Tightly SIM-SO-CCA Secure Public Key Encryption from Standard Assumptions*

Lin Lyu[1,2], Shengli Liu[1,2,3](✉), Shuai Han[1,2,4], and Dawu Gu[1,5]

[1] Dept. of Computer Science and Engineering,
Shanghai Jiao Tong University, Shanghai 200240, China
{lvlin,slliu,dalen17,dwgu}@sjtu.edu.cn
[2] State Key Laboratory of Cryptology, P.O. Box 5159, Beijing 100878, China
[3] Westone Cryptologic Research Center, Beijing 100070, China
[4] Karlsruhe Institute of Technology, Karlsruhe, Germany
[5] Shanghai Institute for Advanced Communication and Data Science,
Shanghai, China

Abstract. Selective opening security (SO security) is desirable for public key encryption (PKE) in a multi-user setting. In a selective opening attack, an adversary receives a number of ciphertexts for possibly correlated messages, then it opens a subset of them and gets the corresponding messages together with the randomnesses used in the encryptions. SO security aims at providing security for the unopened ciphertexts. Among the existing simulation-based, selective opening, chosen ciphertext secure (SIM-SO-CCA secure) PKEs, only one (Libert *et al.* Crypto'17) enjoys *tight* security, which is reduced to the Non-Uniform LWE assumption. However, their public key and ciphertext are not compact.

In this work, we focus on constructing PKE with *tight* SIM-SO-CCA security based on standard assumptions. We characterize securities needed for key encapsulation mechanism (KEM) and show how to transform these securities into SIM-SO-CCA security of PKE through a tight security reduction, while the construction of PKE from KEM follows the general framework proposed by Liu and Paterson (PKC'15). We present two KEM constructions with tight securities based on the Matrix Decision Diffie-Hellman assumption. These KEMs in turn lead to two tightly SIM-SO-CCA secure PKE schemes. One of them enjoys not only tight security but also compact public key.

1 Introduction

Selective Opening Security. In the context of public key encryption (PKE), IND-CPA(CCA) security is widely believed to be the right security notion. However, multi-user settings enable more complicated attacks and traditional IND-CPA(CCA) security may not be strong enough. Consider a scenario of N senders and one receiver. The senders encrypt N (possibly correlated) messages $\mathbf{m}_1, \cdots, \mathbf{m}_N$ under the receiver's public key pk using fresh randomnesses $\mathbf{r}_1, \cdots, \mathbf{r}_N$ to get ciphertexts $\mathbf{c}_1, \cdots, \mathbf{c}_N$, respectively, i.e., each sender i compute $\mathbf{c}_i = \mathsf{Enc}(\mathsf{pk}, \mathbf{m}_i; \mathbf{r}_i)$. Upon receiving the ciphertexts $\mathbf{c}_1, \cdots, \mathbf{c}_N$, the adversary might be able to open a subset of them via implementing corruptions. Namely, by corrupting a subset of users, say $I \subset [N]$, the adversary obtains the messages $\{\mathbf{m}_i\}_{i\in I}$ together with the randomnesses $\{\mathbf{r}_i\}_{i\in I}$. Such an attack is called selective opening attack (SOA). It is desirable that the unopened ciphertexts $\{\mathbf{c}_i\}_{i\in[N]\setminus I}$ still protects the privacy of $\{\mathbf{m}_i\}_{i\in[N]\setminus I}$, which is exactly what the SO security concerns.

* This is the full version of a paper that appeared in PKC 2018.

The potential correlation between $\{\mathbf{m}_i\}_{i\in I}$ and $\{\mathbf{m}_i\}_{i\in[N]\setminus I}$ hinders the use of hybrid argument proof technique. Hence, traditional IND-CPA security may not imply SO security. To date, there exist two types of SO security formalization: indistinguishability-based SO security (IND-SO, [BHY09, BHK12]) and simulation-based SO security (SIM-SO, [BHY09, DNRS99]). According to whether the adversary has access to a decryption oracle, these securities are further classified into IND-SO-CPA, IND-SO-CCA, SIM-SO-CPA and SIM-SO-CCA.

Intuitively, IND-SO security requires that, given public key pk, ciphertexts $\{\mathbf{c}_i\}_{i\in[N]}$, the opened messages $\{\mathbf{m}_i\}_{i\in I}$ and randomnesses $\{\mathbf{r}_i\}_{i\in I}$ (together with a decryption oracle in the CCA case), the unopened messages $\{\mathbf{m}_i\}_{i\in[N]\setminus I}$ remain computationally indistinguishable from independently sampled messages conditioned on the already opened messages $\{\mathbf{m}_i\}_{i\in I}$. Accordingly, the IND-SO security usually requires the message distributions be *efficiently conditionally re-samplable* [BHY09, HLOV11, Hof12] (and such security is referred to as *weak* IND-SO security in [BHK12]), which limits its application scenarios.

On the other hand, SIM-SO security is conceptually similar to semantic security [GM84]. It requires that the output of the SO adversary can be simulated by a simulator which only takes the opened messages $\{\mathbf{m}_i\}_{i\in I}$ as its input after it assigns the corruption set I. Since there is no restriction on message distribution, SIM-SO security has an advantage over IND-SO security from an application point of view. SIM-SO security was also shown to be stronger than (weak) IND-SO security in [BHK12]. However, as shown in [HJR16], SIM-SO security turns out to be significantly harder to achieve.

Generally speaking, there are two approaches to achieve SIM-SO-CCA security. The first approach uses lossy trapdoor functions [PW08], All-But-N lossy trapdoor functions [HLOV11] or All-But-Many lossy trapdoor functions [Hof12] to construct lossy encryption schemes. If this lossy encryption has an efficient opener, then the resulting PKE scheme can be proven to be SIM-SO-CCA secure as shown in [BHY09]. A DCR-based scheme in [Hof12] and a LWE-based scheme in [LSSS17] are the only two schemes known to have such an opener. The second approach uses extended hash proof system and cross-authentication codes (XACs) [FHKW10]. As pointed out in [HLQ13, HLQC13], a stronger property of XAC is required to make this proof rigorous. Following this line of research, Liu and Paterson proposed a general framework for constructing SIM-SO-CCA PKE from a special kind of key encapsulation mechanism (KEM) in combination with a strengthened XAC [LP15].

Tight Security Reductions. Usually, the security of a cryptographic primitive is established on the hardness of some underlying mathematical problems through a *security reduction*. It shows that any successful probabilistic polynomial-time (PPT) adversary $\mathcal{A}$ breaking the cryptographic primitive with advantage $\epsilon_{\mathcal{A}}$ can be transformed into a successful PPT problem solver $\mathcal{B}$ for the underlying hard problem with advantage $\epsilon_{\mathcal{B}}$. The ideal case is $\epsilon_{\mathcal{A}} = \epsilon_{\mathcal{B}}$. However, most reductions suffer from a loss in the advantage, for example, $\epsilon_{\mathcal{A}} = L \cdot \epsilon_{\mathcal{B}}$ where L is called *security loss factor* of the reduction. Smaller L always indicates a better security level for a fixed security parameter. For a PKE scheme, L usually depends on λ (the security parameter) as well as Q_e (the number of challenge ciphertexts) and Q_d (the number of decryption queries). A security reduction for a PKE scheme is *tight* and the PKE scheme is called a *tightly secure* one [GHKW16, Hof17] if L depends only on the security parameter λ[6] (and is independent of both Q_e and Q_d).

[6] According to [CW13, GHK17], such a security reduction is called an *almost* tight one and a security reduction is tight only if L is a constant.

Note that for concrete settings, λ is much smaller than Q_e and Q_d (for example, $\lambda = 80$ and Q_e, Q_d can be as large as 2^{20} or even 2^{30} in some settings). Most reductions are not tight and it appears to be a non-trivial problem to construct tightly IND-CCA secure PKE schemes.

Among the existing SIM-SO-CCA secure PKEs, only one of them has a tight security reduction [LSSS17]. Very recently, Libert *et al.* [LSSS17] provide an all-but-many lossy trapdoor function with an efficient opener, leading to a tightly SIM-SO-CCA secure PKE based on the Non-Uniform LWE assumption. Note that, their construction relies on a specific tightly secure PRF which is computable in NC^1. So far, no construction of such a PRF based on standard LWE assumption is known, which is why their PKE has to rely on a non-standard assumption. Meanwhile, there is no PKE scheme enjoying both tight SIM-SO-CCA security and compact public key & ciphertext up to now.

1.1 Our Contribution

We explore how to construct tightly SIM-SO-CCA secure PKE based on standard assumptions. Following the KEM+XAC framework proposed in [LP15],

- we characterize stronger security notions needed for KEM and present a tightness preserving security reduction, which shows the PKE is tightly SIM-SO-CCA secure as long as the underlying KEM is tightly secure;
- we present two KEM instantiations and prove that their security can be tightly reduced to the Matrix Decision Diffie-Hellman (MDDH) assumption, thus leading to two tightly SIM-SO-CCA secure PKE schemes. One of them enjoys not only tight security but also compact public key.

1.2 Technique Overview

Roughly speaking, to prove the SIM-SO-CCA security of a PKE (see for Definition 1), for any PPT adversary, we need to construct a simulator and show that the adversary's outputs are indistinguishable with those of the simulator. Naturally, such a simulator can be realized simply by simulating the entire real SO-CCA environment, invoking the adversary and returning the adversary's outputs. However, due to lack of essential information like messages and randomnesses, the simulator is not able to provide a perfect environment directly. Therefore, both the PKE scheme and the simulator has to be carefully designed, so that the simulator is able to provide the adversary a *computational indistinguishable* environment. To this end, we have to solve two problems.

- The first problem is how the simulator prepares ciphertexts for the adversary without knowing the messages.
- The second problem is how the simulator prepares randomnesses for the adversary according to the opened messages $\{\mathbf{m}_i\}_{i\in I}$ that it receives later.

To solve the first problem, the simulator has to provide ciphertexts that are computational indistinguishable with real ciphertexts in the setting of selective opening (together with chosen-ciphertext attacks). As to the second problem, note that the adversary can always check the consistence between $\{\mathbf{m}_i\}_{i\in I}, \{\mathbf{c}_i\}_{i\in I}$ and the randomnesses by re-encryption. Therefore, the simulator should not only provide indistinguishable ciphertexts but also be able to explain these ciphertexts as encryptions of any designated messages.

Liu and Paterson [LP15] solved these two problems and proposed a general framework for constructing SIM-SO-CCA secure PKE with the help of KEM in combination with XAC. Their PKE construction encrypts message in a bitwise manner. Suppose the message $\mathbf{m}$ has bit length ℓ. If the i-th bit of $\mathbf{m}$ is 1 ($\mathbf{m}_i = 1$), a pair of encapsulation ψ_i and key γ_i is generated from KEM, i.e., $(\psi_i, \gamma_i) \leftarrow_\$ \mathsf{KEnc}(\mathsf{pk}_{\mathsf{kem}})$. If $\mathbf{m}_i = 0$, a random pair is generated, i.e., $(\psi_i, \gamma_i) \leftarrow_\$ \Psi \times \Gamma$. Then a tag T is generated to bind up $(\gamma_1, \cdots, \gamma_\ell)$ and $(\psi_1, \cdots, \psi_\ell)$ via XAC. And the final ciphertext is $C = (\psi_1, \cdots, \psi_\ell, T)$.

They construct a simulator in the following way.

• Without knowledge of the message, the simulator uses an encryption of 1^ℓ as the ciphertext. Thus the encryption involves ℓ encapsulated pairs $(\psi_i, \gamma_i) \leftarrow_\$ \mathsf{KEnc}(\mathsf{pk}_{\mathsf{kem}})$. The simulator then saves all the randomnesses used in these encapsulations.

• When providing the randomnesses for the opened messages, the simulator checks the opened messages bit by bit. If a specific bit is 1, then the simulator outputs the original randomnesses and the simulation is perfect. Otherwise, the simulator views the encapsulated pair as a random pair. Then the simulator resamples randomnesses as if this pair is randomly chosen using these resampled randomnesses.

Thanks to the bit-wise encryption mode and the resampling property of spaces Ψ and Γ, an encapsulation pair (encrypting bit 1) can be easily explained as a random pair (encrypting bit 0). Therefore the second problem is solved.

To solve the first problem, one has to show that the encapsulated pairs are computationally indistinguishable with the random pairs. In [LP15], a special security named IND-tCCCA is formalized for KEM. This security guarantees that *one* encapsulated pair is computationally indistinguishable with *one* random pair even when a constrained decryption oracle is provided. With the help of IND-tCCCA security of KEM, the indistinguishability between the encryption of 1^ℓ and the encryption of real messages are proved with ℓ hybrid arguments, each hybrid replacing only one encapsulated pair with one random pair.

To pursue tight security reduction, the ℓ hybrid arguments have to be avoided. To this end, we enhance the IND-tCCCA security and consider the pseudorandomness for *multiple* pairs even when a constrained decryption oracle is provided. This new security for KEM is formalized as mPR-CCCA security in Definition 5. Armed with this enhanced security, it is possible to replace the ℓ encapsulated pairs once for all in the security reduction from the SIM-SO-CCA security of PKE to the mPR-CCCA security of KEM. However, this gives rise to another problem. The SIM-SO-CCA adversary $\mathcal{A}$ may submit a fresh ciphertext which shares the same encapsulation ψ with some challenge encapsulation. In the security reduction, the adversary $\mathcal{B}$, who invokes $\mathcal{A}$ to attack the mPR-CCCA security of KEM, cannot ask its own decapsulation oracle to decapsulate ψ since ψ is already embedded in some challenge ciphertext for $\mathcal{A}$. To solve this problem, we define another security notion for KEM, namely, the Random Encapsulation Rejection (RER) security of KEM (cf. Definition 6). Equipped with the RER security of KEM and a security of XAC, $\mathcal{B}$ could simply set 0 as the decryption bit for ψ.

Although the enhancement from IND-tCCCA to mPR-CCCA is conceptually simple, finding an mPR-CCCA secure KEM instantiation with tight reduction to standard assumptions is highly non-trivial. Inspired by the recent work on constructing tightly IND-CCA secure PKE [GHKW16, GHK17], we are able to give two tightly mPR-CCCA & RER secure KEM instantiations, one of which also enjoys compact public key.

1.3 Instantiation Overview

We provide two KEM instantiations.

The first KEM instantiation is inspired by a recent work in Eurocrypt'16. In the work [GHKW16], Gay *et al.* proposed the first tightly multi-challenge IND-CCA secure

PKE scheme based on the MDDH assumption. From their PKE construction, we extract a KEM and tightly prove its mPR-CCCA security & RER security based on the MDDH assumption.

The second KEM instantiation is contained in a very recent work by Gay *et al.* [GHK17] in Crypto'17. In [GHK17], a qualified proof system (QPS) is proposed to construct multi-challenge IND-CCCA secure KEM, which can be used to obtain a tightly multi-challenge IND-CCA secure PKE scheme with help of an authenticated encryption scheme. Note that our mPR-CCCA security is stronger than multi-challenge IND-CCCA security. To achieve mPR-CCCA security, we formalize a so-called Pseudorandom Simulated Proof property for QPS. We prove that if QPS has this property, the KEM from QPS is mPR-CCCA secure. Finally, we show that the QPS in [GHK17] possesses the pseudorandom simulated proof property.

Compared with the first instantiation, the public key of our second KEM instantiation has a constant number of group elements. The compactness of public key is in turn transferred to the PKE, resulting in the first tightly SIM-SO-CCA secure PKE based on standard assumptions together with a compact public key.

2 Preliminaries

We use λ to denote the security parameter in this work. Let ε be the empty string. For $n \in \mathbb{N}$, denote by $[n]$ the set $\{1, \cdots, n\}$. Denote by $s_1, \cdots, s_n \leftarrow_{\$} S$ the process of picking n elements uniformly from set S. For a PPT algorithm $\mathcal{A}$, we use $y \leftarrow \mathcal{A}(x; r)$ to denote the process of running $\mathcal{A}$ on input x with randomness r and assigning the deterministic result to y. Let $\mathcal{R}_{\mathcal{A}}$ be the randomness space of $\mathcal{A}$, we use $y \leftarrow_{\$} \mathcal{A}(x)$ to denote $y \leftarrow \mathcal{A}(x; r)$ where $r \leftarrow_{\$} \mathcal{R}_{\mathcal{A}}$. We use $\mathbf{T}(\mathcal{A})$ to denote the running time of $\mathcal{A}$, which is a polynomial in λ if $\mathcal{A}$ is PPT.

We use boldface letters to denote vectors or matrices. For a vector $\mathbf{m}$ of finite dimension, $|\mathbf{m}|$ denotes the dimension of the vector and $\mathbf{m}_i$ denotes the i-th component of $\mathbf{m}$. For a set $I = \{i_1, i_2, \cdots, i_{|I|}\} \subseteq [|\mathbf{m}|]$, define $\mathbf{m}_I := (\mathbf{m}_{i_1}, \mathbf{m}_{i_2}, \cdots, \mathbf{m}_{i_{|I|}})$. For all matrix $\mathbf{A} \in \mathbb{Z}_q^{\ell \times k}$ with $\ell > k$, $\overline{\mathbf{A}} \in \mathbb{Z}_q^{k \times k}$ denotes the upper square matrix of $\mathbf{A}$ and $\underline{\mathbf{A}} \in \mathbb{Z}_q^{(\ell-k)\times k}$ denotes the lower $\ell - k$ rows of $\mathbf{A}$. By $\mathsf{span}(\mathbf{A}) := \{\mathbf{Ar} \mid \mathbf{r} \in \mathbb{Z}_q^k\}$, we denote the span of $\mathbf{A}$. By $\mathsf{Ker}(\mathbf{A}^\top)$, we denote the orthogonal space of $\mathsf{span}(\mathbf{A})$. For $\ell = k$, we define the *trace* of $\mathbf{A}$ as the sum of all diagonal elements of $\mathbf{A}$, i.e., $\mathsf{trace}(\mathbf{A}) := \sum_{i=1}^{k} \mathbf{A}_{i,i}$.

A function $f(\lambda)$ is *negligible*, if for every $c > 0$ there exists a λ_c such that $f(\lambda) < 1/\lambda^c$ for all $\lambda > \lambda_c$.

We review the definitions of collision resistant hash function and universal hash function, together with leftover hash lemma in Appendix A.1.

We use game-based security proof. The games are illustrated using pseudo-codes in figures. By a box in a figure, we denote that the codes in the box appears in a specific game. For example, $\boxed{\boxed{G_4}\ \boxed{G_5}}$ means that G_4 contains the codes in $\boxed{\text{dash box}}$, G_5 contains the codes in $\boxed{\text{oval box}}$, and both of them contain codes in $\boxed{\text{square box}}$. Moreover, we assume that the unboxed codes are contained in all games. We use the notation $\Pr_i[\mathsf{E}]$ to denote the probability that event E occurs in game G_i, and use the notation $G \Rightarrow 1$ to denote the event that game G returns 1. All variables in games are initialized to $\perp$. We use "□" to denote the end of proof of lemmas and use "■" to denote the end of proof of theorems.

2.1 Prime-order Groups

Let GGen be a PPT algorithm that on input 1^λ returns $\mathcal{G} = (\mathbb{G}, q, P)$, a description of an additive cyclic group $\mathbb{G}$ with a generator P of order q which is a λ-bit prime. For $a \in \mathbb{Z}_q$, define $[a] := aP \in \mathbb{G}$ as the *implicit representation* of a in $\mathbb{G}$. More generally, for a matrix $\mathbf{A} = (a_{ij}) \in \mathbb{Z}_q^{n\times m}$, we define $[\mathbf{A}]$ as the implicit representation of $\mathbf{A}$ in $\mathbb{G}$, i.e., $[\mathbf{A}] := (a_{ij}P) \in \mathbb{G}^{n\times m}$. Note that from $[a] \in \mathbb{G}$ it is generally hard to compute the value a (discrete logarithm problem is hard in $\mathbb{G}$). Obviously, given $[a], [b] \in \mathbb{G}$ and a scalar $x \in \mathbb{Z}$, one can efficiently compute $[ax] \in \mathbb{G}$ and $[a + b] \in \mathbb{G}$. Similarly, for $\mathbf{A} \in \mathbb{Z}_q^{m\times n}, \mathbf{B} \in \mathbb{Z}_q^{n\times t}$, given $\mathbf{A}, \mathbf{B}$ or $[\mathbf{A}], \mathbf{B}$ or $\mathbf{A}, [\mathbf{B}]$, one can efficiently compute $[\mathbf{AB}] \in \mathbb{G}^{m\times t}$.

We review the Matrix Decision Diffie-Hellman Assumption relative to GGen in Appendix A.2.

2.2 Simulation-based, Selective-Opening CCA Security of PKE

We recall the definition of public key encryption in Appendix A.3. Let $\mathbf{m}$ and $\mathbf{r}$ be two vectors of dimension $n := n(\lambda)$. Define $\mathsf{Enc}(\mathsf{pk}, \mathbf{m}; \mathbf{r}) := (\mathsf{Enc}(\mathsf{pk}, \mathbf{m}_1; \mathbf{r}_1), \cdots, \mathsf{Enc}(\mathsf{pk}, \mathbf{m}_n; \mathbf{r}_n))$ where $\mathbf{r}_i$ is a fresh randomness used for the encryption of $\mathbf{m}_i$ for $i \in [n]$. Then we review the SIM-SO-CCA security definition in [FHKW10]. Let $\mathcal{M}$ denote an n-message sampler, which on input a string $\alpha \in \{0, 1\}^*$ outputs a message vector $\mathbf{m}$ of dimension n, i.e., $\mathbf{m} = (\mathbf{m}_1, \cdots, \mathbf{m}_n)$. Let R be any PPT relation.

$\mathsf{Exp}^{\text{so-cca-real}}_{\mathsf{PKE},\mathcal{A},n,\mathcal{M},R}(\lambda)$:	$\mathsf{Exp}^{\text{so-cca-ideal}}_{\mathcal{S},n,\mathcal{M},R}(\lambda)$:
$(\mathsf{pk}, \mathsf{sk}) \leftarrow_\$ \mathsf{Gen}(1^\lambda)$ $(\alpha, a_1) \leftarrow_\$ \mathcal{A}_1^{\mathsf{Dec}(\cdot)}(\mathsf{pk})$	$(\alpha, s_1) \leftarrow_\$ \mathcal{S}_1(1^\lambda)$
$\mathbf{m} \leftarrow_\$ \mathcal{M}(\alpha), \mathbf{r} \leftarrow_\$ (\mathcal{R}_{\mathsf{Enc}})^n$	$\mathbf{m} \leftarrow_\$ \mathcal{M}(\alpha)$
$\mathbf{C} \leftarrow \mathsf{Enc}(\mathsf{pk}, \mathbf{m}; \mathbf{r})$ $(I, a_2) \leftarrow_\$ \mathcal{A}_2^{\mathsf{Dec}_{\notin \mathbf{C}}(\cdot)}(a_1, \mathbf{C})$	$(I, s_2) \leftarrow_\$ \mathcal{S}_2(s_1, (1^{\lvert \mathbf{m}_i \rvert})_{i\in[n]})$
$\hat{\mathbf{r}}_I \leftarrow \mathbf{r}_I$ $out_{\mathcal{A}} \leftarrow_\$ \mathcal{A}_3^{\mathsf{Dec}_{\notin \mathbf{C}}(\cdot)}(a_2, \mathbf{m}_I, \hat{\mathbf{r}}_I)$	$out_{\mathcal{S}} \leftarrow_\$ \mathcal{S}_3(s_2, \mathbf{m}_I)$
Return $R(\mathbf{m}, I, out_{\mathcal{A}})$	Return $R(\mathbf{m}, I, out_{\mathcal{S}})$

Fig. 1. Experiments used in the definition of SIM-SO-CCA security of PKE

Definition 1 (SIM-SO-CCA Security). *A PKE scheme* $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ *is simulation-based, selective-opening, chosen-ciphertext secure (SIM-SO-CCA secure) if for every PPT n-message sampler $\mathcal{M}$, every PPT relation R, every stateful PPT adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3)$, there is a stateful PPT simulator $\mathcal{S} = (\mathcal{S}_1, \mathcal{S}_2, \mathcal{S}_3)$ such that* $\mathsf{Adv}^{\text{so-cca}}_{\mathsf{PKE},\mathcal{A},\mathcal{S},n,\mathcal{M},R}(\lambda)$ *is negligible, where*

$$\mathsf{Adv}^{\text{so-cca}}_{\mathsf{PKE},\mathcal{A},\mathcal{S},n,\mathcal{M},R}(\lambda) := \left| \Pr\left[\mathsf{Exp}^{\text{so-cca-real}}_{\mathsf{PKE},\mathcal{A},n,\mathcal{M},R}(\lambda) = 1\right] - \Pr\left[\mathsf{Exp}^{\text{so-cca-ideal}}_{\mathcal{S},n,\mathcal{M},R}(\lambda) = 1\right] \right|.$$

Experiments $\mathsf{Exp}^{\text{so-cca-real}}_{\mathsf{PKE},\mathcal{A},n,\mathcal{M},R}(\lambda)$ *and* $\mathsf{Exp}^{\text{so-cca-ideal}}_{\mathcal{S},n,\mathcal{M},R}(\lambda)$ *are defined in Figure 1. Here the restriction on $\mathcal{A}$ is that $\mathcal{A}_2, \mathcal{A}_3$ are not allowed to query the decryption oracle* $\mathsf{Dec}(\cdot)$ *with any challenge ciphertext* $\mathbf{C}_i \in \mathbf{C}$.

2.3 Efficiently Samplable and Explainable (ESE) Domain

A domain $\mathcal{D}$ is said to be *efficiently samplable and explainable* (ESE) [FHKW10] if there exist two PPT algorithms $(\mathsf{Sample}_{\mathcal{D}}, \mathsf{Sample}_{\mathcal{D}}^{-1})$ where $\mathsf{Sample}_{\mathcal{D}}(1^\lambda)$ outputs a uniform element over $\mathcal{D}$ and $\mathsf{Sample}_{\mathcal{D}}^{-1}(x)$, on input $x \in \mathcal{D}$, outputs r that is uniformly distributed over the set $\{r \in \mathcal{R}_{\mathsf{Sample}_{\mathcal{D}}} \mid \mathsf{Sample}_{\mathcal{D}}(1^\lambda; r) = x\}$.

It was shown by Damgård and Nielsen in [DN00] that any dense subset of an efficiently samplable domain is ESE as long as the dense subset admits an efficient membership test.

2.4 Cross-Authentication Codes

The concept of XAC was first proposed by Fehr *et al.* in [FHKW10] and later adapted to strong XAC in [HLQC13] and strengthened XAC in [LDL$^+$14].

Definition 2 (ℓ-Cross-Authentication Code, XAC). *An ℓ-cross-authentication code* XAC *(for $\ell \in \mathbb{N}$) consists of three PPT algorithms* $(\mathsf{XGen}, \mathsf{XAuth}, \mathsf{XVer})$ *and two associated spaces, the key space $\mathcal{XK}$ and the tag space $\mathcal{XT}$. The key generation algorithm* $\mathsf{XGen}(1^\lambda)$ *outputs a uniformly random key $K \in \mathcal{XK}$, the authentication algorithm* $\mathsf{XAuth}(K_1, \cdots, K_\ell)$ *takes ℓ keys $(K_1, \cdots, K_\ell) \in \mathcal{XK}^\ell$ as input and outputs a tag $T \in \mathcal{XT}$, and the verification algorithm* $\mathsf{XVer}(K, T)$ *outputs a decision bit.*

Correctness. $\mathsf{fail}_{\mathsf{XAC}}(\lambda) := \Pr[\mathsf{XVer}(K_i, \mathsf{XAuth}(K_1, \cdots, K_\ell)) \neq 1]$ *is negligible for all $i \in [\ell]$, where the probability is taken over $K_1, \cdots, K_\ell \leftarrow_\$ \mathcal{XK}$.*

Security against impersonation and substitution attacks. *Define* $\epsilon^{\mathrm{imp}}_{\mathsf{XAC}}(\lambda) := \max_{T'} \Pr[\mathsf{XVer}(K, T') = 1 \mid K \leftarrow_\$ \mathcal{XK}]$ *where* max *is over all $T' \in \mathcal{XT}$, and* $\epsilon^{\mathrm{sub}}_{\mathsf{XAC}}(\lambda) := \max\limits_{i, K_{\neq i}, F} \Pr\left[\begin{array}{c} T' \neq T \\ \mathsf{XVer}(K_i, T') = 1 \end{array} \middle| \begin{array}{c} K_i \leftarrow_\$ \mathcal{XK}, \\ T \leftarrow \mathsf{XAuth}(K_1, \cdots, K_\ell), \\ T' \leftarrow F(T) \end{array}\right]$ *where* max *is over all $i \in [\ell]$, all $K_{\neq i} := (K_j)_{j \in [\ell \setminus i]} \in \mathcal{XK}^{\ell-1}$ and all (possibly randomized) functions $F : \mathcal{XT} \to \mathcal{XT}$. Then we say* XAC *is secure against impersonation and substitution attacks if both $\epsilon^{\mathrm{imp}}_{\mathsf{XAC}}(\lambda)$ and $\epsilon^{\mathrm{sub}}_{\mathsf{XAC}}(\lambda)$ are negligible.*

Definition 3 (Strong and semi-unique XACs). *An ℓ-cross-authentication code* XAC *is* strong *and* semi-unique *if it has the following two properties.*

Strongness [HLQC13]. *There exists a PPT algorithm* ReSamp*, which takes as input $T \in \mathcal{XT}$ and $i \in [\ell]$, with $K_1, \cdots, K_\ell \leftarrow_\$ \mathsf{XGen}(1^\lambda), T \leftarrow \mathsf{XAuth}(K_1, \cdots, K_\ell)$, and outputs $\hat{K}_i \in \mathcal{XK}$, denoted by $\hat{K}_i \leftarrow_\$ \mathsf{ReSamp}(T, i)$. Suppose for each fixed $(k_1, \cdots, k_{\ell-1}, t) \in (\mathcal{XK})^{\ell-1} \times \mathcal{XT}$, the statistical distance between $\hat{K}_i$ and K_i, conditioned on $(K_{\neq i}, T) = (k_1, \cdots, k_{\ell-1}, t)$, is bounded by $\delta(\lambda)$, i.e.,*

$$\frac{1}{2} \sum_{k \in \mathcal{XK}} \left| \begin{array}{c} \Pr[\hat{K}_i = k \mid (K_{\neq i}, T) = (k_1, \cdots, k_{\ell-1}, t)] \\ - \\ \Pr[K_i = k \mid (K_{\neq i}, T) = (k_1, \cdots, k_{\ell-1}, t)] \end{array} \right| \leq \delta(\lambda).$$

Then the code XAC *is said to be $\delta(\lambda)$-*strong *or* strong *if $\delta(\lambda)$ is negligible.*

Semi-Uniqueness [LDL$^+$14]. *The code* XAC *is said to be* semi-unique *if $\mathcal{XK} = \mathcal{K}_x \times \mathcal{K}_y$, and given $T \in \mathcal{XT}$ and $K^x \in \mathcal{K}_x$, there exists at most one $K^y \in \mathcal{K}_y$ such that $\mathsf{XVer}((K^x, K^y), T) = 1$.*

A concrete XAC instantiation by Fehr *et al.* in [FHKW10] is shown in Appendix A.4.

3 Key Encapsulation Mechanism

In this section, we recall the definition of key encapsulation mechanism and formalize two new security notions for it.

Definition 4 (Key Encapsulation Mechanism). *A* KEM KEM *is a tuple of PPT algorithms* $(\mathsf{KGen}, \mathsf{KEnc}, \mathsf{KDec})$ *such that,* $\mathsf{KGen}(1^\lambda)$ *generates a (public, secret) key pair* $(\mathsf{pk_{kem}}, \mathsf{sk_{kem}})$*;* $\mathsf{KEnc}(\mathsf{pk_{kem}})$ *returns an encapsulation* $\psi \in \Psi$ *and a key* $\gamma \in \Gamma$*, where* Ψ *is the encapsulation space and* Γ *is the key space;* $\mathsf{KDec}(\mathsf{sk_{kem}}, \psi)$ *deterministically decapsulates* ψ *with* $\mathsf{sk_{kem}}$ *to get* $\gamma \in \Gamma$ *or* $\perp$.

We say KEM *is* perfectly correct *if for all* λ, $\Pr[\mathsf{KDec}(\mathsf{sk_{kem}}, \psi) = \gamma] = 1$, *where* $(\mathsf{pk_{kem}}, \mathsf{sk_{kem}}) \leftarrow_\$ \mathsf{KGen}(1^\lambda)$ *and* $(\psi, \gamma) \leftarrow_\$ \mathsf{KEnc}(\mathsf{pk_{kem}})$.

3.1 mPR-CCCA Security for KEM

We formalize a new security notion for KEM, namely mPR-CCCA. Roughly speaking, mPR-CCCA security guarantees pseudorandomness of multiple (ψ, γ) pairs outputted by KEnc even if a constrained decapsulation oracle is provided.

Definition 5 (mPR-CCCA Security for KEM). *Let* $\mathcal{A}$ *be an adversary and* $b \in \{0, 1\}$ *be a bit. Let* $\mathsf{KEM} = (\mathsf{KGen}, \mathsf{KEnc}, \mathsf{KDec})$ *be a KEM with encapsulation space* Ψ *and key space* Γ*. Define the experiment* $\mathsf{Exp}^{\text{mpr-ccca-}b}_{\mathsf{KEM},\mathcal{A}}(\lambda)$ *in Figure 2.*

```
$\mathsf{Exp}^{\text{mpr-ccca-}b}_{\mathsf{KEM},\mathcal{A}}(\lambda)$: //$b \in \{0,1\}$
    $(\mathsf{pk_{kem}}, \mathsf{sk_{kem}}) \leftarrow_\$ \mathsf{KGen}(1^\lambda)$
    $b' \leftarrow_\$ \mathcal{A}^{\mathcal{O}_{\rm enc}(), \mathcal{O}_{\rm dec}(\cdot,\cdot)}(\mathsf{pk_{kem}})$
    Return $b'$

$\mathcal{O}_{\rm enc}()$:
    $(\psi_0, \gamma_0) \leftarrow_\$ \Psi \times \Gamma$
    $(\psi_1, \gamma_1) \leftarrow_\$ \mathsf{KEnc}(\mathsf{pk_{kem}})$
    $\boldsymbol{\psi}_{\rm enc} \leftarrow \boldsymbol{\psi}_{\rm enc} \cup \{\psi_b\}$
    Return $(\psi_b, \gamma_b)$

$\mathcal{O}_{\rm dec}(\mathsf{pred}, \psi)$:
    $\gamma \leftarrow \mathsf{KDec}(\mathsf{sk_{kem}}, \psi)$
    Return $\begin{cases} \gamma & \text{If } \begin{pmatrix} \psi \notin \boldsymbol{\psi}_{\rm enc} \wedge \\ \mathsf{pred}(\gamma) = 1 \end{pmatrix} \\ \perp & \text{Otherwise} \end{cases}$
```

Fig. 2. Experiment used in the definition of mPR-CCCA security of KEM

In $\mathsf{Exp}^{\text{mpr-ccca-}b}_{\mathsf{KEM},\mathcal{A}}(\lambda)$, $\mathsf{pred} : \Gamma \cup \{\perp\} \to \{0, 1\}$ *denotes a PPT predicate and* $\mathsf{pred}(\perp) := 0$. *Let* Q_{dec} *be the total number of decapsulation queries made by* $\mathcal{A}$*, which is independent of the environment without loss of generality. The* uncertainty *of* $\mathcal{A}$ *is defined as* $uncert_{\mathcal{A}}(\lambda) := \frac{1}{Q_{dec}} \sum_{i=1}^{Q_{dec}} \Pr_{\gamma \leftarrow_\$ \Gamma}[\mathsf{pred}_i(\gamma) = 1]$, *where* pred_i *is the predicate in the* i*-th* $\mathcal{O}_{dec}$ *query.*

We say KEM *has* multi-encapsulation pseudorandom security against constrained CCA adversaries (mPR-CCCA security) *if for each PPT adversary* $\mathcal{A}$ *with negligible uncertainty* $uncert_{\mathcal{A}}(\lambda)$*, the advantage* $\mathsf{Adv}^{\text{mpr-ccca}}_{\mathsf{KEM},\mathcal{A}}(\lambda)$ *is negligible, where* $\mathsf{Adv}^{\text{mpr-ccca}}_{\mathsf{KEM},\mathcal{A}}(\lambda) := \left| \Pr\left[\mathsf{Exp}^{\text{mpr-ccca-}0}_{\mathsf{KEM},\mathcal{A}}(\lambda) = 1\right] - \Pr\left[\mathsf{Exp}^{\text{mpr-ccca-}1}_{\mathsf{KEM},\mathcal{A}}(\lambda) = 1\right] \right|$.

Note that the afore-defined mPR-CCCA security implies multi-challenge IND-CCCA security defined in [GHK17].

3.2 RER Security of KEM

We define Random Encapsulation Rejection security for KEM which requires the decapsulation of a random encapsulation is rejected overwhelmingly.

Definition 6 (Random Encapsulation Rejection Security for KEM). *Let* $\mathsf{KEM} = (\mathsf{KGen}, \mathsf{KEnc}, \mathsf{KDec})$ *be a KEM with encapsulation space* Ψ *and key space* Γ*. Let* $\mathcal{A}$ *be a stateful adversary and* $b \in \{0,1\}$ *be a bit. Define the following experiment* $\mathsf{Exp}^{\text{rer-}b}_{\mathsf{KEM},\mathcal{A}}(\lambda)$ *in Figure 3.*

```
$\underline{\mathsf{Exp}^{\text{rer-}b}_{\mathsf{KEM},\mathcal{A}}(\lambda)\text{:}}$        $//b \in \{0,1\}$
$(\mathsf{pk_{kem}}, \mathsf{sk_{kem}}) \leftarrow_\$ \mathsf{KGen}(1^\lambda)$
$\boldsymbol{\psi}_{\text{ran}} \leftarrow \emptyset$
$(st, 1^n) \leftarrow_\$ \mathcal{A}^{\mathcal{O}_{\text{cha}}(\cdot,\cdot)}(\mathsf{pk_{kem}})$
$\boldsymbol{\psi}_{\text{ran}} = \{\psi_1, \cdots, \psi_n\} \leftarrow_\$ \Psi^n$
$b' \leftarrow_\$ \mathcal{A}^{\mathcal{O}_{\text{cha}}(\cdot,\cdot)}(st, \boldsymbol{\psi}_{\text{ran}})$
Return $b'$
```

```
$\underline{\mathcal{O}_{\text{cha}}(\mathsf{pred}, \psi)\text{:}}$
If $\psi \notin \boldsymbol{\psi}_{\text{ran}}$:
    Return $\mathsf{pred}(\mathsf{KDec}(\mathsf{sk_{kem}}, \psi))$
If $b = 1$:
    Return $\mathsf{pred}(\mathsf{KDec}(\mathsf{sk_{kem}}, \psi))$
Else:
    Return 0
```

Fig. 3. Experiment used in the definition of RER property of KEM

In $\mathsf{Exp}^{\text{rer-}b}_{\mathsf{KEM},\mathcal{A}}(\lambda)$*,* $\mathsf{pred} : \Gamma \cup \{\perp\} \rightarrow \{0,1\}$ *denotes a PPT predicate and* $\mathsf{pred}(\perp) := 0$*. Let* Q_{cha} *be the total number of* $\mathcal{O}_{cha}$ *queries made by* $\mathcal{A}$*, which is independent of the environment without loss of generality. The* uncertainty *of* $\mathcal{A}$ *is defined as* $\mathit{uncert}_{\mathcal{A}}(\lambda) := \frac{1}{Q_{cha}} \sum_{i=1}^{Q_{cha}} \Pr_{\gamma \leftarrow_\$ \Gamma}[\mathsf{pred}_i(\gamma) = 1]$*, where* pred_i *is the predicate in the* i*-th* $\mathcal{O}_{cha}$ *query.*

We say KEM *has* Random Encapsulation Rejection security (RER security) *if for each PPT adversary* $\mathcal{A}$ *with negligible uncertainty* $\mathit{uncert}_{\mathcal{A}}(\lambda)$*, the advantage*

$$\mathsf{Adv}^{\text{rer}}_{\mathsf{KEM},\mathcal{A}}(\lambda) := \left|\Pr\left[\mathsf{Exp}^{\text{rer-0}}_{\mathsf{KEM},\mathcal{A}}(\lambda) = 1\right] - \Pr\left[\mathsf{Exp}^{\text{rer-1}}_{\mathsf{KEM},\mathcal{A}}(\lambda) = 1\right]\right| \text{ is negligible.}$$

4 SIM-SO-CCA Secure PKE from KEM

4.1 PKE Construction

In Figure 4, we recall the general framework for constructing SIM-SO-CCA secure PKE proposed in [LP15]. A small difference from [LP15] is that we make use of hash function H_1 to convert the key space of KEM to the key space of XAC.

