A Modular Analysis of the Fujisaki-Okamoto Transformation

Dennis Hofheinz [1] Kathrin Hövelmanns [2] Eike Kiltz [2]

June 23, 2017

[1] Karlsruhe Institute of Technology
Dennis.Hofheinz@kit.edu
[2] Ruhr Universität Bochum
{Kathrin.Hoevelmanns,Eike.Kiltz}@rub.de

Abstract

The Fujisaki-Okamoto (FO) transformation (CRYPTO 1999 and Journal of Cryptology 2013) turns any weakly secure public-key encryption scheme into a strongly (i.e., IND-CCA) secure one in the random oracle model. Unfortunately, the FO analysis suffers from several drawbacks, such as a non-tight security reduction, and the need for a perfectly correct scheme. While several alternatives to the FO transformation have been proposed, they have stronger requirements, or do not obtain all desired properties.

In this work, we provide a fine-grained and modular toolkit of transformations for turning weakly secure into strongly secure public-key encryption schemes. All of our transformations are robust against schemes with correctness errors, and their combination leads to several tradeoffs among tightness of the reduction, efficiency, and the required security level of the used encryption scheme. For instance, one variant of the FO transformation constructs an IND-CCA secure scheme from an IND-CPA secure one with a tight reduction and very small efficiency overhead. Another variant assumes only an OW-CPA secure scheme, but leads to an IND-CCA secure scheme with larger ciphertexts.

We note that we also analyze our transformations in the quantum random oracle model, which yields security guarantees in a post-quantum setting.

Keywords: public-key encryption, Fujisaki-Okamoto transformation, tight reductions, quantum random oracle model

1 Introduction

The notion of INDistinguishability against Chosen-Ciphertext Attacks (IND-CCA) [RS92] is now widely accepted as the standard security notion for asymmetric encryption schemes. Intuitively, IND-CCA security requires that no efficient adversary can recognize which of two messages is encrypted in a given ciphertext, even if the two candidate messages are chosen by the adversary himself. In contrast to the similar but weaker notion of INDistinguishability against Chosen-Plaintext Attacks (IND-CPA), an IND-CCA adversary is given access to a decryption oracle throughout the attack.

Generic Transformations achieving IND-CCA Security. While IND-CCA security is in many applications the desired notion of security, it is usually much more difficult to prove than IND-CPA security. Thus, several transformations have been suggested that turn a public-key encryption (PKE) scheme with weaker security properties into an IND-CCA one generically. For instance, in a seminal paper, Fujisaki and Okamoto [FO99, FO13] proposed a generic transformation (FO transformation) combining any One-Way (OW-CPA) secure asymmetric encryption scheme with any one-time secure symmetric encryption scheme into a Hybrid encryption scheme that is (IND-CCA) secure in the random oracle model [BR93]. Subsequently, Okamoto and Pointcheval [OP01] and Coron et al. [CHJ$^+$02] proposed two more generic transformations (called REACT and GEM) that are considerably simpler but require the underlying asymmetric scheme to be One-Way against Plaintext Checking Attacks (OW-PCA). OW-PCA security is a non-standard security notion that provides the adversary with a plaintext checking oracle

$\textsc{Pco}(c, m)$ that returns 1 iff decryption of ciphertext c yields message m. A similar transformation was also implicitly used in the "Hashed ElGamal" encryption scheme by Abdalla et al. [ABR01].

KEMs. In his "A Designer's Guide to KEMs" paper, Dent [Den03] provides "more modern" versions of the FO [Den03, Table 5] and the REACT/GEM [Den03, Table 2] transformations that result in IND-CCA secure key-encapsulation mechanisms (KEMs). Recall that any IND-CCA secure KEM can be combined with any (one-time) chosen-ciphertext secure symmetric encryption scheme to obtain a IND-CCA secure PKE scheme [CS03]. Due to their efficiency and versatility, in practice one often works with such hybrid encryption schemes derived from a KEM. For that reason the primary goal of our paper will be constructing IND-CCA secure KEMs.

We remark that all previous variants of the FO transformation require the underlying PKE scheme to be γ-spread [FO99], which essentially means that ciphertexts (generated by the probabilistic encryption algorithm) have sufficiently large entropy.

Security against Quantum Adversaries. Recently, the above mentioned generic transformations have gathered renewed interest in the quest of finding an IND-CCA secure asymmetric encryption scheme that is secure against quantum adversaries, i.e., adversaries equipped with a quantum computer. In particular, the NIST announced a competition with the goal to standardize new asymmetric encryption systems [NIS17] with security against quantum adversaries. Natural candidates base their IND-CPA security on the hardness of certain problems over lattices and codes, which are generally believed to resists quantum adversaries. Furthermore, quantum computers may execute all "offline primitives" such as hash functions on arbitrary superpositions, which motivated the introduction of the quantum (accessible) random oracle model [BDF^{+}11]. Targhi and Unruh recently proved a variant of the FO transformation secure in the quantum random oracle model [TU16]. Helping to find IND-CCA secure KEM with provable (post-quantum) security will thus be an important goal in this paper.

Discussion. Despite their versatility, the above FO and REACT/GEM transformations have a couple of small but important disadvantages.

- **Tightness.** The security reduction of the FO transformation [FO99, FO13] in the random oracle model is not tight, i.e., it loses a factor of q_{G}, the number of random oracle queries. A non-tight security proof requires to adapt the system parameters accordingly, which results in considerably less efficient schemes. The REACT/GEM transformations have a tight security reduction, but they require the underlying encryption scheme to be OW-PCA secure. As observed by Peikert [Pei14], due to their decision/search equivalence, many natural lattice-based encryption scheme are not OW-PCA secure and it is not clear how to modify them to be so. In fact, the main technical difficulty is to build an IND-CPA or OW-PCA secure encryption scheme from an OW-CPA secure one, with a tight security reduction.
- **Correctness error.** The FO, as well as the REACT/GEM transformation require the underlying asymmetric encryption scheme to be perfectly correct, i.e., not having a decryption error. In general, one cannot exclude the fact that even a (negligibly) small decryption error could be exploited by an IND-CCA attack against FO-like transformed schemes.[1] As a matter of fact, all known (practical) lattice-based encryption schemes have a small correctness error.[2]

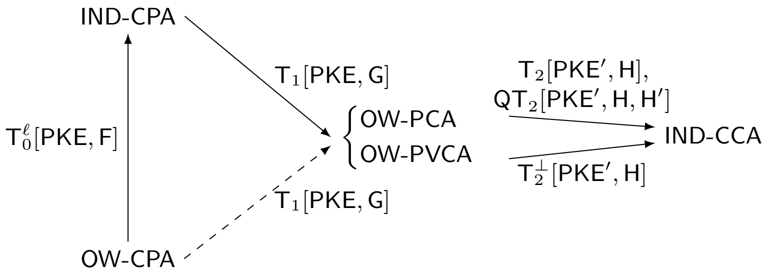
These deficiencies were of little or no concern when the FO and REACT/GEM transformations were originally devised. Due to the emergence of large-scale scenarios (which benefit heavily from tight security reductions) and the increased popularity of lattice-based schemes (with correctness defects), however, we view these deficiencies as acute problems.

1.1 Our contribution

Our main contribution is a modular treatment of FO-like transformations. That is, we provide fine-grained transformations that can be used to turn an OW-CPA secure PKE scheme into an IND-CCA secure one in several steps. For instance, we provide separate OW-CPA $\to$ OW-PCA and OW-PCA $\to$ IND-CCA

[1] It is in fact easy to modify the FO transformation such that it still yields IND-CCA secure encryption but can be attacked if the underlying OW-CPA secure encryption scheme is not perfectly correct.

[2] We remark that while there exist generic transformations (e.g., [DNR04]) that achieve perfect correctness from an almost correct scheme, they are not very efficient and cannot be considered for practical applications.


IND-CPA
$\mathsf{T}_0^\ell[\mathsf{PKE}, \mathsf{F}]$
OW-CPA
$\mathsf{T}_1[\mathsf{PKE}, \mathsf{G}]$
OW-PCA
OW-PVCA
$\mathsf{T}_1[\mathsf{PKE}, \mathsf{G}]$
$\mathsf{T}_2[\mathsf{PKE}', \mathsf{H}]$,
$\mathsf{QT}_2[\mathsf{PKE}', \mathsf{H}, \mathsf{H}']$
$\mathsf{T}_2^\perp[\mathsf{PKE}', \mathsf{H}]$
IND-CCA

Transformation	**Security implication**	**QROM?**	**ROM Tightness?**	**Requirements**
$\mathsf{PKE}' = \mathsf{T}_1[\mathsf{PKE}, \mathsf{G}]$ (§3.1)	OW-CPA ⇒ OW-PCA	✓	—	none
$\mathsf{PKE}' = \mathsf{T}_1[\mathsf{PKE}, \mathsf{G}]$ (§3.1)	IND-CPA ⇒ OW-PCA	✓	✓	none
$\mathsf{PKE}' = \mathsf{T}_1[\mathsf{PKE}, \mathsf{G}]$ (§3.1)	OW-CPA ⇒ OW-PVCA	✓	—	γ-spread
$\mathsf{PKE}' = \mathsf{T}_1[\mathsf{PKE}, \mathsf{G}]$ (§3.1)	IND-CPA ⇒ OW-PVCA	—	✓	γ-spread
$\mathsf{KEM} = \mathsf{T}_2[\mathsf{PKE}', \mathsf{H}]$ (§3.2)	OW-PCA ⇒ IND-CCA	—	✓	none
$\mathsf{KEM} = \mathsf{T}_2^\perp[\mathsf{PKE}', \mathsf{H}]$ (§3.3)	OW-PVCA ⇒ IND-CCA	—	✓	none
$\mathsf{KEM} = \mathsf{QT}_2[\mathsf{PKE}', \mathsf{H}, \mathsf{H}']$ (§4.3)	OW-PCA ⇒ IND-CCA	✓	✓	none
$\mathsf{PKE}_\ell = \mathsf{T}_0^\ell[\mathsf{PKE}, \mathsf{F}]$ (§3.5)	OW-CPA ⇒ IND-PCA	—	✓	none

Figure 1: Our modular transformations. Top: solid errors indicate tight reductions, dashed arrows indicate non-tight reductions. Bottom: properties of the transformations. The tightness row only refers to tightness in the standard random oracle model; all our reduction in the quantum random oracle model are non-tight.

transformations that, taken together, yield the original FO transformation. However, we also provide variants of these individual transformations that achieve different security goals and tightness properties. All of our individual transformations are robust against PKE schemes with correctness errors (in the sense that the correctness error of the resulting schemes can be bounded by the correctness error of the original scheme).

The benefit of our modular treatment is not only a conceptual simplification, but also a larger variety of possible combined transformations (with different requirements and properties). For instance, combining two results about our transformations T_1 and T_2, we can show that the original FO transformation yields IND-CCA security from IND-CPA security with a *tight* security reduction. Combining T_0^ℓ with T_1 and T_2, on the other hand, yields tight IND-CCA security from the weaker notion of OW-CPA security, at the expense of a larger ciphertext. (See Figure 1 for an overview.)

Our transformations in detail. In the following, we give a more detailed overview over our transformations. We remark that all our transformations require a PKE scheme (and not a KEM). We view it as an interesting open problem to construct similar transformations that only assume (and yield) KEMs, since such transformations have the potential of additional efficiency gains.

T_1: from OW-CPA to OW-PCA Security ("Derandomization"+"re-encryption"). Starting from an encryption scheme PKE and a hash function G, we build a deterministic encryption scheme $\mathsf{PKE}' = \mathsf{T}_1[\mathsf{PKE}, \mathsf{G}]$ by defining

$$\mathsf{Enc}'(pk, m) := \mathsf{Enc}(pk, m; \mathsf{G}(m)),$$

where $\mathsf{G}(m)$ is used as the random coins for Enc. Note that Enc' is deterministic. $\mathsf{Dec}'(sk, c)$ first decrypts c into m' and rejects if $\mathsf{Enc}(pk, m'; \mathsf{G}(m')) \neq c$ ("re-encryption"). Modeling G as a random oracle, OW-PCA security of PKE' non-tightly reduces to OW-CPA security of PKE and tightly reduces to IND-CPA security of PKE. If PKE furthermore is γ-spread (for sufficiently large γ), then PKE' is even OW-PVCA secure. OW-PVCA security[3] is essentially PCA security, where the adversary is additionally given access to a validity oracle $\textsc{Val}(c)$ that checks c's validity (cf. Definition 2.1).

[3] OW-PVCA security is called $\mathsf{OW\text{-}CPA}^+$ security with access to a Pco oracle in [Den03].

T_2 ($\mathsf{T}_2^{\perp}$): FROM OW-PCA (OW-PVCA) TO IND-CCA SECURITY ("HASHING"). Starting from an encryption scheme PKE' and a hash function H, we build a key encapsulation mechanism $\mathsf{KEM} = \mathsf{T}_2[\mathsf{PKE}', \mathsf{H}]$ with "implicit rejection" by defining

$$\mathsf{Encaps}(pk) := (c \leftarrow \mathsf{Enc}'(pk, m), K := \mathsf{H}(c, m)),$$

where m is picked at random from the message space.

$$\mathsf{Decaps}(sk, c) = \begin{cases} \mathsf{H}(c, m') & m' \neq \perp \\ \mathsf{H}(c, s) & m' = \perp \end{cases},$$

where $m' := \mathsf{Dec}(sk, c)$ and s is a random seed which is contained in sk. Modeling H as a random oracle, IND-CCA of KEM security tightly reduces to OW-PCA security of PKE'.

We also define $\mathsf{KEM}' = \mathsf{T}_2^{\perp}[\mathsf{PKE}', \mathsf{H}]$ with "explicit rejection" which differs from T_2 only in decapsulation:

$$\mathsf{Decaps}^{\perp}(sk, c) = \begin{cases} \mathsf{H}(c, m') & m' \neq \perp \\ \perp & m' = \perp \end{cases},$$

where $m' := \mathsf{Dec}(sk, c)$. Modeling H as a random oracle, IND-CCA of KEM security tightly reduces to OW-PVCA security of PKE'. We remark that transformation $\mathsf{T}_2^{\perp}$ is essentially [Den03, Table 2], i.e., a KEM variant of the REACT/GEM transformations.

$\mathsf{FO} := \mathsf{T}_2 \circ \mathsf{T}_1$ AND $\mathsf{FO}^{\perp} := \mathsf{T}_2^{\perp} \circ \mathsf{T}_1$: PUTTING THINGS TOGETHER. Our final transformations FO ("FO with implicit rejection") and $\mathsf{FO}^{\perp}$ ("FO with explicit rejection") are defined as

$$\begin{aligned} \mathsf{FO}[\mathsf{PKE}, \mathsf{G}, \mathsf{H}] &:= \mathsf{T}_2[\mathsf{T}_1[\mathsf{PKE}, \mathsf{G}], \mathsf{H}] \\ \mathsf{FO}^{\perp}[\mathsf{PKE}, \mathsf{G}, \mathsf{H}] &:= \mathsf{T}_2^{\perp}[\mathsf{T}_1[\mathsf{PKE}, \mathsf{G}], \mathsf{H}] \ . \end{aligned}$$

As corollaries we obtain that IND-CCA security of $\mathsf{FO}[\mathsf{PKE}, \mathsf{G}, \mathsf{H}]$ and $\mathsf{FO}^{\perp}[\mathsf{PKE}, \mathsf{G}, \mathsf{H}]$ (non-tightly) reduces to the OW-CPA security of PKE, in the random oracle model. The reductions to IND-CPA security are tight. We remark that transformation $\mathsf{FO}^{\perp}$ essentially recovers a KEM variant [Den03, Table 5] of the original FO transformation [FO99]. Whereas $\mathsf{FO}^{\perp}$ requires PKE to be γ-spread, there is no such requirement on FO.

We stress that all our security reduction also take non-zero correctness error into account. Interestingly, the concrete bounds of FO and $\mathsf{FO}^{\perp}$ also give guidance on the required correctness error of the underlying PKE scheme. Concretely for κ bits security, PKE requires a correctness error of $2^{-\kappa}$.

T_0^{ℓ}: FROM OW-CPA TO IND-CPA, TIGHTLY. Note that T_1 requires PKE to be IND-CPA secure to achieve a tight reduction. In case one has to rely on OW-CPA security, transformation T_0^{ℓ} offers the following tradeoff between efficiency and tightness. It transforms an OW-CPA secure PKE into an IND-CPA secure PKE_{ℓ}, where ℓ is a parameter. The ciphertext consists of ℓ independent PKE ciphertexts:

$$\mathsf{Enc}_{\ell}(pk, m) := (\mathsf{Enc}(pk, x_1), \ldots, \mathsf{Enc}(pk, x_{\ell}), m \oplus \mathsf{G}(x_1, \ldots, x_{\ell})).$$

The reduction (to the OW-CPA security of PKE) loses a factor of $q_{\mathsf{G}}^{1/\ell}$, where q_{G} is the number of G-queries an adversary makes.

