Generic Constructions of Compact and Tightly Selective-Opening Secure Public-key Encryption Schemes*

Jiaxin Pan [1] Benedikt Wagner [2,3] Runzhi Zeng [1]

September 5, 2023

[1] Department of Mathematical Sciences,
NTNU – Norwegian University of Science and Technology, Trondheim, Norway
jiaxin.pan@ntnu.no, runzhi.zeng@ntnu.no
[2] CISPA Helmholtz Center for Information Security, Saarbrücken, Germany
benedikt.wagner@cispa.de
[3] Saarland University, Saarbrücken, Germany

Abstract

We propose two generic constructions of public-key encryption (PKE) with tight simulation-based selective-opening security against chosen-ciphertext attacks (SIM-SO-CCA) in the random oracle model. Our constructions can be instantiated with a small constant number of elements in the ciphertext, ignoring smaller contributions from symmetric-key encryption. That is, they have compact ciphertexts. Furthermore, three of our instantiations have compact public keys as well.

Known (almost) tightly SIM-SO-CCA secure PKE schemes are due to the work of Lyu et al. (PKC 2018) and Libert et al. (Crypto 2017). They have either linear-size ciphertexts or linear-size public keys. Moreover, they only achieve almost tightness, namely, with security loss depending on the security parameter.

In contrast to them, our schemes are the *first* ones achieving both tight SIM-SO-CCA security and compactness. More precisely, our two generic constructions are:

From Pseudorandom KEM: Our first generic construction is from a key encapsulation mechanism (KEM) with pseudorandom ciphertexts against plaintext-checking attacks. Such a KEM can be constructed directly from the Strong Diffie-Hellman (StDH), Computational DH (CDH), and Decisional DH assumptions. Both their ciphertexts and public keys are compact. Their security loss is a small constant. Interestingly, our CDH-based construction is the first scheme achieving all these advantages based on a weak search assumption. Furthermore, we also give a generic construction of such a KEM, which yields an efficient tightly SIM-SO-CCA PKE from lattices.

From Lossy Encryption: Our second scheme is the well-known Fujisaki-Okamoto transformation. We show that it can turn a lossy encryption scheme into a tightly SIM-SO-CCA secure PKE. This transformation preserves both tightness and compactness of the underlying lossy encryption, which is in contrast to the non-tight proof of Heuer et al. (PKC 2015).

Keywords: Selective-opening security, public-key encryption, tight security, random oracle model.

*This article is the full version of an earlier article in ASIACRYPT 2022 [PZ22], and it has significantly improved the earlier one. More information is given in Section 1.3.

Contents

1 Introduction

Selective-opening (SO) security is a strong security notion for encryption schemes. It considers encryption security in the multi-challenge setting. More precisely, an adversary is given multiple challenge ciphertexts and it is allowed to corrupt some of them to get the corresponding randomness. SO security guarantees that even with this additional capability an adversary still cannot learn any information about the remaining 'unopened' messages. The motivation of constructing SO secure encryption is that deleting cryptographic secrets is hard and expensive in practice, and adversaries can break into a user's computer and reveal the randomness used for generating a ciphertext. In some scenarios, users may even be required to reveal the randomness to publicly verify their computation.

Definitions of Selective-Opening Security. Definitions for SO security come in two flavors. Namely, there are indistinguishability-based (IND-based) definitions (weak-IND-SO and full-IND-SO) [BHY09, BHK12], and there is a simulation-based (SIM-based) one (SIM-SO) [BHY09]. These two notions are not polynomial-time equivalent. The strong notion of SIM-SO security requires that the output of every SO adversary can be efficiently simulated by a simulator that sees only the opened messages. The SIM-SO notion is the most common one to study [LP15, HJKS15, HJR16, HP16, LLHG18], since it does not require the message distribution to be efficiently conditionally resamplable (cf. [BHY09]). Moreover, previous works showed that SIM-SO-CCA and full-IND-SO-CCA notions are the strongest SO security notions [BHK12, BDWY12, HJR16].

Tight Reductions. When we prove the security of a cryptographic scheme Π, we construct a reduction to show that breaking the security of Π implies breaking the underlying assumption Γ. For concrete security, we argue that if an adversary $\mathcal{A}$ has advantage ϵ against Π then we have another adversary $\mathcal{B}$ that breaks Γ with advantage $\epsilon' = \epsilon/L$. The factor L is called the security loss. A scheme is called tightly secure if L is a small constant, assuming that the running time of $\mathcal{A}$ is approximately the same as $\mathcal{B}$ (up to a constant factor). A tight reduction can give quantitatively higher guarantees than a loose one. From a more practical perspective, a tight reduction allows shorter key-length recommendations based on the best known attacks against the underlying assumption. This can potentially yield more efficient schemes. Currently, our community aims to reduce the cost for tight security and construct efficient and tightly secure cryptographic schemes (such as the signature scheme in [DGJL21]). Hence, efficient schemes with tight security are highly desired.

Our Goal: Compact PKE with Tight SIM-SO-CCA Security. In this work, we aim for efficient and tightly SIM-SO-CCA secure public-key encryption (PKE) schemes, with compact ciphertexts and public keys. Here, 'compact' means constant-size, and SIM-SO-CCA security refers to security against chosen-ciphertext attacks in the SIM-SO setting. Next, we summarize the state of the art for this goal.

(Almost) Tight, yet Non-Compact Schemes. While there are compact and tightly IND-CCA secure PKE schemes [GHK17, HLLG19], known tightly SIM-SO-CCA PKE schemes [LSSS17, LLHG18] are still non-compact wrt. either ciphertext size or public key size. Moreover, the security reductions in both schemes are not fully tight, but almost tight (in the terminology of [CW13]). Namely, the security loss depends on the message bit-length, which is polynomial in the security parameter. Although almost tightness is already interesting, our goal is to achieve a security loss with small constants, which was unknown even with random oracles.

To provide more details, the scheme of Lyu et al. [LLHG18] is a recent PKE scheme with tight SIM-SO-CCA security, and its ciphertexts consist of $\mathbf{O}(|m|)$ group elements, where $|m|$ is the bit-length of the message. In a nutshell, their construction is a generic construction that tightly turns a IND-CCA secure key encapsulation mechanism (KEM) to a SIM-SO-CCA secure PKE, and their technique is to encrypt the message "bit-by-bit". Hence, their resulting construction does not preserve the compactness of the underlying KEM in terms of ciphertext overhead. Namely, even if we instantiate it with a compact KEM, it cannot give us a compact PKE with tight SIM-SO-CCA. We note that such a bit-wise approach is used in many SIM-SO secure schemes [BHY09, FHKW10, LP15].

While the scheme of Libert et al. [LSSS17] has compact ciphertexts, its public keys are not compact. Besides the large public key, their encryption algorithm needs to homomorphically evaluate the evaluation

circuit of a PRF over GSW [GSW13] ciphertexts that encrypts a PRF key. This makes their scheme very inefficient.

Compact, yet Non-Tight Schemes. The work of Heuer et al. [HJKS15] is an exception to the bit-wise approach. It is the first work that proves SIM-SO-CCA security of practical PKE schemes, such as DHIES [ABR01], OAEP [BR95], and Fujisaki-Okamoto (FO) [FO13], in the random oracle model [BR93]. All these schemes have compact ciphertexts. However, their security reduction is not tight, due to a guessing strategy. For instance, their proof for the FO transformation loses a factor of $\mathbf{O}(\mu \cdot Q_h)$, where μ and Q_h are numbers of challenge ciphertexts and random oracle queries, respectively.

Finally, we stress that, even though there exist compact and (almost) tightly SIM-SO-CPA secure schemes from [BHY09, HJR16], it is not known how to transform them into SIM-SO-CCA by preserving its tightness and compactness. This is the case even in the random oracle model, given the non-tight bounds from the work of Heuer et al. [HJKS15].

1.1 Our Contribution

We construct the *first* compact PKE schemes with tight SIM-SO-CCA security in the random oracle model (ROM). More precisely, we propose two different generic constructions for SIM-SO-CCA secure PKE from pseudorandom key encapsulation mechanism (KEM) in the multi-challenge setting and lossy encryption schemes, respectively. Both constructions *preserve the compactness and tightness* of the underlying primitives. In particular, the three Diffie-Hellman-based instantiations of our first generic construction achieve tight SIM-SO-CCA security and compact ciphertexts and compact public keys at the same time. Table 1 compares our schemes with other known SO secure PKE schemes based on the Diffie-Hellman assumption.

Scheme	**Security**	**Ass.**	**Loss**	$\lvert\mathsf{pk}\rvert$	$\lvert\mathsf{m}\rvert$	$\lvert\mathsf{c}\rvert - \lvert\mathsf{m}\rvert$	**RO?**
BHY [BHY09]	IND-SO-CPA	DDH	1	$2\lvert\mathbb{G}\rvert$	$\lvert\mathbb{G}\rvert$	$\lvert\mathbb{G}\rvert$	No
HJR [HJR16]	SIM-SO-CPA	DDH	$\mathbf{O}(\ell)$	$(\ell+1)^2\lvert\mathbb{G}\rvert$	ℓ	$\lvert\mathbb{G}\rvert$	No
LLHG [LLHG18]	SIM-SO-CCA	DDH	$\mathbf{O}(\ell)$	$6\lvert\mathbb{G}\rvert$	ℓ	$3\ell\lvert\mathbb{G}\rvert$	No
DHIES proved in [HJKS15]	SIM-SO-CCA	StDH	$\mathbf{O}(\mu)$	$\lvert\mathbb{G}\rvert$	ℓ	$\lvert\mathbb{G}\rvert$	Yes
FO proved in [HJKS16]	SIM-SO-CCA	DDH	$\mathbf{O}(\mu Q_h)$	$\lvert\mathbb{G}\rvert$	ℓ	$\lvert\mathbb{G}\rvert$	Yes
$\mathsf{PKE}_{\mathsf{StDH}}$ (Figures 7 and 15)	SIM-SO-CCA	StDH	10	$\lvert\mathbb{G}\rvert$	ℓ	$2\lvert\mathbb{G}\rvert$	Yes
$\mathsf{PKE}_{\mathsf{TDH}}$ (Figures 7 and 18)	SIM-SO-CCA	CDH	10	$2\lvert\mathbb{G}\rvert$	ℓ	$2\lvert\mathbb{G}\rvert$	Yes
$\mathsf{PKE}_{\mathsf{DDH}}$ (Figures 7 and 19)	SIM-SO-CCA	DDH	10	$\lvert\mathbb{G}\rvert$	ℓ	$4\lvert\mathbb{G}\rvert$	Yes
FO_1 (Figure 30)	IND-SO-CCA	DDH	2	$2\lvert\mathbb{G}\rvert$	ℓ	$\lvert\mathbb{G}\rvert$	Yes
FO_2 (Figure 31)	SIM-SO-CCA	DDH	$\mathbf{O}(\ell)$	$(\ell+1)^2\lvert\mathbb{G}\rvert$	ℓ	$\lvert\mathbb{G}\rvert$	Yes

Table 1: Comparison of our constructions with other SO secure PKE schemes. We ignore schemes that are non-tight and significantly less efficient than ours. $\lvert\mathbb{G}\rvert$ is the bit-length of group $\mathbb{G}$. ℓ is the message bit-length, which is independent of the group size, and it can be any polynomial in the security parameter λ. μ and Q_h are numbers of challenge ciphertexts and random oracle queries, respectively. The SO security losses of DHIES and FO can be found in [HJKS15, Theorem 6] and [HJKS16, Theorem 6].

Generic Construction from Pseudorandom KEM. Our first generic construction PKE_1 (in Figure 7) is based on a KEM that has multi-challenge pseudorandomness under plaintext-checking attacks (mPR-PCA security) and an explainable ciphertext space. The tight security proof is done in the random oracle model. In a nutshell, mPR-PCA security requires the ciphertexts of a KEM to be pseudorandom even if an adversary is provided a plaintext-checking oracle $\textsc{Pco}_{\mathsf{pr}}(\mathsf{c}, \psi)$. This oracle returns 1 if and only if the decapsulation of c is ψ. Here (c, ψ) is not allowed to be a challenge ciphertext-key pair. Essentially, this notion is an extension of the PCA security in [OP01, CHJ+02]. Furthermore, our construction requires KEM's ciphertexts to be explainable. Namely, a ciphertext can be "obliviously" sampled without running the encryption algorithm

and the sampling randomness can be explained. This is the same as the notion of efficiently samplable and explainable domains in [FHKW10].

The underlying KEM can be constructed directly from the strong Diffie-Hellman (StDH) assumption [ABR01] (cf. $\mathtt{KEM}_{\mathsf{StDH}}$ in Figure 15). We then use the twinning technique from [CKS08] to remove the decision oracle in the StDH assumption and construct our second tight KEM (cf. $\mathtt{KEM}_{\mathsf{TDH}}$ in Figure 18) based on the twin DH (TDH) assumption. The TDH assumption is tightly implied by the standard computational DH (CDH) assumption. Hence, when combined with our generic construction PKE_1, this yields the first tightly SIM-SO-CCA secure PKE based on such a standard search assumption. Both schemes have very short ciphertexts and public keys. Concretely, there are two group elements in the ciphertext overhead for $\mathtt{PKE}_{\mathsf{StDH}}$ (instantiating PKE_1 with $\mathtt{KEM}_{\mathsf{StDH}}$) and $\mathtt{PKE}_{\mathsf{TDH}}$ (instantiating PKE_1 with $\mathtt{KEM}_{\mathsf{TDH}}$), and one element for $\mathtt{PKE}_{\mathsf{StDH}}$'s public key and two for $\mathtt{PKE}_{\mathsf{TDH}}$.

We also show that the decision oracle in the proof of $\mathtt{KEM}_{\mathsf{StDH}}$ can be removed using the decisional DH assumption. However, the resulting scheme $\mathtt{PKE}_{\mathsf{DDH}}$ has longer ciphertexts than the previous two, although it is still compact. All these schemes have small-constant security loss, compact ciphertexts, and compact public keys.

Finally, we show that a mPR-PCA secure KEM can be constructed generically and tightly from mPR-CPA secure public-key encryption in the ROM. The natural notion of mPR-CPA security states that ciphertexts are pseudorandom under chosen-plaintext attacks. We note that several well-known public-key encryption schemes achieve mPR-CPA security. For example, we show that Regev's scheme [Reg05] is tightly mPR-CPA secure, which yields an efficient lattice-based SIM-SO-CCA secure PKE, tightly. This affirmatively answers the open problem in our previous version [PZ22] about how to extend our approach to the lattice setting.

Generic Construction from Lossy Encryption. Our last contribution is to prove that a lossy encryption [BHY09] can be transformed to a PKE with tight SO security via the well-known Fujisaki-Okamoto (FO) transformation [FO13]. The transformation preserves the tightness (up to a small constant) and compactness of the underlying lossy encryption.

Roughly speaking, a lossy encryption scheme has normal and lossy keys. Under normal keys, the scheme behaves as a normal PKE. But under lossy keys, there exists an opener that can explain a ciphertext to any message by outputting the suitable randomness. An opener is not necessarily efficient. Especially, if the lossy encryption does not have an efficient opener (e.g., the BHY scheme [BHY09]), then we can only show tight IND-SO-CCA security of the FO transformation. However, if the lossy encryption has an efficient opener (e.g., the HJR scheme [HJR16]), then it yields tight SIM-SO-CCA security of the FO transformation.

Our result implies that tight IND-SO-CCA and SIM-SO-CCA security can be achieved from any assumption that has a suitable lossy encryption. For comparison, we implement our generic construction with DDH-based lossy encryption schemes from [BHY09, HJR16]. They both have only one group element in the ciphertext (cf. Table 1). Our proof for the FO transformation is compactness- and tightness-preserving. Hence, for SIM-SO-CCA security, since the HJR scheme has non-compact public keys, it is also the case for our scheme. Similarly, the HJR scheme has only almost tightness, so has ours. We suppose that the size of ciphertexts is more critical than that of public keys, since ciphertexts have to be sent frequently over the internet for each communication, while public keys are stored on a server and can be used for a long time. Similar to the first generic construction, we also implement our generic construction with lattice-based lossy encryption in Section 5.4.

Efficiency Comparison among Diffie-Hellman-based Schemes. We focus on schemes based the Diffie-Hellman assumptions and compare their efficiency. We instantiate our generic constructions with suitable DH assumptions. In Table 2 we estimate the concrete efficiency of ours and compare it with other known SO secure schemes. Our comparison ignores schemes that are non-tight and significantly less efficient than ours (e.g., [Hof12]). We estimate the efficiency of all schemes using the same NIST P256 curve. According to the corresponding security proofs, we also consider the security level achieved by those schemes.

Our schemes significantly reduce the cost for tight SIM-SO-CCA, compared to LLHG. Moreover, our schemes are comparable to the practical PKE schemes, such as FO and DHIES. For instance, our $\mathtt{FO}_2$ has the same ciphertext size, but it achieves a higher level of security, thanks to the tight security proof. Both $\mathtt{PKE}_{\mathsf{StDH}}$ and $\mathtt{PKE}_{\mathsf{TDH}}$ are comparable to DHIES.

Scheme	Security	Ass.	Bit Security	$\lvert\mathsf{pk}\rvert$	$\lvert\mathsf{m}\rvert$	$\lvert\mathsf{c}\rvert - \lvert\mathsf{m}\rvert$
BHY [BHY09]	IND-SO-CPA	DDH	128	64	32	32
HJR [HJR16]	SIM-SO-CPA	DDH	120	2113568	32	32
LLHG [LLHG18]	SIM-SO-CCA	DDH	120	192	32	24576
DHIES proved in [HJKS15]	SIM-SO-CCA	StDH	96	32	32	64
FO proved in [HJKS16]	SIM-SO-CCA	CDH	64	32	32	32
$\mathsf{PKE}_{\mathsf{StDH}}$ (Figures 7 and 15)	SIM-SO-CCA	StDH	124	32	32	96
$\mathsf{PKE}_{\mathsf{TDH}}$ (Figures 7 and 18)	SIM-SO-CCA	CDH	124	64	32	96
$\mathsf{PKE}_{\mathsf{DDH}}$ (Figures 7 and 19)	SIM-SO-CCA	DDH	124	32	32	160
FO_1 (Figure 30)	IND-SO-CCA	DDH	127	64	32	32
FO_2 (Figure 31)	SIM-SO-CCA	DDH	120	2113568	32	32

Table 2: Concrete security and efficiency comparison. All schemes are instantiated with P256, and we consider $\mu = 2^{32}$, $q_H = 2^{32}$, $|\mathsf{m}| = 32$ bytes, and the output length of hash is 32 bytes. We consider the concrete security loss in the "Bit Security". All sizes are in bytes.

Practical Relevance. When an RO-based scheme is implemented in practice, one would instantiate the RO with a hash function, such as SHA-3. For SIM-SO-CCA PKE schemes in the ROM (including the previous work of Heuer et al. [HJKS15] and ours), we should be more careful and pay extra attention to the impossibility result of Bellare et al. [BDWY12]. More precisely, it shows that if a PKE scheme is binding then it cannot be SIM-SO secure. In a nutshell, it uses the binding property to construct an adversary such that there is no appropriate simulator for SIM-SO security. Hence, in the programmable ROM, the work of Heuer et al. and our schemes can all bypass it, since they are not binding according to the definition in [BDWY12].

However, if one simply replaces the RO with, for instance, SHA-3, the situation becomes rather complex. For our construction from lossy encryption, it is not binding and the security results remain, since it uses lossy encryption and it allows us to generate encryption collisions. This is also the reason why [BDWY12] does not apply to lossy encryption schemes. For the scheme of Heuer et al. and our first generic constructions, they will become binding in this case. Hence, the impossibility result of Bellare et al. applies, and they cannot have SIM-SO-CCA security. But the attack in [BDWY12] does not imply an adversary breaking IND-SO security, which means the scheme of Heuer et al. and our first constructions can have IND-SO-CCA security. An alternative solution could be finding a suitable programmable hash function in the standard model to instantiate our first three direction constructions. We leave constructing compact and tight SIM-SO-CCA secure PKE in the standard model as an interesting open problem.

1.2 Technical Overview

We use the Diffie-Hellman-based scheme as an example to give intuition behind our two generic constructions.

Technical Goal: Openability and Tightness. Selective-opening security is difficult to achieve. This is because the simulator $\mathcal{S}$ has to be able to 'open' any challenge ciphertext by producing the corresponding message and randomness. An adversary can verify whether a ciphertext has been correctly opened using the public encryption algorithm. It is not entirely trivial how to provide this openability efficiently. During the security proof, the simulator needs to embed a problem instance into the unopened ciphertexts, since usually it cannot open a ciphertext with a problem instance. Even worse, achieving tightness introduces an additional layer of complexity, as this opening procedure should be done in a tight fashion.

The work of Heuer et al. provides efficient openability by reprogramming the random oracle (RO) and guessing one unopened ciphertext. Then, the reduction embeds a problem challenge into this unopened ciphertext. We recall Heuer et al.'s strategy [HJKS15] of proving DHIES as an example to illustrate the aforementioned challenges in achieving tight SIM-SO-CCA security. The work of Heuer et al. is also the

starting point of our work.

We consider the DHIES scheme with the one-time pad for symmetric encryption. Let $\mathbb{G} := \langle g \rangle$ be a cyclic group with order p, and $\mathsf{pk} := g^x$ be a public key. A ciphertext C of DHIES has the form

$$C := (R := g^r, \mathsf{d} := K \oplus \mathsf{m}, \mathsf{MAC}_k(R, \mathsf{d})),$$

where $(K, k) := H(R, \mathsf{pk}^r)$ and H is modeled as an RO. MAC_k produces a MAC tag using k.

To prove its SIM-SO-CCA security, we use the strong Diffie-Hellman (StDH) assumption which states that given a StDH instance $(X = g^x, Y)$ and oracle access to $\textsc{dhp}_X$, it is hard to compute Y^x. Here, $\textsc{dhp}_X(\hat{Y}, \hat{Z})$ outputs whether $\hat{Z} = \hat{Y}^x$. The reduction for SIM-SO-CCA security of DHIES first defines $\mathsf{pk} := X$ and guesses that the i^*-th ciphertext will not be opened ($i^* \xleftarrow{\$} [\mu]$). Then Y is embedded into C_{i^*} by $R_{i^*} := Y$. By using the $\textsc{dhp}_X$ oracle and the RO patching technique [HJKS15], the reduction simulates the whole security game without knowing the secret x. We can prove that the adversary cannot get any information about $(K_{i^*}, k_{i^*}) = H(Y, Y^x)$ unless it computes Y^x, which breaks the StDH assumption. Thus, d_{i^*} is uniformly random and independent of R_{i^*}. This idea will be generalized by the mPR-PCA security in our paper.

Unfortunately, the guessing step in above strategy impedes a tight security proof. Concretely, the security bound depends on the number of challenge ciphertexts. One may consider using the random self-reducibility of StDH and embedding a randomized instance into challenge ciphertext C_i as $R_i := Y \cdot g^{s_i}$ where $s_i \xleftarrow{\$} \mathbb{Z}_p$ (for all $i \in [\mu]$). However, after doing so, one cannot open any ciphertext, since the discrete logarithm of Y is unknown.

Our Solution I: DHIES with Double Randomness and Its Generalization. Our first solution is a direct improvement on the DHIES scheme by doubling the randomness R in the ciphertext.

More precisely, we modify the generation of ciphertexts in DHIES: Instead of sampling a single r, we firstly choose a random bit $b \xleftarrow{\$} \{0, 1\}$, and then we choose $r_b \xleftarrow{\$} \mathbb{Z}_p$ and $R_{1-b} \xleftarrow{\$} \mathbb{G}$ (without knowing R_{1-b}'s discrete logarithm). Our modified DHIES scheme has ciphertexts with form:

$$C = (R_0, R_1, \mathsf{d} = K \oplus \mathsf{m}, h(k, R_0, R_1, \mathsf{d})),$$

where $(K, k) := H(b, R_0, R_1, \mathsf{pk}^{r_b})$, H is an RO, and h is a collision-resistant hash function. We note that sampling a random group element without knowing its discrete logarithm can be done in many widely-used groups like a subgroup of $\mathbb{Z}_q^*$ where q is a safe prime and prime-order elliptic curves.

After this modification, a ciphertext can have two valid randomness, namely, (b, r_b, R_{1-b}) and $(1 - b, r_{1-b}, R_b)$, in the view of an adversary, by carefully programming the RO H. Based on this, our simulator can embed the StDH instances to all challenge ciphertexts and open any ciphertext. This property is generalized by the notion of explainable ciphertext spaces in our first generic construction.

Our Solution II: Lossy Encryption. The idea of having multiple valid randomness can also be implemented using a lossy encryption, since under its lossy keys a ciphertext can be explained to different messages. Based on this, we use the lossy encryption as a tool to revise the security proof for the Fujisaki-Okamoto transformation and give a tight proof for its SIM-SO-CCA security. Another view of our second solution is that we transform the lossy-encryption-based SIM-SO-CPA secure PKE to a SIM-SO-CCA secure one, tightly.

Open Problem. We leave constructing (almost) tightly SIM-SO-CCA secure PKE with compact ciphertexts and compact public keys in the standard model as an interesting open problem.

1.3 History of This Paper

A preliminary version of this paper was published at Asiacrypt 2022 [PZ22]. After the publication, we generalized our three direct constructions in [PZ22] with mPR-PCA security as in Section 3. Our previous direct constructions can be obtained by instantiating our generic construction in Section 3.2 with the direct Diffie-Hellman instantiations of mPR-PCA security in Section 3.3. Our new generic construction yields an efficient tightly SIM-SO-CCA secure PKE from lattices. This part of the work was done with Benedikt

Wagner, while he was visiting NTNU. Hence, he was invited to the author list. We also introduce the lattice-based lossy encryption in Section 5.4 that also gives us another lattice-based tight construction. All these together improve our preliminary version significantly. Given these new contributions, we rewrote the abstract and introduction accordingly.

2 Preliminaries

Let n be an integer. By $[n]$ we denote the set $\{1, ..., n\}$. Let $\mathcal{A}$ be an algorithm. If $\mathcal{A}$ is probabilistic, then $y \leftarrow \mathcal{A}(x)$ means that the variable y is assigned to the output of $\mathcal{A}$ on input x. If $\mathcal{A}$ is deterministic, then we write $y := \mathcal{A}(x)$. We write $\mathcal{A}^{\mathcal{O}}$ to indicate that $\mathcal{A}$ has access to oracle $\mathcal{O}$. By $\mathcal{A} \Rightarrow \mathsf{out}$ we denote the event that $\mathcal{A}$ outputs out. Unless we state it explicitly, all our algorithms are probabilistic polynomial-time (PPT) and all relations considered in this paper can be decided in polynomial time (namely, PPT relations). Further, all reductions have the same running time as the adversary, up to a constant factor. Therefore, we omit specifying running times in our theorems. Throughout this paper, λ is the security parameter. The terms such as 'PPT' and 'negligible' are defined wrt λ. Let $\mathcal{X}$ be a finite set, $x \xleftarrow{\$} \mathcal{X}$ means that x is sampled at uniformly random from $\mathcal{X}$.

GAMES. We use the code-based games [BR06] to define and prove security. We implicitly assume that Boolean flags are initialized to false, numerical types are initialized to 0, sets are initialized to $\emptyset$, while strings are initialized to the empty string ϵ. The term $\Pr[\mathbf{G}^{\mathcal{A}} \Rightarrow 1]$ denotes the probability that the final output $\mathbf{G}^{\mathcal{A}}$ of game $\mathbf{G}$ running an adversary $\mathcal{A}$ is 1. Let EVENT be an event. We write $\Pr[\text{EVENT} : \mathbf{G}]$ to denote the probability that EVENT occurs during the game $\mathbf{G}$.

RANDOM ORACLES. We use lazy sampling to simulate random oracles in this paper. Let $\mathcal{X}$ and $\mathcal{Y}$ be two finite sets and $H : \mathcal{X} \to \mathcal{Y}$ be a random oracle in a security game $\mathbf{G}$. During the simulation of $\mathbf{G}$, we use a list $\mathcal{L}_{\mathsf{H}}$ to record all query-response pairs of H. On query x, the game samples $y \xleftarrow{\$} \mathcal{Y}$, sets $\mathcal{L}_{\mathsf{H}}[x] := y$ (which means that now $H(x) = y$), and then returns y as the response. We say x has been queried, or simply $x \in \mathcal{L}_{\mathsf{H}}$, if and only if $\mathcal{L}_{\mathsf{H}}[x] = y$ for some $y \in \mathcal{Y}$. For $x \notin \mathcal{L}_{\mathsf{H}}$, we always have $\mathcal{L}_{\mathsf{H}}[x] = \perp \notin \mathcal{Y}$.

2.1 Diffie-Hellman Assumptions

Let $\mathbb{G}$ be an cyclic group with a generator g and prime order p. Let $X = g^x$ and $Y = g^y$ for some $x, y \in \mathbb{Z}_p$. The CDH value of X and Y is written as $\mathsf{cdh}(X, Y) := g^{xy}$. Here we assume that $(\mathbb{G}, g, p)$ is a public parameter.

Definition 2.1 (Multi-Instance DDH (mDDH)). We say the mDDH problem is hard on $\mathbb{G}$ if for any $\mathcal{A}$, the mDDH advantage of $\mathcal{A}$ against $\mathbb{G}$

$$\mathsf{Adv}^{\mathsf{mDDH}}_{\mathbb{G}}(\mathcal{A}) := \left| \Pr\left[\mathcal{A}(g_1, (g_0^{r_i}, g_1^{r_i})_{i\in[\mu]}) \Rightarrow 1\right] - \Pr\left[\mathcal{A}(g_1, (g_0^{r_i}, g_1^{r'_i})_{i\in[\mu]}) \Rightarrow 1\right] \right|$$

is negligible, where μ is the number of challenges, $g_0 := g$, $g_1 := g_0^{\omega}$ for some $\omega \xleftarrow{\$} \mathbb{Z}_p$, and $r_i, r'_i \xleftarrow{\$} \mathbb{Z}_p$ for some $i \in [\mu]$.

By the random self-reducibility of DDH [EHK+13], the mDDH assumption is tightly equivalent to DDH assumption (i.e., single-instance version of mDDH).

Definition 2.2 (Strong Diffie-Hellman (StDH) Problem [ABR01]). For a fixed $X \in \mathbb{G}$, let DHP_X be the gap oracle that given $(Y', Z') \in \mathbb{G}^2$ outputs whether $\mathsf{cdh}(X, Y') = Z'$ or not. We say the StDH problem is hard on $\mathbb{G}$ if for any $\mathcal{A}$, the StDH advantage of $\mathcal{A}$ against $\mathbb{G}$, $\mathsf{Adv}^{\mathsf{StDH}}_{\mathbb{G}}(\mathcal{A})$, is negligible, where

$$\mathsf{Adv}^{\mathsf{StDH}}_{\mathbb{G}}(\mathcal{A}) := \Pr\left[(X, Y) \xleftarrow{\$} \mathbb{G}^2, \mathcal{A}^{\text{DHP}_X(\cdot,\cdot)}(X, Y) \Rightarrow \mathsf{cdh}(X, Y)\right].$$

Definition 2.3 (Twin Diffie-Hellman (TDH) Problem [CKS08]). For fixed $X_0, X_1 \in \mathbb{G}$, let $2\text{DHP}_{X_0,X_1}$ be an oracle that on input $(Y', Z_0', Z_1') \in \mathbb{G}^3$, determines whether $\mathsf{cdh}(X_0, Y') = Z_0'$ and $\mathsf{cdh}(X_1, Y') = Z_1'$. We say the TDH problem is hard on $\mathbb{G}$ if for any $\mathcal{A}$, the TDH advantage of $\mathcal{A}$ against $\mathbb{G}$

$$\mathsf{Adv}^{\mathsf{TDH}}_{\mathbb{G}}(\mathcal{A}) := \Pr\left[\mathcal{A}^{2\text{DHP}_{X_0,X_1}(\cdot,\cdot,\cdot)}(X_0, X_1, Y) \Rightarrow (\mathsf{cdh}(X_0, Y), \mathsf{cdh}(X_1, Y))\right]$$

is negligible, where $X_0, X_1, Y \xleftarrow{\$} \mathbb{G}$.

The StDH and TDH problems can be extended to multi-instance versions.

