as follows:

Definition 2.21 (Secret Sharing (SS)). *Let $P = \{P_1, \ldots, P_N\}$ be a set of parties. A secret sharing scheme SS for a secret space $\mathcal{K}$ and an access structure $\mathbb{A}$ is a tuple of PPT algorithms $\mathsf{SS} = (\mathsf{SS.Share}, \mathsf{SS.Combine})$ defined as follows:*

- $\mathsf{SS.Share}(\mathsf{k} \in \mathcal{K}, \mathbb{A}) \to (\mathsf{s}_1, \ldots, \mathsf{s}_N)$*: On the input of a secret $\mathsf{k} \in \mathcal{K}$ and an access structure $\mathbb{A}$, the sharing algorithm returns a set of shares $\mathsf{s}_1, \ldots, \mathsf{s}_N$ for each party $P_1, \ldots, P_N$.*
- $\mathsf{SS.Combine}(\{\mathsf{s}_i\}_{i\in S}) \to \mathsf{k}$*: On the input of a set of shares $\{\mathsf{s}_i\}_{i\in S}$, the combining algorithm outputs a secret $\mathsf{k} \in \mathcal{K}$.*

Furthermore, SS schemes must satisfy correctness and privacy:

- *Correctness: For all $S \in \mathbb{A}$, $\mathsf{k} \in \mathcal{K}$, $(\mathsf{s}_1, \ldots, \mathsf{s}_N) \leftarrow \mathsf{SS.Share}(\mathsf{k}, \mathbb{A})$, we have $\mathsf{SS.Combine}(\{\mathsf{s}_i\}_{i\in S}) = \mathsf{k}$.*
- *Privacy: For all $S \notin \mathbb{A}$, and $\mathsf{k}_0, \mathsf{k}_1 \in \mathcal{K}$, $(\mathsf{s}_{b,1}, \ldots, \mathsf{s}_{b,N}) \leftarrow \mathsf{SS.Share}(\mathsf{k}_b, \mathbb{A})$ for $b \in \{0, 1\}$, the following distributions are identical: $\{\mathsf{s}_{0,i}\}_{i\in S} \approx \{\mathsf{s}_{1,i}\}_{i\in S}$.*

We describe the construction of linear secret sharing of [9], which we use in our ThPKE schemes, as follows[3]:

[3] Note that we consider only the *binary coefficients* linear secret sharing scheme, while [9] constructs linear secret sharing with general coefficients.

Definition 2.22 (BinLSS: Binary Coefficients Linear Secret Sharing). *Let $P = \{P_1, ..., P_N\}$ be a set of parties and $\mathbb{S}$ be a class of efficient access structures on P. A secret sharing scheme* BinLSS *with secret space $\mathcal{K} = \mathbb{Z}_p$ for some prime p is called a binary coefficients linear secret sharing scheme*[1] *if the following properties are satisfied:*

- $\mathsf{BinLSS.Share}(\mathsf{k} \in \mathbb{Z}_p, \mathbb{A} \in \mathbb{S}) \to (\mathsf{s}_1, \ldots \mathsf{s}_N)$*: There exists a matrix $\mathbf{M} \in \mathbb{Z}_p^{l \times N}$ called the share matrix, and each party P_i is associated with a partition $T_i \subseteq [l]$. To create the shares on a secret* k*, the sharing algorithm first samples random values $r_2, \ldots, r_N \xleftarrow{\$} \mathbb{Z}_p$, and calculate $\mathbf{w} := (w_1, \ldots, w_l)^\intercal$ defined as*

$$\mathbf{w} = \mathbf{M} \cdot (\mathsf{k}, r_2, \ldots, r_N)^\intercal.$$

  Then, outputs $\mathsf{s}_i := \{w_j\}_{j \in T_i}$ as the share for P_i.
- $\mathsf{BinLSS.Combine}(\{\mathsf{s}_i\}_{i \in S})$*: Any valid set of parties $S \in \mathbb{A}$ can efficiently find the binary coefficients $\{c_j \in \{0,1\}\}_{j \in \bigcup_{i \in S} T_i}$ satisfying $\sum_{j \in \bigcup_{i \in S} T_i} c_j \cdot \mathbf{M}[j] = (1, 0, \ldots, 0)$. Thus, the secret is recovered by computing $\mathsf{k} = \sum_{j \in \bigcup_{i \in S} T_i} c_j w_j$.*

We define an analogue of Definition 2.20 for the set of shares as follows:

Definition 2.23 ((In)valid Share Set). *Let $P = \{P_1, \ldots, P_N\}$ be a set of parties, $\mathbb{S}$ a class of efficient access structures on P, and* SS *be a* BinLSS *for $\mathbb{S}$ with share matrix $\mathbf{M} \in \mathbb{Z}_q^{l \times N}$. For a set of indices $T \subseteq [l]$, we say that T is a valid share set if there exist binary coefficients $\{c_j \in \{0,1\}\}_{j \in T}$ satisfying $\sum_{j \in T} c_j \cdot \mathbf{M}[j] = (1, 0, \ldots, 0)$. Otherwise, T is an invalid share set. We also define the following:*

- *Maximal invalid share set:* $\{$*Invalid $T \subseteq [l] \mid \forall i \in [l] \backslash T, (T \cup i)$ is valid* $\}$
- *Minimal valid share set:* $\{$ *Valid $T \subseteq [l] \mid \forall T' \subsetneq T, T$ is invalid* $\}$

The access structure used in threshold cryptosystems is defined as follows:

Definition 2.24 (Threshold Access Structures). *Let $P = \{P_1, \ldots, P_N\}$ be a set of parties. An access structure $\mathbb{A}_{(t,N)}$ is called a (t, N)-threshold access structure if for every set of parties $S \subseteq P$, we have $S \in \mathbb{A}_{(t,N)}$ if $|S| \geq t$.*

The following Theorem 2.25 proves that BinLSS (Definition 2.22) corresponds to any threshold access structure. Therefore, constructing PKE with a decryption access structure following BinLSS is sufficient for constructing ThPKE.

Theorem 2.25 ([9, Theorem 4.15]). *There exists an efficient* BinLSS *for any (t, N)-threshold access structure.*

For ease of understanding the construction of BinLSS, we provide the following simple example:

Example 2.26. *A* BinLSS *for the (N, N)-threshold access structure can be constructed as follows:*

[1] This is called $\{0,1\}$-linear secret sharing scheme in [9]. While [9] defines $\{0,1\}$-LSSS as the class of access structure that is supported by $\{0,1\}$-linear secret sharing scheme, we define BinLSS as $\{0,1\}$-linear secret sharing scheme itself.

- $\mathsf{SS.Share}(\mathsf{k} \in \mathbb{Z}_p) \to (\mathsf{s}_1, \ldots \mathsf{s}_N)$*: Let* $l := N$ *and define the share matrix as:*

$$\mathbf{M} = \begin{pmatrix} 1 & -1 & \cdots & -1 \\ 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix}$$

Sample uniformly random values $r_2, \ldots, r_N \xleftarrow{\$} \mathbb{Z}_p$*, and define a vector* $\mathbf{w} = (w_1, \ldots, w_N)^\intercal$ *as* $\mathbf{w} := \mathbf{M}(\mathsf{k}, r_2, \ldots, r_N)^\intercal$*. Then, we have*

$$\begin{cases} w_1 = \mathsf{k} - \sum_{i=2}^{N} r_i, \ \textit{and} \\ w_i = r_i \quad \textit{for } i = 2, \ldots, N. \end{cases}$$

Let the partition for P_i *be* $T_i = \{i\}$*; output the share* $\mathsf{s}_i := w_i$ *for each* P_i*.*

- $\mathsf{SS.Combine}(\{\mathsf{s}_i\}_{i \in S})$*: The valid party set is* $S = \{P_1, \ldots, P_N\}$*, and the valid share set is* $T = [N] \subseteq \bigcup_{i \in [N]} T_i$*. On input* $\{\mathsf{s}_i\}_{i \in [N]}$*, recover the secret by computing* $\sum_{j \in T} w_j = \sum_{j=1}^{N} w_j = (\mathsf{k} - \sum_{i=2}^{N} r_i) + \sum_{i=2}^{N} r_i = \mathsf{k}$*.*

Secret Sharing Vectors. Although we described only how to share a single scalar in $\mathbb{Z}_p$, we can also share a vector $\mathbf{s} \in \mathbb{Z}_p^n$ by sharing each entry of the vector using independent randomness. It is easy to see that correctness and privacy hold even when we share a vector.

3 Reformulation of Reused-A-LWE

Micciancio and Suhl [35] introduced *the Reused-A-LWE problem*, but the error distribution used in the problem is restricted to the continuous Gaussian distribution. In this section, the error distribution of the Reused-A-LWE problem is generalized (Definition 3.12) and the reduction from LWE with arbitrary error distributions is shown (Theorem 3.15). This reduction is used in Section 4 to prove the security of our ThPKE scheme from LWE with arbitrary error distributions. Furthermore, in Corollary 3.18, we demonstrate that the loss of error parameters in this reduction is smaller than that in [35].

We first define a quasi order between probability distributions in Section 3.1. Then, in Section 3.2, we define the LWE problem [40] and present Lemma 3.11 and Lemma 3.10 by using the quasi order. The lemmas are used to prove the security of our ThPKE scheme in Section 4. Finally, we reformulate the Reused-A-LWE problem in Section 3.3.

3.1 Quasi Order between Distributions

The purpose of this subsection is to provide Definition 3.2, which is used to prove Lemmas 3.10 and 3.11 in Section 3.2. First, we describe the standard definition of *order* as follows:

Definition 3.1 (Order). *Let $\leq$ be a binary relation on a set S, and define 4 conditions as follows:*

1. *Reflexive: For any $a \in S$, $a \leq a$ holds*
2. *Transitive: For any $a, b, c \in S$, if $a \leq b$ and $b \leq c$, then $a \leq c$ holds*
3. *Antisymmetric: For any $a, b \in S$, if $a \leq b$ and $b \leq a$, then $a = b$ holds*
4. *Comparable: For any $a, b \in S$, $a \leq b$ or $b \leq a$ holds*

The binary relation $\leq$ is quasi order *if it satisfies Items 1 and 2,* partial order *if it satisfies Items 1 to 3, and* total order *if it satisfies Items 1 to 4.*

In addition, for any distributions χ_1 and χ_2, we denote by $\chi_1 + \chi_2$ the distribution $\{X_1 + X_2 \mid X_1 \leftarrow \chi_1, X_2 \leftarrow \chi_2,\ X_1$ and X_2 are mutually independent$\}$. Then, we define a binary relation $\leq$ on probability distributions and show that it is a quasi order:

Definition 3.2 (Quasi Order between Distributions). *Let $n \in \mathbb{N}$. For (continuous or discrete) distributions χ_1 and χ_2 over $\mathbb{R}^n$, if there exists a distribution χ_δ such that $(\chi_1 + \chi_\delta) \overset{\mathsf{stat}}{\approx} \chi_2$, we write $\chi_1 \leq \chi_2$.*

Fact 3.3. *The binary relation $\leq$ defined in Definition 3.2 is quasi order but not partial order.*

Proof. We show that "$\leq$" defined in Definition 3.2 is reflexive (Item 1) and transitive (Item 2) but is not antisymmetric (Item 3), as follows:

Reflexive: Let χ_δ be a distribution such that $\Pr_{X \leftarrow \chi_\delta}[X = \mathbf{0}] = 1$, then, for any χ, $\chi \leq \chi + \chi_\delta = \chi$ holds.

Transitive: Let χ_1, χ_2, and χ_3 be distributions such that $\chi_1 \leq \chi_2$ and $\chi_2 \leq \chi_3$ hold; then, there exist χ_{δ_1} and χ_{δ_2} such that $\chi_1 + \chi_{\delta_1} \overset{\mathsf{stat}}{\approx} \chi_2$ and $\chi_2 + \chi_{\delta_2} \overset{\mathsf{stat}}{\approx} \chi_3$ hold. Thus, $\chi_1 \leq \chi_3$ holds because $\chi_1 + \chi_{\delta_1} + \chi_{\delta_2} \overset{\mathsf{stat}}{\approx} \chi_2 + \chi_{\delta_2} \overset{\mathsf{stat}}{\approx} \chi_3$.

Not Antisymmetric: We show that there exist some $\chi_1 \neq \chi_2$ such that $\chi_1 \leq \chi_2$ and $\chi_2 \leq \chi_1$ hold. Let χ_δ and χ'_δ be distributions such that $\chi_\delta \neq \chi'_\delta$ and $\Pr_{X \leftarrow \chi_\delta}[X = \mathbf{0}] = 1 - \mathrm{negl}(\lambda)$, $\Pr_{X \leftarrow \chi'_\delta}[X = \mathbf{0}] = 1 - \mathrm{negl}(\lambda)$. Let χ_1 be an arbitrary distribution and define $\chi_2 := \chi_1 + \chi_\delta$, then $\chi_2 \neq \chi_1$ and $\chi_1 \leq \chi_2$ hold. We also have $\chi_2 \leq \chi_1$ because $\chi_2 + \chi'_\delta = \chi_1 + \chi_\delta + \chi'_\delta \overset{\mathsf{stat}}{\approx} \chi_1$. □

As a typical example, the quasi order between (continuous / discrete) Gaussians is determined by the order of the parameter σ or s in $\mathbb{R}$.

Fact 3.4. *For any $0 < \sigma_1 < \sigma_2$, we have $\mathcal{N}_{\sigma_1} \leq \mathcal{N}_{\sigma_2}$.*

Proof. For $e_1 \leftarrow \mathcal{N}_{\sigma_1}$ and $e_\delta \leftarrow \mathcal{N}_{\sqrt{\sigma_2^2 - \sigma_1^2}}$, we have $(e_1 + e_\delta) \sim \mathcal{N}_{\sigma_2}$. □

Fact 3.5. *Let $\epsilon = \mathrm{negl}(\lambda)$. For any $\tilde{\eta}_\epsilon^+(\mathbb{Z}) < s_1 < s_2$ such that $\tilde{\eta}_\epsilon^+(\mathbb{Z}) < \sqrt{s_2^2 - s_1^2}$, we have $D_{\mathbb{Z},s_1} \leq D_{\mathbb{Z},s_2}$.*

Proof. For $e_1 \leftarrow D_{\mathbb{Z},s_1}$ and $e_\delta \leftarrow D_{\mathbb{Z},\sqrt{s_2^2 - s_1^2}}$, we have $(e_1 + e_\delta) \overset{\mathsf{stat}}{\approx} D_{\mathbb{Z},s_2}$ by Lemma 2.13. □

3.2 Learning with Errors (LWE)

The goal of this subsection is to present Lemmas 3.10 and 3.11, by utilizing the quasi order that we defined in Section 3.1. Interestingly, these lemmas can be applied to any continuous / discrete error distributions, which may be of independent interest. First, we define the LWE distribution and the LWE problem.

Definition 3.6 (LWE). *Let $n \in \mathbb{N}$ be a security parameter and $m = \mathrm{poly}(n)$ and the modulus $q = q(n) \geq 2$ be integers. Let $\mathbb{X}_q$ be $\mathbb{Z}_q$ or $\mathbb{R}_q$, and χ be an error distribution over $\mathbb{X}_q$. The LWE distribution for a fixed secret vector $\mathbf{s} \xleftarrow{\$} \mathbb{Z}_q^n$ is defined as follows:* $\mathsf{LWE}_{\mathbf{s}}(n, m, q, \chi) := \left\{ (\mathbf{A}, \mathbf{A}\mathbf{s} + \mathbf{e}) \mid \mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{m \times n}, \mathbf{e} \leftarrow \chi^m \right\}$. *We omit some arguments when they are clear from the context.*

Definition 3.7 (Decision-LWE). $\mathsf{d\text{-}LWE}_{\mathbf{s}}(n, m, q, \chi)$ *is the problem to distinguish* $\mathsf{LWE}_{\mathbf{s}}(n, m, q, \chi)$ *from the uniform distribution* $\mathcal{U}(\mathbb{Z}_q^{m \times n} \times \mathbb{X}_q^m)$. *The* advantage *of an algorithm* $\mathcal{A} : \mathbb{Z}_q^{m \times n} \times \mathbb{X}_q^m \to \{0, 1\}$ *for solving* d-LWE *is defined as* $\mathrm{Adv}_{\mathcal{A}}^{\mathsf{d\text{-}LWE}} = |\Pr[\mathcal{A}(\mathsf{LWE}_{\mathbf{s}}(n, m, q, \chi)) = 1] - \Pr[\mathcal{A}(\mathcal{U}(\mathbb{Z}_q^{m \times n} \times \mathbb{X}_q^n)) = 1]|$. *We say that* d-LWE *is hard if* $\mathrm{Adv}_{\mathcal{A}}^{\mathsf{d\text{-}LWE}} = \mathrm{negl}(n)$ *for any PPT algorithm* $\mathcal{A}$ *(i.e.,* $\mathsf{LWE}_{\mathbf{s}}(n, m, q, \chi)$ *is pseudorandom).*

Definition 3.8 (Search-LWE). $\mathsf{s\text{-}LWE}_{\mathbf{s}}(n, m, q, \chi)$ *is the problem to find the vector* $\mathbf{s}$ *from a sample* $(\mathbf{A}, \mathbf{b}) \leftarrow \mathsf{LWE}_{\mathbf{s}}(n, m, q, \chi)$. *We say that* s-LWE *is* hard *if the success probability of any PPT algorithm for solving it is* $\mathrm{negl}(n)$.