Observe that the only way to gather information about m is to explicitly query $\mathsf{G}(x_1, \ldots, x_n)$, which requires to find all x_i. The reduction can use this observation to embed an OW-CPA challenge as one $\mathsf{Enc}(pk, x_{i^*})$ and hope to learn x_{i^*} from the G-queries of a successful IND-CPA adversary. In this, the reduction will know all x_i except x_{i^*}. The difficulty in this reduction is to identify the "right" G-query (that reveals x_{i^*}) in all of the adversary's G-queries. Intuitively, the more instances we have, the easier it is for the reduction to spot the G-query $(x_1, \ldots, x_{\ell})$ (by comparing the x_i for $i \neq i^*$), and the less guessing is necessary. Hence, we get a tradeoff between the number of instances ℓ (and thus the size of the ciphertext) and the loss of the reduction.

QT_2: FROM OW-PCA TO IND-CCA SECURITY IN THE QUANTUM ROM. Whereas, as we prove, transformation T_1 also works in the quantum random oracle model, to go from OW-PCA to IND-CCA in the QROM, we build a key encapsulation mechanism $\mathsf{KEM} = \mathsf{QT}_2[\mathsf{PKE}', \mathsf{H}, \mathsf{H}']$ with explicit rejection by defining

$$\mathsf{Encaps}(pk) := ((c \leftarrow \mathsf{Enc}'(pk, m), d := \mathsf{H}'(m)), K := \mathsf{H}(c, m)),$$

where m is picked at random from the message space.

$$\mathsf{Decaps}(sk, c, d) = \begin{cases} \mathsf{H}(c, m') & m' \neq \bot \\ \bot & m' = \bot \vee \mathsf{H}'(m') \neq d \end{cases},$$

where $m' := \mathsf{Dec}(sk, c)$. QT_2 differs from T_2 only in the additional value d from the ciphertext, an idea introduced in [Unr15] and in [TU16] in the context of the FO transformation. Modeling H and H' as a quantum random oracles, $\mathsf{IND\text{-}CCA}$ of KEM security reduces to $\mathsf{OW\text{-}PCA}$ security of PKE'.

Our transformation QFO (Quantum FO with explicit rejection), is defined as

$$\mathsf{QFO}[\mathsf{PKE}, \mathsf{G}, \mathsf{H}, \mathsf{H}'] \quad := \quad \mathsf{QT}_2[\mathsf{T}_1[\mathsf{PKE}, \mathsf{G}], \mathsf{H}, \mathsf{H}']$$

and essentially recovers a KEM variant of the modified FO transformation by Targhi and Unruh [TU16]. As a corollary we obtain that $\mathsf{IND\text{-}CCA}$ security of $\mathsf{QFO}[\mathsf{PKE}, \mathsf{G}, \mathsf{H}, \mathsf{H}']$ reduces to the $\mathsf{OW\text{-}CPA}$ security of PKE, in the quantum random oracle model. As it is common in the quantum random oracle model, all our reductions are non-tight. We leave it as an open problem to derive a tighter security reduction of T_1, for example to $\mathsf{IND\text{-}CPA}$ security of PKE.

Relation to FO/GEM/REACT. As already pointed out, $\mathsf{FO}^{\bot} = \mathsf{T}_2^{\bot} \circ \mathsf{T}_1$ is essentially a KEM variant [Den03] of the Fujisaki-Okamoto transform [FO99] and $\mathsf{T}_2^{\bot}$ is a KEM variant [Den03] of the GEM/REACT transform [OP01, CHJ+02, ABR01]. Our modular view suggest that the FO transform implicitly contains the GEM/REACT transform, at least the proof technique. With this more general view, the FO transform and its variants remains the only known transformation from CPA to CCA security. It is an interesting open problem to come up with alternative transformations that get rid of derandomization or that dispense with re-encryption (which preserving efficiency). Note that for the ElGamal encryption scheme, the “twinning” technique [CKS08, CKS09] does exactly this, but it uses non-generic zero-knowledge proofs that are currently not available for all schemes (e.g., for lattice-based schemes).

Example Instantiations. In the context of ElGamal encryption one can apply $\mathsf{FO}/\mathsf{FO}^{\bot}$ to obtain the schemes of [KML03, BLK00, GMMV05] whose $\mathsf{IND\text{-}CCA}$ security non-tightly reduces to the CDH assumption, and tightly reduces to the DDH assumption. Alternatively, one can directly use $\mathsf{T}_2/\mathsf{T}_2^{\bot}$ to obtain the more efficient schemes of [OP01, CHJ+02, ABR01, Sho04a] whose $\mathsf{IND\text{-}CCA}$ security tightly reduces to the gap-DH (a.k.a. strong CDH) assumption. In the context of deterministic encryption schemes such as RSA, Paillier, etc, one can apply $\mathsf{T}_2/\mathsf{T}_2^{\bot}$ to obtain schemes mentioned in [Sho04a, Den03] whose $\mathsf{IND\text{-}CCA}$ security tightly reduces to the one-way security. Finally, in the context of lattices-based encryption (e.g., [Reg05, LPR13]), one can apply $\mathsf{FO}/\mathsf{FO}^{\bot}$ to obtain $\mathsf{IND\text{-}CCA}$ secure variants.

1.2 Other related work

In concurrent and independent work, [AOP+17] also considers the $\mathsf{IND\text{-}CCA}$ security of $\mathsf{LIMA} := \mathsf{FO}^{\bot}[\mathsf{RLWE}, \mathsf{G}, \mathsf{H}]$, where RLWE is a specific encryption scheme based on lattices associated to polynomial rings from [LPR10], which is $\mathsf{IND\text{-}CPA}$ secure under the Ring-LWE assumption. As the main result, [AOP+17] provides a tight reduction of LIMA's $\mathsf{IND\text{-}CCA}$ security to the Ring-LWE assumption, in the random oracle model. The proof exploits “some weakly homomorphic properties enjoyed by the underlying encryption scheme” and therefore does not seem to be applicable to other schemes. The main result of [AOP+17] is obtained as a special case of our general security results on $\mathsf{FO}^{\bot}$. Furthermore, the security reduction of [AOP+17] does not seem to take the correctness error of RLWE into account.

2 Preliminaries

For $n \in \mathbb{N}$, let $[n] := \{1, \ldots, n\}$. For a set S, $|S|$ denotes the cardinality of S. For a finite set S, we denote the sampling of a uniform random element x by $x \xleftarrow{\$} S$, while we denote the sampling according to some distribution $\mathfrak{D}$ by $x \leftarrow \mathfrak{D}$. For a polynomial $p(X)$ with integer coefficients, we denote by $\mathsf{Roots}(p)$ the (finite) set of (complex) roots of p. By $x =_? y$ we denote the integer that is 1 if $x = y$, and otherwise 0.

Algorithms. We denote deterministic computation of an algorithm A on input x by $y := A(x)$. We denote algorithms with access to an oracle O by A^{O}. Unless stated otherwise, we assume all our algorithms to be probabilistic and denote the computation by $y \leftarrow A(x)$.

```
GAME OW-ATK:
$(pk, sk) \leftarrow \mathsf{Gen}$
$m^* \xleftarrow{\$} \mathcal{M}$
$c^* \leftarrow \mathsf{Enc}(pk, m^*)$
$m' \leftarrow \mathsf{A}^{\mathrm{O}_{\mathsf{ATK}}}(pk, c)$
return $\textsc{Pco}(m', c^*)$

$\textsc{Pco}(m, c)$
if $m \notin \mathcal{M}$
    return $\bot$
else return
    $\mathsf{Dec}(sk, c) =_? m$

$\textsc{Val}(c)$
$m := \mathsf{Dec}(sk, c)$
if $m \in \mathcal{M}$ or $c = c^*$
    return $1$
else return $0$

$\bot$
return $\bot$
```

Figure 2: Games $\mathsf{OW\text{-}ATK}$ ($\mathsf{ATK} \in \{\mathsf{CPA}, \mathsf{PCA}, \mathsf{PVCA}\}$) for PKE, where $\mathrm{O}_{\mathsf{ATK}}$ is defined in Definition 2.1.

Random Oracles. We will at times model hash functions $\mathsf{H} : \mathfrak{D}_{\mathsf{H}} \to \Im(\mathsf{H})$ as random oracles. To keep record of the queries issued to H, we will use a hash list $\mathfrak{L}_{\mathsf{H}}$ that contains all tuples $(x, \mathsf{H}(x))$ of arguments $x \in \mathfrak{D}_{\mathsf{H}}$ that H was queried on and the respective answers $\mathsf{H}(x)$. We make the convention that $\mathsf{H}(x) = \bot$ for all $x \notin \mathfrak{D}_{\mathsf{H}}$.

Games. Following [Sho04b, BR06], we use code-based games. We implicitly assume boolean flags to be initialized to false, numerical types to 0, sets to $\varnothing$, and strings to the empty string ϵ. We make the convention that a procedure terminates once it has returned an output.

2.1 Public-Key Encryption

Syntax. A public-key encryption scheme $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ consists of three algorithms and a finite message space $\mathcal{M}$ (which we assume to be efficiently recognizable). The key generation algorithm Gen outputs a key pair (pk, sk), where pk also defines a randomness space $\mathcal{R} = \mathcal{R}(pk)$. The encryption algorithm Enc, on input pk and a message $m \in \mathcal{M}$, outputs an encryption $c \leftarrow \mathsf{Enc}(pk, m)$ of m under the public key pk. If necessary, we make the used randomness of encryption explicit by writing $c := \mathsf{Enc}(pk, m; r)$, where $r \xleftarrow{\$} \mathcal{R}$ and $\mathcal{R}$ is the randomness space. The decryption algorithm Dec, on input sk and a ciphertext c, outputs either a message $m = \mathsf{Dec}(sk, c) \in \mathcal{M}$ or a special symbol $\bot \notin \mathcal{M}$ to indicate that c is not a valid ciphertext.

We call a public-key encryption scheme PKE *δ-correct* if for all messages $m \in \mathcal{M}$,

$$\Pr\left[\mathsf{Dec}(sk, c) \neq m \mid (pk, sk) \leftarrow \mathsf{Gen}; c \leftarrow \mathsf{Enc}(pk, m)\right] \leq \delta \ .$$

Min-Entropy. [FO13] For $(pk, sk) \leftarrow \mathsf{Gen}$ and $m \in \mathcal{M}$, we define the *min-entropy* of $\mathsf{Enc}(pk, m)$ by $\gamma(pk, m) := -\log \max_{c \in \mathcal{C}} \Pr_{r \leftarrow \mathcal{R}}\left[c = \mathsf{Enc}(pk, m; r)\right]$. We say that PKE is *γ-spread* if, for every key pair $(pk, sk) \leftarrow \mathsf{Gen}$ and every message $m \in \mathcal{M}$, $\gamma(pk, m) \geq \gamma$. In particular, this implies that for every possible ciphertext $c \in \mathcal{C}$, $\Pr_{r \leftarrow \mathcal{R}}\left[c = \mathsf{Enc}(pk, m; r)\right] \leq 2^{-\gamma}$.

Security. We now define three security notions for public-key encryption: One-Wayness under Chosen Plaintext Attacks ($\mathsf{OW\text{-}CPA}$), One-Wayness under Plaintext Checking Attacks ($\mathsf{OW\text{-}PCA}$) and One-Wayness under Plaintext and Validity Checking Attacks ($\mathsf{OW\text{-}PVCA}$).

Definition 2.1 ($\mathsf{OW\text{-}ATK}$). Let $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a public-key encryption scheme with message space $\mathcal{M}$. For $\mathsf{ATK} \in \{\mathsf{CPA}, \mathsf{PCA}, \mathsf{PVCA}\}$, we define $\mathsf{OW\text{-}ATK}$ games as in Figure 2, where

$$\mathrm{O}_{\mathsf{ATK}} := \begin{cases} \bot & \mathsf{ATK} = \mathsf{CPA} \\ \textsc{Pco}(\cdot, \cdot) & \mathsf{ATK} = \mathsf{PCA} \\ \textsc{Pco}(\cdot, \cdot), \textsc{Val}(\cdot) & \mathsf{ATK} = \mathsf{PVCA} \end{cases} .$$

We define the $\mathsf{OW\text{-}ATK}$ *advantage function of an adversary* A *against* PKE as $\mathrm{Adv}_{\mathsf{PKE}}^{\mathsf{OW\text{-}ATK}}(\mathsf{A}) := \Pr[\mathsf{OW\text{-}ATK}_{\mathsf{PKE}}^{\mathsf{A}} \Rightarrow 1]$.

A few remarks are in place. We stress that the plaintext checking oracle $\textsc{Pco}(m, c)$ defined in Figure 3 rejects all queries with $m \notin \mathcal{M}$. This restriction is important since otherwise the validity oracle $\textsc{Val}(m)$could be simulated as $\textsc{Val}(m) = \textsc{Pco}(\bot, c)$. To simplify our notation we sometimes write $\textsc{Pco}(m \notin \mathcal{M}, c)$ to indicate that w.l.o.g. $\textsc{Pco}$ is not queried on $m = \bot$. (Recall that $\mathcal{M}$ is efficiently recognizable.) Similarly, we sometimes write $\textsc{Val}(c \neq c^*)$ to indicate that $\textsc{Val}$ is not queried on c^*.

```
GAME IND-CPA
$(pk, sk) \leftarrow \mathsf{Gen}$
$b \xleftarrow{\$} \{0,1\}$
$(m_0^*, m_1^*, st) \leftarrow \mathsf{A}_1(pk)$
$c^* \leftarrow \mathsf{Enc}(pk, m_b^*)$
$b' \leftarrow \mathsf{A}_2(pk, c^*, st)$
return $b' =_? b$
```

```
GAME IND-CCA
$(pk, sk) \leftarrow \mathsf{Gen}$
$b \xleftarrow{\$} \{0,1\}$
$(K_0^*, c^*) \leftarrow \mathsf{Encaps}(pk)$
$K_1^* \xleftarrow{\$} \mathcal{K}$
$b' \leftarrow \mathsf{A}^{\text{DECAPS}}(c^*, K_b^*)$
return $b' =_? b$
```

```
DECAPS($c$)
if $c = c^*$
    return $\perp$
else
    $K := \mathsf{Decaps}(sk, c)$
    return $K$
```

Figure 3: Games IND-CPA for PKE and IND-CCA game for KEM.

Usually, the adversary wins the one-way game iff its output m' equals the challenge message m^*. Instead, in game OW-ATK the correctness of m' is checked using the PCO oracle, i.e., it returns 1 iff $\mathsf{Dec}(sk, c^*) = m'$. The two games have statistical difference δ, if PKE is δ-correct.

Additionally, we define Indistinguishability under Chosen Plaintext Attacks (IND-CPA).

Definition 2.2 (IND-CPA). Let $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a public-key encryption scheme with message space $\mathcal{M}$. We define the IND-CPA game as in Figure 3, and the IND-CPA advantage function of an adversary $\mathsf{A} = (\mathsf{A}_1, \mathsf{A}_2)$ against PKE (such that A_2 has binary output) as $\mathrm{Adv}_{\mathsf{PKE}}^{\mathsf{IND\text{-}CPA}}(\mathsf{A}) := |\Pr[\mathsf{IND\text{-}CPA}^{\mathsf{A}} \Rightarrow 1] - 1/2|$.

It is well known (see, e.g., [KL07]) that IND-CPA security of PKE with sufficiently large message space implies its OW-CPA security.

Lemma 2.3 *For any adversary* B *there exists an adversary* A *with the same running time as that of* B *such that* $\mathrm{Adv}_{\mathsf{PKE}}^{\mathsf{OW\text{-}PCA}}(\mathsf{B}) \leq \mathrm{Adv}_{\mathsf{PKE}}^{\mathsf{IND\text{-}CPA}}(\mathsf{A}) + 1/|\mathcal{M}|$.

2.2 Key Encapsulation

SYNTAX. A key encapsulation mechanism $\mathsf{KEM} = (\mathsf{Gen}, \mathsf{Encaps}, \mathsf{Decaps})$ consists of three algorithms. The key generation algorithm Gen outputs a key pair (pk, sk), where pk also defines a finite key space $\mathcal{K}$. The encapsulation algorithm Encaps, on input pk, outputs a tuple (K, c) where c is said to be an encapsulation of the key K which is contained in key space $\mathcal{K}$. The deterministic decapsulation algorithm Decaps, on input sk and an encapsulation c, outputs either a key $K := \mathsf{Decaps}(sk, c) \in \mathcal{K}$ or a special symbol $\perp \notin \mathcal{K}$ to indicate that c is not a valid encapsulation. We call KEM *δ-correct* if

$$\Pr\left[\mathsf{Decaps}(sk, c) \neq K \mid (pk, sk) \leftarrow \mathsf{Gen}; (K, c) \leftarrow \mathsf{Encaps}(pk)\right] \leq \delta \ .$$

SECURITY. We now define a security notion for key encapsulation: Indistinguishbility under Chosen Ciphertext Attacks (IND-CCA).