Definition 2.4 (Multi-Instance StDH (mStDH)). Let μ be the number of instances. We say the mStDH problem is hard on $\mathbb{G}$ if for any $\mathcal{A}$, given $X, Y_1, ..., Y_\mu \xleftarrow{\$} \mathbb{G}$, the mStDH advantage of $\mathcal{A}$ against $\mathbb{G}$, $\mathsf{Adv}^{\mathsf{mStDH}}_{\mathbb{G}}(\mathcal{A})$, is negligible, where

$$\mathsf{Adv}^{\mathsf{mStDH}}_{\mathbb{G}}(\mathcal{A}) := \Pr\left[\mathcal{A}^{\text{DHP}_X(\cdot,\cdot)}(X, (Y_i)_{i\in[\mu]}) \Rightarrow \mathsf{cdh}(X, Y_i) \text{ for some } i \in [\mu]\right].$$

Definition 2.5 (Multi-Instance TDH (mTDH)). Let μ be the number of instances. We say the mTDH problem is hard on $\mathbb{G}$ if for any $\mathcal{A}$, given $X_0, X_1, Y_1, ..., Y_\mu \xleftarrow{\$} \mathbb{G}$, the mTDH advantage of $\mathcal{A}$ against $\mathbb{G}$, $\mathsf{Adv}^{\mathsf{mStDH}}_{\mathbb{G}}(\mathcal{A})$, is negligible, where

$$\mathsf{Adv}^{\mathsf{mTDH}}_{\mathbb{G}}(\mathcal{A}) := \Pr\Big[\mathcal{A}^{2\text{DHP}_{X_0,X_1}(\cdot,\cdot,\cdot)}(X_0, X_1, (Y_i)_{i\in[\mu]}) \Rightarrow (\mathsf{cdh}(X_0, Y_i), \mathsf{cdh}(X_1, Y_i)) \text{ for some } i \in [\mu]\Big].$$

Definition 2.6 (Multi-Instance CDH (mCDH), [GJ18, Theorem 1]). Let μ be the number of instances. We say the mCDH problem is hard on $\mathbb{G}$ if for any $\mathcal{A}$, given $X, Y_1, ..., Y_\mu \xleftarrow{\$} \mathbb{G}$, the mCDH advantage of $\mathcal{A}$ against $\mathbb{G}$, $\mathsf{Adv}^{\mathsf{mCDH}}_{\mathbb{G}}(\mathcal{A})$, is negligible, where

$$\mathsf{Adv}^{\mathsf{mCDH}}_{\mathbb{G}}(\mathcal{A}) := \Pr\Big[\mathcal{A}(X, (Y_i)_{i\in[\mu]}) \Rightarrow \mathsf{cdh}(X, Y_i) \text{ for some } i \in [\mu]\Big].$$

The mStDH and mTDH assumptions are tightly implied by the StDH and TDH assumption, respectively. This follows naturally from the random self-reducibility of the Diffie-Hellman assumption.

Lemma 2.7 ($\mathsf{StDH} \xrightarrow{\text{tight}} \mathsf{mStDH}$). *For any* mStDH *adversary* $\mathcal{A}$*, there exists an* StDH *adversary* $\mathcal{B}$ *such that* $\mathsf{Adv}^{\mathsf{mStDH}}_{\mathbb{G}}(\mathcal{A}) \leq \mathsf{Adv}^{\mathsf{StDH}}_{\mathbb{G}}(\mathcal{B})$.

Lemma 2.8 ($\mathsf{TDH} \xrightarrow{\text{tight}} \mathsf{mTDH}$). *For any* mTDH *adversary* $\mathcal{A}$*, there exists an* TDH *adversary* $\mathcal{B}$ *such that* $\mathsf{Adv}^{\mathsf{mTDH}}_{\mathbb{G}}(\mathcal{A}) \leq \mathsf{Adv}^{\mathsf{TDH}}_{\mathbb{G}}(\mathcal{B})$.

Proof. StDH $\implies$ mStDH: Given an mStDH adversary $\mathcal{A}_0$, we construct an StDH adversary $\mathcal{B}_0$ as follows: $\mathcal{B}_0$'s input is a StDH problem instance $(\mathcal{G}, X, Y)$, and it also has access to DHP_X. It needs to simulate a mStDH instance and DHP_X for $\mathcal{A}_0$. Let μ be the number of challenge. Figure 1 shows the construction of $\mathcal{B}_0$. If $\mathcal{A}_0$ output $\mathsf{cdh}(X, Y_{i^*})$ for some $i^* \in [\mu]$, then we have $\mathsf{cdh}(X, Y) = \mathsf{cdh}(X, Y_{i^*}) \cdot X^{-r_i}$. Therefore, $\mathsf{Adv}^{\mathsf{mStDH}}_{\mathsf{GGen}}(\mathcal{A}_0) \leq \mathsf{Adv}^{\mathsf{StDH}}_{\mathsf{GGen}}(\mathcal{B}_0)$.

TDH $\implies$ mTDH: The argument is similar to StDH $\implies$ mStDH. Given an mTDH adversary $\mathcal{A}_1$, we construct an TDH adversary $\mathcal{B}_1$ (in Figure 1). We have $\mathsf{Adv}^{\mathsf{mTDH}}_{\mathsf{GGen}}(\mathcal{A}_1) \leq \mathsf{Adv}^{\mathsf{TDH}}_{\mathsf{GGen}}(\mathcal{B}_1)$.

□

2.2 Lattices

Let $s > 0$ be a parameter. The discrete Gaussian distribution over $\mathbb{Z}$ with parameter s, denoted by $D_{\mathbb{Z},s}$, is defined to be the distribution proportional to $\rho_s(\mathbf{x}) := \exp(-\pi\|\mathbf{x}\|^2/s^2)$, restricted to $\mathbb{Z}$. We recall the LWE assumption [Reg05] and some well-known regularity lemmas and tail bounds about Gaussian distributions [MR04, GPV07].

$\underline{\mathcal{B}_0^{\mathrm{DHP}_X}(\mathcal{G}, X, Y)}$
01 **for** $i \in [\mu]$
02 $\quad r_i \xleftarrow{\$} \mathbb{Z}_p, Y_i := Y g^{r_i}$
03 $Z \xleftarrow{\$} \mathcal{A}_0^{\mathrm{DHP}_X}(X, Y_1, ..., Y_\mu)$
04 Finds $i^* \in [\mu]$
05 $\quad$ s.t. $\mathrm{DHP}_X(Y_{i^*}, Z) = 1$
06 $Z' := Z \cdot X^{-r_i}$
07 **return** Z'

$\underline{\mathcal{B}_1^{\mathrm{2DHP}_{X_0,X_1}}(\mathcal{G}, X_0, X_1, Y)}$
08 **for** $i \in [\mu]$
09 $\quad r_i \xleftarrow{\$} \mathbb{Z}_p, Y_i := Y g^{r_i}$
10 $(Z_0, Z_1) \xleftarrow{\$} \mathcal{A}_1^{\mathrm{2DHP}_{X_0,X_1}}(X_0, X_1, Y_1, ..., Y_\mu)$
11 Finds $i^* \in [\mu]$ s.t. $\mathrm{2DHP}_{X_0,X_1}(Y_{i^*}, Z_0, Z_1) = 1$
12 $Z_0' := Z_0 \cdot X_0^{-r_i}, Z_1' := Z_1 \cdot X_1^{-r_i}$
13 **return** Z'

Figure 1: Reductions in the proofs of Lemmata 2.7 and 2.8

Definition 2.9 (LWE Assumption). Let $n \in \mathbb{N}$ and $m \in \mathbb{N}$ be positive integers and q be a prime. Let χ be a distribution over $\mathbb{Z}$. All of these are implicitly parameterized by the security parameter λ. We say that the $\mathsf{LWE}_{n,m,q,\chi}$ assumption holds, if for every algorithm $\mathcal{B}$, the following advantage is negligible in λ:

$$\begin{aligned}\mathsf{Adv}^{\mathsf{LWE}_{n,m,q,\chi}}(\mathcal{B}) := |\Pr[\mathcal{B}(\mathbf{A}, \mathbf{b}) = 1 \mid \mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n\times m}, \mathbf{b} \xleftarrow{\$} \mathbb{Z}_q^m] \\ - \Pr[\mathcal{B}(\mathbf{A}, \mathbf{A}^\top \mathbf{s} + \mathbf{e}) = 1 \mid \mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n\times m}, \mathbf{s} \xleftarrow{\$} \mathbb{Z}_q^n, \mathbf{e} \leftarrow \chi^m]|.\end{aligned}$$

Lemma 2.10 *Consider positive integers $n, m \in \mathbb{N}$ and a prime q at least polynomial in n. Assume $m \geq 2n \log q$ and $s \geq \omega(\sqrt{\log m})$. Then, for all but a negligible fraction of all matrices $\mathbf{A} \in \mathbb{Z}_q^{n\times m}$ the distribution of $\mathbf{Ae}$ with $\mathbf{e} \leftarrow D_{\mathbb{Z},s}^m$ is within negligible statistical distance to the uniform distribution over $\mathbb{Z}_q^n$.*

Lemma 2.11 *For any $s \geq \omega(\sqrt{\log m})$, and $\mathbf{x} \leftarrow D_{\mathbb{Z},s}^m$, the probability that $\|\mathbf{x}\| > s\sqrt{m}$ is at most 2^{-m+1}.*

2.3 Public-Key Encryption

In this section, we recall the syntax of public-key encryption (PKE) and several security notions, including notions for selective opening security.

Definition 2.12 (Public-Key Encryption). A PKE scheme PKE consists of three algorithms $(\mathsf{KG}, \mathsf{Enc}, \mathsf{Dec})$ and a message space $\mathcal{M}$, a randomness space $\mathcal{R}$, and a ciphertext space $\mathcal{C}$. KG outputs a public and secret key pair $(\mathsf{pk}, \mathsf{sk})$. The encryption algorithm Enc, on input pk and a message $\mathsf{m} \in \mathcal{M}$, outputs a ciphertext $\mathsf{c} \in \mathcal{C}$. We also write $\mathsf{c} := \mathsf{Enc}(\mathsf{pk}, \mathsf{m}; r)$ to indicate the randomness $r \in \mathcal{R}$ explicitly. The decryption algorithm Dec, on input sk and a ciphertext c, deterministically outputs a message $\mathsf{m}' \in \mathcal{M}$ or a rejection symbol $\perp \notin \mathcal{M}$.

Correctness of PKE. Some of our PKE schemes do not have perfect correctness, and the correctness bound of PKE might depend on some computational bound, e.g., the collision bound of hash function and the maximal number of queries to random oracle. Following [HHK17], we use a game COR to define PKE correctness.

GAME $\mathsf{COR}_{\mathsf{PKE}}^{\mathcal{A}}$
01 $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{PKE.KG}$
02 $\mathsf{m} \leftarrow \mathcal{A}^{\mathcal{O}}(\mathsf{pk}, \mathsf{sk})$
03 $\mathsf{c} \leftarrow \mathsf{Enc}(\mathsf{pk}, \mathsf{m})$
04 **if** $\mathsf{Dec}(\mathsf{sk}, \mathsf{c}) \neq \mathsf{m}$: **return** 1
05 **return** 0

Figure 2: The COR game for a PKE scheme PKE and $\mathcal{A}$. $\mathcal{A}$ might have access to some oracle $\mathcal{O}$ (e.g., random oracles, decryption oracles). It depends on the specific reduction.

Definition 2.13 (PKE Correctness). Let $\mathsf{PKE} := (\mathsf{KG}, \mathsf{Enc}, \mathsf{Dec})$ be a PKE scheme with message space $\mathcal{M}$ and $\mathcal{A}$ be an adversary against PKE. The COR advantage of $\mathcal{A}$ is defined as

$$\mathsf{Adv}^{\mathsf{COR}}_{\mathsf{PKE}}(\mathcal{A}) := \Pr\left[\mathsf{COR}^{\mathcal{A}}_{\mathsf{PKE}} \Rightarrow 1\right],$$

where the COR game is defined in Figure 2. If there exists a constant δ such that for all adversary $\mathcal{A}$, $\mathsf{Adv}^{\mathsf{COR}}_{\mathsf{PKE}}(\mathcal{A}) \leq \delta$, then we say PKE is $(1-\delta)$-correct.

Multi-challenge PR-CPA Security. Let $\mathsf{PKE} := (\mathsf{KG}, \mathsf{Enc}, \mathsf{Dec})$ be a public-key encryption scheme with message space $\mathcal{M}$ and ciphertext space $\mathcal{C}$. We define mPR-CPA (multi-challenge pseudorandomness under chosen-plaintext attacks) security in Figure 3.

GAME $\mathsf{mPR\text{-}CPA}^{\mathcal{A},\mu}_{\mathsf{PKE},b}$

$(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KG}$
$(\mathbf{m}, \mathsf{st}) \leftarrow \mathcal{A}(\mathsf{pk})$
for $i \in [\mu]$:
  $\mathbf{c}_0[i] \leftarrow \mathsf{Enc}(\mathsf{pk}, \mathbf{m}[i])$
  $\mathbf{c}_1[i] \xleftarrow{\$} \mathcal{C}$
$b' \leftarrow \mathcal{A}(\mathsf{st}, \mathbf{c}_b)$
return b'

Figure 3: Security game $\mathsf{mPR\text{-}CPA}$ for PKE scheme PKE.

Definition 2.14 Let μ be the number of challenge ciphertexts and $\mathcal{A}$ be an adversary against PKE. Consider the games $\mathsf{mPR\text{-}CPA}^{\mathcal{A},\mu}_{\mathsf{PKE},b}$ $(b \in \{0,1\})$ defined in Figure 3. We define the mPR-CPA advantage function

$$\mathsf{Adv}^{\mathsf{mPR\text{-}CPA}}_{\mathsf{PKE}}(\mathcal{A}, \mu) := \left|\Pr\left[\mathsf{mPR\text{-}CPA}^{\mathcal{A},\mu}_{\mathsf{PKE},0} \Rightarrow 1\right] - \Pr\left[\mathsf{mPR\text{-}CPA}^{\mathcal{A},\mu}_{\mathsf{PKE},1} \Rightarrow 1\right]\right|.$$

PKE is mPR-CPA secure if $\mathsf{Adv}^{\mathsf{mPR\text{-}CPA}}_{\mathsf{PKE}}(\mathcal{A}, \mu)$ is negligible for any $\mathcal{A}$.

Selective Opening Security of PKE. Selective Opening (SO) security preserves confidentiality even if an adversary opens the randomnesses of some ciphertexts. We consider two types of SO security: Simulation-based SO security against Chosen-Ciphertext Attacks (SIM-SO-CCA, Definition 2.15) and the weak version of Indistinguishability-based SO security against Chosen-Ciphertext Attacks (IND-SO-CCA, Definition 2.16).

GAME $\mathsf{REAL\text{-}SO\text{-}CCA}^{\mathcal{A}}_{\mathsf{PKE}}$

$(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KG}$
$(\mathcal{M}_a, \mathsf{st}) \leftarrow \mathcal{A}_0^{\mathrm{Dec}}(\mathsf{pk})$
for $i \in [\mu]$:
  $\mathbf{m}[i] := \mathsf{m}_i \leftarrow \mathcal{M}_a$
  $r_i \leftarrow \mathcal{R}$
  $\mathbf{c}[i] := \mathsf{Enc}(pk, \mathsf{m}_i; r_i)$
$out \xleftarrow{\$} \mathcal{A}_1^{\mathrm{Open},\mathrm{Dec}}(\mathsf{st}, \mathbf{c})$
return $\mathsf{Rel}(\mathcal{M}_a, \mathbf{m}, I, out)$

Dec(c) // for $\mathsf{c} \notin \mathbf{c}$

$\mathsf{m} := \mathsf{Dec}(\mathsf{sk}, \mathsf{c})$
return m

GAME $\mathsf{IDEAL\text{-}SO\text{-}CCA}^{\mathcal{S}}_{\mathsf{PKE}}$

$(\mathcal{M}_a, st) \leftarrow \mathcal{S}_0$
for $i \in [\mu]$:
  $\mathbf{m}[i] := \mathsf{m}_i \xleftarrow{\$} \mathcal{M}_a$
  $\mathbf{m}''[i] := |\mathsf{m}_i|$
$out \leftarrow \mathcal{S}_1^{\mathrm{Open}}(\mathsf{st}, \mathbf{m}'')$
return $\mathsf{Rel}(\mathcal{M}_a, \mathbf{m}, I, out)$

Open(i) // $i \in [\mu]$

$I := I \cup \{i\}$
return (m_i, r_i) // $\mathsf{REAL\text{-}SO\text{-}CCA}_{\mathsf{PKE}}$
return m_i // $\mathsf{IDEAL\text{-}SO\text{-}CCA}_{\mathsf{PKE}}$

Figure 4: The SO security games for PKE schemes. $\mathcal{S}_1$ only learn the lengths of challenge messages m_i instead of the challenge ciphertexts.

Definition 2.15 (SIM-SO-CCA security). Let PKE be a PKE scheme with message space $\mathcal{M}$ and randomness space $\mathcal{R}$ and $\mathcal{A} := (\mathcal{A}_0, \mathcal{A}_1)$ be an adversary against PKE. Let μ be the number of challenge ciphertexts Let Rel be a relation. We consider two games defined in Figure 4, where $\mathcal{A}$ is run in $\mathsf{REAL\text{-}SO\text{-}CCA}_{\mathsf{PKE}}$ and a SO simulator $\mathcal{S} := (\mathcal{S}_0, \mathcal{S}_1)$ in $\mathsf{IDEAL\text{-}SO\text{-}CCA}_{\mathsf{PKE}}$. $\mathcal{M}_a$ is a distribution over $\mathcal{M}$ chosen by $\mathcal{A}_0$. We define the SIM-SO-CCA advantage function

$$\mathsf{Adv}^{\mathsf{SIM\text{-}SO\text{-}CCA}}_{\mathsf{PKE}}(\mathcal{A}, \mathcal{S}, \mu, \mathsf{Rel}) := \left| \Pr\left[\mathsf{REAL\text{-}SO\text{-}CCA}^{\mathcal{A}}_{\mathsf{PKE}} \Rightarrow 1\right] - \Pr\left[\mathsf{IDEAL\text{-}SO\text{-}CCA}^{\mathcal{S}}_{\mathsf{PKE}} \Rightarrow 1\right]\right|,$$

PKE is SIM-SO-CCA secure if, for every adversary $\mathcal{A}$ and every PPT relation Rel, there exists a simulator $\mathcal{S}$ such that $\mathsf{Adv}^{\mathsf{SIM\text{-}SO\text{-}CCA}}_{\mathsf{PKE}}(\mathcal{A}, \mathcal{S}, \mu, \mathsf{Rel})$ is negligible.

Definition 2.16 (IND-SO-CCA security). Let PKE be a PKE scheme with message space $\mathcal{M}$ and randomness space $\mathcal{R}$ and $\mathcal{A} := (\mathcal{A}_0, \mathcal{A}_1, \mathcal{A}_2)$ be an adversary against PKE. Let μ be the number of challenge ciphertext.

We consider the game defined in Figure 5. Samp and ReSamp are efficient algorithms output by $\mathcal{A}_0$, where Samp outputs μ messages according to some distribution (determined by $\mathcal{A}_0$) over $\mathcal{M}$, and $\mathsf{ReSamp}(I, \mathbf{m}_0)$ resamples $\mathbf{m}_0[i]$ for $i \notin I$ according to the same distribution of Samp and then outputs $\mathbf{m}_1$. For $i \in I$, $\mathbf{m}_0[i] = \mathbf{m}_1[i]$. We define the IND-SO-CCA advantage function

$$\mathsf{Adv}^{\mathsf{IND\text{-}SO\text{-}CCA}}_{\mathsf{PKE}}(\mathcal{A}, \mu) := \left|\Pr\left[\mathsf{IND\text{-}SO\text{-}CCA}^{\mathcal{A}}_{\mathsf{PKE},0} \Rightarrow 1\right] - \Pr\left[\mathsf{IND\text{-}SO\text{-}CCA}^{\mathcal{A}}_{\mathsf{PKE},1} \Rightarrow 1\right]\right|.$$

PKE is IND-SO-CCA secure if $\mathsf{Adv}^{\mathsf{IND\text{-}SO\text{-}CCA}}_{\mathsf{PKE}}(\mathcal{A}, \mu)$ is negligible for any $\mathcal{A}$.

GAME $\mathsf{IND\text{-}SO\text{-}CCA}^{\mathcal{A}}_{\mathsf{PKE},b}$

$(\mathsf{pk}, \mathsf{sk}) \xleftarrow{\$} \mathsf{KG}$
$(\mathsf{Samp}, \mathsf{ReSamp}, \mathsf{st}_0) \leftarrow \mathcal{A}_0(\mathsf{pk})$
$\mathbf{m}_0 \leftarrow \mathsf{Samp}$
for $i \in [\mu]$:
  $r_i \xleftarrow{\$} \mathcal{R}$
  $\mathbf{c}[i] := \mathsf{Enc}(pk, \mathbf{m}_0[i]; r_i)$
$\mathsf{st}_1 \leftarrow \mathcal{A}_1^{\textsc{Open}, \textsc{Dec}}(\mathbf{c}, \mathsf{st}_0)$
for $i \in [\mu] \backslash I$:
  $\mathbf{m}_1[i] := \mathsf{ReSamp}(I, \mathbf{m}_0)$
$b' \leftarrow \mathcal{A}_2^{\textsc{Dec}}(\mathsf{st}_1, \mathbf{m}_b)$
return b'

$\textsc{Dec}(\mathsf{c})$ // for $\mathsf{c} \notin \mathbf{c}$
$\mathsf{m} := \mathsf{Dec}(\mathsf{sk}, \mathsf{c})$
return m

$\textsc{Open}(i)$ // $i \in [\mu]$
$I := I \cup \{i\}$
return (m_i, r_i)

Figure 5: The SO security games for PKE schemes. $\mathcal{S}_1$ only learn the lengths of challenge messages m_i instead of the challenge ciphertexts. For $i \in I, \mathbf{m}_0[i] = \mathbf{m}_1[i]$, and for $i \in [\mu] \backslash I$, $\mathbf{m}_0[i]$ has the same distribution with $\mathbf{m}_1[i]$ but not necessary to be the same.

3 Generic Construction I: SO from mPR-PCA

This section is new to our proceedings version [PZ22], and it generates our direct constructions in [PZ22, Section 3] with a mPR-CPA secure KEM that has a explainable ciphertext space.

3.1 Definition of mPR-PCA secure KEM

In this section, we focus on multi-challenge pseudorandom key encapsulation mechanisms. This notion will be essential for our first generic construction. Here, we formally define the notion.

Definition 3.1 (Key Encapsulation Mechanism). A KEM KEM consists of three algorithms $(\mathsf{KG}, \mathsf{Encaps}, \mathsf{Decaps})$ and a ciphertext space $\mathcal{C}$, a randomness space $\mathcal{R}$, and a KEM key space Ψ. KG outputs a public and secret key pair $(\mathsf{pk}, \mathsf{sk})$. The encapsulation algorithm Encaps, on input pk, outputs a ciphertext $\mathsf{c} \in \mathcal{C}$. We also write $\mathsf{c} := \mathsf{Encaps}(\mathsf{pk}; r)$ to indicate the randomness $r \in \mathcal{R}$ explicitly. The decapsulation algorithm Decaps, on input sk and a ciphertext c, deterministically outputs a KEM key $\psi \in \Psi$ or a rejection symbol $\perp \notin \Psi$.

The correctness definition of KEM schemes is given in Definition 3.2. Here we do not use a game to define it because in this paper, all KEM constructions have statistically negligible correctness error.

Definition 3.2 (KEM Correctness). Let $\mathsf{KEM} := (\mathsf{KG}, \mathsf{Encaps}, \mathsf{Decaps})$ be a KEM scheme and $\mathcal{A}$ be an adversary against KEM. We say KEM is $(1-\delta)$-correct if

$$\Pr_{\substack{(\mathsf{pk},\mathsf{sk})\leftarrow\mathsf{KG}\\(\mathsf{c},\psi)\leftarrow\mathsf{Encaps}(\mathsf{pk})}}[\psi \neq \mathsf{Decaps}(\mathsf{sk},\mathsf{c})] \leq \delta.$$

Multi-challenge PR-PCA Security. Let $\mathsf{KEM} := (\mathsf{KG}, \mathsf{Encaps}, \mathsf{Decaps})$ be a KEM scheme with randomness space $\mathcal{R}$ and ciphertext space $\mathcal{C}$. We define mPR-PCA (multi-challenge pseudorandomness under plaintext-checking attacks) security for KEM in Definition 3.3.

GAME $\mathsf{mPR\text{-}PCA}^{\mathcal{A},\mu}_{\mathsf{KEM},b}$

01 $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KG}$
02 **for** $i \in [\mu]$:
03 $\quad (\mathbf{c}[i], \boldsymbol{\psi}[i]) \leftarrow \mathsf{Encaps}(\mathsf{pk})$ // $b = 0$
04 $\quad \mathbf{c}[i] \xleftarrow{\$} \mathcal{C}, \boldsymbol{\psi}[i] \xleftarrow{\$} \Psi$ // $b = 1$
05 $b' \leftarrow \mathcal{A}^{\mathrm{PCO}_{\mathsf{pr}}}(\mathsf{pk}, \mathbf{c}, \boldsymbol{\psi})$
06 **return** b'

Oracle $\mathrm{PCO}_{\mathsf{pr}}(\mathsf{c}, \psi \in \Psi)$

07 **if** $\exists i \in [\mu]$ s.t. $(\mathsf{c}, \psi) = (\mathbf{c}[i], \boldsymbol{\psi}[i])$
08 $\quad$ **return** $\perp$
09 **return** $\psi =_? \mathsf{Decaps}(\mathsf{sk}, \mathsf{c})$

Figure 6: Security game $\mathsf{mPR\text{-}PCA}$ for KEM scheme KEM.

Definition 3.3 (mPR-PCA security). Let μ be the number of challenge ciphertexts and $\mathcal{A}$ be an adversary against KEM. Consider the games $\mathsf{mPR\text{-}PCA}^{\mathcal{A},\mu}_{\mathsf{KEM},b}$ ($b \in \{0,1\}$) defined in Figure 6. We define the mPR-PCA advantage function

$$\mathsf{Adv}^{\mathsf{mPR\text{-}PCA}}_{\mathsf{KEM}}(\mathcal{A}, \mu) := \left|\Pr\left[\mathsf{mPR\text{-}PCA}^{\mathcal{A},\mu}_{\mathsf{KEM},0} \Rightarrow 1\right] - \Pr\left[\mathsf{mPR\text{-}PCA}^{\mathcal{A},\mu}_{\mathsf{KEM},1} \Rightarrow 1\right]\right|.$$

We say that KEM is mPR-PCA secure if $\mathsf{Adv}^{\mathsf{mPR\text{-}PCA}}_{\mathsf{KEM}}(\mathcal{A}, \mu)$ is negligible for any $\mathcal{A}$.

In this paper, we require the ciphertext spaces of KEM schemes to be explainable. The formal definition is given in Definition 3.4, which can be viewed as a special case of the definition of ESE domains in [LLHG18].

Definition 3.4 ($\mathcal{C}$-Explainable). Let PKE (resp., KEM) be a PKE (resp., KEM) scheme with ciphertext space $\mathcal{C}$. We say PKE (resp., KEM) is $\mathcal{C}$-explainable (or has explainable ciphertext space $\mathcal{C}$) if there exist two algorithms $\mathsf{Sample}_{\mathcal{C}}$ and $\mathsf{Sample}^{-1}_{\mathcal{C}}$ and a randomness domain $\mathcal{R}_{\mathsf{Sample}_{\mathcal{C}}}$ such that

- The algorithm $\mathsf{Sample}_{\mathcal{C}}$, on input $\hat{R}$, outputs an element from $\mathcal{C}$, such that the following distribution is the uniform distribution over $\mathcal{C}$:

$$\left\{\mathsf{Sample}_{\mathcal{C}}(\hat{R}) \;\middle|\; \hat{R} \xleftarrow{\$} \mathcal{R}_{\mathsf{Sample}_{\mathcal{C}}}\right\}.$$

- The algorithm $\mathsf{Sample}^{-1}_{\mathcal{C}}$, on input $x \in \mathcal{C}$, outputs an element $\hat{R}$ such that for any fixed $x \in \mathcal{C}$, the following distributions are the same:

$$\left\{\hat{R} \;\middle|\; \hat{R} \leftarrow \mathsf{Sample}^{-1}_{\mathcal{C}}(x)\right\} \text{ and } \left\{\hat{R} \;\middle|\; \hat{R} \xleftarrow{\$} \{\hat{R} \in \mathcal{R}_{\mathsf{Sample}_{\mathcal{C}}} \mid x = \mathsf{Sample}_{\mathcal{C}}(\hat{R})\}\right\}.$$

3.2 From mPR-PCA secure KEM to SO

Let $\mathsf{KEM} := (\mathsf{KG}, \mathsf{Encaps}, \mathsf{Decaps})$ be a key encapsulation mechanism scheme with randomness space $\mathcal{R}$, ciphertext space $\mathcal{C}$, and key space Ψ. We also let KEM be $\mathcal{C}$-explainable (cf. Definition 3.4) with two algorithms $(\mathsf{Sample}_{\mathcal{C}}, \mathsf{Sample}_{\mathcal{C}}^{-1})$. Let $H : \{0,1\} \times \mathcal{C}^2 \times \Psi \to \mathcal{M} \times \{0,1\}^l$ and $h : \{0,1\}^l \times \mathcal{C}^2 \to \{0,1\}^\ell$ be random oracles. We construct a PKE scheme $\mathsf{PKE}_1 := (\mathsf{KG}_1, \mathsf{Enc}_1, \mathsf{Dec}_1)$ with message space $\mathcal{M}$ based on KEM, where $\mathsf{KG}_1 := \mathsf{KG}$, Enc_1 and Dec_1 are shown in Figure 7.

$\mathsf{Enc}_1(\mathsf{pk}, \mathsf{m} \in \mathcal{M})$	$\mathsf{Dec}_1(\mathsf{sk}, (\mathsf{c}_0, \mathsf{c}_1, \mathsf{d}, \mathcal{T}))$
01 $b \xleftarrow{\$} \{0,1\}$	09 $\mathsf{m} := \bot$
02 $r_b \xleftarrow{\$} \mathcal{R}, \hat{R} \xleftarrow{\$} \mathcal{R}_{\mathsf{Sample}_{\mathcal{C}}}$	10 **for** $b \in \{0,1\}$:
03 $(\mathsf{c}_b, \psi_b) := \mathsf{Encaps}(\mathsf{pk}; r_b)$	11 $\quad \psi_b := \mathsf{Decaps}(\mathsf{sk}, \mathsf{c}_b)$
04 $\mathsf{c}_{1-b} := \mathsf{Sample}_{\mathcal{C}}(\hat{R})$	12 $\quad (K_b, k_b) := H(b, \mathsf{c}_0, \mathsf{c}_1, \psi_b)$
05 $(K, k) := H(b, \mathsf{c}_0, \mathsf{c}_1, \psi_b)$	13 $\quad \mathcal{T}_b := h(k_b, \mathsf{c}_0, \mathsf{c}_1, \mathsf{d})$
06 $\mathsf{d} := K \oplus \mathsf{m}$	14 $\quad$ **if** $\mathcal{T}_b = \mathcal{T} : \mathsf{m} := \mathsf{d} \oplus K_b$
07 $\mathcal{T} := h(k, \mathsf{c}_0, \mathsf{c}_1, \mathsf{d})$	15 **if** $\mathcal{T}_0 = \mathcal{T}_1 : \mathsf{m} := \bot$
08 **return** $(\mathsf{c}_0, \mathsf{c}_1, \mathsf{d}, \mathcal{T})$	16 **return** m

Figure 7: Our generic construction of SIM-SO-CCA secure PKE schemes $\mathsf{PKE}_1 := (\mathsf{KG}_1 = \mathsf{KG}, \mathsf{Enc}_1, \mathsf{Dec}_1)$.

Correctness. The correctness of PKE_1 is implied by the correctness of KEM and the collision bounds of h and H. More precisely, there are two kinds of decryption errors:

- KEM cannot decrypt a ciphertext correctly (cf. Definition 3.2).
- A PKE_1 ciphertext $(\mathsf{c}_0, \mathsf{c}_1, \mathsf{d}, \mathcal{T})$ is generated using b but $\mathcal{T}_b = \mathcal{T}_{1-b}$, where $\mathcal{T}_b = h(k_b, \mathsf{c}_0, \mathsf{c}_1, \mathsf{d})$ and $\mathcal{T}_{1-b} = h(k_{1-b}, \mathsf{c}_0, \mathsf{c}_1, \mathsf{d})$. In this case, Dec_1 outputs $\bot$. $\mathcal{T}_b = \mathcal{T}_{1-b}$ means
  - either $k_b = k_{1-b}$, then we have $(b, \mathsf{c}_0, \mathsf{c}_1, \psi_b) \neq (1-b, \mathsf{c}_0, \mathsf{c}_1, \psi_{1-b})$ and $H(b, \mathsf{c}_0, \mathsf{c}_1, \psi_b)[2] = k_b = k_{1-b} = H(1-b, \mathsf{c}_0, \mathsf{c}_1, \psi_{1-b})[2]$ (where "[2]" means the second half of the hash function output), which is a truncation collision pair of H[1],
  - or $h(k_b, \mathsf{c}_0, \mathsf{c}_1, \mathsf{d}) = h(k_{1-b}, \mathsf{c}_0, \mathsf{c}_1, \mathsf{d})$ and $k_b \neq k_{1-b}$, then we find a collision for h.

