Separations between simulation-based and simulation-free formulations of security for public key encryption*

Yodai Watanabe

University of Aizu, Aizuwakamatsu, Fukushima 9658580, Japan
yodai@u-aizu.ac.jp

Abstract. Simulation-based formulation of security enables us to naturally capture our intuition for security. However, since the simulation-based formulation is rather complicated, it is convenient to consider alternative simulation-free formulations which are easy to manipulate but can be employed to give the same security as the simulation-based one. So far the indistinguishability-based and comparison-based formulations have been introduced as such ones. Regarding the security for public key encryption, while these three formulations are shown equivalent in most settings, some relations among these formulations of non-malleability under the valid ciphertext condition, in which an adversary fails if it outputs an invalid ciphertext, remain open. This work aims to help to consider the appropriateness of the formulations of security by clarifying the above open relations among the formulations of non-malleable encryption.

1 Introduction

Simulation is one of the fundamental methods for formulating a cryptographic primitive (see e.g. [20]). It compares the "real world" in which the primitive works and an "ideal world" in which its desired properties follow by definition, and requires that the two worlds are indistinguishable. For example, in formulating security of a public key encryption scheme, it compares an adversary attacking the scheme, given a real ciphertext called a challenge ciphertext, to its simulator without the ciphertext (and access to the decryption oracle), and requires that the adversary and simulator are indistinguishable in attacking the scheme. Since the main difference between an adversary and its simulator is in whether a challenge ciphertext is given or not, the above requirement ensures that the ciphertext does not help to attack the scheme.

As can be seen above, simulation provides an intuitively natural formulation of security, but it requires two parties, an adversary and its simulator. Hence, it is convenient to introduce alternative simulation-free formulations

* This work was supported in part by JSPS Grants-in-Aid for Scientific Research (C) No. 19K11831.

which require only an adversary (and can be employed to give the same security as the simulation-based one). As such simulation-free formulations, we have the comparison-based and indistinguishability-based ones. Here, the simulation-based and comparison-based formulations have the same goal but differ in the baseline to which the adversary is compared, and the indistinguishability-based one has a simple goal (different from that of the simulation-based one). To see these formulations in more detail, let us consider public key encryption schemes.

The security of public key encryption schemes is commonly specified by the security goal and the attack model. Here, the security goals formulate what type of security of the scheme is intended to be protected from an adversary, and the attack models formulate what type of external resources is assumed to be available to an adversary. Definitions of security in the common framework [3] are informally described as follows. An adversary A is a pair of algorithms, $A = (A_1, A_2)$, corresponding to two stages of an attack. At the first stage of the attack, A_1 takes as input the public key pk and outputs a distribution M over messages (plaintexts). Next, a plaintext x is sampled according to M and then encrypted to give a challenge ciphertext y. At the second stage of the attack, A_2 takes as input the challenge ciphertext y, and the success condition for A is determined according to the security goal. Here, the standard security goals are semantic security and non-malleability.[1] Semantic security relates to the secrecy that a ciphertext does not leak any partial information about its plaintext. In the (simulation-based)[2] semantic security (SSS) [16], an adversary with y and its simulator without y are considered successful if they can compute partial information about x described by a function F, and an encryption scheme is considered secure if for any adversary and for any function F, there exists a simulator such that the difference between their success probabilities is "negligible." In the indistinguishability-based semantic security [16], now called (ciphertext) indistinguishability (IND), the distribution M is restricted to the form $M = \{x_0, x_1\}$, and an encryption scheme is considered secure if any adversary with y can guess whether $x = x_0$ or $x = x_1$ with probability only negligibly larger than the baseline probability $\frac{1}{2}$.