Definition 2.4 (IND-CCA). We define the IND-CCA game as in Figure 3 and the IND-CCA *advantage function of an adversary* A *(with binary output) against* KEM as $\mathrm{Adv}_{\mathsf{KEM}}^{\mathsf{IND\text{-}CCA}}(\mathsf{A}) := |\Pr[\mathsf{IND\text{-}CCA}^{\mathsf{A}} \Rightarrow 1] - 1/2|$.

3 Modular FO Transformation

In Section 3.1, we will introduce T_1 that transforms any OW-CPA secure encryption scheme PKE into a OW-PCA secure encryption scheme PKE′. If PKE is furthermore IND-CPA, then the reduction is tight. Furthermore, if PKE is λ-spread, then PKE′ even satisfied the stronger security notion of OW-PVCA security. Next, in Section 3.2 (Section 3.3), we will introduce transformation T_2 ($\mathsf{T}_2^{\perp}$) that transforms any OW-PCA (OW-PVCA) secure encryption scheme PKE′ into an IND-CCA secure KEM. The security reduction is tight. Combining the above transformations, in Section 3.4 we provide concrete bounds for the IND-CCA security of the resulting $\mathsf{KEM} = \mathsf{FO}[\mathsf{PKE}, \mathsf{G}, \mathsf{H}]$ and $\mathsf{KEM}^{\perp} = \mathsf{FO}^{\perp}[\mathsf{PKE}, \mathsf{G}, \mathsf{H}]$ in the random oracle model. Finally, in Section 3.5 we introduce T_0^{ℓ} that transforms any OW-CPA secure scheme into an IND-CPA secure one, offering a tradeoff between tightness and ciphertext size.

3.1 T_1: from OW-CPA to OW-PCA/PVCA Security

T_1 transforms an OW-CPA secure public-key encryption scheme into an OW-PCA secure one.

The Construction. To a public-key encryption scheme $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ with message space $\mathcal{M}$ and randomness space $\mathcal{R}$, and a hash function $\mathsf{G} : \mathcal{M} \to \mathcal{R}$, we associate $\mathsf{PKE}' = \mathsf{T}_1[\mathsf{PKE}, \mathsf{G}]$. The algorithms of $\mathsf{PKE}' = (\mathsf{Gen}, \mathsf{Enc}', \mathsf{Dec}')$ are defined in Figure 4. Note that Enc' is deterministic.

```
$\mathsf{Enc}'(pk, m)$
$c := \mathsf{Enc}(pk, m; \mathsf{G}(m))$
return $c$
```

```
$\mathsf{Dec}'(sk, c)$
$m' := \mathsf{Dec}(sk, c)$.
if $m' = \perp$ or $\mathsf{Enc}(pk, m'; \mathsf{G}(m')) \neq c$
    return $\perp$
else return $m'$
```

Figure 4: OW-PVCA-secure encryption scheme $\mathsf{PKE}' = \mathsf{T}_1[\mathsf{PKE}, \mathsf{G}]$ with deterministic encryption.

Non-tight Security from OW-CPA. The following theorem establishes that OW-PVCA security of PKE' (cf. Definition 2.1) non-tightly reduces to the OW-CPA security of PKE, in the random oracle model, given that PKE is γ-spread (for sufficiently large γ). If PKE is not γ-spread, then PKE' is still OW-PCA secure.

Theorem 3.1 (PKE OW-CPA $\overset{\text{ROM}}{\Rightarrow}$ PKE' OW-PVCA)**.** *If* PKE *is δ-correct, so is* PKE'. *Assume* PKE *to be γ-spread. Then, for any* OW-PVCA *adversary* B *that issues at most q_{G} queries to the random oracle* G*, q_P queries to a plaintext checking oracle* Pco*, and q_V queries to a validity checking oracle* Val*, there exists an* OW-CPA *adversary* A *such that*

$$\mathrm{Adv}^{\mathsf{OW\text{-}PVCA}}_{\mathsf{PKE}'}(\mathsf{B}) \leq q_P \cdot \delta + q_V \cdot 2^{-\gamma} + (q_{\mathsf{G}} + 1) \cdot \mathrm{Adv}^{\mathsf{OW\text{-}CPA}}_{\mathsf{PKE}}(\mathsf{A})$$

and the running time of A *is about that of* B.

Proof. We first prove δ correctness of PKE'. For every fixed $m \in \mathcal{M}$,

$$\begin{aligned} &\Pr\left[\mathsf{Dec}(sk, c) \neq m \mid (pk, sk) \leftarrow \mathsf{Gen}; r := \mathsf{G}(m); c \ = \mathsf{Enc}(pk, m; r)\right] \\ &= \Pr\left[\mathsf{Dec}(sk, c) \neq m \mid (pk, sk) \leftarrow \mathsf{Gen}; r \xleftarrow{\$} \mathcal{R}; c \ = \mathsf{Enc}(pk, m; r)\right] \leq \delta \ , \end{aligned}$$

where the first equality comes from the fact that G is a random oracle.

Let B be an adversary against the OW-PVCA security of PKE', issuing at most q_{G} queries to G, at most q_P queries to Pco and at most q_V queries to Val. Consider the sequence of games given in Figure 5.

Game G_0. This is the original OW-PVCA game. Random oracle queries are stored in set $\mathfrak{L}_G$ with the convention that $\mathsf{G}(m) = r$ iff $(m, r) \in \mathfrak{L}_G$. Hence,

$$\Pr[G_0^{\mathsf{B}} \Rightarrow 1] = \mathrm{Adv}^{\mathsf{OW\text{-}PVCA}}_{\mathsf{PKE}'}(\mathsf{B}) \ .$$

Game G_1. In game G_1 the plaintext checking oracle $\textsc{Pco}(m, c)$ is replaced with a simulation that does not make use of the secret key, by returning whether $\mathsf{Enc}(pk, m, \mathsf{G}(m)) = c$ instead of whether $\mathsf{Dec}'(sk, c) = m$. Clearly, $\textsc{Pco}(m, c) = 1$ in game G_0 implies $\textsc{Pco}(m, c) = 1$ in game G_1. Moreover, since PKE is δ-correct and G is a random oracle, $\textsc{Pco}(m, c) = 1$ in game G_1 implies $\textsc{Pco}(m, c) = 1$ in game G_0 with probability at least $1 - \delta$. By the union bound we obtain

$$|\Pr[G_1^{\mathsf{B}} \Rightarrow 1] - \Pr[G_0^{\mathsf{B}} \Rightarrow 1]| \leq q_P \cdot \delta. \tag{1}$$

Game G_2. In game G_2, the validity checking oracle $\textsc{Val}(c \neq c^*)$ is replaced with one that returns 1 iff there exists a previous query m to G such that $\mathsf{Enc}(pk, m; \mathsf{G}(m)) = c$. Clearly, $\textsc{Val}(c) = 1$ in game G_2 implies that $\textsc{Val}(c) = 1$ in Game G_1. Therefore, the games only differ when B queries Val on $c = \mathsf{Enc}(pk, m; \mathsf{G}(m))$ in game G_2 without first having queried G on m. For each individual query $\textsc{Val}(c)$ his can only happen with probability $2^{-\gamma}$, where γ is the parameter from the γ-spreadness of PKE. By the union bound we obtain

$$|\Pr[G_2^{\mathsf{B}} \Rightarrow 1] - \Pr[G_1^{\mathsf{B}} \Rightarrow 1]| \leq q_V \cdot 2^{-\gamma}.$$

```
GAMES $G_0$-$G_3$
01 $(pk, sk) \leftarrow \mathsf{Gen}$
02 $m^* \xleftarrow{\$} \mathcal{M}$
03 $c^* \leftarrow \mathsf{Enc}(pk, m^*)$
04 $m' \leftarrow \mathsf{B}^{\mathsf{G},\textsc{Pco},\textsc{Val}}(pk, c^*)$.
05 return $m' =_? m^*$

$\mathsf{G}(m)$
06 if $\exists r$ s. th.$(m, r) \in \mathfrak{L}_G$
07     return $r$
08 if $m = m^*$                          // $G_3$
09     QUERY := true                     // $G_3$
10     abort                             // $G_3$
11 $r \xleftarrow{\$} \mathcal{R}$
12 $\mathfrak{L}_G := \mathfrak{L}_G \cup \{(m, r)\}$
13 return $r$

$\textsc{Pco}(m \in \mathcal{M}, c)$
14 $m' := \mathsf{Dec}(sk, c)$                                   // $G_0$
15 if $m' = m$ and $\mathsf{Enc}(pk, m'; \mathsf{G}(m')) = c$   // $G_0$
16     return 1                                                 // $G_0$
17 else return 0                                                // $G_0$
18 return $\mathsf{Enc}(pk, m, \mathsf{G}(m)) =_? c$            // $G_1$-$G_3$

$\textsc{Val}(c \neq c^*)$                                      // $G_0$-$G_1$
19 $m' := \mathsf{Dec}(sk, c)$
20 if $m \notin \mathcal{M}$
21     return 0
22 else return 1

$\textsc{Val}(c \neq c^*)$                                      // $G_2$-$G_3$
23 if $\exists r$ s. th. $(m, r) \in \mathfrak{L}_G$
     and $\mathsf{Enc}(pk, m; \mathsf{G}(m)) = c$
24     return 1
25 else return 0
```

Figure 5: Games G_0 - G_3 for the proof of Theorem 3.1.

```
$\mathsf{D}(pk, c^*)$
$m \leftarrow \mathsf{B}^{\mathsf{G},\textsc{Pco}}(pk, c^*)$
$(m', r') \xleftarrow{\$} \mathfrak{L}_G$
return $m'$

$\mathsf{C}(pk, c^*)$
$m' \leftarrow \mathsf{B}^{\mathsf{G},\textsc{Pco}}(pk, c^*)$
return $m'$

$\mathsf{G}(m)$
if $\exists r$ s. th. $(m, r) \in \mathfrak{L}_G$
    return $r$
$r \xleftarrow{\$} \mathcal{R}$
$\mathfrak{L}_G := \mathfrak{L}_G \cup \{(m, r)\}$
return $r$
```

Figure 6: Adversaries C and D for the proof of Theorem 3.1. Oracles Pco and Val are defined as in game G_3 of Figure 5.

Game G_3. In Game G_3, we add a flag QUERY in line 09 and abort when it is raised. Hence, G_2 and G_3 only differ if QUERY is raised, meaning that B queried G on m^*, meaning $(m^*, \cdot) \in \mathfrak{L}_G$. Due to the difference lemma [Sho04b],

$$|\Pr[G_3^{\mathsf{B}} \Rightarrow 1] - \Pr[G_2^{\mathsf{B}} \Rightarrow 1]| \leq \Pr[\text{QUERY}].$$

We bound $\Pr[G_3^{\mathsf{B}} \Rightarrow 1]$ by constructing an adversary C in Figure 6 against the OW-CPA security of the original encryption scheme PKE. C inputs $(pk, c^* \leftarrow \mathsf{Enc}(pk, m^*))$, perfectly simulates game G_3 for B, and finally outputs $m' = m^*$ if B wins in game G_3.

$$\Pr[G_3^{\mathsf{B}} \Rightarrow 1] = \mathrm{Adv}_{\mathsf{PKE}}^{\mathsf{OW\text{-}CPA}}(\mathsf{C}) \ .$$

So far we have established the bound

$$\mathrm{Adv}_{\mathsf{PKE'}}^{\mathsf{OW\text{-}PVCA}}(\mathsf{B}) \leq q_P \cdot \delta + q_V \cdot 2^{-\gamma} + \Pr[\text{QUERY}] + \mathrm{Adv}_{\mathsf{PKE}}^{\mathsf{OW\text{-}CPA}}(\mathsf{C}) \ . \qquad (2)$$

Finally, in Figure 6 we construct an adversary D against the OW-CPA security of the original encryption scheme PKE, that inputs $(pk, c^* \leftarrow \mathsf{Enc}(pk, m^*))$, perfectly simulates game G_3 for B. If flag QUERY is set in G_3 then there exists en entry $(m^*, \cdot) \in \mathfrak{L}_{\mathsf{G}}$ and D returns the correct $m' = m^*$ with probability $1/q_{\mathsf{G}}$. We just showed

$$\Pr[\text{QUERY}] \leq q_{\mathsf{G}} \cdot \mathrm{Adv}_{\mathsf{PKE}}^{\mathsf{OW\text{-}CPA}}(\mathsf{D}) \ .$$

Combining the latter bound with Equation (2) and folding C and D into one single adversary A against OW-CPA yields the required bound. □

$\mathsf{OW\text{-}PCA}$ security is $\mathsf{OW\text{-}PVCA}$ security with $q_V := 0$ queries to the validity checking oracle. Hence, the bound of Theorem 3.1 shows that PKE' is in particular $\mathsf{OW\text{-}PCA}$ secure, without requiring PKE to be γ-spread.

Tight Security from $\mathsf{IND\text{-}CPA}$. Whereas the reduction to $\mathsf{OW\text{-}CPA}$ security in Theorem 3.1 was non-tight, the following theorem establishes that $\mathsf{OW\text{-}PVCA}$ security of PKE' tightly reduces to the $\mathsf{IND\text{-}CPA}$ security of PKE, in the random oracle model, given that PKE is γ-spread. If PKE is not γ-spread, then PKE' is still $\mathsf{OW\text{-}PCA}$ secure.

Theorem 3.2 (PKE $\mathsf{IND\text{-}CPA}$ $\overset{\text{ROM}}{\Rightarrow}$ PKE' $\mathsf{OW\text{-}PVCA}$). *Assume* PKE *to be* δ*-correct and* γ*-spread. Then, for any* $\mathsf{OW\text{-}PVCA}$ *adversary* B *that issues at most* q_{G} *queries to the random oracle* G*,* q_P *queries to a plaintext checking oracle* $\textsc{Pco}$*, and* q_V *queries to a validity checking oracle* $\textsc{Val}$*, there exists an* $\mathsf{IND\text{-}CPA}$ *adversary* A *such that*

$$\mathrm{Adv}_{\mathsf{PKE}'}^{\mathsf{OW\text{-}PVCA}}(\mathsf{B}) \leq q_P \cdot \delta + q_V \cdot 2^{-\gamma} + \frac{2q_{\mathsf{G}}+1}{|\mathcal{M}|} + 3 \cdot \mathrm{Adv}_{\mathsf{PKE}}^{\mathsf{IND\text{-}CPA}}(\mathsf{A})$$

and the running time of A *is about that of* B.

Proof. Considering the games of Figure 5 from the proof of Theorem 3.1 we obtain by Equation (2)

$$\begin{aligned}\mathrm{Adv}_{\mathsf{PKE}'}^{\mathsf{OW\text{-}PVCA}}(\mathsf{B}) &\leq q_P \cdot \delta + q_V \cdot 2^{-\gamma} + \Pr[\mathrm{QUERY}] + \mathrm{Adv}_{\mathsf{PKE}}^{\mathsf{OW\text{-}CPA}}(\mathsf{D}) \\ &\leq q_P \cdot \delta + q_V \cdot 2^{-\gamma} + \Pr[\mathrm{QUERY}] + \frac{1}{|\mathcal{M}|} + \mathrm{Adv}_{\mathsf{PKE}}^{\mathsf{IND\text{-}CPA}}(\mathsf{D}) \ ,\end{aligned}$$

where the last inequation uses Lemma 2.3.

In Figure 7 we construct an adversary $\mathsf{D} = (\mathsf{D}_1, \mathsf{D}_2)$ against the $\mathsf{IND\text{-}CPA}$ security of the original encryption scheme PKE that wins if flag QUERY is set in G_3. The first adversary D_1 picks two random messages m_0^*, m_1^*. The second adversary D_2 inputs $(pk, c^* \leftarrow \mathsf{Enc}(pk, m_b^*))$, for a random bit b, and runs $\mathsf{B}(pk, c^*)$, simulating its view in game G_3. Note that by construction message m_b^* is uniformly distributed.

Consider game $\mathsf{IND\text{-}CPA}^{\mathsf{D}}$. Let BADG be the event that B queries random oracle G on m_{1-b}^*. Since m_{1-b}^* is uniformly distributed and independent from B's view, we have $\Pr[\mathrm{BADG}] \leq q_G/|\mathcal{M}|$. For the remainder of the proof we assume BADG did not happen, i.e. $|\mathfrak{L}_G(m_{1-b}^*)| = 0$.