Hence, the correctness error $\mathsf{Adv}^{\mathsf{COR}}_{\mathsf{PKE}_1}(\mathcal{A})$ is the sum of KEM's decryption error, the probability of truncation collision for H, and the probability of collision for h. In this paper H and h are modeled as random oracles, and thus $\mathsf{Adv}^{\mathsf{COR}}_{\mathsf{PKE}_1}(\mathcal{A}) \leq \frac{q_h^2 + q_H^2}{2^\ell} + \delta_{\mathsf{KEM}}$ for any adversary $\mathcal{A}$, where δ_{KEM} is the error bound of KEM. In practice, we require h to have collision resistance and H to have truncation collision resistance.

Theorem 3.5 PKE_1 *in Figure 7 is SIM-SO-CCA secure (Definition 2.15) if H and h are modeled as random oracles and* KEM *is mPR-PCA secure and $\mathcal{C}$-Explainable. For any SIM-SO-CCA adversary $\mathcal{A}$ and relation* Rel*, there exists a simulator $\mathcal{S}$ and adversaries $\mathcal{B}_{\mathsf{PR}}$ and $\mathcal{A}_{\mathsf{hash}}$ such that:*

$$\mathsf{Adv}^{\mathsf{SIM\text{-}SO\text{-}CCA}}_{\mathsf{PKE}_1}(\mathcal{A}, \mathcal{S}, \mu, \mathsf{Rel}) \leq 5\mathsf{Adv}^{\mathsf{mPR\text{-}PCA}}_{\mathsf{KEM}}(\mathcal{B}_{\mathsf{PR}}, \mu) + \frac{5\mu q_H}{|\Psi|} + 2\left(\frac{\mu^2 + q_H^2}{|\mathcal{M}|} + \frac{\mu^2 + q_H^2 + q_h^2}{2^l}\right),$$

where q_H and q_h are the numbers of $\mathcal{A}$'s queries to H and h, respectively, and μ is the number of challenge ciphertexts.

Proof. The theorem is proved by the game sequences in Figures 8, 9 and 12. We assume that there is no collision among all K_i's, k_i's, and the outputs of random oracles. This adds $\frac{\mu^2+q_H^2}{|\mathcal{M}|} + \frac{\mu^2+q_H^2+q_h^2}{2^l}$ to our security bound, and we have

$$\left|\Pr\left[\mathsf{REAL\text{-}SO\text{-}CCA}^{\mathcal{A}}_{\mathsf{PKE}_1} \Rightarrow 1\right] - \Pr\left[\mathbf{G}_0^{\mathcal{A}} \Rightarrow 1\right]\right| \leq \frac{\mu^2 + q_H^2}{|\mathcal{M}|} + \frac{\mu^2 + q_H^2 + q_h^2}{2^l}.$$

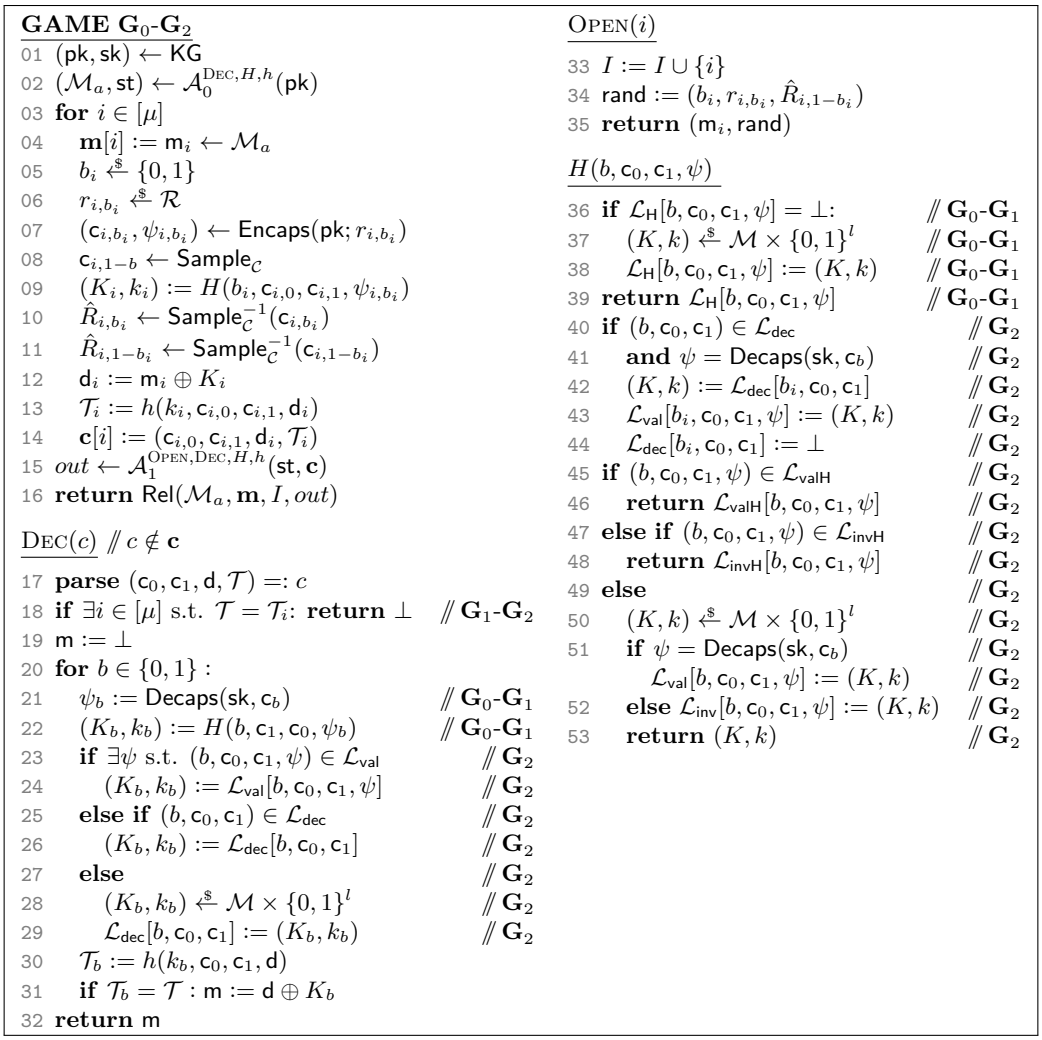

Figure 8: Games $\mathbf{G}_0$-$\mathbf{G}_2$ for proving Theorem 3.5. The random oracle h is simulated in the standard way, so here we ignore the details.

Game $\mathbf{G}_1$: We modify the DEC oracle. When $\mathcal{A}$ queries DEC on $c = (\mathsf{c}_0, \mathsf{c}_1, \mathsf{d}, \mathcal{T})$, where $\mathcal{T}$ is the hash value of one of the challenge ciphertexts (i.e., $\mathcal{T} = \mathcal{T}_i$ for some $i \in [\mu]$), then DEC returns $\perp$.

$\mathcal{A}$ notices this change if it queries DEC on $(\mathsf{c}_0, \mathsf{c}_1, \mathsf{d}, \mathcal{T})$ where $(\mathsf{c}_0, \mathsf{c}_1, \mathsf{d}, \mathcal{T}) \notin \mathbf{c}$, $\mathcal{T} = \mathcal{T}_i (i \in [\mu])$, and $\textsc{Dec}(\mathsf{c}_0, \mathsf{c}_1, \mathsf{d}, \mathcal{T}) \neq \perp$. For such $(\mathsf{c}_0, \mathsf{c}_1, \mathsf{d}, \mathcal{T})$, by the definition of DEC, we have $(\mathsf{c}_0, \mathsf{c}_1, \mathsf{d}) \neq (\mathsf{c}_{i,0}, \mathsf{c}_{i,1}, \mathsf{d}_i)$ and $\mathcal{T}_i = \mathcal{T} = h(k', \mathsf{c}_0, \mathsf{c}_1, \mathsf{d})$ where k' is either k_0 or k_1. Here we know the secret key sk and thus can check which of the previous equations holds by computing k_0 and k_1. So, if $\mathcal{A}$ queries such $(\mathsf{c}_0, \mathsf{c}_1, \mathsf{d}, \mathcal{T})$, we have $h(k', \mathsf{c}_0, \mathsf{c}_1, \mathsf{d}) = h(k_i, \mathsf{c}_{i,0}, \mathsf{c}_{i,1}, \mathsf{d}_i)$ and $((k', \mathsf{c}_0, \mathsf{c}_1, \mathsf{d}), (k_i, \mathsf{c}_{i,0}, \mathsf{c}_{i,1}, \mathsf{d}_i))$ is a collision for h. Since this collision event of RO h has been excluded in $\mathbf{G}_0$, we have

$$\Pr\left[\mathbf{G}_0^{\mathcal{A}} \Rightarrow 1\right] = \Pr\left[\mathbf{G}_1^{\mathcal{A}} \Rightarrow 1\right].$$

Game $\mathbf{G}_2$: In this game, we simulate DEC by searching for the corresponding decapsulated keys from the random oracle queries. Intuitively, this does not change the view of $\mathcal{A}$, since a ciphertext is valid if $\mathcal{A}$

[1]Truncation collision resistance was defined in [JK18]. We do not recall it here, but compute the probability directly, since we model our hash functions as random oracles.

has asked the corresponding random oracle queries before. Otherwise, the ciphertext is invalid and DEC will output $\perp$.

Concretely, $\mathbf{G}_2$ use three lists $\mathcal{L}_{\mathsf{valH}}, \mathcal{L}_{\mathsf{invH}}$, and $\mathcal{L}_{\mathsf{dec}}$ to keep track of the oracle queries to H, and each of them stores a particular type of oracle queries, namely:

- $(b, \mathsf{c}_0, \mathsf{c}_1, \psi) \in \mathcal{L}_{\mathsf{valH}}$ if $\mathcal{A}$ has queried H on $(b, \mathsf{c}_0, \mathsf{c}_1, \psi)$ and $\psi = \mathsf{Decaps}(\mathsf{sk}, \mathsf{c}_b)$. We call this type of hash queries valid.
- $(b, \mathsf{c}_0, \mathsf{c}_1, \psi) \in \mathcal{L}_{\mathsf{invH}}$ if $\mathcal{A}$ has queried H on $(b, \mathsf{c}_0, \mathsf{c}_1, \psi)$ and $\psi \neq \mathsf{Decaps}(\mathsf{sk}, \mathsf{c}_b)$. We call this type of hash queries invalid.
- $(b, \mathsf{c}_0, \mathsf{c}_1) \in \mathcal{L}_{\mathsf{dec}}$ if $\mathcal{A}$ has queried DEC with $(\mathsf{c}_0, \mathsf{c}_1)$ as parts of a ciphertext and there does not exist a ψ such that $(b, \mathsf{c}_0, \mathsf{c}_1, \psi) \in \mathcal{L}_{\mathsf{valH}}$.

  We have $\mathcal{L}_{\mathsf{invH}} \cap \mathcal{L}_{\mathsf{valH}} = \emptyset$ and if $(b, \mathsf{c}_0, \mathsf{c}_1, \psi) \in \mathcal{L}_{\mathsf{valH}}$ then $(b, \mathsf{c}_0, \mathsf{c}_1) \notin \mathcal{L}_{\mathsf{dec}}$.

Oracles H and DEC in $\mathbf{G}_2$ are simulated in the following ways: DEC searches decapsulated key ψ and (K, k) from $\mathcal{L}_{\mathsf{valH}}$. If it fails, then it returns a random (K, k) and records it in $\mathcal{L}_{\mathsf{dec}}$. H maintains $\mathcal{L}_{\mathsf{valH}}$ and $\mathcal{L}_{\mathsf{invH}}$ so that its outputs are consistent with the outputs of DEC. For more details, we refer to Lines 23 to 29 and Lines 40 to 52. We note that the use of these three lists is internal but the outputs of H and DEC are the same as in $\mathbf{G}_1$. Thus,

$$\Pr\left[\mathbf{G}_1^{\mathcal{A}} \Rightarrow 1\right] = \Pr\left[\mathbf{G}_2^{\mathcal{A}} \Rightarrow 1\right].$$

Game $\mathbf{G}_3$: We generate μ KEM keys $(\psi_{1,1-b_1}, \ldots, \psi_{\mu,1-b_\mu})$ uniformly at random (cf. Line 09) and we abort if $\mathcal{A}$'s queries to H include $\psi_{i,1-b_i} (i \in [\mu])$ (cf. Lines 27 to 28 in Figure 9), then $\mathbf{G}_3$ aborts. Let FLIPQRY be the event that $\mathcal{A}$ queries H on $\psi_{i,1-b_i} (i \in [\mu])$ before opening $\mathbf{c}[i]$, and FLIPQRY$'$ be the event that $\mathcal{A}$ queries H on $\psi_{i,1-b_i} (i \in [\mu])$ after opening $\mathbf{c}[i]$. Let FLIPQRY$_j$ and FLIPQRY$'_j (3 \leq j \leq 5)$ be the events that FLIPQRY and FLIPQRY$'$ happen in $\mathbf{G}_j$, respectively.

We distinguish whether $\mathcal{A}$ queries H on $\psi_{i,1-b_i}$ before opening $\mathbf{c}[i]$ (i.e., FLIPQRY) or after opening $\mathbf{c}[i]$ (i.e., FLIPQRY$'$), because in the later game, FLIPQRY$'$ will no longer cause the game to abort, while FLIPQRY will still abort the game.

If FLIPQRY$_3$ and FLIPQRY$'_3$ do not happen, then $\mathbf{G}_3$ and $\mathbf{G}_2$ proceed identically. Since these random KEM keys $\psi_{i,1-b_i}$'s are uniformly random and independent of $\mathbf{c}[i]$'s, we have

$$\left|\Pr\left[\mathbf{G}_2^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathbf{G}_3^{\mathcal{A}} \Rightarrow 1\right]\right| \leq \Pr\left[\text{FLIPQRY}_3\right] + \Pr\left[\text{FLIPQRY}'_3\right] \leq \frac{\mu q_H}{|\Psi|}.$$

Game $\mathbf{G}_4$: We modify the generation of $\mathsf{c}_{i,1-b_i}$. In this game, we generate $\mathsf{c}_{i,1-b_i}$ by choosing $r_{i,1-b_i}$ and computing $(\mathsf{c}_{i,1-b_i}, \psi_{i,1-b_i}) := \mathsf{Encaps}(\mathsf{pk}; r_{i,1-b_i})$ (cf. Lines 10 to 11 in Figure 9), instead of sampling $\mathsf{c}_{i,1-b_i}$ by running $\mathsf{Sample}_{\mathcal{C}}$.

We use the mPR-PCA security of KEM to bound the probability difference between $\mathbf{G}_3$ and $\mathbf{G}_4$. The game simulators of $\mathbf{G}_4$ does not need $r_{i,1-b_i}$ to respond OPEN queries, so we can construct a mPR-PCA adversary $\mathcal{B}_1$ that simulates $\mathbf{G}_3$ or $\mathbf{G}_4$ for $\mathcal{A}$.

$\mathcal{B}_1$ is constructed in Figure 10. If $\mathcal{B}_1$ is interacting with game $\mathsf{mPR\text{-}PCA}^{\mathcal{B}_1,\mu}_{\mathsf{KEM},1}$, then it is simulating $\mathbf{G}_3$, since in this case, $\mathsf{c}_{i,1-b_i} \xleftarrow{\$} \mathcal{C}$ and $\psi_{i,1-b_i} \xleftarrow{\$} \Psi$ (which is the same as $\psi_{i,1-b_i} \leftarrow \mathsf{Sample}_{\mathcal{C}}$ if KEM is $\mathcal{C}$-Explainable) for all $i \in [\mu]$. Otherwise, $\mathcal{B}_1$ is simulating $\mathbf{G}_4$ since now $(\hat{\mathsf{c}}_i, \hat{\psi}_i) := \mathsf{Encaps}(\mathsf{pk}; r_{i,1-b_i})$ for some unknown $r_{i,1-b_i} \xleftarrow{\$} \mathcal{R}$ for all $i \in [\mu]$. When simulating the H oracle, $\mathcal{B}_1$ uses PCO_{pr} oracle to check if $\psi = \mathsf{Decaps}(\mathsf{sk}, \mathsf{c}_b)$. By the abort event introduced in $\mathbf{G}_3$ (Lines 27 to 28 in Figure 9), $\mathcal{A}$ cannot query $\psi_i^* (1 \leq i \leq \mu)$. Therefore, $\mathcal{B}_1$ perfectly simulates $\mathbf{G}_3$ or $\mathbf{G}_4$, and we have

$$\left|\Pr\left[\mathbf{G}_3^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathbf{G}_4^{\mathcal{A}} \Rightarrow 1\right]\right| \leq \mathsf{Adv}^{\mathsf{mPR\text{-}PCA}}_{\mathsf{KEM}}(\mathcal{B}_1, \mu).$$

We also need to bound $\Pr\left[\text{FLIPQRY}'_4\right]$ because it will be used later. The modification introduced in $\mathbf{G}_4$ change the probability that FLIPQRY$'$ happens in $\mathbf{G}_4$. By changing $\mathbf{G}_3$ (resp., $\mathbf{G}_4$) such that it outputs 1 if

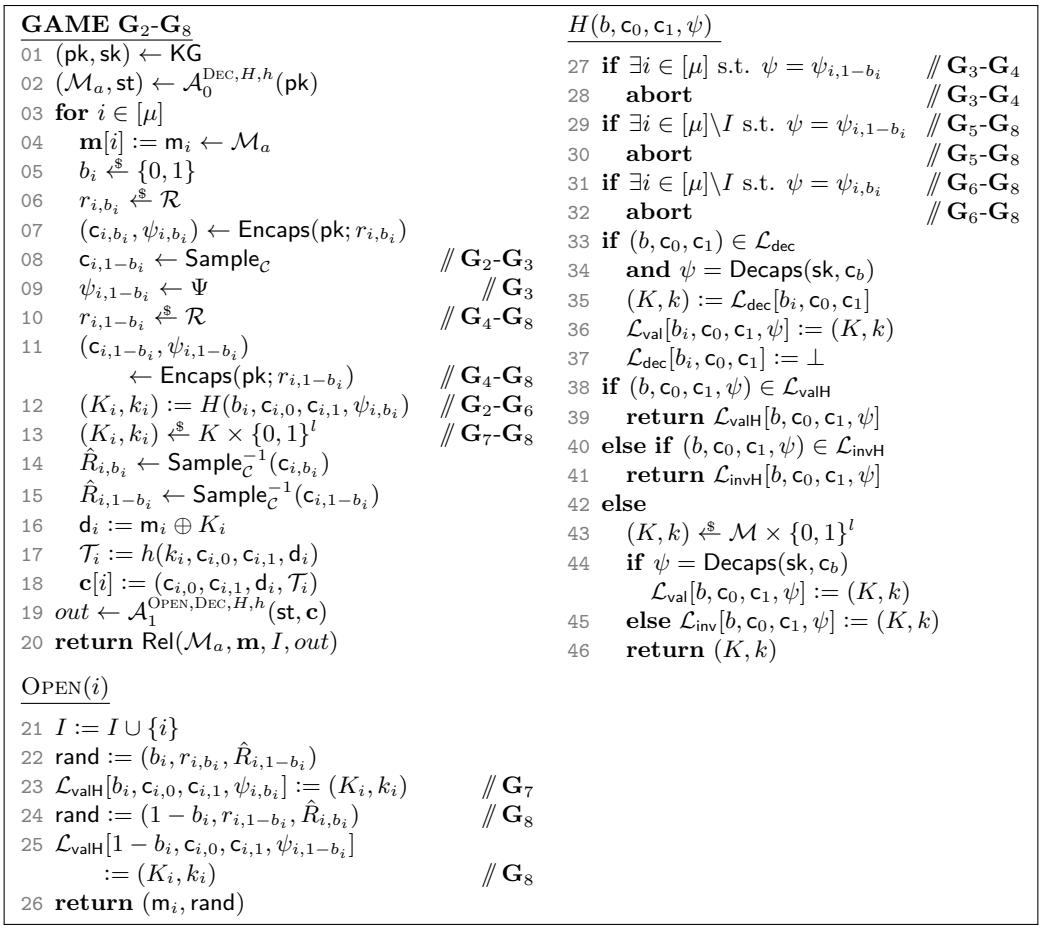

Figure 9: Games $\mathbf{G}_2$-$\mathbf{G}_7$ for proving Theorem 3.5. The decryption oracle DEC is the same as the one in $\mathbf{G}_2$ in Figure 8. The random oracle h is simulated in the standard way, so here we ignore the details.

$\text{FLIPQRY}'_3$ (resp., $\text{FLIPQRY}'_4$) happens, then the reduction $\mathcal{B}_1$ also bound the probability difference between $\Pr\left[\text{FLIPQRY}'_3\right]$ and $\Pr\left[\text{FLIPQRY}'_4\right]$. That is, we have

$$\left|\Pr\left[\text{FLIPQRY}'_3\right] - \Pr\left[\text{FLIPQRY}'_4\right]\right| \leq \mathsf{Adv}^{\mathsf{mPR\text{-}PCA}}_{\mathsf{KEM}}(\mathcal{B}_1, \mu).$$

Game $\mathbf{G}_5$: We change the abort condition introduced in $\mathbf{G}_3$. Now $\text{FLIPQRY}'$ will no longer make the game abort. $\mathcal{A}$ notices this modification if $\text{FLIPQRY}'_4$ happens. We have

$$\begin{aligned}\left|\Pr\left[\mathbf{G}_4^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathbf{G}_5^{\mathcal{A}} \Rightarrow 1\right]\right| \leq \Pr\left[\text{FLIPQRY}'_4\right] &\leq \left|\Pr\left[\text{FLIPQRY}'_4\right] - \Pr\left[\text{FLIPQRY}'_3\right]\right| + \Pr\left[\text{FLIPQRY}'_3\right] \\ &\leq \mathsf{Adv}^{\mathsf{mPR\text{-}PCA}}_{\mathsf{KEM}}(\mathcal{B}_1, \mu) + \frac{\mu q_H}{|\Psi|}.\end{aligned}$$

Game $\mathbf{G}_6$: We introduce a new abort condition in the H oracle: If $\mathcal{A}$ queries H on ψ_{i,b_i} for some $i \in [\mu]$, then $\mathbf{G}_6$ aborts (cf. Lines 31 to 32). Let QRY be this event and QRY_j be the event that QRY happens in $\mathbf{G}_j$. The adversary cannot detect this modification unless it triggers QRY_6. We have

$$\left|\Pr\left[\mathbf{G}_5^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathbf{G}_6^{\mathcal{A}} \Rightarrow 1\right]\right| \leq \Pr\left[\text{QRY}_6\right].$$

Here we cannot bound $\Pr\left[\text{QRY}_6\right]$ using mPR-PCA security of KEM , since if the adversary queries OPEN(i), then the simulator has to return r_{i,b_i}, which is unknown to the reduction from $\mathsf{mPR\text{-}PCA}$. We

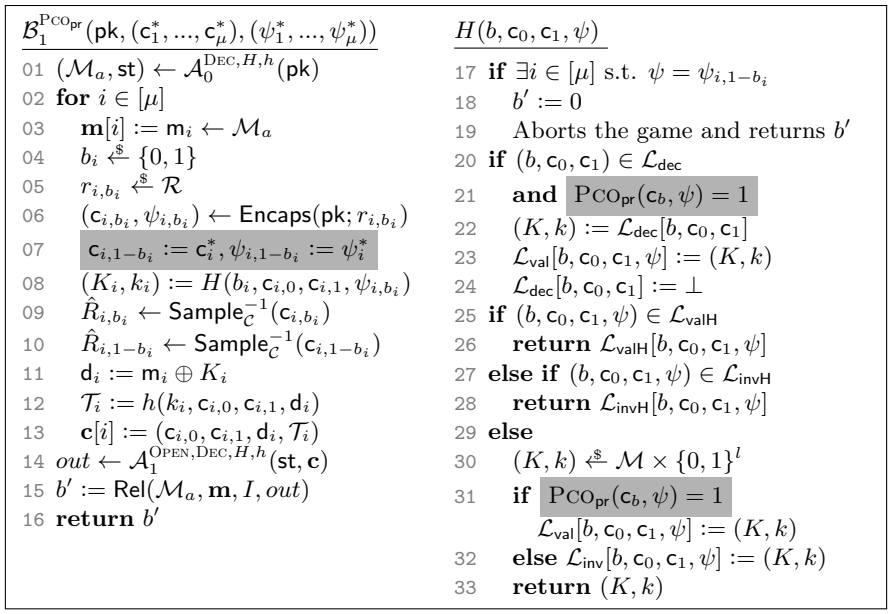

Figure 10: mPR-PCA adversary $\mathcal{B}_1$ in bounding the difference between $\mathbf{G}_3$ and $\mathbf{G}_4$. The simulation of DEC and OPEN is the same as in Figure 9. The highlight codes show how $\mathcal{B}_1$ use its inputs and oracles to simulate $\mathbf{G}_3$ or $\mathbf{G}_4$. If $\mathcal{A}$ queries H on $\psi_i^* (1 \le i \le \mu)$, $\mathcal{B}_1$ aborts and outputs 0.

will bound it later. Our strategy is to decouple $\mathbf{c}[i]$ with $H(b_i, \mathsf{c}_{i,0}, \mathsf{c}_{i,1}, \psi_{i,b_i})$ and then use the randomness $(1 - b_i, r_{i,1-b_i}, \hat{R}_{i,b_i})$ to explain $\mathbf{c}[i]$, where $\hat{R}_{i,b_i} \leftarrow \mathsf{Sample}_{\mathcal{C}}^{-1}(\mathbf{c}[i])$ (and thus we do not need r_{i,b_i} and can construct reduction from mPR-PCA).

Game $\mathbf{G}_7$: The difference to $\mathbf{G}_6$ is that when generating $\mathbf{c}[i]$, we choose random key pairs (K_i, k_i) independent of $H(b_i, \mathsf{c}_{i,0}, \mathsf{c}_{i,1}, \psi_{i,b_i})$, and when $\mathcal{A}$ opens $\mathbf{c}[i]$, we define $H(b_i, \mathsf{c}_{i,0}, \mathsf{c}_{i,1}, \psi_{i,b_i})$ as (K_i, k_i) (cf. Line 23).

By the abort condition in H, $H(b_i, \mathsf{c}_{i,0}, \mathsf{c}_{i,1}, \psi_{i,b_i})$ will not be defined before $\mathbf{c}[i]$ is opened. Hence, this modification does not change $\mathcal{A}$'s view, we have

$$\Pr\left[\mathbf{G}_6^{\mathcal{A}} \Rightarrow 1\right] = \Pr\left[\mathbf{G}_7^{\mathcal{A}} \Rightarrow 1\right], \Pr\left[\mathrm{QRY}_6\right] = \Pr\left[\mathrm{QRY}_7\right].$$

Game $\mathbf{G}_8$: We modify the simulation of OPEN: When $\mathcal{A}$ opens $\mathbf{c}[i]$, we sets $H(1-b_i, \mathsf{c}_{i,0}, \mathsf{c}_{i,1}, \psi_{i,1-b_i}) := (K_i, k_i)$. Instead of returning the actual randomness $(b_i, r_{i,b_i}, \hat{R}_{i,1-b_i})$, we return its complement, $(1 - b_i, r_{i,1-b_i}, \hat{R}_{i,b_i})$ (cf. Lines 24 to 25).

We argue that if QRY_8 does not occur, then the view of $\mathcal{A}$ in $\mathbf{G}_8$ is the same as in $\mathbf{G}_7$. This is because $\mathbf{G}_8$ does not abort means that $\mathcal{A}$ has queried neither $H(b_i, \mathsf{c}_{i,0}, \mathsf{c}_{i,1}, \psi_{i,b_i})$ for any $i \in [\mu] \backslash I$ nor $H(1 - b_i, \mathsf{c}_{i,0}, \mathsf{c}_{i,1}, \psi_{i,1-b_i})$ for any $i \in [\mu] \backslash I$. Hence, $\mathcal{A}$ has no information about these two values, and, as a result, $\mathcal{A}$ cannot see the change in OPEN. We have

$$\Pr\left[\mathbf{G}_7^{\mathcal{A}} \Rightarrow 1\right] = \Pr\left[\mathbf{G}_8^{\mathcal{A}} \Rightarrow 1\right], \Pr\left[\mathrm{QRY}_7\right] = \Pr\left[\mathrm{QRY}_8\right].$$

Now we can bound $\Pr[\mathrm{QRY}_8]$ by constructing a reduction $\mathcal{B}_2$ against the mPR-PCA security of KEM. It simulates $\mathbf{G}_8$ for $\mathcal{A}$. $\mathcal{B}_2$ has a similar structure with $\mathcal{B}_1$ in Figure 10 except that now $\mathcal{B}_2$ embeds $(\mathsf{c}_i^*, \psi_i^*)$ into $(\mathsf{c}_{i,b_i}, \psi_{i,b_i})$. The construction of $\mathcal{B}_2$ is shown in Figure 11.

We analyze the mPR-PCA advantage of $\mathcal{B}_2$ and bound $\Pr[\mathrm{QRY}_8]$. If $\mathcal{B}_2$ is playing the game $\mathsf{mPR\text{-}PCA}_{\mathsf{KEM},0}^{\mathcal{B}_2,\mu}$, then it perfectly simulates $\mathbf{G}_8$, and it outputs 1 if QRY_8 does not happen. So, we have

$$\Pr\left[\mathsf{mPR\text{-}PCA}_{\mathsf{KEM},0}^{\mathcal{B}_2,\mu} \Rightarrow 1\right] = 1 - \Pr\left[\mathrm{QRY}_8\right].$$

$\underline{\mathcal{B}_2^{\mathrm{PCO_{pr}}}(\mathsf{pk}, (\mathsf{c}_1^*, ..., \mathsf{c}_\mu^*), (\psi_1^*, ..., \psi_\mu^*))}$

01 $b' := 1$
02 $(\mathcal{M}_a, \mathsf{st}) \leftarrow \mathcal{A}_0^{\mathrm{DEC},H,h}(\mathsf{pk})$
03 **for** $i \in [\mu]$
04 $\quad \mathbf{m}[i] := \mathsf{m}_i \leftarrow \mathcal{M}_a$
05 $\quad b_i \xleftarrow{\$} \{0,1\}$
06 $\quad r_{i,1-b_i} \xleftarrow{\$} \mathcal{R}$
07 $\quad (\mathsf{c}_{i,1-b_i}, \psi_{i,1-b_i}) \leftarrow \mathsf{Encaps}(\mathsf{pk}; r_{i,1-b_i})$
08 $\quad \mathsf{c}_{i,b_i} := \mathsf{c}_i^*, \psi_{i,b_i} := \psi_i^*$
09 $\quad (K_i, k_i) \xleftarrow{\$} \mathcal{M} \times \{0,1\}^l$
10 $\quad \hat{R}_{i,b_i} \leftarrow \mathsf{Sample}_{\mathcal{C}}^{-1}(\mathsf{c}_{i,b_i})$
11 $\quad \hat{R}_{i,1-b_i} \leftarrow \mathsf{Sample}_{\mathcal{C}}^{-1}(\mathsf{c}_{i,1-b_i})$
12 $\quad \mathsf{d}_i := \mathsf{m}_i \oplus K_i$
13 $\quad \mathcal{T}_i := h(k_i, \mathsf{c}_{i,0}, \mathsf{c}_{i,1}, \mathsf{d}_i)$
14 $\quad \mathbf{c}[i] := (\mathsf{c}_{i,0}, \mathsf{c}_{i,1}, \mathsf{d}_i, \mathcal{T}_i)$
15 $out \leftarrow \mathcal{A}_1^{\mathrm{OPEN},\mathrm{DEC},H,h}(\mathsf{st}, \mathbf{c})$
16 **return** b'

$\underline{\mathrm{OPEN}(i)}$

17 $\mathsf{rand} := (1 - b_i, r_{i,1-b_i}, \mathsf{c}_{i,b_i})$
18 $\mathcal{L}_{\mathsf{valH}}[1 - b_i, \mathsf{c}_{i,0}, \mathsf{c}_{i,1}, \psi_{i,1-b_i}] := (K_i, k_i)$
19 **return** $(\mathsf{m}_i, \mathsf{rand})$

$\underline{H(b, \mathsf{c}_0, \mathsf{c}_1, \psi)}$

20 **if** $\exists i \in [\mu] \backslash I$ s.t. $\psi = \psi_{i,1-b_i}$
21 $\quad$ Aborts the game and returns b'
22 **if** $\exists i \in [\mu] \backslash I$ s.t. $\psi = \psi_{i,b_i}$
23 $\quad b' := 0$ $\quad$ // $\mathcal{A}$ triggers QRY_8
24 $\quad$ Aborts the game and returns b'
25 **if** $(b, \mathsf{c}_0, \mathsf{c}_1) \in \mathcal{L}_{\mathsf{dec}}$
26 $\quad$ **and** $\mathrm{PCO}_{\mathsf{pr}}(\mathsf{c}_b, \psi) = 1$
27 $\quad (K, k) := \mathcal{L}_{\mathsf{dec}}[b_i, \mathsf{c}_0, \mathsf{c}_1]$
28 $\quad \mathcal{L}_{\mathsf{val}}[b_i, \mathsf{c}_0, \mathsf{c}_1, \psi] := (K, k)$
29 $\quad \mathcal{L}_{\mathsf{dec}}[b_i, \mathsf{c}_0, \mathsf{c}_1] := \perp$
30 **if** $(b, \mathsf{c}_0, \mathsf{c}_1, \psi) \in \mathcal{L}_{\mathsf{valH}}$
31 $\quad$ **return** $\mathcal{L}_{\mathsf{valH}}[b, \mathsf{c}_0, \mathsf{c}_1, \psi]$
32 **else if** $(b, \mathsf{c}_0, \mathsf{c}_1, \psi) \in \mathcal{L}_{\mathsf{invH}}$
33 $\quad$ **return** $\mathcal{L}_{\mathsf{invH}}[b, \mathsf{c}_0, \mathsf{c}_1, \psi]$
34 **else**
35 $\quad (K, k) \xleftarrow{\$} \mathcal{M} \times \{0,1\}^l$
36 $\quad$ **if** $\mathrm{PCO}(\mathsf{c}_b, \psi) = 1$
$\quad\quad \mathcal{L}_{\mathsf{val}}[b, \mathsf{c}_0, \mathsf{c}_1, \psi] := (K, k)$
37 $\quad$ **else** $\mathcal{L}_{\mathsf{inv}}[b, \mathsf{c}_0, \mathsf{c}_1, \psi] := (K, k)$
38 $\quad$ **return** (K, k)

Figure 11: mPR-PCA adversary $\mathcal{B}_2$ in bounding $\Pr[\mathrm{QRY}_8]$. The simulation of DEC is the same as in Figure 9. The highlight codes show how $\mathcal{B}_2$ use its inputs and oracles to simulate $\mathbf{G}_8$. If $\mathcal{A}$ queries H on $\psi_{i,1-b_i}$ for some $i \notin [\mu] \backslash I$, then $\mathcal{B}_2$ aborts the simulation and return 1 (the same as if the game ends in normal). If $\mathcal{A}$ triggers QRY_8 (i.e., $\mathcal{A}$ queries H on ψ_{i,b_i} for some $i \notin [\mu] \backslash I$), then $\mathcal{B}_2$ returns 0, which indicates that it is interacting with $\mathsf{mPR\text{-}PCA}_{\mathsf{KEM},0}^{\mathcal{B}_2,\mu}$.