In contrast, non-malleability relates to the resistance against ciphertext modifications that a ciphertext cannot be modified into other ciphertexts so that their plaintexts are "meaningfully" related. In the simulation-based non-malleability (SNM) [11], an adversary with y and its simulator without y are considered successful if they can generate ciphertexts $\boldsymbol{y}$ of plaintexts $\boldsymbol{x}$ other than y so that x and $\boldsymbol{x}$ satisfy a relation R, and an encryption scheme is considered secure if for any adversary and for any relation R, there exists a simulator such that the difference between their success probabilities is negligible. In the comparison-based non-malleability (SNM) [3], an adversary with y is considered successful

[1] In addition to encryption schemes, non-malleability has been formulated for various primitives (see e.g. [6, 9, 11, 12, 17, 26]).

[2] The comparison-based semantic security (CSS) was introduced in [2] for private key encryption. Semantic security not based on simulation may seem contrary to its "spirit," but it frees us from considering the encryption oracle for a simulator.

as in SNM, but in this case, the relation R is output by the adversary and its success probability is compared with that of "random guess" which corresponds to the coincidence that a plaintext x' independently sampled according to M and $\boldsymbol{x}$ satisfy the relation R;[3] and an encryption scheme is considered secure if for any adversary, the difference in success probability between the adversary and "random guess" is negligible. The indistinguishability-based non-malleability (IND) [4] is described against non-standard attack models called parallel chosen-ciphertext attacks. Here, the standard attack models are chosen plaintext attack (CPA), non-adaptive chosen ciphertext attack (CCA1) [22] and adaptive chosen ciphertext attack (CCA2) [25], where access to the decryption oracle is allowed only to A_1 in CCA1 and to both A_1 and A_2 in CCA2 (no access is allowed in CPA). In the parallel chosen-ciphertext attacks, PCA0, PCA1 and PCA2, an adversary has the same access to the decryption oracle as in CPA, CCA1 and CCA2, respectively, and can further make one parallel decryption query after receiving the challenge ciphertext y. Then, in IND against PCAX with PCAX$\in$ {PCA0,PCA1,PCA2}, an encryption scheme is considered secure if any adversary with y can guess whether $x = x_0$ or $x = x_1$ with probability only negligibly larger than the baseline probability $\frac{1}{2}$.

Now, we have nine formulations of non-malleability (depending on the three formulations and three attack models), and each has its variant in which the valid ciphertext condition is imposed, where under the valid ciphertext condition (below indicated by $*$ attached to the security goal), an adversary fails if it outputs an invalid ciphertext. Among these definitions, the simulation-based formulation under the valid ciphertext condition (SNM*), introduced by the original work [11], would be the most natural one at least from our intuition for non-malleability.[4] However, whether imposing the valid ciphertext condition is more appropriate or not depends on applications, as mentioned in [4,19,23].[5] So far SNM, CNM and IND have been shown equivalent against the same level of the attack models [4]; on the other hand, SNM*, CNM* and IND* have been shown equivalent only against the strongest attack model (CCA2/PCA2), and the relations among them against the weaker attack models (CPA/PCA0 and CCA1/PCA1) remain open. This work clarifies the above open relations among formulations of non-malleability by showing the separations between the simulation-based and indistinguishability-based formulations and the simulation-based and comparison-based formulations.

[3] It may seem somewhat strange that the success probability of "random guess" is considered with respect to the ciphertexts $\boldsymbol{y}$ output by the adversary, in particular in comparison with the simulation-based formulation in which the success probability of a simulator is considered with respect to the ciphertexts $\boldsymbol{y}$ output by the simulator itself. This difference is essential in the proof of SNM$^* \not\Rightarrow$ CNM*, provided in appendix A.

[4] Katz and Yung [19] imposed the valid ciphertext condition to formulate non-malleability for private-key encryption based on the consideration that "the current definition more closely corresponds to our intuitive notion."

[5] An illustrative example (quorum voting application), in which a CNM adversary is more advantageous than a CNM* adversary, was presented in [23].

