An Asymptotically Optimal Method for Converting Bit Encryption to Multi-Bit Encryption

Takahiro Matsuda(✉) and Goichiro Hanaoka

National Institute of Advanced Industrial Science and Technology (AIST),
Tokyo, Japan
{t-matsuda,hanaoka-goichiro}@aist.go.jp

Abstract. Myers and Shelat (FOCS 2009) showed how to convert a chosen ciphertext secure (CCA secure) PKE scheme that can encrypt only 1-bit plaintexts into a CCA secure scheme that can encrypt arbitrarily long plaintexts (via the notion of key encapsulation mechanism (KEM) and hybrid encryption), and subsequent works improved efficiency and simplicity. In terms of efficiency, the best known construction of a CCA secure KEM from a CCA secure 1-bit PKE scheme, has the public key size $\Omega(k) \cdot |pk|$ and the ciphertext size $\Omega(k^2) \cdot |c|$, where k is a security parameter, and $|pk|$ and $|c|$ denote the public key size and the ciphertext size of the underlying 1-bit scheme, respectively.

In this paper, we show a new CCA secure KEM based on a CCA secure 1-bit PKE scheme which achieves the public key size $2 \cdot |pk|$ and the ciphertext size $(2k+o(k)) \cdot |c|$. These sizes are asymptotically optimal in the sense that they are the same as those of the simplest "bitwise-encrypt" construction (seen as a KEM by encrypting a k-bit random session-key) that works for the chosen plaintext attack and non-adaptive chosen ciphertext attack settings. We achieve our main result by developing several new techniques and results on the "double-layered" construction (which builds a KEM from an inner PKE/KEM and an outer PKE scheme) by Myers and Shelat and on the notion of detectable PKE/KEM by Hohenberger, Lewko, and Waters (EUROCRYPT 2012).

1 Introduction

1.1 Background and Motivation

In this paper, we revisit the problem of how to construct a chosen ciphertext secure (CCA2, or just CCA) public key encryption (PKE) scheme that can encrypt plaintexts of arbitrary length from a CCA secure PKE scheme whose plaintext space is only 1-bit. (Hereafter, we call a PKE scheme whose plaintext space is $\{0,1\}^n$ an *n-bit PKE scheme*.) It is well-known that if we only consider chosen plaintext attack (CPA) and non-adaptive chosen ciphertext attack (CCA1) settings, then the simple(st) "*bitwise-encrypt*" construction suffices, in which a plaintext is encrypted bit-by-bit (under the same public key) by a 1-bit

T. Iwata and J.H. Cheon (Eds.): ASIACRYPT 2015, Part I, LNCS 9452, pp. 415–442, 2015.
DOI: 10.1007/978-3-662-48797-6_18

PKE scheme, and the concatenation of all ciphertexts is regarded as a ciphertext of the construction. However, for the CCA setting, until recently, the simple question of how (and even whether) one can realize such a "1-bit-to-multi-bit" conversion had been left open.

This open problem was resolved affirmatively by Myers and Shelat [20]. They actually constructed a CCA secure key encapsulation mechanism (KEM) which encrypts a random session-key, and can be used together with a CCA secure symmetric key encryption (SKE) scheme to achieve a full-fledged CCA secure PKE scheme via hybrid encryption [8]. One of the important steps of the approach by Myers and Shelat is to consider the "*double-layered*" construction of a KEM from an "inner" PKE scheme and an "outer" PKE scheme, where the inner ciphertext encrypts a plaintext (or a session-key if one wants to construct a KEM) and a randomness used for outer encryption, and the outer ciphertext encrypts the inner ciphertext using the randomness encrypted in the inner ciphertext. To decrypt a ciphertext, one first decrypts the outer ciphertext, and then the resulting inner ciphertext, to recover a plaintext and a randomness (for outer encryption), and the plaintext is output if the re-encryption of the inner ciphertext using the recovered randomness results in the outer ciphertext. Myers and Shelat showed that if the outer scheme that is built from a 1-bit scheme satisfies the security notion called "unquoted CCA" (UCCA) security (which is a weaker security notion than CCA security that can be considered only for a PKE scheme constructed based on 1-bit PKE scheme), and the inner scheme satisfies "1-wise non-malleability against UCCA" (which has a similar flavor to 1-bounded CCA security [7]), the resulting construction achieves CCA security.

The efficiency and simplicity of the construction by Myers and Shelat were improved by Hohenberger, Lewko, and Waters [16]. Specifically, they introduced the notion of a *detectable PKE* scheme, which is a PKE scheme that has an efficiently computable predicate F as part of the syntax, and whose security notions are defined with respect to this F. In particular, they introduced the notions of *detectable CCA* (DCCA) security (which is a relaxed variant of CCA security) and *unpredictability*, and considered a construction which has a mixed flavor of the double-layered construction of Myers and Shelat, and the double (parallel) encryption of Naor and Yung [21] (this construction has two PKE schemes for the outer encryption). They showed that if the "inner" PKE scheme satisfies DCCA security and unpredictability, and the "outer" PKE schemes are CPA secure and 1-bounded CCA secure [7], respectively, then the resulting PKE scheme is CCA secure. They also showed that the "bitwise-encrypt" construction based on a CCA secure 1-bit PKE scheme yields a DCCA secure and unpredictable detectable PKE scheme for long plaintexts, and thus achieves a 1-bit-to-multi-bit conversion for CCA security. (In their construction, in fact a 1-bit scheme satisfying only DCCA security and unpredictability suffices as the building block.) The efficiency of the construction in [16] was further improved by Matsuda and Hanaoka [19] using the ideas and techniques of hybrid encryption.

Despite the elegant ideas employed in [16,19,20], however, even in the best construction of [19] (in terms of efficiency), the public key size is $\Omega(k) \cdot |pk|$ and the ciphertext size (when seen as a KEM) is $\Omega(k^2) \cdot |c|$, where k is a security parameter, and $|pk|$ and $|c|$ denote the public key size and the ciphertext size of a CCA secure 1-bit scheme, respectively. On the other hand, for constructing a CPA (resp. CCA1) secure KEM from a CPA (resp. CCA1) secure 1-bit scheme, one can use the above mentioned bitwise-encrypt construction in which one encrypts a k-bit random string and regards this as a session-key of a KEM. Note that the public key size of this KEM is just $|pk|$ and the ciphertext size is $k \cdot |c|$. Compared to this simplest and most straightforward method, in the CCA setting, the known constructions have the public key size and the ciphertext size that are at least $\Omega(k)$ times larger.

Motivated by the above, in this paper we study the following question: *How efficient can a* 1*-bit-to-multi-bit conversion for CCA security be?*

1.2 Our Contributions

As our main result, we show a new 1-bit-to-multi-bit construction for the CCA setting, i.e., a construction of a CCA secure KEM based on a CCA secure 1-bit PKE scheme, with much better asymptotic efficiency than the existing constructions. Specifically, our construction achieves the public key size $2 \cdot |pk|$, and the ciphertext size $(2k + o(k)) \cdot |c| = O(k) \cdot |c|$, which are asymptotically optimal in the sense that these sizes are (except for a constant factor) the same as for the simple bitwise-encrypt construction for CPA and CCA1 security.

We achieve our main result by developing several new techniques and results on the double-layered construction of Myers and Shelat [20] and on the notion of detectable PKE/KEM by Hohenberger, Lewko, and Waters [16]. Our technical contributions in this paper lie in (1) coming up with appropriate security notions for detectable PKE/KEM so that we can conduct CCA security proofs for the double-layered construction using the language of detectable PKE/KEM (without addressing the details of how each of the inner and outer schemes is constructed) which we believe helps us understanding our proposed construction (and more generally the double-layered approach itself) in a clearer manner, and (2) showing how one can realize the inner and outer schemes (satisfying the requirements of our security proofs) from a CCA secure 1-bit PKE scheme, so that the resulting CCA secure KEM achieves asymptotically optimal efficiency with respect to the bitwise-encrypt construction.

Below we explain more technical details of our results.

New Security Notions for Detectable PKE/KEM. In Sect. 3, we introduce new security notions for detectable PKE and detectable KEMs. Recall that DCCA security of [16] is defined like ordinary CCA security, except that in the security experiment, the decryption oracle is restricted according to the predicate F (which is a part of the syntax of detectable PKE/KEM): an adversary is not allowed to query a ciphertext c such that $\mathsf{F}(c^*, c) = 1$ where c^* is the challenge ciphertext. The first notion we introduce is a weak form of *non-malleability*

[3,12,22] under DCCA that we simply name wNM-DCCA *security*, which is defined like DCCA security except that we allow an adversary to make one "unrestricted" decryption query (which is not affected by the restriction of F). We also introduce an even weaker variant, which is a "replayable"-CCA-analogue [4] of wNM-DCCA security, which we call wRNM-DCCA *security*, that is defined like wNM-DCCA security except that the final unrestricted decryption query (and only this query) is answered like a decryption query in the replayable CCA security.

We also introduce a new security notion for detectable PKE/KEM that we call *randomness-inextractability*. Recall that a DCCA secure detectable PKE scheme is meaningful only if it also satisfies another security notion that prevents the predicate F from outputting 1 for every input (which makes DCCA security equivalent to CPA security). *Unpredictability* [16] is one example of a security notion that prevents DCCA security from being trivial, which ensures that a ciphertext c satisfying $\mathsf{F}(c^*, c) = 1$ is hard to find without seeing c^*. Randomness-inextractability is another such security notion for detectable PKE: Informally, it requires that if an adversary is given a ciphertext c^* (that encrypts a plaintext m of the adversary's choice), it cannot come up with a pair of a (possibly different) plaintext m' and randomness r' such that $\mathsf{F}(c^*, c') = 1$, where c' is the encryption of m' generated using the randomness r'. We also show that randomness-inextractability and unpredictability do not imply each other, even if we combine one notion with wNM-DCCA security. See Sect. 3 for the details.

New CCA Security Proofs for the Double-Layered Construction Based on Detectable PKE/KEM. In Sect. 4, we show our main technical results: two new CCA security proofs for the double-layered construction of Myers and Shelat [20]. Our first security proof shows that if the inner KEM is a detectable KEM satisfying DCCA security and unpredictability, and the outer PKE scheme is a detectable PKE scheme satisfying wRNM-DCCA security and randomness-inextractability, then the KEM obtained from the double-layered construction is CCA secure. Our main result with asymptotically optimal efficiency is obtained from this security proof.

Our second security proof shows that if the inner KEM is wNM-DCCA secure and unpredictable, and the outer PKE scheme is DCCA secure and randomness-inextractable, then the KEM obtained from the double-layered construction is CCA secure. Interestingly, this security proof can be seen as a generalization of Myers-Shelat's original security proof of their construction [20].

Both of the security proofs have similar flavors to the security proofs of [16,19]. Namely, DCCA security of the inner KEM guarantees that a session-key (hidden in the challenge ciphertext) is random as long as an adversary does not submit a "dangerous" decryption query (which are defined with respect to the predicate F from the inner detectable KEM), and we then upperbound the probability that the adversary comes up with such "dangerous" decryption queries to be negligible by the combination of the security properties of the outer PKE scheme and the inner KEM. However, unlike the previous works [16,19] that use a "detectable" primitive only for the inner scheme, we employ a detectable primitive also for the outer scheme. Consequently, we have to deal with two types

of "dangerous" decryption queries in the security proofs: an "inner-dangerous" query and an "outer-dangerous" query, which, as the names indicate, are related to the inner KEM and the outer PKE scheme, respectively. Our two security proofs differ in the treatment of the inner- and outer-dangerous queries, which lead to the difference between which of the inner KEM or the outer PKE scheme needs to be "non-malleable" under DCCA. In both of the proofs, randomness-inextractability of the outer PKE scheme is used to show that the adversary's outer-dangerous queries do not help.

We also show an evidence that indicates that our reliance on "non-malleability" under DCCA for either the inner KEM or the outer PKE scheme would be unavoidable, by showing a counterexample for the double-layered construction that does not achieve CCA security if the inner and outer schemes only satisfy DCCA security, unpredictability, and randomness-inextractability. For the details, see Sect. 4.

A Detectable PKE Scheme Satisfying `wRNM-DCCA` *Security and Randomness-Inextractability from CCA Secure* 1*-bit PKE.* In Sect. 5, we show a construction of a detectable PKE scheme satisfying `wRNM-DCCA` security and randomness-inextractability, using a CCA secure 1-bit PKE scheme and a *non-malleable code* [13] for "bitwise-tampering and bit-level permutations" [1,2]. The idea of this construction is based on the recent result by Agrawal et al. [2] who showed how to transform a 1-bit commitment scheme secure against chosen commitment attacks (CCA) into a non-malleable string commitment scheme: We first encode a plaintext by a non-malleable code, and then do "bitwise-encryption" of the encoded value by a CCA secure 1-bit PKE scheme. (Due to its structure, we call this construction the "*Encode-then-Bitwise-Encrypt*" (EtBE) construction.) Our contribution regarding this construction is to clarify that the approach of [2] also works well for detectable PKE as we require.

Agrawal et al. [1] recently constructed a non-malleable code for the above mentioned class of functions with "optimal rate", meaning that the ratio between the length n of a codeword and the length k of a message can be made arbitrarily close to 1 (i.e. $n = k + o(k)$). We employ this non-malleable code to achieve the asymptotic efficiency of our proposed KEM.