It was shown in [40] that s-LWE and d-LWE are hard if the error distribution χ is a continuous or discretized Gaussian distribution under the hardness assumption of worst-case lattice problems (e.g., approximate shortest vector problem (GapSVP)). Note that the hardness of s-LWE/d-LWE with a discrete Gaussian distribution (Definition 2.9) is also shown by the LWE self-reduction presented in, e.g., [24,38].

The reduction from d-LWE to s-LWE is trivial, but we provide the proof for completeness.

Fact 3.9 (d-LWE $\leq$ s-LWE). *If there exists a PPT algorithm $\mathcal{A}$ that solves* $\mathsf{s\text{-}LWE}_{\mathbf{s}}(n, m, q, \chi)$, *then there exists a PPT algorithm $\mathcal{A}'$ that solves* $\mathsf{d\text{-}LWE}_{\mathbf{s}}(n, m, q, \chi)$.

Proof. Let $(\mathbf{A}, \mathbf{b})$ be a sample drawn from $\mathsf{LWE}_{\mathbf{s}}(n, m, q, \chi)$ or $\mathcal{U}(\mathbb{Z}_q^{m \times n} \times \mathbb{X}_q^m)$. Using $\mathcal{A}$, we can construct $\mathcal{A}'(\mathbf{A}, \mathbf{b})$ as follows: compute $\bar{\mathbf{s}} \leftarrow \mathcal{A}(\mathbf{A}, \mathbf{b})$ and determine whether the distribution of $\mathbf{b} - \mathbf{A}\bar{\mathbf{s}}$ is χ^m or $\mathcal{U}(\mathbb{X}_q^m)$ using statistical tests. □

The reverse reduction, i.e., s-LWE $\leq$ d-LWE has been shown in, e.g., [30, 40]. The following lemma shows that, if LWE is hard, the probability of the error being extremely small, such as $\mathbf{e} = \mathbf{0}$, is negligible. This lemma is used in Theorem 4.5 to prove the security of our ThPKE scheme.

Lemma 3.10. *If* $\mathsf{d/s\text{-}LWE}_{\mathbf{s}}(n, m, q, \chi)$ *is hard, then, for any* $m \geq n + \omega(\log n)$, *the probability* $P := \Pr_{\mathbf{e} \leftarrow \chi^m}[\forall e \in \mathbf{e}, e \in [0, 1)]$ *is negligible.*

Proof. We prove this by contradiction. First, we show that, if P is nonnegligible, there exists a PPT algorithm that solves s-LWE. Let $(\mathbf{A}, \mathbf{b} = \mathbf{As}+\mathbf{e}) \leftarrow \mathsf{LWE}_{\mathbf{s}}(n, m, q, \chi)$, then, we have $\forall e \in \mathbf{e}$, $e \in [0, 1)$ with probability P. We calculate $\mathbf{b}'$ as follows: When $\mathbb{X}_q = \mathbb{R}_q$, define $\mathbf{b}' := \lfloor \mathbf{b} - 1/2 \cdot \mathbf{1} \rceil$ ($\mathbf{1} := (1, \ldots, 1)^\intercal \in \mathbb{Z}^m$), then we have $\mathbf{b}' = \lfloor \mathbf{As} + \mathbf{e} - 1/2 \cdot \mathbf{1} \rceil = \mathbf{As}$. When $\mathbb{X}_q = \mathbb{Z}_q$, $\forall e \in \mathbf{e}, e \in [0, 1)$ simply means $\mathbf{e} = \mathbf{0}$. Thus, define $\mathbf{b}' := \mathbf{b} = \mathbf{As}$. By Lemma 2.7, with probability of $1-\mathrm{negl}(n)$, there exists a matrix $\mathbf{A}' \in \mathbb{Z}_q^{n \times m}$ such that $\mathbf{A}'\mathbf{A} = \mathbf{I}_n$. Calculate such $\mathbf{A}'$, then output $\mathbf{A}'\mathbf{b}' = \mathbf{s}$. Hence, $\mathsf{d\text{-}LWE}_{\mathbf{s}}(n, m, q, \chi)$ is not hard by Fact 3.9. Because this contradicts the hypothesis, we complete the proof. □

There exists a reduction from LWE with a "small" error distribution to LWE with a "large" error distribution, where "small" and "large" are defined by the quasi order defined in Definition 3.2:

Lemma 3.11. *If* $\mathsf{d/s\text{-}LWE}_{\mathbf{s}}(n, m, q, \chi_1)$ *is hard, then for any* $\chi_2 \geq \chi_1$, $\mathsf{d/s\text{-}LWE}_{\mathbf{s}}(n, m, q, \chi_2)$ *is also hard.*

Proof. By the definition of $\chi_2 \geq \chi_1$, there exists a distribution χ_δ such that $\chi_2 \overset{\mathsf{stat}}{\approx} \chi_1 + \chi_\delta$. Let $(\mathbf{A}, \mathbf{b}) \leftarrow \mathsf{LWE}_{\mathbf{s}}(n, m, q, \chi_1)$, then sample $\mathbf{e}' \leftarrow \chi_\delta^m$ and define $\mathbf{b}' := \mathbf{b} + \mathbf{e}'$. Then, $(\mathbf{A}, \mathbf{b}') \overset{\mathsf{stat}}{\approx} \mathsf{LWE}_{\mathbf{s}}(n, m, q, \chi_2)$ holds. Note that $\mathbf{b}' := \mathbf{b} + \mathbf{e}' \sim \mathcal{U}(\mathbb{X}_q^m)$ when $(\mathbf{A}, \mathbf{b}) \leftarrow \mathcal{U}(\mathbb{Z}_q^{m \times n} \times \mathbb{X}_q^m)$. If $\mathsf{d/s\text{-}LWE}_{\mathbf{s}}(n, m, q, \chi_2)$ is not hard, $\mathsf{d/s\text{-}LWE}_{\mathbf{s}}(n, m, q, \chi_1)$ is also not hard by the transformation described above. This contradicts the hypothesis; thus, we complete the proof. □

3.3 Reused-A-LWE

We reformulate the Reused-A-LWE problem [35] in Definition 3.12. Then, we show a reduction from LWE to Reused-A-LWE for any error distributions (Theorem 3.15). Furthermore, we show in Corollary 3.18 that this reduction incurs a smaller loss in the error parameter than the reduction shown in [35].

Definition 3.12 (Reused-A-LWE **(generalized from [10, Definition 5])).** *Let* n, m, $q \in \mathbb{N}$. *Let* $\mathbb{X}_q$ *be either* $\mathbb{Z}_q$ *or* $\mathbb{R}_q$, *and let* χ_1 *and* χ_2 *be distributions on* $\mathbb{X}_q$. *For a fixed* $\mathbf{s} \xleftarrow{\$} \mathbb{Z}_q^n$, *we define the Reused-A-LWE distribution* $\mathsf{Reused\text{-}A\text{-}LWE}_{\mathbf{s}}(n, m, q, \chi_1, \chi_2)$ *as follows:*

$$\left\{ (\mathbf{A}, \mathbf{b}_1 := \mathbf{As} + \mathbf{e}_1, \mathbf{b}_2 := \mathbf{As} + \mathbf{e}_2) \;\middle|\; \begin{array}{l} \mathbf{A} \sim \mathcal{U}(\mathbb{Z}_q^{m \times n}), \mathbf{e}_1 \sim \chi_1^m, \mathbf{e}_2 \sim \chi_2^m \\ \mathbf{b}_1 - \mathbf{b}_2 = \mathbf{e}_1 - \mathbf{e}_2 \end{array} \right\}$$

Definition 3.13 (d-Reused-A-LWE)**.** *The* $\mathsf{d\text{-}Reused\text{-}A\text{-}LWE}(n, m, q, \chi_1, \chi_2)$ *is a problem to distinguish* $\mathsf{Reused\text{-}A\text{-}LWE}_{\mathbf{s}}(n, m, q, \chi_1, \chi_2)$ *from the following distribution:*

$$\mathcal{V} := \left\{ (\mathbf{A}, \mathbf{u}, \mathbf{v}) \;\middle|\; \begin{array}{l} \mathbf{A} \sim \mathcal{U}(\mathbb{Z}_q^{m \times n}), \mathbf{u}, \mathbf{v} \sim \mathcal{U}(\mathbb{X}_q^m), \mathbf{e}_1 \sim \chi_1^m, \mathbf{e}_2 \sim \chi_2^m \\ \mathbf{u} - \mathbf{v} = \mathbf{e}_1 - \mathbf{e}_2 \end{array} \right\}$$

The advantage *of an algorithm* $\mathcal{A} : \mathbb{Z}_q^{m \times n} \times \mathbb{X}_q^m \times \mathbb{X}_q^m \to \{0, 1\}$ *for solving* $\mathsf{d\text{-}Reused\text{-}A\text{-}LWE}$ *is defined as* $\mathrm{Adv}_{\mathcal{A}}^{\mathsf{d\text{-}Reused\text{-}A\text{-}LWE}} = |\Pr[\mathcal{A}(\mathsf{Reused\text{-}A\text{-}LWE}_{\mathbf{s}}(n, m,$

$q, \chi_1, \chi_2)) = 1] - \Pr[\mathcal{A}(\mathcal{V}) = 1]|$. *We say that* d-Reused-A-LWE *is hard if* $\mathrm{Adv}_{\mathcal{A}}^{\text{d-Reused-A-LWE}} = \mathrm{negl}(n)$ *for any PPT algorithm* $\mathcal{A}$.

Definition 3.14 (s-Reused-A-LWE). s-Reused-A-LWE$(n, m, q, \chi_1, \chi_2)$ *is a problem to find* $\mathbf{s}$ *given a sample from* Reused-A-LWE$_{\mathbf{s}}(n, m, q, \chi_1, \chi_2)$. *We say that* s-Reused-A-LWE *is* hard *if the success probability of any PPT algorithm for solving it is* $\mathrm{negl}(n)$.

We show a reduction from d-LWE to d-Reused-A-LWE for arbitrary error distributions χ_1, χ_2 in the following Theorem 3.15. Note that the counterpart of this theorem, [35, Corollary 3], is limited to the continuous Gaussian distribution.

Theorem 3.15 (d-LWE $\leq$ d-Reused-A-LWE). *If both* d-LWE(n, m, q, χ_1) *and* d-LWE(n, m, q, χ_2) *are hard; then* d-Reused-A-LWE$(n, m, q, \chi_1, \chi_2)$ *is also hard.*

Proof. The proof is performed using a straightforward hybrid argument. For simplicity of notation, let $\mathcal{X}_1 := \mathsf{LWE}(n, m, q, \chi_1)$, $\mathcal{X}_2 := \mathsf{LWE}(n, m, q, \chi_2)$, and $\mathcal{X}_3 := \text{Reused-A-LWE}(n, m, q, \chi_1, \chi_2)$. For each $\mathcal{X}_i$ $(i \in \{1, 2, 3\})$, we denote by $\mathrm{Adv}_{\mathcal{A}}^{\mathcal{X}_i}$ the advantage of an algorithm $\mathcal{A}$ for solving the decision problem of $\mathcal{X}_i$, i.e., d-$\mathcal{X}_i$. We also define the following hybrid distribution $\mathcal{H}$:

$$\mathcal{H} := \left\{ (\mathbf{A}, \mathbf{u}, \mathbf{b}_2) \;\middle|\; \begin{array}{l} \mathbf{A} \sim \mathcal{U}(\mathbb{Z}_q^{m \times n}), \mathbf{u} \sim \mathcal{U}(\mathbb{X}_q^m), \mathbf{e}_1 \sim \chi_1^m, \mathbf{e}_2 \sim \chi_2^m \\ \mathbf{u} - \mathbf{b}_2 = \mathbf{e}_1 - \mathbf{e}_2 \end{array} \right\}$$

Then, we obtain

$$|\Pr[\mathcal{A}(\mathcal{X}_3) = 1] - \Pr[\mathcal{A}(\mathcal{H}) = 1]| = \mathrm{Adv}_{\mathcal{A}}^{\mathcal{X}_1}, \text{ and}$$
$$|\Pr[\mathcal{A}(\mathcal{H}) = 1] - \Pr[\mathcal{A}(\mathcal{V}) = 1]| = \mathrm{Adv}_{\mathcal{A}}^{\mathcal{X}_2},$$

where $\mathcal{V}$ is defined as in Definition 3.12. Therefore, we have

$$\mathrm{Adv}_{\mathcal{A}}^{\mathcal{X}_3} = |\Pr[\mathcal{A}(\mathcal{X}_3) = 1] - \Pr[\mathcal{A}(\mathcal{V}) = 1]| \leq \mathrm{Adv}_{\mathcal{A}}^{\mathcal{X}_1} + \mathrm{Adv}_{\mathcal{A}}^{\mathcal{X}_2}. \tag{2}$$

Because $\mathrm{Adv}_{\mathcal{A}}^{\mathcal{X}_1}, \mathrm{Adv}_{\mathcal{A}}^{\mathcal{X}_2} = \mathrm{negl}(n)$ by hypothesis, we complete the proof. □

The above theorem means that if the d-LWE problems of $(\mathbf{A}, \mathbf{b}_1)$ and $(\mathbf{A}, \mathbf{b}_2)$ are both hard, then it is computationally hard to obtain any information other than that $\mathbf{b}_2 - \mathbf{b}_1 = \mathbf{e}_2 - \mathbf{e}_1$ holds even if the Reused-A-LWE sample $(\mathbf{A}, \mathbf{b}_1, \mathbf{b}_2)$ is given.

Furthermore, if $\chi_1 \leq \chi_2$ (Definition 3.2) holds, we can show that Theorem 3.15 is a tight reduction:

Corollary 3.16. *Let* χ_1 *and* χ_2 *be distributions such that* $\chi_1 \leq \chi_2$. *If there exists an algorithm that solves* d-Reused-A-LWE$(n, m, q, \chi_1, \chi_2)$ *with advantage* ϵ, *then there exists an algorithm that solves* d-LWE(n, m, q, χ_1) *with advantage of at least* $\epsilon/2$.

Proof. Define $\mathrm{Adv}_{\mathcal{A}}^{\mathcal{X}_i}$ as in the proof of Theorem 3.15. By Lemma 3.11 and $\chi_1 \leq \chi_2$, we have $\mathrm{Adv}_{\mathcal{A}}^{\mathcal{X}_1} \geq \mathrm{Adv}_{\mathcal{A}}^{\mathcal{X}_2}$ for any algorithm $\mathcal{A}$. Hence, by Eq. (2), we obtain $\epsilon = \mathrm{Adv}_{\mathcal{A}}^{\mathcal{X}_3} \leq \mathrm{Adv}_{\mathcal{A}}^{\mathcal{X}_1} + \mathrm{Adv}_{\mathcal{A}}^{\mathcal{X}_2} \leq 2\,\mathrm{Adv}_{\mathcal{A}}^{\mathcal{X}_1}$. □

We also show a reduction from s-LWE to s-Reused-A-LWE as follows (although we do not require this for our construction of ThPKE):

Theorem 3.17 (s-LWE $\leq$ s-Reused-A-LWE). *Let χ_1 and χ_2 be distributions such that $\chi_1 \leq \chi_2$ holds (where $\leq$ is the quasi order defined in Definition 3.2). If* s-LWE(n, m, q, χ_1) *is hard, then* s-Reused-A-LWE$(n, m, q, \chi_1, \chi_2)$ *is also hard.*

Proof. Because $\chi_1 \leq \chi_2$ by hypothesis, there exists χ_δ such that $\chi_1 + \chi_\delta \overset{\mathsf{stat}}{\approx} \chi_2$. Given a sample $(\mathbf{A}, \mathbf{b}_1) \leftarrow \mathsf{LWE}(n, m, q, \chi_1)$, define $\mathbf{b}_2 := \mathbf{b}_1 + \mathbf{e}_\delta$, where $\mathbf{e}_\delta \leftarrow \chi_\delta^m$. Then, we have $(\mathbf{A}, \mathbf{b}_2) \overset{\mathsf{stat}}{\approx} \mathsf{LWE}(n, m, q, \chi_2)$; thus, $(\mathbf{A}, \mathbf{b}_1, \mathbf{b}_2) \overset{\mathsf{stat}}{\approx}$ Reused-A-LWE$(n, m, q, \chi_1, \chi_2)$. Therefore, if there exists a PPT algorithm that solves s-Reused-A-LWE$(n, m, q, \chi_1, \chi_2)$, there exists a PPT algorithm that solves s-LWE(n, m, q, χ_1), □

Note that the above theorem requires the relation $\chi_1 \leq \chi_2$, while Theorem 3.15 does not require the relation. Additionally, note that $\chi_1 \leq \chi_2$ does not necessarily hold for any pair of distributions χ_1 and χ_2, i.e., $\leq$ defined in Definition 3.2 is a quasi order and does not satisfy Item 4 in Definition 3.1.