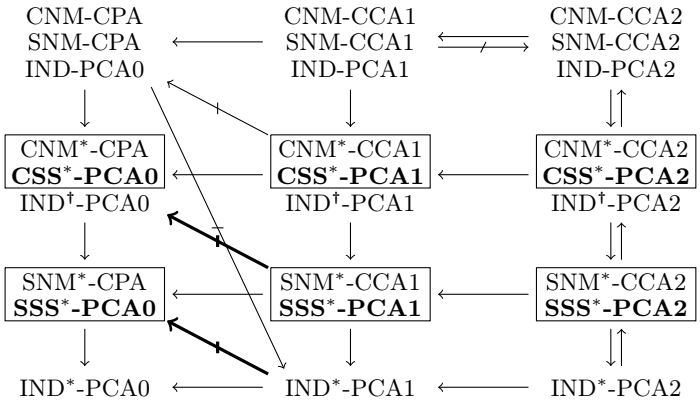

Fig. 1. Relations among formulations of non-malleability. The asterisk attached to the security goals indicates that the valid ciphertext condition is imposed. The bold barred arrows represent the separations shown by this paper. The bold security notions are introduced in this paper, and the notions in each box are shown to be equivalent by this paper. The separation CNM*-CCA1 $\not\Rightarrow$ CNM-CPA follows from the result CNM*-CPA $\not\Rightarrow$ CNM-CPA shown by the full version of [18], together with the idea mentioned in the full version of [4]. The other relations are consequences of [3, 4, 11, 19].

1.1 Contributions and related works

Figure 1 summarizes the relations among formulations of non-malleable encryption shown in related works and this work, where the asterisk attached to the security goals indicates that the valid ciphertext condition is imposed. The equivalence among the notions against the strongest attack model (CCA2/PCA2) readily follows from the equivalence SNM/CNM-CCA2 $\iff$ IND-CCA2 [3, 11]. The separation CNM-CCA1 $\not\Rightarrow$ CNM-CCA2 was shown in [3] and it is straightforward to modify the proof of the separation CNM-CPA $\not\Rightarrow$ IND-CCA1 [3] to that of CNM-CPA $\not\Rightarrow$ IND*-PCA1. The separation CNM*-CCA1 $\not\Rightarrow$ CNM-CPA follows from the result CNM*-CPA $\not\Rightarrow$ CNM-CPA shown by the full version of [18], together with the idea mentioned in the full version of [4]. In addition, the equivalence among CNM-ATK, SNM-ATK and IND-PCAX was shown in [4] and the proof of the equivalence between CNM*-ATK and IND†-PCAX for private-key encryption [19] (see appendix D for the definition of IND†-PCAX) straightforwardly applies to that for public-key encryption, where (ATK,PCAX) $\in$ {(CPA,PCA0),(CCA1,PCA1),(CCA2,PCA2)}.

This work (perhaps surprisingly) shows that SNM*-CCA1 $\not\Rightarrow$ CNM*-CPA, which answers the last open question mentioned in the full version of [4]. Furthermore, this work also shows that IND*-PCA1 $\not\Rightarrow$ SNM*-CPA. We note that these separations complete Figure 1 and no relation remains open. The proofs of these results follow the standard procedure to show the separation $X \not\Rightarrow Y$ for computational security notions X and Y, in which (a) the existence of an

X-secure encryption scheme Π is assumed and then (b) Π is modified to Π' so that Π' is still X-secure but not Y-secure. However, the modifications of encryption schemes and the estimation of adversaries' advantages given in this paper specifically are aimed at showing the separations; in particular, the modifications are quite simple, which may help to clarify the appropriateness of the notions appearing in the separations. In addition, motivated by the proof of the latter separation, this paper introduces simulation-based and comparison-based formulations of semantic security (SSS* and CSS*) against parallel chosen-ciphertext attacks and shows that SSS* and CSS* are equivalent to SNM* and CNM*, respectively. This, together with the latter separation, shows that semantic security and ciphertext indistinguishability, which have been shown equivalent in various settings (see e.g. [2,13,15,16,21,27]), separate against the weaker parallel chosen-ciphertext attacks under the valid ciphertext condition.