The Proposed 1*-Bit-to-Multi-Bit Conversion, and More.* Our main result, i.e. a CCA secure KEM from a CCA secure 1-bit PKE scheme that achieves optimal asymptotic efficiency in terms of the public key and ciphertext sizes, is obtained by using the above mentioned detectable PKE scheme (together with some hybrid encryption techniques) as the outer PKE scheme, and using the bitwise-encrypt construction of a detectable KEM as the inner KEM, in the double-layered construction, via our first security proof. In Sect. 6, we show the full description of our construction. As noted above, our construction uses only two key pairs of the underlying 1-bit PKE scheme.

Interestingly, there we also show that if a 2-bit PKE scheme can be used instead of a 1-bit PKE scheme, then one can construct a CCA secure KEM (with almost the same construction as our main construction) that uses only one key pair.

On the Necessity of Two Key Pairs. As mentioned above, our proposed KEM from a 1-bit PKE scheme uses two key pairs of the underlying CCA secure 1-bit PKE scheme. Given this, it is natural to ask if the number 2 of key pairs of the underlying 1-bit scheme is optimal for 1-bit-to-multi-bit constructions for CCA security. Although we could not answer this question affirmatively or negatively, we show that the one-key variant of our proposed construction is vulnerable to a CCA attack. (This result is shown in the full version.) This negative result shows a necessity of different techniques and ideas than ours towards answering the question. It also contrasts strikingly with our 2-bit-to-multi-bit construction for CCA security that uses only one key pair of the underlying 2-bit scheme.

We leave it as an open problem to clarify whether one can achieve a 1-bit-to-multi-bit conversion using only one key pair of the underlying 1-bit scheme, or it is generally impossible.

1.3 Related Work

The double-layered construction [16,20], and extension of the plaintext space of encryption schemes based on it, have been used in several works: Lin and Tessaro [18] showed how to turn a 1-bit PKE scheme whose correctness is not perfect and which only satisfies weak CCA security (weak in the sense that an adversary may have bounded but non-negligible CCA advantage), into a PKE scheme (with a large plaintext space) satisfying ordinary CCA security, via the construction of [16]. Dachman-Soled et al. [9] studied the notion of "enhanced" CCA security for PKE schemes with randomness recovery property, where the decryption oracle in the security experiment returns not only the decryption result of a queried ciphertext but also a randomness that is consistent with the ciphertext, and (among other things) showed that the construction of [16] can be used to achieve a 1-bit-to-multi-bit conversion for enhanced CCA security. Most recently, Kitagawa et al. [17] showed that a simpler variant of the double-layered construction which does not have validity check by re-encryption in the decryption algorithm, can be used to extend the plaintext space of PKE satisfying key-dependent message (KDM) security against CCA with respect to projection functions (projection-KDM-CCA security).

Very recently, Coretti et al. [6] showed a 1-bit-to-multi-bit conversion for a PKE scheme. However, the security notion considered in their construction is so-called "self-destruct" CCA security, which is defined like ordinary CCA security except that in the security experiment, once an adversary submits an invalid ciphertext (which does not decrypt to a valid plaintext) as a decryption query, the decryption oracle "self-destructs", i.e. it will not answer to subsequent decryption queries. This security notion is strictly weaker than ordinary CCA security. Furthermore, in another recent work, Coretti et al. [5] considered non-malleability under self-destruct CCA, which is also strictly weaker than ordinary CCA security, and showed a 1-bit-to-multi-bit conversion for a PKE scheme satisfying this security notion. The 1-bit-to-multi-bit constructions of [5,6] share the same idea with Agrawal et al.'s conversion (and hence with our "outer" PKE scheme): first encode a plaintext by a suitable non-malleable code, and

then do bitwise encryption. The main differences between these works [5,6] and our "double-layered" construction are: (1) Ours achieves ordinary (full) CCA security, while they achieve weaker security notions. (2) Our construction uses only two key pairs of the underlying 1-bit scheme, while the constructions in [5,6] use $O(k)$ key pairs, of the building block 1-bit scheme. (3) The requirements of the used non-malleable codes are all different: [5,6] need stronger form of non-malleability called "continuous" non-malleability [15] (and its extension), while we only need the original definition of non-malleability in [13] that captures "one-time" tampering.; The tampering functions with respect to which non-malleability is considered in [5,6] are based on bit-wise tampering (extended to take into account continuous non-malleability), while ours requires additionally non-malleability against bit-level permutation (as in [1,2]).

Paper Organization. The rest of this paper is organized as follows: Sect. 2 reviews the basic notation and definitions of cryptographic primitives. In Sect. 3, we define new security notions for detectable PKE, and also show several facts on them. In Sect. 4, we show our main technical result: new security proofs for the "double-layered" construction. We also explain some evidence that justifies our reliance on non-malleability under DCCA. In Sect. 5, we show how to build a detectable PKE scheme satisfying our new security notions based on a CCA secure 1-bit PKE scheme and a non-malleable code. In Sect. 6, we provide the full description of our proposed 1-bit-to-multi-bit construction. There we also explain our 2-bit-to-multi-bit construction with a single key pair. We give a comparison among 1-bit-to-multi-bit constructions in Sect. 7.

Due to space limitation, the proofs of the theorems and lemmas in this paper are omitted and will be given in the full version, and we only give proof sketches or intuitive explanations.

2 Preliminaries

In this section, we review the basic notation and the definitions for cryptographic primitives.

Basic Notation. $\mathbb{N}$ denotes the set of all natural numbers. For $n \in \mathbb{N}$, we define $[n] := \{1, \ldots, n\}$. "$x \leftarrow y$" denotes that x is chosen uniformly at random from y if y is a finite set, x is output from y if y is a function or an algorithm, or y is assigned to x otherwise. If x and y are strings, then "$|x|$" denotes the bit-length of x, "$x\|y$" denotes the concatenation x and y, and "$(x \stackrel{?}{=} y)$" is defined to be 1 if $x = y$ and 0 otherwise. "(P)PTA" stands for a *(probabilistic) polynomial time algorithm*. For a finite set S, "$|S|$" denotes its size. If $\mathcal{A}$ is a probabilistic algorithm then "$y \leftarrow \mathcal{A}(x; r)$" denotes that $\mathcal{A}$ computes y as output by taking x as input and using r as randomness. If furthermore $\mathcal{O}$ is an algorithm, then "$\mathcal{A}^{\mathcal{O}}$" denotes that $\mathcal{A}$ has oracle access to $\mathcal{O}$. A function $\epsilon(\cdot) : \mathbb{N} \to [0, 1]$ is said to be *negligible* if for all positive polynomials $p(k)$ and all sufficiently large $k \in \mathbb{N}$, we have $\epsilon(k) < 1/p(k)$. Throughout this paper, we use the character "k" to denote a security parameter.

2.1 (Detectable) Public Key Encryption

A public key encryption (PKE) scheme Π consists of the three PPTAs $(\mathsf{PKG}, \mathsf{Enc}, \mathsf{Dec})$ with the following interface:

$$\underline{\textbf{Key Generation:}} \quad (pk, sk) \leftarrow \mathsf{PKG}(1^k) \qquad \underline{\textbf{Encryption:}} \quad c \leftarrow \mathsf{Enc}(pk, m) \qquad \underline{\textbf{Decryption:}} \quad m \text{ (or } \perp) \leftarrow \mathsf{Dec}(sk, c)$$

where Dec is a deterministic algorithm, (pk, sk) is a public/secret key pair, and c is a ciphertext of a plaintext m under pk. We say that a PKE scheme satisfies *correctness* if for all $k \in \mathbb{N}$, all keys (pk, sk) output from $\mathsf{PKG}(1^k)$, and all plaintexts m, it holds that $\mathsf{Dec}(sk, \mathsf{Enc}(pk, m)) = m$.

Detectable PKE. In this paper, we use the notion of *detectable PKE* as defined in [16]. It is a PKE scheme that has a predicate F that tests whether two ciphertexts c and c' are "related" in the sense that to decrypt c, the information of the decryption result of c' is useful (and hence, revealing the decryption result of c' is "dangerous"). This predicate F is used to define multiple security notions of the primitive, and hence we explicitly define it as a part of the syntax of the primitive (this approach is also taken in [16,19]).

Formally, a tuple of PPTAs $\Pi = (\mathsf{PKG}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{F})$ is said to be a *detectable* PKE scheme if $(\mathsf{PKG}, \mathsf{Enc}, \mathsf{Dec})$ constitutes PKE, and F is a predicate that takes a public key pk and two ciphertexts c, c' as input, and outputs either 0 or 1.

$\mathsf{Expt}^{\mathtt{ATK}}_{\Pi,\mathcal{A}}(k)$:
$(pk, sk) \leftarrow \mathsf{PKG}(1^k)$
$(m_0, m_1, \mathsf{st}) \leftarrow \mathcal{A}_1^{\mathcal{O}(\cdot)}(pk)$
$b \leftarrow \{0,1\}$
$c^* \leftarrow \mathsf{Enc}(pk, m_b)$
$b' \leftarrow \mathcal{A}_2^{\mathcal{O}(\cdot)}(\mathsf{st}, c^*)$
Return $(b' \stackrel{?}{=} b)$

$\mathsf{Expt}^{\mathtt{ATK}}_{\Gamma,\mathcal{A}}(k)$:
$(pk, sk) \leftarrow \mathsf{KKG}(1^k)$
$(c^*, K_1^*) \leftarrow \mathsf{Encap}(pk)$
$K_0^* \leftarrow \mathcal{K}$
$b \leftarrow \{0,1\}$
$b' \leftarrow \mathcal{A}^{\mathcal{O}(\cdot)}(pk, c^*, K_b^*)$
Return $(b' \stackrel{?}{=} b)$

$\mathsf{Expt}^{\mathcal{F}\text{-}\mathtt{NM}}_{\mathcal{C},\mathcal{A}}(k)$:
$(f, m_0, m_1, \mathsf{st}) \leftarrow \mathcal{A}_1(1^k)$
$b \leftarrow \{0,1\}$
$s^* \leftarrow \mathsf{E}(1^k, m_b)$
$s' \leftarrow f(s^*)$
$m' \leftarrow \mathsf{D}(1^k, s')$
If $m' \in \{m_0, m_1\}$ then $m' \leftarrow \mathsf{same}$
$b' \leftarrow \mathcal{A}_2(\mathsf{st}, m')$
Return $(b' \stackrel{?}{=} b)$

$\mathsf{Expt}^{\mathtt{UNP}}_{\Pi,\mathcal{A}}(k)$:
$(pk, sk) \leftarrow \mathsf{PKG}(1^k)$
$(m, c) \leftarrow \mathcal{A}^{\mathcal{O}(\cdot)}(pk)$
$c^* \leftarrow \mathsf{Enc}(pk, m)$
Return $\mathsf{F}(pk, c^*, c)$

$\mathsf{Expt}^{\mathtt{UNP}}_{\Gamma,\mathcal{A}}(k)$:
$(pk, sk) \leftarrow \mathsf{KKG}(1^k)$
$c \leftarrow \mathcal{A}^{\mathcal{O}(\cdot)}(pk)$
$(c^*, K^*) \leftarrow \mathsf{Encap}(pk)$
Return $\mathsf{F}(pk, c^*, c)$

Fig. 1. The experiments for defining the security of detectable PKE (left-top/bottom), of detectable KEM (center-top/bottom), and of an $\mathcal{F}$-non-malleable code (right). In the $\mathtt{ATK}(\in \{\mathtt{CCA}, \mathtt{DCCA}\})$ and $\mathtt{UNP}$ experiments for PKE (resp. KEM), $\mathcal{O}(\cdot)$ is the decryption oracle $\mathsf{Dec}(sk, \cdot)$ (resp. decapsulation oracle $\mathsf{Decap}(sk, \cdot)$). In the $\mathtt{CCA}$ (resp. $\mathtt{DCCA}$) experiment for PKE, $\mathcal{A}_2$ is not allowed to query c^* (resp. ciphertexts c such that $\mathsf{F}(pk, c^*, c) = 1$). Similar restrictions apply to $\mathcal{A}$ in the $\mathtt{CCA}/\mathtt{DCCA}$ experiment for KEMs.

We require that for all $k \in \mathbb{N}$, all public keys pk output by $\mathsf{PKG}(1^k)$, and all ciphertexts c output by $\mathsf{Enc}(pk, \cdot)$, we have $\mathsf{F}(pk, c, c) = 1$.[1]

Security Notions. Here we recall *chosen ciphertext security (*$\mathtt{CCA}$ *security)* for PKE, and *detectable CCA (*$\mathtt{DCCA}$*) security* and *unpredictability* for detectable PKE [16].

Let $\mathtt{ATK} \in \{\mathtt{CCA}, \mathtt{DCCA}\}$. For a (detectable) PKE scheme Π and an adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$, consider the $\mathtt{ATK}$ experiment $\mathsf{Expt}^{\mathtt{ATK}}_{\Pi,\mathcal{A}}(k)$ described in Fig. 1 (left-top). In the experiment, it is required that $|m_0| = |m_1|$, and $\mathcal{A}_2$ is not allowed to submit the "prohibited" queries to the decryption oracle: If $\mathtt{ATK} = \mathtt{CCA}$, then the prohibited query is c^*, and if $\mathtt{ATK} = \mathtt{DCCA}$, then the prohibited queries are c satisfying $\mathsf{F}(pk, c^*, c) = 1$. We say that a (detectable) PKE scheme Π is $\mathtt{ATK}$ secure if for all PPTAs $\mathcal{A}$, $\mathsf{Adv}^{\mathtt{ATK}}_{\Pi,\mathcal{A}}(k) := 2 \cdot |\Pr[\mathsf{Expt}^{\mathtt{ATK}}_{\Pi,\mathcal{A}}(k) = 1] - 1/2|$ is negligible.