Ingredients. This construction uses the following ingredients.

- KEM=$(\mathsf{KGen}, \mathsf{KEnc}, \mathsf{KDec})$ with key space Γ & ESE encapsulation space Ψ.
- $(\ell + 1)$-XAC XAC with ESE key space $\mathcal{XK} = \mathcal{K}_x \times \mathcal{K}_y$.
- Hash function $\mathsf{H}_1 : \Gamma \rightarrow \mathcal{XK}$ generated by hash function generator $\mathcal{H}_1(1^\lambda)$.
- Hash function $\mathsf{H}_2 : \Psi^\ell \rightarrow \mathcal{K}_y$ generated by hash function generator $\mathcal{H}_2(1^\lambda)$.

4.2 Tight Security Proof of PKE

In this subsection, we prove the SIM-SO-CCA security of PKE with tight reduction to the security of KEM. We state our main result in the following theorem.

```
$\mathsf{Gen}(1^\lambda)$:
    $(\mathsf{pk}_{\mathsf{kem}}, \mathsf{sk}_{\mathsf{kem}}) \leftarrow_\$ \mathsf{KGen}(1^\lambda)$
    $\mathsf{H}_1 \leftarrow_\$ \mathcal{H}_1(1^\lambda)$
    $\mathsf{H}_2 \leftarrow_\$ \mathcal{H}_2(1^\lambda)$
    $K^x \leftarrow_\$ \mathcal{K}_x$
    $\mathsf{pk} \leftarrow (\mathsf{pk}_{\mathsf{kem}}, \mathsf{H}_1, \mathsf{H}_2, K^x)$
    $\mathsf{sk} \leftarrow (\mathsf{pk}, \mathsf{sk}_{\mathsf{kem}})$
Return $(\mathsf{pk}, \mathsf{sk})$
```

```
$\mathsf{Enc}(\mathsf{pk}, \mathbf{m} \in \{0,1\}^\ell)$:
    For $j \leftarrow 1$ to $\ell$:
        If $\mathbf{m}_j = 1$:
            $(\psi_j, \gamma_j) \leftarrow_\$ \mathsf{KEnc}(\mathsf{pk}_{\mathsf{kem}})$
            $K_j \leftarrow \mathsf{H}_1(\gamma_j)$
        Else:
            $\psi_j \leftarrow_\$ \Psi$
            $K_j \leftarrow_\$ \mathcal{XK}$
    $K^y \leftarrow \mathsf{H}_2(\psi_1, \cdots, \psi_\ell)$
    $K_{\ell+1} \leftarrow (K^x, K^y)$
    $T \leftarrow \mathsf{XAuth}(K_1, \cdots, K_{\ell+1})$
Return $C \leftarrow (\psi_1, \cdots, \psi_\ell, T)$
```

```
$\mathsf{Dec}(\mathsf{sk}, C = (\psi_1, \cdots, \psi_\ell, T))$:
    $\mathbf{m}' \leftarrow 0^\ell$
    $K^{y\prime} \leftarrow \mathsf{H}_2(\psi_1, \cdots, \psi_\ell)$
    $K'_{\ell+1} \leftarrow (K^x, K^{y\prime})$
    If $\mathsf{XVer}(K'_{\ell+1}, T) = 1$:
        For $j \leftarrow 1$ to $\ell$:
            $\gamma'_j \leftarrow \mathsf{KDec}(\mathsf{sk}_{\mathsf{kem}}, \psi_j)$
            $K'_j \leftarrow \mathsf{H}_1(\gamma'_j)$
            $\mathbf{m}'_j \leftarrow \mathsf{XVer}(K'_j, T)$
Return $\mathbf{m}'$
```

Fig. 4. Construction of $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$.

Theorem 1. *Suppose the KEM* KEM *is mPR-CCCA and RER secure, the* $(\ell+1)$*-cross-authentication code* XAC *is* $\delta(\lambda)$*-strong, semi-unique, and secure against impersonation and substitution attacks;* $\mathcal{H}_1$ *is universal;* $\mathcal{H}_2$ *outputs collision resistant function. Then the PKE scheme* PKE *constructed in Figure 4 is SIM-SO-CCA secure. More precisely, for each PPT adversary* $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3)$ *against* PKE *in the SIM-SO-CCA real experiment, for each PPT* n*-message sampler* $\mathcal{M}$*, and each PPT relation* R*, we can construct a stateful PPT simulator* $\mathcal{S} = (\mathcal{S}_1, \mathcal{S}_2, \mathcal{S}_3)$ *for the SIM-SO-CCA ideal experiment and PPT adversaries* $\mathcal{B}_1, \mathcal{B}_2, \mathcal{B}_3$ *with* $\mathbf{T}(\mathcal{B}_1) \approx \mathbf{T}(\mathcal{B}_2) \approx \mathbf{T}(\mathcal{B}_3) \leq \mathbf{T}(\mathcal{A}) + Q_{\mathrm{dec}} \cdot \mathsf{poly}(\lambda)$*, such that*

$$\begin{aligned}\mathsf{Adv}^{\text{so-cca}}_{\mathsf{PKE},\mathcal{A},\mathcal{S},n,\mathcal{M},R}(\lambda) \leq\ & \mathsf{Adv}^{\text{mpr-ccca}}_{\mathsf{KEM},\mathcal{B}_2}(\lambda) + \mathsf{Adv}^{\text{rer}}_{\mathsf{KEM},\mathcal{B}_3}(\lambda) + \ell \cdot Q_{\mathrm{dec}} \cdot \epsilon^{\text{sub}}_{\mathsf{XAC}}(\lambda) \\ & + 2\mathsf{Adv}^{\text{cr}}_{\mathcal{H},\mathcal{B}_1}(\lambda) + (n\ell) \cdot (\delta(\lambda) + \Delta),\end{aligned} \qquad (1)$$

where Q_{dec} *denotes the total number of* $\mathcal{A}$*'s decryption oracle queries,* $\mathsf{poly}(\lambda)$ *is a polynomial independent of* $\mathbf{T}(\mathcal{A})$ *and* $\Delta = \frac{1}{2} \cdot \sqrt{|\mathcal{XK}|/|\Gamma|}$*.*

Remark. If we instantiate the construction with the information-theoretically secure XAC in Appendix A.4 and choose proper set $\mathcal{XK}$ and Γ, then $\Delta, \delta(\lambda)$, $\epsilon^{\text{imp}}_{\mathsf{XAC}}(\lambda)$ and $\epsilon^{\text{sub}}_{\mathsf{XAC}}(\lambda)$ are all exponentially small in λ. Then (1) turns out to be

$$\mathsf{Adv}^{\text{so-cca}}_{\mathsf{PKE},\mathcal{A},\mathcal{S},n,\mathcal{M},R}(\lambda) \leq \mathsf{Adv}^{\text{mpr-ccca}}_{\mathsf{KEM},\mathcal{B}_2}(\lambda) + \mathsf{Adv}^{\text{rer}}_{\mathsf{KEM},\mathcal{B}_3}(\lambda) + 2\mathsf{Adv}^{\text{cr}}_{\mathcal{H},\mathcal{B}_1}(\lambda) + 2^{-\Omega(\lambda)}.$$

If the underlying KEM has tight mPR-CCCA security and RER security, then our PKE turns out to be tightly SIM-SO-CCA secure.

Proof of Theorem 1. For each PPT adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3)$, we can construct a stateful PPT simulator $\mathcal{S} = (\mathcal{S}_1, \mathcal{S}_2, \mathcal{S}_3)$ as shown in Figure 5.

In Appendix B, we illustrate the detailed ideas of the construction for $\mathcal{S}$.

The differences between the real and the ideal experiments lie in two aspects. The first is how the challenge ciphertext vector is generated and the second is how the corrupted ciphertexts are opened. In other words, the algorithms $\mathsf{SimCtGen}$ and $\mathsf{SimOpen}$ used by the simulator differ from the real experiment. In the proof, we focus on these two algorithms and gradually change them through a series of games starting with game G_0 and ending with game G_9, with adjacent games being proved to be computationally

```
$\underline{\mathcal{S}_1(1^\lambda)}$:
    $(\mathsf{pk},\mathsf{sk}) \leftarrow_\$ \mathsf{SimKeyGen}(1^\lambda)$
    $(\alpha, a_1) \leftarrow_\$ \mathcal{A}_1^{\mathsf{Dec}(\cdot)}(\mathsf{pk})$
    Return $(\alpha, s_1 = (\mathsf{pk},\mathsf{sk},a_1))$

$\underline{\mathcal{S}_2(s_1, (1^{|\mathbf{m}_i|})_{i\in[n]})}$:
    $(\mathbf{C},\mathbf{R},\mathbf{K}) \leftarrow_\$ \mathsf{SimCtGen}(\mathsf{pk})$
    $(I, a_2) \leftarrow_\$ \mathcal{A}_2^{\mathsf{Dec}_{\notin \mathbf{C}}(\cdot)}(a_1, \mathbf{C})$
    Return $(I, s_2 = (s_1, a_2, I, \mathbf{C}, \mathbf{R}, \mathbf{K}))$

$\underline{\mathcal{S}_3(s_2, \mathbf{m}_I)}$:
    $\hat{\mathbf{R}}_I \leftarrow_\$ \mathsf{SimOpen}(I, \mathbf{m}_I, \mathbf{C}, \mathbf{R}, \mathbf{K})$
    $out_{\mathcal{A}} \leftarrow_\$ \mathcal{A}_3^{\mathsf{Dec}_{\notin \mathbf{C}}(\cdot)}(a_2, \mathbf{m}_I, \hat{\mathbf{R}}_I)$
    Return $out_{\mathcal{A}}$
```

```
$\underline{\mathsf{SimKeyGen}(1^\lambda)}$:
    $(\mathsf{pk_{kem}},\mathsf{sk_{kem}}) \leftarrow_\$ \mathsf{KGen}(1^\lambda), \mathsf{H}_1 \leftarrow_\$ \mathcal{H}_1(1^\lambda), \mathsf{H}_2 \leftarrow_\$ \mathcal{H}_2(1^\lambda), K^x \leftarrow_\$ \mathcal{K}_x$
    $\mathsf{pk} \leftarrow (\mathsf{pk_{kem}}, \mathsf{H}_1, \mathsf{H}_2, K^x), \mathsf{sk} \leftarrow (\mathsf{pk}, \mathsf{sk_{kem}})$
Return $(\mathsf{pk},\mathsf{sk})$
```

```
$\underline{\mathsf{SimCtGen}(\mathsf{pk})}$:
    For $i \leftarrow 1$ to $n$:
        For $j \leftarrow 1$ to $\ell$:
            $r_{i,j} \leftarrow_\$ \mathcal{R}_{\mathsf{KEnc}}$
            $(\psi_{i,j}, \gamma_{i,j}) \leftarrow \mathsf{KEnc}(\mathsf{pk_{kem}}; r_{i,j})$
            $K_{i,j} \leftarrow \mathsf{H}_1(\gamma_{i,j})$
        $K_i^y \leftarrow \mathsf{H}_2(\psi_{i,1}, \cdots, \psi_{i,\ell})$
        $K_{i,\ell+1} \leftarrow (K^x, K_i^y)$
        $T_i \leftarrow \mathsf{XAuth}(K_{i,1}, \cdots, K_{i,\ell+1})$
        $\mathbf{C}_i \leftarrow (\psi_{i,1}, \cdots, \psi_{i,\ell}, T_i)$
        $\mathbf{R}_i \leftarrow (r_{i,1}, \cdots, r_{i,\ell})$
        $\mathbf{K}_i \leftarrow (K_{i,1}, \cdots, K_{i,\ell+1})$
Return $\begin{pmatrix}\mathbf{C}\\ \mathbf{R}\\ \mathbf{K}\end{pmatrix} = \begin{pmatrix}\mathbf{C}_1, \cdots, \mathbf{C}_n\\ \mathbf{R}_1, \cdots, \mathbf{R}_n\\ \mathbf{K}_1, \cdots, \mathbf{K}_n\end{pmatrix}$
```

```
$\underline{\mathsf{SimOpen}(I, \mathbf{m}_I, \mathbf{C}, \mathbf{R}, \mathbf{K})}$:
    For $i \in I$:
        For $j \leftarrow 1$ to $\ell$:
            If $\mathbf{m}_{i,j} = 1$:
                $\hat{r}_{i,j} \leftarrow r_{i,j}$
            Else:
                $\hat{K}_{i,j} \leftarrow_\$ \mathsf{ReSamp}(T_i, j)$
                $\hat{r}^K_{i,j} \leftarrow_\$ \mathsf{Sample}^{-1}_{\mathcal{XK}}(\hat{K}_{i,j})$
                $\hat{r}^\psi_{i,j} \leftarrow_\$ \mathsf{Sample}^{-1}_{\Psi}(\psi_{i,j})$
                $\hat{r}_{i,j} \leftarrow (\hat{r}^K_{i,j}, \hat{r}^\psi_{i,j})$
        $\hat{\mathbf{R}}_i \leftarrow (\hat{r}_{i,1}, \cdots, \hat{r}_{i,\ell})$
    Return $\hat{\mathbf{R}}_I = (\hat{\mathbf{R}}_i)_{i\in I}$
```

Fig. 5. Construction of simulator $\mathcal{S} = (\mathcal{S}_1, \mathcal{S}_2, \mathcal{S}_3)$ for $\mathsf{Exp}^{\text{so-cca-ideal}}_{\mathcal{S},n,\mathcal{M},R}(\lambda)$.

indistinguishable. The full set of games are illustrated in Figure 6.

Game G_0. Game G_0 is exactly the ideal experiment $\mathsf{Exp}^{\text{so-cca-ideal}}_{\mathcal{S},n,\mathcal{M},R}(\lambda)$. Hence

$$\Pr\left[\mathsf{Exp}^{\text{so-cca-ideal}}_{\mathcal{S},n,\mathcal{M},R}(\lambda) = 1\right] = \Pr_0[G \Rightarrow 1]. \tag{2}$$

Game $G_0 - G_1$. The only difference between G_1 and G_0 is that a collision check for H_2 is added in G_1 and G_1 aborts if a collision is found. More precisely, we use a set $\mathcal{Q}$ to log all the (input, output) pairs for H_2 in algorithm $\mathsf{SimCtGen}$. Then in the Dec oracle, if there exists a usage of H_2 such that its output collides with some output in $\mathcal{Q}$ but with different inputs, then a collision for H_2 is found and the game G_1 aborts immediately. It is straightforward to build a PPT adversary $\mathcal{B}_1$ with $\mathbf{T}(\mathcal{B}_1) \approx \mathbf{T}(\mathcal{A}) + Q_{\text{dec}} \cdot \mathsf{poly}(\lambda)$, where $\mathsf{poly}(\lambda)$ is a polynomial independent of $\mathbf{T}(\mathcal{A})$, such that,

$$|\Pr_0[G \Rightarrow 1] - \Pr_1[G \Rightarrow 1]| \leq \mathsf{Adv}^{\text{cr}}_{\mathcal{H},\mathcal{B}_1}(\lambda). \tag{3}$$

Game $G_1 - G_2$. G_2 is essentially the same as G_1 except for one conceptual change in the Dec oracle. More precisely, for a $\mathsf{Dec}(C = (\psi_1, \cdots, \psi_\ell, T))$ query such that $\exists (i,j) \in [n] \times [\ell], \eta \in [\ell]$ s.t. $\mathbf{m}_{i,j} = 0 \wedge \psi_\eta = \psi_{i,j}$,

- in G_1, we proceed exactly the same as the decryption algorithm, i.e.,
  set $\mathbf{m}'_\eta \leftarrow \mathsf{XVer}(\mathsf{H}_1(\gamma'_\eta), T)$ where $\gamma'_\eta = \mathsf{KDec}(\mathsf{sk_{kem}}, \psi_\eta)$;
- in G_2, we set $\mathbf{m}'_\eta \leftarrow \mathsf{XVer}(K_{i,j}, T)$.

Since $\psi_\eta = \psi_{i,j}$, $\gamma'_\eta = \mathsf{KDec}(\mathsf{sk_{kem}}, \psi_\eta)$ and $(\psi_{i,j}, \gamma_{i,j})$ is the output of $\mathsf{KEnc}(\mathsf{pk_{kem}})$, we have that $\gamma'_\eta = \gamma_{i,j}$ due to the perfect correctness of KEM. Then $K_{i,j} = \mathsf{H}_1(\gamma_{i,j}) = \mathsf{H}_1(\gamma'_\eta)$. Thus the difference between G_1 and G_2 is only conceptual, and it follows

$$\Pr_1[G \Rightarrow 1] = \Pr_2[G \Rightarrow 1]. \tag{4}$$

```
$\mathsf{Exp}^{\text{so-cca-ideal}}_{\mathcal{S},n,\mathcal{M},R}(\lambda)$:
$(\mathsf{pk},\mathsf{sk}) \leftarrow_\$ \mathsf{SimKeyGen}(1^\lambda)$
$(\alpha, a_1) \leftarrow_\$ \mathcal{A}_1^{\mathsf{Dec}(\cdot)}(\mathsf{pk})$
$\mathbf{m} \leftarrow_\$ \mathcal{M}(\alpha)$
$(\mathbf{C},\mathbf{R},\mathbf{K}) \leftarrow_\$ \mathsf{SimCtGen}(\mathsf{pk})$
$(I, a_2) \leftarrow_\$ \mathcal{A}_2^{\mathsf{Dec}_{\notin \mathbf{C}}(\cdot)}(a_1,\mathbf{C})$
$\hat{\mathbf{R}}_I \leftarrow_\$ \mathsf{SimOpen}(I,\mathbf{m}_I,\mathbf{C},\mathbf{R},\mathbf{K})$
$out_\mathcal{A} \leftarrow_\$ \mathcal{A}_3^{\mathsf{Dec}_{\notin \mathbf{C}}(\cdot)}(a_2,\mathbf{m}_I,\hat{\mathbf{R}}_I)$
Return $R(\mathbf{m}, I, out_\mathcal{A})$
```

```
$\mathsf{SimCtGen}(\mathsf{pk})$:
  $G_0$ $\boxed{G_1, G_2}$ $\boxed{G_3\ \boxed{G_4 - G_7}}$ $\boxed{G_8\ \boxed{G_9}}$
For $i \leftarrow 1$ to $n$:
  For $j \leftarrow 1$ to $\ell$:
    If $\mathbf{m}_{i,j} = 0$:
      $r^\psi_{i,j} \leftarrow_\$ \mathcal{R}_{\mathsf{Sample}_\Psi}$
      $\psi_{i,j} \leftarrow \mathsf{Sample}_\Psi(1^\lambda; r^\psi_{i,j})$
      $\gamma_{i,j} \leftarrow_\$ \Gamma$
      $K_{i,j} \leftarrow \mathsf{H}_1(\gamma_{i,j})$
      $\boxed{r^K_{i,j} \leftarrow_\$ \mathcal{R}_{\mathsf{Sample}_{\mathcal{XK}}}}$
      $\boxed{K_{i,j} \leftarrow \mathsf{Sample}_{\mathcal{XK}}(1^\lambda; r^K_{i,j})}$
      $\boxed{r_{i,j} \leftarrow (r^K_{i,j}, r^\psi_{i,j})}$
    Else:
      $r_{i,j} \leftarrow_\$ \mathcal{R}_{\mathsf{KEnc}}$
      $(\psi_{i,j},\gamma_{i,j}) \leftarrow \mathsf{KEnc}(\mathsf{pk}_{\mathsf{kem}}; r_{i,j})$
      $K_{i,j} \leftarrow \mathsf{H}_1(\gamma_{i,j})$
  $K^y_i \leftarrow \mathsf{H}_2(\psi_{i,1},\cdots,\psi_{i,\ell})$
  $\boxed{\mathcal{Q} \leftarrow \mathcal{Q} \cup \{(K^y_i,(\psi_{i,1},\cdots,\psi_{i,\ell}))\}}$
  $K_{i,\ell+1} \leftarrow (K^x, K^y_i)$
  $T_i \leftarrow \mathsf{XAuth}(K_{i,1},\cdots,K_{i,\ell+1})$
  $\mathbf{C}_i \leftarrow (\psi_{i,1},\cdots,\psi_{i,\ell},T_i)$
  $\mathbf{R}_i \leftarrow (r_{i,1},\cdots,r_{i,\ell})$
  $\mathbf{K}_i \leftarrow (K_{i,1},\cdots,K_{i,\ell+1})$
Return $\begin{pmatrix}\mathbf{C}\\ \mathbf{R}\\ \mathbf{K}\end{pmatrix} = \begin{pmatrix}\mathbf{C}_1,\cdots,\mathbf{C}_n\\ \mathbf{R}_1,\cdots,\mathbf{R}_n\\ \mathbf{K}_1,\cdots,\mathbf{K}_n\end{pmatrix}$
```

```
$\mathsf{SimOpen}(I,\mathbf{m}_I,\mathbf{C},\mathbf{R},\mathbf{K})$:
  $\boxed{G_0 - G_6}$ $\boxed{G_7, G_8}$ $G_9$
For $i \in I$:
  For $j \leftarrow 1$ to $\ell$:
    If $\mathbf{m}_{i,j} = 1$:
      $\hat{r}_{i,j} \leftarrow r_{i,j}$
    Else:
      $\boxed{\hat{K}_{i,j} \leftarrow_\$ \mathsf{ReSamp}(T_i, j)}$
      $\boxed{\hat{r}^K_{i,j} \leftarrow_\$ \mathsf{Sample}^{-1}_{\mathcal{XK}}(\hat{K}_{i,j})}$
      $\boxed{\hat{r}^K_{i,j} \leftarrow_\$ \mathsf{Sample}^{-1}_{\mathcal{XK}}(K_{i,j})}$
      $\hat{r}^\psi_{i,j} \leftarrow_\$ \mathsf{Sample}^{-1}_\Psi(\psi_{i,j})$
      $\hat{r}_{i,j} \leftarrow (\hat{r}^K_{i,j}, \hat{r}^\psi_{i,j})$
  $\hat{\mathbf{R}}_i \leftarrow (\hat{r}_{i,1},\cdots,\hat{r}_{i,\ell})$
$\hat{\mathbf{R}}_I \leftarrow \mathbf{R}_I$
Return $\hat{\mathbf{R}}_I$
```

```
$\mathsf{Dec}_{\notin \mathbf{C}}(C = (\psi_1,\cdots,\psi_\ell,T))$:
  $G_0$ $\boxed{G_1\ \boxed{G_2, G_3, G_4}\ \boxed{G_5}\ G_6, G_7}$ $G_8, G_9$
If $C \in \mathbf{C}$:
  Return $\perp$
$\mathbf{m} \leftarrow 0^\ell$
$K^{y\prime} \leftarrow \mathsf{H}_2(\psi_1,\cdots,\psi_\ell)$
$\boxed{\text{If } \begin{bmatrix}\exists(\hat{K}^y,(\hat{\psi}_1,\cdots,\hat{\psi}_\ell)) \in \mathcal{Q} \text{ s.t.}\\ K^{y\prime} = \hat{K}^y \wedge (\psi_1,\cdots,\psi_\ell) \neq (\hat{\psi}_1,\cdots,\hat{\psi}_\ell)\end{bmatrix}:}$
$\boxed{\quad \textbf{Abort game} \quad //\text{Find a collisoin for } \mathsf{H}_2}$
$\boxed{\mathcal{Q} \leftarrow \mathcal{Q} \cup \{(K^{y\prime},(\psi_1,\cdots,\psi_\ell))\}}$
$K'_{\ell+1} \leftarrow (K^x, K^{y\prime})$
If $\mathsf{XVer}(K'_{\ell+1}, T) = 1$:
  For $\eta \leftarrow 1$ to $\ell$:
    $\gamma'_\eta \leftarrow \mathsf{KDec}(\mathsf{sk}_{\mathsf{kem}}, \psi_\eta)$
    If $\begin{bmatrix}\exists(i,j) \in [n] \times [\ell] \text{ s.t.}\\ \mathbf{m}_{i,j} = 0 \wedge \psi_\eta = \psi_{i,j}\end{bmatrix}$:
      $\mathbf{m}'_\eta \leftarrow \mathsf{XVer}(\mathsf{H}_1(\gamma'_\eta), T)$
      $\boxed{\mathbf{m}'_\eta \leftarrow \mathsf{XVer}(K_{i,j}, T)}$
      $\boxed{\mathbf{m}'_\eta \leftarrow 0}$
    Else:
      $\mathbf{m}'_\eta \leftarrow \mathsf{XVer}(\mathsf{H}_1(\gamma'_\eta), T)$
Return $\mathbf{m}'$
```

```
$\mathsf{SimKeyGen}(1^\lambda)$:    $G_0$ $\boxed{G_1 - G_7}$ $G_8, G_9$
  $(\mathsf{pk}_{\mathsf{kem}}, \mathsf{sk}_{\mathsf{kem}}) \leftarrow_\$ \mathsf{KGen}(1^\lambda), \mathsf{H}_1 \leftarrow_\$ \mathcal{H}_1(1^\lambda), \mathsf{H}_2 \leftarrow_\$ \mathcal{H}_2(1^\lambda), K^x \leftarrow_\$ \mathcal{K}_x$
  $\mathsf{pk} \leftarrow (\mathsf{pk}_{\mathsf{kem}}, \mathsf{H}_1, \mathsf{H}_2, K^x), \mathsf{sk} \leftarrow (\mathsf{pk}, \mathsf{sk}_{\mathsf{kem}})$ $\boxed{\mathcal{T} \leftarrow \emptyset}$
Return $(\mathsf{pk}, \mathsf{sk})$
```

Fig. 6. Games $G_0 - G_9$ in the proof of Theorem 1.

Game $G_2 - G_3$. G_3 is almost the same as G_2 except for one change in the $\mathsf{SimCtGen}$ algorithm.

- In G_2, all $(\psi_{i,j}, \gamma_{i,j})$ pairs are the output of $\mathsf{KEnc}(\mathsf{pk}_{\mathsf{kem}})$.
- In G_3, for $\mathbf{m}_{i,j} = 1$, $(\psi_{i,j}, \gamma_{i,j})$ pairs are the output of $\mathsf{KEnc}(\mathsf{pk}_{\mathsf{kem}})$;
  for $\mathbf{m}_{i,j} = 0$, $(\psi_{i,j}, \gamma_{i,j})$ pairs are uniformly selected from $\Psi \times \Gamma$.

We will reduce the indistinguishability between game G_2 and G_3 to the mPR-CCCA security of KEM. Given $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3)$, we can build a PPT adversary $\mathcal{B}_2$ with $\mathbf{T}(\mathcal{B}_2) \approx \mathbf{T}(\mathcal{A})$ and uncertainty $uncert_{\mathcal{B}_2}(\lambda) \le \epsilon^{\text{imp}}_{\mathsf{XAC}}(\lambda) + \Delta$ such that

$$|\Pr_2[G \Rightarrow 1] - \Pr_3[G \Rightarrow 1]| \le \mathsf{Adv}^{\text{mpr-ccca}}_{\mathsf{KEM},\mathcal{B}_2}(\lambda). \tag{5}$$

On input $\mathsf{pk}_{\mathsf{kem}}$, $\mathcal{B}_2$ selects $\mathsf{H}_1, \mathsf{H}_2$ and K^x itself and embeds $\mathsf{pk}_{\mathsf{kem}}$ in $\mathsf{pk} = (\mathsf{pk}_{\mathsf{kem}}, \mathsf{H}_1, \mathsf{H}_2, K^x)$. In the first phase, $\mathcal{B}_2$ calls $\mathcal{A}_1^{\mathsf{Dec}(\cdot)}(\mathsf{pk})$. To respond the decryption query $\mathsf{Dec}(C = (\psi_1, \cdots, \psi_\ell, T))$ submitted by $\mathcal{A}$, $\mathcal{B}_2$ simulates Dec until it needs to call $\mathsf{KDec}(\mathsf{sk}_{\mathsf{kem}}, \psi_\eta)$ to decapsulate ψ_η. Since $\mathcal{B}_2$ does not possess $\mathsf{sk}_{\mathsf{kem}}$ relative to $\mathsf{pk}_{\mathsf{kem}}$, $\mathcal{B}_2$ is not able to invoke KDec itself. Then $\mathcal{B}_2$ submits a $\mathcal{O}_{\text{dec}}(\mathsf{pred}, \psi_\eta)$ query to its own oracle $\mathcal{O}_{\text{dec}}$ where $\mathsf{pred}(\cdot) := \mathsf{XVer}(\mathsf{H}_1(\cdot), T)$. Clearly, this predicate is a PPT one. If the response of $\mathcal{O}_{\text{dec}}$ is $\perp$, $\mathcal{B}_2$ sets $\mathbf{m}'_\eta$ to 0. Otherwise $\mathcal{B}_2$ sets $\mathbf{m}'_\eta$ to 1.

Case 1: $\mathcal{O}_{\text{dec}}(\mathsf{XVer}(\mathsf{H}_1(\cdot), T), \psi_\eta) = \perp$. This happens if and only if

$$\psi_\eta \in \boldsymbol{\psi}_{\text{enc}} \vee \mathsf{XVer}(\mathsf{H}_1(\mathsf{KDec}(\mathsf{sk}_{\mathsf{kem}}, \psi_\eta)), T) = 0.$$

In the first phase, $\mathcal{B}_2$ has not submitted any $\mathcal{O}_{\text{enc}}$ query yet and $\boldsymbol{\psi}_{\text{enc}}$ is empty. So $\psi_\eta \notin \boldsymbol{\psi}_{\text{enc}}$. In this case, $\mathcal{O}_{\text{dec}}(\mathsf{XVer}(\mathsf{H}_1(\cdot), T), \psi_\eta) = \perp$ if and only if

$$\mathsf{XVer}(\mathsf{H}_1(\mathsf{KDec}(\mathsf{sk}_{\mathsf{kem}}, \psi_\eta)), T) = 0.$$

Therefore $\mathcal{B}_2$ perfectly simulates the Dec oracle in $G_2(G_3)$ by setting $\mathbf{m}'_\eta \leftarrow 0$.

Case 2: $\mathcal{O}_{\text{dec}}(\mathsf{XVer}(\mathsf{H}_1(\cdot), T), \psi_\eta) \neq \perp$. This happens if and only if

$$\psi_\eta \notin \boldsymbol{\psi}_{\text{enc}} \wedge \mathsf{XVer}(\mathsf{H}_1(\mathsf{KDec}(\mathsf{sk}_{\mathsf{kem}}, \psi_\eta)), T) = 1.$$

For the same reason as case 1, the condition $\psi_\eta \notin \boldsymbol{\psi}_{\text{enc}}$ always holds. In this case, $\mathcal{O}_{\text{dec}}(\mathsf{XVer}(\mathsf{H}_1(\cdot), T), \psi_\eta) \neq \perp$ if and only if $\mathsf{XVer}(\mathsf{H}_1(\mathsf{KDec}(\mathsf{sk}_{\mathsf{kem}}, \psi_\eta)), T) = 1$. Therefore $\mathcal{B}_2$ perfectly simulates the Dec oracle in $G_2(G_3)$ by setting $\mathbf{m}'_\eta \leftarrow 1$.

In either case, $\mathcal{B}_2$ can perfectly simulate the Dec oracle for $\mathcal{A}_1$. At the end of this phase, $\mathcal{B}_2$ gets $\mathcal{A}_1$'s output (α, a_1). Then $\mathcal{B}_2$ calls $\mathbf{m} \leftarrow_\$ \mathcal{M}(\alpha)$ and simulates algorithm $\mathsf{SimCtGen}(\mathsf{pk})$.

- If $\mathbf{m}_{i,j} = 1$, $\mathcal{B}_2$ proceeds just like game $G_2(G_3)$, i.e., $(\psi_{i,j}, \gamma_{i,j}) \leftarrow_\$ \mathsf{KEnc}(\mathsf{pk}_{\mathsf{kem}})$ and set $K_{i,j} \leftarrow \mathsf{H}_1(\gamma_{i,j})$.
- If $\mathbf{m}_{i,j} = 0$, $\mathcal{B}_2$ submits an $\mathcal{O}_{\text{enc}}()$ query to its own oracle and gets the response (ψ, γ) (ψ is added into set $\boldsymbol{\psi}_{\text{enc}}$). Then $\mathcal{B}_2$ sets $(\psi_{i,j}, \gamma_{i,j}) \leftarrow (\psi, \gamma)$.
  If $b = 1$, (ψ, γ) is the output of $\mathsf{KEnc}(\mathsf{pk}_{\mathsf{kem}})$, $\mathcal{B}_2$ perfectly simulates $\mathsf{SimCtGen}(\mathsf{pk})$ to generate challenge ciphertexts $\mathbf{C}$ in G_2.
  If $b = 0$, (ψ, γ) is uniformly over $\Psi \times \Gamma$, $\mathcal{B}_2$ perfectly simulates $\mathsf{SimCtGen}(\mathsf{pk})$ to generate challenge ciphertexts $\mathbf{C}$ in G_3.

In the second phase, $\mathcal{B}_2$ calls $\mathcal{A}_2^{\mathsf{Dec}_{\notin \mathbf{C}}(\cdot)}(a_1, \mathbf{C})$ to get (I, a_2). Upon an decryption query $\mathsf{Dec}_{\notin \mathbf{C}}(C = (\psi_1, \cdots, \psi_\ell, T))$ submitted by $\mathcal{A}_2$, $\mathcal{B}_2$ responds almost in the same way as in the first phase, except that $\mathcal{B}_2$ has to deal with the case of $\exists \psi_\eta \in \boldsymbol{\psi}_{\text{enc}}$. This case does happen: even if $C = (\psi_1, \cdots, \psi_\ell, T) \notin \mathbf{C}$, it is still possible that $\exists \psi_\eta \in \{\psi_i\}_{i \in [\ell]}$ with $\psi_\eta \in \boldsymbol{\psi}_{\text{enc}}$. In this case, there is no chance for $\mathcal{B}_2$ to submit an $\mathcal{O}_{\text{dec}}(\mathsf{pred}, \psi_\eta)$ query for a useful response because the response will always be $\perp$. However, it does not matter. By the specification of $G_2(G_3)$, $\mathbf{m}'_\eta$ should be set to the output of $\mathsf{XVer}(K_{i,j}, T)$ which $\mathcal{B}_2$ can perfectly do.

Note that the execution of algorithm $\mathsf{SimOpen}$ in game $G_2(G_3)$ does not need all information about $\mathbf{R}$. Only those randomnesses with respect to $\mathbf{m}_{i,j} = 1$ are needed. Now that $\mathcal{B}_2$ does have $I, \mathbf{m}_I, \mathbf{C}, \mathbf{K}$ and part of $\mathbf{R}$ (for $\mathbf{m}_{i,j} = 1$), it can call $\mathsf{SimOpen}(I, \mathbf{m}_I, \mathbf{C}, \mathbf{R}, \mathbf{K})$ to get $\hat{\mathbf{R}}_I$.

In the third phase, $\mathcal{B}_2$ calls $\mathcal{A}_3^{\mathsf{Dec}_{\notin \mathbf{C}}(\cdot)}(a_2, \mathbf{m}_I, \hat{\mathbf{R}}_I)$ to get $out_{\mathcal{A}}$. The $\mathsf{Dec}_{\notin \mathbf{C}}$ query submitted by $\mathcal{A}$ in this phase is responded by $\mathcal{B}_2$ in the same way as in the second phase. Finally, $\mathcal{B}_2$ outputs $R(\mathbf{m}, I, out_{\mathcal{A}})$.

According to the above analysis, $\mathcal{B}_2$ perfectly simulates G_2 for $\mathcal{A}$ if $b = 1$ and perfectly simulates G_3 for $\mathcal{A}$ if $b = 0$. Moreover, for $\gamma \leftarrow_{\$} \Gamma$, $\mathsf{H}_1(\gamma)$ is Δ-close to uniform by Lemma 2 since H_1 is universal. Then

$$\Pr_{\gamma \leftarrow_{\$} \Gamma}[\mathsf{pred}(\gamma) = 1] = \Pr_{\gamma \leftarrow_{\$} \Gamma}[\mathsf{XVer}(\mathsf{H}_1(\gamma), T) = 1] \leq \epsilon_{\mathsf{XAC}}^{\mathrm{imp}}(\lambda) + \Delta.$$

By the definition of uncertainty, we have.

$$uncert_{\mathcal{B}_2}(\lambda) \leq \epsilon_{\mathsf{XAC}}^{\mathrm{imp}}(\lambda) + \Delta. \tag{6}$$

Thus (5) follows.

Game $G_3 - G_4$. G_4 is almost the same as G_3 except for one change in the $\mathsf{SimCtGen}$ algorithm. In the $\mathsf{SimCtGen}$ algorithm, if $\mathbf{m}_{i,j} = 0$,

- in G_3, $K_{i,j} \leftarrow \mathsf{H}_1(\gamma_{i,j})$ for $\gamma_{i,j} \leftarrow_{\$} \Gamma$;
- in G_4, $K_{i,j}$ is uniformly selected from $\mathcal{XK}$.

Since H_1 is universal, by Lemma 2 and a union bound, we have that

$$|\Pr_3[G \Rightarrow 1] - \Pr_4[G \Rightarrow 1]| \leq (n\ell) \cdot \Delta. \tag{7}$$

Game $G_4 - G_5$. G_5 is almost the same as G_4 except for one change in the Dec oracle. More precisely, to reply a $\mathsf{Dec}_{\notin \mathbf{C}}(C = (\psi_1, \cdots, \psi_\ell, T))$ query such that $\exists (i,j) \in [n] \times [\ell], \eta \in [\ell]$ s.t. $\mathbf{m}_{i,j} = 0 \wedge \psi_\eta = \psi_{i,j}$,

- in G_4, we set $\mathbf{m}'_\eta \leftarrow \mathsf{XVer}(K_{i,j}, T)$;
- in G_5, we set $\mathbf{m}'_\eta \leftarrow 0$ directly.

Suppose $\psi_\eta = \psi_{i,j} \in \mathbf{C}_i = (\psi_{i,1}, \cdots, \psi_{i,\ell}, T_i)$ where $T_i = \mathsf{XAuth}(K_{i,1}, \cdots, K_{i,\ell+1})$. There are two cases according to whether $T = T_i$.