In the proof of the separation SNM* $\not\Rightarrow$ CNM*, an encryption scheme is modified so that the decryption has an optional mode which allows an adversary to make a successful ciphertext modification if and only if the plaintext of the challenge ciphertext is 0. We may expect that an appropriate security notion does not allow an adversary to take advantage from this modification, but the proof shows that CNM* does. This indicates that CNM* is stronger than expected at least under the valid ciphertext condition. In the proof of the separation IND* $\not\Rightarrow$ SNM*(SSS*), an encryption scheme is modified so that the decryption has an optional mode which allows an adversary to make a successful ciphertext modification with probability $\frac{1}{2}$. We may expect that an appropriate security notion allows an adversary to take advantage from this modification (by choosing a message space of cardinality more than 2), but the proof shows that IND* (in which the cardinality of a message space is restricted to 2) does not. This indicates that IND* is weaker than expected at least under the valid ciphertext condition. Here, we note that the valid ciphertext condition is not introduced for showing the separations, but has been considered since the initial work of non-malleability [11]. We also note that the simulation-free notions (comparison-based non-malleability and ciphertext indistinguishability) are now the first choice for analyzing the security of cryptosystems of interest (see e.g. [1, 5, 7, 10, 23]). This may be because (i) they are simpler and easier to manipulate, and (ii) they have been shown to be equivalent to the corresponding simulation-based (intuitively secure) notions in various settings (see e.g. [2,4,13,15,16,21,27]). Therefore, the results of this work suggest that it is of importance to consider what a simulation-free notion of interest guarantees, in particular by confirming its equivalence to the corresponding simulation-based (intuitively secure) notion.

So far we have considered the security notions appearing in Figure 1, and we now consider those out of Figure 1. One motivation to consider a weaker security notion would be to provide a better construction of cryptosystems secure in the sense of the weaker notion. For example, an NM-CPA encryption scheme has a construction from an IND-CPA encryption scheme (without any other assumptions) [23], while it has not been known that an IND-CCA2 encryption

scheme has such a construction. For generalizing this result, the parallel chosen-ciphertext attack PCA0 was extended to self-destruct attack (SDA) [8], and non-malleability against self-destruct attack, NM-SDA, was introduced in [7]. Here, in SDA, an adversary is allowed to make multiple parallel decryption queries up to the point when the first invalid ciphertext is submitted. It was shown that NM-SDA is strictly stronger than NM-CPA, and an NM-SDA encryption scheme has a black-box construction from an IND-CPA encryption scheme [7].

Here, we mention possible applications of SSS*/CSS*-PCA0 encryption schemes[6] introduced in this paper (see Definitions 4 and 5) in comparing with those of NM-SDA encryption schemes. For this purpose, as in [7, 11], consider an electronic auction in which the auctioneer publishes a public key and invites participants to encrypt their bids under the public key. Then, since an NM-SDA adversary is allowed to submit multiple parallel decryption queries up to the point when it submits the first invalid ciphertext, this interprets that the auctioneer can reuse the public key for subsequent auctions as long as all the encrypted bids are valid. In contrast, since an SSS*/CSS*-PCA0 adversary is allowed to submit only one parallel decryption query, this interprets that the auctioneer should update the public key for each auction. Moreover, since the parallel decryption query containing the first invalid ciphertext is still fully answered for an NM-SDA adversary, this interprets that the corresponding auction is valid. In contrast, since an adversary submitting a decryption query containing an invalid ciphertext fails under the valid ciphertext condition, this interprets that the corresponding auction should be discarded. Note that for IND-CCA2 encryption schemes, the auctioneer can always reuse the public key, and all the auctions are valid, regardless of whether an invalid bid is submitted or not; on the other hand, as mentioned in [7], it has to be assumed that participants submitting an invalid bid are penalized for SDA and SSS*/CSS*-PCA0 schemes. It may be envisioned that SSS*/CSS*-PCA0 encryption schemes have applications illustrated as above, under computational assumptions (potentially) weaker than those for even SNM/CNM-CPA (and so NM-SDA) encryption schemes. We also note that the CNM* formulation was employed for the (main) definition of non-malleability in the theoretical work [19] which investigated the relations among security notions for private key encryption.