If QUERY happens, then B queried the random oracle G on m_b^*, which implies $|\mathfrak{L}_G(m_b^*)| > 0 = |\mathfrak{L}_G(m_{1-b}^*)|$ and therefore $b = b'$. If QUERY does not happen, then B did not query random oracle G on m_b^*. Hence, $|\mathfrak{L}_G(m_b^*)| = |\mathfrak{L}_G(m_{1-b}^*)| = 0$ and $\Pr[b = b'] = 1/2$ since A picks a random bit b'. Overall, we have

$$\begin{aligned}\mathrm{Adv}_{\mathsf{PKE}}^{\mathsf{IND\text{-}CPA}}(\mathsf{D}) + \frac{q_{\mathsf{G}}}{|\mathcal{M}|} &\geq \left|\Pr[b = b'] - \frac{1}{2}\right| \\ &= \left|\Pr[\mathrm{QUERY}] + \frac{1}{2}\Pr[\neg\mathrm{QUERY}] - \frac{1}{2}\right| \\ &= \frac{1}{2}\Pr[\mathrm{QUERY}].\end{aligned}$$

Folding C and D into one single $\mathsf{IND\text{-}CPA}$ adversary A yields the required bound. □

With the same argument as in Theorem 3.1, a tight reduction to $\mathsf{OW\text{-}PCA}$ security is implied without requiring PKE to be γ-spread.

3.2 T_2: from $\mathsf{OW\text{-}PCA}$ to $\mathsf{IND\text{-}CCA}$ Security

T_2 transforms any $\mathsf{OW\text{-}PCA}$ secure public-key encryption scheme (e.g., one obtained via T_1 from Section 3.1) into an $\mathsf{IND\text{-}CCA}$ secure key encapsulation mechanism.

The Construction. To a public-key encryption scheme $\mathsf{PKE}' = (\mathsf{Gen}', \mathsf{Enc}', \mathsf{Dec}')$ with message space $\mathcal{M}$, and a hash function $\mathsf{H} : \{0,1\}^* \to \mathcal{M}$ we associate $\mathsf{KEM} = \mathsf{T}_2[\mathsf{PKE}', \mathsf{H}]$. The algorithms of $\mathsf{KEM} = (\mathsf{Gen}, \mathsf{Encaps}, \mathsf{Decaps})$ are defined in Figure 8.

Security. The following theorem establishes that $\mathsf{IND\text{-}CCA}$ security of KEM tightly reduces to the $\mathsf{OW\text{-}PCA}$ security of PKE', in the random oracle model.

```
$\mathsf{D}_1(pk)$
11 $st := (m_0^*, m_1^*) \xleftarrow{\$} \mathcal{M}^2$
12 return $st$
$\mathsf{D}_2(pk, c^*, st)$
13 $m' \leftarrow \mathsf{B}^{\mathsf{G},\text{PCO},\text{VAL}}(pk, c^*)$
14 $b' := \begin{cases} 0 & |\mathfrak{L}_G(m_0^*)| > |\mathfrak{L}_G(m_1^*)| \\ 1 & |\mathfrak{L}_G(m_1^*)| < |\mathfrak{L}_G(m_0^*)| \\ \xleftarrow{\$} \{0,1\} & \text{otherwise} \end{cases}$
15 return $b'$

$\mathsf{G}(m)$
16 if $\exists r$ s. th. $(m, r) \in \mathfrak{L}_G$
17     return $r$
18 $r \xleftarrow{\$} \mathcal{R}$
19 $\mathfrak{L}_G := \mathfrak{L}_G \cup \{(m, r)\}$
20 return $r$
```

Figure 7: Adversary $\mathsf{D} = (\mathsf{D}_1, \mathsf{D}_2)$ for the proof of Theorem 3.2. For fixed $m \in \mathcal{M}$, $\mathfrak{L}_G(m)$ is the set of all $(m, r) \in \mathfrak{L}_G$. Oracles PCO and VAL are defined as in game G_3 of Figure 5.

```
$\mathsf{Gen}$
01 $(pk', sk') \leftarrow \mathsf{Gen}'$
02 $s \xleftarrow{\$} \mathcal{M}$
03 $sk := (sk', s)$
04 return $(pk', sk)$

$\mathsf{Encaps}(pk)$
05 $m \xleftarrow{\$} \mathcal{M}$
06 $c \leftarrow \mathsf{Enc}'(pk, m)$
07 $K := \mathsf{H}(m, c)$
08 return $(K, c)$

$\mathsf{Decaps}(sk, c)$
09 Parse $sk = (sk', s)$
10 $m' := \mathsf{Dec}'(sk', c)$
11 if $m' \neq \bot$
12     return $K := \mathsf{H}(m', c)$
13 else return $K := \mathsf{H}(s, c)$
```

Figure 8: IND-CCA-secure key encapsulation mechanism $\mathsf{KEM} = \mathsf{T}_2[\mathsf{PKE}', \mathsf{H}]$.

Theorem 3.3 (PKE′ OW-PCA ⇒ KEM IND-CCA). *If* PKE′ *is δ-correct, so is* KEM. *For any* IND-CCA *adversary* B *against* KEM*, issuing at most q_D queries to the decapsulation oracle* DECAPS *and at most q_{H} queries to the random oracle* H*, there exists an* OW-PCA *adversary* A *against* PKE′ *that makes at most q_{H} queries to the* PCO *oracle such that*

$$\mathrm{Adv}_{\mathsf{KEM}}^{\mathsf{IND\text{-}CCA}}(\mathsf{B}) \leq \frac{q_{\mathsf{H}}}{|\mathcal{M}|} + \mathrm{Adv}_{\mathsf{PKE}'}^{\mathsf{OW\text{-}PCA}}(\mathsf{A})$$

and the running time of A *is about that of* B.

Proof. It is easy to verify the correctness bound. Let B be an adversary against the IND-CCA security of KEM, issuing at most q_D queries to DEC and at most q_{H} queries to H. Consider the games given in Figure 9.

GAME G_0. Since game G_0 is the original IND-CCA game,

$$\left|\Pr[G_0^{\mathsf{B}} \Rightarrow 1] - \frac{1}{2}\right| = \mathrm{Adv}_{\mathsf{KEM}}^{\mathsf{IND\text{-}CCA}}(\mathsf{B}) \ .$$

GAME G_1. In game G_1 we make two changes. In DECAPS, we replace $K = \mathsf{H}(s, c)$ by $K = \mathsf{H}'(c)$ and $K = \mathsf{H}''(c)$, respectively, whenever $m' \in \{\bot, s\}$ (lines 13 and 14), where H' and H'' are independent internal random oracles that cannot be accessed by B. This remains unnoticed by B unless $\mathsf{H}(s, \cdot)$ is queried, in which case G_1 aborts (lines 18 and 19). Since B's view is independent of (the uniform secret) s unless G_1 aborts,

$$|\Pr[G_1^{\mathsf{B}} \Rightarrow 1] - \Pr[G_0^{\mathsf{B}} \Rightarrow 1]| \leq \frac{q_{\mathsf{H}}}{|\mathcal{M}|} \ .$$

GAME G_2. In game G_2, the oracles H and DECAPS are changed such that DECAPS does not make use of the secret key any longer except by testing if $\mathsf{Dec}'(sk', c) = m$ for given (m, c) in line 20. In game G_2 we will use two lists, $\mathfrak{L}_H$ and $\mathfrak{L}_D$, for bookkeeping. $(m, c, K) \in \mathfrak{L}_H$ indicates that H was queried on (m, c) and $\mathsf{H}(m, c) = K$ holds; $(c, K) \in \mathfrak{L}_D$ indicates that $\text{DECAPS}(c) = K$ holds and either H was queried on $(m := \mathsf{Dec}'(sk', c), c)$ or DECAPS was queried on c. In order to show that the view of B is identical in games G_1 and G_2, consider the following cases for a fixed ciphertext c and $m' := \mathsf{Dec}'(sk', c)$.

```
GAMES $G_0$ - $G_3$
01 $(pk', sk') \leftarrow \mathsf{Gen}'$
02 $s \xleftarrow{\$} \mathcal{M}$
03 $sk := (sk', s)$
04 $m^* \xleftarrow{\$} \mathcal{M}$
05 $c^* \leftarrow \mathsf{Enc}'(pk, m^*)$
06 $K_0^* := \mathsf{H}(m^*, c^*)$
07 $K_1^* \xleftarrow{\$} \{0,1\}^n$
08 $b \xleftarrow{\$} \{0,1\}$
09 $b' \leftarrow \mathsf{B}^{\textsc{Decaps},\mathsf{H}}(pk', c^*, K_b^*)$
10 return $b' =_? b$

Decaps($c \neq c^*$)                                   // $G_0$-$G_1$
11 $m' := \mathsf{Dec}'(sk', c)$
12 if $m' = \perp$ return $K := \mathsf{H}(s, c)$          // $G_0$
13 if $m' = \perp$ return $K := \mathsf{H}'(c)$            // $G_1$
14 if $m' = s$ return $K := \mathsf{H}''(c)$               // $G_1$
15 return $K := \mathsf{H}(m, c)$

$\mathsf{H}(m, c)$
16 if $\exists K$ s. th. $(m, c, K) \in \mathfrak{L}_H$ return $K$
17 $K \xleftarrow{\$} \mathcal{K}$
18 if $m = s$                                          // $G_1$- $G_3$
19    QUERY := true; abort                             // $G_1$- $G_3$
20 if $\mathsf{Dec}'(sk', c) = m$                      // $G_2$- $G_3$
21    if $c = c^*$                                     // $G_3$
22       CHAL := true; abort                           // $G_3$
23    if $\exists K'$ such that $(c, K') \in \mathfrak{L}_D$   // $G_2$- $G_3$
24       $K := K'$                                     // $G_2$- $G_3$
25    else                                             // $G_2$- $G_3$
26       $\mathfrak{L}_D := \mathfrak{L}_D \cup \{(c, K)\}$     // $G_2$- $G_3$
27 $\mathfrak{L}_H := \mathfrak{L}_H \cup \{(m, c, K)\}$
28 return $K$

Decaps($c \neq c^*$)                                   // $G_2$- $G_3$
29 if $\exists K$ s. th. $(c, K) \in \mathfrak{L}_D$
30    return $K$
31 else
32    $K \xleftarrow{\$} \mathcal{K}$
33    $\mathfrak{L}_D := \mathfrak{L}_D \cup \{(c, K)\}$
34    return $K$
```

Figure 9: Games G_0 - G_3 for the proof of Theorem 3.3 . H' (line 13) and H'' (14) are independent internal random oracles that cannot be accessed by B.

- Case 1: $m' \in \{\perp, s\}$. Since H cannot be queried on (m', c) (i.e., $\mathsf{H}(\perp, \cdot)$ is not allowed and $\mathsf{H}(s, c)$ results in abort), there is no message $m \in \mathcal{M}$ such that $\mathsf{H}(m, c)$ could have added a tuple (c, K) to $\mathfrak{L}_D$. Hence, querying Decaps(c) in game G_2 will return a uniformly random key, as in Game G_1.
- Case 2: $m' \notin \{\perp, s\}$. We will now show that H in game G_2 is "patched", meaning that it is ensures Decaps$(c) = \mathsf{H}(m', c)$, where $m' := \mathsf{Dec}'(sk', c)$, for all valid ciphertexts c with $\mathsf{Dec}'(sk', c) \neq s$. We distinguish two sub-cases: B might either first query H on (m', c), then Decaps on c, or the other way round.
  - If H is queried on (m', c) first, it is recognized that $\mathsf{Dec}'(sk', c) = m$ in line 20. Since Decaps was yet not queried on c, no entry of the form (c, K) already exists in $\mathfrak{L}_D$. Therefore, besides adding $(m, c, K \xleftarrow{\$} \mathcal{K})$ to $\mathfrak{L}_H$, H also adds (c, K) to $\mathfrak{L}_D$ in line 26, thereby defining Decaps$(c) := K = \mathsf{H}(m', c)$.
  - If Decaps is queried on c first, no entry of the form (c, K) exists in $\mathfrak{L}_D$ yet. Therefore, Decaps adds $(c, K \xleftarrow{\$} \mathcal{K})$ to $\mathfrak{L}_D$ thereby defining Decaps$(c) := K$. When queried on (m', c) afterwards, H recognizes that $\mathsf{Dec}'(sk', c) = m$ in line 20 and that an entry of the form (c, K) already exists in $\mathfrak{L}_D$ in line 23. By adding (m, c, K) to $\mathfrak{L}_H$ and returning K, H defines $\mathsf{H}(m', c) := K = $ Decaps(c).

We have shown that B's view is identical in both games and

$$\Pr[G_2^{\mathsf{B}} \Rightarrow 1] = \Pr[G_1^{\mathsf{B}} \Rightarrow 1]| \ .$$

Game G_3. In game G_3, we abort immediately (and raise flag CHAL) on the event that B queries H on (m^*, c^*), where m^* is the challenge message. Due to the difference lemma,

$$|\Pr[G_3^{\mathsf{B}} \Rightarrow 1] - \Pr[G_2^{\mathsf{B}} \Rightarrow 1]| \leq \Pr[\mathrm{CHAL}] \ .$$

In game G_3, $\mathsf{H}(m^*, c^*)$ will not be given to B; neither through a hash nor a decryption query, meaning bit b is independent from B's view. Hence,

$$\Pr[G_3^{\mathsf{B}}] = \frac{1}{2} \ .$$

```
$\mathsf{A}^{\text{PCO}}(pk, c^*)$
$K^* \xleftarrow{\$} \mathcal{K}$
$s \xleftarrow{\$} \mathcal{M}$
$b' \leftarrow \mathsf{B}^{\text{DECAPS},\mathsf{H}}(pk, c^*, K^*)$
if $\exists (m', c', K') \in \mathfrak{L}_H$
  s. th. $\text{PCO}(m', c^*) = 1$
    return $m'$
else
    abort
```

```
$\mathsf{H}(m, c)$
if $\exists K$ s. th. $(m, c, K) \in \mathfrak{L}_H$ return $K$
$K \xleftarrow{\$} \mathcal{K}$
if $m = s$
    abort
if $\text{PCO}(m, c) = 1$
    if $\exists K'$ s. th. $(c, K') \in \mathfrak{L}_D$
        $K := K'$
    else
        $\mathfrak{L}_D := \mathfrak{L}_D \cup \{(c, K)\}$
$\mathfrak{L}_H := \mathfrak{L}_H \cup \{(m, c, K)\}$
return $K$
```

Figure 10: Adversary A for the proof of Theorem 3.3. Oracle DECAPS is defined as in game G_3 of Figure 9.

It remains to bound Pr[CHAL]. To this end, we construct an adversary A against the $\mathsf{OW\text{-}PCA}$ security of PKE' simulating G_3 for B as in Figure 10. Note that the simulation is perfect. Since CHAL implies that B queried $\mathsf{H}(m^*, c^*)$ which implies $(m^*, c^*, K') \in \mathfrak{L}_H$ (for some K'), A returns $m' = m^*$. Hence,

$$\Pr[\text{CHAL}] = \text{Adv}_{\mathsf{PKE}}^{\mathsf{OW\text{-}PCA}}(\mathsf{A}) \ .$$

Collecting the probabilities yields the required bound. □

3.3 $\mathsf{T}_2^{\perp}$: from OW-PVCA to IND-CCA Security

$\mathsf{T}_2^{\perp}$ is a variant of T_2 with explicit rejection. It transforms an $\mathsf{OW\text{-}PVCA}$ secure public-key encryption scheme (e.g., the ones obtained via T_1 from Section 3.1) into an $\mathsf{IND\text{-}CCA}$ secure key encapsulation mechanism.

THE CONSTRUCTION. To a public-key encryption scheme $\mathsf{PKE}' = (\mathsf{Gen}', \mathsf{Enc}', \mathsf{Dec}')$ with message space $\mathcal{M}$, and a hash function $\mathsf{H} : \{0,1\}^* \to \{0,1\}^n$, we associate $\mathsf{KEM}^{\perp} = \mathsf{T}_2^{\perp}[\mathsf{PKE}', \mathsf{H}]$. The algorithms of $\mathsf{KEM}^{\perp} = (\mathsf{Gen}', \mathsf{Encaps}, \mathsf{Decaps}^{\perp})$ are defined in Figure 11. Note that $\mathsf{T}_2^{\perp}$ and T_2 essentially differ in decapsulation: $\mathsf{Decaps}^{\perp}$ from $\mathsf{T}_2^{\perp}$ rejects if c decrypts to $\perp$, whereas Decaps from T_2 returns a pseudorandom key K.

```
$\mathsf{Encaps}(pk)$
$m \xleftarrow{\$} \mathcal{M}$
$c \leftarrow \mathsf{Enc}'(pk, m)$
$K := \mathsf{H}(m, c)$
return $(K, c)$
```

```
$\mathsf{Decaps}^{\perp}(sk, c)$
$m' := \mathsf{Dec}'(sk, c)$
if $m' = \perp$ return $\perp$
else return
    $K := \mathsf{H}(m', c)$
```

Figure 11: $\mathsf{IND\text{-}CCA}$-secure key encapsulation mechanism $\mathsf{KEM}^{\perp} = \mathsf{T}_2^{\perp}[\mathsf{PKE}', \mathsf{H}]$.