Case 1: $T = T_i$. In this case, since $C \notin \mathbf{C}$, we have that $(\psi_1, \cdots, \psi_\ell) \neq (\psi_{i,1}, \cdots, \psi_{i,\ell})$. Note that $K_i^y = \mathsf{H}_2(\psi_{i,1}, \cdots, \psi_{i,\ell})$ and $K^{y\prime} = \mathsf{H}_2(\psi_1, \cdots, \psi_\ell)$. If $K_i^y = K^{y\prime}$, a collision for H_2 occurs, both G_4 and G_5 abort. Otherwise, we must have $K^{y\prime} \neq K_i^y$, hence $K'_{\ell+1} = (K^x, K^{y\prime}) \neq (K^x, K_i^y) = K_{i,\ell+1}$. Since XAC is semi-unique and $\mathsf{XVer}(K_{i,\ell+1}, T) = 1$, it holds that $\mathsf{XVer}(K'_{\ell+1}, T) \neq 1$ which implies that $\mathbf{m}'_\eta = 0$. In this case, the responses of $\mathsf{Dec}_{\notin \mathbf{C}}$ make no difference in G_4 and G_5.

Case 2: $T \neq T_i$. Note that all the information about $K_{i,j}$ is leaked to $\mathcal{A}$ only through T_i in game G_4. Thus, the probability that $\mathsf{XVer}(K_{i,j}, T) = 1$ for $T \neq T_i$ will be no more than $\epsilon_{\mathsf{XAC}}^{\mathrm{sub}}(\lambda)$.

By a union bound, we have that

$$|\Pr_4[G \Rightarrow 1] - \Pr_5[G \Rightarrow 1]| \leq \ell \cdot Q_{\mathrm{dec}} \cdot \epsilon_{\mathsf{XAC}}^{\mathrm{sub}}(\lambda). \tag{8}$$

Game $G_5 - G_6$. G_6 is almost the same as G_5 except for one change in the Dec oracle. More precisely, for a $\mathsf{Dec}(C = (\psi_1, \cdots, \psi_\ell, T))$ query such that $\exists (i,j) \in [n] \times [\ell]$ s.t. $\mathbf{m}_{i,j} = 0 \wedge \psi_\eta = \psi_{i,j}$ for any $\eta \in [\ell]$,

- in G_5, we set $\mathbf{m}'_\eta \leftarrow 0$ directly;
- in G_6, we proceed exactly the same as the decryption algorithm, i.e., setting $\mathbf{m}'_\eta \leftarrow \mathsf{XVer}(\mathsf{H}_1(\gamma'_\eta), T)$, where $\gamma'_\eta = \mathsf{KDec}(\mathsf{sk}_{\mathsf{kem}}, \psi_\eta)$.

We will reduce the indistinguishability between game G_5 and G_6 to the RER security of KEM. More precisely, we can build a PPT adversary $\mathcal{B}_3$ with $\mathbf{T}(\mathcal{B}_3) \approx \mathbf{T}(\mathcal{A})$ and with uncertainty $uncert_{\mathcal{B}_3}(\lambda) \leq \epsilon^{\mathrm{imp}}_{\mathsf{XAC}}(\lambda) + \varDelta$ such that

$$|\Pr_5[G \Rightarrow 1] - \Pr_6[G \Rightarrow 1]| \leq \mathsf{Adv}^{\mathrm{rer}}_{\mathsf{KEM},\mathcal{B}_3}(\lambda). \tag{9}$$

On input $\mathsf{pk}_{\mathsf{kem}}$, $\mathcal{B}_3$ selects $\mathsf{H}_1, \mathsf{H}_2$ and K^x itself and embeds $\mathsf{pk}_{\mathsf{kem}}$ in $\mathsf{pk} = (\mathsf{pk}_{\mathsf{kem}}, \mathsf{H}_1, \mathsf{H}_2, K^x)$. In the first phase, $\mathcal{B}_3$ calls $\mathcal{A}_1^{\mathsf{Dec}(\cdot)}(\mathsf{pk})$. To respond the decryption query $\mathsf{Dec}(C = (\psi_1, \cdots, \psi_\ell, T))$ submitted by $\mathcal{A}$, $\mathcal{B}_3$ simulates Dec until it needs to call $\mathsf{KDec}(\mathsf{sk}_{\mathsf{kem}}, \psi_\eta)$ to decapsulate ψ_η. Since $\mathcal{B}_3$ does not hold $\mathsf{sk}_{\mathsf{kem}}$ relative to $\mathsf{pk}_{\mathsf{kem}}$, $\mathcal{B}_3$ is not able to invoke KDec itself. Then $\mathcal{B}_3$ submits a $\mathcal{O}_{\mathrm{cha}}(\mathsf{pred}, \psi)$ query to its own oracle $\mathcal{O}_{\mathrm{cha}}$ where $\mathsf{pred}(\cdot) := \mathsf{XVer}(\mathsf{H}_1(\cdot), T)$ and $\psi = \psi_\eta$. Clearly, this predicate is a PPT one. Since $\boldsymbol{\psi}_{\mathrm{ran}}$ is empty set in this phase, the condition $\psi \notin \boldsymbol{\psi}_{\mathrm{ran}}$ will always hold and $\mathcal{B}_3$ will get a bit $\beta = \mathsf{pred}(\mathsf{KDec}(\mathsf{sk}_{\mathsf{kem}}, \psi)) = \mathsf{XVer}(\mathsf{H}_1(\mathsf{KDec}(\mathsf{sk}_{\mathsf{kem}}, \psi_\eta)), T)$ in return. Then $\mathcal{B}_3$ sets $\mathbf{m}'_\eta \leftarrow \beta$ and perfectly simulates Dec for $\mathcal{A}$ in this phase.

At the end of this phase, $\mathcal{B}_3$ gets $\mathcal{A}$'s output (α, a_1). Then $\mathcal{B}_3$ calls $\mathbf{m} \leftarrow_{\$} \mathcal{M}(\alpha)$ and then simulates algorithm $\mathsf{SimCtGen}(\mathsf{pk})$ as follows. $\mathcal{B}_3$ first outputs $1^{n\ell}$ and get $\boldsymbol{\psi}_{\mathrm{ran}} = \{\psi^{\mathrm{ran}}_1, \cdots, \psi^{\mathrm{ran}}_{n\ell}\}$ which are $n\ell$ random encapsulations. During the generation of the challenge ciphertexts, $\mathcal{B}_3$ sets $(\psi_{i,j}, K_{i,j})$ according to $\mathbf{m}$.

- If $\mathbf{m}_{i,j} = 1$, $\mathcal{B}_3$ sets $(\psi_{i,j}, \gamma_{i,j}) \leftarrow_{\$} \mathsf{KEnc}(\mathsf{pk}_{\mathsf{kem}})$ and sets $K_{i,j} \leftarrow \mathsf{H}_1(\gamma_{i,j})$.
- If $\mathbf{m}_{i,j} = 0$, $\mathcal{B}_3$ sets $\psi_{i,j} \leftarrow \psi^{\mathrm{ran}}_{(i-1)\ell+j}$ and $K_{i,j} \leftarrow_{\$} \mathcal{XK}$. Since $(i,j) \in [n] \times [\ell]$, the subscript $(i-1)\ell + j \in \{1, \cdots, n\ell\}$ is well defined.

Then $\mathcal{B}_3$ proceeds just like algorithm $\mathsf{SimCtGen}(\mathsf{pk})$ in game $G_5(G_6)$.

In the second phase, $\mathcal{B}_3$ calls $\mathcal{A}_2^{\mathsf{Dec}_{\notin \mathbf{C}}(\cdot)}(a_1, \mathbf{C})$ to get (I, a_2). To respond the decryption query $\mathsf{Dec}_{\notin \mathbf{C}}(C = (\psi_1, \cdots, \psi_\ell, T))$ submitted by $\mathcal{A}$, $\mathcal{B}_3$ proceeds just like game $G_5(G_6)$. When a decapsulation of ψ_η is needed, $\mathcal{B}_3$ submits a $\mathcal{O}_{\mathrm{cha}}(\mathsf{pred}, \psi_\eta)$ query to its own oracle $\mathcal{O}_{\mathrm{cha}}$ where $\mathsf{pred}(\cdot) := \mathsf{XVer}(\mathsf{H}_1(\cdot), T)$. After that, $\mathcal{B}_3$ will get a bit β in return and $\mathcal{B}_3$ sets $\mathbf{m}'_\eta \leftarrow \beta$. Note that

- In case of $\psi_\eta \notin \boldsymbol{\psi}_{\mathrm{ran}}$, $\mathbf{m}'_\eta = \mathsf{XVer}(\mathsf{H}_1(\mathsf{KDec}(\mathsf{sk}_{\mathsf{kem}}, \psi_\eta)), T)$, which is exactly how $\mathbf{m}'_\eta$ is computed in both game G_5 and G_6.
- In case of $\psi_\eta \in \boldsymbol{\psi}_{\mathrm{ran}}$, there must exist $(i,j) \in [n] \times [\ell]$ s.t. $\mathbf{m}_{i,j} = 0 \wedge \psi_\eta = \psi_{i,j}$. Thus $\mathbf{m}'_\eta = \mathsf{XVer}(\mathsf{H}_1(\mathsf{KDec}(\mathsf{sk}_{\mathsf{kem}}, \psi_\eta)), T)$ if $b = 1$ and $\mathbf{m}'_\eta = 0$ if $b = 0$. The former case is exactly how $\mathbf{m}'_\eta$ is computed in game G_6 and the latter case is exactly how $\mathbf{m}'_\eta$ is computed in game G_5.

As a result, $\mathcal{B}_3$ perfectly simulates $\mathsf{Dec}_{\notin \mathbf{C}}$ in the second phase of game G_5 for $\mathcal{A}$ if $b = 0$ and perfectly simulates $\mathsf{Dec}_{\notin \mathbf{C}}$ in the second phase of game G_6 for $\mathcal{A}$ if $b = 1$. After $\mathcal{B}_3$ gets (I, a_2), $\mathcal{B}_3$ is able to call $\mathsf{SimOpen}(I, \mathbf{m}_I, \mathbf{C}, \mathbf{R}, \mathbf{K})$ to get $\hat{\mathbf{R}}_I$ for the similar reason as in the proof of $G_2 - G_3$.

In the third phase, $\mathcal{B}_3$ calls $\mathcal{A}_3^{\mathsf{Dec}_{\notin \mathbf{C}}(\cdot)}(a_2, \mathbf{m}_I, \hat{\mathbf{R}}_I)$ to get $out_{\mathcal{A}}$. The $\mathsf{Dec}_{\notin \mathbf{C}}$ query submitted by $\mathcal{A}$ in this phase is responded using the same way as in the second phase. Finally, $\mathcal{B}_3$ outputs $R(\mathbf{m}, I, out_{\mathcal{A}})$.

Thus $\mathcal{B}_3$ perfectly simulates G_6 for $\mathcal{A}$ if $b = 1$ and perfectly simulates G_5 for $\mathcal{A}$ if $b = 0$. Similar to (6), $uncert_{\mathcal{B}_3}(\lambda) \leq \epsilon^{\mathrm{imp}}_{\mathsf{XAC}}(\lambda) + \varDelta$. Thus (9) follows.

Game $G_6 - G_7$. G_7 is almost the same as G_6 except for one change in the $\mathsf{SimOpen}$ algorithm. More precisely,

- in G_6, $\hat{r}^K_{i,j}$ is the output of $\mathsf{Sample}^{-1}_{\mathcal{XK}}(\hat{K}_{i,j})$ where $\hat{K}_{i,j} \leftarrow_{\$} \mathsf{ReSamp}(T_i, j)$;
- in G_7, $\hat{r}^K_{i,j}$ is the output of $\mathsf{Sample}^{-1}_{\mathcal{XK}}(K_{i,j})$ for the original $K_{i,j}$ generated in algorithm $\mathsf{SimCtGen}$.

In game G_6 and G_7, before the invocation of algorithm $\mathsf{SimOpen}$, only T_i leaks information about $K_{i,j}$ to $\mathcal{A}$ when $\mathbf{m}_{i,j} = 0$. Since XAC is $\delta(\lambda)$-strong, the statistical distance between the resampled $\hat{K}_{i,j} \leftarrow_\$ \mathsf{ReSamp}(T_i, j)$ and the original $K_{i,j}$ is at most $\delta(\lambda)$. By a union bound, we have that

$$|\Pr_6[G \Rightarrow 1] - \Pr_7[G \Rightarrow 1]| \leq (n\ell) \cdot \delta(\lambda). \tag{10}$$

Game $G_7 - G_8$. G_8 is almost the same as G_7 except for the dropping of the collision check added in G_1. Similar to the proof of $G_0 - G_1$, we can show that

$$|\Pr_7[G \Rightarrow 1] - \Pr_8[G \Rightarrow 1]| \leq \mathsf{Adv}^{\mathrm{cr}}_{\mathcal{H},\mathcal{B}_1}(\lambda). \tag{11}$$

Game $G_8 - G_9$. G_9 is almost the same as G_8 except for one change in $\mathsf{SimOpen}$. More precisely,

- in G_8, the opened randomness is a "reverse sampled" randomness, i.e., $\hat{r}^K_{i,j} \leftarrow_\$ \mathsf{Sample}^{-1}_{\mathcal{XK}}(K_{i,j})$ and $\hat{r}^\psi_{i,j} \leftarrow_\$ \mathsf{Sample}^{-1}_{\Psi}(\psi_{i,j})$;
- in G_9, the opened randomness $(\hat{r}^K_{i,j}, \hat{r}^\psi_{i,j})$ is changed to be the original randomness used to sample $K_{i,j}$ and $\psi_{i,j}$, i.e., $(\hat{r}^K_{i,j}, \hat{r}^\psi_{i,j}) \leftarrow (r^K_{i,j}, r^\psi_{i,j})$.

This change is conceptual since Ψ and $\mathcal{XK}$ are ESE domains. Thus

$$\Pr_8[G \Rightarrow 1] = \Pr_9[G \Rightarrow 1]. \tag{12}$$

Game G_9. Game G_9 is exactly the real experiment $\mathsf{Exp}^{\text{so-cca-real}}_{\mathsf{PKE},\mathcal{A},n,\mathcal{M},R}(\lambda)$. Thus

$$\Pr_9[G \Rightarrow 1] = \Pr\left[\mathsf{Exp}^{\text{so-cca-real}}_{\mathsf{PKE},\mathcal{A},n,\mathcal{M},R}(\lambda) = 1\right]. \tag{13}$$

Finally, Theorem 1 follows from (2, 3, 4, 5, 7, 8, 9, 10, 11, 12) and (13). ■

5 Instantiations

We give two instantiations of KEM with mPR-CCCA security and RER security.

5.1 KEM from MDDH

We present a KEM which is extracted from the multi-challenge IND-CCA secure PKE proposed by Gay *et al.* in [GHKW16]. The KEM $\mathsf{KEM}_{\mathsf{mddh}} = (\mathsf{KGen}, \mathsf{KEnc}, \mathsf{KDec})$ is shown in Figure 7.

Suppose $\mathcal{G} = (\mathbb{G}, q, P) \leftarrow_\$ \mathsf{GGen}(1^\lambda)$ and $\mathcal{H}$ is a hash generator outputting functions $\mathsf{H} : \mathbb{G}^k \rightarrow \{0,1\}^\lambda$. For a vector $\mathbf{y} \in \mathbb{Z}_q^{3k}$, we use $\overline{\mathbf{y}} \in \mathbb{Z}_q^k$ to denote the upper k components and $\underline{\mathbf{y}} \in \mathbb{Z}_q^{2k}$ to denote the lower $2k$ components.

Perfectly correctness of $\mathsf{KEM}_{\mathsf{mddh}}$ is straightforward. By Theorem 2 and 3, we will prove that it is tightly mPR-CCCA secure and tightly RER secure.

Theorem 2. *The KEM $\mathsf{KEM}_{\mathsf{mddh}}$ in Figure 7 is mPR-CCCA secure if $\mathcal{U}_k$-MDDH assumption holds and $\mathcal{H}$ outputs collision-resistant hash function. Specifically, for each PPT adversary $\mathcal{A}$ with negligible uncertainty $uncert_{\mathcal{A}}(\lambda)$, there exist two PPT adversaries $\mathcal{B}_1, \mathcal{B}_2$ with $\mathbf{T}(\mathcal{B}_1) \approx \mathbf{T}(\mathcal{B}_2) \leq \mathbf{T}(\mathcal{A}) + (Q_{\mathrm{enc}} + Q_{\mathrm{dec}}) \cdot \mathsf{poly}(\lambda)$ such that the advantage*

$$\begin{aligned}\mathsf{Adv}^{\text{mpr-ccca}}_{\mathsf{KEM}_{\mathsf{mddh}},\mathcal{A}}(\lambda) \leq{} & (8\lambda+6)\mathsf{Adv}^{\text{mddh}}_{\mathcal{U}_k,\mathsf{GGen},\mathcal{B}_1}(\lambda) + 2\mathsf{Adv}^{\text{cr}}_{\mathcal{H},\mathcal{B}_2}(\lambda) \\ & + (8\lambda+4)Q_{dec} \cdot uncert_{\mathcal{A}}(\lambda) + 2^{-\Omega(\lambda)},\end{aligned}$$

where $Q_{\mathrm{enc}}(Q_{\mathrm{dec}})$ is the total number of $\mathcal{O}_{\mathrm{enc}}(\mathcal{O}_{\mathrm{dec}})$ queries made by $\mathcal{A}$ and $\mathsf{poly}(\lambda)$ is a polynomial independent of $\mathbf{T}(\mathcal{A})$.

```
$\mathsf{KGen}(1^\lambda):$
    $\mathbf{M} \leftarrow_\$ \mathcal{U}_{3k,k}, \mathsf{H} \leftarrow_\$ \mathcal{H}(1^\lambda).$
    $\mathbf{k}_{1,0}, \cdots, \mathbf{k}_{\lambda,1} \leftarrow_\$ \mathbb{Z}_q^{3k}$
    $\mathsf{pk}_{\mathsf{kem}} \leftarrow \begin{pmatrix} \mathcal{G}, \mathsf{H}, [\mathbf{M}] \\ ([\mathbf{M}^\top \mathbf{k}_{j,\beta}])_{1\le j\le\lambda}^{0\le\beta\le 1} \end{pmatrix}$
    $\mathsf{sk}_{\mathsf{kem}} \leftarrow (\mathbf{k}_{j,\beta})_{1\le j\le\lambda, 0\le\beta\le 1}$
Return $(\mathsf{pk}_{\mathsf{kem}}, \mathsf{sk}_{\mathsf{kem}})$

$\mathsf{KEnc}(\mathsf{pk}_{\mathsf{kem}}):$
    $\mathbf{r} \leftarrow_\$ \mathbb{Z}_q^k, [\mathbf{y}] \leftarrow [\mathbf{M}]\mathbf{r}$
    $\tau \leftarrow \mathsf{H}([\overline{\mathbf{y}}])$
    $\gamma \leftarrow \mathbf{r}^\top \cdot \sum_{j=1}^{\lambda} [\mathbf{M}^\top \mathbf{k}_{j,\tau_j}]$
Return $(\psi \leftarrow [\mathbf{y}], \gamma)$
$//\Psi = \mathbb{G}^{3k}, \Gamma = \mathbb{G}$

$\mathsf{KDec}(\mathsf{sk}_{\mathsf{kem}}, \psi):$
    $\psi = [\mathbf{y}]$
    $\tau \leftarrow \mathsf{H}([\overline{\mathbf{y}}])$
    $\mathbf{k}_\tau \leftarrow \sum_{j=1}^{\lambda} \mathbf{k}_{j,\tau_j}$
    $\gamma \leftarrow [\mathbf{y}^\top] \cdot \mathbf{k}_\tau$
Return $\gamma$
```

Fig. 7. The KEM $\mathsf{KEM}_{\mathsf{mddh}} = (\mathsf{KGen}, \mathsf{KEnc}, \mathsf{KDec})$ extracted from [GHKW16].

Theorem 3. *The KEM $\mathsf{KEM}_{\mathsf{mddh}}$ in Figure 7 is RER secure if $\mathsf{KEM}_{\mathsf{mddh}}$ is mPR-CCCA secure and the $\mathcal{U}_k$-MDDH assumption holds. Specifically, for each PPT adversary $\mathcal{A}$ with negligible uncertainty $uncert_{\mathcal{A}}(\lambda)$, there exist two PPT adversaries $\mathcal{B}_1, \mathcal{B}_2$ with $\mathbf{T}(\mathcal{B}_1) \approx \mathbf{T}(\mathcal{B}_2) \le \mathbf{T}(\mathcal{A}) + Q_{\mathrm{cha}} \cdot \mathsf{poly}(\lambda)$ and $uncert_{\mathcal{B}_1}(\lambda) = uncert_{\mathcal{A}}(\lambda)$ such that*

$$\mathsf{Adv}^{\mathrm{rer}}_{\mathsf{KEM}_{\mathsf{mddh}},\mathcal{A}}(\lambda) \le 2\mathsf{Adv}^{\mathrm{mpr\text{-}ccca}}_{\mathsf{KEM},\mathcal{B}_1}(\lambda) + 2\mathsf{Adv}^{\mathrm{mddh}}_{\mathcal{U}_k,\mathsf{GGen},\mathcal{B}_2}(\lambda) + 2Q_{\mathrm{cha}} \cdot uncert_{\mathcal{A}}(\lambda) + 2^{-\Omega(\lambda)},$$

where Q_{cha} is the total number of $\mathcal{O}_{\mathrm{cha}}$ queries made by $\mathcal{A}$ and $\mathsf{poly}(\lambda)$ is a polynomial independent of $\mathbf{T}(\mathcal{A})$.

The public key of $\mathsf{KEM}_{\mathsf{mddh}}$ is not compact, so we put the proof of these two theorems in Appendix C and D.

5.2 KEM from Qualified Proof System with Compact Public Key

First we recall the definition of a *proof system* described in [GHK17].

Definition 7 (Proof System). *Let $\mathcal{L} = \{\mathcal{L}_{\mathsf{pars}}\}$ be a family of languages indexed by public parameters pars, with $\mathcal{L}_{\mathsf{pars}} \subseteq \mathcal{X}_{\mathsf{pars}}$ and an efficiently computable witness relation $\mathcal{R}$. A* proof system *$\mathsf{PS} = (\mathsf{PGen}, \mathsf{PPrv}, \mathsf{PVer}, \mathsf{PSim})$ for $\mathcal{L}$ consists of a tuple of PPT algorithms.*

- $\mathsf{PGen}(\mathsf{pars})$*. It outputs a public key ppk and a secret key psk.*
- $\mathsf{PPrv}(\mathsf{ppk}, x, w)$*. On input a statement $x \in \mathcal{L}$ and a witness w with $\mathcal{R}(x, w) = 1$, it deterministically outputs a proof $\Pi \in \mathbf{\Pi}$ and a key $K \in \mathcal{K}$.*
- $\mathsf{PVer}(\mathsf{ppk}, \mathsf{psk}, x, \Pi)$*. On input $\mathsf{ppk}, \mathsf{psk}, x \in \mathcal{X}$ and Π, it deterministically outputs $b \in \{0, 1\}$ together with a key $K \in \mathcal{K}$ if $b = 1$ or $\perp$ if $b = 0$.*
- $\mathsf{PSim}(\mathsf{ppk}, \mathsf{psk}, x)$*. Given $\mathsf{ppk}, \mathsf{psk}, x \in \mathcal{X}$, it deterministically outputs a proof Π and a key $K \in \mathcal{K}$.*

Next we recall the definition of a qualified proof system.

Definition 8 (Qualified Proof System [GHK17]). *Let $\mathsf{PS} = (\mathsf{PGen}, \mathsf{PPrv}, \mathsf{PVer}, \mathsf{PSim})$ be a proof system for a family of languages $\mathcal{L} = \mathcal{L}_{\mathsf{pars}}$. Let $\mathcal{L}^{\mathrm{snd}} = \{\mathcal{L}^{\mathrm{snd}}_{\mathsf{pars}}\}$ be a family of languages, such that $\mathcal{L}_{\mathsf{pars}} \subseteq \mathcal{L}^{\mathrm{snd}}_{\mathsf{pars}}$. We say that PS is $\mathcal{L}^{\mathrm{snd}}$-*qualified*, if the following properties hold.*

- **Completeness:** *For all possible public parameters pars, for all statements $x \in \mathcal{L}$ and all witnesses w such that $\mathcal{R}(x, w) = 1$, $\Pr[\mathsf{PVer}(\mathsf{ppk}, \mathsf{psk}, x, \Pi)] = 1$, where $(\mathsf{ppk}, \mathsf{psk}) \leftarrow_\$ \mathsf{PGen}(\mathsf{pars})$ and $(\Pi, K) \leftarrow_\$ \mathsf{PPrv}(\mathsf{ppk}, x, w)$.*

- **Perfect zero-knowledge:** *For all possible public parameters* pars*, all key pairs* $(\mathsf{ppk}, \mathsf{psk})$ *in the output range of* $\mathsf{PGen}(\mathsf{pars})$*, all statements* $x \in \mathcal{L}$ *and all witnesses* w *with* $\mathcal{R}(x, w) = 1$*, we have* $\mathsf{PPrv}(\mathsf{ppk}, x, w) = \mathsf{PSim}(\mathsf{ppk}, \mathsf{psk}, x)$*.*
- **Unique of the proofs:** *For all possible public parameters* pars*, all key pairs* $(\mathsf{ppk}, \mathsf{psk})$ *in the output range of* $\mathsf{PGen}(\mathsf{pars})$ *and all statements* $x \in \mathcal{X}$*, there exists at most one* Π^* *such that* $\mathsf{PVer}(\mathsf{ppk}, \mathsf{psk}, x, \Pi^*) = 1$*.*
- **Constrained** $\mathcal{L}^{\text{snd}}$**-Soundness:** *For any stateful PPT adversary* $\mathcal{A}$*, consider the soundness experiment in Figure 8 (where* PSim *and* PVer *are implicitly assumed to have access to* ppk*).*

```
$\text{Exp}^{\text{csnd}}_{\mathcal{L}^{\text{snd}},\mathsf{PS},\mathcal{A}}(\lambda)$:
win = 0
$(\mathsf{ppk}, \mathsf{psk}) \leftarrow_{\$} \mathsf{PGen}(\mathsf{pars})$
$\mathcal{A}^{\mathcal{O}_{\text{sim}}(),\mathcal{O}_{\text{ver}}(\cdot,\cdot,\cdot)}(\mathsf{ppk})$

$\mathcal{O}_{\text{sim}}()$:
$x \leftarrow_{\$} \mathcal{L}^{\text{snd}} \setminus \mathcal{L}$
$(\Pi, K) \leftarrow \mathsf{PSim}(\mathsf{psk}, x)$
Return $(x, \Pi, K)$
```

```
$\mathcal{O}_{\text{ver}}(x, \Pi, \mathsf{pred})$:
$(v, K) \leftarrow \mathsf{PVer}(\mathsf{psk}, x, \Pi)$
If $v = 1 \wedge \mathsf{pred}(K) = 1$:
    If $x \in \mathcal{L}$:
        Return $K$
    Else:
        win = $\begin{cases} 0 & \text{If } x \in \mathcal{L}^{\text{snd}} \\ 1 & \text{Otherwise} \end{cases}$
        Abort game
Return $\perp$
```

Fig. 8. Experiment used in the definition of constrained $\mathcal{L}^{\text{snd}}$-soundness of PS.

Let Q_{ver} *be the total number of* $\mathcal{O}_{ver}$ *queries, which is independent of the environment without loss of generality. Let* $\mathsf{pred}_i : \mathcal{K} \cup \{\perp\} \rightarrow \{0, 1\}$ *be the predicate submitted by* $\mathcal{A}$ *in the i-th query, where* $\mathsf{pred}_i(\perp) = 0$ *for all* i*. The* uncertainty *of* $\mathcal{A}$ *is defined as*

$$uncert_{\mathcal{A}}(\lambda) := \tfrac{1}{Q_{ver}} \textstyle\sum_{i=1}^{Q_{ver}} \Pr_{K \leftarrow_{\$} \mathcal{K}}[\mathsf{pred}_i(K) = 1].$$

We say constrained $\mathcal{L}^{\text{snd}}$-soundness *holds for* PS *if for each PPT adversary* $\mathcal{A}$ *with negligible uncertainty,* $\mathsf{Adv}^{\text{csnd}}_{\mathcal{L}^{\text{snd}},\mathsf{PS},\mathcal{A}}(\lambda)$ *is negligible, where*

$$\mathsf{Adv}^{\text{csnd}}_{\mathcal{L}^{\text{snd}},\mathsf{PS},\mathcal{A}}(\lambda) := \Pr[\text{win} = 1 \text{ in } \text{Exp}^{\text{csnd}}_{\mathcal{L}^{\text{snd}},\mathsf{PS},\mathcal{A}}(\lambda)]$$

In Appendix E, we review the definition for $\mathcal{L}^{\text{snd}}$-indistinguishability of two proof systems and the definition for $\widetilde{\mathcal{L}^{\text{snd}}}$-extensibility of a proof system. Here we define a new property for qualified proof system, which stresses that the simulated proof Π for a random $x \in \mathcal{L}^{\text{snd}} \setminus \mathcal{L}$ is pseudorandom when providing verification oracle for only $x \in \mathcal{L}$.

Definition 9 (Pseudorandom Simulated Proof of Qualified Proof System). *Let* $\mathsf{PS} = (\mathsf{PGen}, \mathsf{PPrv}, \mathsf{PVer}, \mathsf{PSim})$ *be a* $\mathcal{L}^{\text{snd}}$*-qualified proof system for a family of languages* $\mathcal{L}$*. Let* $\mathcal{A}$ *be a stateful adversary and* $b \in \{0, 1\}$ *be a bit. Define the following experiment* $\mathsf{Exp}^{\text{pr-proof-}b}_{\mathsf{PS},\mathcal{A}}(\lambda)$ *in Figure 9. We say* PS *has* pseudorandom simulated proof *if for each PPT adversary* $\mathcal{A}$*, the advantage*

$$\mathsf{Adv}^{\text{pr-proof}}_{\mathsf{PS},\mathcal{A}}(\lambda) := \left|\Pr\left[\mathsf{Exp}^{\text{pr-proof-0}}_{\mathsf{PS},\mathcal{A}}(\lambda) = 1\right] - \Pr\left[\mathsf{Exp}^{\text{pr-proof-1}}_{\mathsf{PS},\mathcal{A}}(\lambda) = 1\right]\right| \text{ } \textit{is negl.}$$

```
$\mathrm{Exp}^{\text{pr-proof-}b}_{\mathsf{PS},\mathcal{A}}(\lambda)$: //$b \in \{0,1\}$
    $(\mathsf{ppk}, \mathsf{psk}) \leftarrow_\$ \mathsf{PGen}(\mathsf{pars})$
    $b' \leftarrow_\$ \mathcal{A}^{\mathcal{O}_{\mathrm{sim}}(),\mathcal{O}_{\mathrm{ver}}(\cdot,\cdot)}(\mathsf{ppk})$
    Return $b'$
```

```
$\mathcal{O}_{\mathrm{sim}}()$:
    $x \leftarrow_\$ \mathcal{L}^{\mathrm{snd}} \setminus \mathcal{L}$
    $\Pi_0 \leftarrow_\$ \mathbf{\Pi}$
    $(\Pi_1, K) \leftarrow \mathsf{PSim}(\mathsf{psk}, x)$
    Return $(x, \Pi_b)$
```

```
$\mathcal{O}_{\mathrm{ver}}(x, \Pi)$:
    $(v, K) \leftarrow \mathsf{PVer}(\mathsf{psk}, x, \Pi)$
    If $x \notin \mathcal{L} \lor v = 0$:
        Return $\perp$
    Return $K$
```

Fig. 9. Experiment used in the definition of pseudorandom simulated proof of PS.

The Qualified Proof System in [GHK17]. First we explain how the public parameters pars are sampled. Fix some $k \in \mathbb{N}$, invoke $\mathcal{G} \leftarrow_\$ \mathsf{GGen}(1^\lambda)$ where $\mathcal{G} = (\mathbb{G}, q, P)$. Let $\mathcal{D}_{2k,k}$ be a fixed matrix distribution, we sample $\mathbf{A} \leftarrow_\$ \mathcal{D}_{2k,k}$ and $\mathbf{A}_0 \leftarrow_\$ \mathcal{U}_{2k,k}$ where $\overline{\mathbf{A}}$ and $\overline{\mathbf{A}}_0$ are both full rank. Additionally select $\mathbf{A}_1 \in \mathbb{Z}_q^{2k\times k}$ according to $\mathcal{U}_{2k,k}$ with the restriction $\overline{\mathbf{A}}_0 = \overline{\mathbf{A}}_1$. Let $\mathcal{H}_0$ and $\mathcal{H}_1$ be universal hash function generators returning functions $\mathsf{h}_0 : \mathbb{G}^{k^2+1} \to \mathbb{Z}_q^{k\times k}$ and $\mathsf{h}_1 : \mathbb{G}^{k+1} \to \mathbb{Z}_q^k$ respectively. Let $\mathsf{h}_0 \leftarrow_\$ \mathcal{H}_0$ and $\mathsf{h}_1 \leftarrow_\$ \mathcal{H}_1$. Let $\mathsf{pars} \leftarrow (k, \mathcal{G}, [\mathbf{A}], [\mathbf{A}_0], [\mathbf{A}_1], \mathsf{h}_0, \mathsf{h}_1)$ be the public parameters and we assume pars is an implicit input of all algorithms. The languages are defined as $\mathcal{L} := \mathsf{span}([\mathbf{A}])$, $\mathcal{L}^{\mathrm{snd}} := \mathsf{span}([\mathbf{A}]) \cup \mathsf{span}([\mathbf{A}_0])$ and $\widetilde{\mathcal{L}^{\mathrm{snd}}} := \mathsf{span}([\mathbf{A}]) \cup \mathsf{span}([\mathbf{A}_0]) \cup \mathsf{span}([\mathbf{A}_1])$.

The construction[7] of $\mathcal{L}^{\mathrm{snd}}$-qualified proof system $\mathsf{PS} = (\mathsf{PGen}, \mathsf{PPrv}, \mathsf{PVer}, \mathsf{PSim})$ in [GHK17] is shown in Figure 10.

```
$\mathsf{PGen}(\mathsf{pars})$:
    $\mathbf{K_x} \leftarrow_\$ \mathbb{Z}_q^{(k^2+1)\times 2k}$
    $\mathbf{K_y} \leftarrow_\$ \mathbb{Z}_q^{(k+1)\times 2k}$
    $[\mathbf{P_x}] \leftarrow \mathbf{K_x}[\mathbf{A}] \in \mathbb{G}^{(k^2+1)\times k}$
    $[\mathbf{P_y}] \leftarrow \mathbf{K_y}[\mathbf{A}] \in \mathbb{G}^{(k+1)\times k}$
    $\mathsf{ppk} \leftarrow ([\mathbf{P_x}], [\mathbf{P_y}])$
    $\mathsf{psk} \leftarrow (\mathbf{K_x}, \mathbf{K_y})$
Return $(\mathsf{ppk}, \mathsf{psk})$
```

```
$\mathsf{PSim}(\mathsf{ppk}, \mathsf{psk}, [\mathbf{c}])$:
    $\mathbf{X} \leftarrow \mathsf{h}_0(\mathbf{K_x}[\mathbf{c}])$
    $\mathbf{y} \leftarrow \mathsf{h}_1(\mathbf{K_y}[\mathbf{c}])$
    $[\pi] \leftarrow [\overline{\mathbf{A}_0}] \cdot \mathbf{X} + [\overline{\mathbf{c}}] \cdot \mathbf{y}^\top$
    $[\mathbf{K}] \leftarrow [\underline{\mathbf{A}_0}] \cdot \mathbf{X} + [\underline{\mathbf{c}}] \cdot \mathbf{y}^\top$
    $[\kappa] \leftarrow \mathsf{trace}([\mathbf{K}])$
Return $([\pi], [\kappa])$
```

```
$\mathsf{PPrv}(\mathsf{ppk}, [\mathbf{c}], \mathbf{r})$:
    $\mathbf{X} \leftarrow \mathsf{h}_0([\mathbf{P_x}]\mathbf{r}) \in \mathbb{Z}_q^{k\times k}$
    $\mathbf{y} \leftarrow \mathsf{h}_1([\mathbf{P_y}]\mathbf{r}) \in \mathbb{Z}_q^{k}$
    $[\pi] \leftarrow [\overline{\mathbf{A}_0}] \cdot \mathbf{X} + [\overline{\mathbf{c}}] \cdot \mathbf{y}^\top \in \mathbb{G}^{k\times k}$
    $[\mathbf{K}] \leftarrow [\underline{\mathbf{A}_0}] \cdot \mathbf{X} + [\underline{\mathbf{c}}] \cdot \mathbf{y}^\top \in \mathbb{G}^{k\times k}$
    $[\kappa] \leftarrow \mathsf{trace}([\mathbf{K}]) \in \mathbb{G}$
Return $([\pi], [\kappa])$
```

```
$\mathsf{PVer}(\mathsf{ppk}, \mathsf{psk}, [\mathbf{c}], [\pi^*])$:
    $([\pi], [\kappa]) \leftarrow \mathsf{PSim}(\mathsf{ppk}, \mathsf{psk}, [\mathbf{c}])$
Return $\begin{cases} (1, [\kappa]) & \text{If } [\pi] = [\pi^*] \\ (0, \perp) & \text{Otherwise} \end{cases}$
```

Fig. 10. Construction of the $\mathcal{L}^{\mathrm{snd}}$-qualified proof system $\mathsf{PS} = (\mathsf{PGen}, \mathsf{PPrv}, \mathsf{PVer}, \mathsf{PSim})$ in [GHK17].

[7] This construction in Figure 10 is an updated version of [GHK17] from a personal communication.

According to Theorem 1 of [GHK17], PS is $\mathcal{L}^{\mathrm{snd}}$-qualified and $\widetilde{\mathcal{L}^{\mathrm{snd}}}$-extensible, both admitting tight security reductions to the MDDH assumption. More precisely, $\mathsf{Adv}^{\mathrm{csnd}}_{\mathcal{L}^{\mathrm{snd}},\mathsf{PS},\mathcal{A}}(\lambda), \mathsf{Adv}^{\mathrm{csnd}}_{\widetilde{\mathcal{L}^{\mathrm{snd}}},\widetilde{\mathsf{PS}},\mathcal{A}}(\lambda) \le 2k \cdot \mathsf{Adv}^{\mathrm{mddh}}_{\mathcal{D}_{2k,k},\mathsf{GGen},\mathcal{B}}(\lambda) + 2^{-\Omega(\lambda)}$, $\mathsf{Adv}^{\mathsf{PS}\text{-}\mathrm{ind}}_{\mathcal{L}^{\mathrm{snd}}} \le 2^{-\Omega(\lambda)}$.

We now prove that PS has pseudorandom simulated proof with Theorem 4.

Theorem 4. *The $\mathcal{L}^{\mathrm{snd}}$-qualified proof system PS in Figure 10 has pseudorandom simulated proof if $\mathcal{U}_k$-MDDH assumption holds. Specifically, for each PPT adversary $\mathcal{A}$, we can build a PPT adversary $\mathcal{B}$ with $\mathbf{T}(\mathcal{B}) \le \mathbf{T}(\mathcal{A}) + (Q_{\mathrm{sim}} + Q_{\mathrm{ver}}) \cdot \mathsf{poly}(\lambda)$ such that the advantage*

$$\mathsf{Adv}^{\mathrm{pr\text{-}proof}}_{\mathsf{PS},\mathcal{A}}(\lambda) \le 2\mathsf{Adv}^{\mathrm{mddh}}_{\mathcal{U}_k,\mathsf{GGen},\mathcal{B}}(\lambda) + 2^{-\Omega(\lambda)}.$$

where $Q_{\mathrm{sim}}(Q_{\mathrm{ver}})$ is the total number of $\mathcal{O}_{\mathrm{sim}}(\mathcal{O}_{\mathrm{ver}})$ queries made by $\mathcal{A}$ and $\mathsf{poly}(\lambda)$ is a polynomial independent of $\mathbf{T}(\mathcal{A})$.