Finally, we demonstrate that Theorem 3.15 and Theorem 3.17 instantiated with the continuous Gaussian distribution subsume and slightly improve [35, Corollary 3]:

Corollary 3.18. *Let $0 < \sigma_1, \sigma_2$ and define $\sigma_{\min} := \min(\sigma_1, \sigma_2)$. If* d-LWE$(n, m, q, \mathcal{N}_{\sigma_{\min}})$ *is hard, then* d-Reused-A-LWE$(n, m, q, \mathcal{N}_{\sigma_1}, \mathcal{N}_{\sigma_2})$ *is also hard. Similarly, if* s-LWE$(n, m, q, \mathcal{N}_{\sigma_{\min}})$ *is hard, then* s-Reused-A-LWE$(n, m, q, \mathcal{N}_{\sigma_1}, \mathcal{N}_{\sigma_2})$ *is also hard.*

Proof. Because $\mathcal{N}_{\sigma_{\min}} \leq \mathcal{N}_{\sigma_1}, \mathcal{N}_{\sigma_2}$ holds by Definition 3.2, we obtain the reduction from d-LWE$(n, m, q, \mathcal{N}_{\sigma_{\min}})$ to d-Reused-A-LWE$(n, m, q, \mathcal{N}_{\sigma_1}, \mathcal{N}_{\sigma_2})$ by Lemma 3.11 and Theorem 3.15. The reduction from s-LWE$(n, m, q, \mathcal{N}_{\sigma_{\min}})$ to s-Reused-A-LWE$(n, m, q, \mathcal{N}_{\sigma_1}, \mathcal{N}_{\sigma_2})$ follows from Theorem 3.17. □

[35, Corollary 3] requires the hardness assumption of d/s-LWE$(n, m, q, \mathcal{N}_{\sigma_b})$, where $\sigma_b = (\sigma_1^{-2} + \sigma_2^{-2})^{-1/2}$, for showing the hardness of d/s-Reused-A-LWE$(n, m, q, \mathcal{N}_{\sigma_1}, \mathcal{N}_{\sigma_2})$. In contrast, our Corollary 3.18 only requires the hardness of d/s-LWE$(n, m, q, \mathcal{N}_{\sigma_{\min}})$, which is a slightly weaker assumption than d/s-LWE$(n, m, q, \mathcal{N}_{\sigma_b})$ because $\sigma_b \lneq \min(\sigma_1, \sigma_2)$.

We can also show the counterpart of Corollary 3.18 for discrete Gaussians.

Corollary 3.19. *Let $\epsilon = \mathrm{negl}(\lambda)$, $\tilde{\eta}_\epsilon^+(\mathbb{Z}) < s_1 < s_2$ and $\tilde{\eta}_\epsilon^+(\mathbb{Z}) < \sqrt{s_2^2 - s_1^2}$. If* d-LWE$(n, m, q, D_{\mathbb{Z}, s_1})$ *is hard, then* d-Reused-A-LWE$(n, m, q, D_{\mathbb{Z}, s_1}, D_{\mathbb{Z}, s_2})$ *is also hard. Similarly, if* s-LWE$(n, m, q, D_{\mathbb{Z}, s_1})$ *is hard, then* s-Reused-A-LWE$(n, m, q, D_{\mathbb{Z}, s_1}, D_{\mathbb{Z}, s_2})$ *is also hard.*

Proof. We have $D_{\mathbb{Z}, s_1} \leq D_{\mathbb{Z}, s_2}$ by Fact 3.5. Hence, the proof is identical to that of Corollary 3.18. □

Algorithm 1: Our LWE-based ThPKE := (Params, KeyGen, Setup, Enc, PartDec, FinDec)

$\underline{\mathsf{Params}(1^\lambda, 1^N) \to \mathsf{pp}}$:

1 Choose public parameters $\mathsf{pp} := (n, m, q, \chi_{\mathsf{pk}}, \chi_{\mathsf{err}}, \chi_{\mathsf{enc}}, \chi_{\mathsf{sm}})$.
Note: The following functions implicitly take pp as an argument.

$\underline{\mathsf{KeyGen}() \to (\mathsf{pk}, \mathsf{sk}, \mathsf{err}, \chi_{\mathsf{Sim}})}$:

2 $\mathsf{sk} := \mathbf{s} \xleftarrow{\$} \mathbb{Z}_q^n$, $\mathsf{pk} := (\mathbf{A}, \mathbf{b} := \mathbf{As} + \mathbf{e}) \leftarrow \mathsf{LWE}_{\mathbf{s}}(n, m, q, \chi_{\mathsf{pk}})$
3 $\mathsf{err} := \boldsymbol{\zeta} \leftarrow \chi_{\mathsf{err}}^m$
4 Define a distribution $\chi_{\mathsf{Sim}}(\mathbf{e}, \boldsymbol{\zeta}, \chi_{\mathsf{sm}}, \chi_{\mathsf{enc}})$ as follows:

$$\chi_{\mathsf{Sim}}(\mathbf{e}, \boldsymbol{\zeta}) := \{e^{\mathsf{sm}} + \mathbf{r}_1^\intercal \boldsymbol{\zeta} - \mathbf{r}_2^\intercal \mathbf{e} \mid e^{\mathsf{sm}} \sim \chi_{\mathsf{sm}}, \mathbf{r}_1, \mathbf{r}_2 \overset{\text{iid}}{\sim} \chi_{\mathsf{enc}}^m\} \tag{3}$$

Example: $\chi_{\mathsf{Sim}} \overset{\text{stat}}{\approx} D_{\mathbb{Z}, \sqrt{s_{\mathsf{sm}}^2 + s_{\mathsf{enc}}^2(\|\mathbf{e}\|^2 + \|\boldsymbol{\zeta}\|^2)}}$ if $\chi_{\mathsf{enc}} := D_{\mathbb{Z}, s_{\mathsf{enc}}}, \chi_{\mathsf{sm}} := D_{\mathbb{Z}, s_{\mathsf{sm}}}$.

$\underline{\mathsf{Setup}(\mathsf{sk}, \mathsf{err}, \mathbb{A}) \to (\mathsf{sk}_1, \ldots, \mathsf{sk}_N, \mathsf{err}_1, \ldots, \mathsf{err}_N)}$:

5 $\mathsf{BinLSS.Share}((\mathbf{s}, \boldsymbol{\zeta}), \mathbb{A}) \to \{(\mathsf{sk}_i, \mathsf{err}_i) := \{(\mathbf{s}_j, \boldsymbol{\zeta}_j)\}_{j \in T_i}\}_{i \in [N]}$ (Definition 2.22)

$\underline{\mathsf{Enc}(\mathsf{pk}, \mu \in \{0,1\}) \to \mathsf{ct}}$:

6 Define $\mathsf{msg} := \lfloor \frac{q}{2} \rfloor \cdot \mu$, and sample $\mathbf{r}, \mathbf{r}_{\mathsf{aux}} \leftarrow \chi_{\mathsf{enc}}^m$
7 Define $(\mathbf{a}', b') := (\mathbf{r}^\intercal \mathbf{A}, \mathbf{r}^\intercal \mathbf{b} + \mathsf{msg})$ and output a ciphertext $\mathsf{ct} := (\mathbf{a}', b', \mathbf{r}_{\mathsf{aux}})$
Note: $\mathbf{r}_{\mathsf{aux}}$ is auxiliary information which will be used in PartDec.

$\underline{\mathsf{PartDec}(\mathsf{ct}, \mathsf{sk}_i, \mathsf{err}_i) \to \mathsf{pd}_i}$:

8 Parse $\mathsf{sk}_i = \{\mathbf{s}_j\}_{j \in T_i}$ and $\mathsf{err}_i = \{\boldsymbol{\zeta}_j\}_{j \in T_i}$
9 **for** $j \in T_i$ **do** Sample $e_j^{\mathsf{sm}} \leftarrow \chi_{\mathsf{sm}}$, and define $\mathbf{p}_j := (\mathbf{a}')^\intercal \mathbf{s}_j + e_j^{\mathsf{sm}} + \mathbf{r}_{\mathsf{aux}}^\intercal \boldsymbol{\zeta}_j$
10 Output a partial decryption $\mathsf{pd}_i := \{\mathbf{p}_j\}_{j \in T_i}$

$\underline{\mathsf{FinDec}(\{\mathsf{pd}_i\}_{i \in S}) \to \overline{\mu} \in \{0,1\} \text{ or } \perp}$:

11 **if** $S \notin \mathbb{A}$ **then** Output $\perp$ and **break**
12 Otherwise, parse $\{\mathsf{pd}_i\}_{i \in S} = \{\{\mathbf{p}_j\}_{j \in T_i}\}_{i \in S}$
13 Calculate a minimal valid share set $T \subseteq \bigcup_{i \in S} T_i$ (Definition 2.23)
14 Output $\overline{\mu} := \lfloor (b' - \sum_{i \in T} \mathbf{p}_i) / \lfloor \frac{q}{2} \rfloor \rceil$

4 Simulation-Secure ThPKE from LWE

We describe the construction of our scheme in Section 4.1. Then, we define and prove the correctness and security in Section 4.2 and Section 4.3, respectively. Finally, we provide an instantiation that simultaneously satisfies correctness and security in Section 4.4.

4.1 Construction

Our ThPKE scheme is presented in Algorithm 1. This scheme is constructed based on the ThPKE of [35] instantiated with the Regev-like PKE [40]. We modify the scheme by distributing shares $(\mathsf{err}_1, \ldots, \mathsf{err}_N)$ of a small error $\mathsf{err} := \boldsymbol{\zeta} \leftarrow \chi_{\mathsf{err}}^m$ for "masking" the partial decryption to the parties with secret sharing, in addition to the shares $(\mathsf{sk}_1, \ldots, \mathsf{sk}_N)$ of the secret key $\mathsf{sk} := \mathbf{s}$. Then, we add a

randomized value of err_i (specifically, $\mathbf{r}_{\mathsf{aux}}^\intercal \boldsymbol{\zeta}_j$) in the partial decryption ($\mathsf{PartDec}$, Line 9), in addition to the conventional "smudging noise" ($e^{\mathsf{sm}} \sim \chi_{\mathsf{sm}}$).

Furthermore, we generalize the error distribution χ_{pk} of the public key and the distribution χ_{enc} for encryption to arbitrary (continuous/discrete) distributions. In addition, our construction supports any access structure $\mathbb{A}$ that can be constructed using BinLSS as defined in Definition 2.22, which includes any threshold access structures, as shown in Theorem 2.25.

4.2 Correctness

We define the correctness of ThPKE in Definition 4.1, and show that our scheme is correct in Lemma 4.2.

Definition 4.1 (Correctness). *We say that the ThPKE scheme defined in Algorithm 1 is* correct *if* $\mathsf{FinDec}(\{\mathsf{pd}_i\}_{i\in S}) = \mu$ *holds with overwhelming probability for any* $S \in \mathbb{A}$ *for an overwhelming proportion of* $(\mathsf{pk}, \mathsf{sk}, \mathsf{err})$ *generated by* $\mathsf{KeyGen}()$.

As preparation, we define a distribution $\chi_{\mathsf{Sim},t}(\mathbf{e}, \boldsymbol{\zeta})$ with a parameter $t \in \mathbb{N}$ ($t \leq N$), which is a generalization of χ_{Sim} that is defined in Eq. (3):

$$\chi_{\mathsf{Sim},t}(\mathbf{e}, \boldsymbol{\zeta}) := \left\{ \sum_{i=1}^{t} e_i^{\mathsf{sm}} + \mathbf{r}_1^\intercal \boldsymbol{\zeta} - \mathbf{r}_2^\intercal \mathbf{e} \;\middle|\; e_1^{\mathsf{sm}}, \ldots, e_t^{\mathsf{sm}} \overset{\text{iid}}{\sim} \chi_{\mathsf{sm}}, \mathbf{r}_1, \mathbf{r}_2 \overset{\text{iid}}{\sim} \chi_{\mathsf{enc}}^m \right\} \quad (4)$$

Then, we derive the sufficient condition for Algorithm 1 to be correct:

Lemma 4.2. *The ThPKE scheme defined in Algorithm 1 is correct if we have* $\Pr_{x \leftarrow \chi_{\mathsf{Sim},t}}\left[|x| < \lfloor \frac{q}{4} \rfloor\right] = 1 - \mathrm{negl}(\lambda)$ *for an overwhelming proportion of* $(\mathsf{pk} := (\mathbf{A}, \mathbf{As}+\mathbf{e}), \mathsf{sk} := \mathbf{s}, \mathsf{err} := \boldsymbol{\zeta})$ *generated by* $\mathsf{KeyGen}()$, *where* $\chi_{\mathsf{Sim},t} := \chi_{\mathsf{Sim},t}(\mathbf{e}, \boldsymbol{\zeta})$ *is defined as in* (4) *and* $t = |T| (\leq N)$.

Proof. At Line 14 in Algorithm 1, we have:

$$\begin{aligned} b' - \textstyle\sum_{i\in T} \mathbf{p}_i &= b' - (\mathbf{a}')^\intercal \mathbf{s} - \textstyle\sum_{i\in T} e_i^{\mathsf{sm}} - \mathbf{r}_{\mathsf{aux}}^\intercal \boldsymbol{\zeta} \\ &= \mathsf{msg} + \mathbf{r}^\intercal \mathbf{e} - \textstyle\sum_{i\in T} e_i^{\mathsf{sm}} - \mathbf{r}_{\mathsf{aux}}^\intercal \boldsymbol{\zeta} \end{aligned} \quad (5)$$

By hypothesis, $|\mathbf{r}^\intercal \mathbf{e} - \sum_{i\in T} e_i^{\mathsf{sm}} - \mathbf{r}_{\mathsf{aux}}^\intercal \boldsymbol{\zeta}| < \lfloor \frac{q}{4} \rfloor$ holds with overwhelming probability. Thus, $\overline{\mu} := \lfloor (b' - \sum_{i\in T} \mathbf{p}_i) / \lfloor \frac{q}{2} \rfloor \rceil = \mu + \lfloor (\mathbf{r}^\intercal \mathbf{e} - \sum_{i\in T} e_i^{\mathsf{sm}} - \mathbf{r}_{\mathsf{aux}}^\intercal \boldsymbol{\zeta}) / \lfloor \frac{q}{2} \rfloor \rceil = \mu$ also holds with overwhelming probability. □

For reference, we provide a typical example of parameter setting that satisfies the correctness.

Example 4.3. *Let* χ_{pk} *and* χ_{err} *be the* B_{pk}*-bounded and* B_{err}*-bounded (Definition 2.8) distribution over* $\mathbb{Z}_q$, *respectively. Let* $\chi_{\mathsf{enc}} := D_{\mathbb{Z},s_{\mathsf{enc}}}$ *and* $\chi_{\mathsf{sm}} := D_{\mathbb{Z},s_{\mathsf{sm}}}$ *for* $s_{\mathsf{enc}} \geq \max(B_{\mathsf{pk}}, B_{\mathsf{err}})\tilde{\eta}_\epsilon^+(\mathbb{Z})$ *and* $s_{\mathsf{sm}} \geq \tilde{\eta}_\epsilon^+(\mathbb{Z})$. *The ThPKE scheme defined in Algorithm 1 is correct for any* N, $m = \mathrm{poly}(n)$, q, B_{pk}, B_{err}, s_{enc} *and* s_{sm} *such that* $\sqrt{\lambda(Ns_{\mathsf{sm}}^2 + B^2 s_{\mathsf{enc}}^2)} < \lfloor \frac{q}{4} \rfloor$, *where* $B := B(m) := \sqrt{m(B_{\mathsf{pk}}^2 + B_{\mathsf{err}}^2)}$.