SECURITY. The following theorem establishes that $\mathsf{IND\text{-}CCA}$ security of $\mathsf{KEM}^{\perp}$ tightly reduces to the $\mathsf{OW\text{-}PVCA}$ security of PKE', in the random oracle model.

Theorem 3.4 (PKE' $\mathsf{OW\text{-}PVCA}$ $\overset{\text{ROM}}{\Rightarrow}$ $\mathsf{KEM}^{\perp}$ $\mathsf{IND\text{-}CCA}$). *If PKE' is δ-correct, so is $\mathsf{KEM}^{\perp}$. For any $\mathsf{IND\text{-}CCA}$ adversary B against $\mathsf{KEM}^{\perp}$, issuing at most q_D queries to the decapsulation oracle $\text{DECAPS}^{\perp}$ and at most q_{H} queries to the random oracle H, there exists an $\mathsf{OW\text{-}PVCA}$ adversary A against PKE' that makes at most q_{H} queries both to the PCO oracle and to the VAL oracle such that*

$$\text{Adv}_{\mathsf{KEM}^{\perp}}^{\mathsf{IND\text{-}CCA}}(\mathsf{B}) \leq \text{Adv}_{\mathsf{PKE}'}^{\mathsf{OW\text{-}PVCA}}(\mathsf{A})$$

and the running time of A is about that of B.

```
GAMES $G_0$ - $G_2$
01 $(pk, sk) \leftarrow \mathsf{Gen}'$
02 $m^* \xleftarrow{\$} \mathcal{M}$
03 $c^* \leftarrow \mathsf{Enc}'(pk, m^*)$
04 $K_0^* := \mathsf{H}(m^*, c^*)$
05 $K_1^* \xleftarrow{\$} \{0,1\}^n$
06 $b \xleftarrow{\$} \{0,1\}$
07 $b' \leftarrow \mathsf{B}^{\textsc{Decaps}^\perp, \mathsf{H}}(pk, c^*, K_b^*)$
08 return $b' =_? b$
```

```
$\mathsf{H}(m, c)$
12 if $\exists K$ such that $(m, c, K) \in \mathfrak{L}_H$
13   return $K$
14 $K \xleftarrow{\$} \mathcal{K}$
15 if $\mathsf{Dec}'(sk, c) = m$                          // $G_1$-$G_2$
16   if $c = c^*$                                        // $G_2$
17     CHAL := true                                      // $G_2$
18     abort                                             // $G_2$
19   if $\exists K'$ such that $(c, K') \in \mathfrak{L}_D$   // $G_1$-$G_2$
20     $K := K'$                                         // $G_1$-$G_2$
21   else                                                // $G_1$-$G_2$
22     $\mathfrak{L}_D := \mathfrak{L}_D \cup \{(c, K)\}$        // $G_1$-$G_2$
23 $\mathfrak{L}_H := \mathfrak{L}_H \cup \{(m, c, K)\}$
24 return $K$
```

```
$\textsc{Decaps}^\perp(c \neq c^*)$                      // $G_0$
09 $m' := \mathsf{Dec}'(sk, c)$
10 if $m' = \perp$ return $\perp$
11 return $K := \mathsf{H}(m', c)$
```

```
$\textsc{Decaps}^\perp(c \neq c^*)$                      // $G_1$-$G_2$
25 if $\exists K$ s. th. $(c, K) \in \mathfrak{L}_D$
26   return $K$
27 if $\textsc{Val}(c) = 0$
28   return $\perp$
29 $K \xleftarrow{\$} \mathcal{K}$
30 $\mathfrak{L}_D := \mathfrak{L}_D \cup \{(c, K)\}$
31 return $K$
```

Figure 12: Games G_0 - G_2 for the proof of Theorem 3.4

Proof. It is easy to verify the correctness bound. Let B be an adversary against the $\mathsf{IND\text{-}CCA}$ security of $\mathsf{KEM}^\perp$, issuing at most q_D queries to $\textsc{Decaps}^\perp$ and at most q_H queries to H. Consider the games given in Figure 12.

Game G_0. Since game G_0 is the original $\mathsf{IND\text{-}CCA}$ game,

$$\left|\Pr[G_0^\mathsf{B} \Rightarrow 1] - \frac{1}{2}\right| = \mathrm{Adv}_{\mathsf{KEM}^\perp}^{\mathsf{IND\text{-}CCA}}(\mathsf{B}) \ .$$

Game G_1. In game G_1, the oracles H and $\textsc{Decaps}^\perp$ are changed such that they make no use of the secret key any longer except by testing if $\mathsf{Dec}'(sk', c) = m$ for given (m, c) in line 15 and if $\mathsf{Dec}'(sk, c) \in \mathcal{M}$ for given c in line 27. Game G_1 contains two sets: hash list $\mathfrak{L}_H$ that contains all entries (m, c, K) where H was queried on (m, c), and set $\mathfrak{L}_D$ that contains all entries (c, K) where either H was queried on (m', c), $m' := \mathsf{Dec}'(sk', c)$, or $\textsc{Decaps}^\perp$ was queried on c. In order to show that the view of B is identical in games G_0 and G_1, consider the following cases for a fixed ciphertext c and $m' := \mathsf{Dec}'(sk', c)$.

- Case 1: $m' \notin \mathcal{M}$. Since $\textsc{Val}(c) = 0$ is equivalent to $m' = \perp$, $\textsc{Decaps}(c)$ returns $\perp$ as in both games.
- Case 2: $m' \in \mathcal{M}$. We will now show that H in game G_1 is "patched", meaning that it is ensures $\textsc{Decaps}^\perp(c) = \mathsf{H}(m', c)$, where $m' := \mathsf{Dec}'(sk, c)$, for all ciphertexts c with $m' \in \mathcal{M}$. We distinguish two sub-cases: B might either first query H on (m', c), then $\textsc{Decaps}^\perp$ on c, or the other way round.
  - If H is queried on (m', c) first, it is recognized that $\mathsf{Dec}'(sk, c) = m$ in line 15. Since $\textsc{Decaps}$ was not yet queried on c, no entry of the form (c, K) can already exist in $\mathfrak{L}_D$. Therefore, besides adding $(m, c, K \xleftarrow{\$} \mathcal{K})$ to $\mathfrak{L}_H$, H also adds (c, K) to $\mathfrak{L}_D$ in line 22, thereby defining $\textsc{Decaps}^\perp(c) := K = \mathsf{H}(m', c)$.
  - If $\textsc{Decaps}^\perp$ is queried on c first, no entry of the form (c, K) exists in $\mathfrak{L}_D$ yet. Therefore, $\textsc{Decaps}^\perp$ adds $(c, K \xleftarrow{\$} \mathcal{K})$ to $\mathfrak{L}_D$, thereby defining $\textsc{Decaps}^\perp(c) := K$. When queried on (m', c) afterwards, H recognizes that $\mathsf{Dec}'(sk, c) = m'$ in line 15 and that an entry of the form (c, K) already exists in $\mathfrak{L}_D$ in line 19. By adding (m, c, K) to $\mathfrak{L}_H$ and returning K, H defines $\mathsf{H}(m', c) := K = \textsc{Decaps}^\perp(c)$.

```
$\mathsf{A}^{\textsc{Pco}}(pk, c^*)$
$K^* \xleftarrow{\$} \mathcal{K}$
$b' \leftarrow \mathsf{B}^{\textsc{Decaps}^{\perp},\mathsf{H}}(pk, c^*, K^*)$
if $\exists (m', c', K') \in \mathfrak{L}_H$
      s. th. $\textsc{Pco}(m', c^*) = 1$
    return $m'$
else
    abort
```

```
$\mathsf{H}(m, c)$
if $\exists K$ such that $(m, c, K) \in \mathfrak{L}_H$
    return $K$
$K \xleftarrow{\$} \mathcal{K}$
if $\textsc{Pco}(m, c) = 1$
    if $\exists K'$ such that $(c, K') \in \mathfrak{L}_D$
        $K := K'$
    else
        $\mathfrak{L}_D := \mathfrak{L}_D \cup \{(c, K)\}$
$\mathfrak{L}_H := \mathfrak{L}_H \cup \{(m, c, K)\}$
return $K$
```

Figure 13: Adversary A for the proof of Theorem 3.4, where $\textsc{Decaps}^{\perp}$ is defined as in Game G_2 of Figure 12.

We have shown that B's view is identical in both games and

$$\Pr[G_1^{\mathsf{B}} \Rightarrow 1] = \Pr[G_0^{\mathsf{B}} \Rightarrow 1]| \ .$$

Game G_2. In game G_2, we abort immediately on the event that B queries H on (m^*, c^*). Denote this event as CHAL. Due to the difference lemma,

$$|\Pr[G_2^{\mathsf{B}} \Rightarrow 1] - \Pr[G_1^{\mathsf{B}} \Rightarrow 1]| \leq \Pr[\mathrm{CHAL}] \ .$$

In game G_2, $\mathsf{H}(m^*, c^*)$ will not be given to B; neither through a hash nor a decryption query, meaning bit b is independent from B's view. Hence,

$$\Pr[G_2^{\mathsf{B}}] = \frac{1}{2} \ .$$

It remains to bound Pr[CHAL]. To this end, we construct an adversary A against the $\mathsf{OW\text{-}PVCA}$ security of PKE' simulating G_2 for B as in Figure 13. Note that the simulation is perfect. Since CHAL implies that B queried $\mathsf{H}(m^*, c^*)$ which implies $(m^*, c^*, K') \in \mathfrak{L}_H$ for some K', and A returns $m' = m^*$. Hence,

$$\Pr[\mathrm{CHAL}] = \mathrm{Adv}_{\mathsf{PKE}}^{\mathsf{OW\text{-}PVCA}}(\mathsf{A}) \ .$$

Collecting the probabilities yields the required bound. □

3.4 The resulting KEMs

For completeness, we combine transformations T_1, T_2, and $\mathsf{T}_2^{\perp}$ from the previous sections to obtain two variants of the FO transformation $\mathsf{FO} := \mathsf{T}_2 \circ \mathsf{T}_1$ and $\mathsf{FO}^{\perp} := \mathsf{T}_2^{\perp} \circ \mathsf{T}_1$. To a public-key encryption scheme $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ with message space $\mathcal{M}$ and randomness space $\mathcal{R}$, and hash functions $\mathsf{G} : \mathcal{M} \to \mathcal{R}$, $\mathsf{H} : \{0,1\}^* \to \{0,1\}^n$ we associate

$$\begin{aligned} \mathsf{KEM} &= \mathsf{FO}[\mathsf{PKE}, \mathsf{G}, \mathsf{H}] := \mathsf{T}_2[\mathsf{T}_1[\mathsf{PKE}, \mathsf{G}], \mathsf{H}] \\ \mathsf{KEM}^{\perp} &= \mathsf{FO}^{\perp}[\mathsf{PKE}, \mathsf{G}, \mathsf{H}] := \mathsf{T}_2^{\perp}[\mathsf{T}_1[\mathsf{PKE}, \mathsf{G}], \mathsf{H}] \ . \end{aligned}$$

For concreteness, the algorithms of $\mathsf{KEM} = (\mathsf{Gen}', \mathsf{Encaps}, \mathsf{Decaps})$ and $\mathsf{KEM}^{\perp} = (\mathsf{Gen}, \mathsf{Encaps}, \mathsf{Decaps}^{\perp})$ are given in Figure 14. The following two corollaries are directly obtained by combining Theorems 3.1–3.4.

Corollary 3.5 *If* PKE *is* δ*-correct, so is* $\mathsf{KEM} = \mathsf{FO}[\mathsf{PKE}, \mathsf{G}, \mathsf{H}]$*. Further, for any adversary* $\mathsf{IND\text{-}CCA}$ B *against* KEM*, issuing at most* q_D *queries to the decapsulation oracle* $\textsc{Dec}$*, at most* q_{G} *queries to the random oracle* G *and at most* q_{H} *queries to the random oracle* H*, there exist an* $\mathsf{OW\text{-}CPA}$ *adversary* A *and an* $\mathsf{IND\text{-}CPA}$ *adversary* A' *against* PKE *such that*

$$\mathrm{Adv}_{\mathsf{KEM}}^{\mathsf{IND\text{-}CCA}}(\mathsf{B}) \leq \begin{cases} q_{\mathsf{H}} \cdot \delta + \frac{q_{\mathsf{H}}}{|\mathcal{M}|} + (q_{\mathsf{G}} + 1) \cdot \mathrm{Adv}_{\mathsf{PKE}}^{\mathsf{OW\text{-}CPA}}(\mathsf{A}) \\ q_{\mathsf{H}} \cdot \delta + \frac{2q_{\mathsf{G}} + q_{\mathsf{H}} + 1}{|\mathcal{M}|} + 3 \cdot \mathrm{Adv}_{\mathsf{PKE}}^{\mathsf{IND\text{-}CPA}}(\mathsf{A}') \end{cases} ,$$

and the running times of A *and* A' *are about that of* B*.*

```
$\underline{\mathsf{Gen}'}$
$(pk, sk) \leftarrow \mathsf{Gen}$
$s \xleftarrow{\$} \mathcal{M}$
$sk' := (sk, s)$
$\mathbf{return}\ (pk, sk')$

$\underline{\mathsf{Decaps}(sk' = (sk, s), c)}$
$m' := \mathsf{Dec}(sk, c)$
$\mathbf{if}\ c \neq \mathsf{Enc}(pk, m'; G(m'))\ \mathbf{or}\ m' = \bot$
    $\mathbf{return}\ K := \mathsf{H}(s, c)$
$\mathbf{else\ return}\ K := \mathsf{H}(m', c)$

$\underline{\mathsf{Encaps}(pk)}$
$m \xleftarrow{\$} \mathcal{M}$
$c := \mathsf{Enc}(pk, m; G(m))$
$K := \mathsf{H}(m, c)$
$\mathbf{return}\ (K, c)$

$\underline{\mathsf{Decaps}^{\bot}(sk, c)}$
$m' := \mathsf{Dec}(sk, c)$
$\mathbf{if}\ c \neq \mathsf{Enc}(pk, m'; \mathsf{G}(m'))\ \mathbf{or}\ m' = \bot$
    $\mathbf{return}\ \bot$
$\mathbf{else\ return}\ K := \mathsf{H}(m', c)$
```

Figure 14: IND-CCA secure Key Encapsulation $\mathsf{FO}[\mathsf{PKE}, \mathsf{G}, \mathsf{H}] = (\mathsf{Gen}', \mathsf{Encaps}, \mathsf{Decaps})$ and $\mathsf{FO}^{\bot}[\mathsf{PKE}, \mathsf{G}, \mathsf{H}] = (\mathsf{Gen}, \mathsf{Encaps}, \mathsf{Decaps}^{\bot})$ obtained from PKE.

Corollary 3.6 *If* PKE *is δ-correct, so is* $\mathsf{KEM}^{\bot} = \mathsf{FO}^{\bot}[\mathsf{PKE}, \mathsf{G}, \mathsf{H}]$. *Assume* PKE *to be γ-spread. Then, for any* IND-CCA *adversary* B *against* $\mathsf{KEM}^{\bot}$*, issuing at most q_D queries to the decapsulation oracle* DEC*, at most q_{G} queries to the random oracle* G *and at most q_{H} queries to the random oracle* H*, there exist an* OW-CPA *adversary* A *and an* IND-CPA *adversary* A′ *against* PKE *such that*

$$\mathrm{Adv}^{\mathsf{IND\text{-}CCA}}_{\mathsf{KEM}^{\bot}}(\mathsf{B}) \leq \begin{cases} q_{\mathsf{H}} \cdot (\delta + 2^{-\gamma}) + (q_{\mathsf{G}} + 1) \cdot \mathrm{Adv}^{\mathsf{OW\text{-}CPA}}_{\mathsf{PKE}}(\mathsf{A}) \\ q_{\mathsf{H}} \cdot (\delta + 2^{-\gamma}) + \frac{2q_G+1}{|\mathcal{M}|} + 3 \cdot \mathrm{Adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{PKE}}(\mathsf{A}') \end{cases},$$

and the running time of both A *and* A′ *is about that of* B.

3.5 T_0^{ℓ}: from OW-CPA to IND-CPA Security, Tightly

T_0^{ℓ} transforms an OW-CPA secure public-key encryption scheme into an IND-CPA secure scheme. The security reduction has a parameter ℓ which allows for a tradeoff between the security loss of the reduction and the compactness of ciphertexts.