Proof of Theorem 4.

```
$\underline{\mathrm{Exp}^{\mathrm{pr\text{-}proof}}_{\mathsf{PS},\mathcal{A}}(\lambda)\text{:}}G_0\ \boxed{G_1 - G_3}$
$b \leftarrow_\$ \{0,1\}$
$\boxed{\mathbf{V} \leftarrow_\$ \mathbb{Z}_q^{(k^2+1)\times k}}$
$\mathbf{K_X} \leftarrow_\$ \mathbb{Z}_q^{(k^2+1)\times 2k}$
$\mathbf{K_y} \leftarrow_\$ \mathbb{Z}_q^{(k+1)\times 2k}$
$[\mathbf{P_X}] \leftarrow \mathbf{K_X}[\mathbf{A}]$
$[\mathbf{P_y}] \leftarrow \mathbf{K_y}[\mathbf{A}]$
$\mathsf{ppk} \leftarrow ([\mathbf{P_X}], [\mathbf{P_y}])$
$b' \leftarrow_\$ \mathcal{A}^{\mathcal{O}_{\mathrm{sim}}(),\mathcal{O}_{\mathrm{ver}}(\cdot,\cdot)}(\mathsf{ppk})$
Return $b'$
```

```
$\underline{\mathcal{O}_{\mathrm{sim}}()\text{:}}\ G_0\ \boxed{G_1}\ \boxed{G_2\ \colorbox{lightgray}{$G_3$}}$
$\mathbf{r} \leftarrow_\$ \mathbb{Z}_q^k$, $[\mathbf{c}] \leftarrow [\mathbf{A}_0]\mathbf{r}$
$\mathit{\Pi}_0 \leftarrow_\$ \mathbb{G}^{k\times k}$
$\mathbf{X} \leftarrow \mathsf{h}_0(\mathbf{K_X}[\mathbf{c}])$
$\boxed{\mathbf{X} \leftarrow \mathsf{h}_0([\mathbf{Vr}])}$
$\boxed{\mathbf{X} \leftarrow_\$ \mathbb{Z}_q^{k\times k}}$
$\mathbf{y} \leftarrow \mathsf{h}_1(\mathbf{K_y}[\mathbf{c}])$
$\mathit{\Pi}_1 \leftarrow [\overline{\mathbf{A}_0}] \cdot \mathbf{X} + [\overline{\mathbf{c}}] \cdot \mathbf{y}^\top$
$\colorbox{lightgray}{$\mathit{\Pi}_1 \leftarrow_\$ \mathbb{G}^{k\times k}$}$
Return $([\mathbf{c}], \mathit{\Pi}_b)$
```

```
$\underline{\mathcal{O}_{\mathrm{ver}}([\mathbf{c}], \mathit{\Pi}^*)\text{:}}\quad G_0 - G_3$
$\mathbf{X} \leftarrow \mathsf{h}_0(\mathbf{K_X}[\mathbf{c}])$
$\mathbf{y} \leftarrow \mathsf{h}_1(\mathbf{K_y}[\mathbf{c}])$
$\mathit{\Pi} \leftarrow [\overline{\mathbf{A}_0}] \cdot \mathbf{X} + [\overline{\mathbf{c}}] \cdot \mathbf{y}^\top$
$[\mathbf{K}] \leftarrow [\underline{\mathbf{A}_0}] \cdot \mathbf{X} + [\underline{\mathbf{c}}] \cdot \mathbf{y}^\top$
$[\kappa] \leftarrow \mathsf{trace}([\mathbf{K}])$
If $\begin{bmatrix} [\mathbf{c}] \notin \mathsf{span}([\mathbf{A}]) \\ \vee \mathit{\Pi} \neq \mathit{\Pi}^* \end{bmatrix}$:
    Return $\perp$
Return $[\kappa]$
```

Fig. 11. Games $G_0 - G_3$ in the proof of Theorem 4.

For a fixed PPT adversary $\mathcal{A}$, consider an experiment $\mathsf{Exp}^{\mathrm{pr\text{-}proof}}_{\mathsf{PS},\mathcal{A}}(\lambda)$ which first uniformly selects $b \leftarrow_\$ \{0,1\}$, then calls $\mathsf{Exp}^{\mathrm{pr\text{-}proof}\text{-}b}_{\mathsf{PS},\mathcal{A}}(\lambda)$ and gets its output b'. It is straightforward that

$$\mathsf{Adv}^{\mathrm{pr\text{-}proof}}_{\mathsf{PS},\mathcal{A}}(\lambda) = 2\left|\Pr[b' = b \text{ in } \mathsf{Exp}^{\mathrm{pr\text{-}proof}}_{\mathsf{PS},\mathcal{A}}(\lambda)] - \tfrac{1}{2}\right|.$$

Now we rewrite $\mathsf{Exp}^{\mathrm{pr\text{-}proof}}_{\mathsf{PS},\mathcal{A}}(\lambda)$ in Figure 11 and make changes to it gradually through game G_0 to G_3. Games $G_0 - G_3$ are defined as follows.

Game G_0. This game is the same as $\mathsf{Exp}^{\mathrm{pr\text{-}proof}}_{\mathsf{PS},\mathcal{A}}(\lambda)$. Then

$$\mathsf{Adv}^{\mathrm{pr\text{-}proof}}_{\mathsf{PS},\mathcal{A}}(\lambda) = 2\left|\Pr_0[b' = b] - \frac{1}{2}\right|. \tag{14}$$

Game $G_0 - G_1$. G_1 is almost the same as G_0 except for the $\mathcal{O}_{\mathrm{sim}}$ oracle.

- In G_0, $\mathbf{X} = \mathsf{h}_0(\mathbf{K_X}[\mathbf{c}])$, where $[\mathbf{c}] = [\mathbf{A}_0]\mathbf{r}$ and $\mathbf{r} \leftarrow_\$ \mathbb{Z}_q^k$ for each $\mathcal{O}_{\mathrm{sim}}$ query.
- In G_1, $\mathbf{X} = \mathsf{h}_0([\mathbf{Vr}])$, where (i) a fresh $\mathbf{r}$ is uniformly chosen from $\mathbb{Z}_q^k$ for each $\mathcal{O}_{\mathrm{sim}}$ query; (ii) $\mathbf{V}$ is uniformly chosen from $\mathbb{Z}_q^{(k^2+1)\times k}$ beforehand but will be fixed for each $\mathcal{O}_{\mathrm{sim}}$ query.

Define $\mathbf{U} := \mathbf{K}_\mathbf{X}\mathbf{A}_0$, so $(\mathbf{P}_\mathbf{X}|\mathbf{U}) = \mathbf{K}_\mathbf{X}(\mathbf{A}|\mathbf{A}_0)$. Note that, the square matrix $(\mathbf{A}|\mathbf{A}_0)$ is of full rank with probability $1 - 2^{-\Omega(\lambda)}$, then the entropy of $\mathbf{K}_\mathbf{X}$ is transferred to $(\mathbf{P}_\mathbf{X}|\mathbf{U})$ intactly. Recall that $\mathbf{K}_\mathbf{X}$ is uniform over $\mathbb{Z}_q^{(k^2+1)\times 2k}$. Therefore, $(\mathbf{P}_\mathbf{X}|\mathbf{U})$ is uniform over $\mathbb{Z}_q^{(k^2+1)\times 2k}$ as well. Consequently, $\mathbf{U}$ is uniformly distributed over $\mathbb{Z}_q^{(k^2+1)\times k}$ even conditioned on $\mathbf{P}_\mathbf{X}$.

In G_0, the $\mathcal{O}_{\text{ver}}$ oracle rejects all $[\mathbf{c}] \notin [\mathsf{span}(\mathbf{A})]$. Therefore, the information of $\mathbf{K}_\mathbf{X}$ leaked through $\mathcal{O}_{\text{ver}}$ is characterized by the public key $\mathbf{P}_\mathbf{X}$. Together with the fact that $[\mathbf{c}] = [\mathbf{A}_0]\mathbf{r}$ in $\mathcal{O}_{\text{sim}}$ of G_0 and G_1, the computation of $\mathbf{K}_\mathbf{X}[\mathbf{c}] = [\mathbf{K}_\mathbf{X}\mathbf{A}_0]\mathbf{r}$ in $\mathcal{O}_{\text{sim}}$ of G_0 can be replaced with $[\mathbf{V}]\mathbf{r}$ for $\mathbf{V} \leftarrow_\$ \mathbb{Z}_q^{(k^2+1)\times k}$ in G_1. Thus we have

$$|\Pr_0[b' = b] - \Pr_1[b' = b]| \leq 2^{-\Omega(\lambda)}. \tag{15}$$

<u>**Game $G_1 - G_2$.**</u> G_2 is the same as G_1 except for the $\mathcal{O}_{\text{sim}}$ oracle.

- In G_1, $\mathbf{X} = \mathsf{h}_0([\mathbf{Vr}])$ is computed with the same $\mathbf{V}$ but a fresh $\mathbf{r} \leftarrow_\$ \mathbb{Z}_q^k$.
- In G_2, $\mathbf{X}$ is uniformly selected from $\mathbb{Z}_q^{k\times k}$ for each $\mathcal{O}_{\text{sim}}$ oracle.

We will show that

$$|\Pr_1[b' = b] - \Pr_2[b' = b]| \leq \mathsf{Adv}^{\text{mddh}}_{\mathcal{U}_k,\mathsf{GGen},\mathcal{B}}(\lambda) + 2^{-\Omega(\lambda)}. \tag{16}$$

To prove (16), we define two intermediate games G_1' and G_1''.

G_1' is the same as G_1 except for the generation of $\mathbf{r}$ in $\mathcal{O}_{\text{sim}}$. For each $\mathcal{O}_{\text{sim}}$ query,

– in G_1, $\mathbf{r} \leftarrow_\$ \mathbb{Z}_q^k$;

– in G_1', $\mathbf{r} \leftarrow \mathbf{Ws}$ with a fresh $\mathbf{s} \leftarrow_\$ \mathbb{Z}_q^k$ but the same $\mathbf{W}$, which is uniformly selected from $\mathbb{Z}_q^{k\times k}$ beforehand.

Since $\mathbf{W}$ is invertible with probability $1 - 2^{-\Omega(\lambda)}$, we have that

$$|\Pr_1[b' = b] - \Pr_{1'}[b' = b]| \leq 2^{-\Omega(\lambda)}. \tag{17}$$

G_1'' is the same with G_1' except for the $\mathcal{O}_{\text{sim}}$ oracle. For each $\mathcal{O}_{\text{sim}}$ query,

– G_1' sets $[\mathbf{c}] \leftarrow \mathbf{A}_0[\mathbf{W}]\mathbf{s}$ and $\mathbf{X} \leftarrow \mathsf{h}_0([\mathbf{VW}]\mathbf{s})$, where $\mathbf{s} \leftarrow_\$ \mathbb{Z}_q^k$;

– G_1'' sets $[\mathbf{c}] \leftarrow \mathbf{A}_0[\mathbf{r}]$ and $\mathbf{X} \leftarrow \mathsf{h}_0([\mathbf{u}])$, where $\mathbf{r} \leftarrow_\$ \mathbb{Z}_q^k, \mathbf{u} \leftarrow_\$ \mathbb{Z}_q^{k^2+1}$.

Note that, with overwhelming probability, $[\mathbf{B}] = [\begin{smallmatrix}\mathbf{W}\\ \mathbf{VW}\end{smallmatrix}]$ distributes uniformly over $\mathbb{G}^{(k^2+k+1)\times k}$. Then we can build an adversary $\mathcal{B}$ and show that

$$|\Pr_{1'}[b' = b] - \Pr_{1''}[b' = b]| \leq \mathsf{Adv}^{\text{mddh}}_{\mathcal{U}_k,\mathsf{GGen},\mathcal{B}}(\lambda) + 2^{-\Omega(\lambda)}. \tag{18}$$

To prove (18), we construct an adversary $\mathcal{B}'$ and show that

$$|\Pr_{1'}[b' = b] - \Pr_{1''}[b' = b]| \leq \mathsf{Adv}^{Q_{\text{sim}}\text{-mddh}}_{\mathcal{U}_{k^2+k+1,k},\mathsf{GGen},\mathcal{B}'}(\lambda). \tag{19}$$

Upon receiving a challenge $(\mathcal{G}, [\mathbf{B}] \in \mathbb{G}^{(k^2+k+1)\times k}, [\mathbf{H}] := ([\mathbf{h}_1|\cdots|\mathbf{h}_{Q_{\text{sim}}}]) \in \mathbb{G}^{(k^2+k+1)\times Q_{\text{sim}}})$ for the Q_{sim}-fold $\mathcal{U}_{k^2+k+1,k}$-MDDH problem, $\mathcal{B}'$ simulates game $G_1'(G_1'')$. In the simulation of the i-th $\mathcal{O}_{\text{sim}}$ oracle query for $i \in [Q_{\text{sim}}]$, $\mathcal{B}'$ embeds $[\overline{\mathbf{h}_i}]$ in $[\mathbf{c}]$ with $[\mathbf{c}] \leftarrow \mathbf{A}_0[\overline{\mathbf{h}_i}]$. Then $\mathcal{B}'$ embeds $[\underline{\mathbf{h}_i}]$ in $\mathbf{X}$ with $\mathbf{X} \leftarrow \mathsf{h}_0([\underline{\mathbf{h}_i}])$.

If $[\mathbf{h}_i]$ is uniformly chosen from $\mathsf{span}([\mathbf{B}])$ for all $i \in [Q_{\text{sim}}]$, then $[\mathbf{h}_i] = [\begin{smallmatrix}\mathbf{W}\\ \mathbf{VW}\end{smallmatrix}]\mathbf{s}_i$, $[\overline{\mathbf{h}_i}] = [\mathbf{W}]\mathbf{s}_i$ and $[\underline{\mathbf{h}_i}] = [\mathbf{VW}]\mathbf{s}_i$ with $\mathbf{s}_i \leftarrow_\$ \mathbb{Z}_q^k$. In this case, $\mathcal{B}'$ perfectly simulates G_1'. If $[\mathbf{h}_i]$ is uniformly chosen from $\mathbb{G}^{k^2+k+1}$ for all $i \in [Q_{\text{sim}}]$, then both $[\overline{\mathbf{h}_i}]$ and $[\underline{\mathbf{h}_i}]$ are uniform. In this case, $\mathcal{B}'$ perfectly simulates G_1''.

From above, (19) follows.

Finally, (18) follows from (19), Lemma 6 and Lemma 3.

In G_1'', $\mathbf{X} \leftarrow \mathsf{h}_0([\mathbf{u}])$ for a uniform $\mathbf{u} \leftarrow_\$ \mathbb{Z}_q^{k^2+1}$. Since h_0 is universal, by Lemma 2 and a union bound, we have that

$$|\Pr_{1''}[b' = b] - \Pr_2[b' = b]| \leq \frac{Q_{\text{sim}}}{2\sqrt{q}} = 2^{-\Omega(\lambda)}. \tag{20}$$

Then (16) follows from (17, 18) and (20).

<u>**Game $G_2 - G_3$.**</u> G_3 is the same as G_2 except for the $\mathcal{O}_{\text{sim}}$ oracle.
For each $\mathcal{O}_{\text{sim}}$ query,

- in G_2, $\Pi_1 = [\overline{\mathbf{A}_0}] \cdot \mathbf{X} + [\overline{\mathbf{c}}] \cdot \mathbf{y}^\top$ for $[\mathbf{c}] = [\mathbf{A}_0]\mathbf{r}$ and a fresh $\mathbf{X} \leftarrow_\$ \mathbb{Z}_q^{k \times k}$;
- in G_3, Π_1 is uniformly selected from $\mathbb{G}^{k \times k}$.

Note that in G_2,

$$\Pi_1 = [\overline{\mathbf{A}_0}] \cdot \mathbf{X} + [\overline{\mathbf{c}}] \cdot \mathbf{y}^\top = [\overline{\mathbf{A}_0}](\mathbf{X} + \mathbf{r} \cdot \mathbf{y}^\top).$$

Due to the uniformness of $\mathbf{X}$, Π_1 has the same distribution as $[\overline{\mathbf{A}_0}]\mathbf{X}$. Since $\overline{\mathbf{A}_0}$ is an invertible matrix, $[\overline{\mathbf{A}_0}]\mathbf{X}$ is uniformly distributed over $\mathbb{G}^{k \times k}$. Thus we have

$$\Pr_2[b' = b] = \Pr_3[b' = b]. \tag{21}$$

<u>**Game G_3.**</u> In G_3, Π_0 distributes identically to Π_1 and

$$\Pr_3[b' = b] = \frac{1}{2}. \tag{22}$$

Finally, Theorem 4 follows from (14, 15, 16, 21) and (22). ■

```
$\underline{(\mathsf{pk_{kem}}, \mathsf{sk_{kem}}) \leftarrow_\$ \mathsf{KGen}(1^\lambda):}$
    $(\mathsf{ppk}, \mathsf{psk}) \leftarrow_\$ \mathsf{PGen}(\mathsf{pars})$
    $\mathbf{k}_0, \mathbf{k}_1 \leftarrow_\$ \mathbb{Z}_q^{2k}, \quad [\mathbf{p}_0^\top] \leftarrow \mathbf{k}_0^\top[\mathbf{A}] \in \mathbb{G}^{1 \times k}, \quad [\mathbf{p}_1^\top] \leftarrow \mathbf{k}_1^\top[\mathbf{A}] \in \mathbb{G}^{1 \times k}$
Return  $\mathsf{pk_{kem}} \leftarrow (\mathsf{ppk}, [\mathbf{p}_0^\top], [\mathbf{p}_1^\top]), \quad \mathsf{sk_{kem}} \leftarrow (\mathsf{psk}, \mathbf{k}_0, \mathbf{k}_1)$

$\underline{(\psi, \gamma) \leftarrow_\$ \mathsf{KEnc}(\mathsf{pk_{kem}}):}$
    $\mathbf{r} \leftarrow_\$ \mathbb{Z}_q^k, \quad [\mathbf{c}] \leftarrow [\mathbf{A}]\mathbf{r} \in \mathbb{G}^{2k}$
    $(\Pi, [\kappa]) \leftarrow_\$ \mathsf{PPrv}(\mathsf{ppk}, [\mathbf{c}], \mathbf{r})$
    $\tau \leftarrow \mathsf{H}([\overline{\mathbf{c}}]) \in \{0,1\}^\lambda \subseteq \mathbb{Z}_q$
    $\gamma \leftarrow ([\mathbf{p}_0^\top] + \tau[\mathbf{p}_1^\top]) \cdot \mathbf{r} + [\kappa] \in \mathbb{G}$
Return  $(\psi \leftarrow ([\mathbf{c}], \Pi), \gamma)$
$//\Psi = \mathbb{G}^{2k} \times \mathbb{G}^{k \times k}, \quad \Gamma = \mathbb{G}$

$\underline{\gamma/\perp \leftarrow \mathsf{KDec}(\mathsf{sk_{kem}}, \psi):}$
    Parse $\psi = ([\mathbf{c}], \Pi)$
    $(v \in \{0,1\}, [\kappa]) \leftarrow \mathsf{PVer}(\mathsf{psk}, [\mathbf{c}], \Pi)$
    $\tau \leftarrow \mathsf{H}([\overline{\mathbf{c}}]) \in \{0,1\}^\lambda \subseteq \mathbb{Z}_q$
    $\gamma \leftarrow (\mathbf{k}_0^\top + \tau\mathbf{k}_1^\top) \cdot [\mathbf{c}] + [\kappa] \in \mathbb{G}$
Return $\begin{cases} \gamma & \text{If } v = 1 \\ \perp & \text{Otherwise} \end{cases}$
```

Fig. 12. Construction of $\mathsf{KEM_{qps}} = (\mathsf{KGen}, \mathsf{KEnc}, \mathsf{KDec})$ in [GHK17]

KEM from Qualified Proof System. The construction of the qualified PS based KEM $\mathsf{KEM_{qps}} = (\mathsf{KGen}, \mathsf{KEnc}, \mathsf{KDec})$ from [GHK17] is shown in Figure 12. Suppose $\mathcal{H}$ is a hash generator outputting functions $\mathsf{H} : \mathbb{G}^k \to \{0,1\}^\lambda$. The parameters pars used in this construction are specified in Section 5.2.

Theorem 2 in [GHK17] has shown that $\mathsf{KEM}_{\mathsf{qps}}$ is IND-CCCA secure. Now we prove that $\mathsf{KEM}_{\mathsf{qps}}$ is mPR-CCCA secure (through Theorem 5) and is RER secure (through Theorem 6), both admitting tight security reductions.

Theorem 5. *The KEM $\mathsf{KEM}_{\mathsf{qps}}$ in Figure 12 is mPR-CCCA secure if the $\mathcal{D}_{2k,k}$-MDDH assumption holds, $\mathcal{H}$ outputs collision-resistant hash function, PS is $\mathcal{L}^{snd}$-qualified, $\widetilde{\mathcal{L}^{snd}}$-extensible and has pseudorandom simulated proof. Specifically, for each PPT adversary $\mathcal{A}$ with negligible uncertainty $uncert_{\mathcal{A}}(\lambda)$, we can build PPT adversaries $\mathcal{B}_1, \cdots, \mathcal{B}_7$ with $\mathbf{T}(\mathcal{B}_1) \approx \cdots \approx \mathbf{T}(\mathcal{B}_7) \leq \mathbf{T}(\mathcal{A}) + (Q_{\mathrm{enc}} + Q_{\mathrm{dec}}) \cdot \mathsf{poly}(\lambda)$ and $uncert_{\mathcal{B}_4}(\lambda) = uncert_{\mathcal{B}_6}(\lambda) = uncert_{\mathcal{A}}(\lambda)$, such that the advantage*

$$\begin{aligned}\mathsf{Adv}^{\text{mpr-ccca}}_{\mathsf{KEM}_{\mathsf{qps}},\mathcal{A}}(\lambda) \leq\ & 2\mathsf{Adv}^{\text{cr}}_{\mathcal{H},\mathcal{B}_1}(\lambda) + (4\lambda + 3k)\mathsf{Adv}^{\text{mddh}}_{\mathcal{D}_{2k,k},\mathsf{GGen},\mathcal{B}_2}(\lambda) \\ & + 7\mathsf{Adv}^{\text{mddh}}_{\mathcal{U}_k,\mathsf{GGen},\mathcal{B}_3}(\lambda) + \mathsf{Adv}^{\text{csnd}}_{\mathcal{L}^{\text{snd}},\mathsf{PS},\mathcal{B}_4}(\lambda) + \mathsf{Adv}^{\text{PS-ind}}_{\mathcal{L}^{\text{snd}},\mathsf{PS},\widetilde{\mathsf{PS}},\mathcal{B}_5}(\lambda) \\ & + \lambda\mathsf{Adv}^{\text{csnd}}_{\widetilde{\mathcal{L}^{\text{snd}}},\widetilde{\mathsf{PS}},\mathcal{B}_6}(\lambda) + 2\mathsf{Adv}^{\text{pr-proof}}_{\mathsf{PS},\mathcal{B}_7}(\lambda) \\ & + ((\lambda + 2) \cdot Q_{\mathrm{enc}} + 3) \cdot Q_{\mathrm{dec}} \cdot uncert_{\mathcal{A}}(\lambda) + 2^{-\Omega(\lambda)}.\end{aligned}$$

where $Q_{\mathrm{enc}}(Q_{\mathrm{dec}})$ is the total number of $\mathcal{O}_{\mathrm{enc}}(\mathcal{O}_{\mathrm{dec}})$ queries made by $\mathcal{A}$ and $\mathsf{poly}(\lambda)$ is a polynomial independent of $\mathbf{T}(\mathcal{A})$.

Proof of Theorem 5. For a fixed PPT adversary $\mathcal{A}$ with negligible uncertainty $uncert_{\mathcal{A}}(\lambda)$, consider an experiment $\mathsf{Exp}^{\text{mpr-ccca}}_{\mathsf{KEM}_{\mathsf{qps}},\mathcal{A}}(\lambda)$ which first randomly selects $b \leftarrow_{\$} \{0,1\}$, then calls $\mathsf{Exp}^{\text{mpr-ccca-}b}_{\mathsf{KEM}_{\mathsf{qps}},\mathcal{A}}(\lambda)$ and gets its output b'. It is straightforward that $\mathsf{Adv}^{\text{mpr-ccca}}_{\mathsf{KEM}_{\mathsf{qps}},\mathcal{A}}(\lambda) = 2\left|\Pr[b' = b \text{ in } \mathsf{Exp}^{\text{mpr-ccca}}_{\mathsf{KEM}_{\mathsf{qps}},\mathcal{A}}(\lambda)] - \frac{1}{2}\right|$. Then we rewrite experiment $\mathsf{Exp}^{\text{mpr-ccca}}_{\mathsf{KEM}_{\mathsf{qps}},\mathcal{A}}(\lambda)$ in Figure 13 and make changes to it gradually through game G_0 to G_9 which are defined as follows.

Game G_0. This game is identical to $\mathsf{Exp}^{\text{mpr-ccca}}_{\mathsf{KEM}_{\mathsf{qps}},\mathcal{A}}(\lambda)$. Then

$$\mathsf{Adv}^{\text{mpr-ccca}}_{\mathsf{KEM}_{\mathsf{qps}},\mathcal{A}}(\lambda) = 2\left|\Pr_0[b' = b] - \frac{1}{2}\right|. \tag{23}$$

Game $G_0 - G_1$. G_1 is the same as G_0 except that an additional rejection rule is added in $\mathcal{O}_{\mathrm{dec}}$. More precisely, in G_1, we use a set $\mathcal{T}$ to log all the tags $\tau_b = \mathsf{H}([\overline{\mathbf{c}_b}])$ used in oracle $\mathcal{O}_{\mathrm{enc}}$, and any $\mathcal{O}_{\mathrm{dec}}(\mathsf{pred}, \psi = ([\mathbf{c}], \Pi))$ query will be rejected if $\tau = \mathsf{H}([\overline{\mathbf{c}}]) \in \mathcal{T}$.

Lemma 1.

$$\begin{aligned}|\Pr_0[b' = b] - \Pr_1[b' = b]| \leq\ & \mathsf{Adv}^{\text{cr}}_{\mathcal{H},\mathcal{B}_1}(\lambda) + \frac{k}{2} \cdot \mathsf{Adv}^{\text{mddh}}_{\mathcal{D}_{2k,k},\mathsf{GGen},\mathcal{B}_2}(\lambda) \\ & + \frac{1}{2}\mathsf{Adv}^{\text{mddh}}_{\mathcal{U}_k,\mathsf{GGen},\mathcal{B}_3}(\lambda) + \frac{3}{2}Q_{\mathrm{dec}} \cdot uncert_{\mathcal{A}}(\lambda) + 2^{-\Omega(\lambda)}.\end{aligned}$$

We put the proof of this lemma in Appendix F.

Game $G_1 - G_2$. G_2 is almost the same as G_1 except for two changes in $\mathcal{O}_{\mathrm{enc}}$. The first change is that PPrv is replaced with PSim. The second change is that $\mathsf{sk}_{\mathsf{KEM}}$ is used to calculate γ_1. More precisely, for $[\mathbf{c}_1] = [\mathbf{A}]\mathbf{r}_1$ in oracle $\mathcal{O}_{\mathrm{enc}}$,

- in G_1, $(\Pi_1, [\kappa_1]) \leftarrow \mathsf{PPrv}(\mathsf{ppk}, [\mathbf{c}_1], \mathbf{r}_1)$, $\gamma_1 \leftarrow ([\mathbf{p}_0^\top] + \tau_1[\mathbf{p}_1^\top]) \cdot \mathbf{r}_1 + [\kappa_1]$;
- in G_2, $(\Pi_1, [\kappa_1]) \leftarrow \mathsf{PSim}(\mathsf{psk}, [\mathbf{c}_1])$, $\gamma_1 \leftarrow (\mathbf{k}_0^\top + \tau_1\mathbf{k}_1^\top) \cdot [\mathbf{c}_1] + [\kappa_1]$.

Due to the perfect zero-knowledge property of PS, we have $\mathsf{PPrv}(\mathsf{ppk}, [\mathbf{c}_1], \mathbf{r}_1) = \mathsf{PSim}(\mathsf{psk}, [\mathbf{c}_1])$. Meanwhile, $[\mathbf{p}_0^\top] = \mathbf{k}_0^\top[\mathbf{A}]$ and $[\mathbf{p}_1^\top] = \mathbf{k}_1^\top[\mathbf{A}]$, so we have $([\mathbf{p}_0^\top] + \tau_1[\mathbf{p}_1^\top]) \cdot \mathbf{r}_1 + [\kappa_1] = (\mathbf{k}_0^\top + \tau_1\mathbf{k}_1^\top) \cdot [\mathbf{c}_1] + [\kappa_1]$.

```
$\mathsf{Exp}^{\text{mpr-ccca}}_{\mathsf{KEM_{qps}},\mathcal{A}}(\lambda)$:
    $G_0$ $G_1 - G_5$ $\boxed{G_6}$ $G_7 - G_9$
$b \leftarrow_{\$} \{0,1\}$ $\mathcal{T} \leftarrow \emptyset$ $\boxed{[\mathbf{v}] \leftarrow_{\$} \mathbb{Z}_q^k}$
$(\mathsf{ppk}, \mathsf{psk}) \leftarrow_{\$} \mathsf{PGen}(\mathsf{pars})$
$\mathbf{k}_0, \mathbf{k}_1 \leftarrow_{\$} \mathbb{Z}_q^{2k}$
$[\mathbf{p}_0^\top] \leftarrow \mathbf{k}_0^\top[\mathbf{A}], [\mathbf{p}_1^\top] \leftarrow \mathbf{k}_1^\top[\mathbf{A}]$
$\mathsf{pk_{kem}} \leftarrow (\mathsf{ppk}, [\mathbf{p}_0^\top], [\mathbf{p}_1^\top])$
$b' \leftarrow_{\$} \mathcal{A}^{\mathcal{O}_{\mathrm{enc}}(), \mathcal{O}_{\mathrm{dec}}(\cdot,\cdot)}(\mathsf{pk_{kem}})$
Return $b'$

$\mathcal{O}_{\mathrm{dec}}(\mathsf{pred}, \psi = ([\mathbf{c}], \Pi))$:
    $G_0$ $\boxed{G_1 - G_4 \; G_5 - G_9}$
$(v, [\kappa]) \leftarrow \mathsf{PVer}(\mathsf{psk}, [\mathbf{c}], \Pi)$
$\tau \leftarrow \mathsf{H}([\overline{\mathbf{c}}]) \in \{0,1\}^\lambda \subseteq \mathbb{Z}_q$
$\gamma \leftarrow (\mathbf{k}_0^\top + \tau\mathbf{k}_1^\top) \cdot [\mathbf{c}] + [\kappa]$
If [ $([\mathbf{c}], \Pi) \in \boldsymbol{\psi}_{\mathrm{enc}}$
     $\vee\ v = 0$
     $\vee\ \mathsf{pred}(\gamma) = 0$
     $\vee [\mathbf{c}] \notin \mathsf{span}([\mathbf{A}])$
     $\boxed{\vee \tau \in \mathcal{T}}$ ]:
    Return $\perp$
Return $\gamma$

$\mathcal{O}_{\mathrm{enc}}()$: $G_0$ $\boxed{G_1 \; G_2 \; G_3 \; G_4 G_5}$
    $\boxed{G_6 \; G_7 \; G_8 \; G_9}$
$(\psi_0, \gamma_0) \leftarrow_{\$} \Psi \times \Gamma$
$\psi_0 = ([\mathbf{c}_0], \Pi_0)$
$\tau_0 \leftarrow \mathsf{H}([\overline{\mathbf{c}_0}])$
$\mathbf{r}_1 \leftarrow_{\$} \mathbb{Z}_q^k$
$[\mathbf{c}_1] \leftarrow [\mathbf{A}]\mathbf{r}_1$   $[\mathbf{c}_1] \leftarrow [\mathbf{A}_0]\mathbf{r}_1$
$[\mathbf{c}_1] \leftarrow_{\$} \mathbb{G}^{2k}$
$(\Pi_1, [\kappa_1]) \leftarrow \mathsf{PPrv}(\mathsf{ppk}, [\mathbf{c}_1], \mathbf{r}_1)$
$\boxed{(\Pi_1, [\kappa_1]) \leftarrow \mathsf{PSim}(\mathsf{psk}, [\mathbf{c}_1])}$
$\boxed{\Pi_1 \leftarrow_{\$} \mathbb{G}^{k \times k}}$
$\psi_1 \leftarrow ([\mathbf{c}_1], \Pi_1)$
$\boldsymbol{\psi}_{\mathrm{enc}} \leftarrow \boldsymbol{\psi}_{\mathrm{enc}} \cup \{\psi_b\}$
$\tau_1 \leftarrow \mathsf{H}([\overline{\mathbf{c}_1}]) \in \{0,1\}^\lambda \subseteq \mathbb{Z}_q$
$\boxed{\mathcal{T} \leftarrow \mathcal{T} \cup \{\tau_b\}}$
$\gamma_1 \leftarrow ([\mathbf{p}_0^\top] + \tau_1[\mathbf{p}_1^\top]) \cdot \mathbf{r}_1 + [\kappa_1]$
$\boxed{\gamma_1 \leftarrow (\mathbf{k}_0^\top + \tau_1\mathbf{k}_1^\top) \cdot [\mathbf{c}_1] + [\kappa_1]}$
$\boxed{\gamma_1 \leftarrow [\mathbf{v}^\top\mathbf{r}_1] + \tau_1\mathbf{k}_1^\top[\mathbf{c}_1] + [\kappa_1]}$
$u_1 \leftarrow_{\$} \mathbb{Z}_q$
$\gamma_1 \leftarrow [u_1] + \tau_1\mathbf{k}_1^\top[\mathbf{c}_1] + [\kappa_1]$
Return $(\psi_b, \gamma_b)$
```

Fig. 13. Game $G_0 - G_9$ in the proof of Theorem 5.

These changes are only conceptual, so G_1 is identical to G_2 and

$$\Pr_1[b' = b] = \Pr_2[b' = b]. \tag{24}$$

Game $G_2 - G_3$. G_3 is the same as G_2 except for one difference in $\mathcal{O}_{\mathrm{enc}}$.
- In game G_2, $[\mathbf{c}_1]$ is uniform over $\mathsf{span}([\mathbf{A}])$ for each $\mathcal{O}_{\mathrm{enc}}$ query.
- In game G_3, $[\mathbf{c}_1]$ is uniform over $\mathbb{G}^{2k}$ for each $\mathcal{O}_{\mathrm{enc}}$ query.

We can build an adversary $\mathcal{B}_2$ and show that

$$|\Pr_2[b' = b] - \Pr_3[b' = b]| \le k \cdot \mathsf{Adv}^{\mathrm{mddh}}_{\mathcal{D}_{2k,k},\mathsf{GGen},\mathcal{B}_2}(\lambda) + 2^{-\Omega(\lambda)}. \tag{25}$$

The reduction is straightforward, since $\mathcal{B}_2$ can simulate $G_2(G_3)$ by generating the secret key itself and embed its own challenge in $[\mathbf{c}_1]$. We omit the details.

A similar proof can be found in Appendix F.

Game $G_3 - G_4$. G_4 is the same as G_3 except for one difference in $\mathcal{O}_{\mathrm{enc}}$.
- In game G_3, $[\mathbf{c}_1]$ is uniform over $\mathbb{G}^{2k}$ for each $\mathcal{O}_{\mathrm{enc}}$ query.
- In game G_4, $[\mathbf{c}_1]$ is uniform over $\mathsf{span}([\mathbf{A}_0])$ for each $\mathcal{O}_{\mathrm{enc}}$ query.

We can build an adversary $\mathcal{B}_3$ and show that

$$|\Pr_3[b' = b] - \Pr_4[b' = b]| \le \mathsf{Adv}^{\mathrm{mddh}}_{\mathcal{U}_k,\mathsf{GGen},\mathcal{B}_3}(\lambda) + 2^{-\Omega(\lambda)}. \tag{26}$$

The reduction is straightforward and the proof of (26) is almost the same as (25).

Game $G_4 - G_5$. G_5 is almost the same as G_4 except that a rejection rule is added in $\mathcal{O}_{\text{dec}}$. More precisely, in G_5, an $\mathcal{O}_{\text{dec}}(\mathsf{pred}, \psi = ([\mathbf{c}], \mathit{\Pi}))$ query is directly rejected if $[\mathbf{c}] \notin \text{span}([\mathbf{A}])$. We have that

$$\begin{aligned} |\Pr_4[b' = b] - \Pr_5[b' = b]| \leq & \frac{1}{2}\mathsf{Adv}^{\text{csnd}}_{\mathcal{L}^{\text{snd}},\mathsf{PS},\mathcal{B}_4}(\lambda) + \frac{1}{2}\mathsf{Adv}^{\text{PS-ind}}_{\mathcal{L}^{\text{snd}},\mathsf{PS},\widetilde{\mathsf{PS}},\mathcal{B}_5}(\lambda) \\ & + 2\lambda \cdot \mathsf{Adv}^{\text{mddh}}_{\mathcal{D}_{2k,k},\mathsf{GGen},\mathcal{B}_2}(\lambda) + \frac{\lambda}{2}\mathsf{Adv}^{\text{csnd}}_{\widetilde{\mathcal{L}^{\text{snd}}},\widetilde{\mathsf{PS}},\mathcal{B}_6}(\lambda) \\ & + \frac{\lambda+2}{2} \cdot Q_{\text{enc}} \cdot Q_{\text{dec}} \cdot \mathit{uncert}_{\mathcal{A}}(\lambda) + Q_{\text{enc}} \cdot 2^{-\Omega(\lambda)}. \end{aligned} \tag{27}$$

The proof of (27) is the same as Lemma 9 in [GHK17]. We refer [GHK17] for details.

Game $G_5 - G_6$. G_6 is almost the same as G_5 except for one difference in $\mathcal{O}_{\text{enc}}$.

- In game G_5, $\gamma_1 = (\mathbf{k}_0^\top + \tau_1\mathbf{k}_1^\top) \cdot [\mathbf{c}_1] + [\kappa_1]$ for each $\mathcal{O}_{\text{enc}}$ query.
- In game G_6, $\gamma_1 = [\mathbf{v}^\top \mathbf{r}_1] + \tau_1\mathbf{k}_1^\top[\mathbf{c}_1] + [\kappa_1]$ where $\mathbf{v}$ is uniformly chosen from $\mathbb{Z}_q^k$ beforehand but will be fixed for each $\mathcal{O}_{\text{enc}}$ query.

We have that

$$|\Pr_5[b' = b] - \Pr_6[b' = b]| \leq 2^{-\Omega(\lambda)}. \tag{28}$$

The proof of (28) is almost the same as that of (15). We put it in Appendix G and omit the details here.

Game $G_6 - G_7$. G_7 is almost the same as G_6 except for one difference in $\mathcal{O}_{\text{enc}}$.

- In game G_6, $\gamma_1 = [\mathbf{v}^\top \mathbf{r}_1] + \tau_1\mathbf{k}_1^\top[\mathbf{c}_1] + [\kappa_1]$ for each $\mathcal{O}_{\text{enc}}$ query.
- In game G_7, $\gamma_1 \leftarrow [u_1] + \tau_1\mathbf{k}_1^\top[\mathbf{c}_1] + [\kappa_1]$ where $u_1 \leftarrow_\$ \mathbb{Z}_q$ for each $\mathcal{O}_{\text{enc}}$ query. In other words, γ_1 is uniform for each $\mathcal{O}_{\text{enc}}$ query in G_7.