We next consider slightly different simulation-based and indistinguishability-based formulations of non-malleability, SIM-NME and IND-NME, respectively, introduced in [24]. The main differences between SIM-NME and SNM* and between IND-NME and IND* are in that (i) the former notions consider multiple messages, while the latter notions a single message, and (ii) outputting an invalid ciphertext is prohibited in the former notions, while it is allowed but results in failure of an attack in the latter notions.[7] In contrast to the security notions

[6] It follows from [7] that NM-SDA is strictly stronger than SSS*/CSS*-PCA0, but NM-SDA and SSS*/CSS*-PCA1 seem incomparable.

[7] As mentioned in [24], it is assumed in [4] that there is an efficient algorithm for a simulator to generate an invalid ciphertext; however, under the valid ciphertext condition, this assumption can simply be removed as follows. Namely, the simulator

considered in this paper, for which IND* $\not\Longrightarrow$ SNM* against the weaker attack models, the equivalence IND-NME $\Longleftrightarrow$ SIM-NME against all attack models was shown in [24]. This apparent contradiction is due to the difference (ii); if we modify SNM* and IND* so that the difference (ii) will be eliminated, then the equivalence between the modified SNM* and IND* readily follows from the results of [4], while even if we modify SNM* and IND* so that the difference (i) will be eliminated, the proof of IND* $\not\Longrightarrow$ SNM* in this paper will straightforwardly apply to the modified notions.

2 Preliminaries

Let A be a probabilistic algorithm. The result of running A on inputs $x_1, x_2, \ldots$ and randomness r is denoted by $A(x_1, x_2, \ldots; r)$. The notation $y \leftarrow A(x_1, x_2, \ldots)$ denotes the experiment of choosing r at random and setting $y = A(x_1, x_2, \ldots; r)$. If S is a distribution (resp. a finite set), then S in the notation $x \leftarrow S$ is considered an algorithm which returns a sample drawn according to S (resp. the uniform distribution over S). For an event E, the notation

$$\Pr[x \leftarrow A(a_1, a_2, \ldots); y \leftarrow B(b_1, b_2, \ldots); \cdots : E]$$

denotes the probability that E occurs after ordered execution of the listed experiments.

The length of a string s is denoted by $|s|$. The concatenation of strings s_1 and s_2 is denoted by $s_1 s_2$. A sequence is denoted in boldface. The length of a sequence $\boldsymbol{x}$ is denoted by $|\boldsymbol{x}|$ and its i-th component by $\boldsymbol{x}_i$, so that $\boldsymbol{x} = (\boldsymbol{x}_1, \cdots, \boldsymbol{x}_l)$ with $l = |\boldsymbol{x}|$. The concatenation of sequences $\boldsymbol{x}$ and $\boldsymbol{y}$ is denoted by $\boldsymbol{x}||\boldsymbol{y}$. For an operation F and a sequence $\boldsymbol{x}$ of length l whose components are in the domain of F, we use the notation $F(\boldsymbol{x})$ to denote

$$F(\boldsymbol{x}) = (F(\boldsymbol{x}_1), \cdots, F(\boldsymbol{x}_l)).$$

For a sequence $\boldsymbol{x}$ of length l_1 whose components are sequences of length l_2,

$$\boldsymbol{x} = ((\boldsymbol{x}_{11}, \cdots, \boldsymbol{x}_{1l_2}), \cdots, (\boldsymbol{x}_{l_1 1}, \cdots, \boldsymbol{x}_{l_1 l_2})),$$

we define a sequence $\boldsymbol{x}_{:j}$ for $j \in [l_2]$ by

$$\boldsymbol{x}_{:j} = (\boldsymbol{x}_{1j}, \cdots, \boldsymbol{x}_{l_1 j}). \tag{1}$$