If $\mathcal{B}_2$ is playing the game $\mathsf{mPR\text{-}PCA}_{\mathsf{KEM},1}^{\mathcal{B}_2,\mu}$, then $\psi_1^*, ..., \psi_\mu^*$ are uniformly at random, which means that $\mathcal{B}_2$ outputs 1 with probability at least $1 - \frac{\mu q_H}{|\Psi|}$. Therefore, we have

$$\begin{aligned}\Pr[\mathrm{QRY}_8] &\leq \left|\Pr\left[\mathsf{mPR\text{-}PCA}_{\mathsf{KEM},0}^{\mathcal{B}_2,\mu} \Rightarrow 1\right] - \Pr\left[\mathsf{mPR\text{-}PCA}_{\mathsf{KEM},1}^{\mathcal{B}_2,\mu} \Rightarrow 1\right]\right| + \frac{\mu q_H}{|\Psi|} \\ &= \mathsf{Adv}_{\mathsf{KEM}}^{\mathsf{mPR\text{-}PCA}}(\mathcal{B}_2, \mu) + \frac{\mu q_H}{|\Psi|},\end{aligned}$$

and we also have

$$\begin{aligned}&\Pr[\mathrm{QRY}_6] = \Pr[\mathrm{QRY}_7] = \Pr[\mathrm{QRY}_8] \leq \mathsf{Adv}_{\mathsf{KEM}}^{\mathsf{mPR\text{-}PCA}}(\mathcal{B}_2, \mu) + \frac{\mu q_H}{|\Psi|} \\ &\left|\Pr\left[\mathbf{G}_5^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathbf{G}_8^{\mathcal{A}} \Rightarrow 1\right]\right| \leq \mathsf{Adv}_{\mathsf{KEM}}^{\mathsf{mPR\text{-}PCA}}(\mathcal{B}_2, \mu) + \frac{\mu q_H}{|\Psi|}.\end{aligned}$$

Game $\mathbf{G}_9$: We rewrite $\mathbf{G}_8$ in Figure 12 with some conceptual modifications. The simulator only chooses $b_i \xleftarrow{\$} \{0,1\}$ when $\mathcal{A}$ opens $\mathbf{c}[i]$ (cf. Line 32). Moreover, the two abort conditions in H are rewritten so that both of them are independent of b_i.

We argue that $\mathbf{G}_9$ is equivalent to $\mathbf{G}_8$. In $\mathbf{G}_8$, all challenge ciphertexts are encrypted by random keys (K_i, k_i), and thus $\mathbf{c}[i]$ is independent of b_i, and the simulator can specify b_i when $\mathcal{A}$ opens $\mathbf{c}[i]$. Hence, the abort conditions of H are actually independent of b_i, and thus can be rewritten independent of b_i in both $\mathbf{G}_8$ and $\mathbf{G}_9$. Therefore, we have

$$\Pr\left[\mathbf{G}_8^{\mathcal{A}} \Rightarrow 1\right] = \Pr\left[\mathbf{G}_9^{\mathcal{A}} \Rightarrow 1\right].$$

Game $\mathbf{G}_{10}$: We undo all the abort conditions in H. $\mathcal{A}$ cannot detect this change unless it triggers one of the abort events in H in $\mathbf{G}_9$. We construct an mPR-PCA adversary $\mathcal{B}_3$ to bound this difference.

GAME $\mathbf{G}_9$-$\mathbf{G}_{11}$

01 $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KG}$
02 $(\mathcal{M}_a, \mathsf{st}) \leftarrow \mathcal{A}_0^{\textsc{Dec},H,h}(\mathsf{pk})$
03 **for** $i \in [\mu]$
04 $\quad \mathbf{m}[i] := \mathsf{m}_i \leftarrow \mathcal{M}_a$
05 $\quad$ **for** $b \in \{0,1\}$:
06 $\quad\quad r_{i,b} \xleftarrow{\$} \mathcal{R}$
07 $\quad\quad (\mathsf{c}_{i,b}, \psi_{i,b}) \leftarrow \mathsf{Encaps}(\mathsf{pk}; r_{i,b})$
08 $\quad\quad \hat{R}_{i,b} \leftarrow \mathsf{Sample}_{\mathcal{C}}^{-1}(\mathsf{c}_{i,b})$
09 $\quad (K_i, k_i) \xleftarrow{\$} \mathcal{M} \times \{0,1\}^l$
10 $\quad \mathsf{d}_i := \mathsf{m}_i \oplus K_i$
11 $\quad \mathcal{T}_i := h(k_i, \mathsf{c}_{i,0}, \mathsf{c}_{i,1}, \mathsf{d}_i)$
12 $\quad \mathbf{c}[i] := (\mathsf{c}_{i,0}, \mathsf{c}_{i,1}, \mathsf{d}_i, \mathcal{T}_i)$
13 $out \leftarrow \mathcal{A}_1^{\textsc{Open},\textsc{Dec},H,h}(\mathsf{st}, \mathbf{c})$
14 **return** $\mathsf{Rel}(\mathcal{M}_a, \mathbf{m}, I, out)$

$\textsc{Dec}(c)$ // $c \notin \mathbf{c}$

15 **parse** $(\mathsf{c}_0, \mathsf{c}_1, \mathsf{d}, \mathcal{T}) =: c$
16 **if** $\exists i \in [\mu]$ s.t. $\mathcal{T} = \mathcal{T}_i$
$\quad$ **return** $\perp$ // $\mathbf{G}_9$-$\mathbf{G}_{10}$
17 $\mathsf{m} := \perp$
18 **for** $b \in \{0,1\}$:
19 $\quad$ **if** $\exists \psi$ s.t. $(b, \mathsf{c}_0, \mathsf{c}_1, \psi) \in \mathcal{L}_{\mathsf{val}}$ // $\mathbf{G}_9$-$\mathbf{G}_{10}$
20 $\quad\quad (K_b, k_b) := \mathcal{L}_{\mathsf{val}}[b, \mathsf{c}_0, \mathsf{c}_1, \psi]$ // $\mathbf{G}_9$-$\mathbf{G}_{10}$
21 $\quad$ **else if** $(b, \mathsf{c}_0, \mathsf{c}_1) \in \mathcal{L}_{\mathsf{dec}}$ // $\mathbf{G}_9$-$\mathbf{G}_{10}$
22 $\quad\quad (K_b, k_b) := \mathcal{L}_{\mathsf{dec}}[b, \mathsf{c}_0, \mathsf{c}_1]$ // $\mathbf{G}_9$-$\mathbf{G}_{10}$
23 $\quad$ **else** // $\mathbf{G}_9$-$\mathbf{G}_{10}$
24 $\quad\quad (K_b, k_b) \xleftarrow{\$} \mathcal{M} \times \{0,1\}^l$ // $\mathbf{G}_9$-$\mathbf{G}_{10}$
25 $\quad\quad \mathcal{L}_{\mathsf{dec}}[b, \mathsf{c}_0, \mathsf{c}_1] := (K_b, k_b)$ // $\mathbf{G}_9$-$\mathbf{G}_{10}$
26 $\quad \psi_b := \mathsf{Decaps}(\mathsf{sk}, \mathsf{c}_b)$ // $\mathbf{G}_{11}$
27 $\quad (K_b, k_b) := H(b, \mathsf{c}_1, \mathsf{c}_0, \psi_b)$ // $\mathbf{G}_{11}$
28 $\quad \mathcal{T}_b := h(k_b, \mathsf{c}_0, \mathsf{c}_1, \mathsf{d})$
29 $\quad$ **if** $\mathcal{T}_b = \mathcal{T}$: $\mathsf{m} := \mathsf{d} \oplus K_b$
30 **return** m

$\textsc{Open}(i)$

31 $I := I \cup \{i\}$
32 $b_i \xleftarrow{\$} \{0,1\}$
33 $\mathsf{rand} := (b_i, r_{i,b_i}, \mathsf{c}_{i,1-b_i})$
34 $\mathcal{L}_{\mathsf{valH}}[b_i, \mathsf{c}_{i,0}, \mathsf{c}_{i,1}, \psi_{i,b_i}] := (K_i, k_i)$ // $\mathbf{G}_9$-$\mathbf{G}_{10}$
35 $\mathcal{L}_{\mathsf{H}}[b_i, \mathsf{c}_{i,0}, \mathsf{c}_{i,1}, \psi_{i,b_i}] := (K_i, k_i)$ // $\mathbf{G}_{11}$
36 **return** $(\mathsf{m}_i, \mathsf{rand})$

$H(b, \mathsf{c}_0, \mathsf{c}_1, \psi)$

37 **if** $\exists i \in [\mu] \backslash I$ s.t. $\psi = \psi_{i,0}$ // $\mathbf{G}_9$
38 $\quad$ **abort** // $\mathbf{G}_9$
39 **if** $\exists i \in [\mu] \backslash I$ s.t. $\psi = \psi_{i,1}$ // $\mathbf{G}_9$
40 $\quad$ **abort** // $\mathbf{G}_9$
41 **if** $(b, \mathsf{c}_0, \mathsf{c}_1) \in \mathcal{L}_{\mathsf{dec}}$ // $\mathbf{G}_9$-$\mathbf{G}_{10}$
42 $\quad$ **and** $\psi = \mathsf{Decaps}(\mathsf{sk}, \mathsf{c}_b)$ // $\mathbf{G}_9$-$\mathbf{G}_{10}$
43 $\quad (K, k) := \mathcal{L}_{\mathsf{dec}}[b_i, \mathsf{c}_0, \mathsf{c}_1]$ // $\mathbf{G}_9$-$\mathbf{G}_{10}$
44 $\quad \mathcal{L}_{\mathsf{val}}[b_i, \mathsf{c}_0, \mathsf{c}_1, \psi] := (K, k)$ // $\mathbf{G}_9$-$\mathbf{G}_{10}$
45 $\quad \mathcal{L}_{\mathsf{dec}}[b_i, \mathsf{c}_0, \mathsf{c}_1] := \perp$ // $\mathbf{G}_9$-$\mathbf{G}_{10}$
46 **if** $(b, \mathsf{c}_0, \mathsf{c}_1, \psi) \in \mathcal{L}_{\mathsf{valH}}$ // $\mathbf{G}_9$-$\mathbf{G}_{10}$
47 $\quad$ **return** $\mathcal{L}_{\mathsf{valH}}[b, \mathsf{c}_0, \mathsf{c}_1, \psi]$ // $\mathbf{G}_9$-$\mathbf{G}_{10}$
48 **else if** $(b, \mathsf{c}_0, \mathsf{c}_1, \psi) \in \mathcal{L}_{\mathsf{invH}}$ // $\mathbf{G}_9$-$\mathbf{G}_{10}$
49 $\quad$ **return** $\mathcal{L}_{\mathsf{invH}}[b, \mathsf{c}_0, \mathsf{c}_1, \psi]$ // $\mathbf{G}_9$-$\mathbf{G}_{10}$
50 **else** // $\mathbf{G}_9$-$\mathbf{G}_{10}$
51 $\quad (K, k) \xleftarrow{\$} \mathcal{M} \times \{0,1\}^l$ // $\mathbf{G}_9$-$\mathbf{G}_{10}$
52 $\quad$ **if** $\psi = \mathsf{Decaps}(\mathsf{sk}, \mathsf{c}_b)$ // $\mathbf{G}_9$-$\mathbf{G}_{10}$
$\quad\quad \mathcal{L}_{\mathsf{val}}[b, \mathsf{c}_0, \mathsf{c}_1, \psi] := (K, k)$ // $\mathbf{G}_9$-$\mathbf{G}_{10}$
53 $\quad$ **else**
$\quad\quad \mathcal{L}_{\mathsf{inv}}[b, \mathsf{c}_0, \mathsf{c}_1, \psi] := (K, k)$ // $\mathbf{G}_9$-$\mathbf{G}_{10}$
54 $\quad$ **return** (K, k) // $\mathbf{G}_9$-$\mathbf{G}_{10}$
55 **if** $\mathcal{L}_{\mathsf{H}}[b, \mathsf{c}_0, \mathsf{c}_1, \psi] = \perp$: // $\mathbf{G}_{11}$
56 $\quad (K, k) \xleftarrow{\$} \mathcal{M} \times \{0,1\}^l$ // $\mathbf{G}_{11}$
57 $\quad \mathcal{L}_{\mathsf{H}}[b, \mathsf{c}_0, \mathsf{c}_1, \psi] := (K, k)$ // $\mathbf{G}_{11}$
58 **return** $\mathcal{L}_{\mathsf{H}}[b, \mathsf{c}_0, \mathsf{c}_1, \psi]$ // $\mathbf{G}_{11}$

Figure 12: Games $\mathbf{G}_9$-$\mathbf{G}_{11}$ for proving Theorem 3.5. The random oracle h is simulated in the standard way.

In $\mathcal{B}_3$'s construction, we embed $(\mathsf{c}_i^*, \psi_i^*)$ into $(\mathsf{c}_{i,0}, \psi_{i,0})$ or $(\mathsf{c}_{i,1}, \psi_{i,1})$ randomly (cf. Lines 05 to 06), and specify b_i as $1 - \hat{b_i}$. When $\mathcal{A}$ opens $\mathbf{c}[i]$, we explain $\mathbf{c}[i]$ by using the $(1 - \hat{b_i})$-randomness (cf. Line 20). Note that $\hat{b_i}$ is independent of $\mathcal{A}$'s view before it opens $\mathbf{c}[i]$, and thus the distribution of b_i in $\mathcal{B}_3$'s construction is the same as the one in $\mathbf{G}_9$. If $\mathcal{A}$ triggers one of the abort events in H oracle, then $\mathcal{B}_3$ outputs 0 with probability $\frac{1}{2}$, since $\hat{b_i}$ is sampled independently in uniformly random. Similar to the arguments in bounding $\Pr[\textsc{Qry}_8]$, we have

$$\left|\Pr\left[\mathbf{G}_9^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathbf{G}_{10}^{\mathcal{A}} \Rightarrow 1\right]\right| \leq 2\mathsf{Adv}_{\mathsf{KEM}}^{\mathsf{mPR\text{-}PCA}}(\mathcal{B}_3) + \frac{2\mu q_H}{|\Psi|}.$$

Game $\mathbf{G}_{11}$: We undo the modifications of $\mathbf{G}_2$ and $\mathbf{G}_1$. We have

$$\Pr[\mathbf{G}_{10} \Rightarrow 1] = \Pr[\mathbf{G}_{11} \Rightarrow 1].$$

Now we can construct a SIM-SO-CCA simulator $\mathcal{S}$ that simulates $\mathbf{G}_{10}$ for $\mathcal{A}$ and interacts with the $\mathsf{IDEAL\text{-}SO\text{-}CCA}_{\mathsf{PKE}_1}$ game to conclude the proof. The construction of simulator is shown in Figure 14.

$\mathcal{S}$ samples d_i uniformly from $\mathcal{M}$ and computes K_i as $\mathsf{d}_i \oplus \mathsf{m}_i$ (when $\mathcal{A}$ opens $\mathbf{c}[i]$), which is equivalent to sampling K_i firstly and then computing $\mathsf{d}_i := K_i \oplus \mathsf{m}_i$. Therefore, $\mathcal{S}$ perfectly simulates $\mathbf{G}_{10}$. Note that at

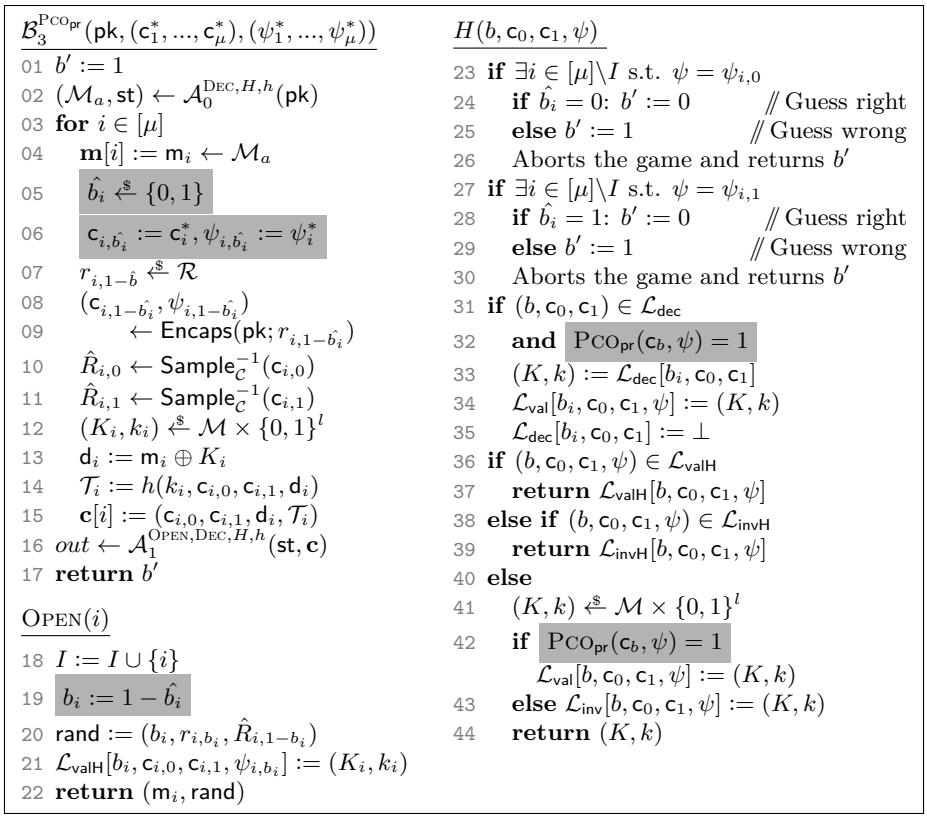

Figure 13: mPR-PCA adversary $\mathcal{B}_3$ in bounding $|\Pr\left[\mathbf{G}_9^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathbf{G}_{10}^{\mathcal{A}} \Rightarrow 1\right]|$. The simulation of DEC is the same as in Figure 12. The highlighted codes show how $\mathcal{B}_3$ uses its inputs and oracles to simulate $\mathbf{G}_{10}$. Specifically, for every $i \in [\mu]$, $\mathcal{B}_3$ embeds the challenge into the $\hat{b_i}$-ciphertexts, and expects $\mathcal{A}$ will trigger the abort event of such $\hat{b_i}$-ciphertexts.

the start of the proof we assume that there is no collision among all K_i's, k_i's, and the outputs of H. This introduces some collision bounds here. That is,

$$\left|\Pr\left[\mathbf{G}_{11}^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathsf{IDEAL\text{-}SO\text{-}CCA}_{\mathsf{PKE}_1}^{\mathcal{S}} \Rightarrow 1\right]\right| \leq \frac{\mu^2 + q_H^2}{|\mathcal{M}|} + \frac{\mu^2 + q_H^2 + q_h^2}{2^l}.$$

By combining all the probability bounds and viewing $\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3$, and $\mathcal{B}_4$ as an adversary $\mathcal{B}_{\mathsf{PR}}$ (choose the one with highest mPR-PCA advantage), we have

$$\begin{aligned}
&\mathsf{Adv}_{\mathsf{PKE}_1}^{\mathsf{SIM\text{-}SO\text{-}CCA}}(\mathcal{A}, \mathcal{S}, \mu, \mathsf{Rel}) \\
&:= \left|\Pr\left[\mathsf{REAL\text{-}SO\text{-}CCA}_{\mathsf{PKE}_1}^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathsf{IDEAL\text{-}SO\text{-}CCA}_{\mathsf{PKE}_1}^{\mathcal{S}} \Rightarrow 1\right]\right| \\
&\leq 5\mathsf{Adv}_{\mathsf{KEM}}^{\mathsf{mPR\text{-}PCA}}(\mathcal{B}_{\mathsf{PR}}, \mu) + \frac{5\mu q_H}{|\Psi|} + 2\left(\frac{\mu^2 + q_H^2}{|\mathcal{M}|} + \frac{\mu^2 + q_H^2 + q_h^2}{2^l}\right),
\end{aligned}$$

as stated in Theorem 3.5. □

3.3 Direct Diffie-Hellman-based Constructions for mPR-PCA secure KEM

We propose three Diffie-Hellman-based constructions of mPR-PCA secure KEM. Throughout this section, we let Ψ be a KEM key space and $H : \{0,1\}^* \to \Psi$ be a random oracle. Let $\mathbb{G}$ be a p-order group with generator g. All constructions in this section have perfect correctness (Figure 2).

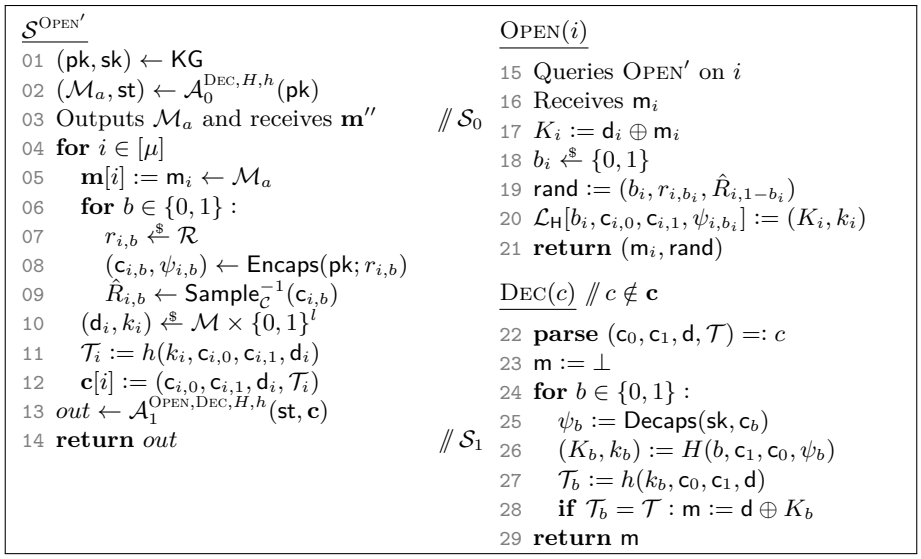

Figure 14: SIM-SO-CCA simulator $\mathcal{S}$ that simulates $\mathbf{G}_{11}$ to conclude the proof of Theorem 3.5. $\mathcal{S}$ simulates ROs h and H in the standard way.

A CONSTRUCTION BASED ON STRONG DH. In Figure 15, we construct a mPR-PCA KEM, $\mathtt{KEM}_{\mathsf{StDH}}$, from the multi-instance strong Diffie-Hellman assumption (Definition 2.4). The ciphertext space of $\mathtt{KEM}_{\mathsf{StDH}}$ is $\mathbb{G}$ and the randomness space is $\mathbb{Z}_p$. $\mathtt{KEM}_{\mathsf{StDH}}$ is essentially the hashed ElGamal KEM [ABR01, CKS08].

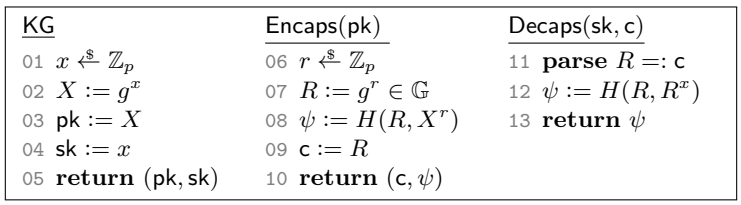

Figure 15: Our direct construction of mPR-PCA secure KEM schemes from the mStDH assumption, $\mathtt{KEM}_{\mathsf{StDH}} = (\mathsf{KG}, \mathsf{Encaps}, \mathsf{Decaps})$.

$\mathtt{KEM}_{\mathsf{StDH}}$ is $\mathbb{G}$-explainable (Definition 3.4) if $\mathbb{G}$ can be sampled (uniformly at random) without using generator and exponent. A concrete example is as follow: Let p be a prime s.t. $q = rp + 1$ is also a prime for some r. Let $\mathbb{G}$ be a subgroup of $\mathbb{Z}_q^*$ and with order p. Canetti et al. [CF01, Section 4.3.2] showed how to sample a group element from such $\mathbb{G}$ without knowing exponent. We can design $\mathsf{Sample}_{\mathbb{G}}$ and $\mathsf{Sample}_{\mathbb{G}}^{-1}$ that works similarly with the sample and fake processes in [CF01, Section 4.3.2], respectively. Such technique can also be used in some widely-used elliptic-curve groups, such as NIST P256, NIST P384, and Curve25519.

Theorem 3.6 $\mathtt{KEM}_{\mathsf{StDH}}$ *in Figure 15 is mPR-PCA secure (Definition 3.3) if the* mStDH *problem is hard on* $\mathbb{G}$ *and* H *is modeled as a random oracle. For any adversary* $\mathcal{A}$ *and relation* Rel*, there exists an adversary* $\mathcal{B}$ *such that*

$$\mathsf{Adv}^{\mathsf{mPR\text{-}PCA}}_{\mathtt{KEM}_{\mathsf{StDH}}}(\mathcal{A}, \mu) \leq 2\mathsf{Adv}^{\mathsf{mStDH}}_{\mathbb{G}}(\mathcal{B}) + \frac{\mu^2}{|\mathbb{G}|} + \frac{\mu^2 + q_H^2 + 2q_H}{|\Psi|},$$

where q_H *is the number of* $\mathcal{A}$*'s queries to* H *and* μ *is the number of challenge ciphertexts.*

Proof. We prove Theorem 3.6 by the games sequence in Figure 16. We assume that there is no collision among all R_i's, ψ_i's, and the outputs of H. This assumption adds collision bounds $\frac{\mu^2}{|\mathbb{G}|} + \frac{\mu^2 + q_H^2}{|\Psi|}$ to the bound

GAME $\mathbf{G}_{0,b}$-$\mathbf{G}_{3,b}, b \in \{0,1\}$

$(X, x) \leftarrow \mathsf{KG}$
for $i \in [\mu]$:
  $r_i \xleftarrow{\$} \mathbb{Z}_p, R_i := g^{r_i}$ // $\mathbf{G}_{0,b}$-$\mathbf{G}_{2,b}$
  $\psi_i := H(R_i, X^{r_i})$ // $\mathbf{G}_{0,b}$-$\mathbf{G}_{1,b}$
  $\psi_i \xleftarrow{\$} \Psi$ // $\mathbf{G}_{2,b}$-$\mathbf{G}_{3,b}$
  $R_i \xleftarrow{\$} \mathbb{G}$ // $\mathbf{G}_{3,b}$
  $\mathbf{c}[i] := R_i, \boldsymbol{\psi}[i] := \psi_i$ // $b = 0$
  $\mathbf{c}[i] \xleftarrow{\$} \mathbb{G}, \boldsymbol{\psi}[i] \xleftarrow{\$} \Psi$ // $b = 1$
$b \leftarrow \mathcal{A}^{H, \mathrm{PCO}_{\mathsf{pr}}}(X, \mathbf{c}, \boldsymbol{\psi})$
return b

$H(R, Z)$

if $\exists i \in [\mu]$ s.t. $Z = R_i^x$ // $\mathbf{G}_{2,b}$-$\mathbf{G}_{3,b}$
  abort // $\mathbf{G}_{2,b}$-$\mathbf{G}_{3,b}$
if $\mathcal{L}_{\mathsf{H}}[R, Z] = \bot$
  $\mathcal{L}_{\mathsf{H}}[R, Z] := \psi \xleftarrow{\$} \Psi$
return $\mathcal{L}_{\mathsf{H}}[R, Z]$

$\mathrm{PCO}_{\mathsf{pr}}(R, \psi)$

if $\exists i \in [\mu]$ s.t. $(R, \psi) = (\mathbf{c}[i], \boldsymbol{\psi}[i])$
  return $\bot$
$Z := R^x$ // $\mathbf{G}_{0,b}$
$\psi := H(R, Z)$ // $\mathbf{G}_{0,b}$
return $\psi' =_? \psi$ // $\mathbf{G}_{0,b}$
if $\exists Z \in \mathbb{G}$ s.t. // $\mathbf{G}_{1,b}$-$\mathbf{G}_{3,b}$
  $\mathcal{L}_{\mathsf{H}}[R, Z] = \psi \wedge R^x = Z$ // $\mathbf{G}_{1,b}$-$\mathbf{G}_{3,b}$
  return 1 // $\mathbf{G}_{1,b}$-$\mathbf{G}_{3,b}$
else return 0 // $\mathbf{G}_{1,b}$-$\mathbf{G}_{3,b}$

Figure 16: Games $\mathbf{G}_{0,b}$-$\mathbf{G}_{3,b}(b \in \{0,1\})$ for proving Theorem 3.6.

of our proof. For $b \in \{0,1\}$, $\mathbf{G}_{0,b}^{\mathcal{A}}$ is equivalent to $\mathsf{mPR\text{-}PCA}_{\mathtt{KEM}_{\mathsf{StDH}},b}^{\mathcal{A},\mu}$, so we have

$$\Pr\left[\mathsf{mPR\text{-}PCA}_{\mathtt{KEM}_{\mathsf{StDH}},b}^{\mathcal{A},\mu} \Rightarrow 1\right] = \Pr\left[\mathbf{G}_{0,b}^{\mathcal{A}} \Rightarrow 1\right].$$

Game $\mathbf{G}_{1,b}$: We change the simulation of $\mathrm{PCO}_{\mathsf{pr}}$. In this game, $\mathrm{PCO}_{\mathsf{pr}}$ does not follow the decapsulation algorithm Decaps of $\mathtt{KEM}_{\mathsf{StDH}}$. Instead, on query (R, ψ), the game simulator checks whether (R, ψ) corresponds to a Z such that $\mathcal{A}$ has queried $H(R, Z) = \psi$ and $Z = R^x$. If such Z does not exist, then it returns 0. Otherwise, it returns 1.

If $\mathrm{PCO}_{\mathsf{pr}}(R, \psi)$ returns 1 in $\mathbf{G}_0$, then $\psi = H(R, R^x)$, and thus $\mathrm{PCO}_{\mathsf{pr}}(R, \psi)$ also returns 1 in $\mathbf{G}_1$ except that $\mathcal{A}$ never queried $H(R, R^x)$ but finds $\psi = H(R, R^x)$. Since H is a RO, the probability that $\mathcal{A}$ gets $H(R, R^x)$ without querying H is $\frac{q_H}{|\Psi|}$. So, we have

$$\left|\Pr\left[\mathbf{G}_{0,b}^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathbf{G}_{1,b}^{\mathcal{A}} \Rightarrow 1\right]\right| \leq \frac{q_H}{|\Psi|}.$$

Game $\mathbf{G}_{2,b}$: We introduce an abort condition in the H oracle and change the generation of ψ_i. If $\mathcal{A}$'s H query include R_i^x $(i \in [\mu])$, then the game simulator aborts. Moreover, in this game, ψ_i is generated by uniformly sampling instead of by computing $H(R_i, R_i^x)$.