We have that

$$|\Pr_6[b' = b] - \Pr_7[b' = b]| \leq \mathsf{Adv}^{\text{mddh}}_{\mathcal{U}_k,\mathsf{GGen},\mathcal{B}_3}(\lambda) + 2^{-\Omega(\lambda)}. \tag{29}$$

The proof of (29) is almost the same as that of (16). We can set $\mathbf{r}_1 = \mathbf{Ws}$ and $[\mathbf{B}] = \left[\begin{smallmatrix} \mathbf{W} \\ \mathbf{v}^\top\mathbf{W} \end{smallmatrix}\right] \in \mathbb{G}^{(k+1)\times k}$ which has the distribution $\mathcal{U}_{k+1,k}$ overwhelmingly. Then we can reduce the indistinguishability between G_6 and G_7 to the Q_{enc}-fold $\mathcal{U}_{k+1,k}$-MDDH assumption. We omit the detailed proof here.

Note that, in game G_7, $[\kappa_1]$ is not needed any longer since we can just select a uniform γ_1 for each $\mathcal{O}_{\text{enc}}$ query.

Game $G_7 - G_8$. G_8 is almost the same as G_7 except for one difference in $\mathcal{O}_{\text{enc}}$.

- In game G_7, $\mathit{\Pi}_1$ is the output of $\mathsf{PSim}(\mathsf{psk}, [\mathbf{c}_1])$ for each $\mathcal{O}_{\text{enc}}$ query.
- In game G_8, $\mathit{\Pi}_1$ is uniform selected for each $\mathcal{O}_{\text{enc}}$ query.

We can build an adversary $\mathcal{B}_7$ and show that

$$|\Pr_7[b' = b] - \Pr_8[b' = b]| \leq \mathsf{Adv}^{\text{pr-proof}}_{\mathsf{PS},\mathcal{B}_7}(\lambda). \tag{30}$$

On input ppk, $\mathcal{B}_7$ uniformly selects $b \leftarrow_\$ \{0,1\}$ and sets $\mathcal{T} \leftarrow \emptyset$. Then $\mathcal{B}_7$ uniformly selects $\mathbf{k}_0, \mathbf{k}_1 \leftarrow_\$ \mathbb{Z}_q^{2k}$ and sets $[\mathbf{p}_0^\top] \leftarrow \mathbf{k}_0^\top[\mathbf{A}], [\mathbf{p}_1^\top] \leftarrow \mathbf{k}_1^\top[\mathbf{A}], \mathsf{pk}_{\mathsf{KEM}} \leftarrow (\mathsf{ppk}, [\mathbf{p}_0^\top], [\mathbf{p}_1^\top])$. Then $\mathcal{B}_7$ calls $\mathcal{A}^{\mathcal{O}_{\text{enc}}(),\mathcal{O}_{\text{dec}}(\cdot,\cdot)}(\mathsf{pk}_{\mathsf{KEM}})$ by simulating the two oracles for $\mathcal{A}$ in the following way.

– For $\mathcal{A}$'s $\mathcal{O}_{\mathrm{enc}}()$ query, $\mathcal{B}_7$ uniformly chooses (ψ_0, γ_0) and calculates τ_0 just like game $G_7(G_8)$. Then $\mathcal{B}_7$ submits an $\mathcal{O}_{\mathrm{sim}}$ query to its own oracle and gets $([\mathbf{c}], \Pi)$ where $[\mathbf{c}]$ is uniform over $\mathcal{L}^{\mathrm{snd}} \backslash \mathcal{L} = \mathsf{span}([\mathbf{A}_0])$ and Π is either an output of $\mathsf{PSim}(\mathsf{psk}, [\mathbf{c}])$ or uniformly chosen from $\mathbf{\Pi}$. After that $\mathcal{B}_7$ sets $[\mathbf{c}_1] \leftarrow [\mathbf{c}]$ and $\Pi_1 \leftarrow \Pi$. Then $\mathcal{B}_7$ sets $\boldsymbol{\psi}_{\mathrm{enc}}$, calculates τ_1 from $[\overline{\mathbf{c}_1}]$ and uniformly selects γ_1 just like game $G_7(G_8)$. Finally $\mathcal{B}_7$ returns (ψ_b, γ_b) to $\mathcal{A}$.
– For $\mathcal{A}$'s $\mathcal{O}_{\mathrm{dec}}(\mathsf{pred}, \psi = ([\mathbf{c}], \Pi))$ query, $\mathcal{B}_7$ submits $\mathcal{O}_{\mathrm{ver}}([\mathbf{c}], \Pi)$ query to its own oracle and gets the response K. If $K = \perp$, $\mathcal{B}_7$ returns $\perp$ to $\mathcal{A}$. Since $K = \perp$ means $[\mathbf{c}] \notin \mathsf{span}([\mathbf{A}])$ or the verification $\mathsf{PVer}(\mathsf{psk}, [\mathbf{c}], \Pi)$ does not pass, $\mathcal{B}_7$ acts exactly the same as game $G_7(G_8)$ in such cases. If $[\kappa] = K \neq \perp$, $\mathcal{B}_7$ calculates τ and γ just like game $G_7(G_8)$. Then $\mathcal{B}_7$ tests if $([\mathbf{c}], \Pi) \in \boldsymbol{\psi}_{\mathrm{enc}}$ or $\mathsf{pred}(\gamma) = 0$ or $\vee \tau \in \mathcal{T}$ happens. If so, $\mathcal{B}_7$ returns $\perp$ to $\mathcal{A}$. Otherwise $\mathcal{B}_7$ returns γ to $\mathcal{A}$.

Finally, according to $\mathcal{A}$'s output b', $\mathcal{B}_7$ outputs 1 if and only if $b' = b$. It is clear that if Π is an output of $\mathsf{PSim}(\mathsf{psk}, [\mathbf{c}])$ for each $\mathcal{O}_{\mathrm{sim}}$ query, $\mathcal{B}_7$ perfectly simulates game G_7 for $\mathcal{A}$. And if Π is uniformly chosen from $\mathbf{\Pi}$ for each $\mathcal{O}_{\mathrm{sim}}$ query, $\mathcal{B}_7$ perfectly simulates game G_8 for $\mathcal{A}$. Thus (30) follows.

<u>**Game** $G_8 - G_9$**.**</u> G_9 is the same as G_8 except for one difference in $\mathcal{O}_{\mathrm{enc}}$.

- In game G_8, $[\mathbf{c}_1]$ is uniform selected from $\mathsf{span}([\mathbf{A}_0])$ for each $\mathcal{O}_{\mathrm{enc}}$ query.
- In game G_9, $[\mathbf{c}_1]$ is uniform selected from $\mathbb{G}^{2k}$ for each $\mathcal{O}_{\mathrm{enc}}$ query.

We can build an adversary $\mathcal{B}_3$ and show that

$$|\Pr_8[b' = b] - \Pr_9[b' = b]| \leq \mathsf{Adv}^{\mathrm{mddh}}_{\mathcal{U}_k, \mathsf{GGen}, \mathcal{B}_3}(\lambda) + 2^{-\Omega(\lambda)}. \tag{31}$$

The reduction is straightforward and the proof of (31) is the same as the proof for (25). We omit the details here.

<u>**Game** G_9**.**</u> In game G_9, (ψ_1, Π_1) is uniform over $\Psi \times \Gamma$ for each $\mathcal{O}_{\mathrm{enc}}$ query, which distributes exactly the same as (ψ_0, Π_0). Thus we have

$$\Pr_9[b' = b] = \frac{1}{2}. \tag{32}$$

Finally, Theorem 5 follows from (23), Lemma 1, (24)–(32). ■

Theorem 6. *The KEM* $\mathsf{KEM}_{\mathsf{qps}}$ *in Figure 12 is RER secure. Specifically, for each PPT adversary* $\mathcal{A}$ *with negligible uncertainty* $\mathit{uncert}_{\mathcal{A}}(\lambda)$*, the advantage* $\mathsf{Adv}^{\mathrm{rer}}_{\mathsf{KEM}_{\mathsf{qps}}, \mathcal{A}}(\lambda) \leq 2^{-\Omega(\lambda)}$.

Proof of Theorem 6. In $\mathsf{Exp}^{\mathrm{rer}\text{-}b}_{\mathsf{KEM}_{\mathsf{qps}}, \mathcal{A}}(\lambda)$, among all the $\mathcal{O}_{\mathrm{cha}}(\psi, \mathsf{pred})$ queries submitted by $\mathcal{A}$, if $\psi \notin \boldsymbol{\psi}_{\mathrm{ran}}$, the oracle $\mathcal{O}_{\mathrm{cha}}$ will answer $\mathcal{A}$ with $\mathsf{pred}(\mathsf{KDec}(\mathsf{sk}_{\mathsf{KEM}}, \psi))$. Thus no information about b is leaked to $\mathcal{A}$.

Therefore, we only consider those $\mathcal{O}_{\mathrm{cha}}(\psi, \mathsf{pred})$ queries such that $\psi = ([\mathbf{c}], \Pi) \in \boldsymbol{\psi}_{\mathrm{ran}}$. In this case, both $[\mathbf{c}]$ and Π are uniform.

If $b = 0$, $\mathcal{O}_{\mathrm{cha}}(\psi, \mathsf{pred})$ will always return 0 in $\mathsf{Exp}^{\mathrm{rer}\text{-}0}_{\mathsf{KEM}_{\mathsf{qps}}, \mathcal{A}}(\lambda)$.

If $b = 1$, $\mathcal{O}_{\mathrm{cha}}(\psi, \mathsf{pred})$ will use $\mathsf{KDec}(\mathsf{sk}_{\mathsf{KEM}}, \psi)$ to decapsulate ψ. More precisely, it will invoke $\mathsf{PVer}(\mathsf{psk}, [\mathbf{c}], \Pi)$ to obtain $(v, [\kappa])$ and output $\perp$ if $v = 0$. By the proof uniqueness of PS and the uniformness of Π, the probability that $v = 1$ in this query is at most $\frac{1}{|\mathbf{\Pi}|}$. Taking into account all the Q_{cha} queries, a union bound suggests that $\mathcal{O}_{\mathrm{cha}}(\psi, \mathsf{pred})$ always outputs 0 in $\mathsf{Exp}^{\mathrm{rer}\text{-}1}_{\mathsf{KEM}_{\mathsf{qps}}, \mathcal{A}}(\lambda)$ except with probability at most $\frac{Q_{\mathrm{cha}}}{|\mathbf{\Pi}|} = 2^{-\Omega(\lambda)}$.

Thus

$$\mathsf{Adv}^{\mathrm{rer}}_{\mathsf{KEM}_{\mathsf{qps}},\mathcal{A}}(\lambda) = \left|\Pr\left[\mathsf{Exp}^{\mathrm{rer}\text{-}0}_{\mathsf{KEM}_{\mathsf{qps}},\mathcal{A}}(\lambda) = 1\right] - \Pr\left[\mathsf{Exp}^{\mathrm{rer}\text{-}1}_{\mathsf{KEM}_{\mathsf{qps}},\mathcal{A}}(\lambda) = 1\right]\right| \leq 2^{-\Omega(\lambda)}.$$

■

Acknowledgments. Lin Lyu, Shengli Liu and Shuai Han are supported by the National Natural Science Foundation of China Grant (61672346, 61373153). Dawu Gu is supported by the National Natural Science Foundation of China Grant (U1636217) together with Program of Shanghai Academic Research Leader (16XD1401300).

References

[BHK12] Böhl, F., Hofheinz, D., Kraschewski, D.: On definitions of selective opening security. In: Fischlin, M., Buchmann, J.A., Manulis, M. (eds.) Public Key Cryptography - PKC 2012 - 15th International Conference on Practice and Theory in Public Key Cryptography, Darmstadt, Germany, May 21-23, 2012. Proceedings, Lecture Notes in Computer Science, vol. 7293, pp. 522–539. Springer (2012)

[BHY09] Bellare, M., Hofheinz, D., Yilek, S.: Possibility and impossibility results for encryption and commitment secure under selective opening. In: Joux, A. (ed.) Advances in Cryptology - EUROCRYPT 2009, 28th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Cologne, Germany, April 26-30, 2009. Proceedings, Lecture Notes in Computer Science, vol. 5479, pp. 1–35. Springer (2009)

[CG13] Canetti, R., Garay, J.A. (eds.): Advances in Cryptology - CRYPTO 2013 - 33rd Annual Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2013. Proceedings, Part II, Lecture Notes in Computer Science, vol. 8043. Springer (2013)

[CW13] Chen, J., Wee, H.: Fully, (almost) tightly secure IBE and dual system groups. In: Canetti and Garay [CG13], pp. 435–460

[DN00] Damgård, I., Nielsen, J.B.: Improved non-committing encryption schemes based on a general complexity assumption. In: Bellare, M. (ed.) Advances in Cryptology - CRYPTO 2000, 20th Annual International Cryptology Conference, Santa Barbara, California, USA, August 20-24, 2000, Proceedings, Lecture Notes in Computer Science, vol. 1880, pp. 432–450. Springer (2000)

[DNRS99] Dwork, C., Naor, M., Reingold, O., Stockmeyer, L.J.: Magic functions. In: 40th Annual Symposium on Foundations of Computer Science, FOCS '99, 17-18 October, 1999, New York, NY, USA, pp. 523–534. IEEE Computer Society (1999)

[EHK$^+$13] Escala, A., Herold, G., Kiltz, E., Ràfols, C., Villar, J.L.: An algebraic framework for diffie-hellman assumptions. In: Canetti and Garay [CG13], pp. 129–147

[FHKW10] Fehr, S., Hofheinz, D., Kiltz, E., Wee, H.: Encryption schemes secure against chosen-ciphertext selective opening attacks. In: Gilbert, H. (ed.) Advances in Cryptology - EUROCRYPT 2010, 29th Annual International Conference on the Theory and Applications of Cryptographic Techniques, French Riviera, May 30 - June 3, 2010. Proceedings, Lecture Notes in Computer Science, vol. 6110, pp. 381–402. Springer (2010)

[GHK17] Gay, R., Hofheinz, D., Kohl, L.: Kurosawa-desmedt meets tight security. In: Katz and Shacham [KS17], pp. 133–160

[GHKW16] Gay, R., Hofheinz, D., Kiltz, E., Wee, H.: Tightly cca-secure encryption without pairings. In: Fischlin, M., Coron, J. (eds.) Advances in Cryptology - EUROCRYPT 2016 - 35th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Vienna, Austria, May 8-12, 2016, Proceedings, Part I, Lecture Notes in Computer Science, vol. 9665, pp. 1–27. Springer (2016)

[GM84] Goldwasser, S., Micali, S.: Probabilistic encryption. J. Comput. Syst. Sci. vol. 28(2), pp. 270–299 (1984)

[HILL99] Håstad, J., Impagliazzo, R., Levin, L.A., Luby, M.: A pseudorandom generator from any one-way function. SIAM J. Comput. vol. 28(4), pp. 1364–1396 (1999)

[HJR16] Hofheinz, D., Jager, T., Rupp, A.: Public-key encryption with simulation-based selective-opening security and compact ciphertexts. In: Hirt, M., Smith, A.D. (eds.) Theory of Cryptography - 14th International Conference, TCC 2016-B, Beijing, China, October 31 - November 3, 2016, Proceedings, Part II, Lecture Notes in Computer Science, vol. 9986, pp. 146–168 (2016)

[HLOV11] Hemenway, B., Libert, B., Ostrovsky, R., Vergnaud, D.: Lossy encryption: Constructions from general assumptions and efficient selective opening chosen ciphertext security. In: Lee, D.H., Wang, X. (eds.) Advances in Cryptology - ASIACRYPT 2011 - 17th International Conference on the Theory and Application of Cryptology and Information Security, Seoul, South Korea, December 4-8, 2011. Proceedings, Lecture Notes in Computer Science, vol. 7073, pp. 70–88. Springer (2011)

[HLQ13] Huang, Z., Liu, S., Qin, B.: Sender-equivocable encryption schemes secure against chosen-ciphertext attacks revisited. In: Kurosawa, K., Hanaoka, G. (eds.) Public-Key Cryptography - PKC 2013 - 16th International Conference on Practice and Theory in Public-Key Cryptography, Nara, Japan, February 26 - March 1, 2013. Proceedings, Lecture Notes in Computer Science, vol. 7778, pp. 369–385. Springer (2013)

[HLQC13] Huang, Z., Liu, S., Qin, B., Chen, K.: Fixing the sender-equivocable encryption scheme in eurocrypt 2010. In: 2013 5th International Conference on Intelligent Networking and Collaborative Systems, Xi'an city, Shaanxi province, China, September 9-11, 2013, pp. 366–372. IEEE (2013)

[Hof12] Hofheinz, D.: All-but-many lossy trapdoor functions. In: Pointcheval, D., Johansson, T. (eds.) Advances in Cryptology - EUROCRYPT 2012 - 31st Annual International Conference on the Theory and Applications of Cryptographic Techniques, Cambridge, UK, April 15-19, 2012. Proceedings, Lecture Notes in Computer Science, vol. 7237, pp. 209–227. Springer (2012)

[Hof17] Hofheinz, D.: Adaptive partitioning. In: Coron, J., Nielsen, J.B. (eds.) Advances in Cryptology - EUROCRYPT 2017 - 36th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Paris, France, April 30 - May 4, 2017, Proceedings, Part III, Lecture Notes in Computer Science, vol. 10212, pp. 489–518 (2017)

[KS17] Katz, J., Shacham, H. (eds.): Advances in Cryptology - CRYPTO 2017 - 37th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 20-24, 2017, Proceedings, Part III, Lecture Notes in Computer Science, vol. 10403. Springer (2017)

[LDL$^+$14] Lai, J., Deng, R.H., Liu, S., Weng, J., Zhao, Y.: Identity-based encryption secure against selective opening chosen-ciphertext attack. In: Nguyen, P.Q., Oswald, E. (eds.) Advances in Cryptology - EUROCRYPT 2014 - 33rd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Copenhagen, Denmark, May 11-15, 2014. Proceedings, Lecture Notes in Computer Science, vol. 8441, pp. 77–92. Springer (2014)

[LP15] Liu, S., Paterson, K.G.: Simulation-based selective opening CCA security for PKE from key encapsulation mechanisms. In: Katz, J. (ed.) Public-Key Cryptography - PKC 2015 - 18th IACR International Conference on Practice and Theory in Public-Key Cryptography, Gaithersburg, MD, USA, March 30 - April 1, 2015, Proceedings, Lecture Notes in Computer Science, vol. 9020, pp. 3–26. Springer (2015)

[LSSS17] Libert, B., Sakzad, A., Stehlé, D., Steinfeld, R.: All-but-many lossy trapdoor functions and selective opening chosen-ciphertext security from LWE. In: Katz and Shacham [KS17], pp. 332–364

[PW08] Peikert, C., Waters, B.: Lossy trapdoor functions and their applications. In: Dwork, C. (ed.) Proceedings of the 40th Annual ACM Symposium on Theory of Computing, Victoria, British Columbia, Canada, May 17-20, 2008, pp. 187–196. ACM (2008)

[WC81] Wegman, M.N., Carter, L.: New hash functions and their use in authentication and set equality. J. Comput. Syst. Sci. vol. 22(3), pp. 265–279 (1981)

A Supplementary Materials for Preliminaries

A.1 Hash Functions

A hash function generator $\mathcal{H}$ is a PPT algorithm that, on input 1^λ, outputs an efficiently computable function $\mathsf{H} : \mathcal{X} \rightarrow \mathcal{Y}$.

Definition 10 (Collision Resistance). *A hash function generator $\mathcal{H}$ outputs collision resistant hash function H or H is collision resistant if for each PPT adversary $\mathcal{A}$, $\mathsf{Adv}^{\mathrm{cr}}_{\mathcal{H},\mathcal{A}}(\lambda) := \Pr[x \neq x' \wedge \mathsf{H}(x) = \mathsf{H}(x') \mid \mathsf{H} \leftarrow_{\$} \mathcal{H}(1^\lambda), (x, x') \leftarrow \mathcal{A}(1^\lambda, \mathsf{H})]$ is negligible.*

Definition 11 (Universal hash [WC81]). *A hash function generator $\mathcal{H}$ outputs universal hash function $\mathsf{H} : \mathcal{X} \rightarrow \mathcal{Y}$, or H is universal, if for all $x, x' \in \mathcal{X}$ with $x \neq x'$, it follows that $\Pr[\mathsf{H}(x) = \mathsf{H}(x') \mid \mathsf{H} \leftarrow_{\$} \mathcal{H}(1^\lambda)] \leq 1/|\mathcal{Y}|$.*

We state a simplified version of Leftover Hash Lemma with uniform input.

Lemma 2 (Leftover Hash Lemma [HILL99]). *Suppose that a hash function generator $\mathcal{H}$ outputs universal hash function $\mathsf{H} : \mathcal{X} \rightarrow \mathcal{Y}$. Then for $\mathsf{H} \leftarrow_{\$} \mathcal{H}(1^\lambda)$, it holds that $\Delta\left((\mathsf{H}, \mathsf{H}(U_{\mathcal{X}})), (\mathsf{H}, U_{\mathcal{Y}})\right) \leq \frac{1}{2} \cdot \sqrt{|\mathcal{Y}|/|\mathcal{X}|}$, where $U_{\mathcal{X}} \leftarrow_{\$} \mathcal{X}, U_{\mathcal{Y}} \leftarrow_{\$} \mathcal{Y}$, H and $U_{\mathcal{X}}$ are independent and $\Delta(\cdot)$ denotes the statistical distance.*

A.2 Matrix Decision Diffie-Hellman Assumption

We recall the definitions of the Matrix Decision Diffie-Hellman (MDDH) Assumptions in [EHK+13].

Definition 12 (Matrix Distribution). *Let $k, \ell \in \mathbb{N}$, with $\ell > k$. $\mathcal{D}_{\ell,k}$ is called a matrix distribution if it outputs matrices in $\mathbb{Z}_q^{\ell \times k}$ of full rank k in polynomial time. Define $\mathcal{D}_k := \mathcal{D}_{k+1,k}$.*

Without loss of generality, for $\mathbf{A} \leftarrow_{\$} \mathcal{D}_{\ell,k}$, we assume that $\overline{\mathbf{A}}$ is invertible.

Definition 13 ($\mathcal{D}_{\ell,k}$-Matrix Decision Diffie-Hellman Assumption, $\mathcal{D}_{\ell,k}$-MDDH). *Let $\mathcal{D}_{\ell,k}$ be a matrix distribution. The $\mathcal{D}_{\ell,k}$-Matrix Decision Diffie-Hellman ($\mathcal{D}_{\ell,k}$-MDDH) Assumption holds relative to GGen if for each PPT adversary $\mathcal{A}$,*

$$\mathsf{Adv}^{\mathrm{mddh}}_{\mathcal{D}_{\ell,k},\mathsf{GGen},\mathcal{A}}(\lambda) := |\Pr[\mathcal{A}(\mathcal{G}, [\mathbf{A}], [\mathbf{Aw}]) = 1] - \Pr[\mathcal{A}(\mathcal{G}, [\mathbf{A}], [\mathbf{u}]) = 1]|$$

is negligible, where the probability is taken over $\mathcal{G} \leftarrow_{\$} \mathsf{GGen}(1^\lambda), \mathbf{A} \leftarrow_{\$} \mathcal{D}_{\ell,k}, \mathbf{w} \leftarrow_{\$} \mathbb{Z}_q^k$ and $\mathbf{u} \leftarrow_{\$} \mathbb{Z}_q^\ell$.

For each $k \geq 1$, specific distributions $\mathcal{L}_k, \mathcal{SC}_k, \mathcal{C}_k$ (and others) over $\mathbb{Z}_q^{(k+1)\times k}$ were specified in [EHK+13]. $\mathcal{L}_k$-MDDH is the well-known k-Linear Assumption.

Definition 14 (Uniform Distribution). *Let $\ell, k \in \mathbb{N}$, with $\ell > k$. Denote by $\mathcal{U}_{\ell,k}$ the uniform distribution over the set of all full-rank $\ell \times k$ matrices over $\mathbb{Z}_q$. Let $\mathcal{U}_k := \mathcal{U}_{k+1,k}$.*

Lemma 3 **($\mathcal{U}_k$-MDDH $\Leftrightarrow$ $\mathcal{U}_{\ell,k}$-MDDH [GHKW16]).** *Let $\ell, k \in \mathbb{N}$, with $\ell > k$. For any PPT adversary $\mathcal{A}$, there exists a PPT adversary $\mathcal{B}$ (and vice versa) such that $\mathbf{T}(\mathcal{B}) \approx \mathbf{T}(A)$ and $\mathsf{Adv}^{\mathrm{mddh}}_{\mathcal{U}_{\ell,k},\mathsf{GGen},\mathcal{A}}(\lambda) = \mathsf{Adv}^{\mathrm{mddh}}_{\mathcal{U}_k,\mathsf{GGen},\mathcal{B}}(\lambda)$.*

Lemma 4 **($\mathcal{D}_{\ell,k}$-MDDH $\Rightarrow \mathcal{U}_k$-$\mathsf{MDDH}$ [EHK$^+$13]).** *Let $\mathcal{D}_{\ell,k}$ be a matrix distribution. For any PPT adversary $\mathcal{A}$, there exists a PPT adversary $\mathcal{B}$ such that $\mathbf{T}(\mathcal{B}) \approx \mathbf{T}(A)$ and $\mathsf{Adv}^{\mathrm{mddh}}_{\mathcal{D}_{\ell,k},\mathsf{GGen},\mathcal{B}}(\lambda) = \mathsf{Adv}^{\mathrm{mddh}}_{\mathcal{U}_k,\mathsf{GGen},\mathcal{A}}(\lambda)$.*

Definition 15 **(Q-Fold $\mathcal{D}_{\ell,k}$-Matrix Decision Diffie-Hellman Assumption,** Q-fold $\mathcal{D}_{\ell,k}$-MDDH**).** *Let $Q \geq 1$ and $\mathcal{D}_{\ell,k}$ be a matrix distribution. The Q-fold $\mathcal{D}_{\ell,k}$-Matrix Decision Diffie-Hellman (Q-fold $\mathcal{D}_{\ell,k}$-MDDH) Assumption holds relative to* GGen *if for each PPT adversary $\mathcal{A}$,*

$$\mathsf{Adv}^{\mathrm{Q\text{-}mddh}}_{\mathcal{D}_{\ell,k},\mathsf{GGen},\mathcal{A}}(\lambda) := |\Pr[\mathcal{A}(\mathcal{G}, [\mathbf{A}], [\mathbf{AW}]) = 1] - \Pr[\mathcal{A}(\mathcal{G}, [\mathbf{A}], [\mathbf{U}]) = 1]|$$

is negligible, where the probability is taken over $\mathcal{G} \leftarrow_{\$} \mathsf{GGen}(1^\lambda), \mathbf{A} \leftarrow_{\$} \mathcal{U}_{\ell,k}, \mathbf{W} \leftarrow_{\$} \mathbb{Z}_q^{k\times Q}, \mathbf{U} \leftarrow_{\$} \mathbb{Z}_q^{\ell\times Q}$.

Lemma 5 **(Random Self-Reducibility of $\mathcal{D}_{\ell,k}$-MDDH [EHK$^+$13]).** *Let $\ell, k, Q \in \mathbb{N}$ with $\ell > k$ and $Q > \ell - k$. For any PPT adversary $\mathcal{A}$, there exists a PPT adversary $\mathcal{B}$ such that $\mathbf{T}(\mathcal{B}) \approx \mathbf{T}(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$ where* poly *is a polynomial independent of $\mathbf{T}(\mathcal{A})$ and*

$$\begin{aligned}\mathsf{Adv}^{\mathrm{Q\text{-}mddh}}_{\mathcal{D}_{\ell,k},\mathsf{GGen},\mathcal{A}}(\lambda) &\leq (\ell - k) \cdot \mathsf{Adv}^{\mathrm{mddh}}_{\mathcal{D}_{\ell,k},\mathsf{GGen},\mathcal{B}}(\lambda) + \frac{1}{q-1} \\ &= (\ell - k) \cdot \mathsf{Adv}^{\mathrm{mddh}}_{\mathcal{D}_{\ell,k},\mathsf{GGen},\mathcal{B}}(\lambda) + 2^{-\Omega(\lambda)}.\end{aligned}$$

According to [EHK$^+$13], for the special case of $\mathcal{D}_{\ell,k} = \mathcal{U}_{\ell,k}$ there exists a tight reduction between the Q-fold $\mathcal{U}_{\ell,k}$-MDDH problem and the $\mathcal{U}_{\ell,k}$-MDDH problem.

Lemma 6 **(Random Self-Reducibility of $\mathcal{U}_{\ell,k}$-MDDH [EHK$^+$13]).** *Let $\ell, k, Q \in \mathbb{N}$ with $\ell > k$. For any PPT adversary $\mathcal{A}$, there exists a PPT adversary $\mathcal{B}$ such that $\mathbf{T}(\mathcal{B}) \approx \mathbf{T}(\mathcal{A}) + Q \cdot \mathsf{poly}(\lambda)$ where* poly *is a polynomial independent of $\mathbf{T}(\mathcal{A})$ and $\mathsf{Adv}^{\mathrm{Q\text{-}mddh}}_{\mathcal{U}_{\ell,k},\mathsf{GGen},\mathcal{A}}(\lambda) \leq \mathsf{Adv}^{\mathrm{mddh}}_{\mathcal{U}_{\ell,k},\mathsf{GGen},\mathcal{B}}(\lambda) + \frac{1}{q-1} = \mathsf{Adv}^{\mathrm{mddh}}_{\mathcal{U}_{\ell,k},\mathsf{GGen},\mathcal{B}}(\lambda) + 2^{-\Omega(\lambda)}$.*

Recall that the Decisional Diffie-Hellman (DDH) assumption is a special case of the MDDH assumption.

Definition 16 **(DDH Assumption).** *We say that the DDH assumption holds relative to a prime order group $\mathbb{G}$ if for each PPT adversary $\mathcal{A}$,*

$$\mathsf{Adv}^{\mathrm{ddh}}_{\mathbb{G},\mathcal{A}}(\lambda) := |\Pr[\mathcal{A}(\mathcal{G}, [a], [r], [ar]) = 1] - \Pr[\mathcal{A}(\mathcal{G}, [a], [r], [s]) = 1]|$$

is negligible, where the probability is taken over $\mathcal{G} = (\mathbb{G}, q, P) \leftarrow_{\$} \mathsf{GGen}(1^\lambda)$ and $a, r, s \leftarrow_{\$} \mathbb{Z}_q$.

DDH assumption is equivalent to $\mathcal{D}_{2,1}$-MDDH assumption where $\mathcal{D}_{2,1}$ is the distribution that outputs matrix $\left(\begin{smallmatrix}1\\a\end{smallmatrix}\right)$ for $a \leftarrow_{\$} \mathbb{Z}_q$.

A.3 Public Key Encryption

A PKE scheme PKE is made up of three PPT algorithms $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$, $\mathsf{Gen}(1^\lambda)$ outputs a public key and a secret key $(\mathsf{pk}, \mathsf{sk})$; $\mathsf{Enc}(\mathsf{pk}, m)$ takes as input the public key pk and a message m, and outputs a ciphertext C; $\mathsf{Dec}(\mathsf{sk}, C)$ takes as input the secret key sk and a ciphertext C, and it either outputs a message m or a failure symbol $\perp$. The correctness of a PKE scheme is relaxed to allow a negligible decryption error $\epsilon(\lambda)$. That is, for all m in the message space, all $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^\lambda)$, $\Pr[\mathsf{Dec}(\mathsf{sk}, \mathsf{Enc}(\mathsf{pk}, m)) = m] \geq 1 - \epsilon(\lambda)$ where the probability is taken over the randomnesses used in encryption.

A.4 Concrete Instance of XAC.

For completeness, we include below the construction of ℓ-cross-authentication codes proposed by Fehr et al. in [FHKW10]. It is also strong and semi-unique as shown in [LDL$^+$14].

- Let $\mathbb{F}_p$ be a finite field of size p, where p's bit-length is a function of the security parameter λ.
- $\mathcal{XK} = \mathcal{K}_x \times \mathcal{K}_y = \mathbb{F}_p^2$ and $\mathcal{XT} = \mathbb{F}_p^\ell \cup \{\perp\}$.
- $(x, y) \leftarrow_{\$} \mathsf{XGen}(1^\lambda)$, where $(x, y) \leftarrow_{\$} \mathbb{F}_p^2$.
- $T \leftarrow \mathsf{XAuth}((x_1, y_1), \cdots, (x_\ell, y_\ell))$. Let $\mathbf{A} \in \mathbb{F}_p^{\ell\times\ell}$ be a matrix consisting of rows $(1, x_i, x_i^2, \cdots, x_i^{\ell-1})$ for $i \in [\ell]$ and $\mathbf{B} = (y_1, \cdots, y_\ell)^\top \in \mathbb{F}_p^\ell$. If $\mathbf{A}T = \mathbf{B}$ has no solution or more than one solutions, set $T := \perp$. Otherwise, $\mathbf{A}$ is a Vandermonde matrix. Let tag $T = (T_0, \cdots, T_{\ell-1})^\top$ be the unique solution of the linear system $\mathbf{A}T = \mathbf{B}$.
- Define $T(z) = T_0 + T_1 z + \cdots + T_{\ell-1} z^{\ell-1} \in \mathbb{F}_p[z]$ with $T = (T_0, \cdots, T_{\ell-1})^\top$. $\mathsf{XVer}((x, y), T) = 1$ if and only if $T \neq \perp \wedge T(x) = y$.
- For $(x, y) \leftarrow_{\$} \mathcal{XK} = \mathbb{F}_p^2$ and any fixed $T \in \mathcal{XT}$, $\Pr[T(x) = y] = \frac{1}{p}$. So $\epsilon_{\mathsf{XAC}}^{\text{imp}}(\lambda) \leq \frac{1}{p}$.
- According to [FHKW10], $\epsilon_{\mathsf{XAC}}^{\text{sub}}(\lambda) \leq 2 \cdot \frac{\ell-1}{p}$.
- $(x, y) \leftarrow \mathsf{ReSamp}(T, i)$. Choose $x \leftarrow_{\$} \mathbb{F}_p$ and compute $y := T(x)$. Conditioned on $T = \mathsf{XAuth}((x_1, y_1), \cdots, (x_\ell, y_\ell))$ and $(x_j, y_j)_{j\in[\ell\setminus i]}$, the statistical distance between (x, y) and (x_i, y_i) is $\frac{\ell-1}{p}$. So $\delta(\lambda) = \frac{\ell-1}{p}$.
- Any $x \in \mathbb{F}_p$ uniquely determines $y := T(x) = T_0 + T_1 x + \cdots + T_{\ell-1} x^{\ell-1}$ such that $\mathsf{XVer}((x, y), T) = 1$.

B Detailed description of simulator construction

Here, we illustrate how the simulator is constructed for the SIM-SO-CCA proof.

- $\mathcal{S}_1$ calls Gen to obtain $(\mathsf{pk}, \mathsf{sk})$. Then it calls $\mathcal{A}_1^{\mathsf{Dec}(\cdot)}(\mathsf{pk})$ to obtain α and the state a_1. Note that $\mathcal{S}_1$ possesses sk and is able to provide the decryption oracle $\mathsf{Dec}(\cdot)$ to $\mathcal{A}_1$. The view of $\mathcal{A}_1$ is exactly the same as that in $\mathsf{Exp}_{\mathsf{PKE},\mathcal{A},n,\mathcal{M},R}^{\text{so-cca-real}}(\lambda)$.
- Without knowledge of $\mathbf{m} = (\mathbf{m}_1, \cdots, \mathbf{m}_n)$, which is the output of $\mathcal{M}(\alpha)$, $\mathcal{S}_2$ generates the challenge ciphertext vector $\mathbf{C} = (\mathbf{C}_1, \cdots, \mathbf{C}_n)$ with each $\mathbf{C}_i$ being an encryption of ℓ ones, i.e.,
$$\mathbf{C}_i = \mathsf{Enc}(\mathsf{pk}, 1^\ell; \mathbf{R}_i).$$
Then $\mathcal{S}_2$ calls $\mathcal{A}_2^{\mathsf{Dec}_{\notin \mathbf{C}}(\cdot)}(a_1, \mathbf{C})$ to get the corruption set I and the state a_2. Recall in the 'real' experiment $\mathsf{Exp}_{\mathsf{PKE},\mathcal{A},n,\mathcal{M},R}^{\text{so-cca-real}}(\lambda)$, $\mathcal{A}_2$ receives encryptions of real messages $\mathbf{m}$.
- $\mathcal{S}_3$ opens the challenge ciphertext vector $\mathbf{C}_I$ where $\mathbf{C}_i = \mathsf{Enc}(\mathsf{pk}, 1^\ell; \mathbf{R}_i) = (\psi_{i,1}, \cdots, \psi_{i,\ell}, T_i)$ according to the corrupted set of messages $\mathbf{m}_I$.
  - If $\mathbf{m}_{i,j} = 1$, $\mathcal{S}_3$ opens with the original randomnesses;
  - If $\mathbf{m}_{i,j} = 0$, $\mathcal{S}_3$ utilizes ReSamp to re-sample $\hat{K}_{i,j}$ so as to hide the real key $K_{i,j}$, and then uses $\mathsf{Sample}_{\mathcal{XK}}^{-1}$ to recover a properly distributed randomness for $\hat{K}_{i,j}$. It also uses $\mathsf{Sample}_{\Psi}^{-1}$ to recover a properly distributed randomness for $\psi_{i,j}$.

  Finally, $\mathcal{S}_3$ collects the newly opened randomness $\hat{\mathbf{R}}_I$ and calls $\mathcal{A}_3^{\mathsf{Dec}_{\notin \mathbf{C}}(\cdot)}(a_2, \mathbf{m}_I, \hat{\mathbf{R}}_I)$ to get the output $out_{\mathcal{A}}$ as its own output.

C Proof of Theorem 2

For a bit string $\boldsymbol{\tau} \in \{0,1\}^*$, we view $\boldsymbol{\tau}$ as a vector and denote by $\boldsymbol{\tau}_i$ the i-th bit of $\boldsymbol{\tau}$. We use $||$ to denote the concatenation of bit strings. For $i \in [|\boldsymbol{\tau}|]$, define $\boldsymbol{\tau}_{|i} := \boldsymbol{\tau}_1||\cdots||\boldsymbol{\tau}_i$ which is the prefix of $\boldsymbol{\tau}$. And let $\boldsymbol{\tau}_{|0} := \varepsilon$.