THE CONSTRUCTION. Fix an $\ell \in \mathbb{N}$. To a public-key encryption scheme $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ with message space $\mathcal{M} = \{0,1\}^n$ and a hash function $\mathsf{F} : \mathcal{M}^{\ell} \to \mathcal{R}$, we associate $\mathsf{PKE}_{\ell} = \mathsf{T}_0^{\ell}[\mathsf{PKE}, \mathsf{F}]$. The algorithms of PKE_{ℓ} are defined in Figure 15.

```
$\underline{\mathsf{Enc}_{\ell}(pk, m)}$
$\mathbf{x} := (x_1, \ldots, x_{\ell}) \xleftarrow{\$} (\{0,1\}^n)^{\ell}$
$c_0 := m \oplus \mathsf{F}(\mathbf{x})$
$\mathbf{for}\ i = 1\ \mathbf{to}\ \ell\ \mathbf{do}$
    $c_i := \mathsf{Enc}(pk, x_i)$
$\mathbf{return}\ c := (c_0, \ldots, c_{\ell})$

$\underline{\mathsf{Dec}_{\ell}(sk, c)}$
$\mathbf{parse}\ c = (c_0, \ldots, c_{\ell})$
$\mathbf{for}\ i = 1\ \mathbf{to}\ \ell\ \mathbf{do}$
    $x_i := \mathsf{Dec}(sk, c_i)$
$\mathbf{x} := (x_1, \ldots, x_{\ell})$
$\mathbf{return}\ c_0 \oplus \mathsf{F}(\mathbf{x})$
```

Figure 15: Tightly IND-CPA secure encryption PKE_{ℓ} obtained from PKE

SECURITY. The following theorem shows that PKE_{ℓ} is IND-CPA secure, provided that PKE is OW-CPA secure.

Theorem 3.7 (PKE OW-CPA ⇒ PKE_{ℓ} IND-CPA)**.** *If* PKE *is δ-correct, then* PKE_{ℓ} *is $\ell \cdot \delta$-correct. Moreover, for any* IND-CPA *adversary* B *that issues at most q_{F} queries to random oracle* F*, there exists an* OW-CPA *adversary* A *such that*

$$\mathrm{Adv}^{\mathsf{IND\text{-}CPA}}_{\mathsf{PKE}_{\ell}}(\mathsf{B}) \leq q_{\mathsf{F}}^{1/\ell} \cdot \mathrm{Adv}^{\mathsf{OW\text{-}CPA}}_{\mathsf{PKE}}(\mathsf{A})$$

and the running time of A *is about that of* B.

Proof. We first sketch correctness. Consider a public key pk and an encryption $c = (c_0, \dots, c_\ell)$ of generated by Enc_ℓ. Let x_i denote the respective value chosen by Enc_ℓ when generating c. Furthermore, let QUERY_i denote the event that, when decrypting c, the partial ciphertext c_i is decrypted to a value $x_i' \neq x_i$. If no QUERY_i occurs (for any i), then this implies that c is decrypted correctly. Hence, we have

$$\Pr[c \text{ decrypts incorrectly}] \quad \leq \quad \Pr[\bigvee_{i=1}^{\ell} \mathrm{QUERY}_i] \quad \leq \quad \sum_{i=1}^{\ell} \Pr[\mathrm{QUERY}_i] \quad \overset{(*)}{=} \quad \ell \cdot \delta,$$

where the probability is over the random coins of Gen_ℓ, Enc_ℓ, and Dec_ℓ, and $(*)$ follows from the δ-correctness of PKE.

As for security, let $\mathsf{B} = (\mathsf{B}_1, \mathsf{B}_2)$ be an adversary against the $\mathsf{IND\text{-}CPA}$ security of PKE_ℓ, issuing at most q_F queries to F. Consider the games given in Figure 16.

```
GAMES $G_0$-$G_1$
01 $(pk, sk) \leftarrow \mathsf{Gen}()$
02 $b \overset{\$}{\leftarrow} \{0,1\}$
03 $(m_0, m_1, st) \overset{\$}{\leftarrow} \mathsf{B}_1(pk)$
04 $\mathbf{x}^* := (x_1^*, \dots, x_\ell^*) \overset{\$}{\leftarrow} (\{0,1\}^n)^\ell$
05 $c_0^* := m_b \oplus \mathsf{F}(\mathbf{x}^*)$
06 for $i = 1$ to $\ell$ do
07     $c_i^* := \mathsf{Enc}(pk, x_i)$
08 $c^* := (c_0^*, \dots, c_\ell^*)$
09 $b' \overset{\$}{\leftarrow} \mathsf{B}_2(pk, c^*, st)$
10 return $b' =_? b$
```

```
$\mathsf{F}(\mathbf{x})$
11 if $\exists r$ s.t. $(\mathbf{x}, r) \in \mathfrak{L}_\mathsf{F}$
12     return $r$
13 if $\mathbf{x} = \mathbf{x}^*$                // $G_1$
14     QUERY := true                  // $G_1$
15     abort                          // $G_1$
16 $r \overset{\$}{\leftarrow} \mathcal{R}$
17 $\mathfrak{L}_\mathsf{F} := \mathfrak{L}_\mathsf{F} \cup \{(\mathbf{x}, r)\}$
18 return $r$
```

Figure 16: Games G_0 - G_1 for the proof of Theorem 3.7

GAME G_0. Since game G_0 is the original $\mathsf{IND\text{-}CPA}$ game,

$$\left|\Pr[G_0^\mathsf{B} \Rightarrow 1] - 1/2\right| = \mathrm{Adv}_{\mathsf{PKE}_\ell}^{\mathsf{IND\text{-}CPA}}(\mathsf{B}) \ . \tag{3}$$

GAME G_1. In Game G_1, we add lines 13-15, and in particular a flag QUERY in line 14, and abort (such that the game outputs an independently random bit) when QUERY is raised. QUERY is raised whenever random oracle F is queried with the vector $\mathbf{x}^*$ that was chosen during the generation of the challenge ciphertext c^*. Games G_0 and G_1 proceed identically until QUERY occurs. Hence, we have

$$\left|\Pr[G_0^\mathsf{B} \Rightarrow 1] - \Pr[G_1^\mathsf{B} \Rightarrow 1]\right| \ \leq \ \Pr[\mathrm{QUERY}] \ . \tag{4}$$

Moreover, observe that in Game G_1, B's view is independent of the bit b chosen by the game: b is only used in the computation of c_0^*, which in turn is blinded by $\mathsf{F}(\mathbf{x}^*)$. But since the game aborts (with a random output) as soon as B queries $\mathsf{F}(\mathbf{x}^*)$, this means that c_0^* is independently random in B's view. This means that also B's output b' and b are independent, which implies that the game's output $b' =_? b$ is a uniformly random bit in case no abort occurs. But since the game also outputs a random bit upon an abort, we get that

$$\Pr[G_1^\mathsf{B} \Rightarrow 1] \ = \ 1/2. \tag{5}$$

Taking (3-5) together, we thus get

$$\mathrm{Adv}_{\mathsf{PKE}_\ell}^{\mathsf{IND\text{-}CPA}}(\mathsf{B}) \ \leq \ \Pr[\mathrm{QUERY}] \ ,$$

and the theorem follows from the next lemma. □

Lemma 3.8 *In the situation of Game G_1, we have*

$$\Pr[\mathrm{QUERY}] \ \leq \ q_\mathsf{F}^{1/\ell} \cdot \mathrm{Adv}_{\mathsf{PKE}}^{\mathsf{OW\text{-}CPA}}(\mathsf{A})$$

for an adversary A (of roughly the same complexity as Game G_1).

```
$\mathsf{A}(pk, \widehat{c})$:
01 $(m_0, m_1, st) \overset{\$}{\leftarrow} \mathsf{B}_1^{\mathsf{F}}(pk)$
02 $c_0^* \overset{\$}{\leftarrow} \{0,1\}^n$
03 for $i = 1$ to $\ell$ with $i \neq i^*$
04     $x_i^* \overset{\$}{\leftarrow} \{0,1\}^n$
05     $c_i^* \overset{\$}{\leftarrow} \mathsf{Enc}(pk, x_i^*)$
06 $i^* \overset{\$}{\leftarrow} [\ell]$
07 $c_{i^*}^* := \widehat{c}$
08 $c^* := (c_0^*, \ldots, c_\ell^*)$
09 $b' \overset{\$}{\leftarrow} \mathsf{B}_2^{\mathsf{F}}(pk, c^*, st)$
10 if $\mathfrak{L}_{i^*}$ empty
11     $x = \perp$
12 else
13     $x \overset{\$}{\leftarrow} \mathfrak{L}_{i^*}$
14 return $x$

$\mathsf{F}(\mathbf{x})$
15 if $\exists r$ s. th.$(\mathbf{x}, r) \in \mathfrak{L}_{\mathsf{F}}$
16     return $r$
17 $r \overset{\$}{\leftarrow} \mathcal{R}$
18 $\mathfrak{L}_{\mathsf{F}} := \mathfrak{L}_{\mathsf{F}} \cup \{(\mathbf{x}, r)\}$
19 parse $\mathbf{x} = (x_1, \ldots, x_\ell)$
20 if $\forall i < i^* : x_i = x_i^*$
21     $\mathfrak{L}_{i^*} := \mathfrak{L}_{i^*} \cup \{x_{i^*}\}$
22 return $r$
```

Figure 17: Adversary A against $\mathsf{IND\text{-}CPA}$ from B against $\mathsf{OW\text{-}PCA}$ for Lemma 3.8. Note that the sampling operation in line 13 refers to the *list* (not the *set*) $\mathfrak{L}_{i^*}$ (such that multiple F queries with the same x_{i^*} may raise the probability that that x_{i^*} is sampled).

Proof. We may assume that Pr[QUERY] > 0 (so that it is possible to condition on QUERY). We describe adversary A in Figure 17.

To analyze B, let $\mathbf{x}^* := (x_1^*, \ldots, x_\ell^*)$, where $x_{i^*}^*$ is the value encrypted in A's own challenge $\widehat{c}$, and, for $i \neq i^*$, the x_i^* are defined in line 4 in Figure 17. (That is, up to decryption errors, $x_i^* = \mathsf{Dec}(sk, c_i^*)$ for all i.) Now observe that B's views in Game G_1 and in the simulation inside A are identical *until* B queries $\mathsf{F}(\mathbf{x}^*)$. In this latter case, Game G_1 would abort, while A would simply continue the simulation. In particular, if we let QUERY denote the event that B queries $\mathsf{F}(\mathbf{x}^*)$, then the probability of QUERY is the same in Game G_1 and in A's simulation. We can thus show the lemma by bounding the probability for QUERY in A's simulation.

To this end, for each $i \in [\ell]$, consider the probability

$$p_i \;:=\; \Pr[\, x_i = x_i^* \;\mid\; (x_1, \ldots, x_{i-1}) = (x_1^*, \ldots, x_{i-1}^*) \;\wedge\; \text{QUERY}\,]$$

in an execution with A, where the probability is over a uniform choice of $\mathbf{x} = (x_1, \ldots, x_\ell)$ among the set of all of F-queries from B. (Note that the condition QUERY guarantees that at least one such $\mathbf{x}$ exists.) Intuitively, p_i denotes the probability that a F-query matches the challenge message in the i-th component when they already match in the first $i-1$ components (assuming that QUERY occurs).

It will be helpful to first note a useful property of the p_i: namely, we have

$$\prod_{i=1}^{\ell} p_i \;\overset{(i)}{=}\; \Pr[\,\mathbf{x} = \mathbf{x}^* \;\mid\; \text{QUERY}\,] \;\overset{(ii)}{=}\; 1/q_{\mathsf{F}} \;, \tag{6}$$

where (i) follows by using $\Pr[A \mid B] \cdot \Pr[B] = \Pr[A \wedge B]$ for arbitrary events A, B (such that B is possible), and (ii) follows by definition of QUERY.

Furthermore, we can connect the p_i to A's output as follows. Observe that B's view in A's simulation does not depend on i^*, and thus, that the p_i do not change when conditioning on a specific choice of i^*. Now by construction of A and the list $\mathfrak{L}_{i^*}$, for each fixed choice of i^*, and assuming that QUERY occurs, we have that $x = x_{i^*}^*$ is sampled in line 13 with probability p_{i^*}. Note that in this case, A wins its own

$\mathsf{OW\text{-}CPA}$ game. Formally:

$$\begin{aligned}
\mathrm{Adv}^{\mathsf{OW\text{-}CPA}}_{\mathsf{PKE}}(\mathsf{A}) &= \Pr[\,\mathsf{A} \Rightarrow x^*_{i^*}\,] = \frac{1}{\ell}\sum_{i=1}^{\ell} \Pr[\,\mathsf{A} \Rightarrow x^*_i \mid i^* = i\,] \\
&= \frac{\Pr[\mathrm{QUERY}]}{\ell}\sum_{i=1}^{\ell} \Pr[\,\mathsf{A} \Rightarrow x^*_{i^*} \mid i^* = i \wedge \mathrm{QUERY}\,] = \frac{\Pr[\mathrm{QUERY}]}{\ell}\sum_{i=1}^{\ell} p_i \\
&\overset{(*)}{\geq} \Pr[\mathrm{QUERY}] \cdot \left(\prod_{i=1}^{\ell} p_i\right)^{1/\ell} \overset{(6)}{=} \Pr[\mathrm{QUERY}] \cdot \frac{1}{q_{\mathsf{F}}^{1/\ell}} \ ,
\end{aligned}$$

where $(*)$ follows by the inequality between the arithmetic and geometric means. Rearranging terms yields the lemma. □

4 Modular FO Transformation in the QROM

In this section, we will revisit our transformations in the quantum random oracle model. In Section 4.1, we give a short primer on quantum computation and define the quantum random oracle model (QROM). In Section 4.2, we will prove that transformation T_1 from Figure 4 (Section 3.1) is also secure in the quantum random oracle model. Next, in Section 4.3 we will introduce QT_2, a variant of T_2, which has provable security in the quantum random oracle model. Combining the two above transformations, in Section 4.4 we provide concrete bounds for the $\mathsf{IND\text{-}CCA}$ security of $\mathsf{QKEM} = \mathsf{QFO}[\mathsf{PKE}, \mathsf{G}, \mathsf{H}, \mathsf{H}']$ in the QROM.

4.1 Quantum Computation

Qbits. For simplicity, we will treat a *qbit* as a vector $|b\rangle \in \mathbb{C}^2$, i.e., a linear combination $|b\rangle = \alpha \cdot |0\rangle + \beta \cdot |1\rangle$ of the two *basis states* (vectors) $|0\rangle$ and $|1\rangle$ with the additional requirement to the probability amplitudes $\alpha, \beta \in \mathbb{C}$ that $|\alpha|^2 + |\beta|^2 = 1$. The basis $\{|0\rangle, |1\rangle\}$ is called *standard orthonormal computational basis.* The qbit $|b\rangle$ is said to be *in superposition.* Classical bits can be interpreted as quantum bits by considering $(b \mapsto 1 \cdot |b\rangle + 0 \cdot |1 - b\rangle)$.

Quantum Registers. We will treat a quantum register as a collection of multiple qbits, i.e. a linear combination $\sum_{b_1,\cdots,b_n \in \{0,1\}} \alpha_{b_1\cdots b_n} \cdot |b_1 \cdots b_n\rangle$, where $\alpha_{b_1,\cdots,b_N} \in \mathbb{C}^n$, with the additional restriction that $\sum_{b_1,\cdots,b_n \in \{0,1\}} |\alpha_{b_1\cdots b_n}|^2 = 1$. As in the one-dimensional case, we call the basis $\{|b_1 \cdots b_n\rangle\}_{b_1,\cdots,b_n \in \{0,1\}}$ the *standard orthonormal computational basis.*

Measurements. Qbits can be measured with respect to a basis. In this paper, we will only consider measurements in the standard orthonormal computational basis, and denote this measurement by Measure$(\cdot)$, where the outcome of measuring a single qbit $|b\rangle = \alpha \cdot |0\rangle + \beta \cdot |1\rangle$ will be $|0\rangle$ with probability $|\alpha|^2$ and $|1\rangle$ with probability $|\beta|^2$, and the outcome of measuring a qbit register $\sum\limits_{b_1,\cdots,b_n \in \{0,1\}} \alpha_{b_1\cdots b_n} \cdot |b_1 \cdots b_n\rangle$ will be $|b_1 \cdots b_n\rangle$ with probability $|\alpha_{b_1\cdots b_n}|^2$. Note that the amplitudes *collapse* during a measurement, this means that by measuring $\alpha \cdot |0\rangle + \beta \cdot |1\rangle$, α and β are switched to one of the combinations in $\{\pm(1,0),\ \pm(0,1)\}$. Likewise, in the n-dimensional case, all amplitudes are switched to 0 except for the one that belongs to the measurement outcome and which will be switched to 1.