Proof. Define $\chi_{\mathsf{Sim},|T|}$ as in (4). Because χ_{pk} and χ_{err} are bounded by B_{pk} and B_{err}, respectively, and $m = \mathrm{poly}(n)$, we have

$$\|\mathbf{e}\| < B_{\mathsf{pk}}\sqrt{m} \quad \text{and} \quad \|\boldsymbol{\zeta}\| < B_{\mathsf{err}}\sqrt{m} \tag{6}$$

with overwhelming probability over the choice of $\mathsf{pk} := (\mathbf{A}, \mathbf{A}\mathbf{s}+\mathbf{e})$ and $\mathsf{err} := \boldsymbol{\zeta}$. By Lemma 2.13, $\chi_{\mathsf{Sim},|T|} \overset{\mathsf{stat}}{\approx} D_{\mathbb{Z},\sqrt{|T|s_{\mathsf{sm}}^2 + s_{\mathsf{enc}}^2(\|\mathbf{e}\|^2+\|\boldsymbol{\zeta}\|^2)}}$ holds. Furthermore, by Eq. (6) and Fact 3.5 and $|T| \le N$, $\chi_{\mathsf{Sim},|T|} \le D_{\mathbb{Z},\sqrt{Ns_{\mathsf{sm}}^2 + B^2 s_{\mathsf{enc}}^2}}$ holds, where $\le$ is the quasi order defined in Definition 3.2. By the tail bound of the Gaussian distribution, i.e., Lemma 2.18, $\Pr_{x\leftarrow D_{\mathbb{Z},\sqrt{Ns_{\mathsf{sm}}^2 + B^2 s_{\mathsf{enc}}^2}}}[x < \sqrt{\lambda(Ns_{\mathsf{sm}}^2 + B^2 s_{\mathsf{enc}}^2)}] = 1 - \mathrm{negl}(\lambda)$. Thus, we also have $\Pr_{x\leftarrow\chi_{\mathsf{Sim},|T|}}[x < \sqrt{\lambda(Ns_{\mathsf{sm}}^2 + B^2 s_{\mathsf{enc}}^2)}] = 1 - \mathrm{negl}(\lambda)$, thereby we complete the proof. □

4.3 Simulation Security

We define the simulation security of our ThPKE scheme:

Definition 4.4 (Simulation Security (SS)). *We say that the ThPKE scheme is simulation secure if, for any PPT distinguisher $\mathcal{D}$, for any stateful*[4] *PPT algorithm $\mathcal{A} := (\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3)$, there exists a PPT algorithm* Sim *such that*

$$\mathrm{Adv}^{\mathsf{SS\text{-}ThPKE}}_{\mathcal{D},\mathcal{A},\mathsf{Sim}}(1^\lambda) := \left| \begin{array}{c} \Pr[\mathcal{D}(\mathsf{Expt}_{\mathcal{A},\mathsf{Real}}(1^\lambda)) = 1] \\ -\Pr[\mathcal{D}(\mathsf{Expt}_{\mathcal{A},\mathsf{Sim},\mathsf{Ideal}}(1^\lambda)) = 1] \end{array} \right| = \mathrm{negl}(\lambda), \tag{7}$$

where the experiments $\mathsf{Expt}_{\mathcal{A},\mathsf{Real}}(1^\lambda)$ and $\mathsf{Expt}_{\mathcal{A},\mathsf{Sim},\mathsf{Ideal}}(1^\lambda)$ are defined as in Algorithm 2. Additionally, for fixed outputs of Line 1-3 of Algorithm 2, the adversary can repeat Line 4 and subsequent steps for arbitrary $\mathrm{poly}(\lambda)$ times.

Here, we prove the security or our ThPKE scheme under a strong assumption that $\chi_{\mathsf{Sim}}(\mathbf{e}, \boldsymbol{\zeta})$ defined in Eq. (3) does not leak any information about the fixed $\mathbf{e}$ and $\boldsymbol{\zeta}$ generated by KeyGen. This assumption is removed with the instantiation described in subsequent Theorem 4.6.

Theorem 4.5. *Let $m \ge n\log q + 2\lambda$, and assume that $\mathsf{d\text{-}LWE}(n, m, q, \chi_{\mathsf{pk}})$, $\mathsf{d\text{-}LWE}(n, m, q, \chi_{\mathsf{enc}})$, and $\mathsf{d\text{-}LWE}(n, m, q, \chi_{\mathsf{sm}})$ are all hard. In addition, assume that it is hard to obtain any information about $\mathbf{e}$ and $\boldsymbol{\zeta}$ (other than that $\mathbf{e} \sim \chi_{\mathsf{pk}}$ and $\boldsymbol{\zeta} \sim \chi_{\mathsf{err}}$) from (the probability function of) $\chi_{\mathsf{Sim}}(\mathbf{e}, \boldsymbol{\zeta})$ defined in Eq. (3), where $\mathbf{e} \leftarrow \chi_{\mathsf{pk}}$ and $\boldsymbol{\zeta} \leftarrow \chi_{\mathsf{err}}$. Then, Algorithm 1 satisfies SS (Definition 4.4).*

Proof. We show that, for any PPT distinguisher $\mathcal{D}$, for any stateful PPT algorithm $\mathcal{A}$, there exists some PPT algorithm Sim such that Eq. (7) holds. In addition to $\mathsf{Expt}_{\mathcal{A},\mathsf{Real}}$ and $\mathsf{Expt}_{\mathcal{A},\mathsf{Sim},\mathsf{Ideal}}$, we also define an intermediate hybrid experiment $\mathsf{Expt}_{\mathcal{A},\mathsf{Sim},\mathsf{Hybrid}}(1^\lambda) \to \{0,1\}$ as described in Algorithm 2. Then, to

[4] means that $\mathcal{A}_i$ inherits the inputs, outputs, and internal state of $\mathcal{A}_1, \ldots, \mathcal{A}_{i-1}$ for $i = 2, 3$.

Algorithm 2: Experiments ($\mathsf{Expt}_{\mathcal{A},\mathsf{Real}}$ and $\mathsf{Expt}_{\mathcal{A},\mathsf{Sim},\mathsf{Ideal}}$) that define the simulation security (Definition 4.4) of our ThPKE and the hybrid experiment ($\mathsf{Expt}_{\mathcal{A},\mathsf{Sim},\mathsf{Hybrid}}$). Note that $\mathcal{A} := (\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3)$ is a stateful algorithm although we omit the writing state in the the inputs/outputs.

$\underline{\mathsf{Expt}_{\mathcal{A},\mathsf{Real}}(1^\lambda)}$:

1 $\mathsf{pp} \leftarrow \mathsf{Params}(1^\lambda, 1^N)$, $(\mathsf{sk}, \mathsf{err}, \mathsf{pk}, \chi_{\mathsf{Sim}}) \leftarrow \mathsf{KeyGen}()$
2 $(\mathsf{sk}_1, \ldots, \mathsf{sk}_N, \mathsf{err}_1, \ldots, \mathsf{err}_N) \leftarrow \mathsf{Setup}(\mathsf{sk}, \mathsf{err}, \mathbb{A})$
3 Plaintext $\mu \in \{0,1\}$ and a maximal invalid party set (corrupted by $\mathcal{A}$) $S_{\mathsf{mal}} \subsetneq [N] \leftarrow \mathcal{A}_1(\mathsf{pp}, \mathsf{pk}, \chi_{\mathsf{Sim}})$
4 $\mathsf{ct} \leftarrow \mathsf{Enc}(\mathsf{pk}, \mu)$
5 A (valid) party set $S \subseteq [N] \leftarrow \mathcal{A}_2(\{\mathsf{sk}_i, \mathsf{err}_i\}_{i \in S_{\mathsf{mal}}}, \mathsf{ct})$
6 $\{\mathsf{pd}_i\}_{i \in S} \leftarrow \{\mathsf{PartDec}(\mathsf{pk}, \mathsf{ct}, \mathsf{sk}_i, \mathsf{err}_i)\}_{i \in S}$
7 **return** $\{0,1\} \leftarrow \mathcal{A}_3(\{\mathsf{pd}_i\}_{i \in S})$

$\underline{\mathsf{Expt}_{\mathcal{A},\mathsf{Sim},\mathsf{Hybrid}}(1^\lambda)}$: Identical to $\mathsf{Expt}_{\mathcal{A},\mathsf{Real}}(1^\lambda)$ except for that Line 6 is replaced with:

8 $\{\mathsf{pd}_i\}_{i \in S} \leftarrow \mathsf{Sim}(\{\mathsf{sk}_i, \mathsf{err}_i\}_{i \in S_{\mathsf{mal}}}, \chi_{\mathsf{Sim}}, \mathsf{ct}, \mu)$

$\underline{\mathsf{Expt}_{\mathcal{A},\mathsf{Sim},\mathsf{Ideal}}(1^\lambda)}$: In addition, Line 2 in $\mathsf{Expt}_{\mathcal{A},\mathsf{Real}}(1^\lambda)$ is changed as follows:

9 $(\mathsf{sk}_1, \ldots, \mathsf{sk}_N, \mathsf{err}_1, \ldots, \mathsf{err}_N) \leftarrow \mathsf{Setup}(\mathbf{0}, \mathbf{0}, \mathbb{A})$

prove Eq. (7), it is sufficient to show that there exists Sim such that the following equations hold:

$$\mathsf{Expt}_{\mathcal{A},\mathsf{Real}} \overset{\mathsf{comp}}{\approx} \mathsf{Expt}_{\mathcal{A},\mathsf{Sim},\mathsf{Hybrid}} \tag{8}$$

$$\mathsf{Expt}_{\mathcal{A},\mathsf{Sim},\mathsf{Hybrid}} \overset{\mathsf{comp}}{\approx} \mathsf{Expt}_{\mathcal{A},\mathsf{Sim},\mathsf{Ideal}} \tag{9}$$

Eq. (9) follows by definition of the privacy of BinLSS (Definitions 2.21 and 2.22). Additionally, Theorem 2.25 shows that BinLSS supports any threshold access structure (Definition 2.24). Therefore, satisfying Eq. (8) is sufficient. We show how to construct a simulator Sim that satisfies Eq. (8) in the following.

The adversary $\mathcal{A}$ can calculate a maximal invalid share set T_{mal} (Definition 2.23) from given $\bigcup_{i \in S_{\mathsf{mal}}} T_i$ because $S_{\mathsf{mal}} \subsetneq \{P_1, \ldots, P_N\}$ in Line 2 is a maximal invalid party set (Definition 2.20). Now, we (conservatively) assume that $\mathcal{A}_2$ chooses a *valid* party set S in Line 5 and analyze the distribution of $\mathbf{p}_j$ in $\{\mathsf{pd}_i\}_{i \in S} = \{\{\mathbf{p}_j\}_{j \in T_i} \leftarrow \mathsf{PartDec}(\mathsf{pk}, \mathsf{ct}, \mathsf{sk}_i, \mathsf{err}_i)\}_{i \in S}$ such that $j \notin T_{\mathsf{mal}}$. Let $T := T_{\mathsf{mal}} \cup \{j\}$; then, T is a minimal valid share set because T_{mal} is a maximal invalid share set. Thus, from the correctness of BinLSS (Definition 2.21), we have $\sum_{i \in T} \mathbf{s}_i = \sum_{i \in T_{\mathsf{mal}}} \mathbf{s}_i + \mathbf{s}_j = \mathbf{s}^{\mathsf{mal}} + \mathbf{s}_j = \mathbf{s}$ and $\sum_{i \in T} \boldsymbol{\zeta}_i = \sum_{i \in T_{\mathsf{mal}}} \boldsymbol{\zeta}_i + \boldsymbol{\zeta}_j = \boldsymbol{\zeta}^{\mathsf{mal}} + \boldsymbol{\zeta}_j = \boldsymbol{\zeta}$, where $\mathbf{s}^{\mathsf{mal}} := \sum_{i \in T_{\mathsf{mal}}} \mathbf{s}$ and $\boldsymbol{\zeta}^{\mathsf{mal}} := \sum_{i \in T_{\mathsf{mal}}} \boldsymbol{\zeta}$. Then, by Eq. (5),

$$\begin{aligned} & b' - (\mathbf{a}')^\intercal \mathbf{s} - \mathsf{msg} = \mathbf{r}^\intercal \mathbf{e} \\ \Leftrightarrow\ & b' - (\mathbf{a}')^\intercal \mathbf{s}^{\mathsf{mal}} - \mathsf{msg} = (\mathbf{a}')^\intercal \mathbf{s}_j + \mathbf{r}^\intercal \mathbf{e} \end{aligned} \tag{10}$$

holds. The left-hand side of this equation can be computed from $\mathbf{s}^{\mathsf{mal}}$, ct, and μ, which are given to the adversary. We define this as:

$$\mathsf{Atk}_j := \mathsf{Atk}_j(\mathbf{s}^{\mathsf{mal}}, \mathsf{ct}, \mu) := b' - (\mathbf{a}')^\intercal \mathbf{s}^{\mathsf{mal}} - \mathsf{msg} = (\mathbf{a}')^\intercal \mathbf{s}_j + \mathbf{r}^\intercal \mathbf{e} \tag{11}$$

Furthermore, we define the sum of $\mathbf{p}_j = (\mathbf{a}')^\intercal \mathbf{s}_j + e_j^{\mathsf{sm}} + \mathbf{r}_{\mathsf{aux}}^\intercal \boldsymbol{\zeta}_j$ and $\mathbf{r}_{\mathsf{aux}}^\intercal \boldsymbol{\zeta}^{\mathsf{mal}}$ as Real_j, where $\mathbf{r}_{\mathsf{aux}}^\intercal \boldsymbol{\zeta}^{\mathsf{mal}}$ can be computed by the adversary:

$$\mathsf{Real}_j := \mathbf{p}_j + \mathbf{r}_{\mathsf{aux}}^\intercal \boldsymbol{\zeta}^{\mathsf{mal}} = (\mathbf{a}')^\intercal \mathbf{s}_j + e_j^{\mathsf{sm}} + \mathbf{r}_{\mathsf{aux}}^\intercal \boldsymbol{\zeta} \tag{12}$$

Combining Eqs. (11) and (12) yields:

$$(\mathbf{a}', \mathsf{Real}_j, \mathsf{Atk}_j) = (\mathbf{a}', (\mathbf{a}')^\intercal \mathbf{s}_j + e_j^{\mathsf{sm}} + \mathbf{r}_{\mathsf{aux}}^\intercal \boldsymbol{\zeta}, (\mathbf{a}')^\intercal \mathbf{s}_j + \mathbf{r}^\intercal \mathbf{e})$$

We have $\mathbf{a}' \overset{\mathsf{stat}}{\approx} \mathcal{U}(\mathbb{Z}_q^n)$ from Lemma 2.5 and Fact 2.6, and $\mathbf{s}_j \sim \mathcal{U}(\mathbb{Z}_q^n)$ from Definition 2.22. Thus, we have

$$\mathsf{Real}_j \overset{\mathsf{stat}}{\approx} \mathsf{LWE}_{\mathbf{s}_j}(n, 1, q, \chi_{\mathsf{Real}}), \tag{13}$$

$$\mathsf{Atk}_j \overset{\mathsf{stat}}{\approx} \mathsf{LWE}_{\mathbf{s}_j}(n, 1, q, \chi_{\mathsf{Atk}}), \text{ and} \tag{14}$$

$$(\mathbf{a}', \mathsf{Real}_j, \mathsf{Atk}_j) \overset{\mathsf{stat}}{\approx} \text{Reused-A-LWE}_{\mathbf{s}_j}(n, 1, q, \chi_{\mathsf{Real}}, \chi_{\mathsf{Atk}}), \tag{15}$$

where Reused-A-LWE is defined as in Definition 3.12 and

$$\chi_{\mathsf{Real}} := \chi_{\mathsf{Real}}(\boldsymbol{\zeta}, \chi_{\mathsf{enc}}, \chi_{\mathsf{sm}}) := \{e^{\mathsf{sm}} + \mathbf{r}_{\mathsf{aux}}^\intercal \boldsymbol{\zeta} \mid e^{\mathsf{sm}} \sim \chi_{\mathsf{sm}}, \mathbf{r}_{\mathsf{aux}} \sim \chi_{\mathsf{enc}}^m\}, \text{ and}$$
$$\chi_{\mathsf{Atk}} := \chi_{\mathsf{Atk}}(\mathbf{e}, \chi_{\mathsf{enc}}) := \{\mathbf{r}^\intercal \mathbf{e} \mid \mathbf{r} \sim \chi_{\mathsf{enc}}^m\}.$$

Note that the adversary can obtain fresh Real_j, Atk_j, and $(\mathbf{a}', \mathsf{Real}_j, \mathsf{Atk}_j)$ from any $m' = \mathrm{poly}(\lambda)$ ciphertexts for a fixed pk. In this case, we have $\{\mathbf{a}'^k, \mathsf{Real}_j^k, \mathsf{Atk}_j^k\}_{k \in [l]} \overset{\mathsf{stat}}{\approx} \text{Reused-A-LWE}_{\mathbf{s}_j}(n, m', q, \chi_{\mathsf{Real}}, \chi_{\mathsf{Atk}})$; the rest of proof follows similarly.