For a detectable PKE scheme Π (with predicate F) and an adversary $\mathcal{A}$, consider the unpredictability experiment $\mathsf{Expt}^{\mathtt{UNP}}_{\Pi,\mathcal{A}}(k)$ described in Fig. 1 (left-bottom). We say that a detectable PKE scheme Π is *unpredictable* if for all PPTAs $\mathcal{A}$, $\mathsf{Adv}^{\mathtt{UNP}}_{\Pi,\mathcal{A}}(k) := \Pr[\mathsf{Expt}^{\mathtt{UNP}}_{\Pi,\mathcal{A}}(k) = 1]$ is negligible.

2.2 (Detectable) Key Encapsulation Mechanism

A key encapsulation mechanism (KEM) Γ consists of the three PPTAs $(\mathsf{KKG}, \mathsf{Encap}, \mathsf{Decap})$ with the following interface:

Key Generation:	**Encapsulation:**	**Decapsulation:**
$(pk, sk) \leftarrow \mathsf{KKG}(1^k)$	$(c, K) \leftarrow \mathsf{Encap}(pk)$	K (or $\perp$) $\leftarrow \mathsf{Decap}(sk, c)$

where Decap is a deterministic algorithm, (pk, sk) is a public/secret key pair that defines a session-key space $\mathcal{K}$, and c is a ciphertext of a session-key $K \in \mathcal{K}$ under pk. We say that a KEM satisfies *correctness* if for all $k \in \mathbb{N}$, all keys (pk, sk) output from $\mathsf{KKG}(1^k)$ and all ciphertext/session-key pairs (c, K) output from $\mathsf{Encap}(pk)$, it holds that $\mathsf{Decap}(sk, c) = K$.

We also define a KEM-analogue of detectable PKE, which we call *detectable KEM*, as a KEM that has an efficiently computable predicate F whose interface is exactly the same as that of detectable PKE.

Security Notions. Here we review the definition of $\mathtt{CCA}$ *security* for a KEM, and the definitions of $\mathtt{DCCA}$ *security* and *unpredictability* for a detectable KEM.

Let $\mathtt{ATK} \in \{\mathtt{CCA}, \mathtt{DCCA}\}$. For a (detectable) KEM Γ and an adversary $\mathcal{A}$, consider the $\mathtt{ATK}$ experiment $\mathsf{Expt}^{\mathtt{ATK}}_{\Gamma,\mathcal{A}}(k)$ described in Fig. 1 (center-top). In the experiment, $\mathcal{A}$ is not allowed to submit the "prohibited" queries that are defined in the same way as those for the PKE case. We say that a (detectable) KEM Γ is $\mathtt{ATK}$ secure if for all PPTAs $\mathcal{A}$, $\mathsf{Adv}^{\mathtt{ATK}}_{\Gamma,\mathcal{A}}(k) := 2 \cdot |\Pr[\mathsf{Expt}^{\mathtt{ATK}}_{\Gamma,\mathcal{A}}(k) = 1] - 1/2|$ is negligible.

[1] This requirement is not explicitly defined in [16], but is actually necessary for $\mathtt{DCCA}$ security to be meaningful. Without this requirement, $\mathtt{DCCA}$ security is unachievable, as an adversary can submit the challenge ciphertext to the decryption oracle.

For a detectable KEM Γ (with predicate F) and an adversary $\mathcal{A}$, consider the unpredictability experiment $\mathsf{Expt}^{\mathtt{UNP}}_{\Gamma,\mathcal{A}}(k)$ described in Fig. 1 (center-bottom). We say that a detectable KEM Γ is *unpredictable* if for all PPTAs $\mathcal{A}$, $\mathsf{Adv}^{\mathtt{UNP}}_{\Gamma,\mathcal{A}}(k) := \Pr[\mathsf{Expt}^{\mathtt{UNP}}_{\Gamma,\mathcal{A}}(k) = 1]$ is negligible.

2.3 Non-malleable Codes

Here, we recall the definition of *non-malleable codes* [13].

A code $\mathcal{C}$ with message length $\kappa = \kappa(k)$ and codeword length $n = n(k)$ (called also an (n, κ)-code) consists of the two PPTAs (E, D): E is the encoding algorithm that takes 1^k and a message $m \in \{0,1\}^\kappa$ as input, and outputs a codeword $c \in \{0,1\}^n$.; D takes 1^k and c as input, and outputs $m \in \{0,1\}^\kappa$ or the special symbol $\perp$ indicating that c is invalid. We require for all $k \in \mathbb{N}$ and all messages $m \in \{0,1\}^\kappa$, it holds that $\mathsf{D}(1^k, \mathsf{E}(1^k, m)) = m$.

Non-malleability. Non-malleability for codes, formalized by Dziembowski et al. [13], is defined with respect to a class of tampering functions $\mathcal{F}$. Intuitively, non-malleability guarantees that if an encoding c of a message m is modified into $c' = f(c)$ by a function $f \in \mathcal{F}$, then the decoded value m' of c' is either the original message m itself, or a completely unrelated message (or $\perp$). Here we recall the indistinguishability-based definition which is most convenient for us to work with, which is called the "alternative-non-malleability" in [14, Definition A.1]. It was shown in [14] that this definition is equivalent to the original simulation-based definition for codes whose message length κ is superlogarithmic in k.

Let $n, \kappa : \mathbb{N} \to \mathbb{N}$ be positive polynomials of k such that $n(k) \geq \kappa(k)$. For an (n, κ)-code $\mathcal{C} = (\mathsf{E}, \mathsf{D})$, a class of functions $\mathcal{F} = \{\mathcal{F}_k : \{0,1\}^k \to \{0,1\}^k\}_{k \in \mathbb{N}}$, and an adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$, consider the $\mathcal{F}$-$\mathtt{NM}$ experiment $\mathsf{Expt}^{\mathcal{F}\text{-}\mathtt{NM}}_{\mathcal{C},\mathcal{A}}(k)$ described in Fig. 1 (right). In the experiment, "same" is the special symbol indicating that the decoded message m' was either m_0 or m_1, and it is required that $f \in \mathcal{F}_n$ and $|m_0| = |m_1| = \kappa(k)$. We say that $\mathcal{C}$ is *non-malleable with respect to the function class* $\mathcal{F}$ ($\mathcal{F}$-non-malleable, for short) if for all PPTAs[2] $\mathcal{A}$, $\mathsf{Adv}^{\mathcal{F}\text{-}\mathtt{NM}}_{\mathcal{C},\mathcal{A}}(k) := 2 \cdot |\Pr[\mathsf{Expt}^{\mathcal{F}\text{-}\mathtt{NM}}_{\mathcal{C},\mathcal{A}}(k) = 1] - 1/2|$ is negligible. We also say that $\mathcal{C}$ is an $\mathcal{F}$-non-malleable code.

Classes of Tampering Functions. In this paper, we consider the following classes of functions.

Composition of "Bitwise Tampering" and "Bit-Level Permutation" $\mathcal{P}$:
Let $\mathtt{set}, \mathtt{reset}, \mathtt{forward}, \mathtt{toggle} : \{0,1\} \to \{0,1\}$ be the functions over a bit, defined by $\mathtt{set}(x) := 1$, $\mathtt{reset}(x) := 0$, $\mathtt{forward}(x) := x$, and $\mathtt{toggle}(x) := 1 - x$. We define $\mathcal{F}_{\mathtt{BIT}} := \{\mathtt{set}, \mathtt{reset}, \mathtt{forward}, \mathtt{toggle}\}$.
Let $\mathcal{P} = \{\mathcal{P}_n\}_{n \in \mathbb{N}}$ be the class of functions which first perform "bitwise tampering" to an input, followed by a "bit-level permutation." Namely, $\mathcal{P}_n$ is the set of all functions $f : \{0,1\}^n \to \{0,1\}^n$ that can be described by

[2] The original definition [13] considered security against computationally unbounded adversaries. In this paper, however, we only need security against PPTAs.

using n bitwise-tampering functions $f_1, \ldots, f_n \in \mathcal{F}_{\mathtt{BIT}}$ and a permutation $\pi : [n] \to [n]$, as follows:

$$x = (x_1 \| \ldots \| x_n) \overset{f}{\mapsto} \Big(f_{\pi^{-1}(1)}(x_{\pi^{-1}(1)}) \parallel \ \ldots \ \parallel f_{\pi^{-1}(n)}(x_{\pi^{-1}(n)}) \Big).$$

"Bit-Fixing" or "Quoting an Input without Duplicated Positions" $\mathcal{Q}$: Let $\mathtt{one} : \{0,1\}^n \to \{0,1\}$ and $\mathtt{zero} : \{0,1\}^n \to \{0,1\}$ be the constant functions that output 1 and 0 for any n-bit inputs, respectively. Furthermore, for $j \in [n]$, let $\mathtt{quote}^j : \{0,1\}^n \to \{0,1\}$ be the "quoting" function that always outputs the j-th bit of its input.
Let $\mathcal{Q} = \{\mathcal{Q}_n\}_{n\in\mathbb{N}}$ be the class of functions each of whose output bits is either a "fixed value" or "quoting the input without duplicated positions." More formally, $\mathcal{Q}_n$ is the set of all functions $f : \{0,1\}^n \to \{0,1\}^n$ that can be decomposed to n functions $f_1, \ldots, f_n : \{0,1\}^n \to \{0,1\}$ so that $f(x) = (f_1(x) \| \ldots \| f_n(x))$ for all $x \in \{0,1\}^n$, and furthermore it holds that for every $i \in [n]$:

$$f_i \in \{\mathtt{one}, \mathtt{zero}\} \cup \Big(\{\mathtt{quote}^j\}_{j\in[n]} \backslash \{f_j\}_{j\in[i-1]} \Big).$$

Note that the above guarantees that there exist no indices $i, i', j \in [n]$ such that $f_i = f_{i'} = \mathtt{quote}^j$ and $i \neq i'$. We call this condition the *no duplicated quoting condition.*

Agrawal et al. [1] showed the following elegant result, which is crucial for the efficiency of our proposed KEM:

Lemma 1. *[1] There exists an explicit (n, k)-code such that (1) it is $\mathcal{P}$-non-malleable, and (2) its "rate", defined by k/n, asymptotically approaches to 1 as k increases (and hence $n = k + o(k)$).*

Furthermore, the following is implicitly used by Agrawal et al. [2], and also is useful for our purpose. (Although it is almost straightforward from the definitions of $\mathcal{P}$ and $\mathcal{Q}$, we will show its formal proof in the full version.)

Lemma 2. *For all $n \in \mathbb{N}$, $\mathcal{Q}_n \subseteq \mathcal{P}_n$. (This holds even if $\mathcal{F}_{\mathtt{BIT}}$ does not contain* `toggle`*.) Hence, any $\mathcal{P}$-non-malleable code is also $\mathcal{Q}$-non-malleable.*

2.4 Other Standard Primitives

In this paper we also use a pseudorandom generator (PRG) G, and a `CCA` secure deterministic symmetric key encryption (SKE) $E = (\mathsf{SEnc}, \mathsf{SDec})$: For notation, encryption of a plaintext m using a key $K \in \{0,1\}^k$ is denoted by "$c \leftarrow \mathsf{SEnc}(K, m)$" where c is a ciphertext, and decryption of c using K is denoted by "$m \leftarrow \mathsf{SDec}(K, c)$" where m could be the invalid symbol $\perp$. Since their security definitions are standard, we omit them in the proceedings version.

3 New Security Notions for Detectable PKE and KEM

In this section, we introduce new security notions for detectable PKE: wNM-DCCA *security* and wRNM-DCCA *security* in Sect. 3.1, and *randomness-inextractability* in Sect. 3.2. We also show some useful facts regarding the new security notions in Sect. 3.3.

We also define wNM-DCCA security and randomness-inextractability for detectable KEMs. Since their definitions are straightforward KEM-analogues of those for detectable PKE in this section, we omit them here and formally provide them in the full version.

3.1 "Weak" Non-malleability Under DCCA and Its "Replayable" Variant

Here, we define a "weak" form of non-malleability against DCCA for detectable PKE, which we call wNM-DCCA *security*, that captures the intuition that a DCCA adversary who works in the DCCA experiment cannot come up with a ciphertext that is "meaningfully related" to the challenge ciphertext. Recall that the original definitions of non-malleability for PKE [3,12,22] ensure that an adversary cannot come up with even a vector of ciphertexts that are "meaningfully related" to the challenge ciphertext, while our notion here only requires that it cannot come up with only a single related ciphertext. Technically, following the formalizations in [3,20,22], we formalize wNM-DCCA security by modifying the original DCCA experiment (in which originally the usage of the decryption oracle is restricted according the predicate F of detectable PKE), so that at the end of the experiment an adversary is allowed to make a single "unrestricted" decryption query, regardless of F. Thus, it is like "1-bounded" CCA security [7], albeit an adversary has additionally access to DCCA decryption oracle. Myers and Shelat [20] defined a security notion for PKE-to-PKE constructions called "q-wise-non-malleability under UCCA." Our definition of wNM-DCCA security is a detectable-PKE-analogue of their 1-wise-non-malleability.