For sequences $\boldsymbol{a}$, $\boldsymbol{b}$, $\boldsymbol{c}$ and $\boldsymbol{d}$ of the same length l, we introduce the notation $(\boldsymbol{c} = \boldsymbol{d}\ ?\ \boldsymbol{a} : \boldsymbol{b})$ to denote the sequence of length l whose i-th component is given by

$$(\boldsymbol{a} = \boldsymbol{b}\ ?\ \boldsymbol{c} : \boldsymbol{d})_i = \begin{cases} \boldsymbol{c}_i & \text{if } \boldsymbol{a}_i = \boldsymbol{b}_i, \\ \boldsymbol{d}_i & \text{otherwise,} \end{cases} \tag{2}$$

---

can replace the symbol $\perp$ in a message to be encrypted with any string of polynomial length other than the challenge ciphertext (say, z in the proof of Lemma 1, provided in appendix A), which does not disadvantage the simulator under the valid ciphertext condition.

with $i \in [l]$.[8] In this notation, a symbol x not in boldface is considered as the sequence $(x)^l$ of length l whose components are all x; e.g.,

$$(a = \boldsymbol{b}\ ?\ \boldsymbol{c} : d) = ((a)^l = \boldsymbol{b}\ ?\ \boldsymbol{c} : (d)^l).$$

A function ϵ from $\mathbb{N}$ to $\mathbb{R}$, $\epsilon : \mathbb{N} \to \mathbb{R}$, is called *negligible* if for all $c > 0$, there exists an integer n_c such that $\epsilon(n) \leq n^{-c}$ for all $n \geq n_c$.

A *public key encryption scheme* is a triple of algorithms, $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$, such that

- $\mathcal{K}$, the *key generation algorithm*, is a probabilistic, polynomial-time algorithm which takes as input a security parameter $k \in \mathbb{N}$ (in unary) and returns a pair (pk, sk) of matching public and secret keys,
- $\mathcal{E}$, the *encryption algorithm*, is a probabilistic, polynomial-time algorithm which takes as input a public key pk and a plaintext $x \in \{0,1\}^*$ and returns a ciphertext y,
- $\mathcal{D}$, the *decryption algorithm*, is a deterministic, polynomial-time algorithm which takes as input a secret key sk and a ciphertext y and returns either a plaintext $x \in \{0,1\}^*$ or a special symbol $\perp$ to indicate that the ciphertext is invalid,

where the correctness condition $\Pr[\mathcal{D}_{sk}(\mathcal{E}_{pk}(x)) = x] = 1$ has to hold for all $k \in \mathbb{N}$, for all (pk, sk) which can be output by $\mathcal{K}(1^k)$ and for all $x \in \{0,1\}^*$. In this paper, we assume that all algorithms have access to the key generation algorithm $\mathcal{K}(1^k)$ given the security parameter k.[9]

2.1 Formulations of non-malleability

We refer to section 1 for an informal explanation of the formulations of non-malleability, and here make several remarks not described there. In the simulation-based formulation of non-malleability, introduced in [11] and refined in [4], the first stage adversary A_1 outputs state information s_1 for A_2 and side information s_2 for R, in addition to a distribution M over plaintexts. Here, all plaintexts in the support of M are of the same length. This is because a ciphertext inevitably leaks information about the length of its plaintext (see e.g. [14]), and so without restriction on the length of plaintexts, all encryption schemes would be insecure. The relation R is computable in polynomial-time, and takes as input not only the plaintexts x and $\boldsymbol{x}$ but also M and s_2. The reason why R takes input M is for a fair comparison between the adversary A and its simulator S; since a simulator which outputs M consisting of a single plaintext can always be successful, all encryption schemes would be secure if R did not take input M. In contrast, it does not affect the strength of security whether or not R takes input s_2; the second stage adversary A_2 can input side information s_2 into R by concatenating

[8] The notations $\boldsymbol{x}_{:j}$ and $(\boldsymbol{c} = \boldsymbol{d}\ ?\ \boldsymbol{a} : \boldsymbol{b})$ are based on the notations for subarrays in numpy and for the conditional operator in programming languages, respectively.