We show that both $\text{d-LWE}_{\mathbf{s}_j}(\chi_{\mathsf{Real}})$ in Eq. (13) and $\text{d-LWE}_{\mathbf{s}_j}(\chi_{\mathsf{Atk}})$ in Eq. (14) are hard. By Lemma 3.10, at least one element in $\mathbf{e}$ is larger than 1 with overwhelming probability over the choice of $(\mathsf{pk}, \mathsf{sk}, \mathsf{err}, \chi_{\mathsf{Sim}}) \leftarrow \mathsf{KeyGen}()$. Thus, we have $\chi_{\mathsf{enc}} \leq \chi_{\mathsf{Atk}}$ (Definition 3.2). In addition, $\text{d-LWE}_{\mathbf{s}_j}(n, m, q, \chi_{\mathsf{enc}})$ is hard by hypothesis. Hence, $\text{d-LWE}_{\mathbf{s}_j}(\chi_{\mathsf{Atk}})$ is hard by Lemma 3.11. We have $\chi_{\mathsf{sm}} \leq \chi_{\mathsf{Real}}$ because $e_{\mathsf{sm}} \leftarrow \chi_{\mathsf{sm}}$ is sampled independently on $\mathbf{r}_{\mathsf{aux}}^\intercal \boldsymbol{\zeta}$. In addition, $\text{d-LWE}_{\mathbf{s}_j}(n, m, q, \chi_{\mathsf{sm}})$ is hard by hypothesis. Thus, $\text{d-LWE}_{\mathbf{s}_j}(\chi_{\mathsf{Real}})$ is also hard by Lemma 3.11, even though $\mathbf{r}_{\mathsf{aux}}$ is known to the adversary (and $\boldsymbol{\zeta}$ is possibly dependent on $\mathbf{e}$, as in subsequent Theorem 4.6).

Therefore, from Theorem 3.15 and Eq. (15), we have

$$(\mathbf{a}', \mathsf{Real}_j, \mathsf{Atk}_j) \overset{\mathsf{comp}}{\approx} \mathcal{V} := \left\{ (\mathbf{a}', \mathbf{u}, \mathbf{v}) \;\middle|\; \begin{array}{l} \mathbf{u}, \mathbf{v} \sim \mathcal{U}(\mathbb{X}_q^m), \\ \mathbf{u} - \mathbf{v} = e_j^{\mathsf{sm}} + \mathbf{r}_{\mathsf{aux}}^\intercal \boldsymbol{\zeta} - \mathbf{r}^\intercal \mathbf{e}. \end{array} \right\}$$

Because $\mathbf{p}_j = \mathsf{Real}_j - \mathbf{r}_{\mathsf{aux}}^\intercal \boldsymbol{\zeta}^{\mathsf{mal}}$ by Eq. (12), we also have:

$$(\mathbf{a}', \mathbf{p}_j, \mathsf{Atk}_j - \mathbf{r}_{\mathsf{aux}}^\intercal \boldsymbol{\zeta}^{\mathsf{mal}}) \overset{\mathsf{comp}}{\approx} \mathcal{V}' := \left\{ (\mathbf{a}', \mathbf{u}', \mathbf{v}') \;\middle|\; \begin{array}{l} \mathbf{u}', \mathbf{v}' \sim \mathcal{U}(\mathbb{X}_q^m), \\ \mathbf{u}' - \mathbf{v}' = e_j^{\mathsf{sm}} + \mathbf{r}_{\mathsf{aux}}^\intercal \boldsymbol{\zeta} - \mathbf{r}^\intercal \mathbf{e} \end{array} \right\}$$

This means that, from $\mathbf{p}_j$ and $\mathsf{Atk}_j - \mathbf{r}_{\mathsf{aux}}^{\intercal}\boldsymbol{\zeta}^{\mathsf{mal}}$, it is computationally hard to obtain any information other than that we have $\mathbf{p}_j - \mathsf{Atk}_j - \mathbf{r}_{\mathsf{aux}}^{\intercal}\boldsymbol{\zeta}^{\mathsf{mal}} = e_j^{\mathsf{sm}} + \mathbf{r}_{\mathsf{aux}}^{\intercal}\boldsymbol{\zeta} - \mathbf{r}^{\intercal}\mathbf{e}$. Note that the right-hand side of this equation is a variable that follows χ_{Sim}, which is defined in Eq. (3). Hence, we sample $e_{\mathsf{sim}} \leftarrow \chi_{\mathsf{Sim}}$ and define

$$\mathsf{Sim}'_j := \mathsf{Atk}_j(\mathbf{s}^{\mathsf{mal}}, \mathsf{ct}, \mu) - \mathbf{r}_{\mathsf{aux}}^{\intercal}\boldsymbol{\zeta}^{\mathsf{mal}} + e_{\mathsf{sim}}.$$

Then, Sim'_j and $\mathbf{p}_j$ are identically distributed. Therefore, if we generate each $\mathbf{p}_j$ of $\{\mathsf{pd}_i\}_{i\in S} = \{\{\mathbf{p}_j\}_{j\in T_i}\}_{i\in S}$ in Line 8, Algorithm 2, as

$$\mathsf{Sim}_j(\{\mathbf{s}_i, \boldsymbol{\zeta}_i\}_{i\in T_{\mathsf{mal}}}, \chi_{\mathsf{Sim}}, \mathsf{ct}, \mu) := \begin{cases} \mathbf{p}_j & (j \in T_{\mathsf{mal}}) \\ \mathsf{Sim}'_j & (j \notin T_{\mathsf{mal}}) \end{cases}, \tag{16}$$

then $\{\mathsf{pd}_i\}_{i\in S}$ in Line 6 and Line 8 are identically distributed. Thus, we obtain Eq. (8) for Sim, which is constructed as in Eq. (16). □

4.4 Instantiation: ThPKE without Known-norm LWE

In this subsection, we show in Theorem 4.6 that there exist instances that simultaneously satisfy correctness (Lemma 4.2) and security (Theorem 4.5).

We disclose χ_{Sim} (Eq. (3)) to the adversary $\mathcal{A}$ and Sim in Algorithm 2 because the adversary can observe variables that follow χ_{Sim} repeatedly for any $\mathrm{poly}(\lambda)$ iterations by calculating $\mathsf{Real}_j - \mathsf{Atk}_j$, where Atk_j and Real_j are defined in Eq. (11) and Eq. (12), respectively.

In the construction of [35], χ_{Sim} corresponds to $\mathcal{N}_{\sqrt{\sigma_{\mathsf{sm}}^2 + \sigma_{\mathsf{enc}}^2\|\mathbf{e}\|^2}}$. Hence, the adversary can accurately estimate $\|\mathbf{e}\|$ by calculating the variance of χ_{Sim}. Therefore, the ThPKE scheme of [35] require to use "known-norm LWE" to prove the security of the underlying PKE.

In contrast, our scheme discloses (the probability function of) χ_{Sim} as defined in Eq. (3), and it is assumed that no information about $\mathbf{e}$ can be obtained from χ_{Sim} in Theorem 4.5. We show in subsequent Theorem 4.6 that there exist distributions χ_{pk}, χ_{err}, χ_{enc} and χ_{sm} that satisfy this assumption. We select χ_{pk} as a B_{pk}-bounded distribution. As a naïve attempt, we may define χ_{err} as a B_{err}-bounded distribution. We also select $\chi_{\mathsf{enc}} := D_{\mathbb{Z},s_{\mathsf{enc}}}$ and $\chi_{\mathsf{sm}} := D_{\mathbb{Z},s_{\mathsf{sm}}}$; then, by Lemma 2.13, we have $\chi_{\mathsf{Sim}} \overset{\mathsf{stat}}{\approx} D_{\sqrt{s_{\mathsf{sm}}^2 + s_{\mathsf{enc}}^2(\|\mathbf{e}\|^2 + \|\boldsymbol{\zeta}\|^2)}}$ for some s_{enc} and s_{sm}. Thus, we disclose $\|\mathbf{e}\|^2 + \|\boldsymbol{\zeta}\|^2$ to the adversary. In this case, unfortunately, $\|\mathbf{e}\|^2 + \|\boldsymbol{\zeta}\|^2$ discloses a (nontrivially) small upper-bound $\|\mathbf{e}\| \le \sqrt{\|\mathbf{e}\|^2 + \|\boldsymbol{\zeta}\|^2} < \sqrt{m}B_{\mathsf{pk}}$ when $\sqrt{\|\mathbf{e}\|^2 + \|\boldsymbol{\zeta}\|^2} < \sqrt{m}B_{\mathsf{pk}}$, which occurs with nonnegligible probability. Similarly, when $\sqrt{\|\mathbf{e}\|^2 + \|\boldsymbol{\zeta}\|^2} > B_{\mathsf{err}}$, a lower-bound is revealed as follows: $\sqrt{\|\mathbf{e}\|^2 + \|\boldsymbol{\zeta}\|^2} - B_{\mathsf{err}} \le \|\mathbf{e}\|$. The possibility of other information leaking cannot be denied.

In Theorem 4.6, we avoid these leakages by selecting $\boldsymbol{\zeta} \leftarrow \chi_{\mathsf{err}} := \chi_{\mathsf{err}}(\mathbf{e}, B)$ conditioned on a fixed $\mathbf{e} \leftarrow \chi_{\mathsf{pk}}$ generated in the KeyGen algorithm so that $\sqrt{\|\mathbf{e}\|^2 + \|\boldsymbol{\zeta}\|^2}$ becomes a public parameter B that is defined from χ_{pk}. Then,

we have $\chi_{\mathsf{Sim}} \overset{\mathsf{stat}}{\approx} D_{\sqrt{s_{\mathsf{sm}}^2+s_{\mathsf{enc}}^2 B^2}}$; thus, χ_{Sim} contains no information about $\mathbf{e}$ (other than $\mathbf{e} \sim \chi_{\mathsf{pk}}$). Note that we instantiate $\chi_{\mathsf{pk}} := D_{\mathbb{Z},s_{\mathsf{pk}}}$ in Theorem 4.6 for concreteness. Some other bounded distributions over $\mathbb{Z}$ can also be used.

Theorem 4.6. *Select parameters N, n, q, $m \geq n\log q + 2\lambda$, $s_{\mathsf{pk}} \geq \tilde{\eta}_\epsilon^+(\mathbb{Z})$, $s_{\mathsf{sm}} \geq \tilde{\eta}_\epsilon^+(\mathbb{Z})$, and $s_{\mathsf{enc}} \geq \sqrt{\lambda} s_{\mathsf{pk}} \tilde{\eta}_\epsilon^+(\mathbb{Z})$ such that $\sqrt{\lambda(N s_{\mathsf{sm}}^2 + B^2 s_{\mathsf{enc}}^2)} < \lfloor \frac{q}{4} \rfloor$, where $B := \sqrt{\lceil 2m s_{\mathsf{pk}}^2 \rceil}$. Let $\chi_{\mathsf{pk}} = D_{\mathbb{Z},s_{\mathsf{pk}}}$, $\chi_{\mathsf{enc}} = D_{\mathbb{Z},s_{\mathsf{enc}}}$ and $\chi_{\mathsf{sm}} = D_{\mathbb{Z},s_{\mathsf{sm}}}$. In $\mathsf{KeyGen}()$, we generate and fix $\mathsf{pk} := (\mathbf{A}, \mathbf{b} := \mathbf{A}\mathbf{s} + \mathbf{e})$, and then, define*

$$\chi_{\mathsf{err}} := \chi_{\mathsf{err}}(\mathbf{e}, B) \text{ as a distribution over } \{\boldsymbol{\zeta} \in \mathbb{Z}_q^m \mid \|\boldsymbol{\zeta}\|^2 = B^2 - \|\mathbf{e}\|^2\} \text{ such that } \|\boldsymbol{\zeta}\|_\infty < 2s_{\mathsf{pk}}. \tag{17}$$

Assume that $\mathsf{d\text{-}LWE}_{\mathbf{s}}(n, m, q, D_{\mathbb{Z},s_{\mathsf{pk}}})$, $\mathsf{d\text{-}LWE}_{\mathbf{s}}(n, m, q, D_{\mathbb{Z},s_{\mathsf{enc}}})$, and $\mathsf{d\text{-}LWE}_{\mathbf{s}}(n, m, q, D_{\mathbb{Z},s_{\mathsf{sm}}})$ are all hard. Then, Algorithm 1 instantiated (and modified) as above satisfies SS (Definition 4.4) and correctness (Definition 4.1).

Proof. By Lemma 2.17, $\|\mathbf{e}\| \leq \sqrt{m} s_{\mathsf{pk}}$ holds with overwhelming probability over the choice of $\mathbf{e} \leftarrow (D_{\mathbb{Z},s_{\mathsf{pk}}})^m$. Note that $(D_{\mathbb{Z},s_{\mathsf{pk}}})^m = D_{\mathbb{Z}^m,s_{\mathsf{pk}}}$ by Fact 2.10. Thus, $R := \sqrt{B^2 - \|\mathbf{e}\|^2} = \sqrt{\lceil 2m s_{\mathsf{pk}}^2 \rceil - \|\mathbf{e}\|^2} \in (\sqrt{m} s_{\mathsf{pk}}, \sqrt{2m} s_{\mathsf{pk}})$ and $\beta := \lfloor R/\sqrt{\frac{m}{2}} \rfloor < 2s_{\mathsf{pk}}$ holds with overwhelming probability. Hence, there exists χ_{err} that satisfies Eq. (17) by Lemma 4.7 (which is shown subsequently).

By Lemma 2.13, we have $\chi_{\mathsf{Sim}} \overset{\mathsf{stat}}{\approx} D_{\sqrt{s_{\mathsf{sm}}^2 + (\|\mathbf{e}\|^2 + \|\boldsymbol{\zeta}\|^2) s_{\mathsf{enc}}^2}} = D_{\sqrt{s_{\mathsf{sm}}^2 + B^2 s_{\mathsf{enc}}^2}}$ (Eq. (3)). This has no information about $\mathbf{e}$ because $B^2 = \lceil 2m s_{\mathsf{pk}}^2 \rceil$. By hypothesis described in Eq. (17), the probability of recovering $\boldsymbol{\zeta} \sim \chi_{\mathsf{err}}$ is less than $2^{-\lambda}$. Therefore, this instantiation satisfies SS by Theorem 4.5.

The correctness can be proven in a manner similar to that in Example 4.3. By Lemma 2.18, we have $\Pr_{x \leftarrow \chi_{\mathsf{Sim},|T|}}[x < \sqrt{\lambda(N s_{\mathsf{sm}}^2 + B^2 s_{\mathsf{enc}}^2)}] = 1 - \mathrm{negl}(n)$, where $\chi_{\mathsf{Sim},|T|}$ is defined as in (4). Thus, the correctness holds when we select n, $m, q, s_{\mathsf{enc}}, s_{\mathsf{sm}}$ such that $\sqrt{\lambda(N s_{\mathsf{sm}}^2 + B^2 s_{\mathsf{enc}}^2)} < \lfloor \frac{q}{4} \rfloor$ by Lemma 4.2. □

We complete the proof of Theorem 4.6 by proving deferred Lemma 4.7: We show an example of the distribution χ_{err} that satisfies Eq. (17):

Lemma 4.7. *Let $m \geq 2\lambda$, $q \in \mathbb{N}$ and R be a real number such that $R^2 \in \mathbb{N}$, $10\sqrt{\frac{m}{2}} < R < \sqrt{m} q$ and $R < \sqrt{\frac{m}{2}} \cdot 2^{\frac{m}{4}-6}$. Define a set $S_R := \{\boldsymbol{\zeta} \in \mathbb{Z}_q^m \mid \|\boldsymbol{\zeta}\|^2 = R^2\}$. Then, there exists a (efficiently samplable) distribution χ_{err} over S_R that satisfies $\|\boldsymbol{\zeta}\|_\infty \leq \beta := \lfloor R/\sqrt{\frac{m}{2}} \rfloor$ for any $\boldsymbol{\zeta} \leftarrow \chi_{\mathsf{err}}$.*

Proof. We define χ_{err} as the distribution formed by $\boldsymbol{\zeta}$ sampled by Algorithm 3. The algorithm first samples binary random numbers $\boldsymbol{\zeta}_1 \xleftarrow{\$} \{(\beta-1), \beta\}^{\frac{m}{2}}$.Then it deterministically selects $\boldsymbol{\zeta}_2 \in \mathbb{Z}_q^{m/2}$ such that $\boldsymbol{\zeta} := (\boldsymbol{\zeta}_1 \,\|\, \boldsymbol{\zeta}_2)$ satisfy $\|\boldsymbol{\zeta}\|^2 = R^2$ and $\|\boldsymbol{\zeta}\|_\infty \leq \beta$.