We also define a weaker variant of wNM-DCCA security, in the security experiment of which the final "unrestricted" decryption query is answered like a decryption query in the "replayable" CCA experiment [4], namely, if the decryption result is one of the challenge plaintexts that an adversary uses, then the adversary is only informed so and is not given the actual decryption result. Due to the lack of a better name, we call it wRNM-DCCA security (where R stands for "**R**eplayable").

Fomally, for a detectable PKE scheme $\Pi = (\mathsf{PKG}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{F})$ and an adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3)$, we define the wNM-DCCA experiment $\mathsf{Expt}^{\text{wNM-DCCA}}_{\Pi,\mathcal{A}}(k)$ and the wRNM-DCCA experiment $\mathsf{Expt}^{\text{wRNM-DCCA}}_{\Pi,\mathcal{A}}(k)$ described in Fig. 2 (left and center, respectively). In both of the experiments, it is required that $|m_0| = |m_1|$, and as in the DCCA experiment, $\mathcal{A}_2$ is not allowed to submit a decryption query c satisfying $\mathsf{F}(pk, c^*, c) = 1$ to the decryption oracle. The adversary's final "unrestricted" decryption query is captured by the ciphertext c' that is finally output by $\mathcal{A}_2$, and naturally it is required that $c' \neq c^*$. *However, we allow c' to be such*

that $\mathsf{F}(pk, c^*, c') = 1$. In the wRNM-DCCA experiment, "same" is the special symbol (which is distinguished from $\perp$) that indicates that $\mathsf{Dec}(sk, c') \in \{m_0, m_1\}$.

Definition 1. *We say that a detectable PKE scheme Π is* wNM-DCCA *secure if for all PPTAs $\mathcal{A}$,* $\mathsf{Adv}^{\text{wNM-DCCA}}_{\Pi,\mathcal{A}}(k) := 2 \cdot |\Pr[\mathsf{Expt}^{\text{wNM-DCCA}}_{\Pi,\mathcal{A}}(k) = 1] - 1/2|$ *is negligible. We define* wRNM-DCCA *security analogously.*

3.2 Randomness-Inextractability

Here we introduce another security notion for detectable PKE that we call *randomness-inextractability*. Roughly, this security notion ensures that given the challenge ciphertext c^* (which is an encryption of a plaintext of an adversary's choice), an adversary cannot come up with a pair (m', r') of a plaintext and a randomness such that $\mathsf{F}(pk, c^*, \mathsf{Enc}(pk, m'; r')) = 1$. If the predicate $\mathsf{F}(pk, c^*, c')$ tests the equality $(c^* \stackrel{?}{=} c')$, then this notion exactly demands that the randomness used in c^* cannot be recovered, and hence we use the name "randomness-inextractability" (although we allow more general predicates for F).

Formally, for a detectable PKE scheme $\Pi = (\mathsf{PKG}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{F})$ and an adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$, consider the R-Inext experiment described in Fig. 2 (right).

Definition 2. *We say that a detectable PKE scheme Π satisfies* randomness-inextractability *if for all PPTAs $\mathcal{A}$,* $\mathsf{Adv}^{\text{R-Inext}}_{\Pi,\mathcal{A}}(k) := \Pr[\mathsf{Expt}^{\text{R-Inext}}_{\Pi,\mathcal{A}}(k) = 1]$ *is negligible.*

Remark. We could have defined the randomness-inextractability experiment so that we let an adversary choose its challenge message m after given a public key pk. This makes the security stronger. However, we do not need this stronger variant for our results.

3.3 Useful Facts

Stretching a Session-Key. As in the case of ordinary KEMs, for a detectable KEM, session-keys can be stretched by using a PRG. More formally, let $\Gamma = (\mathsf{KKG}, \mathsf{Encap}, \mathsf{Decap}, \mathsf{F})$ be a detectable KEM whose session-key space is $\{0,1\}^k$. Let $\mathsf{G} : \{0,1\}^k \to \{0,1\}^\ell$ be a PRG with $\ell = \ell(k) > k$, where for convenience we define $\mathsf{G}(\perp) := \perp$. Then, consider the detectable KEM $\Gamma' = (\mathsf{KKG}, \mathsf{Encap}', \mathsf{Decap}', \mathsf{F})$ whose session-key space is $\{0,1\}^\ell$, which is naturally constructed by combining Γ and G: $\mathsf{Encap}'(pk)$ runs $(c, K) \leftarrow \mathsf{Encap}(pk)$ and outputs a ciphertext/session key pair $(c, \mathsf{G}(K))$.; We define $\mathsf{Decap}'(sk, c) := \mathsf{G}(\mathsf{Decap}(sk, c))$. The following is straightforward, and thus its proof is omitted.

Lemma 3. *If the detectable KEM Γ satisfies randomness-inextractability (resp. unpredictability), then so does the detectable KEM Γ'. Furthermore, if Γ is* DCCA *(resp.* wNM-DCCA*) secure and G is a PRG, then Γ' is* DCCA *(resp.* wNM-DCCA*) secure.*

$\mathsf{Expt}^{\mathtt{wNM\text{-}DCCA}}_{\Pi,\mathcal{A}}(k)$:
$(pk, sk) \leftarrow \mathsf{PKG}(1^k)$
$(m_0, m_1, \mathsf{st}) \leftarrow \mathcal{A}_1^{\mathcal{O}(\cdot)}(pk)$
$b \leftarrow \{0,1\}$
$c^* \leftarrow \mathsf{Enc}(pk, m_b)$
$(c', \mathsf{st}') \leftarrow \mathcal{A}_2^{\mathcal{O}(\cdot)}(\mathsf{st}, c^*)$
$m' \leftarrow \mathsf{Dec}(sk, c')$
$b' \leftarrow \mathcal{A}_3(\mathsf{st}', m')$
Return $(b' \stackrel{?}{=} b)$.

$\mathsf{Expt}^{\mathtt{wRNM\text{-}DCCA}}_{\Pi,\mathcal{A}}(k)$:
$(pk, sk) \leftarrow \mathsf{PKG}(1^k)$
$(m_0, m_1, \mathsf{st}) \leftarrow \mathcal{A}_1^{\mathcal{O}(\cdot)}(pk)$
$b \leftarrow \{0,1\}$
$c^* \leftarrow \mathsf{Enc}(pk, m_b)$
$(c', \mathsf{st}') \leftarrow \mathcal{A}_2^{\mathcal{O}(\cdot)}(\mathsf{st}, c^*)$
$m' \leftarrow \mathsf{Dec}(sk, c')$
If $m' \in \{m_0, m_1\}$ then $m' \leftarrow \mathsf{same}$
$b' \leftarrow \mathcal{A}_3(\mathsf{st}', m')$
Return $(b' \stackrel{?}{=} b)$.

$\mathsf{Expt}^{\mathtt{R\text{-}Inext}}_{\Pi,\mathcal{A}}(k)$:
$(m, \mathsf{st}) \leftarrow \mathcal{A}_1(1^k)$
$(pk, sk) \leftarrow \mathsf{PKG}(1^k)$
$c^* \leftarrow \mathsf{Enc}(pk, m)$
$(m', r') \leftarrow \mathcal{A}_2^{\mathcal{O}(\cdot)}(\mathsf{st}, pk, c^*)$
$c' \leftarrow \mathsf{Enc}(pk, m'; r')$
Return $\mathsf{F}(pk, c^*, c')$.

Fig. 2. Security experiments for wNM-DCCA security (left), wRNM-DCCA security (center), and randomness-inextractability (right). In the experiments, $\mathcal{O}(\cdot)$ is the decryption oracle $\mathsf{Dec}(sk, \cdot)$, and in the wNM/wRNM-CCA experiments, the decryption oracle for $\mathcal{A}_2$ has the same restriction as in the DCCA experiment.

Hybrid Encryption. For a detectable PKE scheme, a straightforward application of hybrid encryption preserves w(R)NM-DCCA security and randomness-inextractability, when combined with a CCA secure SKE scheme. Since a CCA secure SKE scheme with "zero" ciphertext overhead can be realized from a strong pseudorandom permutation [23] (which is in turn realized based on any one-way function), the ciphertext overhead of a detectable PKE scheme with w(R)NM-DCCA security and randomness-inextractability, can be as small as the ciphertext size of the scheme for encrypting a random session-key (usually a k-bit string).

Formally, let $\Pi = (\mathsf{PKG}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{F})$ be a detectable PKE scheme where the randomness space of Enc is $\{0,1\}^\ell$, and let $E = (\mathsf{SEnc}, \mathsf{SDec})$ be a deterministic SKE scheme (i.e. its encryption algorithm SEnc is deterministic). Then, we naturally construct the detectable PKE scheme $\Pi_{\mathtt{HYB}} = (\mathsf{PKG}_{\mathtt{HYB}}, \mathsf{Enc}_{\mathtt{HYB}}, \mathsf{Dec}_{\mathtt{HYB}}, \mathsf{F}_{\mathtt{HYB}})$ via hybrid encryption, as in Fig. 3. (We describe the randomness of $\mathsf{Enc}_{\mathtt{HYB}}$ explicitly so that it is convenient to consider its randomness-inextractability.) The randomness space of $\mathsf{Enc}_{\mathtt{HYB}}$ is $\{0,1\}^{\ell+k}$.

$\mathsf{PKG}_{\mathtt{HYB}}(1^k)$:	$\mathsf{Enc}_{\mathtt{HYB}}(pk, m; R)$:	$\mathsf{Dec}_{\mathtt{HYB}}(sk, C)$:	$\mathsf{F}_{\mathtt{HYB}}(pk, C^*, C')$:
$(pk, sk) \leftarrow \mathsf{PKG}(1^k)$	Parse R as (r, K) $\in \{0,1\}^\ell \times \{0,1\}^k$.	$(c, \widehat{c}) \leftarrow C$	$(c^*, \widehat{c}^*) \leftarrow C^*$
Return (pk, sk).	$c \leftarrow \mathsf{Enc}(pk, K; r)$	$K \leftarrow \mathsf{Dec}(sk, c)$	$(c', \widehat{c}') \leftarrow C'$
	$\widehat{c} \leftarrow \mathsf{SEnc}(K, m)$	If $K = \perp$ then return $\perp$.	$b \leftarrow \mathsf{F}(pk, c^*, c')$
	$C \leftarrow (c, \widehat{c})$	$m \leftarrow \mathsf{SDec}(K, \widehat{c})$	Return b.
	Return C.	Return m.	

Fig. 3. Hybrid encryption $\Pi_{\mathtt{HYB}}$ for detectable PKE.

Regarding the security of the hybrid encryption construction, the following lemma is straightforward to see.

Lemma 4. *If the detectable PKE scheme Π is* wNM-DCCA *secure (resp.* wRNM-DCCA *secure) and the SKE scheme E is* CCA *secure, then the detectable PKE scheme $\Pi_{\mathtt{HYB}}$ in Fig. 3 is* wNM-DCCA *secure (resp.* wRNM-DCCA *secure). Furthermore, if Π satisfies randomness-inextractability (resp. unpredictability), then so does $\Pi_{\mathtt{HYB}}$.*

From wRNM-DCCA *Security to* wNM-DCCA *Security.* Canetti, Krawczyk, and Nielsen [4] showed how to convert a "replayable" CCA secure PKE scheme into an ordinary CCA secure KEM, using a message authentication code (MAC), with almost no overhead. This method can be used for converting a wRNM-DCCA secure detectable PKE scheme into a wNM-DCCA secure detectable KEM. We review this transformation in the full version.

On the Non-triviality of Randomness-Inextractability. One might wonder whether there is an implication from randomness-inextractability to unpredictability and/or vice versa (especially in case if a detectable PKE scheme already satisfies wNM-DCCA security). We show that this is not the case, for both directions. Specifically, (via artificial counterexamples) we can show the following lemma that shows the non-triviality of these notions, which we formally show in the full version.

Lemma 5. *A detectable PKE scheme satisfying* wNM-DCCA *security and unpredictability simultaneously does not necessarily satisfy randomness-inextractability. Furthermore, a detectable PKE scheme satisfying* wNM-DCCA *security and randomness-inextractability simultaneously does not necessarily satisfy unpredictability.*

4 Chosen Ciphertext Security of the Double-Layered Construction

In this section, we show our main result: two new CCA security proofs for the "double-layered" construction $\Gamma_{\mathtt{DL}}$ (of a KEM) constructed from the "inner" detectable KEM $\Gamma_{\mathtt{in}}$ and the "outer" detectable PKE scheme $\Pi_{\mathtt{out}}$. We also show a partial evidence that we need to rely on "non-malleability" that we defined in the previous section.