Fix a PPT adversary $\mathcal{A}$, consider an experiment $\mathsf{Exp}_{\mathsf{KEM}_{\mathsf{mddh}},\mathcal{A}}(\lambda)$ which selects a random bit $b \leftarrow_\$ \{0,1\}$, then calls $\mathsf{Exp}^{\text{mpr-ccca-}b}_{\mathsf{KEM}_{\mathsf{mddh}},\mathcal{A}}(\lambda)$ and gets its output b'. It is straightforward that

$$\mathsf{Adv}^{\text{mpr-ccca}}_{\mathsf{KEM}_{\mathsf{mddh}},\mathcal{A}}(\lambda) = 2\left|\Pr[b' = b \text{ in } \mathsf{Exp}_{\mathsf{KEM}_{\mathsf{mddh}},\mathcal{A}}(\lambda)] - \frac{1}{2}\right|.$$

We will focus on the event $b' = b$ in experiment $\mathsf{Exp}_{\mathsf{KEM}_{\mathsf{mddh}},\mathcal{A}}(\lambda)$ and gradually change this experiment to one in which $\Pr[b' = b] = \frac{1}{2}$. The difference caused by each change can be shown to be very small through a tight security reduction or an information theoretical analysis. These changes is shown in Figure 14 and game H_0 is almost the same as $\mathsf{Exp}_{\mathsf{KEM}_{\mathsf{mddh}},\mathcal{A}}(\lambda)$. If we use the notation $\Pr_i[\mathsf{E}]$ to denote the probability that event E happens in game H_i, then $\mathsf{Adv}^{\text{mpr-ccca}}_{\mathsf{KEM}_{\mathsf{mddh}},\mathcal{A}}(\lambda) = 2\left|\Pr_0[b' = b] - \frac{1}{2}\right|$.

```
$\mathsf{Exp}_{\mathsf{KEM}_{\mathsf{mddh}},\mathcal{A}}(\lambda)$:
        $H_0 - H_2$ $\boxed{H_3 - H_6\ \boxed{H_{7.i}}}$
$b \leftarrow_\$ \{0,1\}$ $\boxed{\mathcal{T}_{\text{enc}}, \mathcal{T}_{\text{dec}} \leftarrow \emptyset}$
$\mathbf{M} \leftarrow_\$ \mathcal{U}_{3k,k}$
$\boxed{\mathbf{M}^\perp \leftarrow_\$ \mathcal{U}_{3k,2k} \text{ s.t. } \mathbf{M}^\top\mathbf{M}^\perp = \mathbf{0}}$
$\boxed{\text{Simulate random function}}$
$\boxed{\mathsf{RF}_i : \{0,1\}^i \rightarrow \mathbb{Z}_q^{2k}}$
$\mathbf{k}_{1,0}, \cdots, \mathbf{k}_{\lambda,1} \leftarrow_\$ \mathbb{Z}_q^{3k}$
$\mathsf{pk}_{\mathsf{kem}} \leftarrow \begin{pmatrix} \mathcal{G}, \mathsf{H}, [\mathbf{M}] \\ ([\mathbf{M}^\top \mathbf{k}_{j,\beta}])^{1 \le j \le \lambda}_{0 \le \beta \le 1} \end{pmatrix}$
$b' \leftarrow_\$ \mathcal{A}^{\mathcal{O}_{\text{enc}}(), \mathcal{O}_{\text{dec}}(\cdot,\cdot)}(\mathsf{pk}_{\mathsf{kem}})$
Return $b'$
```

```
$\mathcal{O}_{\text{enc}}()$:        $H_0$ $\boxed{H_1, H_2}$
        $\boxed{\boxed{H_3, H_4}}$ $\boxed{H_5\ \boxed{H_6\ \boxed{H_{7.i}}}}$
$(\psi_0, \gamma_0) \leftarrow_\$ \mathbb{G}^{3k} \times \mathbb{G}$
$\boxed{\psi_0 \leftarrow_\$ \mathsf{span}([\mathbf{M}])}$ $\boxed{\tau^0 \leftarrow \mathsf{H}(\overline{\psi_0})}$
$[\mathbf{y}_1] \leftarrow_\$ \mathsf{span}([\mathbf{M}])$ $\boxed{[\mathbf{y}_1] \leftarrow_\$ \mathbb{G}^{3k}}$
$\psi_1 \leftarrow [\mathbf{y}_1], \tau^1 \leftarrow \mathsf{H}([\overline{\mathbf{y}_1}])$
$\boxed{\text{If } \tau^b \in \mathcal{T}_{\text{enc}} \cup \mathcal{T}_{\text{dec}}}$
$\boxed{\quad \text{Return } \perp}$
$\boxed{\mathcal{T}_{\text{enc}} \leftarrow \mathcal{T}_{\text{enc}} \cup \{\tau^b\}}$
$\mathbf{k}_{\tau^1} \leftarrow \sum_{j=1}^{\lambda} \mathbf{k}_{j,\tau^1_j}$ $\boxed{+\mathbf{M}^\perp \mathsf{RF}_i(\tau^1_{|i})}$
$\gamma_1 \leftarrow [\mathbf{y}_1^\top]\mathbf{k}_{\tau^1}$
$\boldsymbol{\psi}_{\text{enc}} \leftarrow \boldsymbol{\psi}_{\text{enc}} \cup \{\psi_b\}$
Return $(\psi_b, \gamma_b)$
```

```
$\mathcal{O}_{\text{dec}}(\mathsf{pred}, \psi = [\mathbf{y}])$:        $H_0, H_1$
        $\boxed{H_2\ \boxed{H_3}}$ $\boxed{H_4 - H_6\ \boxed{H_{7.i}}}$
If $\psi \in \boldsymbol{\psi}_{\text{enc}}$ $\boxed{\vee \mathbf{y} \notin \mathsf{span}(\mathbf{M})}$:
    Return $\perp$
$\tau \leftarrow \mathsf{H}([\overline{\mathbf{y}}])$
$\boxed{\text{If } \begin{bmatrix} \exists [\mathbf{y}'] \in \boldsymbol{\psi}_{\text{enc}} \text{ s.t.} \\ \tau = \mathsf{H}([\overline{\mathbf{y}'}]) \wedge \mathbf{y} \neq \mathbf{y}' \end{bmatrix}}$
$\boxed{\quad \text{Return } \perp}$
$\boxed{\mathcal{T}_{\text{dec}} \leftarrow \mathcal{T}_{\text{dec}} \cup \{\tau\}}$
$\mathbf{k}_\tau \leftarrow \sum_{j=1}^{\lambda} \mathbf{k}_{j,\tau_j}$ $\boxed{+\mathbf{M}^\perp \mathsf{RF}_i(\tau_{|i})}$
$\gamma \leftarrow [\mathbf{y}^\top]\mathbf{k}_\tau$
If $\mathsf{pred}(\gamma) = 0$:
    Return $\perp$
Return $\gamma$
```

Fig. 14. Games $H_0 - H_6$ and $H_{7.i}$ for $i \in \{0, \cdots, \lambda\}$.

Game H_0. This game is almost the same as $\mathsf{Exp}_{\mathsf{KEM}_{\mathsf{mddh}},\mathcal{A}}(\lambda)$ except for only one difference in $\mathcal{O}_{\text{enc}}$.

- In $\mathsf{Exp}_{\mathsf{KEM}_{\mathsf{mddh}},\mathcal{A}}(\lambda)$, the calculation of γ_1 is done publicly. It randomly selects $[\mathbf{y}_1]$ from $\mathsf{span}([\mathbf{M}])$ with randomness $\mathbf{r}_1$, $\tau^1 \leftarrow \mathsf{H}([\overline{\mathbf{y}_1}])$ and calculates γ_1 using public key and $\mathbf{r}_1$, i.e., $\gamma_1 \leftarrow \mathbf{r}_1^\top \cdot \sum_{j=1}^{\lambda} [\mathbf{M}^\top \mathbf{k}_{j,\tau^1_j}]$.
- In H_0, the calculation of γ_1 uses secret key. It randomly selects $[\mathbf{y}_1]$ from $\mathsf{span}([\mathbf{M}])$, $\tau^1 \leftarrow \mathsf{H}([\overline{\mathbf{y}_1}])$ and calculates $\mathbf{k}_{\tau^1} \leftarrow \sum_{j=1}^{\lambda} \mathbf{k}_{j,\tau^1_j}$ using the secret key. Finally it sets $\gamma_1 \leftarrow [\mathbf{y}_1^\top]\mathbf{k}_{\tau^1}$.

Since $\mathbf{r}_1^\top \cdot \sum_{j=1}^{\lambda}[\mathbf{M}^\top \mathbf{k}_{j,\tau_j^1}] = [(\mathbf{M}\mathbf{r}_1)^\top]\sum_{j=1}^{\lambda}\mathbf{k}_{j,\tau_j^1} = [\mathbf{y}_1^\top]\mathbf{k}_{\tau^1}$, this difference is only conceptual and H_0 is almost the same as $\mathsf{Exp}_{\mathsf{KEM}_{\mathsf{mddh}},\mathcal{A}}(\lambda)$. So

$$\mathsf{Adv}^{\text{mpr-ccca}}_{\mathsf{KEM}_{\mathsf{mddh}},\mathcal{A}}(\lambda) = 2\left|\Pr_0[b' = b] - \frac{1}{2}\right|. \tag{33}$$

Game $H_0 - H_1$. H_1 is almost the same as H_0 except for only one difference in $\mathcal{O}_{\text{enc}}$.

- In H_0, ψ_0 is randomly selected from $\mathbb{G}^{3k}$ in each $\mathcal{O}_{\text{enc}}$ query.
- In H_1, ψ_0 is randomly selected from $\mathsf{span}([\mathbf{M}])$ in each $\mathcal{O}_{\text{enc}}$ query.

We can build an adversary $\mathcal{B}$ and show that

$$|\Pr_0[b' = b] - \Pr_1[b' = b]| \leq \mathsf{Adv}^{\text{mddh}}_{\mathcal{U}_k,\mathsf{GGen},\mathcal{B}}(\lambda) + 2^{-\Omega(\lambda)}. \tag{34}$$

Let Q_{enc} be the total number of $\mathcal{O}_{\text{enc}}$ queries submitted by $\mathcal{A}$. To prove (34), we construct an adversary $\mathcal{B}'$ and show that

$$|\Pr_0[b' = b] - \Pr_1[b' = b]| \leq \mathsf{Adv}^{Q_{\text{enc}}\text{-mddh}}_{\mathcal{U}_{3k,k},\mathsf{GGen},\mathcal{B}'}(\lambda). \tag{35}$$

Upon receiving a challenge $(\mathcal{G}, [\mathbf{M}] \in \mathbb{G}^{3k\times k}, [\mathbf{H}] := ([\mathbf{h}_1|\cdots|\mathbf{h}_{Q_{\text{enc}}}]) \in \mathbb{G}^{3k\times Q_{\text{enc}}})$ for the Q_{enc}-fold $\mathcal{U}_{3k,k}$-MDDH problem, $\mathcal{B}'$ simulates game $H_0(H_1)$. It randomly selects $b \leftarrow_{\$} \{0,1\}$ and invoke $\mathcal{A}^{\mathcal{O}_{\text{enc}}(),\mathcal{O}_{\text{dec}}(\cdot,\cdot)}(\mathsf{pk}_{\mathsf{KEM}})$. Note that $\mathsf{pk}_{\mathsf{KEM}}$ and $\mathcal{O}_{\text{dec}}$ oracle can be perfectly simulated by $\mathcal{B}'$. To reply the i-th $\mathcal{O}_{\text{enc}}$ query made by $\mathcal{A}$, $\mathcal{B}'$ embeds $[\mathbf{h}_i]$ to ψ_0, i.e., $\psi_0 \leftarrow [\mathbf{h}_i]$. Finally $\mathcal{B}'$ gets $\mathcal{A}$'s output b' and outputs $1 \Leftrightarrow (b' = b)$. Thus, if each column $[\mathbf{h}_i]$ of $[\mathbf{H}]$ is uniformly random over $\mathbb{G}^{3k}$, $\mathcal{B}'$ perfectly simulates H_0. If each column $[\mathbf{h}_i]$ of $[\mathbf{H}]$ is uniformly random over $\mathsf{span}([\mathbf{M}])$, $\mathcal{B}'$ perfectly simulates H_1. So (35) follows.

Finally (34) follows from (35), Lemma 6 and Lemma 3.

Game $H_1 - H_2$. H_2 is almost the same as H_1 except for only one difference in $\mathcal{O}_{\text{dec}}$. A new "rejection rule"(outputting failure symbol $\perp$ if some condition is satisfied) is added into $\mathcal{O}_{\text{dec}}$. This new rule rejects any $\mathcal{O}_{\text{dec}}(\mathsf{pred}, \psi = [\mathbf{y}])$ query if $\mathbf{y} \notin \mathsf{span}(\mathbf{M})$. Note that this condition can be determined efficiently by fist sampling $\mathbf{M}^\perp \leftarrow_{\$} \mathcal{U}_{3k,2k}$ s.t. $\mathbf{M}^\top\mathbf{M}^\perp = \mathbf{0}$ and utilizing the relation $\mathbf{y} \notin \mathsf{span}(\mathbf{M}) \Leftrightarrow (\mathbf{M}^\perp)^\top\mathbf{y} \neq \mathbf{0} \Leftrightarrow (\mathbf{M}^\perp)^\top[\mathbf{y}] \neq [\mathbf{0}]$.

So, H_2 differs from H_1 only when $\mathcal{A}$ submits some $\mathcal{O}_{\text{dec}}(\mathsf{pred}, \psi = [\mathbf{y}])$ query s.t.

$$\mathbf{y} \notin \mathsf{span}(\mathbf{M}) \wedge \mathsf{pred}([\mathbf{y}^\top]\mathbf{k}_\tau) = 1. \tag{36}$$

If we denote this event as Bad, then it is straightforward that

$$|\Pr_1[b' = b] - \Pr_2[b' = b]| \leq \Pr_1[\mathsf{Bad}] = \Pr_2[\mathsf{Bad}] \tag{37}$$

Suppose the adversary $\mathcal{A}$ submits Q_{dec} $\mathcal{O}_{\text{dec}}$ queries in total. Then

$$\Pr_2[\mathsf{Bad}] \leq \sum_{i=1}^{Q_{\text{dec}}} \Pr_2[\mathsf{Bad}\text{ happens in the }i\text{-th query}]. \tag{38}$$

Let's fix some $i \in [Q_{\text{dec}}]$ and consider in H_2 the probability that Bad happens in the i-th $\mathcal{O}_{\text{dec}}$ query. To do this, we use the fact that $\mathbf{k}_{1,\beta} \leftarrow_{\$} \mathbb{Z}_q^{3k}$ are identically distributed as $\mathbf{k}_{1,\beta} + \mathbf{M}^\perp\mathbf{w}$ for $\beta \in \{0,1\}$, where $\mathbf{k}_{1,\beta} \leftarrow_{\$} \mathbb{Z}_q^{3k}, \mathbf{w} \leftarrow_{\$} \mathbb{Z}_q^{2k}$ and $\mathbf{M}^\perp \in \mathbb{Z}_q^{3k\times 2k}$ s.t. $\mathbf{M}^\top\mathbf{M}^\perp = \mathbf{0}$. Then we will show that $\mathbf{w}$ is hidden from $\mathcal{A}$ until the i-th $\mathcal{O}_{\text{dec}}$ query.

- The public key $\mathsf{pk}_{\mathsf{KEM}}$ does not leak any information about $\mathbf{w}$ since

$$\mathbf{M}^\top(\mathbf{k}_{1,\beta} + \mathbf{M}^\perp \mathbf{w}) = \mathbf{M}^\top \mathbf{k}_{1,\beta}.$$

- $\mathcal{O}_{\mathrm{enc}}$ also hides $\mathbf{w}$ since $\mathbf{w}$ is only used in the generation of γ_1.

$$\gamma_1 = [\mathbf{y}_1^\top(\mathbf{k}_{\tau^1} + \mathbf{M}^\perp \mathbf{w})] = [\mathbf{y}_1^\top \mathbf{k}_{\tau^1}] \tag{39}$$

due to $\mathbf{y}_1 \in \mathsf{span}(\mathbf{M})$ in H_2 and $\mathbf{y}_1^\top \mathbf{M}^\perp = \mathbf{0}$.
- The first $i-1$ $\mathcal{O}_{\mathrm{dec}}(\mathsf{pred}, [\mathbf{y}])$ queries also hides $\mathbf{w}$. Since in H_2, all $\mathbf{y} \notin \mathsf{span}(\mathbf{M})$ will be rejected by the rejection rule and will be independent of $\mathbf{w}$; all $\mathbf{y} \in \mathsf{span}(\mathbf{M})$ will not leak $\mathbf{w}$ due to similar reason of (39).

Thus $\mathbf{w}$ is not leaked to $\mathcal{A}$ at all until the i-th $\mathcal{O}_{\mathrm{dec}}$ query. So in this query $\mathcal{O}_{\mathrm{dec}}(\mathsf{pred}_i, [\mathbf{y}])$, if $\mathbf{y} \notin \mathsf{span}(\mathbf{M})$,

$$\gamma = [\mathbf{y}^\top(\mathbf{k}_\tau + \mathbf{M}^\perp \mathbf{w})] = [\mathbf{y}^\top \mathbf{k}_\tau + \underbrace{\mathbf{y}^\top \mathbf{M}^\perp}_{\neq \mathbf{0}} \mathbf{w}]$$

will be random due to the randomness of $\mathbf{w}$. In this case,

$$\Pr_2[\mathsf{Bad} \text{ happens in the } i\text{-th query}] = \Pr_{\gamma \leftarrow_\$ \Gamma}[\mathsf{pred}_i(\gamma)].$$

So by (37) and (38), we have

$$|\Pr_1[b' = b] - \Pr_2[b' = b]| \leq \Pr_2[\mathsf{Bad}] \leq \sum_{i=1}^{Q_{\mathrm{dec}}} \Pr_{\gamma \leftarrow_\$ \Gamma}[\mathsf{pred}_i(\gamma)] = Q_{\mathrm{dec}} \cdot \mathit{uncert}_{\mathcal{A}}(\lambda). \tag{40}$$

Game $H_2 - H_3$. H_3 is almost the same as H_2 except for adding one reject rule in $\mathcal{O}_{\mathrm{enc}}$ and one reject rule in $\mathcal{O}_{\mathrm{dec}}$. H_3 initializes two sets $\mathcal{T}_{\mathrm{enc}}$ and $\mathcal{T}_{\mathrm{dec}}$, and use them to store all $\tau^b = \mathsf{H}(\overline{\psi_b})$ used in $\mathcal{O}_{\mathrm{enc}}$ and all $\tau = \mathsf{H}(\overline{\psi})$ used in $\mathcal{O}_{\mathrm{dec}}$, respectively.

- In $\mathcal{O}_{\mathrm{enc}}$, the oracle rejects if $\tau^b \in \mathcal{T}_{\mathrm{enc}} \cup \mathcal{T}_{\mathrm{dec}}$.
- In $\mathcal{O}_{\mathrm{dec}}$, the oracle rejects if $\exists [\mathbf{y}'] \in \boldsymbol{\psi}_{\mathrm{enc}}$ s.t.$\tau = \mathsf{H}(\overline{[\mathbf{y}']}) \wedge \mathbf{y} \neq \mathbf{y}'$.

We will use $\mathsf{Bad}_{\mathrm{enc}}$ and $\mathsf{Bad}_{\mathrm{dec}}$ to denote these two events, respectively. It is straightforward that

$$|\Pr_2[b' = b] - \Pr_3[b' = b]| \leq \Pr_3[\mathsf{Bad}_{\mathrm{enc}} \vee \mathsf{Bad}_{\mathrm{dec}}]. \tag{41}$$

We will show that $\Pr_3[\mathsf{Bad}_{\mathrm{enc}} \vee \mathsf{Bad}_{\mathrm{dec}}] \leq \mathsf{Adv}^{\mathrm{cr}}_{\mathcal{H},\mathcal{B}}(\lambda)$. We construct an adversary $\mathcal{B}$ against the collision resistant property of $\mathcal{H}$ as follows.

On input $(1^\lambda, \mathsf{H})$ where $\mathsf{H} \leftarrow_\$ \mathcal{H}(1^\lambda)$, $\mathcal{B}$ can use H to perfectly simulate game H_3 and detect whether event $\mathsf{Bad}_{\mathrm{enc}}$ or event $\mathsf{Bad}_{\mathrm{dec}}$ happens.

- If $\mathsf{Bad}_{\mathrm{enc}}$ happens, with probability $1 - \frac{Q_{\mathrm{enc}}(Q_{\mathrm{enc}}+Q_{\mathrm{dec}})}{q^k} = 1 - 2^{-\Omega(\lambda)}$, each $\mathcal{O}_{\mathrm{enc}}$ query will sample a ψ_b such that its upper part $\overline{\psi_b}$ is fresh. By "fresh", we mean that this $\overline{\psi_b}$ is distinct from all previous upper parts sampled in $\mathcal{O}_{\mathrm{enc}}$ or submitted to $\mathcal{O}_{\mathrm{dec}}$. The reason is that in the $\mathcal{O}_{\mathrm{enc}}$ of H_3, each $\overline{\psi_b}$ is uniformly random over $\mathsf{span}([\overline{\mathbf{M}}]) = \mathbb{Z}_q^k$. So if $\tau^b = \mathsf{H}(\overline{\psi_b}) \in \mathcal{T}_{\mathrm{enc}} \cup \mathcal{T}_{\mathrm{dec}}$ happens, we found a collision.
- If $\mathsf{Bad}_{\mathrm{dec}}$ happens, i.e., $\tau = \mathsf{H}([\overline{\mathbf{y}}]) = \mathsf{H}([\overline{\mathbf{y}'}])$ for some $\mathbf{y}' \neq \mathbf{y}$ and $\mathbf{y}' \in \boldsymbol{\psi}_{\mathrm{enc}}$, we also find a collision for H. The reason is that $\mathbf{y} \in \mathsf{span}(\mathbf{M})$ (otherwise it is rejected by $\mathcal{O}_{\mathrm{dec}}$), $\mathbf{y}' \in \mathsf{span}(\mathbf{M})$ (since $\boldsymbol{\psi}_{\mathrm{enc}}$ contains ψ_b in $\mathcal{O}_{\mathrm{enc}}$ and they are all in $\mathbf{M}$'s span) and $\mathbf{y}' \neq \mathbf{y}$ can imply $\overline{\mathbf{y}'} \neq \overline{\mathbf{y}}$ (since $\overline{\mathbf{M}}$ is invertible). Thus $\mathsf{H}([\overline{\mathbf{y}}]) = \mathsf{H}([\overline{\mathbf{y}'}])$ implies a collision for H.

Overall, if $\mathsf{Bad}_{\mathrm{enc}} \vee \mathsf{Bad}_{\mathrm{dec}}$ happens, $\mathcal{B}$ finds a collision for H with probability $1 - 2^{-\Omega(\lambda)}$, i.e., $(1-2^{-\Omega(\lambda)})\Pr_3[\mathsf{Bad}_{\mathrm{enc}} \vee \mathsf{Bad}_{\mathrm{dec}}] \leq \mathsf{Adv}^{\mathrm{cr}}_{\mathcal{H},\mathcal{B}}(\lambda)$. Thus $\Pr_3[\mathsf{Bad}_{\mathrm{enc}} \vee \mathsf{Bad}_{\mathrm{dec}}] \leq \mathsf{Adv}^{\mathrm{cr}}_{\mathcal{H},\mathcal{B}}(\lambda) + 2^{-\Omega(\lambda)}$. Together with inequality (41), we have

$$|\Pr_2[b' = b] - \Pr_3[b' = b]| \leq \mathsf{Adv}^{\mathrm{cr}}_{\mathcal{H},\mathcal{B}}(\lambda) + 2^{-\Omega(\lambda)}. \tag{42}$$

Game $H_3 - H_4$**.** H_4 is almost the same as H_3 except for canceling the rejection rule added in H_2, i.e., it does not reject $\mathbf{y} \notin \mathsf{span}(\mathbf{M})$ anymore in $\mathcal{O}_{\mathrm{dec}}$. The analysis for this difference is almost the same as the analysis for the difference between H_1 and H_2, we omit the details and only state the conclusion here.

$$|\Pr_3[b' = b] - \Pr_4[b' = b]| \leq Q_{\mathrm{dec}} \cdot \mathit{uncert}_{\mathcal{A}}(\lambda). \tag{43}$$

Game $H_4 - H_5$**.** H_5 is almost the same as H_4 with only one difference in $\mathcal{O}_{\mathrm{enc}}$.

- In H_4, ψ_0 is randomly selected from $\mathsf{span}([\mathbf{M}])$ in each $\mathcal{O}_{\mathrm{enc}}$ query.
- In H_5, ψ_0 is randomly selected from $\mathbb{G}^{3k}$ in each $\mathcal{O}_{\mathrm{enc}}$ query.

The analysis for this difference is almost the same as the analysis for the difference between H_0 and H_1, we omit the details and only state the conclusion here.

$$|\Pr_4[b' = b] - \Pr_5[b' = b]| \leq \mathsf{Adv}^{\mathrm{mddh}}_{\mathcal{U}_k,\mathsf{GGen},\mathcal{B}}(\lambda) + 2^{-\Omega(\lambda)}. \tag{44}$$

Game $H_5 - H_6$**.** H_6 is almost the same as H_5 with only one difference in $\mathcal{O}_{\mathrm{enc}}$.

- In H_5, ψ_1 is randomly selected from $\mathsf{span}([\mathbf{M}])$ in each $\mathcal{O}_{\mathrm{enc}}$ query.
- In H_6, ψ_1 is randomly selected from $\mathbb{G}^{3k}$ in each $\mathcal{O}_{\mathrm{enc}}$ query.

The analysis for this difference is almost the same as the analysis for the difference between H_0 and H_1, we omit the details and only state the conclusion here.

$$|\Pr_5[b' = b] - \Pr_6[b' = b]| \leq \mathsf{Adv}^{\mathrm{mddh}}_{\mathcal{U}_k,\mathsf{GGen},\mathcal{B}}(\lambda) + 2^{-\Omega(\lambda)}. \tag{45}$$

Game $H_{7.i}$**.** For $i \in \{0, \cdots, \lambda\}$, we define game $H_{7.i}$. In this game, a random function $\mathsf{RF}_i : \{0,1\}^i \to \mathbb{Z}_q^{2k}$ is simulated. For $i = 0$, all $\tau \in \{0,1\}^\lambda$ is mapped to the same random variable $\mathsf{RF}_0(\tau_{|0}) = \mathsf{RF}_0(\varepsilon)$ by RF_0. For $i = \lambda$, each $\tau \in \{0,1\}^\lambda$ is mapped to a distinct random variable $\mathsf{RF}_\lambda(\tau_{|\lambda}) = \mathsf{RF}_\lambda(\tau)$ by RF_λ.

Furthermore, in $H_{7.i}$, an additional term is added when calculating $\mathbf{k}_\tau$ in both $\mathcal{O}_{\mathrm{enc}}$ and $\mathcal{O}_{\mathrm{dec}}$, i.e.,

$$\mathbf{k}_\tau \leftarrow \sum\nolimits_{j=1}^{\lambda} \mathbf{k}_{j,\tau_j} \boxed{+\mathbf{M}^{\perp}\mathsf{RF}_i(\tau_{|i})}.$$

Game $H_6 - H_{7.0}$**.** $H_{7.0}$ is almost the same as H_6 except for adding $\mathbf{M}^{\perp}\mathsf{RF}_0(\varepsilon)$ when calculating $\mathbf{k}_\tau$ in both $\mathcal{O}_{\mathrm{enc}}$ and $\mathcal{O}_{\mathrm{dec}}$. Observe that $\mathbf{k}_{1,\beta} \leftarrow_{\$} \mathbb{Z}_q^{3k}$ are identically distributed as $\mathbf{k}_{1,\beta} + \mathbf{M}^{\perp}\mathsf{RF}_0(\varepsilon)$ for $\beta \in \{0,1\}$, where $\mathbf{k}_{1,\beta} \leftarrow_{\$} \mathbb{Z}_q^{3k}$. So this change is only conceptual and

$$\Pr_6[b' = b] = \Pr_{7.0}[b' = b]. \tag{46}$$

Game $H_{7.0} - H_{7.\lambda}$**.** We will prove that

$$|\Pr_{7.0}[b' = b] - \Pr_{7.\lambda}[b' = b]| \leq 4\lambda\mathsf{Adv}^{\mathrm{mddh}}_{\mathcal{U}_k,\mathsf{GGen},\mathcal{B}}(\lambda) + 4\lambda Q_{\mathrm{dec}} \cdot \mathit{uncert}_{\mathcal{A}}(\lambda) + 2^{-\Omega(\lambda)}. \tag{47}$$

First we prove the following lemma.

```
$\underline{\mathsf{Exp}_{\mathsf{KEM}_{\mathsf{mddh}},\mathcal{A}}(\lambda)\text{:}}$
                    $H_{7.i}$ $\boxed{H_{7.i.1}\ H_{7.i.2}\ H_{7.i.3}}$
$b \leftarrow_{\$} \{0,1\}$
$\mathcal{T}_{\mathrm{enc}}, \mathcal{T}_{\mathrm{dec}} \leftarrow \emptyset$
$\mathbf{M} \leftarrow_{\$} \mathcal{U}_{3k,k}$ $\boxed{\mathbf{M}_0, \mathbf{M}_1 \leftarrow_{\$} \mathcal{U}_{2k,k}}$
$\mathbf{M}^{\perp} \leftarrow_{\$} \mathcal{U}_{3k,2k}$ s.t. $\mathbf{M}^{\top}\mathbf{M}^{\perp} = \mathbf{0}$
$\mathbf{M}_0^*, \mathbf{M}_1^* \leftarrow_{\$} \mathcal{U}_{3k,k}$ with special span
Simulate random functions
$\mathsf{RF}_i : \{0,1\}^i \rightarrow \mathbb{Z}_q^{2k}$
$\mathsf{RF}_i^{(0)}, \mathsf{RF}_i^{(1)} : \{0,1\}^i \rightarrow \mathbb{Z}_q^{k}$
$\mathbf{k}_{1,0}, \cdots, \mathbf{k}_{\lambda,1} \leftarrow_{\$} \mathbb{Z}_q^{3k}$
$\mathsf{pk}_{\mathsf{kem}} \leftarrow \begin{pmatrix} \mathcal{G}, \mathsf{H}, [\mathbf{M}] \\ ([\mathbf{M}^{\top}\mathbf{k}_{j,\beta}])_{0 \le \beta \le 1}^{1 \le j \le \lambda} \end{pmatrix}$
$b' \leftarrow_{\$} \mathcal{A}^{\mathcal{O}_{\mathrm{enc}}(), \mathcal{O}_{\mathrm{dec}}(\cdot,\cdot)}(\mathsf{pk}_{\mathsf{kem}})$
Return $b'$
```

```
$\underline{\mathcal{O}_{\mathrm{enc}}()\text{:}}$
                    $H_{7.i}$ $\boxed{H_{7.i.1}\ H_{7.i.2}\ H_{7.i.3}}$
$(\psi_0, \gamma_0) \leftarrow_{\$} \mathbb{G}^{3k} \times \mathbb{G}$
$\tau^0 \leftarrow \mathsf{H}(\overline{\psi_0}), \mathbf{r} \leftarrow_{\$} \mathbb{Z}_q^k$
$[\overline{\mathbf{y}_1}] \leftarrow [\overline{\mathbf{M}}]\mathbf{r}, \tau^1 \leftarrow \mathsf{H}([\overline{\mathbf{y}_1}])$
If $\tau^b \in \mathcal{T}_{\mathrm{enc}} \cup \mathcal{T}_{\mathrm{dec}}$
    Return $\perp$
$\mathcal{T}_{\mathrm{enc}} \leftarrow \mathcal{T}_{\mathrm{enc}} \cup \{\tau^b\}$
$\mathbf{k}_{\tau^1} \leftarrow \sum_{j=1}^{\lambda} \mathbf{k}_{j,\tau_j^1} + \mathbf{M}^{\perp}\mathsf{RF}_i(\tau_{|i}^1)$
$\mathbf{k}_{\tau^1} \leftarrow \sum_{j=1}^{\lambda} \mathbf{k}_{j,\tau_j^1}$
$+\mathbf{M}_0^*\mathsf{RF}_i^{(0)}(\tau_{|i}^1) + \mathbf{M}_1^*\mathsf{RF}_i^{(1)}(\tau_{|i}^1)$
If $\tau_{i+1}^1 = 0$:
    $[\underline{\mathbf{y}_1}] \leftarrow_{\$} \mathbb{G}^{2k}$
    $\boxed{\mathbf{r}_0 \leftarrow_{\$} \mathbb{Z}_q^k, [\underline{\mathbf{y}_1}] \leftarrow [\underline{\mathbf{M}}\mathbf{r} + \mathbf{M}_0\mathbf{r}_0]}$
Else: $//\tau_{i+1}^1 = 1$
    $[\underline{\mathbf{y}_1}] \leftarrow_{\$} \mathbb{G}^{2k}$
    $\mathbf{r}_1 \leftarrow_{\$} \mathbb{Z}_q^k, [\underline{\mathbf{y}_1}] \leftarrow [\underline{\mathbf{M}}\mathbf{r} + \mathbf{M}_1\mathbf{r}_1]$
$\psi_1 \leftarrow \begin{bmatrix} \overline{\mathbf{y}_1} \\ \underline{\mathbf{y}_1} \end{bmatrix}, \gamma_1 \leftarrow [\mathbf{y}_1^{\top}]\mathbf{k}_{\tau^1}$
$\boldsymbol{\psi}_{\mathrm{enc}} \leftarrow \boldsymbol{\psi}_{\mathrm{enc}} \cup \{\psi_b\}$
Return $(\psi_b, \gamma_b)$
```

```
$\underline{\mathcal{O}_{\mathrm{dec}}(\mathsf{pred}, \psi = [\mathbf{y}])\text{:}}$
                    $H_{7.i}, H_{7.i.1}, H_{7.i.2}$ $H_{7.i.3}$
If $\psi \in \boldsymbol{\psi}_{\mathrm{enc}}$:
    Return $\perp$
$\tau \leftarrow \mathsf{H}([\overline{\mathbf{y}}])$
If $\begin{bmatrix} \exists [\mathbf{y}'] \in \boldsymbol{\psi}_{\mathrm{enc}} \text{ s.t.} \\ \tau = \mathsf{H}([\overline{\mathbf{y}'}]) \wedge \mathbf{y} \neq \mathbf{y}' \end{bmatrix}$
    Return $\perp$
$\mathcal{T}_{\mathrm{dec}} \leftarrow \mathcal{T}_{\mathrm{dec}} \cup \{\tau\}$
$\mathbf{k}_{\tau} \leftarrow \sum_{j=1}^{\lambda} \mathbf{k}_{j,\tau_j} + \mathbf{M}^{\perp}\mathsf{RF}_i(\tau_{|i})$
$\mathbf{k}_{\tau} \leftarrow \sum_{j=1}^{\lambda} \mathbf{k}_{j,\tau_j}$
$+\mathbf{M}_0^*\mathsf{RF}_i^{(0)}(\tau_{|i}) + \mathbf{M}_1^*\mathsf{RF}_i^{(1)}(\tau_{|i})$
$\gamma \leftarrow [\mathbf{y}^{\top}]\mathbf{k}_{\tau}$
If $\mathsf{pred}(\gamma) = 0$:
    Return $\perp$
Return $\gamma$
```

Fig. 15. Games $H_{7.i}, H_{7.i.1}, H_{7.i.2}$ and $H_{7.i.3}$.

Lemma 7 **($H_{7.i} - H_{7.i+1}$).** *Let Q_{dec} be the total number of $\mathcal{O}_{\mathrm{dec}}$ queries submitted by $\mathcal{A}$. Then, for all $i \in \{0, \cdots, \lambda - 1\}$,*

$$|\Pr_{7.i}[b' = b] - \Pr_{7.i+1}[b' = b]| \le 4\mathsf{Adv}_{\mathcal{U}_k,\mathsf{GGen},\mathcal{B}}^{\mathrm{mddh}}(\lambda) + 4Q_{\mathrm{dec}} \cdot \mathit{uncert}_{\mathcal{A}}(\lambda) + 2^{-\Omega(\lambda)}.$$

Then (47) follows from Lemma 7 since there are λ hops between $H_{7.0}$ and $H_{7.\lambda}$.
Proof of Lemma 7. We first rewrite game $H_{7.i}$ and define new games $H_{7.i.1} - H_{7.i.3}$ in Figure 15. We make some change in oracle $\mathcal{O}_{\mathrm{enc}}$ and game $H_{7.i}$ in Figure 15 appears to be different from the one in Figure 14. In Figure 15, we first select $[\overline{\mathbf{y}_1}]$ randomly from $\mathsf{span}([\mathbf{M}])$ and calculate $\tau^1 = \mathsf{H}([\overline{\mathbf{H}_1}])$. Then we select $[\underline{\mathbf{y}_1}]$ randomly from $\mathbb{G}^{2k}$. Since $\overline{\mathbf{M}}$ is invertible, we have that $[\overline{\mathbf{y}_1}]$ is uniform over $\mathbb{G}^k$. So $[\mathbf{y}_1] = \begin{bmatrix} \overline{\mathbf{y}_1} \\ \underline{\mathbf{y}_1} \end{bmatrix}$ is uniform over $\mathbb{G}^{3k}$ and the oracles $\mathcal{O}_{\mathrm{enc}}$ in these two figures are actually the same.

Game $\underline{H_{7.i} - H_{7.i.1}}$. $H_{7.i.1}$ is almost the same as $H_{7.i}$ except for changing how $[\underline{\mathbf{y}_1}]$ is generated in $\mathcal{O}_{\mathrm{enc}}$.

- In $H_{7.i}$, $[\underline{\mathbf{y}_1}]$ is uniform over $\mathbb{G}^{2k}$.
- In $H_{7.i.1}$, when $\tau_{i+1}^1 = 0$, $\mathbf{r}_0 \leftarrow_{\$} \mathbb{Z}_q^k$ is selected and $[\underline{\mathbf{y}_1}]$ is set to $[\underline{\mathbf{M}}\mathbf{r} + \mathbf{M}_0\mathbf{r}_0]$ for some $\mathbf{M}_0 \leftarrow_{\$} \mathcal{U}_{2k,k}$.

We can build an adversary $\mathcal{B}$ and show that

$$|\Pr_{7.i}[b' = b] - \Pr_{7.i.1}[b' = b]| \le \mathsf{Adv}_{\mathcal{U}_k,\mathsf{GGen},\mathcal{B}}^{\mathrm{mddh}}(\lambda) + 2^{-\Omega(\lambda)}. \tag{48}$$

Let Q_{enc} be the total number of $\mathcal{O}_{\text{enc}}$ queries submitted by $\mathcal{A}$. To prove (48), we construct an adversary $\mathcal{B}'$ and show that

$$|\Pr_{7.i}[b' = b] - \Pr_{7.i.1}[b' = b]| \leq \mathsf{Adv}^{Q_{\text{enc}}\text{-mddh}}_{\mathcal{U}_{2k,k},\mathsf{GGen},\mathcal{B}'}(\lambda). \tag{49}$$

Upon receiving a challenge $(\mathcal{G}, [\mathbf{M}_0] \in \mathbb{G}^{2k\times k}, [\mathbf{H}] := ([\mathbf{h}_1|\cdots|\mathbf{h}_{Q_{\text{enc}}}]) \in \mathbb{G}^{2k\times Q_{\text{enc}}})$ for the Q_{enc}-fold $\mathcal{U}_{2k,k}$-MDDH problem, $\mathcal{B}'$ simulates game $H_{7.i}(H_{7.i.1})$. To reply the i-th $\mathcal{O}_{\text{enc}}$ query made by $\mathcal{A}$, $\mathcal{B}'$ embeds $[\mathbf{h}_i]$ to $[\underline{\mathbf{y}_1}]$ if $\tau^1_{i+1} = 0$, i.e., $[\underline{\mathbf{y}_1}] \leftarrow [\mathbf{h}_i] + [\underline{\mathbf{M}}]\mathbf{r}$. Finally $\mathcal{B}'$ gets $\mathcal{A}$'s output b' and outputs $1 \Leftrightarrow (b' = b)$. Thus, if each column $[\mathbf{h}_i]$ of $[\mathbf{H}]$ is uniformly random over $\mathbb{G}^{3k}$, $\mathcal{B}'$ perfectly simulates $H_{7.i}$ (since if $\tau^1_{i+1} = 0$, $[\underline{\mathbf{y}_1}] = [\mathbf{h}_i] + [\underline{\mathbf{M}}]\mathbf{r}$ is uniform). If each column $[\mathbf{h}_i]$ of $[\mathbf{H}]$ is uniformly random over $\mathsf{span}([\mathbf{M}])$, $\mathcal{B}'$ perfectly simulates $H_{7.i.1}$. So (49) follows.