Let QRYDH_b be the event that $\mathcal{A}$ queries H on (R_i, R_i^x) $(i \in [\mu])$ in Game $\mathbf{G}_{2,b}$. We claim that if QRYDH_b does not happen, then $\mathbf{G}_{2,b}$ is equivalent to $\mathbf{G}_{1,b}$. This is because, to distinguish $\mathbf{G}_{1,b}$ and $\mathbf{G}_{2,b}$, $\mathcal{A}$ needs to queries H on (R_i, R_i^x) for some $i \in [\mu]$, namely, triggers the event QRYDH_b. If this event does not happen, then $\mathbf{G}_{2,b}$ does not abort and $\mathcal{A}$ cannot learn $H(R_i, R_i^x)$, which assures that ψ_i is uniformly distributed in $\mathcal{A}$'s view, and thus it is equivalent to computing ψ_i uniformly at random. We have

$$\left|\Pr\left[\mathbf{G}_{1,b}^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathbf{G}_{2,b}^{\mathcal{A}} \Rightarrow 1\right]\right| \leq \Pr\left[\mathrm{QRYDH}_b\right].$$

We construct an mStDH adversary $\mathcal{B}_b$ to bound $\Pr[\mathrm{QRYDH}_b]$. In $\mathcal{B}_b$'s construction, we embed $(Y_i)_{1 \leq i \leq \mu}$ into R_i, and use DHP_X to determine if $R^x = Z$. $\mathcal{B}_b$ perfectly simulates $\mathbf{G}_{2,b}$. If QRYDH happens, namely, $\mathcal{A}$ queried H on R_i^x, then by using DHP_X, $\mathcal{B}_b$ can determine such R_i^x and output it as its mStDH solution. Therefore, we have

$$\Pr\left[\mathrm{QRYDH}_b\right] \leq \mathsf{Adv}_{\mathbb{G}}^{\mathsf{mStDH}}(\mathcal{B}_b).$$

$\underline{\mathcal{B}_b^{\mathrm{DHP}_X}(X, Y_1, Y_2, ..., Y_\mu), b \in \{0,1\}}$

01 **for** $i \in [\mu]$:
02 $\quad R_i := Y_i$
03 $\quad \psi_i \xleftarrow{\$} \Psi$
04 $\quad \mathbf{c}[i] := R_i, \boldsymbol{\psi}[i] := \psi_i$
05 $b \leftarrow \mathcal{A}^{H, \mathrm{PCO}_{\mathsf{pr}}}(X, \mathbf{c}, \boldsymbol{\psi})$
06 **return** b

$\underline{H(R, Z)}$

07 **if** $\exists i \in [\mu]$ s.t. $\mathrm{DHP}_X(R_i, Z) = 1$
08 $\quad$ Abort the simulation and output Z
09 **if** $\mathcal{L}_{\mathsf{H}}[R, Z] = \perp$
10 $\quad \mathcal{L}_{\mathsf{H}}[R, Z] := \psi \xleftarrow{\$} \Psi$
11 **return** $\mathcal{L}_{\mathsf{H}}[R, Z]$

$\underline{\mathrm{PCO}_{\mathsf{pr}}(R, \psi)}$

12 **if** $\exists i \in [\mu]$ s.t. $(R, \psi) = (\mathbf{c}[i], \boldsymbol{\psi}[i])$
13 $\quad$ **return** $\perp$
14 **if** $\exists Z \in \mathbb{G}$ s.t.
15 $\quad \mathcal{L}_{\mathsf{H}}[R, Z] = \psi \wedge \mathrm{DHP}_X(R, Z) = 1$
16 $\quad$ **return** 1
17 **else return** 0

Figure 17: mStDH adversary $\mathcal{B}_b$ in bounding QRYDH_b in the proof of Theorem 3.6. Highlighted codes show how to use the DHP oracle to simulate $\mathbf{G}_{2,b}$.

Game $\mathbf{G}_{3,b}$: We generate $\mathbf{c}[i]$ by uniformly sampling instead of following $\mathtt{KEM}_{\mathsf{StDH}}.\mathsf{Encaps}$. The change is conceptual, because in $\mathbf{G}_{2,b}$, we generate $\mathbf{c}[i]$ by computing $R_i := g^{r_i} \in \mathbb{G}$, which is equivalent to sampling R_i from $\mathbb{G}$. Moreover, in $\mathbf{G}_{3,b}$, the game simulator no longer uses r_i to generate ψ_i. Therefore, we have

$$\Pr\left[\mathbf{G}_{2,b}^{\mathcal{A}} \Rightarrow 1\right] = \Pr\left[\mathbf{G}_{3,b}^{\mathcal{A}} \Rightarrow 1\right].$$

Now $\mathbf{G}_{3,0}$ is equivalent to $\mathbf{G}_{3,1}$, so we have $\Pr\left[\mathbf{G}_{3,0}^{\mathcal{A}} \Rightarrow 1\right] = \Pr\left[\mathbf{G}_{3,1}^{\mathcal{A}} \Rightarrow 1\right]$. Combining all the probability differences in the games sequence, and combining $\mathcal{B}_0$ and $\mathcal{B}_1$ as one adversary $\mathcal{B}$, we have

$$\begin{aligned}\mathsf{Adv}^{\mathsf{mPR\text{-}PCA}}_{\mathtt{KEM}_{\mathsf{StDH}}}(\mathcal{A}, \mu) &:= \left| \Pr\left[\mathsf{mPR\text{-}PCA}^{\mathcal{A},\mu}_{\mathtt{KEM}_{\mathsf{StDH}},0} \Rightarrow 1\right] - \Pr\left[\mathsf{mPR\text{-}PCA}^{\mathcal{A},\mu}_{\mathtt{KEM}_{\mathsf{StDH}},1} \Rightarrow 1\right] \right| \\ &\leq 2\mathsf{Adv}^{\mathsf{mStDH}}_{\mathbb{G}}(\mathcal{B}) + \frac{\mu^2}{|\mathbb{G}|} + \frac{\mu^2 + q_H^2 + 2q_H}{|\Psi|}.\end{aligned}$$

□

A Construction based on Twin DH. Using the twinning technique from [CKS08], we can remove the use of StDH assumption in $\mathtt{KEM}_{\mathsf{StDH}}$ and have a scheme $\mathtt{KEM}_{\mathsf{TDH}}$ based on the standard CDH assumption at the cost of being less efficient. $\mathtt{KEM}_{\mathsf{TDH}}$ is shown in Figure 18. The ciphertext space and randomness space of $\mathtt{KEM}_{\mathsf{TDH}}$ are the same as those of $\mathtt{KEM}_{\mathsf{StDH}}$.

$\underline{\mathsf{KG}}$

01 $(x_0, x_1) \xleftarrow{\$} \mathbb{Z}_p$
02 $X_0 := g^{x_0}, X_1 := g^{x_1}$
03 $\mathsf{pk} := (X_0, X_1)$
04 $\mathsf{sk} := (x_0, x_1)$
05 **return** $(\mathsf{pk}, \mathsf{sk})$

$\underline{\mathsf{Encaps}(\mathsf{pk})}$

06 **let** $(X_0, X_1) := \mathsf{pk}$
07 $r \xleftarrow{\$} \mathbb{Z}_p$
08 $R := g^r \in \mathbb{G}$
09 $\psi := H(R, X_0^r, X_1^r)$
10 $\mathsf{c} := R$
11 **return** (c, ψ)

$\underline{\mathsf{Decaps}(\mathsf{sk}, \mathsf{c})}$

12 **let** $R := \mathsf{c}$
13 **let** $(x_0, x_1) := \mathsf{sk}$
14 $\psi := H(R, R^{x_0}, R^{x_1})$
15 **return** ψ

Figure 18: Our Direct Construction of $\mathsf{mPR\text{-}PCA}$ secure KEM schemes from the mTDH assumption, $\mathtt{KEM}_{\mathsf{TDH}} = (\mathsf{KG}, \mathsf{Encaps}, \mathsf{Decaps})$.

Similar to $\mathtt{KEM}_{\mathsf{StDH}}$, $\mathtt{KEM}_{\mathsf{TDH}}$ is $\mathbb{G}$-explainable (Definition 3.4) if $\mathbb{G}$ can be sampled without using generator and exponent. Theorem 3.7 shows that if mTDH is hard on $\mathbb{G}$, then $\mathtt{KEM}_{\mathsf{TDH}}$ is mPR-PCA secure. The proof

idea of Theorem 3.7 is the same as the one of Theorem 3.6. The only difference is that, instead of using the DHP oracle, here we use the 2DHP oracle to check whether the adversary queried $(X_0^{r_i}, X_1^{r_i})$. By [CKS08, Theorem 3], the TDH problem is tightly equivalent to the CDH problem, so we also have Corollary 3.8.

Theorem 3.7 $\mathtt{KEM}_{\mathsf{TDH}}$ *in Figure 18 is mPR-PCA secure (Definition 3.3) if the* mTDH *problem is hard on* $\mathbb{G}$ *and* H *is modeled as a random oracle. For any adversary* $\mathcal{A}$ *and relation* Rel*, there exists an adversary* $\mathcal{B}$ *such that*

$$\mathsf{Adv}^{\mathsf{mPR\text{-}PCA}}_{\mathtt{KEM}_{\mathsf{TDH}}}(\mathcal{A}, \mu) \leq 2\mathsf{Adv}^{\mathsf{mTDH}}_{\mathbb{G}}(\mathcal{B}) + \frac{\mu^2}{|\mathbb{G}|} + \frac{\mu^2 + q_H^2 + 2q_H}{|\Psi|},$$

where q_H *is the number of* $\mathcal{A}$*'s queries to* H *and* μ *is the number of challenge ciphertexts.*

Corollary 3.8 $\mathtt{KEM}_{\mathsf{TDH}}$ *in Figure 18 is mPR-PCA secure (Definition 3.3) if the* mCDH *problem is hard on* $\mathbb{G}$ *and* H *is modeled as a random oracle. For any adversary* $\mathcal{A}$ *and relation* Rel*, there exists an adversary* $\mathcal{B}$ *such that*

$$\mathsf{Adv}^{\mathsf{mPR\text{-}PCA}}_{\mathtt{KEM}_{\mathsf{TDH}}}(\mathcal{A}, \mu) \leq 2\mathsf{Adv}^{\mathsf{mCDH}}_{\mathbb{G}}(\mathcal{B}) + \frac{2(q_H + q_{\mathrm{PCO}}) + \mu^2}{|\mathbb{G}|} + \frac{\mu^2 + q_H^2 + 2q_H}{|\Psi|},$$

where q_H *and* q_{PCO} *are the numbers of* $\mathcal{A}$*'s queries to* H *and* $\mathrm{PCO}_{\mathsf{pr}}$*, respectively, and* μ *is the number of challenge ciphertexts.*

A Construction based on DDH. In Figure 19, we construct a mPR-PCA KEM, $\mathtt{KEM}_{\mathsf{DDH}}$, from the multi-instance decisional Diffie-Hellman assumption (Definition 2.1). Let $g_0 := g$ and $g_1 := g^\omega$ where $\omega \xleftarrow{\$} \mathbb{Z}_p^*$. g_1 is also a generator of $\mathbb{G}$. The ciphertext space of $\mathtt{KEM}_{\mathsf{StDH}}$ is $\mathbb{G}^2$ and the randomness space is $\mathbb{Z}_p$. Similar to $\mathtt{KEM}_{\mathsf{StDH}}$, $\mathtt{KEM}_{\mathsf{DDH}}$ is $\mathbb{G}^2$-explainable (Definition 3.4) if $\mathbb{G}$ can be sampled without using generator and exponent.

KG	Encaps(pk)	Decaps(sk, c)
01 $(x_0, x_1) \xleftarrow{\$} \mathbb{Z}_p$	06 **let** $X := \mathsf{pk}$	12 **let** $(x_0, x_1) := \mathsf{sk}$
02 $X := g_0^{x_0} g_1^{x_1}$	07 $r \xleftarrow{\$} \mathbb{Z}_p$	13 **let** $(R_0, R_1) := \mathsf{c}$
03 $\mathsf{pk} := X$	08 $(R_0, R_1) := (g_0^r, g_1^r) \in \mathbb{G}^2$	14 $\psi := H(R_0, R_1, R_0^{x_0} R_1^{x_1})$
04 $\mathsf{sk} := (x_0, x_1)$	09 $\psi := H(R_0, R_1, X^r)$	15 **return** ψ
05 **return** $(\mathsf{pk}, \mathsf{sk})$	10 $\mathsf{c} := (R_0, R_1)$	
	11 **return** (c, ψ)	

Figure 19: Our direct construction of mPR-PCA secure KEM schemes from the DDH assumption, $\mathtt{KEM}_{\mathsf{DDH}} = (\mathsf{KG}, \mathsf{Encaps}, \mathsf{Decaps})$.

$\mathtt{KEM}_{\mathsf{DDH}}$ has an identical structure with the DDH-based KEM in [JKRS21, Theorem 4], which is essentially based on a DDH-based hash proof system [CS02]. Theorem 3.9 shows that if multi-instance DDH (mDDH) is hard on $\mathbb{G}$, then $\mathtt{KEM}_{\mathsf{DDH}}$ is mPR-PCA secure.

$\mathtt{KEM}_{\mathsf{DDH}}$ is essentially based on a DDH-based hash proof system [CS02]. Theorem 3.9 shows that if multi-instance DDH (mDDH) is hard on $\mathbb{G}$, then $\mathtt{KEM}_{\mathsf{DDH}}$ is mPR-PCA secure.

Theorem 3.9 $\mathtt{KEM}_{\mathsf{DDH}}$ *in Figure 18 is mPR-PCA secure (Definition 3.3) if the* mDDH *problem is hard on* $\mathbb{G}$ *and* H *is modeled as a random oracle. For any adversary* $\mathcal{A}$ *and relation* Rel*, there exists an adversary* $\mathcal{B}$ *such that*

$$\mathsf{Adv}^{\mathsf{mPR\text{-}PCA}}_{\mathtt{KEM}_{\mathsf{DDH}}}(\mathcal{A}, \mu) \leq 2\mathsf{Adv}^{\mathsf{mDDH}}_{\mathbb{G}}(\mathcal{B}) + \frac{4\mu^2 + \mu}{|\mathbb{G}|} + \frac{\mu^2 + q_H^2 + \mu q_H q_{\mathrm{PCO}}}{|\Psi|},$$

where q_H *is the number of* $\mathcal{A}$*'s queries to* H *and* μ *is the number of challenge ciphertexts.*

GAME $\mathbf{G}_0$-$\mathbf{G}_3$

$(X, (x_0, x_1)) \leftarrow \mathsf{KG}$
for $i \in [\mu]$:
  $r_i \xleftarrow{\$} \mathbb{Z}_p$ // $\mathbf{G}_0$-$\mathbf{G}_1$
  $(R_{i,0}, R_{i,1}) := (g_0^{r_i}, g_1^{r_i})$ // $\mathbf{G}_0$-$\mathbf{G}_1$
  $(R_{i,0}, R_{i,1}) \xleftarrow{\$} \mathbb{G}^2$ // $\mathbf{G}_2$-$\mathbf{G}_3$
  $Z_i := X^{r_i}$ // $\mathbf{G}_0$
  $Z_i := R_{i,0}^{x_0} R_{i,1}^{x_1}$ // $\mathbf{G}_1$-$\mathbf{G}_3$
  $\psi_i := H(R_{i,0}, R_{i,1}, Z_i)$ // $\mathbf{G}_0$-$\mathbf{G}_2$
  $\psi_i \xleftarrow{\$} \Psi$ // $\mathbf{G}_3$
  $\mathbf{c}[i] := R_i, \boldsymbol{\psi}[i] := \psi_i$
$b \leftarrow \mathcal{A}^{H, \mathrm{PCO}_{\mathsf{pr}}}(X, \mathbf{c}, \boldsymbol{\psi})$
return b

$\underline{\mathrm{PCO}_{\mathsf{pr}}(R_0, R_1, \psi)}$
if $\exists i \in [\mu]$ s.t. $(R, \psi) = (\mathbf{c}[i], \boldsymbol{\psi}[i])$
  return $\perp$
$\psi' := H(R_0, R_1, R_0^{x_0} R_1^{x_1})$
return $\psi =_? \psi'$

$\underline{H(R_0, R_1, Z)}$
if $\mathcal{L}_{\mathsf{H}}[R_0, R_1, Z] = \perp$
  $\mathcal{L}_{\mathsf{H}}[R_0, R_1, Z] := \psi \xleftarrow{\$} \Psi$
return $\mathcal{L}_{\mathsf{H}}[R_0, R_1, Z]$

Figure 20: Games $\mathbf{G}_0$-$\mathbf{G}_3$ for proving Theorem 3.9.

Proof. The games sequence for proving Theorem 3.9 is given in Figure 20. We assume that there is no collision among all $R_{i,0}$'s, $R_{i,1}$, ψ_i's, and the outputs of H. This assumption adds collision bounds $\frac{4\mu^2}{|\mathbb{G}|} + \frac{\mu^2 + q_H^2}{|\Psi|}$ to the bound of our proof. $\mathbf{G}_0^{\mathcal{A}}$ is equivalent to $\mathsf{mPR\text{-}PCA}^{\mathcal{A},\mu}_{\mathtt{KEM}_{\mathtt{DDH}},0}$, so we have

$$\Pr\left[\mathsf{mPR\text{-}PCA}^{\mathcal{A},\mu}_{\mathtt{KEM}_{\mathtt{DDH}},0} \Rightarrow 1\right] = \Pr\left[\mathbf{G}_0^{\mathcal{A}} \Rightarrow 1\right].$$

Game $\mathbf{G}_1$: We change the generation of Z_i. In this game, we generate $Z := R_{i,0}^{x_0} R_{i,1}^{x_1}$ instead of $Z := X^{r_i}$. This change is conceptual since $R_{i,0}^{x_0} R_{i,1}^{x_1} = X^{r_i}$. We have

$$\Pr\left[\mathbf{G}_0^{\mathcal{A}} \Rightarrow 1\right] = \Pr\left[\mathbf{G}_1^{\mathcal{A}} \Rightarrow 1\right].$$

Game $\mathbf{G}_2$: We generate $R_{i,0}$'s and $R_{i,1}$'s uniformly at random instead of using exponents r_i's. To bound $\left|\Pr\left[\mathbf{G}_1^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathbf{G}_2^{\mathcal{A}} \Rightarrow 1\right]\right|$, we construct a direct reduction $\mathcal{B}$ from MDDH. Reduction $\mathcal{B}$ works as follows: On input $(g_1, (R'_{1,0}, R'_{1,1}), ..., (R'_{\mu,0}, R'_{\mu,1}))$, $\mathcal{B}$ generates X and (x_0, x_1) as in $\mathbf{G}_2$ and sets $(R_{i,0}, R_{i,1}) := (R'_{i,0}, R'_{i,1})$. Then it simulates challenge ciphertexts, $\mathrm{PCO}_{\mathsf{pr}}$ oracle, and H oracle as in $\mathbf{G}_2$. If the input of $\mathcal{B}$ is $\left(g_1, (g_0^{r_i}, g_1^{r_i})_{i\in[\mu]}\right)$, then $\mathcal{B}$ perfectly simulates $\mathbf{G}_1$; Otherwise, the input of $\mathcal{B}$ is $\left(g_1, (g_0^{r_i}, g_1^{r'_i})_{i\in[\mu]}\right)$, which means that $R_{i,0}$ and $R_{i,1}$ are independently and uniformly random and thus $\mathcal{B}$ perfectly simulates $\mathbf{G}_2$. Therefore, we have

$$\Pr\left[\mathbf{G}_1^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathbf{G}_2^{\mathcal{A}} \Rightarrow 1\right] \leq \mathsf{Adv}^{\mathsf{mDDH}}_{\mathbb{G}}(\mathcal{B}).$$

Game $\mathbf{G}_3$: ψ_i's are generated independently and uniformly at random. $\mathcal{A}$ notices this modification only if it queries H on one of Z_i's. By using a standard argument of DDH-based hash proof system (e.g., [JKRS21, Theorem 4]), one can show that $Z_i = R_{i,0}^{x_0} R_{i,1}^{x_1}$ is independently random (if (x_0, x_1) is random). Here we just give a simple explanation: Since $g_1 = g_0^{\omega}$ where $\omega \xleftarrow{\$} \mathbb{Z}_p^*$, X does not reveal information about (x_0, x_1) since $\log_g X = x_0 + \omega x_1$. The responses of $\mathrm{PCO}_{\mathsf{pr}}$ also reveal nothing about (x_0, x_1) since these responses are from the output of RO H. So, (x_0, x_1) is unknown in $\mathcal{A}$'s view. Moreover, let $g_0^{r_{i,0}} := R_{i,0}$ and $g_1^{r_{i,1}} := R_{i,1}$ for some $r_{i,0} \neq r_{i,1}$, we have

$$\begin{bmatrix} \log_g X \\ \log_g Z_i \end{bmatrix} = \mathbf{M} \begin{bmatrix} x_0 \\ x_1 \end{bmatrix}, \text{ where } \mathbf{M} := \begin{bmatrix} 1 & \omega \\ r_{i,0} & \omega r_{i,1} \end{bmatrix}.$$

If $r_{i,0} \neq r_{i,1}$, then $\det(\mathbf{M}) \neq 0$, which implies that Z_i is uniformly random and independent of X. The probability that $r_{i,0} = r_{i,1}$ for some $i \in [\mu]$ is upper bounded by $\frac{\mu}{|\mathbb{G}|}$. By a union bound and a simple hybrid argument, one can also show that the probability that $\mathcal{A}$ queries H on Z_i for some $i \in [\mu]$ is upper bounded

by $\mu q_H q_{\mathrm{PCO}}/|\Psi|$. We have

$$\Pr\left[\mathbf{G}_2^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathbf{G}_3^{\mathcal{A}} \Rightarrow 1\right] \leq \frac{\mu}{|\mathbb{G}|} + \frac{\mu q_H q_{\mathrm{PCO}}}{|\Psi|}.$$

Now $\mathbf{G}_3$ is equivalent to $\mathsf{mPR\text{-}PCA}^{\mathcal{A},\mu}_{\mathtt{KEM}_{\mathtt{DDH}},1}$ if we undo the modifications of $\mathbf{G}_2$ and $\mathbf{G}_1$. By combining all the probability bounds, we have

$$\begin{aligned}\mathsf{Adv}^{\mathsf{mPR\text{-}PCA}}_{\mathtt{KEM}_{\mathtt{DDH}}}(\mathcal{A},\mu) &= \left|\Pr\left[\mathsf{mPR\text{-}PCA}^{\mathcal{A},\mu}_{\mathsf{KEM},0} \Rightarrow 1\right] - \Pr\left[\mathsf{mPR\text{-}PCA}^{\mathcal{A},\mu}_{\mathsf{KEM},1} \Rightarrow 1\right]\right| \\ &\leq 2\mathsf{Adv}^{\mathsf{mDDH}}_{\mathbb{G}}(\mathcal{B}) + \frac{4\mu^2+\mu}{|\mathbb{G}|} + \frac{\mu^2 + q_H^2 + \mu q_H q_{\mathrm{PCO}}}{|\Psi|}.\end{aligned}$$

□

4 Generic Construction for mPR-PCA secure KEM

Besides the direct constructions for mPR-PCA secure KEM in Section 3.3, we construct a mPR-PCA secure KEM tightly and generically from any mPR-CPA secure public-key encryption (PKE) scheme. In combination with the generic construction in the previous section, this shows that SIM-SO-CCA secure PKE can be constructed tightly from any mPR-CPA secure PKE with explainable ciphertexts. Many existing PKE achieve this security notion, which means our construction can be implemented from a variety of assumptions, including LWE.

4.1 Construction

Let $\mathsf{PKE}_0 = (\mathsf{KG}_0, \mathsf{Enc}_0, \mathsf{Dec}_0)$ be a PKE scheme with message space $\mathcal{M}'$, randomness space $\mathcal{R}'$, and ciphertext space $\mathcal{C}'$. Let $\mathcal{S}$ be some efficiently sampleable set (e.g., set of fix-length bit strings). Let $G : \mathcal{M}' \to \mathcal{R}'$ and $H : \mathcal{M}' \times \mathcal{C}' \to \{0,1\}^L$ be hash functions. We construct a KEM (with randomness space $\mathcal{M}'$, ciphertext space $\mathcal{C}'$, and KEM key space $\Psi := \{0,1\}^L$) in Figure 21. Theorem 4.1 shows that if PKE_0 is mPR-CPA secure, and G, H, and F are modeled as random oracles, then KEM is mPR-PCA secure, and the reduction is tight.

The KEM scheme in Figure 21 has the same structure with the $\mathsf{U}^{\perp} \circ \mathsf{T}$ transformation in Hofheinz et al's work [HHK17]. Their transformation focuses on constructing IND-CCA KEM from OW-CPA PKE. In this paper, we focus on ciphertext pseudorandomness, and thus we cannot use their result directly and need a new security reduction to prove Theorem 4.1.

KG	Encaps(pk)	Decaps(sk, c)
01 $(\mathsf{pk},\mathsf{sk}) \leftarrow \mathsf{KG}_0$	03 $\mathsf{m} \xleftarrow{\$} \mathcal{M}'$	08 $\mathsf{m}' := \mathsf{Dec}_0(\mathsf{sk},\mathsf{c})$
02 **return** $(\mathsf{pk},\mathsf{sk})$	04 $r := G(\mathsf{m})$	09 **if** $\mathsf{m}' = \perp$: **return** $\perp$
	05 $\mathsf{c} := \mathsf{Enc}_0(\mathsf{pk},\mathsf{m};r)$	10 $r' := G(\mathsf{m}')$
	06 $\psi := H(\mathsf{m},\mathsf{c})$	11 $\mathsf{c}' := \mathsf{Enc}_0(\mathsf{pk},\mathsf{m}';r')$
	07 **return** (c,ψ)	12 **if** $\mathsf{c}' = \mathsf{c}$: $\psi' := H(\mathsf{m}',\mathsf{c})$
		13 **else return** $\perp$

Figure 21: Our Generic Construction of mPR-PCA secure KEM schemes $\mathsf{KEM} = (\mathsf{KG}, \mathsf{Encaps}, \mathsf{Decaps})$ from PKE scheme $\mathsf{PKE}_0 = (\mathsf{KG}_0, \mathsf{Enc}_0, \mathsf{Dec}_0)$.

Correctness of KEM. Given $(\mathsf{pk},\mathsf{sk}) \leftarrow \mathsf{KG}_0$, for a KEM ciphertext-key pair (c,ψ), there exists m such that $\mathsf{c} = \mathsf{Enc}_0(\mathsf{pk},\mathsf{m};G(\mathsf{m}))$ and $\psi = H(\mathsf{m},\mathsf{c})$. Suppose that $\mathsf{Decaps}(\mathsf{sk},\mathsf{c})$ does not output $\perp$. Let $\mathsf{m}' := \mathsf{Dec}_0(\mathsf{sk},\mathsf{c})$. If $\psi = H(\mathsf{m},\mathsf{c}) \neq H(\mathsf{m}',\mathsf{c}) = \psi'$, then $\mathsf{m} \neq \mathsf{m}'$, which means that we have a m that makes the game $\mathsf{COR}_{\mathsf{PKE}_0}$ return 1. if PKE_0 is $(1-\delta)$-correct, then such event happens within probability less than δ_{PKE_0}. Moreover, if $\mathsf{Decaps}(\mathsf{sk},\mathsf{c})$ outputs $\perp$, then we also have $\mathsf{m}' \neq \mathsf{m}$. Therefore, if PKE_0 is $(1-\delta)$-correct, then KEM is also $(1-\delta)$-correct.

Theorem 4.1 KEM *in Figure 21 is mPR-PCA secure (Definition 3.3) if G, H, and F are modeled as random oracles and* PKE_0 *is $(1-\delta)$-correct and mPR-CPA secure. For any mPR-PCA adversary $\mathcal{A}$, there exists an adversary $\mathcal{B}_{\mathsf{PR}}$ such that:*

$$\mathsf{Adv}^{\mathsf{mPR\text{-}PCA}}_{\mathsf{KEM}}(\mathcal{A},\mu) \le 3\mathsf{Adv}^{\mathsf{mPR\text{-}CPA}}_{\mathsf{PKE}_0}(\mathcal{B}',\mu) + \frac{2\mu(q_G+q_H)}{|\Psi|} + \left(2q_G\cdot\delta + \frac{\mu^2+q_G^2}{|\mathcal{R}|} + \frac{2q_{\mathrm{PCO}}+\mu^2+q_H^2}{|\Psi|}\right),$$

where q_G, q_H, q_F, and q_{PCO} are the numbers of $\mathcal{A}$'s queries to G, H, F, and $\mathrm{PCO}_{\mathsf{pr}}$ oracles, respectively, and μ is the number of challenge ciphertexts.

Proof. Theorem 4.1 is proved by the game sequences in Figure 22. We assume that there is no collision among all r_i's, ψ_i's, and the outputs of random oracles. This assumption adds collision bounds

$$\frac{\mu^2+q_G^2}{|\mathcal{R}|} + \frac{\mu^2+q_H^2}{|\Psi|}$$

to the final bound of our proof. Except for that, we have

$$\Pr\left[\mathsf{mPR\text{-}PCA}^{\mathcal{A},\mu}_{\mathsf{KEM},0} \Rightarrow 1\right] = \Pr\left[\mathbf{G}_0^{\mathcal{A}} \Rightarrow 1\right].$$

Games $\mathbf{G}_0$-$\mathbf{G}_7$

$(\mathsf{pk},\mathsf{sk}) \leftarrow \mathsf{KG}$
for $i \in [\mu]$:
  $\mathsf{m}_i \xleftarrow{\$} \mathcal{M}'$
  $r_i := G(\mathsf{m}_i)$ // $\mathbf{G}_0$-$\mathbf{G}_3$
  $r_i \xleftarrow{\$} \mathcal{R}'$ // $\mathbf{G}_4$-$\mathbf{G}_7$
  $\mathsf{c}_i := \mathsf{Enc}_0(\mathsf{pk},\mathsf{m}_i;r_i)$ // $\mathbf{G}_0$-$\mathbf{G}_5$
  $\mathsf{c}_i \xleftarrow{\$} \mathcal{C}$ // $\mathbf{G}_6$-$\mathbf{G}_7$
  $\psi_i := H(\mathsf{c}_i,\mathsf{m}_i)$ // $\mathbf{G}_0$-$\mathbf{G}_3$
  $\psi_i \xleftarrow{\$} \{0,1\}^L$ // $\mathbf{G}_4$-$\mathbf{G}_7$
  $\mathbf{c}[i] := \mathsf{c}_i, \boldsymbol{\psi}[i] := \psi_i$
$b \leftarrow \mathcal{A}^{G,H,F,\mathrm{PCO}_{\mathsf{pr}}}(\mathsf{pk},\mathbf{c},\boldsymbol{\psi})$
return b

$H(\mathsf{m},\mathsf{c})$

if $\exists i \in [\mu]$ s.t. $\mathsf{m} = \mathsf{m}_i$ // $\mathbf{G}_3$-$\mathbf{G}_4$
  abort // $\mathbf{G}_3$-$\mathbf{G}_4$
if $\mathsf{Enc}_0(\mathsf{pk},\mathsf{m};G(\mathsf{m})) = \mathsf{c}$ // $\mathbf{G}_1$-$\mathbf{G}_6$
  return $h_1(\mathsf{c})$ // $\mathbf{G}_1$-$\mathbf{G}_6$
if $\mathcal{L}_{\mathsf{H}}[\mathsf{m},\mathsf{c}] = \bot$
  $\mathcal{L}_{\mathsf{H}}[\mathsf{m},\mathsf{c}] := \psi \xleftarrow{\$} \Psi$
return $\mathcal{L}_{\mathsf{H}}[\mathsf{m},\mathsf{c}]$

$\mathrm{PCO}_{\mathsf{pr}}(\mathsf{c},\psi)$

$\mathsf{m}' := \mathsf{Dec}_0(\mathsf{sk},\mathsf{c})$ // $\mathbf{G}_0$-$\mathbf{G}_1$,$\mathbf{G}_7$
if $\mathsf{m}' = \bot$ // $\mathbf{G}_0$-$\mathbf{G}_1$,$\mathbf{G}_7$
  return 0 // $\mathbf{G}_0$-$\mathbf{G}_1$,$\mathbf{G}_7$
else // $\mathbf{G}_0$-$\mathbf{G}_1$,$\mathbf{G}_7$
  $r' := G(\mathsf{m}')$ // $\mathbf{G}_0$-$\mathbf{G}_1$,$\mathbf{G}_7$
  $\mathsf{c}' := \mathsf{Enc}_0(\mathsf{pk},\mathsf{m}';r')$ // $\mathbf{G}_0$-$\mathbf{G}_1$,$\mathbf{G}_7$
  if $\mathsf{c}' = \mathsf{c} \wedge \psi = H(\mathsf{m}',\mathsf{c})$ // $\mathbf{G}_0$-$\mathbf{G}_1$,$\mathbf{G}_7$
    return 1 // $\mathbf{G}_0$-$\mathbf{G}_1$,$\mathbf{G}_7$
  else // $\mathbf{G}_0$-$\mathbf{G}_1$,$\mathbf{G}_7$
    return 0 // $\mathbf{G}_0$-$\mathbf{G}_1$,$\mathbf{G}_7$
if $h_1(\mathsf{c}) = \psi$ // $\mathbf{G}_2$-$\mathbf{G}_6$
  return 1 // $\mathbf{G}_2$-$\mathbf{G}_6$
else return 0 // $\mathbf{G}_2$-$\mathbf{G}_6$

$G(\mathsf{m})$

if $\exists i \in [\mu]$ s.t. $\mathsf{m} = \mathsf{m}_i$ // $\mathbf{G}_3$-$\mathbf{G}_4$
  abort // $\mathbf{G}_3$-$\mathbf{G}_4$
if $\mathcal{L}_{\mathsf{G}}[\mathsf{m}] = \bot$
  $\mathcal{L}_{\mathsf{G}}[\mathsf{m}] := r \xleftarrow{\$} \mathcal{R}$
return $\mathcal{L}_{\mathsf{G}}[\mathsf{m}]$

Figure 22: Games $\mathbf{G}_0$-$\mathbf{G}_7$ for proving Theorem 4.1.