The Double-Layered Construction. Let $\Pi_{\mathtt{out}} = (\mathsf{PKG}_{\mathtt{out}}, \mathsf{Enc}_{\mathtt{out}}, \mathsf{Dec}_{\mathtt{out}}, \mathsf{F}_{\mathtt{out}})$ be a detectable PKE scheme. We assume the plaintext space of $\Pi_{\mathtt{out}}$ to be $\{0,1\}^n$ (where $n = n(k)$ is determined below), and the randomness space of $\mathsf{Enc}_{\mathtt{out}}$ to be $\{0,1\}^\ell$ for some positive polynomial $\ell = \ell(k)$. Let $\Gamma_{\mathtt{in}} = (\mathsf{KKG}_{\mathtt{in}}, \mathsf{Encap}_{\mathtt{in}}, \mathsf{Decap}_{\mathtt{in}}, \mathsf{F}_{\mathtt{in}})$ be a detectable KEM such that the ciphertext length is n bit, and the session-key space is $\{0,1\}^{\ell+k}$. Then we construct the "double-layered" KEM $\Gamma_{\mathtt{DL}} = (\mathsf{KKG}_{\mathtt{DL}}, \mathsf{Encap}_{\mathtt{DL}}, \mathsf{Decap}_{\mathtt{DL}})$ as in Fig. 4. For convenience, we occasionally call $\Gamma_{\mathtt{in}}$ the *inner* KEM and $\Pi_{\mathtt{out}}$ the *outer* PKE scheme.

Our First Security Proof. The CCA security of $\Gamma_{\mathtt{DL}}$ can be shown as follows.

$\mathsf{KKG}_{\mathtt{DL}}(1^k)$:
$(pk_{\mathtt{in}}, sk_{\mathtt{in}}) \leftarrow \mathsf{KKG}_{\mathtt{in}}(1^k)$
$(pk_{\mathtt{out}}, sk_{\mathtt{out}}) \leftarrow \mathsf{PKG}_{\mathtt{out}}(1^k)$
$PK \leftarrow (pk_{\mathtt{in}}, pk_{\mathtt{out}})$
$SK \leftarrow (sk_{\mathtt{in}}, sk_{\mathtt{out}}, PK)$
Return (PK, SK).

$\mathsf{Encap}_{\mathtt{DL}}(PK)$:
$(pk_{\mathtt{in}}, pk_{\mathtt{out}}) \leftarrow PK$
$(c_{\mathtt{in}}, \alpha) \leftarrow \mathsf{Encap}_{\mathtt{in}}(pk_{\mathtt{in}})$
Parse α as $(r, K) \in \{0,1\}^{\ell} \times \{0,1\}^k$.
$c \leftarrow \mathsf{Enc}_{\mathtt{out}}(pk_{\mathtt{out}}, c_{\mathtt{in}}; r)$
Return (c, K).

$\mathsf{Decap}_{\mathtt{DL}}(SK, c)$:
$(sk_{\mathtt{in}}, sk_{\mathtt{out}}, PK) \leftarrow SK$
$(pk_{\mathtt{in}}, pk_{\mathtt{out}}) \leftarrow PK$
$c_{\mathtt{in}} \leftarrow \mathsf{Dec}_{\mathtt{out}}(sk_{\mathtt{out}}, c)$
If $c_{\mathtt{in}} = \perp$ then return $\perp$.
$\alpha \leftarrow \mathsf{Decap}_{\mathtt{in}}(sk_{\mathtt{in}}, c_{\mathtt{in}})$
If $\alpha = \perp$ then return $\perp$.
Parse α as $(r, K) \in \{0,1\}^{\ell} \times \{0,1\}^k$.
If $\mathsf{Enc}_{\mathtt{out}}(pk_{\mathtt{out}}, c_{\mathtt{in}}; r) = c$
then return K else return $\perp$.

Fig. 4. The double-layered KEM construction $\Gamma_{\mathtt{DL}}$ from a detectable PKE scheme $\Pi_{\mathtt{out}}$ and a detectable KEM $\Gamma_{\mathtt{in}}$.

Theorem 1. *Assume that the "outer" PKE scheme $\Pi_{\mathtt{out}}$ is a detectable PKE scheme satisfying* wRNM-DCCA *security and randomness-inextractability, and the "inner" KEM $\Gamma_{\mathtt{in}}$ is a detectable KEM satisfying* DCCA *security and unpredictability. Then, the KEM $\Gamma_{\mathtt{DL}}$ in Fig. 4 is* CCA *secure.*

The structure of the proof is similar to the security proofs for the constructions by Hohenberger et al. [16] and by Matsuda and Hanaoka [19]. However, the details differ due to the difference in the construction and the used assumptions.

We explain the ideas for the proof of Theorem 1. (Here, the values with asterisk (*) represent those related to the challenge ciphertext c^*.) As the first step, note that since a session-key K of $\Gamma_{\mathtt{DL}}$ is part of a session-key $\alpha = (r \| K)$ of the DCCA secure inner KEM $\Gamma_{\mathtt{in}}$, unless a CCA adversary $\mathcal{A}$ submits a decapsulation query c that simultaneously satisfies (1) $\mathsf{Dec}_{\mathtt{out}}(sk_{\mathtt{out}}, c) = c_{\mathtt{in}} \neq \perp$ and (2) $\mathsf{F}_{\mathtt{in}}(pk_{\mathtt{in}}, c^*_{\mathtt{in}}, c_{\mathtt{in}}) = 1$, $\mathcal{A}$ has no chance in distinguishing the real session-key K^*_1 from a random K^*_0. Following [16,19], we call this type of decapsulation query a *dangerous* query. If the probability that $\mathcal{A}$ comes up with a dangerous query is negligible, then we can finish the proof. Furthermore, observe that since $\Gamma_{\mathtt{in}}$ satisfies unpredictability, if we can ensure that the information of the inner ciphertext $c^*_{\mathtt{in}}$ is hidden from $\mathcal{A}$'s view, then the probability that $\mathcal{A}$ comes up with a dangerous query is negligible.

To show that the probability that $\mathcal{A}$ comes up with a dangerous query in the original security game is negligibly close to that in the security game in which $\mathcal{A}$'s view does not contain $c^*_{\mathtt{in}}$ at all (and hence we can invoke the unpredictability of $\Gamma_{\mathtt{in}}$), we rely on the security properties of the outer PKE scheme $\Pi_{\mathtt{out}}$ to gradually change the security game for $\mathcal{A}$ so that in the final game, c^* as well as other values in $\mathcal{A}$'s view contain no information on $c^*_{\mathtt{in}}$. Note that in the actual encapsulation algorithm $\mathsf{Encap}_{\mathtt{DL}}$, the randomness r used for outer encryption is also a part of the session-key α of the inner KEM. Thus, once we invoke the DCCA security of the inner KEM $\Gamma_{\mathtt{in}}$ (which we have already done as the first step), not only the real session-key K^*_1 but also the randomness r^* used to generate

the challenge ciphertext c^* are made uniformly random values, which enables us to rely on the security properties of $\Pi_{\mathtt{out}}$ from that point on.

Now, intuitively, the DCCA security (which is implied by wRNM-DCCA security) of $\Pi_{\mathtt{out}}$ guarantees that $c^*_{\mathtt{in}}$ is hidden from $\mathcal{A}$'s view as long as $\mathcal{A}$ only submits a decapsulation query c such that $\mathsf{F}_{\mathtt{out}}(pk_{\mathtt{out}}, c^*, c) = 0$. However, $\mathcal{A}$ is free to choose its own decapsulation query, and may submit c such that $\mathsf{F}_{\mathtt{out}}(pk_{\mathtt{out}}, c^*, c) = 1$. As mentioned in Sect. 1.2, this is another type of "dangerous" query, in the sense that the condition $\mathsf{F}_{\mathtt{out}}(pk_{\mathtt{out}}, c^*, c) = 1$ prevents us from relying on the DCCA security of the outer PKE scheme $\Pi_{\mathtt{out}}$. To distinguish this from the above mentioned type of dangerous queries with respect to the inner KEM, let us use the names "*inner-dangerous* queries" and "*outer-dangerous* queries" which are associated with the inner KEM and the outer PKE scheme, respectively.

In the full proof, we will show that the randomness-inextractability of the outer PKE scheme allows us to reject decapsulation queries c satisfying $\mathsf{F}_{\mathtt{out}}(pk_{\mathtt{out}}, c^*, c) = 1$, without being noticed by $\mathcal{A}$. Intuitively, this is possible because in order for $\mathcal{A}$ to notice the difference between a security game in which a decryption query c with $\mathsf{F}_{\mathtt{out}}(pk_{\mathtt{out}}, c^*, c) = 1$ is not rejected and a security game in which such c is rejected, $\mathcal{A}$ has to come up with a "valid" query c satisfying $\mathsf{F}_{\mathtt{out}}(pk_{\mathtt{out}}, c^*, c) = 1$ and $\mathsf{Decap}_{\mathtt{DL}}(SK, c) \neq \bot$. However, the latter condition implies $\mathsf{Dec}_{\mathtt{out}}(sk_{\mathtt{out}}, c) = c_{\mathtt{in}} \neq \bot$, $\mathsf{Decap}_{\mathtt{in}}(sk_{\mathtt{in}}, c_{\mathtt{in}}) = (r \| K) \neq \bot$, and $\mathsf{Enc}_{\mathtt{out}}(pk_{\mathtt{out}}, c_{\mathtt{in}}; r) = c$, among which the combination of $\mathsf{F}_{\mathtt{out}}(pk_{\mathtt{out}}, c^*, c) = 1$ and $\mathsf{Enc}_{\mathtt{out}}(pk_{\mathtt{out}}, c_{\mathtt{in}}; r) = c$ is exactly the condition of violating randomness-inextractability, and thus such a valid query c must be hard to find.

If we can safely reject an outer-dangerous query, one might wonder why we need non-malleability for the outer PKE scheme, and why ordinary DCCA security is not sufficient. The reason is that although DCCA security of $\Pi_{\mathtt{out}}$ intuitively ensures that $\mathcal{A}$ cannot "see" the inner challenge ciphertext $c^*_{\mathtt{in}}$, it does not prevent $\mathcal{A}$ from coming up with an inner-dangerous decapsulation query c such that $\mathsf{F}_{\mathtt{out}}(pk_{\mathtt{out}}, c^*, c) = 1$. From the viewpoint of the security proof, we may be able to come up with a DCCA adversary (a reduction algorithm) for $\Pi_{\mathtt{out}}$ that perfectly simulates the security game (in which queries c with $\mathsf{F}_{\mathtt{out}}(pk_{\mathtt{out}}, c^*, c) = 1$ are rejected) for $\mathcal{A}$. However, such DCCA adversary cannot check if $\mathcal{A}$'s query satisfying $\mathsf{F}_{\mathtt{out}}(pk_{\mathtt{out}}, c^*, c) = 1$ is an inner-dangerous query due to the restriction on the decryption oracle.

This is the place where the non-malleability of the outer PKE scheme comes into play. Note that an inner ciphertext is a "plaintext" of the outer PKE scheme, and the notion of "inner-dangerous queries" is a "meaningful relation" between $c^*_{\mathtt{in}}$ and another inner ciphertext. Therefore, the wRNM-DCCA security of $\Pi_{\mathtt{out}}$ ensures that $\mathcal{A}$ cannot come up with even a single inner-dangerous query c, as long as $\mathcal{A}$ can only observe the decapsulation results of queries c' satisfying $\mathsf{F}_{\mathtt{out}}(pk_{\mathtt{out}}, c^*, c') = 0$. From the viewpoint of the security proof, if a reduction algorithm is a wRNM-DCCA adversary for $\Pi_{\mathtt{out}}$, it can check if $\mathcal{A}$'s query c is inner-dangerous by its final "unrestricted" decryption query, even if $\mathsf{F}_{\mathtt{out}}(pk_{\mathtt{out}}, c^*, c) = 1$ holds. This enables us to finally show that the probability that $\mathcal{A}$ comes up

with an inner-dangerous query in the original security game, is negligibly close to the probability that $\mathcal{A}$ does so in the game in which $\mathcal{A}$'s view does not contain the information on $c^*_{\mathtt{in}}$.

Hence, combining all the security properties of the building blocks leads to CCA security. However, the explanation so far hides some technical subtleties that arise due to the "replayable-CCA"-like nature of wRNM-DCCA security, and the treatment of the cases where $\mathcal{A}$'s decapsulation query c satisfies $\mathsf{Dec}_{\mathtt{out}}(sk_{\mathtt{out}}, c) = c^*_{\mathtt{in}}$, etc. For the details, see the proof in the full version.

Our Second Security Proof. We show an alternative security proof for the double-layered construction based on slightly different assumptions on the building blocks.

Theorem 2. *Assume that the "outer" PKE scheme $\Pi_{\mathtt{out}}$ is a detectable PKE scheme satisfying* DCCA *security and randomness-inextractability, and the "inner" KEM $\Gamma_{\mathtt{in}}$ is a detectable KEM satisfying* wNM-DCCA *security and unpredictability. Then, the KEM $\Gamma_{\mathtt{DL}}$ in Fig. 4 is* CCA *secure.*

Recall that Myers and Shelat's original double-layered construction uses an "unquoted" CCA (UCCA) secure construction of a PKE scheme for the outer PKE scheme and a construction of a KEM which is "1-wise-non-malleable under UCCA" for the inner KEM, where UCCA security and its non-malleable variant are security notions considered for PKE-to-PKE constructions (i.e. constructions that use another PKE scheme as a building block). Recall also that DCCA security is an abstraction of UCCA security [16], from a security notion for a PKE-to-PKE construction to that of a wider notion of detectable PKE. Analogously, our definition of wNM-DCCA security can be seen as an abstraction of Myers and Shelat's "1-wise non-malleability under UCCA". Furthermore, we can easily see that the actual instantiations of the inner KEM and the outer PKE scheme used in the original Myers-Shelat construction [20], when respectively seen as a detectable KEM and a detectable PKE scheme, satisfy unpredictability and randomness-inextractability. Therefore, Theorem 2 can be seen as a generalization of Myers and Shelat's result.