Finally (48) follows from (49), Lemma 6 and Lemma 3.

Game $H_{7.i.1} - H_{7.i.2}$. $H_{7.i.2}$ is almost the same as $H_{7.i.1}$ except for changing how $[\underline{\mathbf{y}_1}]$ is generated in $\mathcal{O}_{\text{enc}}$.

- In $H_{7.i.1}$, when $\tau^1_{i+1} = 1$, $[\underline{\mathbf{y}_1}]$ is uniform over $\mathbb{G}^{2k}$.
- In $H_{7.i.2}$, when $\tau^1_{i+1} = 1$, $\mathbf{r}_1 \leftarrow_\$ \mathbb{Z}_q^k$ is selected and $[\underline{\mathbf{y}_1}]$ is set to $[\underline{\mathbf{M}}\mathbf{r} + \mathbf{M}_1\mathbf{r}_1]$ for some $\mathbf{M}_1 \leftarrow_\$ \mathcal{U}_{2k,k}$.

We can build an adversary $\mathcal{B}$ and show that

$$|\Pr_{7.i.1}[b' = b] - \Pr_{7.i.2}[b' = b]| \leq \mathsf{Adv}^{\text{mddh}}_{\mathcal{U}_k,\mathsf{GGen},\mathcal{B}}(\lambda) + \frac{1}{q-1} = \mathsf{Adv}^{\text{mddh}}_{\mathcal{U}_k,\mathsf{GGen},\mathcal{B}}(\lambda) + 2^{-\Omega(\lambda)}. \tag{50}$$

The proof idea is almost the same as the one used in proving (48). We omit the proof details.

Game $H_{7.i.2} - H_{7.i.3}$. We first specify how $\mathbf{M}_0^*$ and $\mathbf{M}_1^*$ are selected. Note that with probability at least $1 - \frac{2k}{q} = 1 - 2^{-\Omega(\lambda)}$ over the randomness of $\mathbf{M}_0$ and $\mathbf{M}_1$, $\begin{pmatrix} \overline{\mathbf{M}} & \mathbf{0} & \mathbf{0} \\ \underline{\mathbf{M}} & \mathbf{M}_0 & \mathbf{M}_1 \end{pmatrix}$ forms an invertible matrix over $\mathbb{Z}_q^{3k\times 3k}$. Therefore, $\mathsf{Ker}(\mathbf{M}^\top) = \mathsf{span}(\mathbf{M}^\perp) = \mathsf{Ker}\left(\begin{pmatrix} \overline{\mathbf{M}} & \mathbf{0} \\ \underline{\mathbf{M}} & \mathbf{M}_0 \end{pmatrix}^\top\right) \oplus \mathsf{Ker}\left(\begin{pmatrix} \overline{\mathbf{M}} & \mathbf{0} \\ \underline{\mathbf{M}} & \mathbf{M}_1 \end{pmatrix}^\top\right)$. We can select $\mathbf{M}_0^*, \mathbf{M}_1^* \in \mathbb{Z}_q^{3k\times k}$ such that $\mathsf{span}(\mathbf{M}_0^*) = \mathsf{Ker}\left(\begin{pmatrix} \overline{\mathbf{M}} & \mathbf{0} \\ \underline{\mathbf{M}} & \mathbf{M}_1 \end{pmatrix}^\top\right)$ and $\mathsf{span}(\mathbf{M}_1^*) = \mathsf{Ker}\left(\begin{pmatrix} \overline{\mathbf{M}} & \mathbf{0} \\ \underline{\mathbf{M}} & \mathbf{M}_0 \end{pmatrix}^\top\right)$. Thus we have $\mathsf{span}(\mathbf{M}^\perp) = \mathsf{span}(\mathbf{M}_1^*) \oplus \mathsf{span}(\mathbf{M}_0^*)$

In this case, for all $\tau \in \{0,1\}^\lambda$, we can replace $\mathbf{M}^\perp\mathsf{RF}_i(\tau_{|i})$ (used in $H_{7.i.2}$) to $\mathbf{M}_0^*\mathsf{RF}_i^{(0)}(\tau_{|i}) + \mathbf{M}_1^*\mathsf{RF}_i^{(1)}(\tau_{|i})$ (used in $H_{7.i.3}$) where $\mathsf{RF}_i^{(0)}$ and $\mathsf{RF}_i^{(1)}$ are two independent random function from $\{0,1\}^i$ to $\mathbb{Z}_q^k$. So with probability at least $1 - 2^{-\Omega(\lambda)}$, game $H_{7.i.3}$ is almost the same with $H_{7.i.2}$ and

$$|\Pr_{7.i.2}[b' = b] - \Pr_{7.i.3}[b' = b]| \leq 2^{-\Omega(\lambda)}. \tag{51}$$

We then rewrite game $H_{7.i.3}$ and define new games $H_{7.i.4} - H_{7.i.7}$ in Figure 16.

Game $H_{7.i.3} - H_{7.i.4}$. $H_{7.i.4}$ is almost the same as $H_{7.i.3}$ except for replacing the random function $\mathsf{RF}_i^{(0)} : \{0,1\}^i \to \mathbb{Z}_q^k$ with $\mathsf{RF}_{i+1}^{(0)} : \{0,1\}^{i+1} \to \mathbb{Z}_q^k$.

```
$\underline{\mathsf{Exp}_{\mathsf{KEM_{mddh}},\mathcal{A}}(\lambda)}$:
$H_{7.i.3}$ $H_{7.i.4}$ $\boxed{H_{7.i.5}\ H_{7.i.6}, H_{7.i.7} = H_{7.i+1}}$
$b \leftarrow_\$ \{0,1\}$
$\mathcal{T}_{\mathrm{enc}}, \mathcal{T}_{\mathrm{dec}} \leftarrow \emptyset$
$\mathbf{M} \leftarrow_\$ \mathcal{U}_{3k,k}, \mathbf{M}_0, \mathbf{M}_1 \leftarrow_\$ \mathcal{U}_{2k,k}$
$\mathbf{M}^\perp \leftarrow_\$ \mathcal{U}_{3k,2k}$ s.t. $\mathbf{M}^\top \mathbf{M}^\perp = \mathbf{0}$
$\mathbf{M}_0^*, \mathbf{M}_1^* \leftarrow_\$ \mathcal{U}_{3k,k}$ with special span
Simulate random functions
$\mathsf{RF}^{(0)}_{i+1} : \{0,1\}^{i+1} \rightarrow \mathbb{Z}_q^k$
$\mathsf{RF}^{(1)}_{i\boxed{+1}} : \{0,1\}^{i\boxed{+1}} \rightarrow \mathbb{Z}_q^k$
$\mathsf{RF}_{i+1} : \{0,1\}^{i+1} \rightarrow \mathbb{Z}_q^{2k}$
$\mathbf{k}_{1,0}, \cdots, \mathbf{k}_{\lambda,1} \leftarrow_\$ \mathbb{Z}_q^{3k}$
$\mathsf{pk}_{\mathsf{kem}} \leftarrow \begin{pmatrix} \mathcal{G}, \mathsf{H}, [\mathbf{M}] \\ ([\mathbf{M}^\top \mathbf{k}_{j,\beta}])_{0 \le \beta \le 1}^{1 \le j \le \lambda} \end{pmatrix}$
$b' \leftarrow_\$ \mathcal{A}^{\mathcal{O}_{\mathrm{enc}}(), \mathcal{O}_{\mathrm{dec}}(\cdot,\cdot)}(\mathsf{pk}_{\mathsf{kem}})$
Return $b'$
```

```
$\underline{\mathcal{O}_{\mathrm{enc}}()}$:
$H_{7.i.3}$ $H_{7.i.4}$ $\boxed{H_{7.i.5}\ H_{7.i.6}\ \boxed{H_{7.i.7} = H_{7.i+1}}}$
$(\psi_0, \gamma_0) \leftarrow_\$ \mathbb{G}^{3k} \times \mathbb{G}$
$\tau^0 \leftarrow \mathsf{H}(\overline{\psi_0}), \mathbf{r} \leftarrow_\$ \mathbb{Z}_q^k$
$[\overline{\mathbf{y}_1}] \leftarrow [\overline{\mathbf{M}}]\mathbf{r}, \tau^1 \leftarrow \mathsf{H}([\overline{\mathbf{y}_1}])$
If $\tau^b \in \mathcal{T}_{\mathrm{enc}} \cup \mathcal{T}_{\mathrm{dec}}$
    Return $\perp$
$\mathcal{T}_{\mathrm{enc}} \leftarrow \mathcal{T}_{\mathrm{enc}} \cup \{\tau^b\}$
$\mathbf{k}_{\tau^1} \leftarrow \sum_{j=1}^{\lambda} \mathbf{k}_{j,\tau^1_j}$
$+\mathbf{M}_0^* \mathsf{RF}^{(0)}_{i+1}(\tau^1_{|i+1}) + \mathbf{M}_1^* \mathsf{RF}^{(1)}_{i\boxed{+1}}(\tau^1_{|i\boxed{+1}})$
$\mathbf{k}_{\tau^1} \leftarrow \sum_{j=1}^{\lambda} \mathbf{k}_{j,\tau^1_j} + \mathbf{M}^\perp \mathsf{RF}_{i+1}(\tau^1_{|i+1})$
If $\tau^1_{i+1} = 0$:
    $\mathbf{r}_0 \leftarrow_\$ \mathbb{Z}_q^k, [\underline{\mathbf{y}_1}] \leftarrow [\underline{\mathbf{M}}\mathbf{r} + \mathbf{M}_0 \mathbf{r}_0]$
Else: $//\tau^1_{i+1} = 1$
    $\mathbf{r}_1 \leftarrow_\$ \mathbb{Z}_q^k, [\underline{\mathbf{y}_1}] \leftarrow [\underline{\mathbf{M}}\mathbf{r} + \mathbf{M}_1 \mathbf{r}_1]$
$\boxed{[\underline{\mathbf{y}_1}] \leftarrow_\$ \mathbb{G}^{2k}}$
$\psi_1 \leftarrow \begin{bmatrix} \overline{\mathbf{y}_1} \\ \underline{\mathbf{y}_1} \end{bmatrix}, \gamma_1 \leftarrow [\mathbf{y}_1^\top]\mathbf{k}_{\tau^1}$
$\boldsymbol{\psi}_{\mathrm{enc}} \leftarrow \boldsymbol{\psi}_{\mathrm{enc}} \cup \{\psi_b\}$
Return $(\psi_b, \gamma_b)$
```

```
$\underline{\mathcal{O}_{\mathrm{dec}}(\mathsf{pred}, \psi = [\mathbf{y}])}$:
$H_{7.i.3}$ $H_{7.i.4}$ $\boxed{H_{7.i.5}\ H_{7.i.6}, H_{7.i.7} = H_{7.i+1}}$
If $\psi \in \boldsymbol{\psi}_{\mathrm{enc}}$:
    Return $\perp$
$\tau \leftarrow \mathsf{H}([\overline{\mathbf{y}}])$
If $\begin{bmatrix} \exists [\mathbf{y}'] \in \boldsymbol{\psi}_{\mathrm{enc}} \text{ s.t.} \\ \tau = \mathsf{H}([\overline{\mathbf{y}'}]) \wedge \mathbf{y} \neq \mathbf{y}' \end{bmatrix}$
    Return $\perp$
$\mathcal{T}_{\mathrm{dec}} \leftarrow \mathcal{T}_{\mathrm{dec}} \cup \{\tau\}$
$\mathbf{k}_\tau \leftarrow \sum_{j=1}^{\lambda} \mathbf{k}_{j,\tau_j}$
$+\mathbf{M}_0^* \mathsf{RF}^{(0)}_{i+1}(\tau_{|i+1}) + \mathbf{M}_1^* \mathsf{RF}^{(1)}_{i\boxed{+1}}(\tau_{|i\boxed{+1}})$
$\mathbf{k}_\tau \leftarrow \sum_{j=1}^{\lambda} \mathbf{k}_{j,\tau_j} + \mathbf{M}^\perp \mathsf{RF}_{i+1}(\tau_{|i+1})$
$\gamma \leftarrow [\mathbf{y}^\top]\mathbf{k}_\tau$
If $\mathsf{pred}(\gamma) = 0$:
    Return $\perp$
Return $\gamma$
```

Fig. 16. Games $H_{7.i.3} - H_{7.i.7}$.

Consider the following function $\mathsf{RF}^{(0)}_{i+1}$,

$$\mathsf{RF}^{(0)}_{i+1}(\tau_{|i+1}) = \begin{cases} \mathsf{RF}^{(0)}_i(\tau_{|i}) & \text{If } \tau_{i+1} = 0 \\ \mathsf{RF}^{(0)}_i(\tau_{|i}) + \mathsf{RF}'^{(0)}_i(\tau_{|i}) & \text{If } \tau_{i+1} = 1 \end{cases}$$

This is indeed a random function with $i+1$ bits input if $\mathsf{RF}'^{(0)}_i : \{0,1\}^i \rightarrow \mathbb{Z}_q^k$ is an independent random function. If we use this function in $H_{7.i.4}$, then for all $\tau \in \{0,1\}^\lambda$ such that $\tau_{i+1} = 0$, $\mathsf{RF}^{(0)}_{i+1}(\tau_{|i+1}) = \mathsf{RF}^{(0)}_i(\tau_{|i})$ and $H_{7.i.4}$ will be the same with $H_{7.i.3}$ in such cases.

Observe that for all $\tau \in \{0,1\}^\lambda$ such that $\tau_{i+1} = 1$ and all $\mathbf{y} \in \mathsf{span}\begin{pmatrix} \overline{\mathbf{M}} & \mathbf{0} \\ \underline{\mathbf{M}} & \mathbf{M}_1 \end{pmatrix}$

$$\begin{aligned} &\overbrace{\mathbf{y}^\top(\sum_{j=1}^{\lambda} \mathbf{k}_{j,\tau_j} + \mathbf{M}_0^* \mathsf{RF}^{(0)}_i(\tau_{|i}) \boxed{+\mathbf{M}_0^* \mathsf{RF}'^{(0)}_i(\tau_{|i})} + \mathbf{M}_1^* \mathsf{RF}^{(1)}_i(\tau_{|i}))}^{H_{7.i.4}} \\ =&\underbrace{\mathbf{y}^\top(\sum_{j=1}^{\lambda} \mathbf{k}_{j,\tau_j} + \mathbf{M}_0^* \mathsf{RF}^{(0)}_i(\tau_{|i}) + \mathbf{M}_1^* \mathsf{RF}^{(1)}_i(\tau_{|i}))}_{H_{7.i.3}} \end{aligned} , \qquad (52)$$

since $\mathsf{span}(\mathbf{M}_0^*) = \mathsf{Ker}\left(\begin{pmatrix} \overline{\mathbf{M}} & \mathbf{0} \\ \underline{\mathbf{M}} & \mathbf{M}_1 \end{pmatrix}^\top\right)$ and $\mathbf{y}^\top \mathbf{M}_0^* = \mathbf{0}$.

Then we have

- $\mathcal{O}_{\mathrm{enc}}$ will be almost the same in $H_{7.i.4}$ and $H_{7.i.3}$. Since when $\tau^1_{i+1} = 1$, $\mathbf{y}_1 \in \mathsf{span}\begin{pmatrix} \overline{\mathbf{M}} & \mathbf{0} \\ \underline{\mathbf{M}} & \mathbf{M}_1 \end{pmatrix}$.

- $\mathcal{O}_{\text{dec}}(\mathsf{pred}, [\mathbf{y}])$ will be almost the same in $H_{7.i.4}$ and $H_{7.i.3}$ for any $\mathbf{y} \in \mathsf{span}\begin{pmatrix}\overline{\mathbf{M}} & \mathbf{0}\\ \underline{\mathbf{M}} & \mathbf{M}_1\end{pmatrix}$ with $\tau_{i+1} = 1$.

Thus, game $H_{7.i.4}$ differs from $H_{7.i.3}$ only if $\mathcal{A}$ submits some $\mathcal{O}_{\text{dec}}(\mathsf{pred}, [\mathbf{y}])$ query such that $\mathbf{y} \notin \mathsf{span}\begin{pmatrix}\overline{\mathbf{M}} & \mathbf{0}\\ \underline{\mathbf{M}} & \mathbf{M}_1\end{pmatrix}$ but $\tau_{i+1} = 1$. We will call such a query an "*ill-formed*" query.

To show that "*ill-formed*" queries are rejected overwhelmingly in both games, we define two intermediate games $H_{7.i.3'/4'}$. These two games are almost the same with $H_{7.i.3/4}$ except for explicitly reject all ill-formed $\mathcal{O}_{\text{dec}}$ queries. According to the analysis above, we have that

$$\Pr_{7.i.3'}[b' = b] = \Pr_{7.i.4'}[b' = b]. \tag{53}$$

We will prove that

$$|\Pr_{7.i.3}[b' = b] - \Pr_{7.i.3'}[b' = b]| \le Q_{\text{dec}} \cdot \mathit{uncert}_{\mathcal{A}}(\lambda), \tag{54}$$

and

$$|\Pr_{7.i.4}[b' = b] - \Pr_{7.i.4'}[b' = b]| \le Q_{\text{dec}} \cdot \mathit{uncert}_{\mathcal{A}}(\lambda). \tag{55}$$

To prove (54), we define the event Bad as the adversary $\mathcal{A}$ submits a $\mathcal{O}_{\text{dec}}(\mathsf{pred}, [\mathbf{y}])$ such that 1). $[\mathbf{y}] \notin \boldsymbol{\psi}_{\text{enc}}$; 2). $\tau_{i+1} = 1$ for $\tau = \mathsf{H}([\overline{\mathbf{y}}])$; 3). $\mathbf{y} \notin \mathsf{span}\begin{pmatrix}\overline{\mathbf{M}} & \mathbf{0}\\ \underline{\mathbf{M}} & \mathbf{M}_1\end{pmatrix}$ and 4). $\mathsf{pred}([\mathbf{y}^\top]\mathbf{k}_\tau) = 1$ where $\mathbf{k}_\tau = \sum_{j=1}^{\lambda} \mathbf{k}_{j,\tau_j} + \mathbf{M}_0^* \mathsf{RF}_i^{(0)}(\tau_{|i}) + \mathbf{M}_1^* \mathsf{RF}_i^{(1)}(\tau_{|i})$. It is straightforward that game $H_{7.i.3}$ will be almost the same with game $H_{7.i.3'}$ if event Bad does not happen. So we have

$$|\Pr_{7.i.3}[b' = b] - \Pr_{7.i.3'}[b' = b]| \le \Pr_{7.i.3}[\mathsf{Bad}] = \Pr_{7.i.3'}[\mathsf{Bad}]. \tag{56}$$

Let Q_{dec} be the total number of decryption queries submitted by $\mathcal{A}$, we have that

$$\Pr_{7.i.3'}[\mathsf{Bad}] \le \sum_{i=1}^{Q_{\text{dec}}} \Pr_{7.i.3'}[\mathsf{Bad}\text{ happens in the }i\text{-th }\mathcal{O}_{\text{dec}}\text{ query}]. \tag{57}$$

So we will fix some $i \in [Q_{\text{dec}}]$ and consider the i-th $\mathcal{O}_{\text{dec}}(\mathsf{pred}_i, [\mathbf{y}])$ query submitted by $\mathcal{A}$ in game $H_{7.i.3'}$. We will show that Bad will not happen overwhelmingly in this $\mathcal{O}_{\text{dec}}$ query.

We use the fact that $\mathbf{k}_{i+1,1}$ contains some entropy that is hidden from $\mathcal{A}$. More precisely, we use the fact that $\mathbf{k}_{i+1,1} \leftarrow_{\$} \mathbb{Z}_q^{3k}$ is identically distributed with $\mathbf{k}_{i+1,1} + \mathbf{M}_0^*\mathbf{w}$ where $\mathbf{k}_{i+1,1} \leftarrow_{\$} \mathbb{Z}_q^{3k}$ and $\mathbf{w} \leftarrow_{\$} \mathbb{Z}_q^{k}$. We will show that in game $H_{7.i.3'}$, $\mathbf{w}$ is hidden from $\mathcal{A}$ until the i-th $\mathcal{O}_{\text{dec}}$ query.

- $\mathsf{pk}_{\mathsf{KEM}}$ does not leak any information about $\mathbf{w}$. Since $\mathbf{M}^\top(\mathbf{k}_{i+1,1} + \mathbf{M}_0^*\mathbf{w}) = \mathbf{M}^\top\mathbf{k}_{i+1,1}$. This is due to the fact that $\mathsf{span}(\mathbf{M}_0^*) = \mathsf{Ker}\left(\begin{pmatrix}\overline{\mathbf{M}} & \mathbf{0}\\ \underline{\mathbf{M}} & \mathbf{M}_1\end{pmatrix}^\top\right) \subset \mathsf{Ker}(\mathbf{M}^\top)$.
- $\mathcal{O}_{\text{enc}}$ oracle does not leak any information about $\mathbf{w}$. This is because in the $\mathcal{O}_{\text{enc}}$ oracle of game $H_{7.i.3'}$, $\mathbf{w}$ is used only in the generation of γ_1 when $\tau_{i+1}^1 = 1$. In such cases, since $\mathbf{y}_1 \in \mathsf{span}\begin{pmatrix}\overline{\mathbf{M}} & \mathbf{0}\\ \underline{\mathbf{M}} & \mathbf{M}_1\end{pmatrix}$, $\gamma_1 = [\mathbf{y}_1^\top(\mathbf{k}_{i+1,1} + \mathbf{M}_0^*\mathbf{w}) + \cdots] = [\mathbf{y}_1^\top\mathbf{k}_{i+1,1} + \mathbf{y}_1^\top\mathbf{M}_0^*\mathbf{w} + \cdots] = [\mathbf{y}_1^\top\mathbf{k}_{i+1,1} + \cdots]$.

- $\mathcal{O}_{\text{dec}}$ oracle does not leak any information about $\mathbf{w}$. This is because in the $\mathcal{O}_{\text{dec}}$ oracle of game $H_{7.i.3'}$, $\mathbf{w}$ is used only in the generation of γ when $\tau_{i+1} = 1$. Similarly, if $\mathbf{y} \in \mathsf{span}\begin{pmatrix}\mathbf{M} & \mathbf{0} \\ \underline{\mathbf{M}} & \mathbf{M}_1\end{pmatrix}$, γ does not contain any information about $\mathbf{w}$. So $\mathbf{w}$ might be used only when $\mathbf{y} \notin \mathsf{span}\begin{pmatrix}\mathbf{M} & \mathbf{0} \\ \underline{\mathbf{M}} & \mathbf{M}_1\end{pmatrix}$ and $\tau_{i+1} = 1$. However, such case satisfies the definition of ill-formed query and will be rejected directly in game $H_{7.i.3'}$, so the response is independent of $\mathbf{w}$.

Therefore, $\mathbf{w}$ is not leaked to $\mathcal{A}$ until the i-th $\mathcal{O}_{\text{dec}}(\mathsf{pred}_i, [\mathbf{y}])$ query.

So, if $\tau_{i+1} = 1$ and $\mathbf{y} \notin \mathsf{span}\begin{pmatrix}\mathbf{M} & \mathbf{0} \\ \underline{\mathbf{M}} & \mathbf{M}_1\end{pmatrix}$,

$$\begin{aligned}
\mathsf{pred}_i(\gamma) &= \mathsf{pred}_i([\mathbf{y}^\top \mathbf{k}_\tau]) \\
&= \mathsf{pred}_i([\mathbf{y}^\top (\sum_{j=1}^{\lambda} \mathbf{k}_{j,\tau_j} + \mathbf{M}_0^* \mathsf{RF}_i^{(0)}(\tau_{|i}) + \mathbf{M}_1^* \mathsf{RF}_i^{(1)}(\tau_{|i}))]) \\
&= \mathsf{pred}_i([\mathbf{y}^\top (\mathbf{k}_{i+1,1} + \mathbf{M}_0^* \mathbf{w}) + \mathbf{y}^\top (\sum_{j \neq i+1} \mathbf{k}_{j,\tau_j} + \mathbf{M}_0^* \mathsf{RF}_i^{(0)}(\tau_{|i}) + \mathbf{M}_1^* \mathsf{RF}_i^{(1)}(\tau_{|i}))]) \\
&= \mathsf{pred}_i([\underbrace{\mathbf{y}^\top \mathbf{M}_0^*}_{\neq \mathbf{0}} \mathbf{w} + \mathbf{y}^\top (\sum_{j=1}^{\lambda} \mathbf{k}_{j,\tau_j} + \mathbf{M}_0^* \mathsf{RF}_i^{(0)}(\tau_{|i}) + \mathbf{M}_1^* \mathsf{RF}_i^{(1)}(\tau_{|i}))]).
\end{aligned}$$

Since $\mathbf{y}^\top \mathbf{M}_0^* \neq \mathbf{0}$, the input of pred_i is uniform due to the fresh randomness of $\mathbf{w}$. Thus we have

$$\Pr_{7.i.3'}[\mathsf{Bad}\text{ happens in the } i\text{-th } \mathcal{O}_{\text{dec}} \text{ query}] = \Pr_{\gamma \leftarrow_\$ \Gamma}[\mathsf{pred}_i(\gamma) = 1]. \tag{58}$$

Thus (54) follows from (58), (57) and (56). Similarly, we can prove (55).

Combining (53), (54) and (55), we have that

$$|\Pr_{7.i.3}[b' = b] - \Pr_{7.i.4}[b' = b]| \leq 2Q_{\text{dec}} \cdot \mathit{uncert}_{\mathcal{A}}(\lambda). \tag{59}$$

Game $H_{7.i.4} - H_{7.i.5}$. $H_{7.i.5}$ is almost the same as $H_{7.i.4}$ except for replacing the random function $\mathsf{RF}_i^{(1)} : \{0,1\}^i \to \mathbb{Z}_q^k$ with $\mathsf{RF}_{i+1}^{(1)} : \{0,1\}^{i+1} \to \mathbb{Z}_q^k$. We have that

$$|\Pr_{7.i.4}[b' = b] - \Pr_{7.i.5}[b' = b]| \leq 2Q_{\text{dec}} \cdot \mathit{uncert}_{\mathcal{A}}(\lambda). \tag{60}$$

The proof of (60) is similar to the one of (59). Consider the following function $\mathsf{RF}_{i+1}^{(1)}$,

$$\mathsf{RF}_{i+1}^{(1)}(\tau_{|i+1}) = \begin{cases} \mathsf{RF}_i^{(1)}(\tau_{|i}) + \mathsf{RF}_i'^{(1)}(\tau_{|i}) & \text{If } \tau_{i+1} = 0 \\ \mathsf{RF}_i^{(1)}(\tau_{|i}) & \text{If } \tau_{i+1} = 1 \end{cases}$$

This is indeed a random function with $i+1$ bits input if $\mathsf{RF}_i'^{(1)} : \{0,1\}^i \to \mathbb{Z}_q^k$ is an independent random function. The rest of the proof is also symmetric. Define the ill-formed query as a $\mathcal{O}_{\text{dec}}(\mathsf{pred}, [\mathbf{y}])$ query such that $\mathbf{y} \notin \mathsf{span}\begin{pmatrix}\mathbf{M} & \mathbf{0} \\ \underline{\mathbf{M}} & \mathbf{M}_0\end{pmatrix}$ but $\tau_{i+1} = 0$. When analyze the probability that event Bad happens, use the entropy in $\mathbf{k}_{i+1,0}$, i.e., $\mathbf{k}_{i+1,0}$ distributes identically to $\mathbf{k}_{i+1,0} + \mathbf{M}_1^* \mathbf{w}$ and show $\mathbf{w}$ is not leaked at all. We omit the proof details here.

Game $H_{7.i.5} - H_{7.i.6}$. $H_{7.i.6}$ is almost the same as $H_{7.i.5}$ except for replacing $\mathbf{M}_0^* \mathsf{RF}_{i+1}^{(0)}(\tau_{|i+1})+ \mathbf{M}_1^* \mathsf{RF}_{i+1}^{(1)}(\tau_{|i+1})$ with $\mathbf{M}^{\perp} \mathsf{RF}_{i+1}(\tau_{|i+1})$ for an independent random function $\mathsf{RF}_{i+1} : \{0,1\}^{i+1} \rightarrow \mathbb{Z}_q^{2k}$. Similar to the analysis of (51), we have that

$$|\Pr_{7.i.5}[b' = b] - \Pr_{7.i.6}[b' = b]| \leq 2^{-\Omega(\lambda)}. \tag{61}$$

We omit the detailed analysis here.

Game $H_{7.i.6} - H_{7.i.7}$. $H_{7.i.7}$ is almost the same as $H_{7.i.6}$ except for selecting $[\underline{\mathbf{y}_1}]$ uniformly random from $\mathbb{G}^{2k}$. Similarly, we can show that

$$|\Pr_{7.i.6}[b' = b] - \Pr_{7.i.7}[b' = b]| \leq 2\mathsf{Adv}^{\mathrm{mddh}}_{\mathcal{U}_k, \mathsf{GGen}, \mathcal{B}}(\lambda) + 2^{-\Omega(\lambda)}. \tag{62}$$

The proof of (62) can be seen as a combination of the proof of (49) and the proof of (50). We omit the proof details here.

Game $H_{7.i.7}$. $H_{7.i.7}$ is almost the same with $H_{7.i+1}$ and

$$\Pr_{7.i.7'}[b' = b] = \Pr_{7.i+1}[b' = b]. \tag{63}$$

Thus, combining (48, 50, 51, 59, 60, 61, 62) and (63), Lemma 7 follows.

□

Game $H_{7.\lambda}$. In this game, b is leaked to $\mathcal{A}$ only through γ_b (the output of $\mathcal{O}_{\text{enc}}$). We will show that, γ_1 is actually uniform random over $\mathbb{G}$, just like γ_0. This conclusion follows from the following facts.

- Note that the oracle $\mathcal{O}_{\text{dec}}$ in game $H_{7.\lambda}$ has the rejection rule

$$([\mathbf{y}] \in \boldsymbol{\psi}_{\text{enc}}) \vee \big(\exists [\mathbf{y}'] \in \boldsymbol{\psi}_{\text{enc}} \text{ s.t. } \mathsf{H}([\overline{\mathbf{y}}]) = \mathsf{H}([\overline{\mathbf{y}'}]) \wedge \mathbf{y} \neq \mathbf{y}'\big).$$

  This condition is equivalent to $\tau = \mathsf{H}([\overline{\mathbf{y}}]) \in \mathcal{T}_{\mathsf{enc}}$. It means if the random function RF_λ takes τ^1 as input in some $\mathcal{O}_{\text{enc}}$ query, it will not take the same input τ^1 in any $\mathcal{O}_{\text{dec}}$ query.
- RF_λ will take distinct input τ^1 in each $\mathcal{O}_{\text{enc}}$ query. This is due to the rejection rule of $\mathcal{O}_{\text{enc}}$.
- With probability $1 - 2^{-\Omega(\lambda)}$, $\mathbf{y}_1 \notin \mathsf{span}(\mathbf{M})$ for all the $\mathcal{O}_{\text{enc}}$ queries.

  Thus, for each γ_1 in $\mathcal{O}_{\text{enc}}$ query,

$$\gamma_1 = \left[\mathbf{y}_1^\top \sum_{j=1}^{\lambda} \mathbf{k}_{j,\tau_j^1} + \underbrace{\mathbf{y}_1^\top \mathbf{M}^\perp}_{\neq \mathbf{0}} \mathsf{RF}_\lambda(\tau^1)\right] \text{ is random.}$$

Since $\mathsf{RF}_\lambda(\tau^1)$ is not used anywhere else. Thus, γ_0 and γ_1 are (almost) identically distributed and we can conclude that

$$\left|\Pr_{7.\lambda}[b' = b] - \frac{1}{2}\right| \leq 2^{-\Omega(\lambda)}. \tag{64}$$

Finally, combining (33, 34, 40, 42, 43, 44, 45, 46, 47) and (64), Theorem 2 follows.

■

D Proof of Theorem 3

Proof of Theorem 3. Before we proving Theorem 3, we first prove the following lemma.

Lemma 8. *For the KEM* $\mathsf{KEM}_{\mathsf{mddh}}$ *in Figure 7, for any polynomial* $n = \mathsf{ploy}(\lambda)$ *and any PPT algorithm* $\mathcal{A}$*,*

$$|\Pr[\mathcal{A}(\mathsf{pk}_{\mathsf{KEM}}, \mathsf{sk}_{\mathsf{KEM}}, \psi_1, \cdots, \psi_n) = 1] - \Pr[\mathcal{A}(\mathsf{pk}_{\mathsf{KEM}}, \mathsf{sk}_{\mathsf{KEM}}, \psi'_1, \cdots, \psi'_n) = 1]| \\ \leq \mathsf{Adv}^{\mathrm{mddh}}_{\mathcal{U}_k, \mathsf{GGen}, \mathcal{B}}(\lambda) + 2^{-\Omega(\lambda)}$$

where $(\mathsf{pk}_{\mathsf{KEM}}, \mathsf{sk}_{\mathsf{KEM}}) \leftarrow_{\$} \mathsf{KGen}(1^\lambda)$ *and* $(\psi_i, \gamma_i) \leftarrow_{\$} \mathsf{KEnc}(\mathsf{pk}_{\mathsf{KEM}}), \psi'_i \leftarrow_{\$} \Psi$ *for all* $i \in [n]$*.*

Proof of Lemma 8. This lemma follows from the fact that the encapsulation ψ (which is the output of $\mathsf{KEnc}(\mathsf{pk}_{\mathsf{KEM}})$) is a random vector over $\mathsf{span}([\mathbf{M}])$ and is independent of $\mathsf{sk}_{\mathsf{KEM}}$. More precisely, we can build a MDDH adversary $\mathcal{B}'$ such that

$$|\Pr[\mathcal{A}(\mathsf{pk}_{\mathsf{KEM}}, \mathsf{sk}_{\mathsf{KEM}}, \psi_1, \cdots, \psi_n) = 1] - \Pr[\mathcal{A}(\mathsf{pk}_{\mathsf{KEM}}, \mathsf{sk}_{\mathsf{KEM}}, \psi'_1, \cdots, \psi'_n) = 1]| \\ \leq \mathsf{Adv}^{n\text{-mddh}}_{\mathcal{U}_{3k,k}, \mathsf{GGen}, \mathcal{B}'}(\lambda). \tag{65}$$

Upon receiving a challenge $(\mathcal{G}, [\mathbf{M}] \in \mathbb{G}^{3k\times k}, [\mathbf{H}] := ([\mathbf{h}_1|\cdots|\mathbf{h}_n]) \in \mathbb{G}^{3k\times n})$ for the n-fold $\mathcal{U}_{3k,k}$-MDDH problem, $\mathcal{B}'$ random selects $\mathbf{k}_{1,0}, \cdots, \mathbf{k}_{\lambda,1} \leftarrow_{\$} \mathbb{Z}_q^{3k}$ and calculates $([\mathbf{M}^\top \mathbf{k}_{j,\beta}])_{1\leq j\leq\lambda, 0\leq\beta\leq 1}$. Thus $\mathcal{B}'$ can perfectly simulate a properly distributed key pair $(\mathsf{pk}_{\mathsf{KEM}}, \mathsf{sk}_{\mathsf{KEM}})$. Then $\mathcal{B}'$ calls $\mathcal{A}(\mathsf{pk}_{\mathsf{KEM}}, \mathsf{sk}_{\mathsf{KEM}}, [\mathbf{h}_1], \cdots, [\mathbf{h}_n])$ and outputs whatever $\mathcal{A}$ outputs. Thus, if each column $[\mathbf{h}_i]$ of $[\mathbf{H}]$ is uniformly random over $\mathbb{G}^{3k}$, $\mathcal{B}'$ outputs $\mathcal{A}(\mathsf{pk}_{\mathsf{KEM}}, \mathsf{sk}_{\mathsf{KEM}}, \psi'_1, \cdots, \psi'_n)$. If each column $[\mathbf{h}_i]$ of $[\mathbf{H}]$ is uniformly random over $\mathsf{span}([\mathbf{M}])$, $\mathcal{B}'$ outputs $\mathcal{A}(\mathsf{pk}_{\mathsf{KEM}}, \mathsf{sk}_{\mathsf{KEM}}, \psi_1, \cdots, \psi_n)$. Thus (65) follows.

Finally Lemma 8 follows from (65), Lemma 6 and Lemma 3.

□

Now we prove Theorem 3. For any PPT adversary $\mathcal{A}$ with negligible uncertainty $uncert_{\mathcal{A}}(\lambda)$, consider an experiment $\mathsf{Exp}^{\mathrm{rer}}_{\mathsf{KEM}_{\mathsf{mddh}}, \mathcal{A}}(\lambda)$ which first randomly selects $b \leftarrow_{\$} \{0,1\}$, then calls $\mathsf{Exp}^{\mathrm{rer}\text{-}b}_{\mathsf{KEM}_{\mathsf{mddh}}, \mathcal{A}}(\lambda)$ and gets its output b'. It is straightforward that

$$\mathsf{Adv}^{\mathrm{rer}}_{\mathsf{KEM}_{\mathsf{mddh}}, \mathcal{A}}(\lambda) = 2\left|\Pr[b' = b \text{ in } \mathsf{Exp}^{\mathrm{rer}}_{\mathsf{KEM}_{\mathsf{mddh}}, \mathcal{A}}(\lambda)] - \frac{1}{2}\right|. \tag{66}$$

Then we rewrite $\mathsf{Exp}^{\mathrm{rer}}_{\mathsf{KEM}_{\mathsf{mddh}}, \mathcal{A}}(\lambda)$ in Figure 17 and make changes to it gradually through G_0 to G_3. Game $G_0 - G_3$ are defined below in Figure 17.

Game G_0. This game is almost the same as $\mathsf{Exp}^{\mathrm{rer}}_{\mathsf{KEM}_{\mathsf{mddh}}, \mathcal{A}}(\lambda)$. Then

$$\mathsf{Adv}^{\mathrm{special}}_{\mathsf{KEM}_{\mathsf{mddh}}, \mathcal{A}}(\lambda) = 2\left|\Pr_0[b' = b] - \frac{1}{2}\right|. \tag{67}$$

Game $G_0 - G_1$. G_1 is almost the same as G_0 except for the generation of ψ_i.

- In G_0, ψ_i is uniform over Ψ for all $i \in [n]$.
- In G_1, ψ_i is the output of $\mathsf{KEnc}(\mathsf{pk}_{\mathsf{KEM}})$ for all $i \in [n]$.