Game $\mathbf{G}_1$: Let $h_1 \colon \mathcal{C} \to \Psi$ be an internal random oracle. In $\mathbf{G}_1$, if $\mathcal{A}$ queries H on (m,c) such that $\mathsf{Enc}(\mathsf{pk},\mathsf{m};G(\mathsf{m})) = \mathsf{c}$, then the oracle returns $h_1(\mathsf{c})$ instead of $H(\mathsf{m},\mathsf{c})$.

If $\mathsf{Enc}(\mathsf{pk},\cdot;G(\cdot))$ is an injection, then $\mathbf{G}_1$ is identical to $\mathbf{G}_0$. So, $\mathcal{A}$ cannot distinguish $\mathbf{G}_1$ from $\mathbf{G}_0$ if it cannot find collisions of $\mathsf{Enc}(\mathsf{pk},\cdot;G(\cdot))$. Suppose that $\mathcal{A}$ finds messages $\mathsf{m}'_0 \neq \mathsf{m}'_1$ such that $\mathsf{Enc}(\mathsf{pk},\mathsf{m}'_0;G(\mathsf{m}'_0)) = \mathsf{Enc}(\mathsf{pk},\mathsf{m}'_1;G(\mathsf{m}'_1))$, then $\mathcal{A}$ breaks the correctness of PKE_0 (see Definition 2.13), since it finds a message m ($= \mathsf{m}'_0$ or m'_1) that the decryption of $\mathsf{Enc}(\mathsf{pk},\mathsf{m})$ is not m. Therefore, if PKE_0 is $(1-\delta)$-correct and G is a random oracle, then we have

$$\left|\Pr\left[\mathbf{G}_0^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathbf{G}_1^{\mathcal{A}} \Rightarrow 1\right]\right| \le q_G\cdot\delta.$$

$\underline{\mathcal{B}'_1(\mathsf{pk})}$

01 $b' := 1$
02 **for** $i \in [\mu]$:
03 $\quad \mathbf{m}[i] := \mathsf{m}_i \xleftarrow{\$} \mathcal{M}'$
04 Outputs $\mathbf{m}$
05 Receives $(\mathsf{c}^*_1, ..., \mathsf{c}^*_\mu)$
06 **for** $i \in [\mu]$:
07 $\quad \mathbf{c}[i] := \mathsf{c}^*_i$
08 $\quad \boldsymbol{\psi}[i] \xleftarrow{\$} \Psi$
09 $b \leftarrow \mathcal{A}^{G,H,\textsc{Pco}_{\mathsf{pr}}}(\mathsf{pk}, \mathbf{c}, \boldsymbol{\psi})$
10 **return** b'

$\underline{\textsc{Pco}_{\mathsf{pr}}(\mathsf{c}, \psi)}$

11 **if** $h_1(\mathsf{c}) = \psi$
12 $\quad$ **return** 1
13 **else return** 0

$\underline{H(\mathsf{m}, \mathsf{c})}$

14 **if** $\exists i \in [\mu]$ s.t. $\mathsf{m} = \mathsf{m}_i$
15 $\quad b' := 0$
16 $\quad$ Aborts the game and returns b'
17 **if** $\mathsf{Enc}_0(\mathsf{pk}, \mathsf{m}; G(\mathsf{m})) = \mathsf{c}$
18 $\quad$ **return** $h_1(\mathsf{c})$
19 **if** $\mathcal{L}_{\mathsf{H}}[\mathsf{m}, \mathsf{c}] = \perp$
20 $\quad \mathcal{L}_{\mathsf{H}}[\mathsf{m}, \mathsf{c}] := \psi \xleftarrow{\$} \Psi$
21 **return** $\mathcal{L}_{\mathsf{H}}[\mathsf{m}, \mathsf{c}]$

$\underline{G(\mathsf{m})}$

22 **if** $\exists i \in [\mu]$ s.t. $\mathsf{m} = \mathsf{m}_i$
23 $\quad b' := 0$
24 $\quad$ Aborts the game and returns b'
25 **if** $\mathcal{L}_{\mathsf{G}}[\mathsf{m}] = \perp$
26 $\quad \mathcal{L}_{\mathsf{G}}[\mathsf{m}] := r \xleftarrow{\$} \mathcal{R}$
27 **return** $\mathcal{L}_{\mathsf{G}}[\mathsf{m}]$

Figure 23: mPR-CPA adversary $\mathcal{B}'_1$ in bounding $\Pr[\textsc{Query}_4]$. The highlight codes show how $\mathcal{B}'_1$ use the challenge ciphertexts to simulate $\mathbf{G}_4$. If $\mathcal{A}$ triggers $\textsc{Qry}_4$, then $\mathcal{B}'_1$ returns 0. Otherwise, $\mathcal{B}'_1$ returns 1.

Game $\mathbf{G}_2$: We modify the $\textsc{Pco}_{\mathsf{pr}}$ oracle. When $\mathcal{A}$ queries $\textsc{Pco}_{\mathsf{pr}}(\mathsf{c}, \psi)$, it returns 1 if and only if $\psi = h_1(\mathsf{c})$ (see Lines 30 to 32).

Here we claim that $\textsc{Pco}_{\mathsf{pr}}$'s output distribution in $\mathbf{G}_2$ is the same as in $\mathbf{G}_1$ except with negligible probability. To prove this, for any $\mathcal{A}$'s $\textsc{Pco}_{\mathsf{pr}}$ queries (c, ψ), let $\mathsf{m}' := \mathsf{Dec}_0(\mathsf{sk}, \mathsf{c})$, and we consider two cases:

- $\mathsf{m}' = \perp$ or $\mathsf{c} \neq \mathsf{Enc}_0(\mathsf{pk}, \mathsf{m}'; G(\mathsf{m}'))$. $\textsc{Pco}_{\mathsf{pr}}$ in $\mathbf{G}_1$ always returns 0 in this case. In $\mathbf{G}_2$, $\textsc{Pco}_{\mathsf{pr}}$ returns 1 only if $h_1(\mathsf{c}) = \psi$. So, to make $\textsc{Pco}_{\mathsf{pr}}$ in $\mathbf{G}_2$ behaves differently, $\mathcal{A}$ needs to know $h_1(\mathsf{c})$. However, the only way for $\mathcal{A}$ to get $h_1(\mathsf{c})$ is to query $H(\mathsf{m}', \mathsf{c})$ where $\mathsf{c} = \mathsf{Enc}_0(\mathsf{pk}, \mathsf{m}'; G(\mathsf{m}'))$. Therefore, in this case, $h_1(\mathsf{c})$ is uniformly random in $\mathcal{A}$'s view, and $\mathcal{A}$ makes $\textsc{Pco}_{\mathsf{pr}}$ returns 1 in $\mathbf{G}_2$ with probability at most $q_{\textsc{Pco}}/|\Psi|$.
- $\mathsf{m}' \neq \perp$ and $\mathsf{c} = \mathsf{Enc}_0(\mathsf{pk}, \mathsf{m}'; G(\mathsf{m}'))$. $\textsc{Pco}_{\mathsf{pr}}$ has the same outputs in $\mathbf{G}_1$ and $\mathbf{G}_2$, since in this case, $H(\mathsf{m}', \mathsf{c}) = h_1(\mathsf{c})$.

So, we have

$$\left| \Pr\left[\mathbf{G}_1^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathbf{G}_2^{\mathcal{A}} \Rightarrow 1\right] \right| \leq \frac{q_{\textsc{Pco}}}{|\Psi|}.$$

Game $\mathbf{G}_3$: We add two abort conditions in G and H. If $\mathcal{A}$ queries G or H on m where $\mathsf{m} = \mathsf{m}_i$ for some $i \in [\mu]$, then the game aborts. Let $\textsc{Query}$ be this querying event and $\textsc{Query}_j$ be the event that $\textsc{Query}$ happens in $\mathbf{G}_j$. The adversary cannot detect this modification unless it triggers $\textsc{Query}_3$, so we have

$$\left| \Pr\left[\mathbf{G}_2^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathbf{G}_3^{\mathcal{A}} \Rightarrow 1\right] \right| \leq \Pr\left[\textsc{Query}_3\right].$$

Game $\mathbf{G}_4$: We change the generation of randomnesses and KEM keys. In $\mathbf{G}_4$ r_i's and ψ_i's are generated at uniformly random and independent of m_i's (see Lines 05 and 09). This modification does not change $\mathcal{A}$'s view if $\textsc{Query}$ does not happen in $\mathbf{G}_3$ and $\mathbf{G}_4$. Therefore, we have

$$\Pr\left[\mathbf{G}_3^{\mathcal{A}} \Rightarrow 1\right] = \Pr\left[\mathbf{G}_4^{\mathcal{A}} \Rightarrow 1\right], \Pr\left[\textsc{Query}_3\right] = \Pr\left[\textsc{Query}_4\right].$$

Since in $\mathbf{G}_4$, c_i can be viewed as being generated via $\mathsf{c}_i \leftarrow \mathsf{Enc}_0(\mathsf{pk}, \mathsf{c}_i)$ (without specifying the independent and uniform randomness r_i, we can construct an adversary $\mathcal{B}'_1$ that simulates $\mathbf{G}_4$ for $\mathcal{A}$ to bound $\Pr[\textsc{Query}_4]$. $\mathcal{B}'_1$ is shown in Figure 23.

If $\mathcal{B}'_1$ is playing the game $\mathsf{mPR\text{-}CPA}^{\mathcal{B}'_1,\mu}_{\mathsf{PKE}_0,0}$, then it perfectly simulates $\mathbf{G}_4$, and it outputs 1 if QUERY_4 does not happen. So we have

$$\Pr\left[\mathsf{mPR\text{-}CPA}^{\mathcal{B}'_1,\mu}_{\mathsf{PKE}_0,0} \Rightarrow 1\right] = 1 - \Pr\left[\text{QUERY}_4\right].$$

If $\mathcal{B}'_1$ is playing the game $\mathsf{mPR\text{-}CPA}^{\mathcal{B}'_1,\mu}_{\mathsf{PKE}_0,1}$, then $(\mathsf{c}^*_1, ..., \mathsf{c}^*_\mu)$ are uniformly random and independent of $(\mathsf{m}_1, ..., \mathsf{m}_\mu)$, which means that $\mathcal{B}'_1$ outputs 1 with probability at least $1 - \frac{\mu(q_G+q_H)}{|\Psi|}$. Therefore, we have

$$\Pr\left[\text{QUERY}_4\right] \leq \mathsf{Adv}^{\mathsf{mPR\text{-}CPA}}_{\mathsf{PKE}_0}(\mathcal{B}'_1, \mu) + \frac{\mu(q_G + q_H)}{|\Psi|}.$$

Game $\mathbf{G}_5$: We undo the abort conditions in H and G. Similar to the argument in bounding $\mathbf{G}_3$ and $\mathbf{G}_4$, if QUERY_4 does not happen, then $\mathbf{G}_5$ proceeds identically with $\mathbf{G}_4$. Therefore, we have

$$\left|\Pr\left[\mathbf{G}^{\mathcal{A}}_4 \Rightarrow 1\right] - \Pr\left[\mathbf{G}^{\mathcal{A}}_5 \Rightarrow 1\right]\right| \leq \Pr\left[\text{QUERY}_4\right] \leq \mathsf{Adv}^{\mathsf{mPR\text{-}CPA}}_{\mathsf{PKE}_0}(\mathcal{B}'_1, \mu) + \frac{\mu(q_G + q_H)}{|\Psi|}.$$

Game $\mathbf{G}_6$: We change the generation of challenge ciphertext. In $\mathbf{G}_6$, c is generated by uniform sampling from $\mathcal{C}'$ instead of by encrypting m_i (see Line 07).

Since in $\mathbf{G}_6$, the game simulator does not need sk to simulate PCO_{pr}, r_i's and ψ_i's are independent of m_i's, and there is no abort event related to m_i's, we can construct a direct reduction to $\mathsf{mPR\text{-}CPA}$ security of PKE_0. That is, there exists an $\mathsf{mPR\text{-}CPA}$ adversary $\mathcal{B}'_2$ such that

$$\left|\Pr\left[\mathbf{G}^{\mathcal{A}}_5 \Rightarrow 1\right] - \Pr\left[\mathbf{G}^{\mathcal{A}}_6 \Rightarrow 1\right]\right| \leq \mathsf{Adv}^{\mathsf{mPR\text{-}CPA}}_{\mathsf{PKE}_0}(\mathcal{B}'_2, \mu).$$

Game $\mathbf{G}_7$: We undo all the modifications did in $\mathbf{G}_1$ and $\mathbf{G}_2$. Now $\mathbf{G}^{\mathcal{A}}_7$ is identical to the game $\mathsf{mPR\text{-}PCA}^{\mathcal{A},\mu}_{\mathsf{KEM},1}$. We have

$$\left|\Pr\left[\mathbf{G}^{\mathcal{A}}_6 \Rightarrow 1\right] - \Pr\left[\mathbf{G}^{\mathcal{A}}_7 \Rightarrow 1\right]\right| \leq q_G \cdot \delta + \frac{q_{\text{PCO}}}{|\Psi|},\ \Pr\left[\mathbf{G}^{\mathcal{A}}_7 \Rightarrow 1\right] = \Pr\left[\mathsf{mPR\text{-}PCA}^{\mathcal{A},\mu}_{\mathsf{KEM},1} \Rightarrow 1\right].$$

By combining all the probability bounds and viewing $\mathcal{B}'_1$ and $\mathcal{B}'_2$ as an adversary $\mathcal{B}'$, we have

$$\begin{aligned}\mathsf{Adv}^{\mathsf{mPR\text{-}PCA}}_{\mathsf{KEM}}(\mathcal{A}, \mu) &:= \left|\Pr\left[\mathsf{mPR\text{-}PCA}^{\mathcal{A},\mu}_{\mathsf{KEM},0} \Rightarrow 1\right] - \Pr\left[\mathsf{mPR\text{-}PCA}^{\mathcal{A},\mu}_{\mathsf{KEM},1} \Rightarrow 1\right]\right| \\ &\leq 3\mathsf{Adv}^{\mathsf{mPR\text{-}CPA}}_{\mathsf{PKE}_0}(\mathcal{B}', \mu) + \frac{2\mu(q_G + q_H)}{|\Psi|} + \left(2q_G \cdot \delta + \frac{\mu^2 + q_G^2}{|\mathcal{R}|} + \frac{2q_{\text{PCO}} + \mu^2 + q_H^2}{|\Psi|}\right)\end{aligned}$$

□

4.2 An Instantiation from LWE

We show that Regev's lattice-based encryption [Reg05] have mPR-CPA security and the explainable property (Definition 3.4). Combining with the generic construction from Section 3, it yields the first tightly SIM-SO-CCA secure PKE from lattices.

SCHEME. Our construction is the multiple-message-bit version of Regev encryption [Reg05, PVW08]. The message space and ciphertext space of the scheme are $\{0,1\}^\ell$ and $\mathbb{Z}_q^n \times \mathbb{Z}_q^\ell$, respectively. In lattice-based encryption schemes, a message $\mathsf{m} \in \{0,1\}^\ell$ has to be encoded on encryption and decoded on decryption. We firstly define the following algorithms:

- Algorithm $\mathsf{Encode}(\mathsf{m})$ computes a vector $\mathsf{m}^\top \in \mathbb{Z}_q^\ell$. The ith coordinate of $\mathsf{m}^\top$ is given as $\lfloor q/2 \rceil \cdot \mathsf{m}_i$ for each $i \in [\ell]$.

- Algorithm $\mathsf{Decode}(\mathsf{m}^\top)$ computes a message $\mathsf{m} \in \{0,1\}^\ell$ by componentwise rounding. That is, for all $i \in [\ell]$, it sets $\mathsf{m}_i = 0$ if m_i is closer to 0 than to $\lfloor q/2 \rceil$. Otherwise, it sets $\mathsf{m}_i = 1$.

The scheme is shown in Figure 24. We set up the parameters n, m, q (prime), $t, g \in \mathbb{N}, s, s', s'' \in \mathbb{R}, s, s', s'' > 0$ used in the scheme such that they satisfy the following conditions:

- $n = \Theta(\lambda)$, $m \geq 2(n+\ell)\log q$ (for Lemma 2.10)
- $s, s' \geq \omega(\sqrt{\log m})$ (for Lemmata 2.10 and 2.11)
- $ss'm \leq q/4$ (for correctness)

These conditions can be satisfied. For example, given a security parameter λ and message length $\ell = n$, a non-optimized instantiation would be $n := \lambda$, $n^3 < q \leq n^4$, $m := 4n\log q$, and $s := s' := \log m$.

$\underline{\mathsf{KG}}$
$\mathsf{sk} := \mathbf{S} \xleftarrow{\$} \mathbb{Z}_q^{n\times\ell},\ \mathbf{E} \leftarrow D_{\mathbb{Z},s}^{m\times\ell}$
$\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n\times m}$
$\mathbf{Y} := \mathbf{S}^\top\mathbf{A} + \mathbf{E}^\top \in \mathbb{Z}_q^{\ell\times m}$
$\mathsf{pk} := (\mathbf{A}, \mathbf{Y})$
return $(\mathsf{pk}, \mathsf{sk})$

$\underline{\mathsf{Enc}(\mathsf{pk} = (\mathbf{A}, \mathbf{Y}), \mathsf{m})}$
$\mathbf{x} \leftarrow D_{\mathbb{Z},s'}^m$
$\mathbf{c} := \mathbf{A}\mathbf{x}$
$\mathbf{v} := \mathbf{Y}\mathbf{x} + \mathsf{Encode}(\mathsf{m})^\top$
return $c := (\mathbf{c}, \mathbf{v})$

$\underline{\mathsf{Dec}(\mathsf{sk} = \mathbf{S}, c = (\mathbf{c}, \mathbf{v}))}$
$\mathbf{m} := \mathbf{v} - \mathbf{S}^\top\mathbf{c}$
return $\mathsf{Decode}(\mathbf{m}^\top)$

Figure 24: The LWE-based mPR-CPA secure PKE scheme $\mathsf{PKE}_{\mathsf{LWE}} := (\mathsf{KG}, \mathsf{Enc}, \mathsf{Dec})$. The scheme is the Regev encryption scheme [Reg05] extended to multiple message bits as in [PVW08].

Lemmata 4.2 and 4.3 show that $\mathsf{PKE}_{\mathsf{LWE}}$ has negligible correctness error and is mPR-CPA secure based on the LWE assumption.

Lemma 4.2 *The scheme $\mathsf{PKE}_{\mathsf{LWE}}$ in Figure 24 is $(1-\delta)$-correct, for negligible δ.*

Proof. We follow the standard arguments in [Reg05, GPV08] to prove the correctness. One can easily see that decryption Dec correctly decrypt a ciphertext $(\mathbf{A}\mathbf{x}, \mathbf{Y}\mathbf{x} + \mathsf{Encode}(\mathsf{m})^\top)$ as long as $|\mathbf{e}^\top\mathbf{x}| < q/4$ for any column $\mathbf{e}$ of $\mathbf{E}$. By Lemma 2.11 and our parameter setting about s, s', m, and q, we have

$$|\mathbf{e}^\top\mathbf{x}| \leq \|\mathbf{e}\|\|\mathbf{x}\| \leq ss'm < q/4$$

with overwhelming probability. □

Lemma 4.3 *If the $\mathsf{LWE}_{n,m,q,D_{\mathbb{Z},s}}$ assumption holds, then the scheme $\mathsf{PKE}_{\mathsf{LWE}}$ is mPR-CPA secure. Namely, for any algorithm $\mathcal{A}$, there is a algorithm $\mathcal{B}$ such that the running time of $\mathcal{B}$ is about that of $\mathcal{A}$ and*

$$\mathsf{Adv}^{\mathsf{mPR\text{-}CPA}}_{\mathsf{PKE}_{\mathsf{LWE}}}(\mathcal{A}) \leq \ell \cdot \mathsf{Adv}^{\mathsf{LWE}_{n,m,q,D_{\mathbb{Z},s}}}(\mathcal{B}) + \mathsf{negl}(\lambda)$$

Proof. The statement follows from the LWE assumption and Lemma 2.10. We firstly apply the LWE assumption to each row of matrix $\mathsf{pk} = \mathbf{Y}$ (totally ℓ LWE instances). Then, since $\mathbf{A}$ and $\mathbf{Y}$ are uniformly random, we can apply Lemma 2.10 to argue that $(\mathbf{A}\mathbf{x}, \mathbf{Y}\mathbf{x})$ is statistically close to uniform, which also means that the ciphertext $(\mathbf{A}\mathbf{x}, \mathbf{Y}\mathbf{x} + \mathsf{Encode}(\mathsf{m})^\top)$ is statistically close to uniform. □

Explainable Ciphertexts. We show that $\mathsf{PKE}_{\mathsf{LWE}}$ is $(\mathbb{Z}_q^n, \mathbb{Z}_q^\ell)$-explainable (namely, the ciphertext space of $\mathsf{PKE}_{\mathsf{LWE}}$ is $(\mathbb{Z}_q^n, \mathbb{Z}_q^\ell)$), since one can simply let $\mathsf{Sample}_{(\mathbb{Z}_q^n, \mathbb{Z}_q^\ell)}$ and $\mathsf{Sample}^{-1}_{(\mathbb{Z}_q^n, \mathbb{Z}_q^\ell)}$ be identity functions and the randomness domain $\mathcal{R}_{\mathsf{Sample}_{(\mathbb{Z}_q^n, \mathbb{Z}_q^\ell)}}$ be the same as the ciphertext space $(\mathbb{Z}_q^n, \mathbb{Z}_q^\ell)$. Namely, $\mathsf{Sample}_{(\mathbb{Z}_q^n, \mathbb{Z}_q^\ell)}$ works as follows: On input $(\mathbf{c}, \mathbf{v}) \in_\$ (\mathbb{Z}_q^n, \mathbb{Z}_q^\ell)$, it simply outputs $(\mathbf{c}, \mathbf{v})$. Similarly, $\mathsf{Sample}^{-1}_{(\mathbb{Z}_q^n, \mathbb{Z}_q^\ell)}(\mathbf{c}, \mathbf{v})$ outputs $(\mathbf{c}, \mathbf{v})$. Such $\mathsf{Sample}_{(\mathbb{Z}_q^n, \mathbb{Z}_q^\ell)}$ and $\mathsf{Sample}^{-1}_{(\mathbb{Z}_q^n, \mathbb{Z}_q^\ell)}$ satisfy Definition 3.4.

5 Generic Construction II: SO from Lossy Encryption

In this section, we prove tight SO security of Fujisaki-Okamoto's (FO) transformation [FO13] assuming that the underlying PKE is a lossy encryption [BHY09]. More precisely, if the lossy encryption scheme has efficient opener (e.g., the one from [HJR16]), then FO is SIM-SO-CCA-secure. If the lossy encryption does not have efficient opener (e.g., the one from hash proof systems [HLOV11, BHY09]), then FO is IND-SO-CCA secure.

5.1 Definition of Lossy Encryption

Before we turn to the construction, we first recall the notion of lossy encryption. We slightly modify the definition of lossy encryption in [BHY09].

Definition 5.1 (Lossy Encryption). Let $\mathsf{PKE}_0 := (\mathsf{KG}_0, \mathsf{Enc}_0, \mathsf{Dec}_0)$ be a PKE scheme with message space $\mathcal{M}$ and randomness space $\mathcal{R}$. PKE_0 is *lossy* if it has the following properties:

- PKE_0 is correct according to Definition 2.13.
- *Key indistinguishability:* We say PKE_0 has key indistinguishability if there is an algorithm LKG such that, for any adversary $\mathcal{B}$, the advantage function

$$\mathsf{Adv}^{\mathsf{key\text{-}ind}}_{\mathsf{PKE}_0}(\mathcal{B}) := |\Pr[\mathcal{B}(\mathsf{pk}) \Rightarrow 1] - \Pr[\mathcal{B}(\mathsf{pk}') \Rightarrow 1]|$$

  is negligible, where $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KG}_0$ and $(\mathsf{pk}', \mathsf{td}) \leftarrow \mathsf{LKG}$.
- *Lossiness:* Let m, m' be arbitrary messages in $\mathcal{M}'$, the statistical distance between the two following distributions is negligible.

$$D := \left\{ (\mathsf{pk}', \mathsf{td}, c) \;\middle|\; (\mathsf{pk}', \mathsf{td}) \leftarrow \mathsf{LKG}, c \leftarrow \mathsf{Enc}_0(\mathsf{pk}', \mathsf{m}) \right\}$$
$$D' := \left\{ (\mathsf{pk}', \mathsf{td}, c') \;\middle|\; (\mathsf{pk}', \mathsf{td}) \leftarrow \mathsf{LKG}, c' \leftarrow \mathsf{Enc}_0(\mathsf{pk}', \mathsf{m}') \right\}$$

- *Openability:* Let $(\mathsf{pk}', \mathsf{td}) \leftarrow \mathsf{LKG}$, m and m' be arbitrary messages, and r be arbitrary randomness. For ciphertext $c := \mathsf{Enc}_0(\mathsf{pk}', \mathsf{m}; r)$, there exists an algorithm open such that $\mathsf{open}(\mathsf{td}, \mathsf{pk}', c, r, \mathsf{m}')$ outputs a uniformly random $r' \in \mathcal{R}$ such that $c = \mathsf{Enc}_0(\mathsf{pk}', \mathsf{m}'; r')$. Here open can be inefficient.

We extend the above lossiness definition to a multi-challenge setting. The multi-challenge lossiness is implied by the single-challenge one using hybrid argument. Since it is only a statistical property, the hybrid argument will not affect tightness of the computational advantage.

Definition 5.2 (Multi-Challenge Lossiness). PKE_0 has multi-challenge lossiness if for any messages $\mathsf{m}_1, \mathsf{m}'_1, \ldots, \mathsf{m}_\mu, \mathsf{m}'_\mu$ in $\mathcal{M}'$, the statistical distance between the distributions

$$D := \left\{ (\mathsf{pk}', \mathsf{td}, c_1, \ldots, c_\mu) \;\middle|\; (\mathsf{pk}', \mathsf{td}) \leftarrow \mathsf{LKG}, \forall i \in [\mu] : c_i = \mathsf{Enc}_0(\mathsf{pk}', \mathsf{m}_i) \right\}$$

and

$$D' := \left\{ (\mathsf{pk}', \mathsf{td}, c'_1, \ldots, c'_\mu) \;\middle|\; (\mathsf{pk}', \mathsf{td}) \leftarrow \mathsf{LKG}, \forall i \in [\mu] : c'_i = \mathsf{Enc}_0(\mathsf{pk}', \mathsf{m}'_i) \right\}$$

is negligible. We denote the distance by $\varepsilon^{\mathsf{m\text{-}enc\text{-}los}}_{\mathsf{PKE}_0}$.

We require γ-spreadness for our construction.

Definition 5.3 (γ-Spreadness). Let $\mathsf{PKE}_0 := (\mathsf{KG}_0, \mathsf{Enc}_0, \mathsf{Dec}_0)$ be a PKE scheme with message space $\mathcal{M}$, randomness space $\mathcal{R}$, and ciphertext space $\mathcal{C}$. Given a pk, we define a function

$$\mathcal{E}(\mathsf{pk}) := \max_{\mathsf{m} \in \mathcal{M}, \mathsf{c} \in \mathcal{C}} \Pr_{r \xleftarrow{\$} \mathcal{R}}[\mathsf{c} = \mathsf{Enc}_0(\mathsf{pk}, \mathsf{m}; r)]$$

We say PKE_0 is γ-spread if

$$\mathop{\mathbb{E}}_{(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KG}_0}[\mathcal{E}(\mathsf{pk})] \leq 2^{-\gamma}$$

5.2 From Lossy Encryption to SO

Let $\mathsf{PKE}_0 := (\mathsf{KG}_0, \mathsf{Enc}_0, \mathsf{Dec}_0)$ be a lossy encryption scheme with message space $\mathcal{M}'$ and randomness space $\mathcal{R}'$. Let $H : \mathcal{M}' \to \mathcal{M}$ and $G : \mathcal{M}' \times \mathcal{M} \to \mathcal{R}'$ be two hash functions. The FO transformation $\mathtt{FO} := (\mathsf{KG}, \mathsf{Enc}, \mathsf{Dec})$ is defined in Figure 25. Here we use the one-time pad as the symmetric part to encrypt the message. The randomness space of $\mathtt{FO}$ is $\mathcal{R}'$.

$\underline{\mathsf{KG}}$	$\underline{\mathsf{Enc}(\mathsf{pk}, \mathsf{m})}$	$\underline{\mathsf{Dec}(\mathsf{sk}, (e, \mathsf{d}))}$
01 $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KG}_0$	03 $r \leftarrow \mathcal{M}'$	09 $\mathsf{m}' := \bot$
02 **return** $(\mathsf{pk}, \mathsf{sk})$	04 $K := H(r)$	10 $r' := \mathsf{Dec}_0(\mathsf{sk}, e)$
	05 $\mathsf{d} := K \oplus \mathsf{m}$	11 $R' := G(r', \mathsf{d}), K' := H(r')$
	06 $R := G(r, \mathsf{d})$	12 **if** $e = \mathsf{Enc}_0(\mathsf{pk}, r'; R')$
	07 $e := \mathsf{Enc}_0(\mathsf{pk}, r; R)$	13 $\quad \mathsf{m}' := \mathsf{d} \oplus K'$
	08 **return** (e, d)	14 **return** m'

Figure 25: Fujisaki-Okamoto's transformation $\mathtt{FO}$ with lossy encryption PKE_0.

As shown in [HHK17], if PKE_0 is $(1-\delta)$-correct and G is modeled as a random oracle, then $\mathtt{FO}$ is $(1 - q_G\delta)$-correct where q_G is the number of queries to G.

We show that $\mathtt{FO}$ is tightly SIM-SO-CCA secure if the underlying lossy encryption is efficiently openable (cf. Theorem 5.4). Otherwise, it is tightly IND-SO-CCA secure (cf. Theorem 5.5).

SIM-SO-CCA Security. We show SIM-SO-CCA security, assuming efficient openability.