Quantum oracles and quantum Adversaries. Following [BDF+11, BBC+98], we view a quantum oracle as a mapping

$$|x\rangle|y\rangle \mapsto |x\rangle|y \oplus \mathrm{O}(x)\rangle \ ,$$

where $\mathrm{O} : \{0,1\}^n \to \{0,1\}^m$, $x \in \{0,1\}^n$ and $y \in \{0,1\}^m$, and model quantum adversaries A with access to O by the sequence $U \circ \mathrm{O}$, where U is a unitary operation. We write $\mathsf{A}^{|\mathrm{O}\rangle}$ to indicate that the oracles are quantum-accessible (contrary to oracles which can only process classical bits).

Quantum random oracle model. We consider security games in the quantum random oracle model (QROM) as their counterparts in the classical random oracle model, with the difference that we consider quantum adversaries that are given **quantum** access to the random oracles involved, and **classical** access to all other oracles (e.g., plaintext checking or decapsulation oracles). Zhandry [Zha12] proved that for any quantum algorithm $\mathsf{A}^{|f\rangle}$, issuing at most q quantum queries to f, cannot distinguish between a random function $f : \{0,1\}^m \to \{0,1\}^n$ or a random $2q$-wise independent function. It allows us to view quantum random oracles as polynomials of sufficient large degree. That is, we define a quantum random oracle $|\mathsf{H}\rangle$ as an oracle evaluating a random polynomial of degree $2q$ over the finite field $\mathbb{F}_{2^n}$.

Oneway to Hiding. To a quantum oracle $|\mathsf{H}\rangle$ and an algorithm A (possibly with access to other oracles) we associate the following extractor algorithm $\mathsf{EXT}[\mathsf{A}, |\mathsf{H}\rangle]$ that returns a measurement x' of a randomly chosen query to $|\mathsf{H}\rangle$.

```
$\mathsf{EXT}[\mathsf{A}, |\mathsf{H}\rangle](inp)$
$i \xleftarrow{\$} [q_\mathsf{H}]$
Run $\mathsf{A}^{|\mathsf{H}\rangle}(inp)$ until the $i$th query $|\hat{x}\rangle$ to $|\mathsf{H}\rangle$
if $i >$ number of queries to $|\mathsf{H}\rangle$
    return $\perp$
else
    return $x' := \textsc{Measure}(|\hat{x}\rangle)$
```

Figure 18: Extractor algorithm $\mathsf{EXT}[\mathsf{A}, |\mathsf{H}\rangle](inp)$ for OW2H.

The following statement is an adaption of OW2H from [Unr14] and will be used heavily during our security proofs.

Lemma 4.1 *(Algorithmic Oneway to hiding (AOW2H)) Let $|\mathsf{H}\rangle : \{0,1\}^n \to \{0,1\}^m$ be a quantum random oracle, and let A be a quantum algorithm issuing at most q_H queries to $|\mathsf{H}\rangle$ that, on input $x \in \{0,1\}^n, y \in \{0,1\}^m$ outputs either 0 or 1. Then, for all (probabilistic) algorithms F that input bit-stings in $\{0,1\}^{n+m}$ (and do not make any hash queries to $|\mathsf{H}\rangle$),*

$$\left|\Pr\left[1 \leftarrow \mathsf{A}^{|\mathsf{H}\rangle}(inp) \mid x \xleftarrow{\$} \{0,1\}^n; inp \leftarrow \mathsf{F}(x, \mathsf{H}(x))\right] - \Pr\left[1 \leftarrow \mathsf{A}^{|\mathsf{H}\rangle}(inp) \mid (x,y) \xleftarrow{\$} \{0,1\}^{n+m}; inp \leftarrow \mathsf{F}(x,y)\right]\right|$$
$$\leq 2q_\mathsf{H} \cdot \sqrt{\Pr[x \leftarrow \mathsf{EXT}[\mathsf{A}, |\mathsf{H}\rangle](inp) \mid (x,y) \xleftarrow{\$} \{0,1\}^{n+m}; inp \leftarrow \mathsf{F}(x,y)]} \ .$$

4.2 T_1: from OW-CPA to OW-PCA Security in the QROM

Recall transformation T_1 from Figure 4 of Section 3.1. The following theorem (whose proof is loosely based on [TU16]) establishes that IND-PCA security of PKE′ reduces to the OW-CPA security of PKE, in the quantum random oracle model.

Theorem 4.2 **(PKE OW-CPA $\overset{\text{QROM}}{\Rightarrow}$ PKE′ OW-PCA).** *Assume PKE to be δ-correct. For any OW-PCA quantum adversary B that issues at most q_G queries to the quantum random oracle $|\mathsf{G}\rangle$ and q_P (classical) queries to the plaintext checking oracle* Pco, *there exists an OW-CPA quantum adversary A such that*

$$\mathrm{Adv}^{\mathsf{OW\text{-}PCA}}_{\mathsf{PKE}'}(\mathsf{B}) \leq q_P \cdot \delta + (1 + 2q_\mathsf{G}) \cdot \sqrt{\mathrm{Adv}^{\mathsf{OW\text{-}CPA}}_{\mathsf{PKE}}(\mathsf{A})} \ ,$$

and the running time of A is about that of B.

Proof. Let B be an adversary against the OW-PCA security of PKE′, issuing at most q_G queries to $|\mathsf{G}\rangle$ and at most q_P queries to Pco. Consider the games given in Figure 19, where G is modeled as a random $2q_\mathsf{G}$-wise independent hash function. Except for G, games G_0 and G_1 are the same as in the proof of the classical random oracle (Theorem 3.1), games G_2 and H are different.

Game G_0. Since game G_0 is the original OW-PCA game,

$$\Pr[G_0^\mathsf{B} \Rightarrow 1] = \mathrm{Adv}^{\mathsf{OW\text{-}PCA}}_{\mathsf{PKE}'}(\mathsf{B}) \ .$$

Game G_1. In game G_1 the plaintext checking oracle is replaced with a simulation that doesn't make use of the secret key anymore. With the same argument we used to show Equation (1) in the proof of Theorem 3.1,

$$|\Pr[G_1^\mathsf{B} \Rightarrow 1] - \Pr[G_0^\mathsf{B} \Rightarrow 1]| \leq q_P \cdot \delta \ .$$

```
GAME $G_0$-$G_2$, $H$
01 $(pk, sk) \leftarrow \mathsf{Gen}$
02 $m^* \xleftarrow{\$} \mathcal{M}$
03 $r^* := \mathsf{G}(m^*)$                                        // $G_0$-$G_1$
04 $r^* \xleftarrow{\$} \mathcal{R}$                                       // $G_2$, $H$
05 $c^* := \mathsf{Enc}(pk, m^*; r^*)$
06 $m' \leftarrow \mathsf{B}^{|\mathsf{G}\rangle, \textsc{Pco}}(pk, c^*)$                     // $G_1$-$G_2$
07 $m' \leftarrow \mathsf{EXT}[\mathsf{B}^{\textsc{Pco}}, |\mathsf{G}\rangle](pk, c^*)$              // $H$
08 return $m' =_? m^*$
```

```
$\textsc{Pco}(m \in \mathcal{M}, c)$
09 $m' := \mathsf{Dec}(sk, c)$                                      // $G_0$
10 if $m' = m$ and $\mathsf{Enc}(pk, m'; \mathsf{G}(m')) = c$               // $G_0$
11     return 1                                               // $G_0$
12 else return 0                                              // $G_0$
13 return $\mathsf{Enc}(pk, m; \mathsf{G}(m)) =_? c$                 // $G_1$, $G_2$, $H$
```

Figure 19: Games G_0, G_1, G_2, H for the proof of Theorem 4.2

```
$\mathsf{C}(pk, c^*)$
01 $m' \leftarrow \mathsf{B}^{|\mathsf{G}\rangle, \textsc{Pco}}(pk, c^*)$
02 return $m'$
```

```
$\mathsf{D}(pk, c^*)$
03 $m' \leftarrow \mathsf{EXT}[\mathsf{B}^{\textsc{Pco}}, |\mathsf{G}\rangle](pk, c^*)$
04 return $m'$
```

Figure 20: Adversaries C (left) and D (right) for the proof of Theorem 4.2. Oracle Pco is defined as in game G_2 of Figure 19.

Game G_2. In game G_2, we replace $r^* := \mathsf{G}(m^*)$ with uniform randomness r^* in line 03. We apply Lemma 4.1 (AOW2H) to $x := m^*$, $y := r^*$, and algorithm F, where $\mathsf{F}(m^*, r^*)$ first computes $(pk, sk) \leftarrow \mathsf{Gen}$, then $c^* := \mathsf{Enc}(pk, m^*; r^*)$, and outputs $inp := (pk, c^*)$. We obtain

$$|\Pr[G_2^{\mathsf{B}} \Rightarrow 1] - \Pr[G_1^{\mathsf{B}} \Rightarrow 1]| \leq 2 \cdot q_{\mathsf{G}} \cdot \sqrt{\Pr[H^{\mathsf{B}} \Rightarrow 1]} \ ,$$

where the extractor algorithm EXT of game H is defined in Figure 18.

Now that r^* is uniformly random we trivially construct an adversary C in Figure 20 against the $\mathsf{OW\text{-}CPA}$ security of the original encryption scheme PKE simulating game G_2 for B that outputs $m' = m^*$ if B wins in game G_2.

$$\Pr[G_2^{\mathsf{B}} \Rightarrow 1] = \mathrm{Adv}_{\mathsf{PKE}}^{\mathsf{OW\text{-}CPA}}(\mathsf{C}) \leq \sqrt{\mathrm{Adv}_{\mathsf{PKE}}^{\mathsf{OW\text{-}CPA}}(\mathsf{C})} \ .$$

Finally, we construct another trivial adversary D in Figure 20 against the $\mathsf{OW\text{-}CPA}$ security of the original encryption scheme PKE simulating game H for B with Advantage

$$\Pr[G_3^{\mathsf{B}} \Rightarrow 1] = \mathrm{Adv}_{\mathsf{PKE}}^{\mathsf{OW\text{-}CPA}}(\mathsf{D}) \ .$$

Collecting the probabilities and combining adversaries C and D into one single adversary A proves the theorem. □

4.3 QT_2: from OW-PCA to IND-CCA Security in the QROM

QT_2 transforms an $\mathsf{OW\text{-}PCA}$ secure public-key encryption scheme into an $\mathsf{IND\text{-}CCA}$ secure key encapsulation mechanism.

The Construction. To a public-key encryption scheme $\mathsf{PKE}' = (\mathsf{Gen}', \mathsf{Enc}', \mathsf{Dec}')$ with message space $\mathcal{M} = \{0,1\}^n$, and hash functions $\mathsf{H} : \{0,1\}^* \to \{0,1\}^n$ and $\mathsf{H}' : \{0,1\}^n \to \{0,1\}^n$, we associate $\mathsf{QKEM} = \mathsf{QT}_2[\mathsf{PKE}', \mathsf{H}, \mathsf{H}']$. The algorithms of $\mathsf{QKEM} = (\mathsf{QGen} := \mathsf{Gen}', \mathsf{QEncaps}, \mathsf{QDecaps})$ are defined in Figure 21.

Security. The following theorem (whose proof is again loosely based on [TU16]) establishes that $\mathsf{IND\text{-}CCA}$ security of KEM reduces to the $\mathsf{OW\text{-}PCA}$ security of PKE', in the quantum random oracle model.

Theorem 4.3 (PKE' $\mathsf{OW\text{-}PCA}$ $\overset{\text{QROM}}{\Rightarrow}$ QKEM $\mathsf{IND\text{-}CCA}$)**.** *If* PKE' *is δ-correct, so is* QKEM*. For any* $\mathsf{IND\text{-}CCA}$ *quantum adversary* B *issuing at most q_D (classical) queries to the decapsulation oracle* QDecaps*,*

```
$\underline{\mathsf{QEncaps}(pk)}$
$m \xleftarrow{\$} \mathcal{M}$
$c \leftarrow \mathsf{Enc}'(pk, m)$
$d := \mathsf{H}'(m)$
$K := \mathsf{H}(m)$
**return** $(K, c, d)$

$\underline{\mathsf{QDecaps}(sk, c, d)}$
$m' := \mathsf{Dec}'(sk, c)$
**if** $m' = \bot$ **or** $\mathsf{H}'(m') \neq d$
    **return** $\bot$
**else return** $K := \mathsf{H}(m')$
```

Figure 21: $\mathsf{IND\text{-}CCA}$-secure key encapsulation mechanism $\mathsf{QKEM} = \mathsf{QT}_2[\mathsf{PKE}', \mathsf{H}, \mathsf{H}']$.

at most q_{H} queries to the quantum random oracle $|\mathsf{H}\rangle$ and at most $q_{\mathsf{H}'}$ queries to the quantum random oracle $|\mathsf{H}'\rangle$, there exists an $\mathsf{OW\text{-}PCA}$ *quantum adversary* A *issuing $2q_D q_{\mathsf{H}'}$ queries to oracle* $\textsc{Pco}$ *such that*

$$\mathrm{Adv}^{\mathsf{IND\text{-}CCA}}_{\mathsf{QKEM}}(\mathsf{B}) \leq (2q_{\mathsf{H}'} + q_{\mathsf{H}}) \cdot \sqrt{\mathrm{Adv}^{\mathsf{OW\text{-}PCA}}_{\mathsf{PKE}'}(\mathsf{A})} \ ,$$

and the running time of A *is about that of* B.

By convention, the number of random oracle queries of B includes the ones B makes explicitly to the random oracle, and the ones B makes implicitly via the decryption oracles.

Proof. Let B be an adversary against the $\mathsf{IND\text{-}CCA}$ security of QKEM, issuing at most q_D queries to $\textsc{QDecaps}$, at most q_{H} queries to $|\mathsf{H}\rangle$ and at most $q_{\mathsf{H}'}$ queries to $|\mathsf{H}'\rangle$. Consider the games $G_{0,b}, G_{1,b}, H_{0,b}, H_{1,b}$ $(b \in \{0,1\})$ given in Figure 22.

```
**GAMES** $G_{0,b},\ G_{1,b},\ H_{0,b},\ H_{1,b}$
01 $(pk, sk) \leftarrow \mathsf{Gen}'$
02 $m^* \xleftarrow{\$} \{0,1\}^n;\ c^* \leftarrow \mathsf{Enc}'(pk, m^*)$
03 $K_0^* := \mathsf{H}(m^*); K_1^* \xleftarrow{\$} \{0,1\}^n$
04 $d^* := \mathsf{H}'(m^*); K^* := K_b^*$                                   $/\!\!/ G_{0,b}$
05 $d^* \xleftarrow{\$} \{0,1\}^n; K^* \xleftarrow{\$} \{0,1\}^n$                         $/\!\!/ G_{1,b},\ H_{0,b},\ H_{1,b}$
06 **return** $b' \leftarrow \mathsf{B}^{\textsc{QDecaps},|\mathsf{H}\rangle,|\mathsf{H}'\rangle}(pk, (c^*, d^*), K^*)$     $/\!\!/ G_{0,b}\text{-}G_{1,b}$
07 $m' \xleftarrow{\$} \mathsf{EXT}[\mathsf{B}^{\textsc{QDecaps}}, |\mathsf{H} \times \mathsf{H}'\rangle](pk, (c^*, d^*), K^*)$   $/\!\!/ H_{0,0}, H_{1,0}$
08 $m' \xleftarrow{\$} \mathsf{EXT}[\mathsf{B}^{\textsc{QDecaps},|\mathsf{H}\rangle}, |\mathsf{H}'\rangle](pk, (c^*, d^*), K^*)$   $/\!\!/ H_{0,1}, H_{1,1}$
09 **return** $m' =_? m^*$                                              $/\!\!/ H_{0,b}, H_{1,b}$

$\underline{\textsc{QDecaps}((c, d) \neq (c^*, d^*))}$   $/\!\!/ G_{0,b},\ G_{1,b},\ H_{0,b}$
10 $m' := \mathsf{Dec}'(sk, c)$
11 **if** $m' \neq \bot$ **and** $\mathsf{H}'(m') = d$
12     **return** $K := \mathsf{H}(m')$
13 **else return** $\bot$

$\underline{\textsc{QDecaps}((c, d) \neq (c^*, d^*))}$                    $/\!\!/ H_{1,b}$
14 $R := \mathsf{Roots}(\mathsf{H}'(X) - d)$
15 **if** $\exists\ m \in R$ s.t. $\mathsf{Dec}'(sk, c) = m$
16     **return** $K := \mathsf{H}(m)$.
17 **else return** $\bot$
```

Figure 22: Games $G_{0,b},\ G_{1,b},\ H_{0,b},\ H_{1,b}$ $(b \in \{0,1\})$ for the proof of Theorem 4.3.