The structure of the proof of Theorem 2 is similar to our first proof. However, there are several subtle but crucial differences. In particular, the definitions of "inner/outer-dangerous queries" are different from those used in the proof of Theorem 1, and correspondingly we consider a different ordering of the sequence of games for this proof. Furthermore, the role of the "non-malleability" in this proof and that of the proof of Theorem 1 are different. Informally speaking, in this proof, the wNM-DCCA security of the inner detectable KEM $\Gamma_{\mathtt{in}}$ is used to ensure that the probability that a CCA adversary comes up with an outer-dangerous query is not noticeably different between the games in which we invoke (the indistinguishability property of) the DCCA security of the inner KEM.

Can We Avoid w(R)NM-DCCA *Security?* Both of our security proofs for the CCA security of the double-layered construction require either the inner detectable KEM or the outer detectable PKE scheme to be "non-malleable" under DCCA.

Looking ahead, in the next section, we will see that the simplest "bitwise-encrypt" construction based on CCA secure 1-bit PKE satisfies DCCA security, unpredictability, and randomness-inextractability. Thus, a natural question would be whether we can prove the CCA security of the double-layered construction without using the non-malleability notions for both of the building blocks (and instead only requiring DCCA security). If such a security proof were possible, then one can use the bitwise-encrypt-based construction both for the inner KEM and the outer PKE scheme, and the resulting CCA secure KEM would be fairly simple.

Unfortunately, however, we show that such a security proof for the double-layered construction is impossible, as there is a counterexample.

Theorem 3. *Assume there exists a detectable PKE scheme which is* DCCA *secure and unpredictable. Then, there exist a detectable KEM* $\Gamma_{\mathtt{in}}$ *and a detectable PKE scheme* $\Pi_{\mathtt{out}}$ *such that the following simultaneously hold: (1)* $\Gamma_{\mathtt{in}}$ *is* DCCA *secure and unpredictable. (2)* $\Pi_{\mathtt{out}}$ *is* DCCA *secure and randomness-inextractable. (3) The double-layered KEM* $\Gamma_{\mathtt{DL}}$ *constructed using* $\Gamma_{\mathtt{in}}$ *as the inner KEM and* $\Pi_{\mathtt{out}}$ *as the outer PKE scheme, is not* CCA *secure (in fact, not secure in the sense of one-wayness under* 1*-bounded* CCA*).*

Our counterexample is based on an observation that the combination of DCCA security, unpredictability, and randomness-inextractability, does not rule out a double-layered KEM with the following property: A ciphertext C is of the form $C = (c_1, c_2)$ and the corresponding session-key K is of the form $K = (K_1, K_2)$, and furthermore it is "blockwise" consistent, meaning that each pair (c_i, K_i) is individually consistent as a ciphertext/session-key pair of the double-layered construction. Thus, the decapsulation result of the "swapped" ciphertext $\widehat{C} = (c_2, c_1)$ is the "swapped" session-key $\widehat{K} = (K_2, K_1)$. Such a KEM is clearly malleable, and its one-wayness is broken by just a single decapsulation query.

5 Concrete Instantiations of Building Blocks

In this section, we show how to construct a detectable PKE scheme, which we call "encode-then-bitwise-encrypt" (EtBE) construction, that uses a CCA secure 1-bit PKE scheme and a $\mathcal{Q}$-non-malleable code as building blocks and simultaneously satisfies wRNM-DCCA security and randomness-inextractability. Since it is much easier to understand it if we first review the simple "bitwise-encrypt" construction, we first review it in Sect. 5.1 together with its security properties, and then we show the EtBE construction in Sect. 5.2.

5.1 Bitwise-Encrypt Construction

Here, we show that the "bitwise-encrypt" construction of a detectable PKE scheme based on a 1-bit PKE scheme, in which each bit of a plaintext is encrypted in a bit-by-bit fashion by the underlying 1-bit scheme, can be shown to satisfy

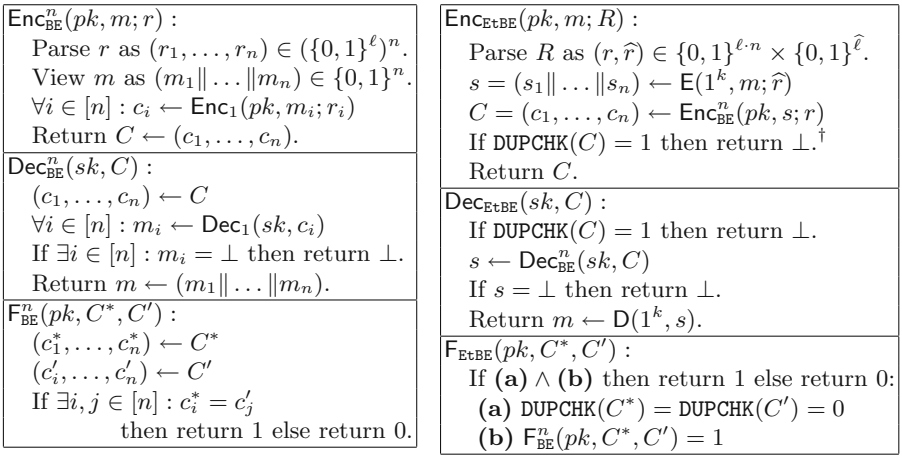

Fig. 5. The "bitwise-encrypt" (n-bit) construction $\Pi^n_{\mathtt{BE}}$ (left), and the "encode-then-bitwise-encrypt" (EtBE) construction $\Pi_{\mathtt{EtBE}}$ (right), both based on a 1-bit PKE scheme Π_1.The key generation algorithms for $\Pi^n_{\mathtt{BE}}$ and $\Pi_{\mathtt{EtBE}}$ are the key generation algorithm PKG_1 of the underlying scheme Π_1.† Regarding the case in which $\mathsf{Enc}_{\mathtt{EtBE}}$ returns $\perp$, see the explanation in the text.

randomness-inextractability, DCCA security, and unpredictability, if the underlying 1-bit PKE scheme is CCA secure.

Let $\Pi_1 = (\mathsf{PKG}_1, \mathsf{Enc}_1, \mathsf{Dec}_1)$ be a 1-bit PKE scheme, and the randomness space of whose encryption algorithm Enc_1 is $\{0,1\}^\ell$ (where $\ell = \ell(k)$ is some positive polynomial). Then, for a polynomial $n = n(k) > 0$, consider the "bitwise-encrypt" construction $\Pi^n_{\mathtt{BE}} = (\mathsf{PKG}^n_{\mathtt{BE}} := \mathsf{PKG}_1, \mathsf{Enc}^n_{\mathtt{BE}}, \mathsf{Dec}^n_{\mathtt{BE}}, \mathsf{F}^n_{\mathtt{BE}})$ of an n-bit detectable PKE scheme described in Fig. 5 (left). The key generation algorithm $\mathsf{PKG}^n_{\mathtt{BE}}$ is actually PKG_1 itself, and we do not show it in the figure. The randomness space of $\mathsf{Enc}_{\mathtt{BE}}$ is $\{0,1\}^{\ell \cdot n}$. In the figure, we make the randomness used by $\mathsf{Enc}^n_{\mathtt{BE}}$ explicit so that it is convenient to consider randomness-inextractability.

The following result was shown by Hohenberger et al. [16]:

Lemma 6. *[16] Let $n = n(k) > 0$ be a polynomial. If the 1-bit PKE scheme Π_1 is CCA secure, then the detectable PKE scheme $\Pi^n_{\mathtt{BE}}$ scheme satisfies DCCA security and unpredictability.*

We show a similar statement regarding randomness-inextractability.

Lemma 7. *Let $n = n(k) > 0$ be a polynomial. If the PKE scheme Π_1 is CCA secure, then the detectable PKE scheme $\Pi^n_{\mathtt{BE}}$ satisfies randomness-inextractability.*

Here we explain an intuition why Lemma 7 is true, which is quite straightforward: Suppose an adversary $\mathcal{A}$, given a public key pk and the challenge ciphertext $C^* = (c^*_1, \ldots, c^*_n)$ and access to the decryption oracle, succeeds in outputting a plaintext $m' = (m'_1 \| \ldots \| m'_n)$ and a randomness $r' = (r'_1, \ldots, r'_n)$ such that

$\mathsf{F}^n_{\mathtt{BE}}(pk, C^*, C') = 1$ with $C' = (c'_1, \ldots, c'_n) = \mathsf{Enc}^n_{\mathtt{BE}}(pk, m'; r')$. Then, by definition, there must be a position $i \in [n]$ such that $c^*_i = c'_j$ holds for some $j \in [n]$, where $c'_a = \mathsf{Enc}_1(pk, m'_a; r'_a)$ for each $a \in [n]$. Note that such $\mathcal{A}$ is in fact "extracting" the randomness used for generating c^*_i. Note also that extracting a randomness used for generating a ciphertext is a harder task than breaking indistinguishability. Thus, it is easy to construct another CCA adversary (a reduction algorithm) $\mathcal{B}$ for Π_1 that initially guesses the position i such that $c^*_i = c'_j$ holds with some j, embeds $\mathcal{B}$'s challenge ciphertext into the i-th position of the challenge ciphertext for $\mathcal{A}$, and has the CCA advantage at least $1/n$ times that of $\mathcal{A}$'s advantage in breaking randomness-inextractability.

5.2 Encode-then-Bitwise-Encrypt Construction

Here, we show the construction of detectable PKE that we call *"Encode-then-Bitwise-Encrypt"* (EtBE) construction, which simultaneously achieves wRNM-DCCA security and randomness-inextractability, based on the security properties of the bitwise-encrypt construction (which are in turn based on the underlying CCA secure 1-bit scheme) and a $\mathcal{Q}$-non-malleable code. Our construction is actually a direct "PKE"-analogue of the transformation of a CCA secure 1-bit commitment scheme into a non-malleable string commitment scheme by Agrawal et al. [2]. We adapt their construction into the (detectable) PKE setting.

Let $\mathcal{C} = (\mathsf{E}, \mathsf{D})$ be a code with message length k and codeword length $n = n(k) \geq k$. Let $\Pi_1 = (\mathsf{PKG}_1, \mathsf{Enc}_1, \mathsf{Dec}_1)$ be a 1-bit PKE scheme. Let $\Pi^n_{\mathtt{BE}} = (\mathsf{PKG}^n_{\mathtt{BE}} = \mathsf{PKG}_1, \mathsf{Enc}^n_{\mathtt{BE}}, \mathsf{Dec}^n_{\mathtt{BE}}, \mathsf{F}^n_{\mathtt{BE}})$ be the bitwise-encrypt construction based on Π_1. For convenience, we introduce the procedure "$\mathtt{DUPCHK}(\cdot)$" which takes a ciphertext $C = (c_1, \ldots, c_n)$ of $\Pi^n_{\mathtt{BE}}$ as input, and returns 1 if there exist distinct $i, j \in [n]$ such that $c_i = c_j$, and returns 0 otherwise. (That is, $\mathtt{DUPCHK}(C)$ checks a duplication in the component ciphertexts $(c_i)_{i \in [n]}$.)

Using $\mathcal{C}$, $\Pi^n_{\mathtt{BE}}$ (and Π_1), and DUPCHK, the EtBE construction $\Pi_{\mathtt{EtBE}} = (\mathsf{PKG}_{\mathtt{EtBE}} := \mathsf{PKG}_1, \mathsf{Enc}_{\mathtt{EtBE}}, \mathsf{Dec}_{\mathtt{EtBE}}, \mathsf{F}_{\mathtt{EtBE}})$ is constructed as in Fig. 5 (right). Like $\Pi^n_{\mathtt{BE}}$, the key generation algorithm $\mathsf{PKG}_{\mathtt{EtBE}}$ is PKG_1 itself, and we do not show it in the figure. The plaintext space of $\Pi_{\mathtt{EtBE}}$ is $\{0,1\}^k$.

On the Correctness of $\Pi_{\mathtt{EtBE}}$. Note that the encryption algorithm $\mathsf{Enc}_{\mathtt{EtBE}}$ returns $\perp$ if it happens to be the case that $\mathtt{DUPCHK}(C) = 1$. This check is to ensure that a valid ciphertext does not have "duplicated" components, which is required due to our use of a $\mathcal{Q}$-non-malleable code whose non-malleability can only take care of a "non-duplicated" quoting. Since the probability (over the randomness of $\mathsf{Enc}_{\mathtt{EtBE}}$) that $\mathsf{Enc}_{\mathtt{EtBE}}$ outputs $\perp$ is not zero, our construction $\Pi_{\mathtt{EtBE}}$ does not satisfy correctness in a strict sense. (The exactly same problem arises in the construction of string commitments in [2].) However, it is easy to show that if Π_1 satisfies CCA security (or even CPA security), the probability of $\mathsf{Enc}_{\mathtt{EtBE}}$ outputting $\perp$ is negligible, and thus it does not do any harm in practice. (In practice, for example, in case $\perp$ is output, one can re-execute $\mathsf{Enc}_{\mathtt{EtBE}}$ with a fresh randomness. The expected execution time of $\mathsf{Enc}_{\mathtt{EtBE}}$ is negligibly close to 1.) Furthermore, if one needs standard correctness, then instead of letting $\mathsf{Enc}_{\mathtt{EtBE}}$ output $\perp$ in

case $\texttt{DUPCHK}(C) = 1$, one can let it output a plaintext m (being encrypted) as an "irregular ciphertext", so that if the decryption algorithm $\mathsf{Dec}_{\texttt{EtBE}}$ takes an irregular ciphertext C as input, it outputs C as a "decryption result" of C. (In order to actually implement this, in case $\texttt{DUPCHK}(C) = 1$ occurs, $m \in \{0,1\}^k$ needs to be padded to the length $n \cdot |c|$ of an ordinary ciphertext, and we furthermore need to put a prefix for every ciphertext that tells the decryption algorithm whether the received ciphertext should be treated as a normal ciphertext or an irregular one.) Such a modification also does no harm to the security properties of $\Pi_{\texttt{EtBE}}$ (it only contributes to increasing an adversary's advantage negligibly), thanks to the CCA security of the building block Π_1. For simplicity, in this paper we focus on the current construction of $\Pi_{\texttt{EtBE}}$.