Theorem 5.4 $\mathtt{FO}$ *in Figure 25 is SIM-SO-CCA secure if G and H are modeled as random oracles, and PKE_0 is a lossy encryption with efficient openability and γ-spreadness. Concretely, for any SIM-SO-CCA adversary $\mathcal{A}$ and relation Rel, there exists a simulator $\mathcal{S}$ and $\mathcal{B}$ such that:*

$$\mathsf{Adv}^{\mathsf{SIM\text{-}SO\text{-}CCA}}_{\mathtt{FO}}(\mathcal{A}, \mathcal{S}, \mu, \mathsf{Rel}) \le \mathsf{Adv}^{\mathsf{key\text{-}ind}}_{\mathsf{PKE}_0}(\mathcal{B}) + 2\varepsilon^{\mathsf{m\text{-}enc\text{-}los}}_{\mathsf{PKE}_0} + \frac{\mu n_{\mathrm{DEC}}}{2^\gamma} + \frac{2n_H^2}{|\mathcal{M}|} + \frac{2n_G^2}{|\mathcal{R}'|} + \frac{4\mu^2 + 5\mu(q_G + q_H)}{|\mathcal{M}'|},$$

where q_H, q_G, and n_{DEC} are the numbers of $\mathcal{A}$'s queries to G, H, and Dec*, respectively, μ is the number of challenge ciphertexts, and $n_G = \mu + n_{\mathrm{DEC}} + q_H$ and $n_H = \mu + n_{\mathrm{DEC}} + q_G$ are the number of queries (including the simulator) to G and H, respectively.*

Proof. We prove Theorem 5.4 by a game sequence as shown in Figure 26. Game $\mathbf{G}_0$ is the original game except that we use lazy sampling to simulate ROs G and H. We assume that, from $\mathbf{G}_0$ to $\mathbf{G}_9$, there is no collision among r_i's and the outputs of H and G. Let n_G and n_H be the number of queries to G and H, respectively. By the security game in Figure 26, $n_G = \mu + n_{\mathrm{DEC}} + q_G$ and $n_H = \mu + n_{\mathrm{DEC}} + q_H$. We have

$$\left|\Pr\left[\mathsf{REAL\text{-}SO\text{-}CCA}^{\mathcal{A}}_{\mathtt{FO}} \Rightarrow 1\right] - \Pr\left[\mathbf{G}_0^{\mathcal{A}} \Rightarrow 1\right]\right| \le \frac{n_H^2}{|\mathcal{M}|} + \frac{\mu^2}{|\mathcal{M}'|} + \frac{n_G^2}{|\mathcal{R}'|}.$$

Game $\mathbf{G}_1$: We modify Dec. Instead of using sk to simulate Dec, we use the randomness recorded in G to decrypt given ciphertexts (see Lines 40 to 42). This simulation method is exact the same as the one in the original FO transformation [FO13]. By the argument in [FO13], if PKE_0 is γ-spread, then we have

$$\left|\Pr\left[\mathbf{G}_0^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathbf{G}_1^{\mathcal{A}} \Rightarrow 1\right]\right| \le \frac{\mu \cdot n_{\mathrm{DEC}}}{2^\gamma}.$$

Game $\mathbf{G}_2$: We switch the public key to lossy mode by $(\mathsf{pk}', \mathsf{td}) \xleftarrow{\$} \mathsf{LKG}$. Since in this game the decryption oracle are simulated without using sk, we can simulate $\mathbf{G}_2$ with pk'. By the key indistinguishability of the lossy encryption, we get

$$\left|\Pr\left[\mathbf{G}_1^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathbf{G}_2^{\mathcal{A}} \Rightarrow 1\right]\right| \le \mathsf{Adv}^{\mathsf{key\text{-}ind}}_{\mathsf{PKE}_0}(\mathcal{B}_0).$$

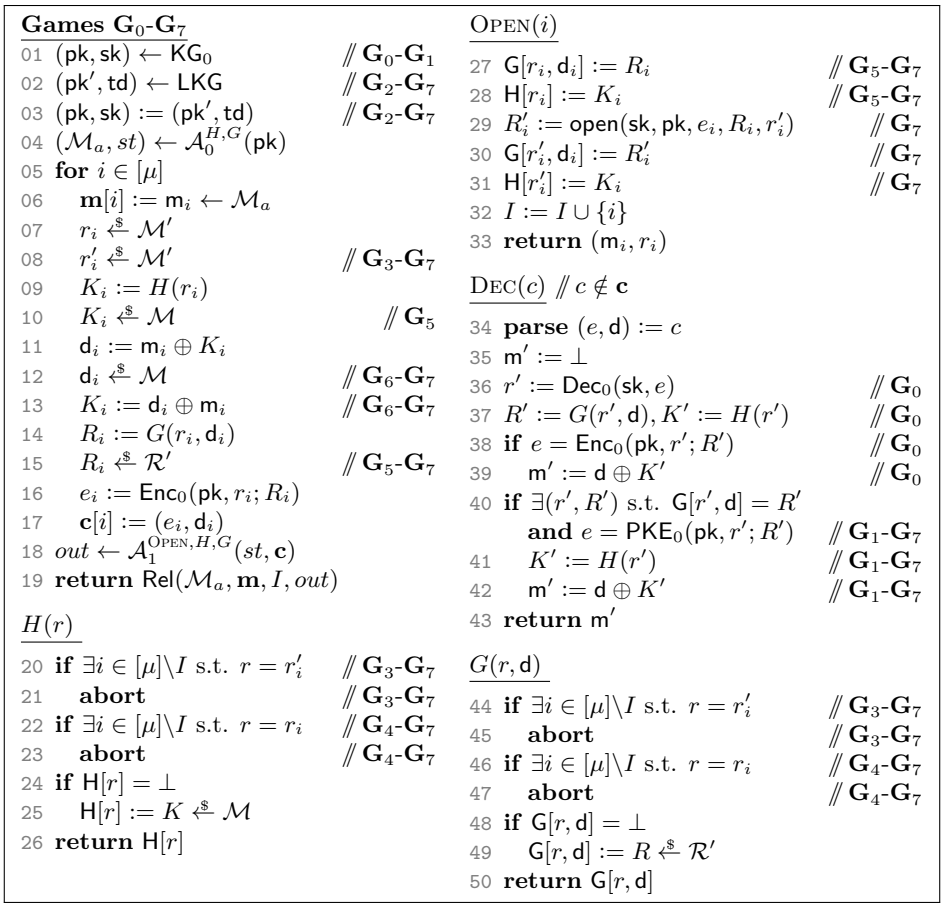

Figure 26: Games $\mathbf{G}_0$-$\mathbf{G}_7$ for proving Theorem 5.4.

Game $\mathbf{G}_3$: This is a preparation step. We choose some internal randomness r'_i for the opening queries in the next games. We abort $\mathbf{G}_3$ if $\mathcal{A}$ queries either H or G with r'_i before opening $\mathbf{c}[i]$. Since r'_i (for $i \in [\mu]$) are internal and never revealed to $\mathcal{A}$, the probability that $\mathcal{A}$ queries r'_i for some i is $\frac{q_H+q_G}{|\mathcal{M}'|}$. We also require all r'_i's are different. By the union bound and collision bound, we have

$$\left|\Pr\left[\mathbf{G}_2^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathbf{G}_3^{\mathcal{A}} \Rightarrow 1\right]\right| \leq \frac{\mu \cdot (q_H + q_G)}{|\mathcal{M}'|} + \frac{\mu^2}{|\mathcal{M}'|}.$$

Game $\mathbf{G}_4$: We further modify the abort rules in H and G. If $\mathcal{A}$ queries H or G with r_i and $\mathbf{c}[i]$ is unopened, then $\mathbf{G}_4$ aborts. Let $\textsc{Query}_j$ be the event that such abort event occurs in $\mathbf{G}_j$, i.e., $\mathcal{A}$ queries H (resp., G) on r_i (resp, (r_i, d_i)) where $\mathbf{c}[i]$ is unopened. Then we have

$$\left|\Pr\left[\mathbf{G}_3^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathbf{G}_4^{\mathcal{A}} \Rightarrow 1\right]\right| \leq \Pr\left[\textsc{Query}_4\right].$$

Here we cannot bound $\Pr[\textsc{Query}_4]$ directly yet, since all e_i are correlated to $H(r_i)$ and $G(r_i, \mathsf{d}_i)$. We will bound $\Pr[\textsc{Query}_4]$ later. Our strategy for that is to decouple e_i with $G(r_i, \mathsf{d}_i)$ and $H(r_i)$. In the end, $\mathcal{A}$ can query r_i for $i \in [\mu]\backslash I$ (i.e., $\mathbf{c}[i]$ is unopened) with negligible probability.

Game $\mathbf{G}_5$: We modify the generation of R_i and K_i. In this game, R_i and K_i are chosen uniformly, instead of using H and G. Moreover, upon $\textsc{Open}(i)$, we set $H(r_i) := K_i$ and $G(r_i, \mathsf{d}_i) := R_i$. By the abort

rules in G and H, $\mathcal{A}$ can learn neither $H(r_i)$ nor $G(r_i, \mathsf{d}_i)$ before opening $\mathbf{c}[i]$. Thus, we have

$$\Pr\left[\mathbf{G}_4^{\mathcal{A}} \Rightarrow 1\right] = \Pr\left[\mathbf{G}_5^{\mathcal{A}} \Rightarrow 1\right],\ \Pr\left[\textsc{Query}_4\right] = \Pr\left[\textsc{Query}_5\right].$$

Game $\mathbf{G}_6$: We further modify the computation of d_i and K_i. In this game, d_i are chosen uniformly at random, and K_i are computed as $K_i := \mathsf{d}_i \oplus \mathsf{m}_i$. In $\mathbf{G}_5$, K_i is distributed uniformly at random. Hence, this modification is only conceptual, and we get

$$\Pr\left[\mathbf{G}_5^{\mathcal{A}} \Rightarrow 1\right] = \Pr\left[\mathbf{G}_6^{\mathcal{A}} \Rightarrow 1\right],\ \Pr\left[\textsc{Query}_5\right] = \Pr\left[\textsc{Query}_6\right].$$

Game $\mathbf{G}_7$: Upon $\textsc{Open}(i)$, we compute the opened randomness R_i' with respect to r_i' and e_i using the open algorithm (see Line 29), and then set $G(r_i', \mathsf{d}_i) := R_i'$ and $H(r_i') := K_i$. Looking ahead, this modification is necessary for the later modification that $\mathbf{c}[i] = (e_i, \mathsf{d}_i)$ can be claimed to r_i'. $\mathcal{A}$ detects this modification if it queries $H(r_i')$ or $G(r_i', \mathsf{d}_i)$. This modification does not affect the occurring probability of $\textsc{Query}_7$, since r_i' is perfectly hidden and independent of r_i. Therefore,

$$\left|\Pr\left[\mathbf{G}_6^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathbf{G}_7^{\mathcal{A}} \Rightarrow 1\right]\right| \leq \frac{\mu(q_G + q_H)}{|\mathcal{M}'|},\ \Pr\left[\textsc{Query}_6\right] = \Pr\left[\textsc{Query}_7\right].$$

In $\mathbf{G}_7$, we have the following observation: Before $\mathcal{A}$ opens i, R_i are independent of r_i, r_i', K_i, and d_i, so e_i can be viewed as a ciphertext that $e_i := \mathsf{PKE}_0(\mathsf{pk}', r_i; R_i)$ where the randomness R_i is sampled independently and uniformly. Therefore, by the *lossiness* of pk', we can replace $\mathsf{PKE}_0(\mathsf{pk}', r_i; R_i)$ as another ciphertext $\mathsf{PKE}_0(\mathsf{pk}', r_i''; R_i'')$ where r_i'' and R_i'' are sampled independently and uniformly, and $\mathcal{A}$ cannot distinguish such replacement except with $\varepsilon^{\mathsf{m\text{-}enc\text{-}los}}_{\mathsf{PKE}_0}$. We move the description of $\mathbf{G}_7$-$\mathbf{G}_9$ to Figure 27.

Game $\mathbf{G}_8$: We modify the generation of ciphertext e_i and simulation of $\textsc{Open}$. In this game, e_i is an encryption of a randomly chosen r_i'' with randomness R_i'' (see Line 14) which are independent of $r_i, r_i', R_i, \mathsf{d}_i$. When $\mathcal{A}$ opens $\mathbf{c}[i] = (e_i, \mathsf{d}_i)$, the game simulator reprograms H and G so that $\mathbf{c}[i]$ can be "explained" by message m_i and randomness r_i' (i.e., $\mathsf{Enc}(\mathsf{pk}, \mathsf{m}_i; r_i') = \mathbf{c}[i]$), and returns (m_i, r_i'). By the lossiness of PKE_0, the statistical distance between $\{\mathsf{PKE}_0(\mathsf{pk}', r_i)\}_{i\in[\mu]}$ and $\{\mathsf{PKE}_0(\mathsf{pk}', r_i'')\}_{i\in[\mu]}$ is $\varepsilon^{\mathsf{m\text{-}enc\text{-}los}}_{\mathsf{PKE}_0}$. Hence, we have

$$\left|\Pr\left[\mathbf{G}_7^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathbf{G}_8^{\mathcal{A}} \Rightarrow 1\right]\right| \leq \varepsilon^{\mathsf{m\text{-}enc\text{-}los}}_{\mathsf{PKE}_0},\ \left|\Pr\left[\textsc{Query}_7\right] - \Pr\left[\textsc{Query}_8\right]\right| \leq \varepsilon^{\mathsf{m\text{-}enc\text{-}los}}_{\mathsf{PKE}_0}.$$

Now $\Pr\left[\textsc{Query}_8\right]$ can be bounded. Since r_i and r_i' are chosen uniformly and independent of $\mathbf{c}[i]$ (for $i \in [\mu]$), we have

$$\Pr\left[\textsc{Query}_8\right] \leq \frac{\mu(q_G + q_H)}{|\mathcal{M}'|},\ \Pr[\textsc{Query}_7] \leq \varepsilon^{\mathsf{m\text{-}enc\text{-}los}}_{\mathsf{PKE}_0} + \frac{\mu(q_G + q_H)}{|\mathcal{M}'|}.$$

Since now r_i' are independent of e_i before opening, and r_i is redundant in the simulation, we withdraw all the abort events defined in H and G, and no longer reprogram $H(r_i)$ and $G(r_i, \mathsf{d}_i)$.

Game $\mathbf{G}_9$: the aborts event defined in H and G are withdraw, and we no longer generate r_i and reprogram $H(r_i)$ and $G(r_i, \mathsf{d}_i)$ when $\mathbf{c}[i]$ is opened. Since in $\mathbf{G}_9$, for $i \in [\mu]$, r_i are independent of $\mathbf{c}[i]$, and r_i' are independent of $\mathbf{c}[i]$ before opening, the probability that $\mathcal{A}$ can detect this modification is $\frac{2\mu(q_G+q_H)}{|\mathcal{M}'|}$. Note that we have assumed that there is no collision among $r_i's$. So, we have

$$\left|\Pr\left[\mathbf{G}_8^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathbf{G}_9^{\mathcal{A}} \Rightarrow 1\right]\right| \leq \frac{2\mu(q_G + q_H)}{|\mathcal{M}'|} + \frac{\mu^2}{|\mathcal{M}'|}.$$

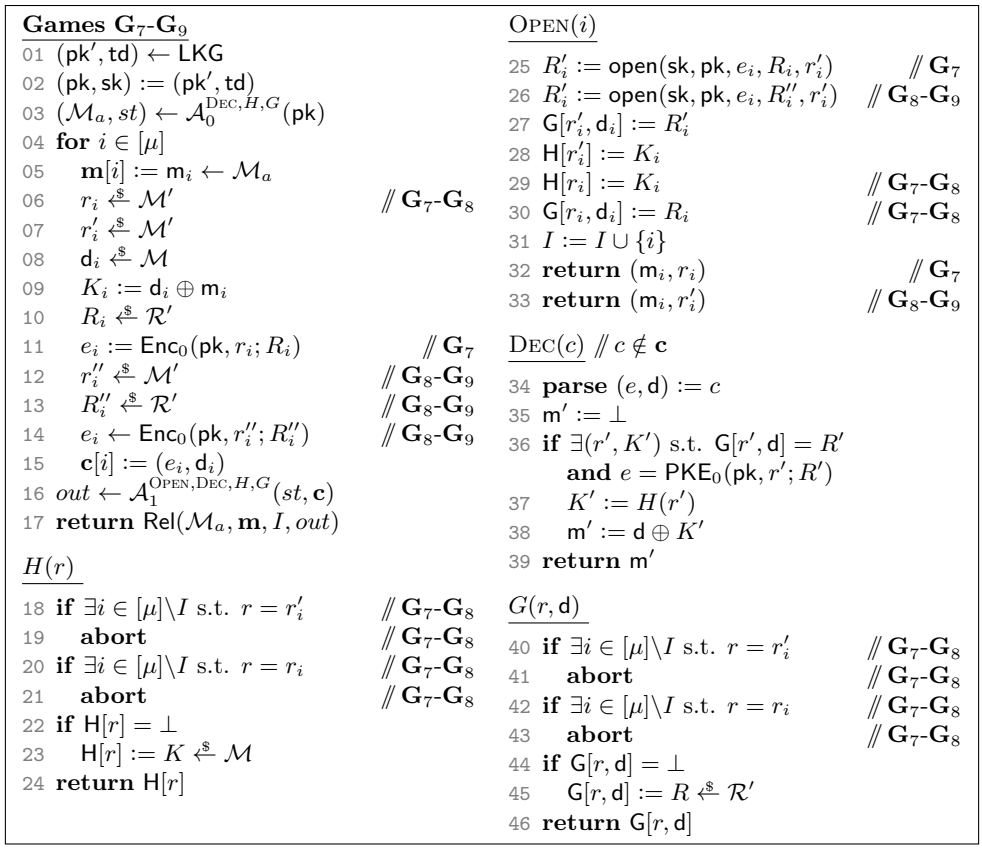

Figure 27: Games $\mathbf{G}_7$-$\mathbf{G}_9$ for proving Theorem 5.4.

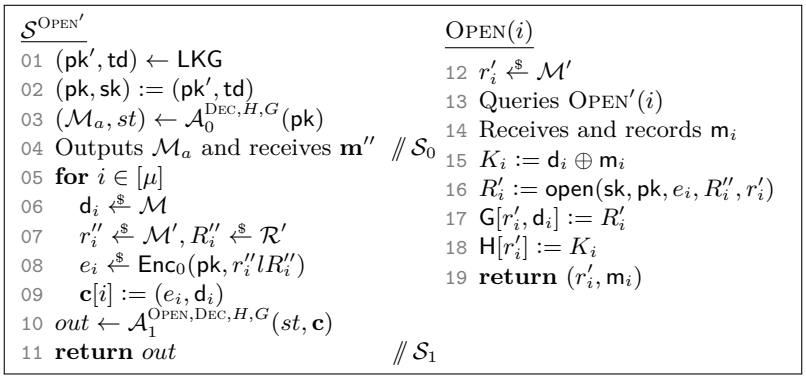

Figure 28: SIM-SO-CCA simulator $\mathcal{S}$ that simulates $\mathbf{G}_9$ to conclude the proof of Theorem 5.4. Here we ignore the details about simulation of H, G, and Dec which are the same as in Figure 27.

Now we can construct a simulator $\mathcal{S}$ that interacts with the IDEAL-SO-CCA game and simulate $\mathbf{G}_9$ for $\mathcal{A}$. The construction of $\mathcal{S}$ is shown in Figure 28. The main difference between $\mathbf{G}_9$ and $\mathcal{S}$ is that r'_i is sampled uniformly and K_i is computed when $\mathcal{A}$ queries Open(i), which is conceptual. We have assumed that all r'_i's and all K's are pair-wise distinct, and the outputs of ROs H and G are different. Hence, we have

$$\left| \Pr\left[\mathbf{G}_9^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathsf{IDEAL\text{-}SO\text{-}CCA}_{\mathsf{F0}}^{\mathcal{S}} \Rightarrow 1\right] \right| \leq \frac{n_H^2}{|\mathcal{M}|} + \frac{\mu^2}{|\mathcal{M}'|} + \frac{n_G^2}{|\mathcal{R}'|}.$$

Combining all the above difference, we conclude Theorem 5.4 as

$$\left|\Pr\left[\mathsf{REAL\text{-}SO\text{-}CCA}^{\mathcal{A}}_{\mathsf{FO}} \Rightarrow 1\right] - \Pr\left[\mathsf{IDEAL\text{-}SO\text{-}CCA}^{\mathcal{S}}_{\mathsf{FO}} \Rightarrow 1\right]\right|$$
$$\leq \mathsf{Adv}^{\mathsf{key\text{-}ind}}_{\mathsf{PKE}_0}(\mathcal{B}) + 2\varepsilon^{\mathsf{m\text{-}enc\text{-}los}}_{\mathsf{PKE}_0} + \frac{\mu n_{\mathrm{DEC}}}{2^\gamma} + \frac{2n_H^2}{|\mathcal{M}|} + \frac{2n_G^2}{|\mathcal{R}'|} + \frac{4\mu^2 + 5\mu(q_G + q_H)}{|\mathcal{M}'|}.$$

□

IND-SO-CCA Security. We show IND-SO-CCA of the construction. For that, we do not have to assume efficient openability.

Theorem 5.5 FO *in Figure 25 is IND-SO-CCA secure (Definition 2.16) if G and H are modeled as random oracles, and PKE_0 is a lossy encryption and γ-spreadness. Concretely, for any IND-SO-CCA adversary $\mathcal{A}$, there exists $\mathcal{B}$ such that:*

$$\mathsf{Adv}^{\mathsf{IND\text{-}SO\text{-}CCA}}_{\mathsf{FO}}(\mathcal{A}, \mu) \leq 2\left(\mathsf{Adv}^{\mathsf{key\text{-}ind}}_{\mathsf{PKE}_0}(\mathcal{B}) + 2\varepsilon^{\mathsf{m\text{-}enc\text{-}los}}_{\mathsf{PKE}_0} + \frac{\mu n_{\mathrm{DEC}}}{2^\gamma} + \frac{n_H^2}{|\mathcal{M}|} + \frac{n_G^2}{|\mathcal{R}'|} + \frac{3\mu^2 + 5\mu(q_G + q_H)}{|\mathcal{M}'|}\right),$$

where q_H, q_G, and n_{DEC} are the numbers of $\mathcal{A}$'s queries to G, H, and DEC, respectively, μ is the number of challenge ciphertexts, and $n_G = \mu + n_{\mathrm{DEC}} + q_H$ and $n_H = \mu + n_{\mathrm{DEC}} + q_G$ are the number of queries (including the simulator) to G and H, respectively.

Proof. The proof idea of Theorem 5.5 is the same as the one of Theorem 5.4. In $\mathbf{G}_{10}$ of the proof of Theorem 5.4 (see Figure 27), $\mathbf{m}[i]$ is independent of $\mathbf{c}[i]$ if $\mathbf{c}[i]$ is unopened. Therefore, we can resample $\mathbf{m}[i]$ for $i \in [\mu]\backslash I$, and finally change the game from $\mathsf{IND\text{-}SO\text{-}CCA}^{\mathcal{A}}_{\mathsf{PKE}_0,0}$ to $\mathsf{IND\text{-}SO\text{-}CCA}^{\mathcal{A}}_{\mathsf{PKE}_0,1}$. Note that now the algorithm open does not need to be efficient, since we do not need to construct an efficient simulator in IND-SO-CCA.

The games of the proof is shown in Figure 29. Similar to the argument in Theorem 5.4, we assume that from $\mathbf{G}_0$ to $\mathbf{G}_9$, there is no collision among all r_i's, R_i's, all K's, and the outputs of ROs G and H. We have

$$\left|\Pr\left[\mathsf{IND\text{-}SO\text{-}CCA}^{\mathcal{A}}_{\mathsf{PKE}_0,0} \Rightarrow 1\right] - \Pr\left[\mathbf{G}_0^{\mathcal{A}} \Rightarrow 1\right]\right| \leq \frac{n_H^2}{|\mathcal{M}|} + \frac{\mu^2}{|\mathcal{M}'|} + \frac{n_G^2}{|\mathcal{R}'|}$$

The game transitions from $\mathbf{G}_0$ to $\mathbf{G}_8$ in Figure 29 are exactly the same as the transitions in the proof of Theorem 5.4. Therefore, we have

$$\left|\Pr\left[\mathbf{G}_0^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathbf{G}_8^{\mathcal{A}} \Rightarrow 1\right]\right| \leq \mathsf{Adv}^{\mathsf{key\text{-}ind}}_{\mathsf{PKE}_0}(\mathcal{B}_0) + 2\varepsilon^{\mathsf{m\text{-}enc\text{-}los}}_{\mathsf{PKE}_0} + \frac{\mu n_{\mathrm{DEC}}}{2^\gamma} + \frac{2\mu^2}{|\mathcal{M}'|} + \frac{5\mu(q_G + q_H)}{|\mathcal{M}'|}$$

Game $\mathbf{G}_9$: We resample $\mathbf{m}[i]$ for all $i \in [\mu]\backslash I$. Since in $\mathbf{G}_8$, $\mathbf{c}[i]$ is independent of $\mathbf{m}[i]$ if $i \in [\mu]\backslash I$, this modification does not change $\mathcal{A}$'s view. So we have

$$\Pr\left[\mathbf{G}_8^{\mathcal{A}} \Rightarrow 1\right] = \Pr\left[\mathbf{G}_9^{\mathcal{A}} \Rightarrow 1\right].$$

Now $\mathbf{G}_9$ is the same as $\mathsf{IND\text{-}SO\text{-}CCA}^{\mathcal{A}}_{\mathsf{PKE}_0,1}$ if we undo all modifications made in $\mathbf{G}_8$-$\mathbf{G}_0$. For simplicity, we ignore the details. We have

$$\left|\Pr\left[\mathbf{G}_{10}^{\mathcal{A}} \Rightarrow 1\right] - \Pr\left[\mathsf{IND\text{-}SO\text{-}CCA}^{\mathcal{A}}_{\mathsf{PKE}_0,1} \Rightarrow 1\right]\right|$$
$$\leq \mathsf{Adv}^{\mathsf{key\text{-}ind}}_{\mathsf{PKE}_0}(\mathcal{B}_0) + 2\varepsilon^{\mathsf{m\text{-}enc\text{-}los}}_{\mathsf{PKE}_0} + \frac{\mu n_{\mathrm{DEC}}}{2^\gamma} + \frac{n_H^2}{|\mathcal{M}|} + \frac{n_G^2}{|\mathcal{R}'|} + \frac{3\mu^2 + 5\mu(q_G + q_H)}{|\mathcal{M}'|}$$

Combining all the above probability difference, we conclude Theorem 5.5 as

$$\left|\Pr\left[\mathsf{IND\text{-}SO\text{-}CCA}^{\mathcal{A}}_{\mathsf{PKE}_0,0} \Rightarrow 1\right] - \Pr\left[\mathsf{IND\text{-}SO\text{-}CCA}^{\mathcal{A}}_{\mathsf{PKE}_0,1} \Rightarrow 1\right]\right|$$
$$\leq 2\left(\mathsf{Adv}^{\mathsf{key\text{-}ind}}_{\mathsf{PKE}_0}(\mathcal{B}_0) + 2\varepsilon^{\mathsf{m\text{-}enc\text{-}los}}_{\mathsf{PKE}_0} + \frac{\mu n_{\mathrm{DEC}}}{2^\gamma} + \frac{n_H^2}{|\mathcal{M}|} + \frac{n_G^2}{|\mathcal{R}'|} + \frac{3\mu^2 + 5\mu(q_G + q_H)}{|\mathcal{M}'|}\right).$$

□

Games $\mathbf{G}_0$-$\mathbf{G}_9$

$(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{KG}_0$ // $\mathbf{G}_0$-$\mathbf{G}_1$
$(\mathsf{pk}', \mathsf{td}) \leftarrow \mathsf{LKG}$ // $\mathbf{G}_2$-$\mathbf{G}_9$
$(\mathsf{pk}, \mathsf{sk}) := (\mathsf{pk}', \mathsf{td})$ // $\mathbf{G}_2$-$\mathbf{G}_9$
$(\mathsf{Samp}, \mathsf{ReSamp}, \mathsf{st}_0) \leftarrow \mathcal{A}_0(\mathsf{pk})$
$\mathbf{m} \leftarrow \mathsf{Samp}$
for $i \in [\mu]$
  $r_i \xleftarrow{\$} \mathcal{M}'$
  $r'_i \xleftarrow{\$} \mathcal{M}'$ // $\mathbf{G}_3$-$\mathbf{G}_9$
  $K_i := H(r_i)$
  $K_i \xleftarrow{\$} \mathcal{M}$ // $\mathbf{G}_5$
  $\mathsf{d}_i := \mathsf{m}_i \oplus K_i$
  $\mathsf{d}_i \xleftarrow{\$} \mathcal{M}$ // $\mathbf{G}_6$-$\mathbf{G}_9$
  $K_i := \mathsf{d}_i \oplus \mathsf{m}_i$ // $\mathbf{G}_6$-$\mathbf{G}_9$
  $R_i := G(r_i, \mathsf{d}_i)$
  $R_i \xleftarrow{\$} \mathcal{R}'$ // $\mathbf{G}_5$-$\mathbf{G}_7$
  $e_i := \mathsf{Enc}_0(\mathsf{pk}, r_i; R_i)$ // $\mathbf{G}_0$-$\mathbf{G}_7$
  $r''_i \xleftarrow{\$} \mathcal{R}'$ // $\mathbf{G}_8$-$\mathbf{G}_9$
  $e_i \leftarrow \mathsf{Enc}_0(\mathsf{pk}, r''_i)$ // $\mathbf{G}_8$-$\mathbf{G}_9$
  $\mathbf{c}[i] := (e_i, \mathsf{d}_i)$
$\mathsf{st}_1 \leftarrow \mathcal{A}_1^{\textsc{Open},\textsc{Dec},G,H}(\mathbf{c}, \mathsf{st}_0)$
$\mathbf{m} := \mathsf{ReSamp}(I, \mathbf{m})$ // $\mathbf{G}_9$
$b' \leftarrow \mathcal{A}_2^{\textsc{Dec},G,H}(\mathsf{st}_1, \mathbf{m})$
return b'

$H(r)$

if $\exists i \in [\mu] \backslash I$ s.t. $r = r'_i$ // $\mathbf{G}_3$-$\mathbf{G}_9$
  abort // $\mathbf{G}_3$-$\mathbf{G}_9$
if $\exists i \in [\mu] \backslash I$ s.t. $r = r_i$ // $\mathbf{G}_4$-$\mathbf{G}_9$
  abort // $\mathbf{G}_4$-$\mathbf{G}_9$
if $\mathsf{H}[r] = \bot$
  $\mathsf{H}[r] := K \xleftarrow{\$} \mathcal{M}$
return $\mathsf{H}[r]$

$\textsc{Open}(i)$

$\mathsf{G}[r_i, \mathsf{d}_i] := R_i$ // $\mathbf{G}_5$-$\mathbf{G}_8$
$\mathsf{H}[r_i] := K_i$ // $\mathbf{G}_5$-$\mathbf{G}_8$
$R'_i := \mathsf{open}(\mathsf{sk}, \mathsf{pk}, e_i, r'_i)$ // $\mathbf{G}_7$-$\mathbf{G}_9$
$\mathsf{G}[r'_i, \mathsf{d}_i] := R'_i$ // $\mathbf{G}_7$-$\mathbf{G}_9$
$\mathsf{H}[r'_i] := K_i$ // $\mathbf{G}_7$-$\mathbf{G}_9$
$I := I \cup \{i\}$
return (r_i, m_i) // $\mathbf{G}_0$-$\mathbf{G}_7$
return (r'_i, m_i) // $\mathbf{G}_8$-$\mathbf{G}_9$

$\textsc{Dec}(c)$ // $c \notin \mathbf{c}$

parse $(e, \mathsf{d}) := c$
$\mathsf{m}' := \bot$
$r' := \mathsf{Dec}_0(\mathsf{sk}, e)$ // $\mathbf{G}_0$
$R' := G(r', \mathsf{d}), K' := H(r')$ // $\mathbf{G}_0$
if $e = \mathsf{Enc}_0(\mathsf{pk}, r'; R')$ // $\mathbf{G}_0$
  $\mathsf{m}' := d \oplus K'$ // $\mathbf{G}_0$
if $\exists (r', R')$ s.t. $\mathsf{G}[r', \mathsf{d}] = R'$
  and $e = \mathsf{PKE}_0(\mathsf{pk}, r'; R')$ // $\mathbf{G}_1$-$\mathbf{G}_9$
  $K' := H(r')$ // $\mathbf{G}_1$-$\mathbf{G}_9$
  $\mathsf{m}' := \mathsf{d} \oplus K'$ // $\mathbf{G}_1$-$\mathbf{G}_9$
return m'

$G(r, \mathsf{d})$

if $\exists i \in [\mu] \backslash I$ s.t. $r = r'_i$ // $\mathbf{G}_3$-$\mathbf{G}_9$
  abort // $\mathbf{G}_3$-$\mathbf{G}_9$
if $\exists i \in [\mu] \backslash I$ s.t. $r = r_i$ // $\mathbf{G}_4$-$\mathbf{G}_9$
  abort // $\mathbf{G}_4$-$\mathbf{G}_9$
if $\mathsf{G}[r, \mathsf{d}] = \bot$
  $\mathsf{G}[r, \mathsf{d}] := R \xleftarrow{\$} \mathcal{R}'$
return $\mathsf{G}[r, \mathsf{d}]$

Figure 29: Games $\mathbf{G}_0$-$\mathbf{G}_9$ for proving Theorem 5.4.

5.3 Two Instantiations from DDH

We instantiate FO using the DDH-based lossy encryption from Bellare et al. [BHY09] and Hofheinz et al. [HJR16]. The one from Bellare et al. does not have efficient openability and hence it only gives us

An Instantiation with Bellare et al.'s Lossy Encryption [BHY09]. We use Bellare et al.'s DDH-based lossy encryption to instantiate the generic construction FO. Let $\mathbb{G}$ be a group with prime order p and generator g, $H : \mathbb{G} \to \mathcal{M}$ and $G : \mathbb{G} \times \mathcal{M} \to \mathbb{Z}_p^2$ be hash functions. The resulting scheme $\mathtt{FO}_1$ is shown in Figure 30. Bellare et al.'s DDH-based lossy encryption does not have efficient opener [BHY09], and it is $\log(p)$-spread, thus by Theorem 5.5 , the resulting scheme $\mathtt{FO}_1$ in Figure 30 has tight IND-SO-CCA security.