$\textsc{Games}$ $G_{0,b}$. We use the $\mathsf{IND\text{-}CPA}$ game in its equivalent left-or-right form:

$$\begin{aligned} \mathrm{Adv}^{\mathsf{IND\text{-}CCA}}_{\mathsf{QKEM}}(\mathsf{B}) &= \frac{1}{2} \cdot \left| \Pr\left[\mathsf{IND\text{-}CCA}^{\mathsf{A}} \Rightarrow 0 \mid b = 0\right] - \Pr\left[\mathsf{IND\text{-}CCA}^{\mathsf{A}} \Rightarrow 1 \mid b = 1\right] \right| \\ &= \frac{1}{2} \left| \Pr[G_{0,0}^{\mathsf{B}} \Rightarrow 1] - \Pr[G_{0,1}^{\mathsf{B}} \Rightarrow 1] \right| \ . \end{aligned}$$

$\textsc{Games}$ $G_{1,b}$. In games $G_{1,b}$, we replace $(d^* := \mathsf{H}'(m^*), K^* := K_b^*)$ with uniform randomness (d^*, K^*) in line 05. Since $G_{1,0} = G_{1,1}$, we obtain

$$|\Pr[G_{0,0}^{\mathsf{B}} \Rightarrow 1] - \Pr[G_{0,1}^{\mathsf{B}} \Rightarrow 1]| \quad \leq \quad |\Pr[G_{0,0}^{\mathsf{B}} \Rightarrow 1] - \Pr[G_{1,0}^{\mathsf{B}} \Rightarrow 1]| + |\Pr[G_{0,1}^{\mathsf{B}} \Rightarrow 1] - \Pr[G_{1,1}^{\mathsf{B}} \Rightarrow 1]|$$

We apply Lemma 4.1 (AOW2H) to $x := m^*$, and $y = (K^*, d^*)$ for $b = 0$ and $y = d^*$ for $b = 1$, and algorithm F, where $\mathsf{F}(m^*, r^*)$ first computes $(pk, sk) \leftarrow \mathsf{Gen}$, then $c^* := \mathsf{Enc}(pk, m^*; r^*)$, and additionally $K^* \xleftarrow{\$} \{0,1\}^n$ for $b = 1$, and outputs $inp = (pk, c^*, d^*, K^*)$. We obtain

$$\begin{aligned} |\Pr\left[G_{0,0}^{\mathsf{B}} \Rightarrow 1\right] - \Pr[G_{1,0}^{\mathsf{B}} \Rightarrow 1]| &\leq 2(q_{\mathsf{H}'} + q_{\mathsf{H}}) \cdot \sqrt{\Pr[H_{0,0}^{\mathsf{B}} \Rightarrow 1]} \\ |\Pr\left[G_{0,1}^{\mathsf{B}} \Rightarrow 1\right] - \Pr[G_{1,1}^{\mathsf{B}} \Rightarrow 1]| &\leq 2q_{\mathsf{H}'} \cdot \sqrt{\Pr[H_{0,1}^{\mathsf{B}} \Rightarrow 1]} \ . \end{aligned}$$

```
$\underline{\mathsf{A}_b^{\textsc{Pco}}(pk, c^*)}$
$d^* \xleftarrow{\$} \{0,1\}^n$; $K^* \xleftarrow{\$} \{0,1\}^n$
$m' \xleftarrow{\$} \mathsf{EXT}[\mathsf{B}^{\textsc{QDecaps}}|\mathsf{H} \times \mathsf{H}'\rangle](pk, c^*, d^*, K^*)$      $/\!\!/ b = 0$
$m' \xleftarrow{\$} \mathsf{EXT}[\mathsf{B}^{\textsc{QDecaps},|\mathsf{H}\rangle}, |\mathsf{H}'\rangle](pk, c^*, d^*, K^*)$      $/\!\!/ b = 1$
return $m'$
```

Figure 23: Adversaries A_b ($b \in \{0,1\}$) for the proof of Theorem 4.3. Oracle QDecaps(c, d) is defined as in game $H_{1,b}$ of Figure 22.

Game $H_{1,b}$. In games $H_{1,b}$, the oracle QDecaps is changed such that it does not make use of the secret key any longer (except for line 15 by testing if $\mathsf{Dec}'(sk, c) = m$ for given c and messages m). Recall that H' is a random polynomial of degree $2q_{\mathsf{H}'}$ over $\mathbb{F}_{2^n}$. Therefore, given that (c, d) is a valid encapsulation (i.e., $m' \in \mathcal{M}$ and $d = \mathsf{H}'(m')$, where $m' := \mathsf{Dec}'(sk, c)$), m' lies within the roots of $\mathsf{H}'(X) - d$. In order to show that QDecaps returns the same output in games $H_{1,b}$ and $H_{0,b}$ for every query $(c, d) \neq (c^*, d^*)$, consider the following cases, where we define $m' := \mathsf{Dec}'(sk, c)$.

- Case 1: QDecaps(c, d) returns $\perp$ in Game $H_{1,b}$, meaning that $m' \notin \mathsf{Roots}(\mathsf{H}'(X) - d)$. That latter can only happen if $\mathsf{H}'(m') \neq d$ or $m' = \perp$, which is exactly the condition that QDecaps(c, d) returns $\perp$ in Game $H_{0,b}$.
- Case 2: QDecaps(c, d) does not return $\perp$ in Game $H_{1,b}$, meaning that $m' \in \mathsf{Roots}(\mathsf{H}'(X) - d)$. Consequently, $\mathsf{H}'(m') = d$ and QDecaps(c, d) returns $K = \mathsf{H}(m')$ in Games $H_{1,b}$. The latter is again exactly the condition that QDecaps(c, d) returns $K = \mathsf{H}(m')$ in Game $H_{0,b}$.

It is easy to verify that the equivalence of QDecaps in the two games follows by negation and combining both cases. We have just shown

$$\Pr[H_{1,b}^{\mathsf{B}} \Rightarrow 1] = \Pr[H_{0,b}^{\mathsf{B}} \Rightarrow 1] \ .$$

For $b \in \{0,1\}$, we trivial construct adversaries A_b against the $\mathsf{OW\text{-}PCA}$ security of PKE' simulating games $H_{1,b}$ for B as in Figure 23.

Hence,

$$\Pr[H_{1,b}^{\mathsf{B}} \Rightarrow 1] = \mathrm{Adv}_{\mathsf{PKE}'}^{\mathsf{OW\text{-}PCA}}(\mathsf{A}_b) \ .$$

Note that both adversaries issue at most $2q_D q_{\mathsf{H}'}$ Pco-queries: For each query of B to QDecaps on (c, d), both A_0 and A_1 compute the set $\mathsf{Roots}(\mathsf{H}'(X) - d)$ of complex roots, which has $2q_{\mathsf{H}'} - 1$ elements since $\mathsf{H}'(X) - d$ is a polynomial of degree $2q_{\mathsf{H}'} - 1$. In the worst case, they need to check for every element m' of $\mathsf{Roots}(\mathsf{H}'(X) - d)$ whether Pco$(m', c) = 1$. Collecting the probabilities and folding adversaries A_0 and A_1 into one single adversary A proves the theorem. □

4.4 The resulting KEM

For completeness, we combine transformations T_1 and QT_2 from the previous sections to obtain $\mathsf{QFO} = \mathsf{T}_1 \circ \mathsf{QT}_2$. To a public-key encryption scheme $\mathsf{PKE} = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ with message space $\mathcal{M} = \{0,1\}^n$ and randomness space $\mathcal{R}$, and hash functions $\mathsf{G} : \mathcal{M} \to \mathcal{R}$, $\mathsf{H} : \{0,1\}^* \to \{0,1\}^n$ and $\mathsf{H}' : \{0,1\}^n \to \{0,1\}^n$, we associate $\mathsf{QKEM} = \mathsf{QFO}[\mathsf{PKE}, \mathsf{G}, \mathsf{H}, \mathsf{H}'] := \mathsf{QT}_2[\mathsf{T}_1[\mathsf{PKE}, \mathsf{G}], \mathsf{H}, \mathsf{H}']$. The algorithms of $\mathsf{QKEM} = (\mathsf{Gen}, \mathsf{QEncaps}, \mathsf{QDecaps})$ are given in Figure 24. The following corollary is obtained by combining Theorems 4.2 and 4.3.

```
$\underline{\mathsf{QEncaps}(pk)}$
$m \xleftarrow{\$} \mathcal{M}$
$c := \mathsf{Enc}(pk, m; \mathsf{G}(m))$
$K := \mathsf{H}(m)$
$d := \mathsf{H}'(m)$
return $(K, c, d)$

$\underline{\mathsf{QDecaps}(sk, c, d)}$
$m' := \mathsf{Dec}(sk, c)$
if $c = \mathsf{Enc}(pk, m', \mathsf{G}(m'))$ and $\mathsf{H}'(m') = d$
    return $K := \mathsf{H}(m')$
else return $\perp$
```

Figure 24: $\mathsf{IND\text{-}CCA}$ secure $\mathsf{QKEM} = \mathsf{QFO}[\mathsf{PKE}, \mathsf{G}, \mathsf{H}, \mathsf{H}']$ obtained from PKE.

Corollary 4.4 *If* PKE *is* δ*-correct, so is* $\mathsf{QKEM} = \mathsf{QFO}[\mathsf{PKE}, \mathsf{G}, \mathsf{H}, \mathsf{H}']$. *For any quantum adversary* B *issuing at most* q_D *(classical) queries to the decapsulation oracle* QDECAPS*, and at most* q_G *(*q_H*,* $q_{\mathsf{H}'}$*) queries to the quantum random oracles* $|\mathsf{G}\rangle$ *(*$|\mathsf{H}\rangle$*,* $|\mathsf{H}'\rangle$*), there exists a quantum adversary* A *such that* $\mathrm{Adv}^{\mathsf{IND\text{-}CCA}}_{\mathsf{QKEM}}(\mathsf{B})$

$$\leq 4(q_\mathsf{H} + q_{\mathsf{H}'}) \cdot \sqrt{q_D q_{\mathsf{H}'} \cdot \delta + q_\mathsf{G} \cdot \sqrt{\mathrm{Adv}^{\mathsf{OW\text{-}CPA}}_{\mathsf{PKE}}(\mathsf{A})}}\ ,$$

and the running time of A *is about that of* B.

References

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

[AOP+17] Martin R. Albrecht, Emmanuela Orsini, Kenneth G. Paterson, Guy Peer, and Nigel P. Smart. Tightly secure ring-lwe based key encapsulation with short ciphertexts. Cryptology ePrint Archive, Report 2017/354, 2017. `http://eprint.iacr.org/2017/354`.

[BBC+98] Robert Beals, Harry Buhrman, Richard Cleve, Michele Mosca, and Ronald de Wolf. Quantum lower bounds by polynomials. In *39th FOCS*, pages 352–361. IEEE Computer Society Press, November 1998.

[BDF+11] Dan Boneh, Özgür Dagdelen, Marc Fischlin, Anja Lehmann, Christian Schaffner, and Mark Zhandry. Random oracles in a quantum world. In Dong Hoon Lee and Xiaoyun Wang, editors, *ASIACRYPT 2011*, volume 7073 of *LNCS*, pages 41–69. Springer, Heidelberg, December 2011.

[BLK00] Joonsang Baek, Byoungcheon Lee, and Kwangjo Kim. Secure length-saving ElGamal encryption under the computational Diffie-Hellman assumption. In Ed Dawson, Andrew Clark, and Colin Boyd, editors, *ACISP 00*, volume 1841 of *LNCS*, pages 49–58. Springer, Heidelberg, July 2000.

[BR93] Mihir Bellare and Phillip Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. In V. Ashby, editor, *ACM CCS 93*, pages 62–73. ACM Press, November 1993.

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

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

[CKS08] David Cash, Eike Kiltz, and Victor Shoup. The twin Diffie-Hellman problem and applications. In Nigel P. Smart, editor, *EUROCRYPT 2008*, volume 4965 of *LNCS*, pages 127–145. Springer, Heidelberg, April 2008.

[CKS09] David Cash, Eike Kiltz, and Victor Shoup. The twin Diffie-Hellman problem and applications. *Journal of Cryptology*, 22(4):470–504, October 2009.

[CS03] Ronald Cramer and Victor Shoup. Design and analysis of practical public-key encryption schemes secure against adaptive chosen ciphertext attack. *SIAM Journal on Computing*, 33(1):167–226, 2003.

[Den03] Alexander W. Dent. A designer's guide to KEMs. In Kenneth G. Paterson, editor, *9th IMA International Conference on Cryptography and Coding*, volume 2898 of *LNCS*, pages 133–151. Springer, Heidelberg, December 2003.

[DNR04] Cynthia Dwork, Moni Naor, and Omer Reingold. Immunizing encryption schemes from decryption errors. In Christian Cachin and Jan Camenisch, editors, *EUROCRYPT 2004*, volume 3027 of *LNCS*, pages 342–360. Springer, Heidelberg, May 2004.

[FO99] Eiichiro Fujisaki and Tatsuaki Okamoto. Secure integration of asymmetric and symmetric encryption schemes. In Michael J. Wiener, editor, *CRYPTO'99*, volume 1666 of *LNCS*, pages 537–554. Springer, Heidelberg, August 1999.

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

[GMMV05] David Galindo, Sebastià Martín, Paz Morillo, and Jorge L. Villar. Fujisaki-okamoto hybrid encryption revisited. *Int. J. Inf. Sec.*, 4(4):228–241, 2005.

[KL07] Jonathan Katz and Yehuda Lindell. *Introduction to Modern Cryptography*. Chapman and Hall/CRC Press, 2007.

[KML03] Eike Kiltz and John Malone-Lee. A general construction of IND-CCA2 secure public key encryption. In Kenneth G. Paterson, editor, *9th IMA International Conference on Cryptography and Coding*, volume 2898 of *LNCS*, pages 152–166. Springer, Heidelberg, December 2003.

[LPR10] Vadim Lyubashevsky, Chris Peikert, and Oded Regev. On ideal lattices and learning with errors over rings. In Henri Gilbert, editor, *EUROCRYPT 2010*, volume 6110 of *LNCS*, pages 1–23. Springer, Heidelberg, May 2010.

[LPR13] Vadim Lyubashevsky, Chris Peikert, and Oded Regev. A toolkit for ring-LWE cryptography. In Thomas Johansson and Phong Q. Nguyen, editors, *EUROCRYPT 2013*, volume 7881 of *LNCS*, pages 35–54. Springer, Heidelberg, May 2013.

[NIS17] NIST. National institute for standards and technology. postquantum crypto project, 2017. `http://csrc.nist.gov/groups/ST/post-quantum-crypto/`.

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

[Pei14] Chris Peikert. Lattice cryptography for the internet. Cryptology ePrint Archive, Report 2014/070, 2014. `http://eprint.iacr.org/2014/070`.

[Reg05] Oded Regev. On lattices, learning with errors, random linear codes, and cryptography. In Harold N. Gabow and Ronald Fagin, editors, *37th ACM STOC*, pages 84–93. ACM Press, May 2005.

[RS92] Charles Rackoff and Daniel R. Simon. Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attack. In Joan Feigenbaum, editor, *CRYPTO'91*, volume 576 of *LNCS*, pages 433–444. Springer, Heidelberg, August 1992.

[Sho04a] Victor Shoup. ISO 18033-2: An emerging standard for public-key encryption. `http://shoup.net/iso/std6.pdf`, December 2004. Final Committee Draft.

[Sho04b] Victor Shoup. Sequences of games: a tool for taming complexity in security proofs. Cryptology ePrint Archive, Report 2004/332, 2004. `http://eprint.iacr.org/2004/332`.

[TU16] Ehsan Ebrahimi Targhi and Dominique Unruh. Post-quantum security of the fujisaki-okamoto and OAEP transforms. In Martin Hirt and Adam D. Smith, editors, *TCC 2016-B, Part II*, volume 9986 of *LNCS*, pages 192–216. Springer, Heidelberg, October / November 2016.

[Unr14] Dominique Unruh. Revocable quantum timed-release encryption. In Phong Q. Nguyen and Elisabeth Oswald, editors, *EUROCRYPT 2014*, volume 8441 of *LNCS*, pages 129–146. Springer, Heidelberg, May 2014.

[Unr15] Dominique Unruh. Non-interactive zero-knowledge proofs in the quantum random oracle model. In Elisabeth Oswald and Marc Fischlin, editors, *EUROCRYPT 2015, Part II*, volume 9057 of *LNCS*, pages 755–784. Springer, Heidelberg, April 2015.

[Zha12] Mark Zhandry. Secure identity-based encryption in the quantum random oracle model. In Reihaneh Safavi-Naini and Ran Canetti, editors, *CRYPTO 2012*, volume 7417 of *LNCS*, pages 758–775. Springer, Heidelberg, August 2012.

Acknowledgments

We would like to thank Andreas Hülsing and Dominique Unruh for interesting discussions on the FO transformation. We are also grateful to Krzysztof Pietrzak and Victor Shoup for discussions on Section 3.5.