As you can see from the comments written in Algorithm 3, we can show that $r_1 \leq \beta$ holds at Line 8 by construction. Here, we show that $r_{t+7} = 0$ holds at

Algorithm 3: An algorithm to sample $\boldsymbol{\zeta} \in S_R$

Input : $R \in \mathbb{R}$ s.t. $R^2 \in \mathbb{N}$, $10\sqrt{\frac{m}{2}} \leq R < \sqrt{mq}$ and $R < \sqrt{\frac{m}{2}} \cdot 2^{\frac{m}{4}-6}$
Output : $\boldsymbol{\zeta} \in S_R := \{\boldsymbol{\zeta} \in \mathbb{Z}_q^m \mid \|\boldsymbol{\zeta}\|^2 = R^2\}$ s.t. $\|\boldsymbol{\zeta}\|_\infty \leq \beta$

1 Define $\beta := \lfloor R/\sqrt{\frac{m}{2}} \rfloor \geq 10$ // $\beta\sqrt{\frac{m}{2}} \leq R < (\beta+1)\sqrt{\frac{m}{2}}$
2 Sample $\boldsymbol{\zeta}_1 \xleftarrow{\$} \{(\beta-1), \beta\}^{\frac{m}{2}}$ // $\|\boldsymbol{\zeta}_1\|^2 \geq (\beta-1)^2\frac{m}{2}$
3 $\overline{R} := \sqrt{R^2 - \|\boldsymbol{\zeta}_1\|^2} \in \mathbb{R}$ // $\overline{R}^2 \leq R^2 - (\beta-1)^2\frac{m}{2} < 2\beta m$
4 $m' := \lfloor \overline{R}^2 / \lceil\sqrt{8\beta}\rceil^2 \rfloor$ // $m' \leq \overline{R}^2/8\beta < \frac{m}{4}$
5 $u_1, \ldots, u_{m'} := \lceil\sqrt{8\beta}\rceil$ // $\sum_{i=1}^{m'} u_i^2 = \lceil\sqrt{8\beta}\rceil^2 m' \leq \overline{R}^2 < \lceil\sqrt{8\beta}\rceil^2(m'+1)$
6 $r_1 := \sqrt{\overline{R}^2 - \sum_{i=1}^{m'} u_i^2} \in \mathbb{R}$ // $0 \leq r_1 < \lceil\sqrt{8\beta}\rceil \leq \beta$
7 $t := \lfloor \log r_1 \rfloor - 1$ // $t \leq \log r_1 < \log\beta < \frac{m}{4} - 6$
8 **for** $i = 2$ **to** $t+7$ **do** $r_i := \sqrt{r_{i-1}^2 - \lfloor r_{i-1} \rfloor^2}$
9 $\boldsymbol{\zeta}_2 := (u_1, \ldots, u_{m'}, \lfloor r_1 \rfloor, \ldots, \lfloor r_{t+6} \rfloor, 0, \ldots, 0) \in \mathbb{Z}_q^{m/2}$ // $r_{t+7} = 0, \|\boldsymbol{\zeta}_2\|^2 = \overline{R}^2$
10 **return** $\boldsymbol{\zeta} := (\boldsymbol{\zeta}_1 \parallel \boldsymbol{\zeta}_2) \in \mathbb{Z}_q^m$, *which is the concatenation of* $\boldsymbol{\zeta}_1, \boldsymbol{\zeta}_2 \in \mathbb{Z}_q^{m/2}$

Line 9. For any $t \geq 2$,

$$r_t := \sqrt{r_{t-1}^2 - \lfloor r_{t-1} \rfloor^2} < \sqrt{2r_{t-1}} \tag{18}$$

holds because we have $x^2 - \lfloor x \rfloor^2 = x^2 - (x-\xi)^2 = 2x\xi - \xi^2 < 2x$ ($\xi := x - \lfloor x \rfloor \in [0,1)$) for any $x \in \mathbb{R}_{>0}$. Hence, because $\sqrt{2x} \leq x/2$ for any $x \geq 8$, we have $r_t < \sqrt{2r_{t-1}} \leq \frac{1}{2}r_{t-1} < \frac{1}{2}\sqrt{2r_{t-2}} \leq \frac{1}{2^2}r_{t-2} < \cdots \leq \frac{1}{2^{t-1}}r_1$, when $r_{t-1} \geq 8$. Therefore, let $t := \lfloor \log r_1 \rfloor - 1$, then we have $r_t < \frac{1}{2^{t-1}}r_1 < 8$ because $\log r_1 < t+2 \Leftrightarrow r_1 < 2^{t+2}$. Furthermore, again by Eq. (18), we have $r_t < 8, r_{t+1} < 4, r_{t+2} < \sqrt{8}$. Additionally, note that $r_t^2 \in \mathbb{Z}$ holds for any t by Eq. (18) because r_1^2 is an integer. Thus, $r_{t+2}^2 \in \{0, 1, \ldots, 7\}$. Furthermore, $r_t < r_{t-1}$ holds for any $r_{t-1} \geq 1$ because we have: $r_t := \sqrt{r_{t-1}^2 - \lfloor r_{t-1} \rfloor^2} < r_{t-1} \Leftrightarrow 1 \leq r_{t-1}$. Thus, we have $r_{t+3} < \sqrt{7}, r_{t+4} < \sqrt{6}, r_{t+5} < \sqrt{5}, r_{t+6} < \sqrt{4} = 2$. Hence, $r_{t+6}^2 \in \{0,1\}$, i.e., $r_{t+6} \in \{0,1\}$ holds, and this implies that $r_{t+7} = 0$. Because $\sqrt{\frac{m}{2}} \cdot 2^{\frac{m}{4}-6}$ by hypothesis, we have $t + 6 < \log r_1 + 6 < \log\beta + 6 < \log 2^{\frac{m}{4}-6} + 6 = \frac{m}{4}$. Thus, we have $m' + t + 6 < \frac{m}{2}$; this $\boldsymbol{\zeta}_2$ defined as in Line 9 is in $\mathbb{Z}_q^{m/2}$. By $r_t^2 := r_{t-1}^2 - \lfloor r_{t-1} \rfloor^2$, $\sum_{i=1}^{t-1} \lfloor r_i \rfloor^2 = r_1^2 - r_t^2$ holds. Hence, we have $\sum_{i=1}^{t+6} \lfloor r_i \rfloor^2 = r_1^2 - r_{t+7}^2 = r_1^2$ and $\|\boldsymbol{\zeta}_2\|^2 = r_1^2 + \sum_{i=1}^{m'} u_i^2 = \overline{R}^2$. Note that every element in $\boldsymbol{\zeta}$ is $\leq \beta$, i.e., $\|\boldsymbol{\zeta}\|_\infty \leq \beta$. □

5 Simulation-Secure ThPKE from Ring-LWE

We construct our ThPKE from the Ring-LWE-based PKE presented by Lyubashevsky, Peikert and Regev [29].

We first provide definitions and preliminaries related with the Ring-LWE problem in Section 5.1. Then, we present our Ring-LWE-based ThPKE in Section 5.2 (with general error distributions). We define and prove the correctness

and simulation-security of the (general) ThPKE scheme in Section 5.3 and Section 5.4, respectively. Finally, in Section 5.5, we provide a concrete scheme instantiated with discrete Gaussian distributions and prove the correctness and simulation-security.

5.1 Preliminaries for Ring-LWE

We first define the Ring-LWE problem as follows:

Definition 5.1 (Ring-LWE). *For security parameter λ, let $n = n(\lambda)$ be a power of 2, and let $q = q(\lambda) \geq 2$ be an integer, and let $\mathcal{R} = \mathbb{Z}[X]/(X^n + 1)$ and $\mathcal{R}_q = \mathbb{Z}_q[X]/(X^n + 1)$. Let $\chi = \chi(\lambda)$ be an error distribution over $\mathcal{R}$. The Ring-LWE distribution for a fixed secret $s \leftarrow \chi$ (note that s is not uniformly random, but sampled from χ) is defined as follows:*

$$\mathsf{RLWE}_s(n, q, \chi) := \{(a, b) \mid a \xleftarrow{\$} \mathcal{R}_q, b = s \cdot a + e, e \leftarrow \chi\}$$

Definition 5.2 (Decision Ring-LWE). $\mathsf{d\text{-}RLWE}_s(n, q, \chi)$ *is a problem to distinguish* $\mathsf{RLWE}_s(n, q, \chi)$ *and* $\mathcal{U}(\mathcal{R}_q \times \mathcal{R}_q)$.

We omit the search version of Ring-LWE because we do not use it. *coefficient vector*, *coefficient matrix*, and *coefficient Gram matrix* of $a \in \mathcal{R}$:

Definition 5.3. *Let $a = \sum_{i=0}^{n-1} a_i X^i \in \mathcal{R}$. We define the* coefficient vector *of a as $\mathbf{a} := (a_0, a_1, \ldots, a_{n-1})^\intercal$. Define a signed permutation matrix $\mathbf{P} := \left(\begin{array}{c|c} \mathbf{0} & -1 \\ \hline \mathbf{I}_{n-1} & \mathbf{0} \end{array}\right) \in \mathbb{Z}^{n \times n}$ and define the* coefficient matrix *of a as $\mathbf{A} := \left(\mathbf{a}\ \mathbf{Pa} \cdots \mathbf{P}^{n-1}\mathbf{a}\right) \in \mathbb{Z}^{n \times n}$. The* coefficient Gram matrix *of a is defined as $\mathbf{\Sigma}_a := \mathbf{A}\mathbf{A}^\intercal \in \mathbb{Z}^{n \times n}$.*

For $a \in \mathcal{R}$, we define $\|a\|_\infty := \|\mathbf{a}\|_\infty$, where $\mathbf{a}$ is the coefficient vector of a. We extend the notation $a \sim \chi$ for distributions χ over $\mathbb{Z}^n$ and $a \in \mathcal{R}$:

Definition 5.4. *For $a \in \mathcal{R}$ and a distribution χ over $\mathbb{Z}^n$, we write $a \sim \chi$ (resp. $a \leftarrow \chi$) to mean $\mathbf{a} \sim \chi$ (resp. $\mathbf{a} \leftarrow \chi$), where $\mathbf{a}$ is the coefficient vector of a.*

The coefficient matrix is useful to derive the coefficient vector of a product:

Fact 5.5. *For $r, e \in \mathcal{R}$, the coefficient vector of $re \in \mathcal{R}$ is $\mathbf{Er}$, where $\mathbf{E}$ is the coefficient matrix of e and $\mathbf{r}$ is the coefficient vector of r.*

We sometimes use a (slight) variant of RLWE where $s \xleftarrow{\$} \mathcal{R}_q$ instead of $s \leftarrow \chi$, which is denoted by $\mathsf{RLWE}_{s \sim \mathcal{U}(\mathcal{R}_q)}$. Note that $\mathsf{d\text{-}RLWE}_{s \sim \mathcal{U}(\mathcal{R}_q)}(n, q, \chi)$ is at least as hard as $\mathsf{d\text{-}RLWE}_s(n, q, \chi)$:

Lemma 5.6. *If $\mathsf{d\text{-}RLWE}_s(n, q, \chi)$, then $\mathsf{d\text{-}RLWE}_{s \sim \mathcal{U}(\mathcal{R}_q)}(n, q, \chi)$ is also hard.*

Proof. Given $(a, b) \leftarrow \mathsf{RLWE}_s(n, q, \chi)$, sample $\tilde{s} \xleftarrow{\$} \mathcal{R}_q$ and output $(a, b') := (a, b + a\tilde{s})$. Then, $(a, b') \sim \mathsf{RLWE}_{s' \sim \mathcal{U}(\mathcal{R}_q)}(n, q, \chi)$. □

We also have the counterpart of Lemmas 3.10 and 3.11 for Ring-LWE:

Corollary 5.7. *If $\mathsf{d\text{-}RLWE}_s(n, q, \chi)$ is hard, the probability $P := \Pr_{e \leftarrow \chi}[e = 0]$ is negligible.*

Corollary 5.8. *If* $\mathsf{d\text{-}RLWE}_{s_1}(n, q, \chi_1)$ *is hard, then for any* $\chi_2 \geq \chi_1$, $\mathsf{d\text{-}RLWE}_{s_2}(n, q, \chi_2)$ *is also hard.*

Proof. By Definition 3.2, there exists χ_δ such that $\chi_2 \overset{\mathsf{stat}}{\approx} \chi_1 + \chi_\delta$. Given $(a, b) \leftarrow \mathsf{RLWE}_{s_1}(n, q, \chi_1)$, sample $s_\delta, e_\delta \xleftarrow{\$} \chi_\delta$ and output $(a, b') := (a, b + as_\delta + e_\delta)$; then, $(a, b') \overset{\mathsf{stat}}{\approx} \mathsf{RLWE}_{s_2}(n, q, \chi)$. □

We define the "Reused-A" variant of RLWE, which is a counterpart of Definition 3.12. Note that here we sample uniformly random s.

Definition 5.9 (Reused-A-RLWE). *Let* $n, q \in \mathbb{N}$. *Let* χ_1, χ_2 *be distributions over* $\mathcal{R}$. *For a fixed* $s \xleftarrow{\$} \mathcal{R}_q$, *we define* $\mathsf{Reused\text{-}A\text{-}RLWE}_s(n, q, \chi_1, \chi_2)$ *as follows:*

$$\left\{ (a, b_1 := as + e_1, b_2 := as + e_2) \;\middle|\; \begin{array}{l} a \sim \mathcal{U}(\mathcal{R}_q), e_1 \sim \chi_1, e_2 \sim \chi_2 \\ b_1 - b_2 = e_1 - e_2 \end{array} \right\}$$

Definition 5.10 (d-Reused-A-LWE). *The* $\mathsf{d\text{-}Reused\text{-}A\text{-}LWE}(n, q, \chi_1, \chi_2)$ *is a problem to distinguish* $\mathsf{Reused\text{-}A\text{-}LWE}_s(n, q, \chi_1, \chi_2)$ *distribution from* $\mathcal{V} := \{(a, u, v) \mid a, u, v \sim \mathcal{U}(\mathcal{R}_q), e_1 \sim \chi_1, e_2 \sim \chi_2, u - v = e_1 - e_2\}$.

We have a counterpart of Corollary 3.16 by Lemma 5.6: and Corollary 5.8

Corollary 5.11 (d-RLWE $\leq$ d-Reused-A-LWE). *Let* χ_1 *and* χ_2 *be distributions over* $\mathcal{R}$ *such that* $\chi_1 \leq \chi_2$. *If there exists an algorithm that solves* $\mathsf{d\text{-}Reused\text{-}A\text{-}RLWE}(n, q, \chi_1, \chi_2)$ *with advantage* ϵ, *then there exists an algorithm that solves* $\mathsf{d\text{-}RLWE}(n, q, \chi_1)$ *with advantage at least* $\epsilon/2$.

5.2 Construction

We present our construction of ThPKE from Ring-LWE in Algorithm 4. This is essentially equivalent to Algorithm 1, except for that the underlying PKE is replaced with the Ring-LWE-based PKE presented by Lyubashevsky, Peikert and Regev [29].

Because the syntax of Algorithm 4 is identical to that of Algorithm 1, we can define the correctness and the simulation security of Algorithm 4 by Definition 4.1 and Definition 4.4, respectively.

5.3 Correctness

As preparation, we define a distribution $\chi_{\mathsf{Sim},t}(s, e, \zeta)$ with a parameter $t \leq N$, which is a generalization of χ_{Sim} that is defined in Eq. (19):

$$\chi_{\mathsf{Sim},t}(s, e, \zeta) := \left\{ \sum_{i=1}^{t} e_i^{\mathsf{sm}} + r_{\mathsf{aux}}\zeta - re - e_1 s - e_2 \;\middle|\; \begin{array}{l} e_1^{\mathsf{sm}}, \ldots, e_t^{\mathsf{sm}} \overset{\mathrm{iid}}{\sim} \chi_{\mathsf{sm}} \\ r, r_{\mathsf{aux}}, e_1, e_2 \overset{\mathrm{iid}}{\sim} \chi_{\mathsf{pk}} \end{array} \right\} \quad (20)$$

Then, we show the sufficient condition for Algorithm 4 to be correct, i.e., the counterpart of Lemma 4.2:

Algorithm 4: Our RLWE-based $\mathsf{ThPKE} := (\mathsf{Params}, \mathsf{KeyGen}, \mathsf{Setup}, \mathsf{Enc}, \mathsf{PartDec}, \mathsf{FinDec})$

$\underline{\mathsf{Params}(1^\lambda, 1^N) \to \mathsf{pp}}$:

1 Output public parameters $\mathsf{pp} := (n, m, q, \chi_{\mathsf{pk}}, \chi_{\mathsf{err}}, \chi_{\mathsf{sm}})$.

Note: The following functions implicitly take pp as an argument.