Corollary 5.6 $\mathtt{FO}_1$ *in Figure 30 is IND-SO-CCA secure (Definition 2.16) if the* DDH *problem is hard on* $\mathbb{G}$ *and* G *and* H *are random oracles. Concretely, for any IND-SO-CCA adversary* $\mathcal{A}$*, there exists* $\mathcal{B}$ *such that:*

$$\mathsf{Adv}^{\mathsf{IND\text{-}SO\text{-}CCA}}_{\mathtt{FO}_1}(\mathcal{A}, \mu) \le 2(\mathsf{Adv}^{\mathsf{DDH}}_{\mathbb{G}}(\mathcal{B}) + \frac{2\mu}{p} + \frac{\mu n_{\textsc{Dec}}}{p}) + \frac{2n_H^2}{|\mathcal{M}|} + \frac{2n_G^2}{p^2} + \frac{6\mu^2 + 5\mu(q_G + q_H)}{p},$$

where q_H, q_G*, and* $n_{\textsc{Dec}}$ *are the numbers of* $\mathcal{A}$*'s queries to* G, H*, and* Dec*, respectively,* μ *is the number of*

$\underline{\mathsf{KG}_1^{\mathsf{fo}}}$	$\underline{\mathsf{Enc}_1^{\mathsf{fo}}(\mathsf{pk},\mathsf{m})}$	$\underline{\mathsf{Dec}(\mathsf{sk},((R_0,R_1),\mathsf{d}))}$
01 $(x,\omega) \xleftarrow{\$} \mathbb{Z}_p^2$	07 $s \leftarrow \mathbb{G}$	14 $\mathsf{m}' := \perp$
02 $g_0 := g, X := g_0^x$	08 $K := H(s)$	15 $s' := R_1/R_0^x$
03 $g_1 := g^\omega, h := g_1^x$	09 $\mathsf{d} := K \oplus \mathsf{m}$	16 $(r_0', r_1') := G(s', \mathsf{d})$
04 $\mathsf{pk} := (X, g_1, h)$	10 $(r_0, r_1) := G(s, \mathsf{d})$	17 $K' := H(s')$
05 $\mathsf{sk} := x$	11 $R_0 := g_0^{r_0} g_1^{r_1}$	18 $R_0' := g_0^{r_0'} g_1^{r_1'}$
06 **return** $(\mathsf{pk}, \mathsf{sk})$	12 $R_1 := X^{r_0} h^{r_1} \cdot s$	19 $R_1' := X^{r_0'} h^{r_1'} \cdot s'$
	13 **return** $((R_0, R_1), \mathsf{d})$	20 **if** $(R_0', R_1') = (R_0, R_1)$
		21 $\quad \mathsf{m}' := \mathsf{d} \oplus K'$
		22 **return** m'

Figure 30: Scheme $\mathtt{FO}_1$ from instantiating $\mathtt{FO}$ using the DDH-based lossy encryption in [BHY09].

challenge ciphertexts, and $n_G = \mu + n_{\mathrm{DEC}} + q_H$ *and* $n_H = \mu + n_{\mathrm{DEC}} + q_G$ *are the number of queries to* G *and* H*, respectively.*

An Instantiation with Hofheinz et al.'s Lossy Encryption [HJR16]. We use Hofheinz et al.'s DDH-based lossy encryption to instantiate $\mathtt{FO}$. Following the notation in [HJR16], we use the matrix Diffie-Hellman notation [EHK+13] to describe this scheme. Let $\mathbb{G}$ be a group with prime order p and generator g. Let $\mathbf{A} := (a_{i,j})_{(i,j)\in[l]\times[k]}$ be a matrix in $\mathbb{Z}_p^{l\times k}$, then the group representation of $\mathbf{A}$, denoted as $[\mathbf{A}]$, is defined as $(g^{a_{i,j}})_{(i,j)\in[l]\times[k]}$. Given $\mathbf{r}$ and $[\mathbf{A}]$, one can efficiently compute $[\mathbf{Ar}]$ (if their sizes match). We refer [EHK+13] for more details.

Let N be a positive integer. Let $H : \{0,1\}^N \to \mathcal{M}$ and $G : \{0,1\}^N \times \mathcal{M} \to \mathbb{Z}_p^{N+1}$ be two hash functions. Let $h : \mathbb{G} \to \{0,1\}$ be a universal hash function. The instantiated PKE scheme $\mathtt{FO}_2$ is shown in Figure 31. Hofheinz et al.'s DDH-based lossy encryption has efficient opener, and it is $\log(p)$-spread, thus by Theorem 5.4, $\mathtt{FO}_2$ has tight SIM-SO-CCA security.

$\underline{\mathsf{KG}_2^{\mathsf{fo}}}$	$\underline{\mathsf{Enc}_2^{\mathsf{fo}}(\mathsf{pk},\mathsf{m})}$	$\underline{\mathsf{Dec}_2^{\mathsf{fo}}(\mathsf{sk},([\mathbf{R}_0],c),\mathsf{d})}$
01 $\mathbf{A}_0 \xleftarrow{\$} \mathbb{Z}_p^{1\times(N+1)}$	07 $s \leftarrow \{0,1\}^N$	17 $\mathsf{m}' := \perp$
02 $\mathbf{T} \xleftarrow{\$} \mathbb{Z}_p^{N\times 1}$	08 $K := H(s)$	18 $[\mathbf{Z}'] := [\mathbf{TR}_0]$
03 $\mathbf{A}_1 := \mathbf{TA}_0 \in \mathbb{Z}_p^{N\times(N+1)}$	09 $\mathsf{d} := K \oplus \mathsf{m}$	19 $c_1 c_2 ... c_N =: c$
04 $\mathsf{pk} := ([\mathbf{A}_0], [\mathbf{A}_1])$	10 $\mathbf{r} := G(s, \mathsf{d}) \in \mathbb{Z}_p^{N+1}$	20 **for** $i \in [N]$
05 $\mathsf{sk} := \mathbf{T}$	11 $[\mathbf{R}_0] := [\mathbf{A}_0\mathbf{r}] \in \mathbb{G}$	21 $\quad s_i' := c_i \oplus h([\mathbf{Z}']_i)$
06 **return** $(\mathsf{pk}, \mathsf{sk})$	12 $[\mathbf{Z}] := [\mathbf{A}_1\mathbf{r}] \in \mathbb{G}^N$	22 $s' := s_1' s_2' ... s_N'$
	13 **for** $i \in [N]$	23 $K' := H(s'), \mathbf{r}' := G(s', \mathsf{d})$
	14 $\quad c_i := h([\mathbf{Z}]_i) \oplus s_i$	24 **if** $[\mathbf{R}_0] = [\mathbf{A}_0\mathbf{r}']$
	15 $c := c_0 c_1 ... c_N$	25 $\quad \mathsf{m}' := \mathsf{d} \oplus K'$
	16 **return** $(([\mathbf{R}_0], c), \mathsf{d})$	26 **return** m'

Figure 31: Scheme $\mathtt{FO}_2$ from instantiating $\mathtt{FO}$ using the DDH-based lossy encryption with efficient opener as in [HJR16].

Corollary 5.7 $\mathtt{FO}_2$ *in Figure 31 is SIM-SO-CCA secure (Definition 2.15) if the* DDH *problem is hard on* $\mathbb{G}$*. Concretely, for any SIM-SO-CCA adversary* $\mathcal{A}$ *and relation* Rel*, there exists a simulator* $\mathcal{S}$ *and* $\mathcal{B}$ *such that:*

$$\mathsf{Adv}^{\mathsf{SIM\text{-}SO\text{-}CCA}}_{\mathtt{FO}_2}(\mathcal{A}, \mathcal{S}, \mu, \mathsf{Rel}) \le N \cdot \mathsf{Adv}^{\mathsf{DDH}}_{\mathbb{G}}(\mathcal{B}) + \frac{2\mu}{p} + \frac{\mu n_{\mathrm{DEC}}}{p} + \frac{2n_H^2}{|\mathcal{M}|} + \frac{2n_G^2}{p^{N+1}} + \frac{4\mu^2 + 5\mu(q_G + q_H)}{2^N},$$

where q_H, q_G*, and* n_{DEC} *are the numbers of* $\mathcal{A}$*'s queries to* G, H*, and* DEC*, respectively,* μ *is the number of challenge ciphertexts, and* $n_G = \mu + n_{\mathrm{DEC}} + q_H$ *and* $n_H = \mu + n_{\mathrm{DEC}} + q_G$ *are the number of queries (including the simulator) to* G *and* H*, respectively.*

5.4 An Instantiation from LWE

We give an instantiation based on the LWE assumption. For that, the reader may recall the lattice background introduced in Section 4.2. The well-known Regev encryption scheme [Reg05] and its extension to multiple message bits from [PVW08] is a lossy encryption. We already presented this scheme in Section 4.2. In Figure 32, we present the scheme that results from applying our second transformation to it.

$\underline{\mathsf{KG}^{\mathsf{fo}}_{\mathsf{LWE}}}$
01 $\mathsf{sk} := \mathbf{S} \xleftarrow{\$} \mathbb{Z}_q^{n\times\ell}$
02 $\mathbf{E} \leftarrow D^{m\times\ell}_{\mathbb{Z},s}$
03 $\mathbf{A} \xleftarrow{\$} \mathbb{Z}_q^{n\times m}$
04 $\mathbf{Y} := \mathbf{S}^\top\mathbf{A} + \mathbf{E}^\top \in \mathbb{Z}_q^{\ell\times m}$
05 $\mathsf{pk} := (\mathbf{A}, \mathbf{Y})$
06 **return** $(\mathsf{pk}, \mathsf{sk})$

$\underline{\mathsf{Enc}^{\mathsf{fo}}_{\mathsf{LWE}}(\mathsf{pk}, \mathsf{m})}$
07 $r \leftarrow \{0,1\}^\ell$
08 $K := H(r)$
09 $\mathsf{d} := K \oplus \mathsf{m}$
10 $R := G(r, \mathsf{d})$
11 $\mathbf{x} := \mathsf{SampG}(s', m; R)$
12 $\mathbf{c} := \mathbf{A}\mathbf{x}$
13 $\mathbf{v} := \mathbf{Y}\mathbf{x} + \mathsf{Encode}(r)^\top$
14 **return** $((\mathbf{c}, \mathbf{v}), \mathsf{d})$

$\underline{\mathsf{Dec}^{\mathsf{fo}}_{\mathsf{LWE}}(\mathsf{sk} = \mathbf{S}, ((\mathbf{c}, \mathbf{v}), \mathsf{d}))}$
15 $\mathsf{m}' := \bot$
16 $r' := \mathsf{Decode}(\mathbf{v}^\top - \mathbf{c}^\top\mathbf{S})$
17 $R' := G(r', \mathsf{d}), K' := H(r')$
18 $\mathbf{x} := \mathsf{SampG}(s', m; R')$
19 **if** $\mathbf{c} = \mathbf{A}\mathbf{x} \wedge \mathbf{v} = \mathbf{Y}\mathbf{x} + \mathsf{Encode}(r')^\top$
20 $\quad \mathsf{m}' := \mathsf{d} \oplus K'$
21 **return** m'

Figure 32: Scheme $\mathsf{PKE}^{\mathsf{fo}}_{\mathsf{LWE}}$ instantiating FO using the multi-bit extension of the Regev encryption scheme [Reg05, PVW08]. Here, $\mathsf{SampG}(s', m; R)$ is an algorithm that samples $\mathbf{x} \leftarrow D^m_{\mathbb{Z},s'}$ using random coins R.

In Section 4.2, we already showed correctness, and the proof of Lemma 4.3 implicitly shows key indistinguishability and lossiness. It remains to sketch spreadness and openability. We start with spreadness. Fix any public key $\mathsf{pk} = (\mathbf{A}, \mathbf{Y}) \in \mathbb{Z}_q^{n\times m} \times \mathbb{Z}_q^{\ell\times m}$, any ciphertext $(\mathbf{c}_0, \mathbf{v}_0) \in \mathbb{Z}_q^n \times \mathbb{Z}_q^m$, and any message $\mathsf{m} \in \{0,1\}^\ell$. Then, taking the probability over the random coins of the encryption algorithm Enc we have

$$\Pr\left[\mathsf{Enc}(\mathsf{pk}, \mathsf{m}) = (\mathbf{c}_0, \mathbf{v}_0)\right] = \Pr_{\mathbf{x}\leftarrow D^m_{\mathbb{Z},s'}}\left[(\mathbf{A}\mathbf{x}, \mathbf{Y}\mathbf{x} + \mathsf{Encode}(\mathsf{m})^\top) = (\mathbf{c}_0, \mathbf{v}_0)\right] \leq \Pr_{\mathbf{x}\leftarrow D^m_{\mathbb{Z},s'}}\left[\mathbf{A}\mathbf{x} = \mathbf{c}_0\right].$$

For all but a negligible fraction of matrices $\mathbf{A}$, the distribution of $\mathbf{A}\mathbf{x}$ is close to uniform (see Lemma 2.10), and therefore above probability is (up to a negligible difference) $1/q^n$. This shows γ-spreadness for some $\gamma \geq \omega(\log\lambda)$, which is sufficiently large. Next, we sketch the openability of the scheme. For openability, we are allowed to set up the public key $\mathsf{pk} = (\mathbf{A}, \mathbf{Y})$ with a trapdoor. The public key should be statistically close to uniform, if we recall the lossiness argument implicitly contained in the proof of Lemma 4.3. Using our trapdoor, it should be possible to open a ciphertext $c = (\mathbf{c}, \mathbf{v})$ to any message m by giving a randomness with which m encrypts to c. In our setting, we can set up the matrix $[\mathbf{A}^\top, \mathbf{Y}^\top]^\top$ with a lattice trapdoor [GPV08, MP12], which allows us to efficiently sample Gaussian preimages with respect to $[\mathbf{A}^\top, \mathbf{Y}^\top]^\top$. The randomness $\mathbf{x}$ that we have to provide to open a ciphertext should be Gaussian and satisfy

$$\begin{bmatrix}\mathbf{c}\\ \mathbf{v}\end{bmatrix} - \begin{bmatrix}\mathbf{0}\\ \mathsf{Encode}(\mathsf{m})^\top\end{bmatrix} = \begin{bmatrix}\mathbf{A}\\ \mathbf{Y}\end{bmatrix}\mathbf{x}.$$

A lattice trapdoor for $[\mathbf{A}^\top, \mathbf{Y}^\top]^\top$ lets us sample such an $\mathbf{x}$ efficiently, if we make appropriate minor adjustments to the parameters given in Section 4.2.

One may think we are finished here. However, if we look more closely, we need to sample the random coins for the encryption algorithm. This is not $\mathbf{x}$, but rather the uniformly random coins R going into the algorithm that samples $\mathbf{x} \leftarrow D^m_{\mathbb{Z},s'}$[2]. As in Line 11, we call this algorithm $\mathsf{SampG}(s', m; R)$. Let us sketch how we can efficiently sample correctly distributed coins R, if SampG is implemented appropriately. Concretely, we consider the implementation of SampG using rejection sampling as suggested in [GPV08]. The implementation samples each coordinate x of $\mathbf{x}$ independently as follows:

[2]This is because in the proof of SIM-SO-CCA security, we would open the ciphertext and then program the random oracle G (see Line 10 in Figure 32) accordingly. If we can only sample a suitable $\mathbf{x}$, we would have to rely on a non-standard random oracle that outputs Gaussian vectors instead of uniform strings.

1. Set $i = 0$.
2. Repeat the following until the first x_i is accepted:
   (a) Set $i = i + 1$.
   (b) Sample uniformly at random an $x_i \xleftarrow{\$} \mathbb{Z} \cup [-d, d]$. Here, d is a bound that depends on the Gaussian parameter s'. It satisfies that with overwhelming probability, a Gaussian with parameter is in $\mathbb{Z} \cup [-d, d]$.
   (c) Let k/l be a rational number statistically close to $\rho(x_i) \in [0, 1]$, where ρ is a function that only depends on s'.
   (d) Accept x_i with probabilility k/l. In other words, sample an integer $r_i \xleftarrow{\$} [l]$ and accept x_i if and only if $r_i \leq k$.
3. Return x_i of $\perp$ if no x_i was accepted within a maximal number of iterations.

We refer to this algorithm as the actual sampler. Now, given a Gaussian x, we can sample uniformly random coins $R = ((x_i)_i, (r_i)_i)$, conditioned on x being sampled from SampG, as follows:

1. If $x \notin \mathbb{Z} \cup [-d, d]$, return $\perp$. This occurs with negligible probability.
2. Run the sampling algorithm to determine a number L of iterations.
3. For $i = 1$ to $L - 1$, repeat the following:
   (a) Sample uniformly at random an $x_i \xleftarrow{\$} \mathbb{Z} \cup [-d, d]$.
   (b) Let k/l be a rational number negligibly close to $\rho(x_i)$.
   (c) Sample an integer $r_i \xleftarrow{\$} \{k + 1, \ldots, l\}$.
4. Let k/l be a rational number negligibly close to $\rho(x)$.
5. Sample an integer $r_L \xleftarrow{\$} [k]$ and set $x_L := x$.
6. Return $R := \left((x_i)_{i=1}^{L}, (r_i)_{i=1}^{L}\right)$.

We refer to this algorithm as the inverse sampler. Now, we briefly explain why for a Gaussian x the distribution of (x, R) output by this inverse sampler is the same as the distribution of (x, R) in the actual sampler conditioned on the event that the actual sampler outputs x. For that, first notice that the number of iterations L in both algorithms is distributed exactly the same. Conditioning on a fixed number of iterations L and on the output being x, consider the distributions of the r_i and x_i. First, consider $i < L$. Here, x_i is sampled from the same distribution in both samplers. Since $i < L$, we know that x_i is not accepted in the actual sampler, meaning that r_i is distributed uniformly over $\{k + 1, \ldots, l\}$, as it is in the inverse sampler. Finally, consider $i = L$. Here, as we condition on the output being x, we know that in the actual sampler we have $x_L = x$. The same holds in the inverse sampler. Further, as x_L is accepted, the distribution of r_L in the actual sampler is uniform over $[k]$, and so it is in the inverse sampler.

Corollary 5.8 $\mathsf{PKE}^{\mathsf{fo}}_{\mathsf{LWE}}$ *in Figure 32 is IND-SO-CCA secure (Definition 2.16) if the* $\mathsf{LWE}_{n,m,q,D_{\mathbb{Z},s}}$ *assumption holds and G and H are random oracles. Concretely, for any SIM-SO-CCA adversary $\mathcal{A}$ and relation* Rel*, there exists a simulator $\mathcal{S}$ and $\mathcal{B}$ such that:*

$$\mathsf{Adv}^{\mathsf{IND\text{-}SO\text{-}CCA}}_{\mathsf{PKE}^{\mathsf{fo}}_{\mathsf{LWE}}}(\mathcal{A}, \mathcal{S}, \mu, \mathsf{Rel}) \leq \ell \cdot \mathsf{Adv}^{\mathsf{LWE}_{n,m,q,D_{\mathbb{Z},s}}}(\mathcal{B}) + \mathsf{negl}(\lambda).$$

Acknowledgments

This work is supported by the Research Council of Norway under Project No. 324235. We thank the anonymous reviewers from Asiacrypt 2022 for referring us to the work of Bellare et al. [BDWY12] and encouraging us to discuss its impacts on previous work in the random oracle model and ours. Moreover, we thank one of our Asiacrypt reviewers for pointing out a mistake in Game 5 of our previous proof for [PZ22, Theorem 1]. The work of generalizing our three direct constructions in [PZ22] was started when Benedikt Wagner was visiting NTNU, and it was also suggested by one of our Asiacrypt reviewers. We are very thankful for this valuable suggestion as well.

References

[ABR01] Michel Abdalla, Mihir Bellare, and Phillip Rogaway. The oracle Diffie-Hellman assumptions and an analysis of DHIES. In David Naccache, editor, *CT-RSA 2001*, volume 2020 of *LNCS*, pages 143–158. Springer, Heidelberg, April 2001. (Cited on page 4, 5, 8, 22.)

[BDWY12] Mihir Bellare, Rafael Dowsley, Brent Waters, and Scott Yilek. Standard security does not imply security against selective-opening. In David Pointcheval and Thomas Johansson, editors, *EUROCRYPT 2012*, volume 7237 of *LNCS*, pages 645–662. Springer, Heidelberg, April 2012. (Cited on page 3, 6, 42.)

[BHK12] Florian Böhl, Dennis Hofheinz, and Daniel Kraschewski. On definitions of selective opening security. In Marc Fischlin, Johannes Buchmann, and Mark Manulis, editors, *PKC 2012*, volume 7293 of *LNCS*, pages 522–539. Springer, Heidelberg, May 2012. (Cited on page 3.)

[BHY09] Mihir Bellare, Dennis Hofheinz, and Scott Yilek. Possibility and impossibility results for encryption and commitment secure under selective opening. In Antoine Joux, editor, *EUROCRYPT 2009*, volume 5479 of *LNCS*, pages 1–35. Springer, Heidelberg, April 2009. (Cited on page 3, 4, 5, 6, 32, 38, 39.)

[BR93] Mihir Bellare and Phillip Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. In Dorothy E. Denning, Raymond Pyle, Ravi Ganesan, Ravi S. Sandhu, and Victoria Ashby, editors, *ACM CCS 93*, pages 62–73. ACM Press, November 1993. (Cited on page 4.)

[BR95] Mihir Bellare and Phillip Rogaway. Optimal asymmetric encryption. In Alfredo De Santis, editor, *EUROCRYPT'94*, volume 950 of *LNCS*, pages 92–111. Springer, Heidelberg, May 1995. (Cited on page 4.)

[BR06] Mihir Bellare and Phillip Rogaway. The security of triple encryption and a framework for code-based game-playing proofs. In Serge Vaudenay, editor, *EUROCRYPT 2006*, volume 4004 of *LNCS*, pages 409–426. Springer, Heidelberg, May / June 2006. (Cited on page 8.)

[CF01] Ran Canetti and Marc Fischlin. Universally composable commitments. In Joe Kilian, editor, *CRYPTO 2001*, volume 2139 of *LNCS*, pages 19–40. Springer, Heidelberg, August 2001. (Cited on page 22.)

[CHJ+02] Jean-Sébastien Coron, Helena Handschuh, Marc Joye, Pascal Paillier, David Pointcheval, and Christophe Tymen. GEM: A generic chosen-ciphertext secure encryption method. In Bart Preneel, editor, *CT-RSA 2002*, volume 2271 of *LNCS*, pages 263–276. Springer, Heidelberg, February 2002. (Cited on page 4.)

[CKS08] David Cash, Eike Kiltz, and Victor Shoup. The twin Diffie-Hellman problem and applications. In Nigel P. Smart, editor, *EUROCRYPT 2008*, volume 4965 of *LNCS*, pages 127–145. Springer, Heidelberg, April 2008. (Cited on page 5, 9, 22, 24, 25.)

[CS02] Ronald Cramer and Victor Shoup. Universal hash proofs and a paradigm for adaptive chosen ciphertext secure public-key encryption. In Lars R. Knudsen, editor, *EUROCRYPT 2002*, volume 2332 of *LNCS*, pages 45–64. Springer, Heidelberg, April / May 2002. (Cited on page 25.)

[CW13] Jie Chen and Hoeteck Wee. Fully, (almost) tightly secure IBE and dual system groups. In Ran Canetti and Juan A. Garay, editors, *CRYPTO 2013, Part II*, volume 8043 of *LNCS*, pages 435–460. Springer, Heidelberg, August 2013. (Cited on page 3.)

[DGJL21] Denis Diemert, Kai Gellert, Tibor Jager, and Lin Lyu. More efficient digital signatures with tight multi-user security. In Juan Garay, editor, *PKC 2021, Part II*, volume 12711 of *LNCS*, pages 1–31. Springer, Heidelberg, May 2021. (Cited on page 3.)

[EHK+13] Alex Escala, Gottfried Herold, Eike Kiltz, Carla Ràfols, and Jorge Villar. An algebraic framework for Diffie-Hellman assumptions. In Ran Canetti and Juan A. Garay, editors, *CRYPTO 2013, Part II*, volume 8043 of *LNCS*, pages 129–147. Springer, Heidelberg, August 2013. (Cited on page 8, 39.)

[FHKW10] Serge Fehr, Dennis Hofheinz, Eike Kiltz, and Hoeteck Wee. Encryption schemes secure against chosen-ciphertext selective opening attacks. In Henri Gilbert, editor, *EUROCRYPT 2010*, volume 6110 of *LNCS*, pages 381–402. Springer, Heidelberg, May / June 2010. (Cited on page 3, 5.)

[FO13] Eiichiro Fujisaki and Tatsuaki Okamoto. Secure integration of asymmetric and symmetric encryption schemes. *Journal of Cryptology*, 26(1):80–101, January 2013. (Cited on page 4, 5, 32, 33.)

[GHK17] Romain Gay, Dennis Hofheinz, and Lisa Kohl. Kurosawa-desmedt meets tight security. In Jonathan Katz and Hovav Shacham, editors, *CRYPTO 2017, Part III*, volume 10403 of *LNCS*, pages 133–160. Springer, Heidelberg, August 2017. (Cited on page 3.)

[GJ18] Kristian Gjøsteen and Tibor Jager. Practical and tightly-secure digital signatures and authenticated key exchange. In Hovav Shacham and Alexandra Boldyreva, editors, *CRYPTO 2018, Part II*, volume 10992 of *LNCS*, pages 95–125. Springer, Heidelberg, August 2018. (Cited on page 9.)

[GPV07] Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. Cryptology ePrint Archive, Report 2007/432, 2007. https://eprint.iacr.org/2007/432. (Cited on page 9.)

[GPV08] Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan. Trapdoors for hard lattices and new cryptographic constructions. In Richard E. Ladner and Cynthia Dwork, editors, *40th ACM STOC*, pages 197–206. ACM Press, May 2008. (Cited on page 31, 40.)

[GSW13] Craig Gentry, Amit Sahai, and Brent Waters. Homomorphic encryption from learning with errors: Conceptually-simpler, asymptotically-faster, attribute-based. In Ran Canetti and Juan A. Garay, editors, *CRYPTO 2013, Part I*, volume 8042 of *LNCS*, pages 75–92. Springer, Heidelberg, August 2013. (Cited on page 4.)

[HHK17] Dennis Hofheinz, Kathrin Hövelmanns, and Eike Kiltz. A modular analysis of the Fujisaki-Okamoto transformation. In Yael Kalai and Leonid Reyzin, editors, *TCC 2017, Part I*, volume 10677 of *LNCS*, pages 341–371. Springer, Heidelberg, November 2017. (Cited on page 10, 27, 33.)

[HJKS15] Felix Heuer, Tibor Jager, Eike Kiltz, and Sven Schäge. On the selective opening security of practical public-key encryption schemes. In Jonathan Katz, editor, *PKC 2015*, volume 9020 of *LNCS*, pages 27–51. Springer, Heidelberg, March / April 2015. (Cited on page 3, 4, 6, 7.)

[HJKS16] Felix Heuer, Tibor Jager, Eike Kiltz, and Sven Schäge. On the selective opening security of practical public-key encryption schemes. Cryptology ePrint Archive, Report 2016/342, 2016. https://eprint.iacr.org/2016/342. (Cited on page 4, 6.)

[HJR16] Dennis Hofheinz, Tibor Jager, and Andy Rupp. Public-key encryption with simulation-based selective-opening security and compact ciphertexts. In Martin Hirt and Adam D. Smith, editors, *TCC 2016-B, Part II*, volume 9986 of *LNCS*, pages 146–168. Springer, Heidelberg, October / November 2016. (Cited on page 3, 4, 5, 6, 32, 38, 39.)

[HLLG19] Shuai Han, Shengli Liu, Lin Lyu, and Dawu Gu. Tight leakage-resilient CCA-security from quasi-adaptive hash proof system. In Alexandra Boldyreva and Daniele Micciancio, editors, *CRYPTO 2019, Part II*, volume 11693 of *LNCS*, pages 417–447. Springer, Heidelberg, August 2019. (Cited on page 3.)

[HLOV11] Brett Hemenway, Benoît Libert, Rafail Ostrovsky, and Damien Vergnaud. Lossy encryption: Constructions from general assumptions and efficient selective opening chosen ciphertext security. In Dong Hoon Lee and Xiaoyun Wang, editors, *ASIACRYPT 2011*, volume 7073 of *LNCS*, pages 70–88. Springer, Heidelberg, December 2011. (Cited on page 32.)

[Hof12] Dennis Hofheinz. All-but-many lossy trapdoor functions. In David Pointcheval and Thomas Johansson, editors, *EUROCRYPT 2012*, volume 7237 of *LNCS*, pages 209–227. Springer, Heidelberg, April 2012. (Cited on page 5.)

[HP16] Felix Heuer and Bertram Poettering. Selective opening security from simulatable data encapsulation. In Jung Hee Cheon and Tsuyoshi Takagi, editors, *ASIACRYPT 2016, Part II*, volume 10032 of *LNCS*, pages 248–277. Springer, Heidelberg, December 2016. (Cited on page 3.)

[JK18] Tibor Jager and Rafael Kurek. Short digital signatures and ID-KEMs via truncation collision resistance. In Thomas Peyrin and Steven Galbraith, editors, *ASIACRYPT 2018, Part II*, volume 11273 of *LNCS*, pages 221–250. Springer, Heidelberg, December 2018. (Cited on page 15.)

[JKRS21] Tibor Jager, Eike Kiltz, Doreen Riepel, and Sven Schäge. Tightly-secure authenticated key exchange, revisited. In Anne Canteaut and François-Xavier Standaert, editors, *EUROCRYPT 2021, Part I*, volume 12696 of *LNCS*, pages 117–146. Springer, Heidelberg, October 2021. (Cited on page 25, 26.)

[LLHG18] Lin Lyu, Shengli Liu, Shuai Han, and Dawu Gu. Tightly SIM-SO-CCA secure public key encryption from standard assumptions. In Michel Abdalla and Ricardo Dahab, editors, *PKC 2018, Part I*, volume 10769 of *LNCS*, pages 62–92. Springer, Heidelberg, March 2018. (Cited on page 3, 4, 6, 13.)

[LP15] Shengli Liu and Kenneth G. Paterson. Simulation-based selective opening CCA security for PKE from key encapsulation mechanisms. In Jonathan Katz, editor, *PKC 2015*, volume 9020 of *LNCS*, pages 3–26. Springer, Heidelberg, March / April 2015. (Cited on page 3.)

[LSSS17] Benoît Libert, Amin Sakzad, Damien Stehlé, and Ron Steinfeld. All-but-many lossy trapdoor functions and selective opening chosen-ciphertext security from LWE. In Jonathan Katz and Hovav Shacham, editors, *CRYPTO 2017, Part III*, volume 10403 of *LNCS*, pages 332–364. Springer, Heidelberg, August 2017. (Cited on page 3.)

[MP12] Daniele Micciancio and Chris Peikert. Trapdoors for lattices: Simpler, tighter, faster, smaller. In David Pointcheval and Thomas Johansson, editors, *EUROCRYPT 2012*, volume 7237 of *LNCS*, pages 700–718. Springer, Heidelberg, April 2012. (Cited on page 40.)

[MR04] Daniele Micciancio and Oded Regev. Worst-case to average-case reductions based on Gaussian measures. In *45th FOCS*, pages 372–381. IEEE Computer Society Press, October 2004. (Cited on page 9.)

[OP01] Tatsuaki Okamoto and David Pointcheval. REACT: Rapid Enhanced-security Asymmetric Cryptosystem Transform. In David Naccache, editor, *CT-RSA 2001*, volume 2020 of *LNCS*, pages 159–175. Springer, Heidelberg, April 2001. (Cited on page 4.)

[PVW08] Chris Peikert, Vinod Vaikuntanathan, and Brent Waters. A framework for efficient and composable oblivious transfer. In David Wagner, editor, *CRYPTO 2008*, volume 5157 of *LNCS*, pages 554–571. Springer, Heidelberg, August 2008. (Cited on page 30, 31, 40.)

[PZ22] Jiaxin Pan and Runzhi Zeng. Compact and tightly selective-opening secure public-key encryption schemes. In Shweta Agrawal and Dongdai Lin, editors, *ASIACRYPT 2022, Part III*, volume 13793 of *LNCS*, pages 363–393. Springer, Heidelberg, December 2022. (Cited on page 1, 5, 7, 12, 42.)

[Reg05] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. In Harold N. Gabow and Ronald Fagin, editors, *37th ACM STOC*, pages 84–93. ACM Press, May 2005. (Cited on page 5, 9, 30, 31, 40.)