By Lemma 8, we have that

$$|\Pr_0[b' = b] - \Pr_1[b' = b]| \leq \mathsf{Adv}^{\mathrm{mddh}}_{\mathcal{U}_k, \mathsf{GGen}, \mathcal{B}}(\lambda) + 2^{-\Omega(\lambda)}. \tag{68}$$

The reduction is straightforward and we omit the details here.

```
$\underline{\mathsf{Exp}^{\mathrm{rer}}_{\mathsf{KEM}_{\mathsf{mddh}},\mathcal{A}}(\lambda)\text{:}}$
                 $G_0$ $\dbox{G_1, G_2}$ $\boxed{G_3}$
$b \leftarrow_\$ \{0,1\}$
$(\mathsf{pk}_{\mathsf{kem}}, \mathsf{sk}_{\mathsf{kem}}) \leftarrow_\$ \mathsf{KGen}(1^\lambda)$
$\boldsymbol{\psi}_{\mathrm{ran}} \leftarrow \emptyset$
$(st, 1^n) \leftarrow_\$ \mathcal{A}^{\mathcal{O}_{\mathrm{cha}}(\cdot,\cdot)}(\mathsf{pk}_{\mathsf{kem}})$
For $i \in [n]$:
    $\psi_i \leftarrow_\$ \Psi$
    $\dbox{(\psi_i, \gamma_i) \leftarrow_\$ \mathsf{KEnc}(\mathsf{pk}_{\mathsf{kem}})}$
    $\boxed{(\psi_i, \gamma_i) \leftarrow_\$ \Psi \times \Gamma}$
$\boldsymbol{\psi}_{\mathrm{ran}} \leftarrow \{\psi_1, \cdots, \psi_n\}$
$b' \leftarrow_\$ \mathcal{A}^{\mathcal{O}_{\mathrm{cha}}(\cdot,\cdot)}(st, \boldsymbol{\psi}_{\mathrm{ran}})$
Return $b'$
```

```
$\underline{\mathcal{O}_{\mathrm{cha}}(\psi, \mathsf{pred})\text{:}}$
                 $G_0, G_1$ $\dbox{G_2, G_3}$
If $\psi \notin \boldsymbol{\psi}_{\mathrm{ran}}$:
    $\gamma \leftarrow \mathsf{KDec}(\mathsf{sk}_{\mathsf{kem}}, \psi)$
    Return $\mathsf{pred}(\gamma)$
Else:    $//\psi = \psi_i \in \boldsymbol{\psi}_{\mathrm{ran}}$
    If $b = 1$:
        $\gamma \leftarrow \mathsf{KDec}(\mathsf{sk}_{\mathsf{kem}}, \psi_i)$
        $\dbox{\gamma \leftarrow \gamma_i}$
        Return $\mathsf{pred}(\gamma)$
    Else:
        Return 0
```

Fig. 17. Games $G_0 - G_3$ with respect to $\mathsf{Exp}^{\mathrm{rer}}_{\mathsf{KEM}_{\mathsf{mddh}},\mathcal{A}}(\lambda)$.

Game $G_1 - G_2$. G_2 is almost the same as G_1 except for one change in $\mathcal{O}_{\mathrm{cha}}$ oracle. In G_2, for a $\mathcal{O}_{\mathrm{cha}}(\psi, \mathsf{pred})$ query where $\psi = \psi_i \in \boldsymbol{\psi}_{\mathrm{ran}}$ and $b = 1$, instead of using $\gamma \leftarrow \mathsf{KDec}(\mathsf{sk}_{\mathsf{KEM}}, \psi_i)$, γ is set to γ_i which is generated by $(\psi_i, \gamma_i) \leftarrow_\$ \mathsf{KEnc}(\mathsf{pk}_{\mathsf{KEM}})$.

Since $\mathsf{KEM}_{\mathsf{mddh}}$ is perfectly correct, this change is conceptual. Then we have

$$\Pr_1[b' = b] = \Pr_2[b' = b]. \tag{69}$$

Game $G_2 - G_3$. G_3 is almost the same as G_2 except for the generation of (ψ_i, γ_i).

- In G_2, (ψ_i, γ_i) is the output of $\mathsf{KEnc}(\mathsf{pk}_{\mathsf{KEM}})$ for all $i \in [n]$.
- In G_3, (ψ_i, γ_i) is uniform over $\Psi \times \Gamma$ for all $i \in [n]$.

We will reduce the indistinguishability between G_2 and G_3 to the mPR-CCCA security of $\mathsf{KEM}_{\mathsf{mddh}}$. More precisely, we will build an adversary $\mathcal{B}$ (with $uncert_{\mathcal{B}}(\lambda) = uncert_{\mathcal{A}}(\lambda)$) against the mPR-CCCA security of $\mathsf{KEM}_{\mathsf{mddh}}$ such that

$$|\Pr_2[b' = b] - \Pr_3[b' = b]| \leq \mathsf{Adv}^{\mathrm{mpr\text{-}ccca}}_{\mathsf{KEM}_{\mathsf{mddh}},\mathcal{B}}(\lambda). \tag{70}$$

On input $\mathsf{pk}_{\mathsf{KEM}}$, $\mathcal{B}$ simulates game $G_2(G_3)$ as follows.

- $\mathcal{B}$ randomly selects $b \leftarrow_\$ \{0,1\}$.
- $\mathcal{B}$ calls $\mathcal{A}^{\mathcal{O}_{\mathrm{cha}}(\cdot,\cdot)}(\mathsf{pk}_{\mathsf{KEM}})$ to get $(st, 1^n)$.
  - To simulate $\mathcal{O}_{\mathrm{cha}}(\psi, \mathsf{pred})$ for $\mathcal{A}$, $\mathcal{B}$ submits a (pred, ψ) query to its own $\mathcal{O}_{\mathrm{dec}}$ oracle. Since $\mathcal{B}$ has not submitted any $\mathcal{O}_{\mathrm{enc}}$ query yet, set $\boldsymbol{\psi}_{\mathrm{ran}}$ is empty. So it will always get a bit $d = \mathsf{pred}(\mathsf{KDec}(\mathsf{sk}_{\mathsf{KEM}}, \psi))$ as response. Then $\mathcal{B}$ forwards to $\mathcal{A}$ the bit d as response.
- $\mathcal{B}$ queries $\mathcal{O}_{\mathrm{enc}}()$ oracle n times and gets the response (ψ_i, γ_i) for $i \in [n]$.
- $\mathcal{B}$ sets $\boldsymbol{\psi}_{\mathrm{ran}} \leftarrow \{\psi_1, \cdots, \psi_n\}$ and calls $\mathcal{A}^{\mathcal{O}_{\mathrm{cha}}(\cdot,\cdot)}(st, \boldsymbol{\psi}_{\mathrm{ran}})$ to get b'.
  - To simulate $\mathcal{O}_{\mathrm{cha}}(\psi, \mathsf{pred})$ for $\mathcal{A}$, note that now $\boldsymbol{\psi}_{\mathrm{ran}} = \boldsymbol{\psi}_{\mathrm{enc}}$. Thus
    - If $\psi \notin \boldsymbol{\psi}_{\mathrm{ran}}$, $\mathcal{B}$ asks it own oracle to answer the $\mathcal{O}_{\mathrm{cha}}(\psi, \mathsf{pred})$ query from $\mathcal{A}$ as before.
    - If $\psi = \psi_i \in \boldsymbol{\psi}_{\mathrm{ran}}$, to make sure that $uncert_{\mathcal{B}}(\lambda) = uncert_{\mathcal{A}}(\lambda)$, $\mathcal{B}$ first queries its own oracle $\mathcal{O}_{\mathrm{dec}}(\mathsf{pred}, \psi_i)$ and gets the response 0 (since $\psi_i \in \boldsymbol{\psi}_{\mathrm{ran}} = \boldsymbol{\psi}_{\mathrm{enc}}$). Then $\mathcal{B}$ returns 0 if $b = 0$ and returns $\mathsf{pred}(\gamma_i)$ if $b = 1$.

- $\mathcal{B}$ outputs 1 if and only if $(b' = b)$.

$\mathcal{B}$ perfectly simulates game G_2 for $\mathcal{A}$ if the response of $\mathcal{O}_{\text{enc}}()$ oracle (ψ_i, γ_i) is the output of $\mathsf{KEnc}(\mathsf{pk}_{\mathsf{KEM}})$ for all $i \in [n]$ and perfectly simulates game G_3 for $\mathcal{A}$ if (ψ_i, γ_i) is uniform over $\Psi \times \Gamma$ for all $i \in [n]$. Thus, (70) follows.

Game G_3. We will show that

$$\left|\Pr_3[b' = b] - \frac{1}{2}\right| \leq Q_{\text{cha}} \cdot \mathit{uncert}_{\mathcal{A}}(\lambda). \tag{71}$$

We first define $\mathcal{I}_{\text{in}} := \{j \in [Q_{\text{cha}}] \mid \psi \in \boldsymbol{\psi}_{\text{ran}} \text{ in the } j\text{-th } \mathcal{O}_{\text{cha}}(\psi, \mathsf{pred}_j) \text{ query}\}$. Consider the j-th $\mathcal{O}_{\text{cha}}(\psi, \mathsf{pred}_j)$ query for $j \in \mathcal{I}_{\text{in}}$. Suppose $\psi = \psi_i \in \boldsymbol{\psi}_{\text{ran}}$. Then $\mathcal{O}_{\text{cha}}$ will respond to $\mathcal{A}$ as follows $\begin{cases} \mathsf{pred}_j(\gamma_i) & \text{if } b = 1 \\ 0 & \text{if } b = 0 \end{cases}$. Suppose $b = 1$, now let's consider the probability that there exists $j \in \mathcal{I}_{\text{in}}$ such that the j-th $\mathcal{O}_{\text{cha}}$ query returns 1, i.e., $\Pr[\exists j \in \mathcal{I}_{\text{in}}, \mathsf{pred}_j(\gamma_i) = 1]$. In game G_3, since each γ_i is uniform over Γ, we have that

$$\begin{aligned} \Pr[\exists j \in \mathcal{I}_{\text{in}}, \mathsf{pred}_j(\gamma_i) = 1] &\leq \sum_{j \in \mathcal{I}_{\text{in}}} \Pr_{\gamma \leftarrow_{\$} \Gamma}[\mathsf{pred}_j(\gamma) = 1] \\ &\leq \sum_{j \in [Q_{\text{cha}}]} \Pr_{\gamma \leftarrow_{\$} \Gamma}[\mathsf{pred}_j(\gamma) = 1] = Q_{\text{cha}} \cdot \mathit{uncert}_{\mathcal{A}}(\lambda). \end{aligned}$$

Thus, no adversary can have an advantage greater than $Q_{\text{cha}} \cdot \mathit{uncert}_{\mathcal{A}}(\lambda)$ in game G_3 and (71) follows.

Finally, combining (67, 68, 69, 70) and (71), Theorem 3 follows.

■

E Supplementary Materials for Qualified Proof System

Recall the definition of $\mathcal{L}^{\text{snd}}$-indistinguishability of two proof systems in [GHK17].

Definition 17 ($\mathcal{L}^{\text{snd}}$-indistinguishability of two proof systems). *Let* $\mathsf{PS}_0 = (\mathsf{PGen}_0, \mathsf{PPrv}_0, \mathsf{PVer}_0, \mathsf{PSim}_0)$ *and* $\mathsf{PS}_1 = (\mathsf{PGen}_1, \mathsf{PPrv}_1, \mathsf{PVer}_1, \mathsf{PSim}_1)$ *be two proof systems for a family of languages* $\mathcal{L} = \mathcal{L}_{\mathsf{pars}}$. *Let* $\mathcal{L}^{\text{snd}} = \{\mathcal{L}^{\text{snd}}_{\mathsf{pars}}\}$ *be a family of languages, such that* $\mathcal{L}_{\mathsf{pars}} \subseteq \mathcal{L}^{\text{snd}}_{\mathsf{pars}}$. *For any adversary* $\mathcal{A}$, *define experiment* $\mathrm{Exp}^{\text{PS-ind}}_{\mathcal{L}^{\text{snd}}, \mathsf{PS}_0, \mathsf{PS}_1, \mathcal{A}}(\lambda)$ *in Figure 18. We say* PS_0 *and* PS_1 *are* $\mathcal{L}^{\text{snd}}$-indistinguishable, *if for all unbounded adversary* $\mathcal{A}$, *the advantage*

$$\mathsf{Adv}^{\text{PS-ind}}_{\mathcal{L}^{\text{snd}}, \mathsf{PS}_0, \mathsf{PS}_1, \mathcal{A}}(\lambda) := \left|\Pr\left[\mathrm{Exp}^{\text{PS-ind}}_{\mathcal{L}^{\text{snd}}, \mathsf{PS}_0, \mathsf{PS}_1, \mathcal{A}}(\lambda) = 1\right] - \frac{1}{2}\right|$$

is negligible in λ.

Now we recall the definition of $\widetilde{\mathcal{L}^{\text{snd}}}$-extensibility of a proof system proposed in [GHK17].

Definition 18 ($\widetilde{\mathcal{L}^{\text{snd}}}$-extensibility of a proof system). *Let* $\mathcal{L} \subseteq \mathcal{L}^{\text{snd}} \subseteq \widetilde{\mathcal{L}^{\text{snd}}}$ *be three family of languages. An* $\mathcal{L}^{\text{snd}}$*-qualified proof system* PS *is said to be* $\widetilde{\mathcal{L}^{\text{snd}}}$-extensible *if there exists a proof system* $\widetilde{\mathsf{PS}}$ *for* $\mathcal{L}$ *that complies with* $\widetilde{\mathcal{L}^{\text{snd}}}$*-constrained soundness and such that* PS *and* $\widetilde{\mathsf{PS}}$ *are* $\mathcal{L}^{\text{snd}}$*-indistinguishable.*

```
$\mathrm{Exp}^{\text{PS-ind}}_{\mathcal{L}^{\mathrm{snd}},\mathsf{PS}_0,\mathsf{PS}_1,\mathcal{A}}(\lambda)$:
$b \leftarrow_\$ \{0,1\}$
$(\mathsf{ppk}, \mathsf{psk}) \leftarrow_\$ \mathsf{PGen}_b(\mathsf{pars})$
$b' \leftarrow_\$ \mathcal{A}^{\mathcal{O}_{\mathrm{sim}}(), \mathcal{O}^{b}_{\mathrm{ver}}(\cdot,\cdot,\cdot)}(\mathsf{ppk})$
Return $\begin{cases} 1 & \text{If } b' = b \\ 0 & \text{Otherwise} \end{cases}$
```

```
$\mathcal{O}_{\mathrm{sim}}()$:
$x \leftarrow_\$ \mathcal{L}^{\mathrm{snd}} \backslash \mathcal{L}$
$(\Pi, K) \leftarrow \mathsf{PSim}_b(\mathsf{psk}, x)$
Return $(x, \Pi, K)$
```

```
$\mathcal{O}_{\mathrm{ver}}(x, \Pi, \mathsf{pred})$:
$(v, K) \leftarrow \mathsf{PVer}_b(\mathsf{psk}, x, \Pi)$
If $\begin{bmatrix} x \notin \mathcal{L}^{\mathrm{snd}} \vee v = 0 \\ \vee \mathsf{pred}(K) = 0 \end{bmatrix}$:
    Return $\perp$
Return $K$
```

Fig. 18. Experiment in the definition of $\mathcal{L}^{\mathrm{snd}}$-indistinguishability of two proof systems.

F Proof of Lemma 1

Proof of Lemma 1. Game G_1 differs from G_0 if and only if $\mathcal{A}$ submits an $\mathcal{O}_{\mathrm{dec}}(\mathsf{pred}, \psi = ([\mathbf{c}], \Pi))$ query such that

$$([\mathbf{c}], \Pi) \notin \boldsymbol{\psi}_{\mathrm{enc}} \wedge v = 1 \wedge \mathsf{pred}(\gamma) = 1 \wedge \tau \in \mathcal{T}.$$

$\tau \in \mathcal{T}$ means there exist a previous $\mathcal{O}_{\mathrm{enc}}$ query such that $[\mathbf{c}_b]$ is sampled and $\mathsf{H}([\overline{\mathbf{c}}])$ equals $\mathsf{H}([\overline{\mathbf{c}_b}])$. We will denote this event as Bad and G_1 differs from G_0 if and only if Bad happens. So we have

$$|\Pr_0[b' = b] - \Pr_1[b' = b]| \leq \Pr_0[\mathsf{Bad}]. \tag{72}$$

Since we have

$$\Pr_0[\mathsf{Bad}] = \frac{1}{2}\Pr_0[\mathsf{Bad} \mid b = 1] + \frac{1}{2}\Pr_0[\mathsf{Bad} \mid b = 0], \tag{73}$$

we then bound $\Pr_0[\mathsf{Bad}]$ with Lemma 9 and Lemma 10.

Lemma 9.

$$\Pr_0[\mathsf{Bad} \mid b = 1] \leq \mathsf{Adv}^{\mathrm{cr}}_{\mathcal{H},\mathcal{B}_1}(\lambda) + Q_{\mathrm{dec}} \cdot \mathit{uncert}_{\mathcal{A}}(\lambda).$$

Proof of Lemma 9. We define game H which is exactly the same with G_0 when $b = 1$. We denote $\Pr_H[\mathsf{E}]$ the probability that event E happens in game H. Then we have

$$\Pr_0[\mathsf{Bad} \mid b = 1] = \Pr_H[\mathsf{Bad}]. \tag{74}$$

Recall that, Bad happens when $\mathcal{A}$ submits an $\mathcal{O}_{\mathrm{dec}}(\mathsf{pred}, \psi = ([\mathbf{c}], \Pi))$ query such that

$$([\mathbf{c}], \Pi) \notin \boldsymbol{\psi}_{\mathrm{enc}} \wedge v = 1 \wedge \mathsf{pred}(\gamma) = 1 \wedge \tau \in \mathcal{T}.$$

We decompose it into two subevents, $\mathsf{Bad}^{\mathsf{in}} := \mathsf{Bad} \wedge [\mathbf{c}] \in \mathsf{span}([\mathbf{A}])$ and $\mathsf{Bad}^{\mathsf{out}} := \mathsf{Bad} \wedge [\mathbf{c}] \notin \mathsf{span}([\mathbf{A}])$. It is straightforward that

$$\Pr_H[\mathsf{Bad}] \leq \Pr_H[\mathsf{Bad}^{\mathsf{in}}] + \Pr_H[\mathsf{Bad}^{\mathsf{out}}]. \tag{75}$$

First we bound $\Pr_H[\mathsf{Bad}^{\mathsf{in}}]$. In H, $\tau \in \mathcal{T}$ means that there exists a previous $\mathcal{O}_{\mathrm{enc}}$ query such that $[\mathbf{c}_1] = [\mathbf{A}]\mathbf{r}_1$ is sampled and $\mathsf{H}([\overline{\mathbf{c}}])$ equals $\mathsf{H}([\overline{\mathbf{c}_1}])$. We further decompose this event into three cases as follows.

- For the case $[\overline{\mathbf{c}}] \neq [\overline{\mathbf{c}_1}]$, we found a collision for H.

- For the case $[\overline{\mathbf{c}}] = [\overline{\mathbf{c}_1}] \wedge [\underline{\mathbf{c}}] \neq [\underline{\mathbf{c}_1}]$, it will never happen. Since $[\mathbf{c}], [\mathbf{c}_1] \in \mathsf{span}([\mathbf{A}])$ and $[\overline{\mathbf{A}}]$ forms an invertible matrix, $[\overline{\mathbf{c}}] = [\overline{\mathbf{c}_1}]$ would imply $[\underline{\mathbf{c}}] = [\underline{\mathbf{c}_1}]$.
- For the case $[\mathbf{c}] = [\mathbf{c}_1]$. $([\mathbf{c}_1], \Pi_1) \in \boldsymbol{\psi}_{\text{enc}}$ for $(\Pi_1, [\kappa_1]) \leftarrow \mathsf{PPrv}(\mathsf{ppk}, [\mathbf{c}_1], \mathbf{r}_1)$. Since $([\mathbf{c}], \Pi) \notin \boldsymbol{\psi}_{\text{enc}}$, we have that $\Pi \neq \Pi_1$. By the perfect completeness property of PS, the verification $\mathsf{PVer}(\mathsf{psk}, [\mathbf{c}_1], \Pi_1)$ will always pass. However, $v = 1$ implies that the verification $\mathsf{PVer}(\mathsf{psk}, [\mathbf{c}] = [\mathbf{c}_1], \Pi)$ also passes. This contradicts to the proof uniqueness property of PS. So this case will never happen.

Thus, we can build a PPT adversary $\mathcal{B}_1$ and show that,

$$\Pr_H[\mathsf{Bad}^{\mathsf{in}}] \leq \mathsf{Adv}^{\text{cr}}_{\mathcal{H},\mathcal{B}_1}(\lambda). \tag{76}$$

Next, we bound $\Pr_H[\mathsf{Bad}^{\mathsf{out}}]$. For $i \in [Q_{\text{dec}}]$, we define event $\mathsf{Bad}^{\mathsf{out}}_i$ be the event that $\mathsf{Bad}^{\mathsf{out}}$ first happens in the i-th $\mathcal{O}_{\text{dec}}$ query. Thus we have,

$$\Pr_H[\mathsf{Bad}^{\mathsf{out}}] = \sum_{i=1}^{Q_{\text{dec}}} \Pr_H[\mathsf{Bad}^{\mathsf{out}}_i]. \tag{77}$$

We define a new event $\widetilde{\mathsf{Bad}}$ be $\mathcal{A}$ submits an $\mathcal{O}_{\text{dec}}(\mathsf{pred}, \psi = ([\mathbf{c}], \Pi))$ query such that

$$\mathsf{pred}(\gamma) = 1 \wedge [\mathbf{c}] \notin \mathsf{span}([\mathbf{A}]).$$

Meanwhile, for $i \in [Q_{\text{dec}}]$, we define $\widetilde{\mathsf{Bad}}_i$ as the event that $\widetilde{\mathsf{Bad}}$ first happens in the i-th $\mathcal{O}_{\text{dec}}$ query. Then $\mathsf{Bad}^{\mathsf{out}}_i$ is a subevent of $\widetilde{\mathsf{Bad}}_i$ and

$$\Pr_H[\mathsf{Bad}^{\mathsf{out}}_i] \leq \Pr_H[\widetilde{\mathsf{Bad}}_i]. \tag{78}$$

Then we bound $\Pr_H[\widetilde{\mathsf{Bad}}_i]$. We use the fact that half of the $\mathbf{k}_0$'s entropy is hidden from $\mathcal{A}$. More precisely, $\mathbf{k}_0 \leftarrow_\$ \mathbb{Z}_q^{2k}$ is identically distributed as $\mathbf{k}_0 + \mathbf{A}^\perp \mathbf{w}$, where $\mathbf{k}_0 \leftarrow_\$ \mathbb{Z}_q^{2k}, \mathbf{w} \leftarrow_\$ \mathbb{Z}_q^k$ and $\mathbf{A}^\perp \in \mathbb{Z}_q^{2k \times k}$ s.t. $\mathbf{A}^\top \mathbf{A}^\perp = \mathbf{0}$. Then we will show that, in game H, $\mathbf{w}$ is hidden from $\mathcal{A}$ before $\widetilde{\mathsf{Bad}}$ first happens in the i-th $\mathcal{O}_{\text{dec}}$ query.

- The public key $\mathsf{pk}_{\mathsf{kem}}$ does not leak any information about $\mathbf{w}$ since

$$(\mathbf{k}_0^\top + \mathbf{w}^\top (\mathbf{A}^\perp)^\top)\mathbf{A} = \mathbf{k}_0^\top \mathbf{A}.$$

- $\mathcal{O}_{\text{enc}}$ also hides $\mathbf{w}$ since $\mathcal{O}_{\text{enc}}$ in H only uses $\mathsf{pk}_{\mathsf{kem}}$.
- The first $i-1$ $\mathcal{O}_{\text{dec}}(\mathsf{pred}, \psi = ([\mathbf{c}], \Pi))$ queries also hides $\mathbf{w}$. Since in H, before the i-th $\mathcal{O}_{\text{dec}}$ query,
  - if $[\mathbf{c}] \notin \mathsf{span}([\mathbf{A}])$ then $\mathsf{pred}(\gamma) = 0$ (otherwise $\widetilde{\mathsf{Bad}}$ first happens before the i-th query). So these queries are rejected directly and is independent of $\mathbf{w}$;
  - if $[\mathbf{c}] \in \mathsf{span}([\mathbf{A}])$, since

$$((\mathbf{k}_0 + \mathbf{A}^\perp \mathbf{w})^\top + \tau \mathbf{k}_1^\top) \cdot [\mathbf{c}] + [\kappa] = (\mathbf{k}_0^\top + \tau \mathbf{k}_1^\top) \cdot [\mathbf{c}] + [\kappa] + \mathbf{w}^\top \underbrace{(\mathbf{A}^\perp)^\top [\mathbf{c}]}_{=[\mathbf{0}]},$$

  $\mathbf{w}$ is not used yet.

Thus $\mathbf{w}$ is completely hidden from $\mathcal{A}$ until the i-th $\mathcal{O}_{\text{dec}}$ query. So in the i-th $\mathcal{O}_{\text{dec}}(\mathsf{pred}_i, \psi = ([\mathbf{c}], \Pi))$ query, if $[\mathbf{c}] \notin \mathsf{span}([\mathbf{A}])$, since

$$\gamma = (\mathbf{k}_0^\top + \tau \mathbf{k}_1^\top) \cdot [\mathbf{c}] + [\kappa] + \mathbf{w}^\top \underbrace{(\mathbf{A}^\perp)^\top [\mathbf{c}]}_{\neq [\mathbf{0}]},$$

γ will be random due to the randomness of $\mathbf{w}$. In this case,

$$\Pr_H[\widetilde{\mathsf{Bad}_i}] = \Pr_{\gamma \leftarrow_{\$} \Gamma}[\mathsf{pred}_i(\gamma)]. \tag{79}$$

So combining (77), (78) and (79). We have that

$$\Pr_H[\mathsf{Bad}^{\mathsf{out}}] = \sum_{i=1}^{Q_{\mathrm{dec}}} \Pr_H[\mathsf{Bad}_i^{\mathsf{out}}] \leq \sum_{i=1}^{Q_{\mathrm{dec}}} \Pr_{\gamma \leftarrow_{\$} \Gamma}[\mathsf{pred}_i(\gamma)] = Q_{\mathrm{dec}} \cdot \mathit{uncert}_{\mathcal{A}}(\lambda). \tag{80}$$

Finally, Lemma 9 follows from (74), (75), (76) and (80).

□

Lemma 10.

$$\Pr_0[\mathsf{Bad} \mid b = 0] \leq \mathsf{Adv}^{\mathrm{cr}}_{\mathcal{H},\mathcal{B}_1}(\lambda) + k \cdot \mathsf{Adv}^{\mathrm{mddh}}_{\mathcal{D}_{2k,k},\mathsf{GGen},\mathcal{B}_2}(\lambda) + \mathsf{Adv}^{\mathrm{mddh}}_{\mathcal{U}_k,\mathsf{GGen},\mathcal{B}_3}(\lambda) \\ + 2Q_{\mathrm{dec}} \cdot \mathit{uncert}_{\mathcal{A}}(\lambda) + 2^{-\Omega(\lambda)}$$

Proof of Lemma 10. We define game J which is identical to G_0 conditioned on $b = 0$. We denote $\Pr_J[\mathsf{E}]$ the probability that event E happens in game J. Then we have

$$\Pr_0[\mathsf{Bad} \mid b = 0] = \Pr_J[\mathsf{Bad}]. \tag{81}$$

Recall Bad is the event that $\mathcal{A}$ submits an $\mathcal{O}_{\mathrm{dec}}(\mathsf{pred}, \psi = ([\mathbf{c}], \Pi))$ query such that $\exists([\mathbf{c}_0], \Pi_0) \in \boldsymbol{\psi}_{\mathrm{enc}}$

$$([\mathbf{c}], \Pi) \notin \boldsymbol{\psi}_{\mathrm{enc}} \wedge v = 1 \wedge \mathsf{pred}(\gamma) = 1 \wedge \mathsf{H}([\overline{\mathbf{c}}]) = \mathsf{H}([\overline{\mathbf{c}_0}]).$$

Similarly, we can further decompose this event into three cases as follows.

- For the case $[\overline{\mathbf{c}}] \neq [\overline{\mathbf{c}_0}]$, we found a collision for H.
- For the case $[\overline{\mathbf{c}}] = [\overline{\mathbf{c}_0}] \wedge [\underline{\mathbf{c}}] \neq [\underline{\mathbf{c}_0}]$. We denote this subevent by $\mathsf{Bad}_{\mathsf{A}}$.
- For the case $[\mathbf{c}] = [\mathbf{c}_0]$. We denote this subevent by $\mathsf{Bad}_{\mathsf{B}}$.

Thus, we can build a PPT adversary $\mathcal{B}_1$ and show that,

$$\Pr_J[\mathsf{Bad}] \leq \mathsf{Adv}^{\mathrm{cr}}_{\mathcal{H},\mathcal{B}_1}(\lambda) + \Pr_J[\mathsf{Bad}_{\mathsf{A}}] + \Pr_J[\mathsf{Bad}_{\mathsf{B}}]. \tag{82}$$

To bound $\Pr_J[\mathsf{Bad}_{\mathsf{A}}]$, we first change game J to J_A.

- In game J, $[\mathbf{c}_0]$ is uniformly chosen from $\mathbb{G}^{2k}$ in each $\mathcal{O}_{\mathrm{enc}}$ query.
- In game J_A, $[\mathbf{c}_0]$ is uniformly chosen from $\mathsf{span}([\mathbf{A}])$ in each $\mathcal{O}_{\mathrm{enc}}$ query.

We can build an adversary $\mathcal{B}_2$ and show that

$$|\Pr_J[\mathsf{Bad}_{\mathsf{A}}] - \Pr_{J_A}[\mathsf{Bad}_{\mathsf{A}}]| \leq k \cdot \mathsf{Adv}^{\mathrm{mddh}}_{\mathcal{D}_{2k,k},\mathsf{GGen},\mathcal{B}_2}(\lambda) + 2^{-\Omega(\lambda)}. \tag{83}$$

To prove (83), we construct an adversary $\mathcal{B}_2'$ and show that

$$|\Pr_J[\mathsf{Bad}_{\mathsf{A}}] - \Pr_{J_A}[\mathsf{Bad}_{\mathsf{A}}]| \leq \mathsf{Adv}^{Q_{\mathrm{enc}}\text{-mddh}}_{\mathcal{D}_{2k,k},\mathsf{GGen},\mathcal{B}_2'}(\lambda). \tag{84}$$

Upon receiving a challenge $(\mathcal{G}, [\mathbf{M}] \in \mathbb{G}^{2k \times k}, [\mathbf{H}] := ([\mathbf{h}_1| \cdots |\mathbf{h}_{Q_{\mathrm{enc}}}]) \in \mathbb{G}^{2k \times Q_{\mathrm{enc}}})$ for the Q_{enc}-fold $\mathcal{U}_{2k,k}$-MDDH problem, $\mathcal{B}_2'$ simulates game $J(J_A)$. To reply the i-th $\mathcal{O}_{\mathrm{enc}}$ query made by $\mathcal{A}$, $\mathcal{B}_2'$ embeds $[\mathbf{h}_i]$ to $[\mathbf{c}_0]$, i.e., $[\mathbf{c}_0] \leftarrow [\mathbf{h}_i]$. Finally $\mathcal{B}_2'$ outputs 1 if and only if event $\mathsf{Bad}_{\mathsf{A}}$ happens. Thus, if each column $[\mathbf{h}_i]$ of $[\mathbf{H}]$ is uniformly random over

$\mathbb{G}^{2k}$, $\mathcal{B}_2'$ perfectly simulates game J. If each column $[\mathbf{h}_i]$ of $[\mathbf{H}]$ is uniformly random over $\mathsf{span}([\mathbf{A}])$, $\mathcal{B}_2'$ perfectly simulates game J_A. So (84) follows.

Furthermore, (83) follows from (84) and Lemma 5.

In game J_A, we further decompose event Bad_A into two subevents. $\mathsf{Bad}_\mathsf{A}^\mathsf{in} := \mathsf{Bad}_\mathsf{A} \wedge [\mathbf{c}] \in \mathsf{span}([\mathbf{A}])$ and $\mathsf{Bad}_\mathsf{A}^\mathsf{out} := \mathsf{Bad}_\mathsf{A} \wedge [\mathbf{c}] \notin \mathsf{span}([\mathbf{A}])$. So we have

$$\Pr_{J_A}[\mathsf{Bad}_\mathsf{A}] \le \Pr_{J_A}[\mathsf{Bad}_\mathsf{A}^\mathsf{in}] + \Pr_{J_A}[\mathsf{Bad}_\mathsf{A}^\mathsf{out}]. \tag{85}$$

Similar to (80), we have

$$\Pr_{J_A}[\mathsf{Bad}_\mathsf{A}^\mathsf{out}] \le Q_{\mathrm{dec}} \cdot \mathit{uncert}_{\mathcal{A}}(\lambda). \tag{86}$$

For $\mathsf{Bad}_\mathsf{A}^\mathsf{in}$, since $[\mathbf{c}], [\mathbf{c}_0] \in \mathsf{span}([\mathsf{A}])$ and $\overline{\mathbf{A}}$ forms an invertible matrix. Then $[\overline{\mathbf{c}}] = [\overline{\mathbf{c}_0}]$ would imply that $[\underline{\mathbf{c}}] = [\underline{\mathbf{c}_0}]$. So $\mathsf{Bad}_\mathsf{A}^\mathsf{in}$ never happens in game J_A and

$$\Pr_{J_A}[\mathsf{Bad}_\mathsf{A}^\mathsf{in}] = 0. \tag{87}$$

Combining (83), (85), (86) and (87), we have that

$$\Pr_J[\mathsf{Bad}_\mathsf{A}] \le k \cdot \mathsf{Adv}^{\mathrm{mddh}}_{\mathcal{D}_{2k,k},\mathsf{GGen},\mathcal{B}_2}(\lambda) + Q_{\mathrm{dec}} \cdot \mathit{uncert}_{\mathcal{A}}(\lambda) + 2^{-\Omega(\lambda)}. \tag{88}$$

To bound $\Pr_J[\mathsf{Bad}_\mathsf{B}]$, we first change game J to J_B.

- In game J, $[\mathbf{c}_0]$ is uniformly chosen from $\mathbb{G}^{2k}$ in each $\mathcal{O}_{\mathrm{enc}}$ query.
- In game J_B, $[\mathbf{c}_0]$ is uniformly chosen from $\mathsf{span}([\mathbf{A}_0])$ in each $\mathcal{O}_{\mathrm{enc}}$ query.

Similar to (84), we can build an adversary $\mathcal{B}_3'$ and show that

$$|\Pr_J[\mathsf{Bad}_\mathsf{B}] - \Pr_{J_B}[\mathsf{Bad}_\mathsf{B}]| \le \mathsf{Adv}^{Q_{\mathrm{enc}}\text{-mddh}}_{\mathcal{U}_{2k,k},\mathsf{GGen},\mathcal{B}_3'}(\lambda). \tag{89}$$

By Lemma 6 and Lemma 3, we can build an adversary $\mathcal{B}_3$ and show that

$$|\Pr_J[\mathsf{Bad}_\mathsf{B}] - \Pr_{J_B}[\mathsf{Bad}_\mathsf{B}]| \le \mathsf{Adv}^{\mathrm{mddh}}_{\mathcal{U}_k,\mathsf{GGen},\mathcal{B}_3}(\lambda) + 2^{-\Omega(\lambda)}. \tag{90}$$

Similarly, we can further decompose event Bad_B into two subevents. $\mathsf{Bad}_\mathsf{B}^\mathsf{in} := \mathsf{Bad}_\mathsf{B} \wedge [\mathbf{c}] \in \mathsf{span}([\mathbf{A}])$ and $\mathsf{Bad}_\mathsf{B}^\mathsf{out} := \mathsf{Bad}_\mathsf{B} \wedge [\mathbf{c}] \notin \mathsf{span}([\mathbf{A}])$. So we have

$$\Pr_{J_B}[\mathsf{Bad}_\mathsf{B}] \le \Pr_{J_B}[\mathsf{Bad}_\mathsf{B}^\mathsf{in}] + \Pr_{J_B}[\mathsf{Bad}_\mathsf{B}^\mathsf{out}]. \tag{91}$$

Similar to (80), we have

$$\Pr_{J_B}[\mathsf{Bad}_\mathsf{B}^\mathsf{out}] \le Q_{\mathrm{dec}} \cdot \mathit{uncert}_{\mathcal{A}}(\lambda). \tag{92}$$

For $\mathsf{Bad}_\mathsf{B}^\mathsf{in}$, since $[\mathbf{c}] \in \mathsf{span}([\mathbf{A}])$ and $[\mathbf{c}_0] \in \mathsf{span}([\mathbf{A}_0])$, $[\mathbf{c}] = [\mathbf{c}_0]$ means that $[\mathbf{c}] = [\mathbf{c}_0] \in \mathsf{span}([\mathbf{A}]) \cap \mathsf{span}([\mathbf{A}_0])$. With overwhelming probability $\mathsf{span}([\mathbf{A}]) \cap \mathsf{span}([\mathbf{A}_0]) = \{[\mathbf{0}] \in \mathbb{G}^{2k}\}$. Since $[\mathbf{c}_0]$ is uniform over $\mathsf{span}([\mathbf{A}_0])$, $[\mathbf{c}_0] = [\mathbf{0}]$ happens with probability only $2^{-\Omega(\lambda)}$. Thus we have

$$\Pr_{J_B}[\mathsf{Bad}_\mathsf{B}^\mathsf{in}] \le 2^{-\Omega(\lambda)}. \tag{93}$$

Combining (90, 91, 92) and (93), we have that

$$\Pr_J[\mathsf{Bad}_\mathsf{B}] \le \mathsf{Adv}^{\mathrm{mddh}}_{\mathcal{U}_k,\mathsf{GGen},\mathcal{B}_3}(\lambda) + Q_{\mathrm{dec}} \cdot \mathit{uncert}_{\mathcal{A}}(\lambda) + 2^{-\Omega(\lambda)}. \tag{94}$$

Then, Lemma 10 follows from (81, 82, 88) and (94).

Finally, Lemma 1 follows from (72), (73), Lemma 9 and Lemma 10.

□

G Proof of (28)

Define $\mathbf{u}^\top := \mathbf{k}_0^\top \mathbf{A}_0$, so $(\mathbf{p}_0^\top | \mathbf{u}^\top) = \mathbf{k}_0^\top (\mathbf{A} | \mathbf{A}_0)$. Note that, the square matrix $(\mathbf{A} | \mathbf{A}_0)$ is of full rank with probability $1 - 2^{-\Omega(\lambda)}$, then the entropy of $\mathbf{k}_0^\top$ is transferred to $(\mathbf{p}_0^\top | \mathbf{u}^\top)$ intactly. Recall that $\mathbf{k}_0^\top$ is uniform over $\mathbb{Z}_q^{1 \times 2k}$. Therefore, $(\mathbf{p}_0^\top | \mathbf{u}^\top)$ is uniform over $\mathbb{Z}_q^{1 \times 2k}$ as well. Consequently, $\mathbf{u}^\top$ is uniformly distributed over $\mathbb{Z}_q^{1 \times k}$ even conditioned on $\mathbf{p}_0^\top$.

In G_5, the $\mathcal{O}_{\text{dec}}$ oracle rejects all $[\mathbf{c}] \notin [\mathsf{span}(\mathbf{A})]$. Therefore, the information of $\mathbf{k}_0^\top$ leaked through $\mathcal{O}_{\text{dec}}$ is characterized by the public key $\mathbf{p}_0^\top$. Together with the fact that $[\mathbf{c}_1] = [\mathbf{A}_0]\mathbf{r}_1$ in $\mathcal{O}_{\text{enc}}$ of G_5 and G_6, the computation of $\mathbf{k}_0^\top [\mathbf{c}_1] = [\mathbf{k}_0^\top \mathbf{A}_0]\mathbf{r}_1$ in $\mathcal{O}_{\text{enc}}$ of G_5 can be replaced with $[\mathbf{v}^\top]\mathbf{r}$ for $\mathbf{v}^\top \leftarrow_\$ \mathbb{Z}_q^{1 \times k}$ in G_6. Thus (28) follows.

Table of Contents