[9] This is necessary in some proofs where a simulator (which is not explicitly given the security parameter k in our definition) runs $\mathcal{K}(1^k)$, and was also assumed e.g. in [4].

an encryption of s_2 to ciphertexts $\boldsymbol{y}$, as in the proof of SNM $\Longrightarrow$ IND in [4]. We also note that A_2 is prohibited from asking the challenge ciphertext y to the decryption oracle for CCA2. A formal definition of SNM* is described below.[10]

Definition 1 (SNM*-ATK [4, 11]). *Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be an encryption scheme and R be a relation. Let $A = (A_1, A_2)$ be an adversary attacking Π and $S = (S_1, S_2)$ be its simulator. For $k \in \mathbb{N}$ and* ATK $\in \{\mathrm{CPA}, \mathrm{CCA1}, \mathrm{CCA2}\}$*, define the advantage of A against S by*

$$\begin{aligned}&\mathsf{Adv}^{SNM^*\text{-}ATK}_{\Pi,R,A,S}(k)\\ &= \Pr[\mathsf{Expt}^{SNM^*\text{-}ATK\text{-}1}_{\Pi,R,A}(k) : w = 1] - \Pr[\mathsf{Expt}^{SNM^*\text{-}ATK\text{-}0}_{\Pi,R,S}(k) : w = 1],\end{aligned}$$

where

$\mathsf{Experiment}\ \mathsf{Expt}^{\mathrm{SNM^*\text{-}ATK\text{-}1}}_{\Pi,R,A}(k)$	$\mathsf{Experiment}\ \mathsf{Expt}^{\mathrm{SNM^*\text{-}ATK\text{-}0}}_{\Pi,R,S}(k)$
$(pk, sk) \leftarrow \mathcal{K}(1^k)$	$(pk, sk) \leftarrow \mathcal{K}(1^k)$
$(M, s_1, s_2) \leftarrow A_1^{\mathcal{O}_1}(pk)$	$(M, s_1, s_2) \leftarrow S_1(pk)$
$x \leftarrow M$; $y \leftarrow \mathcal{E}_{pk}(x)$	$x \leftarrow M$
$\boldsymbol{y} \leftarrow A_2^{\mathcal{O}_2}(s_1, y)$; $\boldsymbol{x} \leftarrow \mathcal{D}_{sk}(\boldsymbol{y})$	$\boldsymbol{y} \leftarrow S_2(s_1)$; $\boldsymbol{x} \leftarrow \mathcal{D}_{sk}(\boldsymbol{y})$
$\mathsf{if}\ R(x, \boldsymbol{x}, M, s_2) = 1 \wedge \perp \notin \boldsymbol{x}\ \mathsf{then}\ w \leftarrow 1$	$\mathsf{if}\ R(x, \boldsymbol{x}, M, s_2) = 1 \wedge \perp \notin \boldsymbol{x}\ \mathsf{then}\ w \leftarrow 1$
$\mathsf{else}\ w \leftarrow 0$	$\mathsf{else}\ w \leftarrow 0$

Here, $\mathcal{O}_1$ and $\mathcal{O}_2$ are oracles given by

$$\begin{aligned}
&\mathcal{O}_1 = \varepsilon(\cdot) \quad \text{and} \quad \mathcal{O}_2 = \varepsilon(\cdot) && \text{if ATK} = \mathrm{CPA},\\
&\mathcal{O}_1 = \mathcal{D}_{sk}(\cdot) \quad \text{and} \quad \mathcal{O}_2 = \varepsilon(\cdot) && \text{if ATK} = \mathrm{CCA1},\\
&\mathcal{O}_1 = \mathcal{D}_{sk}(\cdot) \quad \text{and} \quad \mathcal{O}_2 = \mathcal{D}_{sk}(\cdot) && \text{if ATK} = \mathrm{CCA2},
\end{aligned}$$