$\underline{\mathsf{KeyGen}() \to (\mathsf{pk}, \mathsf{sk}, \mathsf{err}, \chi_{\mathsf{Sim}})}$:

2 $\mathsf{sk} := s \leftarrow \chi_{\mathsf{pk}}$, $\mathsf{pk} := (a, b := sa + e) \leftarrow \mathsf{RLWE}_s(n, q, \chi_{\mathsf{pk}})$

3 $\mathsf{err} := \zeta \leftarrow \chi_{\mathsf{err}}$, and define a distribution χ_{Sim} as follows:

$$\chi_{\mathsf{Sim}}(s, e, \zeta) := \{e^{\mathsf{sm}} + r_{\mathsf{aux}}\zeta - re - e_1 s - e_2 \mid e^{\mathsf{sm}} \sim \chi_{\mathsf{sm}}, r, r_{\mathsf{aux}}, e_1, e_2 \overset{\text{iid}}{\sim} \chi_{\mathsf{pk}}\} \quad (19)$$

$\underline{\mathsf{Setup}(\mathsf{sk}, \mathsf{err}, \mathbb{A}) \to (\mathsf{sk}_1, \ldots, \mathsf{sk}_N, \mathsf{err}_1, \ldots, \mathsf{err}_N)}$:

4 Perform $\mathsf{BinLSS.Share}((s, \zeta), \mathbb{A}) \to \{(\mathsf{sk}_i, \mathsf{err}_i) := \{(s_j, \zeta_j)\}_{j \in T_i}\}_{i \in [N]}$

Note: BinLSS is performed for the coefficient vectors

$\underline{\mathsf{Enc}(\mathsf{pk}, \mu \in R_2) \to \mathsf{ct}}$:

5 Sample $r_{\mathsf{aux}}, r, e_1, e_2 \leftarrow \chi_{\mathsf{pk}}$, and define $\mathsf{msg} := \lfloor \frac{q}{2} \rfloor \cdot \mu$

6 $(a', b') := (ar - e_1, br + e_2 + \mathsf{msg})$

7 Output $\mathsf{ct} := (a', b', r_{\mathsf{aux}})$

$\underline{\mathsf{PartDec}(\mathsf{ct}, \mathsf{sk}_i, \mathsf{err}_i) \to \mathsf{pd}_i}$:

8 Parse $\mathsf{sk}_i = \{s_j\}_{j \in T_i}, \mathsf{err}_i = \{\zeta_j\}_{j \in T_i}$

9 **for** $j \in T_i$ **do** Sample $e_j^{\mathsf{sm}} \leftarrow \chi_{\mathsf{sm}}$, and define $p_j := a' s_j + e_j^{\mathsf{sm}} + r_{\mathsf{aux}}\zeta_j$

10 Output $\mathsf{pd}_i := \{p_j\}_{j \in T_i}$

$\underline{\mathsf{FinDec}(\{\mathsf{pd}_i\}_{i \in S}) \to \overline{\mu} \in \{0, 1\} \text{ or } \perp}$:

11 **if** $S \notin \mathbb{A}$ **then** Output $\perp$ and **break**

12 Otherwise, parse $\{\mathsf{pd}_i\}_{i \in S} = \{\{p_j\}_{j \in T_i}\}_{i \in S}$

13 Calculate minimal valid share set $T \subseteq \bigcup_{i \in S} T_i$ (Definition 2.23)

14 Output $\overline{\mu} := \lfloor (b' - \sum_{i \in T} p_i) / \lfloor \frac{q}{2} \rfloor \rceil$

Lemma 5.12. *The ThPKE scheme defined in Algorithm 4 is correct if we have* $\Pr_{x \leftarrow \chi_{\mathsf{Sim},t}}\left[\|x\|_\infty < \lfloor \frac{q}{4} \rfloor\right] = 1 - \mathrm{negl}(\lambda)$ *for an overwhelming proportion of* $(\mathsf{pk}, \mathsf{sk}, \mathsf{err}) \leftarrow \mathsf{KeyGen}()$*, where* $\chi_{\mathsf{Sim},t} := \chi_{\mathsf{Sim},t}(\mathbf{e}, \boldsymbol{\zeta})$ *is defined as in* (4) *and* $t = |T| (\leq N)$.

Proof. At Line 14 in Algorithm 4, we have:

$$\begin{aligned} b' - \textstyle\sum_{i \in T} p_i &= b' - a's - \textstyle\sum_{i \in T} e_i^{\mathsf{sm}} - r_{\mathsf{aux}}\zeta \\ &= \mathsf{msg} - (\textstyle\sum_{i \in T} e_i^{\mathsf{sm}} + r_{\mathsf{aux}}\zeta - re - e_1 s - e_2) \end{aligned} \quad (21)$$

By hypothesis, $\|\sum_{i \in T} e_i^{\mathsf{sm}} + r_{\mathsf{aux}}\zeta - re - e_1 s - e_2\|_\infty < \lfloor \frac{q}{4} \rfloor$ holds with overwhelming probability. Thus, $\overline{\mu} := \lfloor (b' - \sum_{i \in T} p_i) / \lfloor \frac{q}{2} \rfloor \rceil = \mu$ also holds with overwhelming probability. □

5.4 Simulation Security

We provide the proof of the security of Algorithm 4, which is a counterpart of Theorem 4.5:

Theorem 5.13. *Assume that both* $\mathsf{d\text{-}RLWE}(n, q, \chi_{\mathsf{pk}})$ *and* $\mathsf{d\text{-}RLWE}(n, q, \chi_{\mathsf{sm}})$ *are hard. In addition, assume that it is hard to obtain any information about* e, s *from (the probability function of)* $\chi_{\mathsf{Sim}}(s, e, \zeta)$ *defined in Eq. (19), where* $s, e \leftarrow \chi_{\mathsf{pk}}$ *and* $\zeta \leftarrow \chi_{\mathsf{err}}$*. Then, Algorithm 4 satisfies SS (Definition 4.4).*

Proof. The proof is identical to that of Theorem 4.5 from the beginning to Eq. (10). Thus, we start from the counterpart of Eq. (10). We denote the share of s and ζ as $\{s_i\}_{i\in T}$ and $\{\zeta_i\}_{i\in T}$ and recall that we have $\sum_{i\in T} s_i = s$ and $\sum_{i\in T} \zeta_i = \zeta$ holds. We also define $\sum_{i\in T_{\mathsf{mal}}} s_i = s^{\mathsf{mal}}$ and $\sum_{i\in T_{\mathsf{mal}}} \zeta_i = \zeta^{\mathsf{mal}}$. By Eq. (21) and $s = s^{\mathsf{mal}} + s_j$, we have

$$\begin{aligned} b' - a's - \mathsf{msg} &= re + e_1 s + e_2 \\ \Leftrightarrow b' - a's^{\mathsf{mal}} - \mathsf{msg} &= a's_j + re + e_1 s + e_2 \end{aligned} \tag{22}$$

We define the left-hand side of Eq. (22) as:

$$\mathsf{Atk}_j := \mathsf{Atk}_j(s^{\mathsf{mal}}, \mathsf{ct}, \mu) := b' - a's^{\mathsf{mal}} - \mathsf{msg} = a's_j + re + e_1 s + e_2 \tag{23}$$

Furthermore, we define:

$$\mathsf{Real}_j := p_j + r_{\mathsf{aux}}\zeta^{\mathsf{mal}} = a's_j + e_j^{\mathsf{sm}} + r_{\mathsf{aux}}\zeta \tag{24}$$

Combining Eq. (23) and Eq. (24) yields:

$$(a', \mathsf{Real}_j, \mathsf{Atk}_j) = (a', a's_j + e_j^{\mathsf{sm}} + r_{\mathsf{aux}}\zeta, a's_j + re + e_1 s + e_2)$$

We have $a' := ar - e_1 \overset{\mathsf{comp}}{\approx} \mathcal{U}(R_q)$ because $\mathsf{d\text{-}RLWE}_r(n, q, \chi_{\mathsf{pk}})$ is hard by hypothesis (we use $\mathsf{d\text{-}RLWE}$ assumption instead of the ring version of Lemma 2.5, as in [29]). We also have $s_j \sim \mathcal{U}(R_q)$ from Definition 2.22. Thus, we have

$$\mathsf{Real}_j \overset{\mathsf{comp}}{\approx} \mathsf{RLWE}_{s_j\sim\mathcal{U}(R_q)}(n, q, \chi_{\mathsf{Real}}), \tag{25}$$

$$\mathsf{Atk}_j \overset{\mathsf{comp}}{\approx} \mathsf{RLWE}_{s_j\sim\mathcal{U}(R_q)}(n, q, \chi_{\mathsf{Atk}}), \text{ and} \tag{26}$$

$$(a', \mathsf{Real}_j, \mathsf{Atk}_j) \overset{\mathsf{comp}}{\approx} \mathsf{Reused\text{-}A\text{-}RLWE}_{s_j}(n, q, \chi_{\mathsf{Real}}, \chi_{\mathsf{Atk}}), \tag{27}$$

where $\mathsf{Reused\text{-}A\text{-}RLWE}$ is defined as in Definition 5.9 and

$$\begin{aligned} \chi_{\mathsf{Real}} &:= \chi_{\mathsf{Real}}(\zeta, \chi_{\mathsf{pk}}, \chi_{\mathsf{sm}}) := \{e^{\mathsf{sm}} + r_{\mathsf{aux}}\zeta \mid e^{\mathsf{sm}} \sim \chi_{\mathsf{sm}}, r_{\mathsf{aux}} \sim \chi_{\mathsf{pk}}\}, \text{ and} \\ \chi_{\mathsf{Atk}} &:= \chi_{\mathsf{Atk}}(s, e, \chi_{\mathsf{pk}}) := \{re + e_1 s + e_2 \mid r, e_1, e_2 \sim \chi_{\mathsf{pk}}\}. \end{aligned}$$

We show both $\mathsf{d\text{-}RLWE}_{s_j\sim\mathcal{U}(R_q)}(\chi_{\mathsf{Real}})$ in Eq. (25) and $\mathsf{d\text{-}RLWE}_{s_j\sim\mathcal{U}(R_q)}(\chi_{\mathsf{Atk}})$ in Eq. (26) are hard. It is easy to see that we have $\chi_{\mathsf{pk}} \leq \chi_{\mathsf{Atk}}$ (Definition 3.2) because $e_2 \sim \chi_{\mathsf{pk}}$. In addition, $\mathsf{d\text{-}RLWE}_{s_j\sim\mathcal{U}(R_q)}(n, q, \chi_{\mathsf{pk}})$ is hard by hypothesis

and Lemma 5.6. Hence, $\mathsf{d\text{-}RLWE}_{s_j\sim\mathcal{U}(R_q)}(\chi_{\mathsf{Atk}})$ is hard by Corollary 5.8. We have $\chi_{\mathsf{sm}} \leq \chi_{\mathsf{Real}}$ because $e_{\mathsf{sm}} \leftarrow \chi_{\mathsf{sm}}$ is sampled independently on $r_{\mathsf{aux}}\zeta$. In addition, $\mathsf{d\text{-}RLWE}_{s_j\sim\mathcal{U}(R_q)}(n, q, \chi_{\mathsf{sm}})$ is hard by hypothesis and Lemma 5.6. Thus, $\mathsf{d\text{-}RLWE}_{s_j}(\chi_{\mathsf{Real}})$ is also hard by Lemma 3.11, even though r_{aux} is known to the adversary (and ζ is possibly dependent on e as in subsequent Theorem 5.14).

Therefore, from Corollary 5.11 and Eq. (27), we have

$$(a', \mathsf{Real}_j, \mathsf{Atk}_j) \overset{\mathsf{comp}}{\approx} \mathcal{V} := \left\{ (a', u, v) \;\middle|\; \begin{array}{l} u, v \sim \mathcal{U}(R_q), \\ u - v = e_j^{\mathsf{sm}} + r_{\mathsf{aux}}\zeta - re \end{array} \right\}.$$

Because $p_j = \mathsf{Real}_j - r_{\mathsf{aux}}\zeta^{\mathsf{mal}}$ by Eq. (24), we also have

$$(a', p_j, \mathsf{Atk}_j - r_{\mathsf{aux}}\zeta^{\mathsf{mal}}) \overset{\mathsf{comp}}{\approx} \mathcal{V}' := \left\{ (a', u', v') \;\middle|\; \begin{array}{l} u', v' \sim \mathcal{U}(R_q), \\ u' - v' = e_j^{\mathsf{sm}} + r_{\mathsf{aux}}\zeta - re \end{array} \right\}.$$

The rest of the proof is identical to that of Theorem 4.5. □

5.5 Instantiation: ThPKE without Known-covariance Ring-LWE

Finally, we show the counterpart of Theorem 4.6: we show that there exist parameters and distributions that satisfy SS and correctness:

Theorem 5.14. *Let $\chi_{\mathsf{pk}} = D_{\mathbb{Z}^n, s_{\mathsf{pk}}}$ and $\chi_{\mathsf{sm}} = D_{\mathbb{Z}^n, s_{\mathsf{sm}}}$. We denote by $\mathbf{\Sigma}_\zeta$, $\mathbf{\Sigma}_e$ and $\mathbf{\Sigma}_s$ the coefficient Gram matrices (Definition 5.3) of $\zeta, e, s \in \mathcal{R}_q$, respectively. Let $\beta_{\mathsf{pub}} \in \mathcal{R}$ be a fixed public polynomial whose coefficient Gram matrix $\mathbf{\Sigma}_{\beta_{\mathsf{pub}}}$ satisfies $\mathbf{\Sigma}_{\beta_{\mathsf{pub}}} \succ 2(\mathbf{\Sigma}_e + \mathbf{\Sigma}_s)$ for any $s, e \leftarrow \chi_{\mathsf{pk}}$, and $\mathrm{tr}(\mathbf{\Sigma}_{\beta_{\mathsf{pub}}}) = \mathrm{poly}(\lambda)$. Select parameters N, n, q, $s_{\mathsf{pk}} \geq \tilde{\eta}_\epsilon^+(\mathbb{Z}^n)$, $s_{\mathsf{sm}} \geq \tilde{\eta}_\epsilon^+(\mathbb{Z}^n)$ such that $\sigma_{\max}(\mathbf{S})\sqrt{n} < \lfloor \frac{q}{4} \rfloor$, where $\sigma_{\max}(\mathbf{S})$ denotes the largest singular value of $\mathbf{S} := \sqrt{(Ns_{\mathsf{sm}}^2 + s_{\mathsf{pk}}^2)\mathbf{I}_n + s_{\mathsf{pk}}^2 \mathbf{\Sigma}_{\beta_{\mathsf{pub}}}}$. Conditioned on fixed $s, e \leftarrow \chi_{\mathsf{pk}}$ generated by KeyGen, we define $\chi_{\mathsf{err}} := \chi_{\mathsf{err}}(s, e, \mathbf{\Sigma}_{\beta_{\mathsf{pub}}})$ as a distribution over $\{\zeta \in R_q \mid \mathbf{\Sigma}_\zeta = \mathbf{\Sigma}_{\beta_{\mathsf{pub}}} - \mathbf{\Sigma}_e - \mathbf{\Sigma}_s\}$.*

Assume that both $\mathsf{d\text{-}RLWE}_s(n, q, D_{\mathbb{Z}^n, s_{\mathsf{pk}}})$ and $\mathsf{d\text{-}RLWE}_s(n, q, D_{\mathbb{Z}^n, s_{\mathsf{sm}}})$ are hard. Then, Algorithm 4 instantiated (and modified) as above satisfies SS (Definition 4.4) and correctness (Definition 4.1).

Proof. By subsequent Lemma 5.15, we obtain $\chi_{\mathsf{Sim}}(s, e, \zeta) = \chi_{\mathsf{Sim},1}(s, e, \zeta) \overset{\mathsf{stat}}{\approx} D_{\mathbb{Z}^n, \sqrt{(s_{\mathsf{sm}}^2 + s_{\mathsf{pk}}^2)\mathbf{I}_n + s_{\mathsf{pk}}^2 \mathbf{\Sigma}_{\beta_{\mathsf{pub}}}}}$, which has no information about e nor s. Note that there exist efficient (PPT) algorithms to sample ellipsoid discrete Gaussian, e.g, [21, 23, 26, 31, 36, 38]. Hence, we can prove SS by Theorem 5.13.