Security of $\Pi_{\texttt{EtBE}}$. The security properties of the EtBE construction is guaranteed by the following lemmas.

Lemma 8. *Assume that* Π_1 *is* CCA *secure and* $\mathcal{C}$ *is a* $\mathcal{Q}$*-non-malleable code. Then, the detectable PKE scheme* $\Pi_{\texttt{EtBE}}$ *in Fig. 5 (right) is* wRNM-DCCA *secure.*

Lemma 9. *If* Π_1 *is* CCA *secure, then the detectable PKE scheme* $\Pi_{\texttt{EtBE}}$ *scheme in Fig. 5 (right) satisfies unpredictability and randomness-inextractabilty.*

The proof of Lemma 9 is straightforward given the unpredictability (Lemma 6) and randomness-inextractability (Lemma 7) of the bitwise-encrypt construction $\Pi^n_{\texttt{BE}}$, and thus omitted.

The proof of Lemma 8 follows essentially the same story line as the security proof of the non-malleable string commitment by Agrawal et al. [2]. A high-level idea is as follows: In the wRNM-DCCA experiment, an adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2, \mathcal{A}_3)$ is allowed to submit a single "unrestricted" decryption query $C' = (c'_1, \ldots c'_n)$, which is captured by the ciphertext finally output by $\mathcal{A}_2$. In order for this query to be valid, however, C' has to satisfy $\texttt{DUPCHK}(C') = 0$, which guarantees that C' does not have duplicated components. Thus, since each component is a ciphertext of the CCA secure scheme Π_1, the best $\mathcal{A}$ can do to generate C' that is "related" to the challenge ciphertext $C^* = (c^*_1, \ldots, c^*_n)$ is to "quote" some of c^*_i's into C' in such a way that no c^*_i appears more than once. However, such "quoting without duplicated positions" is exactly the function class $\mathcal{Q}$ with respect to which the code $\mathcal{C}$ is non-malleable. Specifically, the $\mathcal{Q}$-non-malleability of $\mathcal{C}$ guarantees that even if an adversary observes the decryption result of such C' that quotes some of components of C^* without duplicated positions, $\mathcal{A}$ gains essentially no information of the original content m_b of the encoding s^* encrypted in C^*, and hence no information of the challenge bit b. Actually, it might be the case that $\mathcal{A}$ succeeds in generating C' so that $\mathsf{Dec}^n_{\texttt{BE}}(sk, C')$ is s^* itself (and hence its decoded value is exactly the challenge plaintext m_b). According to the rule of the wRNM-DCCA experiment, however, in such a case $\mathcal{A}$ is not given the actual decryption result $\mathsf{Dec}_{\texttt{EtBE}}(sk, C')$ directly but is given the symbol same which only informs that the decryption result is either m_0 or m_1. Furthermore, all other queries without quoting do not leak the information of the challenge bit b because of the DCCA security of the bitwise-encrypt construction $\Pi^n_{\texttt{BE}}$ (Lemma 6).

These ideas lead to wRNM-DCCA security of $\Pi_{\mathtt{EtBE}}$. For the details, see the proof in the full version.

6 Full Description of Our 1-bit-to-Multi-bit Conversion

Given the results in the previous sections, we are now ready to describe our proposed 1-bit-to-multi-bit conversion, i.e. a CCA secure KEM from a CCA secure 1-bit PKE scheme. Let $\Pi_1 = (\mathsf{PKG}_1, \mathsf{Enc}_1, \mathsf{Dec}_1)$ be a 1-bit PKE scheme whose public key size is "$|pk|$", the ciphertext size is "$|c|$", and the randomness space of whose encryption algorithm Enc_1 is $\{0,1\}^{\ell}$. Let $\mathcal{C} = (\mathsf{E}, \mathsf{D})$ be a $\mathcal{Q}$-non-malleable (n,k)-code with $n = n(k) \geq k$, and the randomness space of whose encoding algorithm E is $\{0,1\}^{\widehat{\ell}}$. Let $\ell' = n \cdot \ell + \widehat{\ell} + 2k$, and $\mathsf{G} : \{0,1\}^k \to \{0,1\}^{\ell'}$ be a PRG. Finally, let $E = (\mathsf{SEnc}, \mathsf{SDec})$ be a deterministic SKE scheme whose plaintext space is $\{0,1\}^{k \cdot |c|}$, and it has zero ciphertext overhead (i.e. its ciphertext size is the same as that of a plaintext).

From these building blocks, consider the following detectable KEM $\Gamma_{\mathtt{in}}$ and detectable PKE scheme $\Pi_{\mathtt{out}}$:

- $\Gamma_{\mathtt{in}}$: Consider the bitwise-encrypt construction $\Pi^k_{\mathtt{BE}}$ (Fig. 5) based on the PKE scheme Π_1, and regard it as a detectable KEM by encrypting a random k-bit string as a session-key. For this detectable KEM, use the PRG G with the method explained in the first paragraph of Sect. 3.3 to stretch its session-key into ℓ' bits. $\Gamma_{\mathtt{in}}$ is the resultant KEM.
  The public key size of $\Gamma_{\mathtt{in}}$ is $|pk|$, its ciphertext size is $k \cdot |c|$, and its session-key space is $\{0,1\}^{\ell'}$. Due to Lemmas 3 and 6, $\Gamma_{\mathtt{in}}$ satisfies DCCA security and unpredictability based on the CCA security of Π_1 and the security of G.
- $\Pi_{\mathtt{out}}$: Consider the EtBE construction $\Pi_{\mathtt{EtBE}}$ based on the code $\mathcal{C}$ and the bitwise-encrypt construction $\Pi^n_{\mathtt{BE}}$ (which is in turn based on Π_1) (Fig. 5). Combine this detectable PKE scheme with the SKE scheme E by the method explained in the second paragraph of Sect. 3.3 (see Fig. 3). $\Pi_{\mathtt{out}}$ is the resultant PKE scheme.
  The public key size of $\Pi_{\mathtt{out}}$ is $|pk|$, its ciphertext overhead (the difference between the total ciphertext size minus the plaintext size) is $n \cdot |c|$, its plaintext space is $\{0,1\}^{k \cdot |c|}$, and the randomness space of its encryption algorithm is $\{0,1\}^{\ell' - k}$. Due to Lemmas 4, 6, 7, 8, and 9, $\Pi_{\mathtt{out}}$ satisfies wRNM-DCCA security and randomness-inextractability, based on the CCA security of Π_1, $\mathcal{Q}$-non-malleability of $\mathcal{C}$, and the CCA security of E.

Our proposed KEM $\widetilde{\Gamma} = (\widetilde{\mathsf{KKG}}, \widetilde{\mathsf{Encap}}, \widetilde{\mathsf{Decap}})$ is then obtained from the double-layered construction $\Gamma_{\mathtt{DL}}$ in which the inner KEM is $\Gamma_{\mathtt{in}}$ and the outer PKE scheme is $\Pi_{\mathtt{out}}$ explained above. More concretely, the description of $\widetilde{\Gamma}$ is as in Fig. 6.

The public key size of $\widetilde{\Gamma}$ is $2 \cdot |pk|$, and its ciphertext size is $(n+k) \cdot |c|$ (where $\Gamma_{\mathtt{in}}$ contributes $k \cdot |c|$ and $\Pi_{\mathtt{out}}$ contributes $n \cdot |c|$). Using the $\mathcal{P}$-non-malleable code with "optimal rate" (Lemma 1) by Agrawal et al. [1] which also satisfies

$\widetilde{\mathsf{KKG}}(1^k)$:
$(pk_{\mathtt{in}}, sk_{\mathtt{in}}) \leftarrow \mathsf{PKG}_1(1^k)$
$(pk_{\mathtt{out}}, sk_{\mathtt{out}}) \leftarrow \mathsf{PKG}_1(1^k)$
$PK \leftarrow (pk_{\mathtt{in}}, pk_{\mathtt{out}})$
$SK \leftarrow (sk_{\mathtt{in}}, sk_{\mathtt{out}}, PK)$
Return (PK, SK).

$\widetilde{\mathsf{Encap}}(PK)$:
$(pk_{\mathtt{in}}, pk_{\mathtt{out}}) \leftarrow PK$
$K_{\mathtt{in}} = (K_{\mathtt{in}}^{(1)} \| \dots \| K_{\mathtt{in}}^{(k)}) \leftarrow \{0,1\}^k$
$\forall i \in [k] : c_{\mathtt{in}}^{(i)} \leftarrow \mathsf{Enc}_1(pk_{\mathtt{in}}, K_{\mathtt{in}}^{(i)})$
$\alpha \leftarrow \mathsf{G}(K_{\mathtt{in}})$
Parse α as $(r_1, \dots, r_n, \widehat{r}, K_{\mathtt{out}}, K)$
$\quad \in (\{0,1\}^\ell)^n \times \{0,1\}^{\widehat{\ell}} \times (\{0,1\}^k)^2$.
$s = (s_1 \| \dots \| s_n) \leftarrow \mathsf{E}(1^k, K_{\mathtt{out}}; \widehat{r})$
$\forall i \in [n] : c_i \leftarrow \mathsf{Enc}_1(pk_{\mathtt{out}}, s_i; r_i)$
If $\mathtt{DUPCHK}((c_i)_{i\in[n]}) = 1$ then return $\perp$.
$\widehat{c} \leftarrow \mathsf{SEnc}(K_{\mathtt{out}}, (c_{\mathtt{in}}^{(1)} \| \dots \| c_{\mathtt{in}}^{(k)}))$
$C \leftarrow (c_1, \dots, c_n, \widehat{c})$
Return (C, K).

$\widetilde{\mathsf{Decap}}(SK, C)$:
$(sk_{\mathtt{in}}, sk_{\mathtt{out}}, PK) \leftarrow SK$
$(pk_{\mathtt{in}}, pk_{\mathtt{out}}) \leftarrow PK$; $(c_1, \dots, c_n, \widehat{c}) \leftarrow C$
If $\mathtt{DUPCHK}((c_i)_{i\in[n]}) = 1$ then return $\perp$.
$\forall i \in [n] : s_i \leftarrow \mathsf{Dec}_1(sk_{\mathtt{out}}, c_i)$
If $\exists i \in [n] : s_i = \perp$ then return $\perp$.
$K_{\mathtt{out}} \leftarrow \mathsf{D}(1^k, s = (s_1 \| \dots \| s_n))$
If $K_{\mathtt{out}} = \perp$ then return $\perp$.
$(c_{\mathtt{in}}^{(1)} \| \dots \| c_{\mathtt{in}}^{(k)}) \leftarrow \mathsf{SDec}(K_{\mathtt{out}}, \widehat{c})$
If SDec has returned $\perp$ then return $\perp$.
$\forall i \in [k] : K_{\mathtt{in}}^{(i)} \leftarrow \mathsf{Dec}_1(sk_{\mathtt{in}}, c_{\mathtt{in}}^{(i)})$
If $\exists i \in [k] : K_{\mathtt{in}}^{(i)} = \perp$ then return $\perp$.
$\alpha \leftarrow \mathsf{G}(K_{\mathtt{in}} = (K_{\mathtt{in}}^{(1)} \| \dots \| K_{\mathtt{in}}^{(k)}))$
Parse α as $(r_1, \dots, r_n, \widehat{r}, K'_{\mathtt{out}}, K)$
$\quad \in (\{0,1\}^\ell)^n \times \{0,1\}^{\widehat{\ell}} \times (\{0,1\}^k)^2$.
If **(a)** $\wedge$ **(b)** $\wedge$ **(c)** then return K
else return $\perp$:
(a) $\forall i \in [n] : \mathsf{Enc}_1(pk_{\mathtt{out}}, s_i; r_i) = c_i$
(b) $\mathsf{E}(1^k, K'_{\mathtt{out}}; \widehat{r}) = s$
(c) $\mathsf{SEnc}(K'_{\mathtt{out}}, (c_{\mathtt{in}}^{(1)} \| \dots \| c_{\mathtt{in}}^{(k)})) = \widehat{c}$

Fig. 6. The proposed "1-bit-to-multi-bit" construction (KEM) $\widetilde{\Gamma}$.

$\mathcal{Q}$-non-malleability by Lemma 2, we have $n = k + o(k)$. Thus, the ciphertext size of $\widetilde{\Gamma}$ can be made asymptotically $(2k + o(k)) \cdot |c|$.

The following statement is obtained as a corollary of the combination of Theorem 1 and Lemmas 1, 2, 3, 4, 6, 7, 8, and 9.

Theorem 4. *Assume that the PKE scheme Π_1 is* CCA *secure, $\mathcal{C}$ is a $\mathcal{Q}$-non-malleable code, G is a PRG, and the SKE scheme E is* CCA *secure. Then, the KEM $\widetilde{\Gamma}$ in Fig. 6 is* CCA *secure.*

2-bit-to-multi-bit Construction with a Single Key Pair. Note that our proposed 1-bit-to-multi-bit conversion $\widetilde{\Gamma}$ uses two key pairs. It turns out that if we can use a 2-bit PKE scheme as a building block instead of a 1-bit scheme, then we can construct a CCA secure KEM that uses only one key pair of the underlying 2-bit scheme, with a very similar way to $\widetilde{\Gamma}$. The idea of this 2-bit-to-multi-bit conversion is to use the additional 1-bit of the plaintext space as the "indicator bit" that indicates whether each component ciphertext is generated for the inner layer or the outer layer. That is, each inner ciphertext $c_{\mathtt{in}}^{(i)}$ is an encryption of $(1 \| K_{\mathtt{in}}^{(i)})$, and each outer ciphertext c_i is an encryption of $(0 \| s_i)$, and in the decapsulation algorithm, we check whether the component ciphertexts $\{c_i\}_{i\in[n]}$ and $\{c_{\mathtt{in}}^{(i)}\}_{i\in[k]}$ have appropriate indicator bits ("1" for the inner layer and "0"

for the outer layer). This additional indicator bit and its check prevent a quoting of an inner ciphertext into the outer layer and vice versa, and thus make the encryption/decryption operations for the inner layer and those of the outer layer virtually independent, as if each layer has an individual key pair. This enables us to conduct the security proof in essentially the same way as that of $\widetilde{\Gamma}$. Due to the lack of space, we detail it in the full version.

On the Necessity of Two Key Pairs. As mentioned in Introduction, our positive results on the 1-/2-bit-to-multi-bit constructions for CCA security raise an interesting question in terms of the number of public keys: Is it necessary to use two key pairs in 1-bit-to-multi-bit constructions for CCA security? Motivated by this question, in the full version we consider the one-key variant of our proposed KEM $\widetilde{\Gamma}$, and show that it is vulnerable to a CCA attack. Hence, using two key pairs of the underlying 1-bit scheme is essential for our proposed construction $\widetilde{\Gamma}$. Clarifying the optimality of the number of key pairs in 1-bit-to-multi-bit constructions would be an interesting open problem.

7 Comparison

Table 1 compares the public key size and ciphertext size of the existing "1-bit-to-multi-bit" constructions that achieve CCA security (or related security). Specifically, in the table, "MS" represents the construction by Myers and Shelat [20].; "HLW" represents the construction by Hohenberger et al. [16] which uses a CPA secure PKE scheme, a 1-bounded CCA secure [7] PKE scheme, and a detectable PKE scheme satisfying DCCA security and unpredictability. We assume that for the 1-bounded CCA secure scheme, the construction by Dodis and Fiore [11, Appendix C] is used, which constructs such a scheme from a CPA secure scheme and a one-time signature scheme, and we also assume that its detectable scheme and the CPA secure scheme are realized by the bitwise-encrypt construction $\Pi^k_{\mathtt{BE}}$. (If we need to encrypt a value longer than k-bit, then we assume that hybrid encryption is used everywhere possible by encrypting a k-bit random session-key and using it as a key for SKE (where the length of SKE ciphertexts are assumed to be the same as a plaintext [23]), which we do the same for the constructions explained below.); "MH" represents the construction by Matsuda and Hanaoka [19], which can be seen as an efficient version of HLW [16] due to hybrid encryption techniques, and we assume that the building blocks similar to HLW are used.; "CMTV" represents the construction by Coretti et al. [6], the size parameters of which are taken from the introduction of [6].; "CDTV" represents the construction by Coretti et al. [5], where the size parameters are estimated according to the explanations in [5, Sections 4.2 & 4.3].; "Ours" is the KEM $\widetilde{\Gamma}$ shown in Fig. 6 in Sect. 6.

As is clear from Table 1, if one starts from a CCA secure 1-bit PKE scheme (and assuming that building blocks implied by one-way functions are available for free), then "Ours" achieves asymptotically the best efficiency. Notably, the public size and the ciphertext size of "Ours" are asymptotically "optimal" in the sense that they are asymptotically the same as the bitwise-encrypt construction

Table 1. Comparison among the 1-bit-to-multi-bit constructions for CCA (and related) security.

Scheme	PK size	Ciphertext size	Sec. of Π_1	Add. Bld. Blk
MS [20]	$(20k^2+1)\|pk\|$	$(10k^3\|c\|+\|vk\|+\|\sigma\|)\|c\|$	CCA	Sig., PRG
HLW [16]	$(2k+2)\|pk\|$	$(k^2+3k)\|c\|+\|vk\|+\|\sigma\|+6k$	DCCA &UNP	Sig., PRG, SKE
MH [19]	$(2k+2)\|pk\|$	$(k^2+2k)\|c\|+\|vk\|+\|\sigma\|$	DCCA &UNP	Sig., PRG, SKE
CMTV† [6]	$\approx k\|pk\|$	$\approx 5k\|c\|$	SDA	—
CDTV† [5]	$O(k)\|pk\|$	$O(k)\|c\|$	NM-SDA	—
Ours (Sect. 6)	$2\|pk\|$	$(2k+o(k))\|c\|$	CCA	PRG, SKE

In the columns "PK Size" and "Ciphertext Size", $|pk|$ and $|c|$ denote the public key size and the ciphertext size of the underlying 1-bit PKE scheme Π_1, respectively, and $|vk|$ and $|\sigma|$ denote the size of a verification key and that of a signature of the one-time signature scheme used as a building block, respectively. The column "Sec. on Π_1" shows the assumption on the security of the underlying 1-bit PKE scheme required to show the CCA (or the related) security of the entire construction. Here, "SDA" and "NM-SDA" denote "(indistinguishability against) self-destruct CCA" [6] and "non-malleability against SDA" [5], respectively. The column "Add. Bld. Blk." shows the additional building blocks (used in each construction) that can be realized only from the existence of a one-way function. Here, "Sig" stands for a one-time signature scheme. (†) As explained in Introduction, CMTV [6] and CTDV [5] only achieve SDA security and NM-SDA security, respectively, which are both implied by ordinary CCA security but are strictly weaker than it

Π^k_{BE} that works as a 1-bit-to-multi-bit conversion for the CPA and non-adaptive CCA (CCA1) settings. Note also that all the previous constructions that achieve ordinary CCA security have the public key size $\Omega(k)\cdot|pk|$, and the ciphertext size $\Omega(k^2)\cdot|c|$.

We note that, as mentioned in Sect. 1.3, CMTV [6] and CDTV [5] achieve only indistinguishability under self-destruct CCA (SDA) and non-malleability under self-destruct CCA (NM-SDA), respectively, which are both implied by ordinary CCA security but are strictly weaker than it. Nonetheless, "Ours" actually achieves better asymptotic efficiency than them.

However, for fairness we note that our construction requires CCA security for the underlying 1-bit PKE scheme Π_1, while HLW [16] and MH [19] only require DCCA security and unpredictability, and the constructions CMTV [6] and CDTV [5] only require SDA and NM-SDA security for Π_1, respectively, and thus there is a tradeoff among the assumptions on the building block Π_1.

Acknowledgement. The authors would like to thank the members of the study group "Shin-Akarui-Angou-Benkyou-Kai," and the anonymous reviewers of ASIACRYPT 2015 for their helpful comments and suggestions.

References

1. Agrawal, S., Gupta, D., Maji, H.K., Pandey, O., Prabhakaran, M.: A rate-optimizing compiler for non-malleable codes against bit-wise tampering and permutations. In: Dodis, Y., Nielsen, J.B. (eds.) TCC 2015, Part I. LNCS, vol. 9014, pp. 375–397. Springer, Heidelberg (2015)

2. Agrawal, S., Gupta, D., Maji, H.K., Pandey, O., Prabhakaran, M.: Explicit non-malleable codes against bit-wise tampering and permutations. In: Gennaro, R., Robshaw, M. (eds.) CRYPTO 2015. LNCS, vol. 9216, pp. 538–557. Springer, Heidelberg (2015)
3. Bellare, M., Sahai, A.: Non-malleable encryption: equivalence between two notions, and an indistinguishability-based characterization. In: Wiener, M. (ed.) CRYPTO 1999. LNCS, vol. 1666, pp. 519–536. Springer, Heidelberg (1999)
4. Canetti, R., Krawczyk, H., Nielsen, J.B.: Relaxing chosen-ciphertext security. In: Boneh, D. (ed.) CRYPTO 2003. LNCS, vol. 2729, pp. 565–582. Springer, Heidelberg (2003)
5. Coretti, S., Dodis, Y., Tackmann, B., Venturi, D.: Non-malleable encryption: simpler, shorter, stronger (2015). http://eprint.iacr.org/2015/772
6. Coretti, S., Maurer, U., Tackmann, B., Venturi, D.: From single-bit to multi-bit public-key encryption via non-malleable codes. In: Dodis, Y., Nielsen, J.B. (eds.) TCC 2015, Part I. LNCS, vol. 9014, pp. 532–560. Springer, Heidelberg (2015)
7. Cramer, R., Hanaoka, G., Hofheinz, D., Imai, H., Kiltz, E., Pass, R., Shelat, A., Vaikuntanathan, V.: Bounded CCA2-secure encryption. In: Kurosawa, K. (ed.) ASIACRYPT 2007. LNCS, vol. 4833, pp. 502–518. Springer, Heidelberg (2007)
8. Cramer, R., Shoup, V.: Design and analysis of practical public-key encryption schemes secure against adaptive chosen ciphertext attack. SIAM J. Comput. **33**(1), 167–226 (2003)
9. Dachman-Soled, D., Fuchsbauer, G., Mohassel, P., O'Neill, A.: Enhanced chosen-ciphertext security and applications. In: Krawczyk, H. (ed.) PKC 2014. LNCS, vol. 8383, pp. 329–344. Springer, Heidelberg (2014)
10. Dodis, Y., Fiore, D.: Interactive encryption and message authentication. In: Abdalla, M., De Prisco, R. (eds.) SCN 2014. LNCS, vol. 8642, pp. 494–513. Springer, Heidelberg (2014)
11. Dodis, Y., Fiore, D.: Interactive encryption and message authentication (2013). Full version of [10]. http://eprint.iacr.org/2013/817
12. Dolev, D., Dwork, C., Naor, M.:Non-malleable cryptography. In: STOC 1991, pp. 542–552. ACM (1991)
13. Dziembowski, S., Pietrzak, K., Wichs, D.: Non-malleable codes. In: ICS 2010, pp. 434–452 (2010)
14. Dziembowski, S., Pietrzak, K., Wichs, D.: Non-malleable codes. Full version of [13]. http://eprint.iacr.org/2009/608
15. Faust, S., Mukherjee, P., Nielsen, J.B., Venturi, D.: Continuous non-malleable codes. In: Lindell, Y. (ed.) TCC 2014. LNCS, vol. 8349, pp. 465–488. Springer, Heidelberg (2014)
16. Hohenberger, S., Lewko, A., Waters, B.: Detecting dangerous queries: a new approach for chosen ciphertext security. In: Pointcheval, D., Johansson, T. (eds.) EUROCRYPT 2012. LNCS, vol. 7237, pp. 663–681. Springer, Heidelberg (2012)
17. Kitagawa, F., Matsuda, T., Hanaoka, G., Tanaka, K.: Completeness of single-bit projection-kdm security for public key encryption. In: Nyberg, K. (ed.) CT-RSA 2015. LNCS, vol. 9048, pp. 201–219. Springer, Heidelberg (2015)
18. Lin, H., Tessaro, S.: Amplification of chosen-ciphertext security. In: Johansson, T., Nguyen, P.Q. (eds.) EUROCRYPT 2013. LNCS, vol. 7881, pp. 503–519. Springer, Heidelberg (2013)
19. Matsuda, T., Hanaoka, G.: Achieving chosen ciphertext security from detectable public key encryption efficiently via hybrid encryption. In: Sakiyama, K., Terada, M. (eds.) IWSEC 2013. LNCS, vol. 8231, pp. 226–243. Springer, Heidelberg (2013)

20. Myers, S., Shelat, A.: Bit encryption is complete. In: FOCS 2009, pp. 607–616 (2009)
21. Naor, M., Yung, M.: Public-key cryptosystems provably secure against chosen ciphertext attacks. In: STOC 1990, pp. 427–437. ACM (1990)
22. Pass, R., Shelat, A., Vaikuntanathan, V.: Relations among notions of non-malleability for encryption. In: Kurosawa, K. (ed.) ASIACRYPT 2007. LNCS, vol. 4833, pp. 519–535. Springer, Heidelberg (2007)
23. Phan, D.H., Pointcheval, D.: About the security of ciphers (semantic security and pseudo-random permutations). In: Handschuh, H., Hasan, M.A. (eds.) SAC 2004. LNCS, vol. 3357, pp. 182–197. Springer, Heidelberg (2004)