Let $x \leftarrow \chi_{\mathsf{Sim},t}(s, e, \zeta)$ for any $t \leq N$ and denote its coefficient vector by $\mathbf{x}$. Then, we have $\|x\|_\infty = \|\mathbf{x}\|_\infty \leq \|\mathbf{x}\| \leq \sigma_n(\mathbf{S})\sqrt{n}$ with overwhelming probability by Lemma 2.16. Thus, correctness holds by Lemma 5.12. (Note that we have $\sigma_{\max}(\mathbf{S}) \leq \|\mathbf{S}\|_F = \sqrt{n(Ns_{\mathsf{sm}}^2 + s_{\mathsf{pk}}^2) + s_{\mathsf{pk}}^2 \mathrm{tr}(\mathbf{\Sigma}_{\beta_{\mathsf{pub}}})} = \mathrm{poly}(\lambda)$; thus, we can select $q = \mathrm{poly}(\lambda)$.) □

We prove the deferred Lemma 5.15: we analyze the distribution of $\chi_{\mathsf{Sim},t}$ defined in Eq. (20) when we instantiate $\chi_{\mathsf{sm}} := D_{\mathbb{Z}^n,s_{\mathsf{sm}}}$ and $\chi_{\mathsf{pk}} := D_{\mathbb{Z}^n,s_{\mathsf{pk}}}$:

Lemma 5.15. *Let $\chi_{\mathsf{sm}} := D_{\mathbb{Z}^n,s_{\mathsf{sm}}}$ and $\chi_{\mathsf{pk}} := D_{\mathbb{Z}^n,s_{\mathsf{pk}}}$ for $s_{\mathsf{sm}}, s_{\mathsf{pk}} \geq \tilde{\eta}^+_\epsilon(\mathbb{Z}^n)$. Let $\mathbf{\Sigma}_\zeta, \mathbf{\Sigma}_e$ and $\mathbf{\Sigma}_s$ are the coefficient Gram matrices (Definition 5.3) of $\zeta, e, s \in \mathcal{R}_q$, respectively. Then, for χ_{Sim} defined in Eq. (20), we have:*

$$\chi_{\mathsf{Sim},t}(s,e,\zeta) \overset{\mathsf{stat}}{\approx} D_{\mathbb{Z}^n,\sqrt{(ts_{\mathsf{sm}}^2+s_{\mathsf{pk}}^2)\mathbf{I}_n+s_{\mathsf{pk}}^2(\mathbf{\Sigma}_\zeta+\mathbf{\Sigma}_e+\mathbf{\Sigma}_s)}}.$$

Proof. By definition of $\chi_{\mathsf{Sim},t}$, Definition 5.3, and Fact 5.5, we have

$$\begin{aligned}\chi_{\mathsf{Sim},t}(s,e,\zeta) &:= \left\{ \sum_{i=1}^t e_i^{\mathsf{sm}} + r_{\mathsf{aux}}\zeta - re - e_1 s - e_2 \;\middle|\; \begin{array}{l} e_1^{\mathsf{sm}},\ldots,e_t^{\mathsf{sm}} \overset{\mathrm{iid}}{\sim} D_{\mathbb{Z}^n,s_{\mathsf{sm}}} \\ r, r_{\mathsf{aux}}, e_1, e_2 \overset{\mathrm{iid}}{\sim} D_{\mathbb{Z}^n,s_{\mathsf{pk}}} \end{array} \right\} \\ &= \{\textstyle\sum_{i=1}^t \mathbf{e}_i^{\mathsf{sm}} + \mathbf{Z}\mathbf{r}_{\mathsf{aux}} - \mathbf{E}\mathbf{r} - \mathbf{S}\mathbf{e}_1 - \mathbf{e}_2\},\end{aligned}$$

where $\mathbf{Z}, \mathbf{E}, \mathbf{S}$ are the coefficient matrices of ζ, e, s and $\mathbf{e}_i^{\mathsf{sm}}, \mathbf{r}_{\mathsf{aux}}, \mathbf{r}, \mathbf{e}_1, \mathbf{e}_2$ are the coefficient vectors of $e_i^{\mathsf{sm}}, r_{\mathsf{aux}}, r, e_1, e_2$. By Lemma 2.14, we have $\mathbf{Z}\mathbf{r}_{\mathsf{aux}} \overset{\mathsf{stat}}{\approx} D_{\mathbb{Z}^n,s_{\mathsf{pk}}\mathbf{Z}}$, $\mathbf{E}\mathbf{r} \overset{\mathsf{stat}}{\approx} D_{\mathbb{Z}^n,s_{\mathsf{pk}}\mathbf{E}}$, and $\mathbf{S}\mathbf{e}_1 \overset{\mathsf{stat}}{\approx} D_{\mathbb{Z}^n,s_{\mathsf{pk}}\mathbf{S}}$. Then, we complete the proof by Lemma 2.13 and Lemma 2.15. □

6 Conclusion and Future Works

In this paper, we proposed efficient ThPKE schemes whose simulation-security are (directly) reduced from LWE or Ring-LWE with a polynomial modulus q. We introduced a core technique that we call "error sharing" to prevent leakage of the norm or covariance of the error (and secret) in the public key. In our schemes, we use the shares of a small error $\mathsf{err} := \boldsymbol{\zeta}$ distributed with secret sharing to mask the partial decryptions in addition to the conventional "smudging noise". Using this technique, we improved the ThPKE schemes proposed in [35] by eliminating the need to use "known-norm LWE" or "known-covariance Ring-LWE", which are nonstandard problems.

Our scheme can be easily extended to ThFHE by replacing the underlying PKE in ThPKE with FHE. Thus, we can construct an efficient SS-ThFHE from (Ring-)LWE with a polynomial modulus q. This implies that the applications of ThFHE can also be improved. For example, the round optimal MPC [5, 20, 22] and the universal thresholdizer [9] with simulation-security can be constructed from (Ring-)LWE with a polynomial modulus q. Additionally, the universal thresholdizer can be used to construct many threshold cryptosystems such as CCA-secure ThPKE, threshold signature, threshold PRF and threshold functional encryption. It is a future work to provide the specific details of constructing ThFHE from our ThPKE.

References

[1] Agrawal, S., Gentry, C., Halevi, S., Sahai, A.: Discrete Gaussian leftover hash lemma over infinite domains. In: Sako, K., Sarkar, P. (eds.) ASIACRYPT 2013. pp. 97–116. Springer (2013). https://doi.org/10.1007/978-3-642-42033-7_6

[2] Agrawal, S., Libert, B., Stehlé, D.: Fully secure functional encryption for inner products, from standard assumptions. In: Robshaw, M., Katz, J. (eds.) CRYPTO 2016. pp. 333–362. Springer (2016). https://doi.org/10.1007/978-3-662-53015-3_12

[3] Alperin-Sheriff, J., Peikert, C.: Circular and KDM security for identity-based encryption. In: Fischlin, M., Buchmann, J., Manulis, M. (eds.) PKC 2012. pp. 334–352. Springer (2012). https://doi.org/10.1007/978-3-642-30057-8_20

[4] Asharov, G., Jain, A., López-Alt, A., Tromer, E., Vaikuntanathan, V., Wichs, D.: Multiparty computation with low communication, computation and interaction via threshold FHE. In: Pointcheval, D., Johansson, T. (eds.) EUROCRYPT 2012. pp. 483–501. Springer (2012). https://doi.org/10.1007/978-3-642-29011-4_29

[5] Badrinarayanan, S., Jain, A., Manohar, N., Sahai, A.: Secure MPC: Laziness leads to GOD. In: Moriai, S., Wang, H. (eds.) ASIACRYPT 2020. pp. 120–150. Springer (2020). https://doi.org/10.1007/978-3-030-64840-4_5

[6] Bai, S., Lepoint, T., Roux-Langlois, A., Sakzad, A., Stehlé, D., Steinfeld, R.: Improved security proofs in lattice-based cryptography: Using the Rényi divergence rather than the statistical distance. J. Cryptol. **31**(2), 610–640 (2018). https://doi.org/10.1007/s00145-017-9265-9

[7] Bendlin, R., Damgård, I.: Threshold decryption and zero-knowledge proofs for lattice-based cryptosystems. In: Micciancio, D. (ed.) TCC 2010. pp. 201–218. Springer (2010). https://doi.org/10.1007/978-3-642-11799-2_13

[8] Boneh, D., Gennaro, R., Goldfeder, S., Jain, A., Kim, S., Rasmussen, P.M.R., Sahai, A.: Threshold cryptosystems from threshold fully homomorphic encryption. Cryptology ePrint Archive, Paper 2017/956 (2017), https://eprint.iacr.org/2017/956, (full version of [9])

[9] Boneh, D., Gennaro, R., Goldfeder, S., Jain, A., Kim, S., Rasmussen, P.M.R., Sahai, A.: Threshold cryptosystems from threshold fully homomorphic encryption. In: Shacham, H., Boldyreva, A. (eds.) CRYPTO 2018. pp. 565–596. Springer (2018). https://doi.org/10.1007/978-3-319-96884-1_19

[10] Boudgoust, K., Scholl, P.: Simple threshold (fully homomorphic) encryption from LWE with polynomial modulus. In: Asiacrypt 2023 (2023)

[11] Brakerski, Z., Döttling, N.: Hardness of LWE on general entropic distributions. In: Canteaut, A., Ishai, Y. (eds.) EUROCRYPT 2020. pp. 551–575. Springer (2020). https://doi.org/10.1007/978-3-030-45724-2_19

[12] Brakerski, Z., Gentry, C., Vaikuntanathan, V.: (Leveled) fully homomorphic encryption without bootstrapping. In: ITCS 2012. pp. 309–325. ACM (2012). https://doi.org/10.1145/2090236.2090262

[13] Brakerski, Z., Langlois, A., Peikert, C., Regev, O., Stehlé, D.: Classical hardness of learning with errors. In: STOC '13. p. 575–584. ACM (2013). https://doi.org/10.1145/2488608.2488680

[14] Brandão, L.T., Peralta, R.: NISTIR 8214C ipd: NIST first call for multi-party threshold schemes (initial public draft) (2023), https://doi.org/10.6028/NIST.IR.8214C.ipd

[15] Canetti, R.: Universally composable security: A new paradigm for cryptographic protocols. In: FOCS' 01. pp. 136–145 (2001). https://doi.org/10.1109/SFCS.2001.959888

[16] Cheon, J.H., Kim, A., Kim, M., Song, Y.: Homomorphic encryption for arithmetic of approximate numbers. In: Takagi, T., Peyrin, T. (eds.) ASIACRYPT 2017. pp. 409–437. Springer (2017). https://doi.org/10.1007/978-3-319-70694-8_15

[17] Chillotti, I., Gama, N., Georgieva, M., Izabachène, M.: TFHE: Fast fully homomorphic encryption over the torus. J. Cryptol. **33**(1), 34–91 (2020). https://doi.org/10.1007/s00145-019-09319-x
[18] Chowdhury, S., Sinha, S., Singh, A., Mishra, S., Chaudhary, C., Patranabis, S., Mukherjee, P., Chatterjee, A., Mukhopadhyay, D.: Efficient threshold FHE with application to real-time systems. ePrint 2022/1625 (2022), https://eprint.iacr.org/2022/1625
[19] Dahl, M., Demmler, D., El Kazdadi, S., Meyre, A., Orfila, J.B., Rotaru, D., Smart, N.P., Tap, S., Walter, M.: Noah's ark: Efficient threshold-FHE using noise flooding. In: WAHC '23. p. 35–46. ACM (2023). https://doi.org/10.1145/3605759.3625259
[20] Dov Gordon, S., Liu, F.H., Shi, E.: Constant-round MPC with fairness and guarantee of output delivery. In: Gennaro, R., Robshaw, M. (eds.) CRYPTO 2015. pp. 63–82. Springer (2015). https://doi.org/10.1007/978-3-662-48000-7_4
[21] Ducas, L., Galbraith, S., Prest, T., Yu, Y.: Integral matrix Gram root and lattice Gaussian sampling without floats. In: Canteaut, A., Ishai, Y. (eds.) EUROCRYPT 2020. pp. 608–637. Springer (2020). https://doi.org/10.1007/978-3-030-45724-2_21
[22] Garg, S., Gentry, C., Halevi, S., Raykova, M.: Two-round secure MPC from indistinguishability obfuscation. In: Lindell, Y. (ed.) TCC 2014. pp. 74–94. Springer (2014). https://doi.org/10.1007/978-3-642-54242-8_4
[23] Genise, N., Micciancio, D.: Faster Gaussian sampling for trapdoor lattices with arbitrary modulus. In: Nielsen, J.B., Rijmen, V. (eds.) EUROCRYPT 2018. pp. 174–203. Springer (2018). https://doi.org/10.1007/978-3-319-78381-9_7
[24] Genise, N., Micciancio, D., Peikert, C., Walter, M.: Improved discrete Gaussian and subGaussian analysis for lattice cryptography. In: Kiayias, A., Kohlweiss, M., Wallden, P., Zikas, V. (eds.) PKC 2020. pp. 623–651. Springer (2020). https://doi.org/10.1007/978-3-030-45374-9_21
[25] Gentry, C.: Fully homomorphic encryption using ideal lattices. In: STOC '09. pp. 169–178. ACM (2009). https://doi.org/10.1145/1536414.1536440
[26] Gentry, C., Peikert, C., Vaikuntanathan, V.: Trapdoors for hard lattices and new cryptographic constructions. In: STOC '08. p. 197–206. ACM (2008). https://doi.org/10.1145/1374376.1374407
[27] Horn, R.A., Johnson, C.R.: Matrix Analysis. Cambridge University Press (1985). https://doi.org/10.1017/CBO9780511810817
[28] Kiltz, E., Lyubashevsky, V., Schaffner, C.: A concrete treatment of Fiat-Shamir signatures in the quantum random-oracle model. In: Nielsen, J.B., Rijmen, V. (eds.) EUROCRYPT 2018. pp. 552–586. Springer (2018). https://doi.org/10.1007/978-3-319-78372-7_18
[29] Lyubashevsky, V., Peikert, C., Regev, O.: On ideal lattices and learning with errors over rings. In: Gilbert, H. (ed.) EUROCRYPT 2010. pp. 1–23. Springer (2010). https://doi.org/10.1007/978-3-642-13190-5_1
[30] Micciancio, D., Mol, P.: Pseudorandom knapsacks and the sample complexity of LWE search-to-decision reductions. In: Rogaway, P. (ed.) CRYPTO 2011. pp. 465–484. Springer (2011). https://doi.org/10.1007/978-3-642-22792-9_26
[31] Micciancio, D., Peikert, C.: Trapdoors for lattices: Simpler, tighter, faster, smaller. In: Pointcheval, D., Johansson, T. (eds.) EUROCRYPT 2012. pp. 700–718. Springer (2012). https://doi.org/10.1007/978-3-642-29011-4_41
[32] Micciancio, D., Peikert, C.: Hardness of SIS and LWE with small parameters. In: Canetti, R., Garay, J.A. (eds.) CRYPTO 2013. pp. 21–39. Springer (2013). https://doi.org/10.1007/978-3-642-40041-4_2

[33] Micciancio, D., Regev, O.: Worst-case to average-case reductions based on Gaussian measures. In: FOCS '04. pp. 372–381 (2004). https://doi.org/10.1109/FOCS.2004.72

[34] Micciancio, D., Regev, O.: Worst-case to average-case reductions based on Gaussian measures. SIAM J. Comput. **37**(1), 267–302 (2007). https://doi.org/10.1137/S0097539705447360, full version of [33]

[35] Micciancio, D., Suhl, A.: Simulation-secure threshold PKE from LWE with polynomial modulus. ePrint 2023/1728 (2023), https://eprint.iacr.org/2023/1728

[36] Micciancio, D., Walter, M.: Gaussian sampling over the integers: Efficient, generic, constant-time. In: Katz, J., Shacham, H. (eds.) CRYPTO 2017. pp. 455–485. Springer (2017). https://doi.org/10.1007/978-3-319-63715-0_16

[37] O'Neill, A., Peikert, C., Waters, B.: Bi-deniable public-key encryption. In: Rogaway, P. (ed.) CRYPTO 2011. pp. 525–542. Springer (2011). https://doi.org/10.1007/978-3-642-22792-9_30

[38] Peikert, C.: An efficient and parallel Gaussian sampler for lattices. In: Rabin, T. (ed.) CRYPTO 2010. pp. 80–97. Springer (2010). https://doi.org/10.1007/978-3-642-14623-7_5

[39] Prest, T.: Sharper bounds in lattice-based cryptography using the Rényi divergence. In: Takagi, T., Peyrin, T. (eds.) ASIACRYPT 2017. pp. 347–374. Springer (2017). https://doi.org/10.1007/978-3-319-70694-8_13

[40] Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. J. ACM **56**(6) (2009). https://doi.org/10.1145/1568318.1568324