respectively, where $\varepsilon(\cdot)$ denotes the empty function which, on any input, outputs the empty string ε, and it is supposed that (i) all strings in the support of M are of the same length, (ii) $y \notin \boldsymbol{y}$ and (iii) $y \notin \boldsymbol{query}(A; \mathcal{O}_2)$ in the above experiment $\mathsf{Expt}^{SNM^*\text{-}ATK\text{-}1}_{\Pi,R,A}(k)$, where $\boldsymbol{query}(A; \mathcal{O}_2)$ denotes a sequence of queries from A to $\mathcal{O}_2$ for the case of ATK $=$ CCA2*. An adversary A is called* legitimate *if its outputs and queries satisfy the above conditions (i)–(iii). For a function f of k, an adversary A (resp. a simulator S) is called* bounded by time $f(k)$ *if A (resp. S) runs in time $f(k)$ and outputs M samplable in time $f(k)$. Then, an encryption scheme Π is called* secure in the sense of SNM*-ATK *if for all polynomial p, all probabilistic adversary A bounded by time $p(k)$ and all relation R computable in time $p(k)$, there exist a polynomial $p'(k)$ and a simulator S bounded by time $p'(k)$ such that $\mathsf{Adv}^{SNM^*\text{-}ATK}_{\Pi,R,A,S}(k)$ is negligible.*

In the comparison-based formulation introduced in [3], the first stage adversary A_1 outputs state information s_1 for A_2 and a distribution M over plaintexts.

[10] It is now common to take the absolute value in the definition of the advantage. The equivalence to this common definition for CNM* and IND* can be seen by considering adversaries outputting inversions $\bar{R}$ and $\bar{d}$, respectively. The same holds for SNM* if we may assume, e.g., the existence of an efficient algorithm to output an invalid ciphertext, but it seems not so trivial to show the (in)equivalence for SNM*.

Then, two plaintexts x_0 and x_1, which correspond to x' and x in section 1, respectively, are independently sampled according to M. The relation R output by A_2 is computable in polynomial-time. We note that in the following definition, $\mathsf{Expt}_{\Pi,A}^{\text{CNM}^*\text{-ATK-1}}$ (resp. $\mathsf{Expt}_{\Pi,A}^{\text{CNM}^*\text{-ATK-0}}$) denotes the experiment for an adversary (resp. "random guess"). A formal definition of CNM* is described below.

Definition 2 (CNM*-ATK [3]). *Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be an encryption scheme. Let $A = (A_1, A_2)$ be an adversary attacking Π. For $k \in \mathbb{N}$ and* ATK $\in \{$CPA, CCA1, CCA2$\}$*, define the advantage of A by*

$$\begin{aligned}&\mathsf{Adv}_{\Pi,A}^{CNM^*\text{-}ATK}(k)\\ &= \Pr[\mathsf{Expt}_{\Pi,A}^{CNM^*\text{-}ATK\text{-}1}(k) : w = 1] - \Pr[\mathsf{Expt}_{\Pi,A}^{CNM^*\text{-}ATK\text{-}0}(k) : w = 1],\end{aligned}$$

where, for $b \in \{0, 1\}$,

$$\begin{array}{l}\mathsf{Experiment}\ \mathsf{Expt}_{\Pi,A}^{\text{CNM}^*\text{-ATK-}b}(k)\\ \quad (pk, sk) \leftarrow \mathcal{K}(1^k);\ (M, s) \leftarrow A_1^{\mathcal{O}_1}(pk);\ x_0, x_1 \leftarrow M\\ \quad y \leftarrow \mathcal{E}_{pk}(x_1);\ (R, \boldsymbol{y}) \leftarrow A_2^{\mathcal{O}_2}(s, y);\ \boldsymbol{x} \leftarrow \mathcal{D}_{sk}(\boldsymbol{y})\\ \quad \mathsf{if}\ R(x_b, \boldsymbol{x}) = 1 \wedge \perp \notin \boldsymbol{x}\ \mathsf{then}\ w \leftarrow 1\\ \quad \mathsf{else}\ w \leftarrow 0\end{array}$$

Here, A is supposed to be legitimate as in Definition 1, and $\mathcal{O}_1$ and $\mathcal{O}_2$ are defined as in Definition 1. For a function f of k, an adversary A is called bounded by time $f(k)$ *if A runs in time $f(k)$ and outputs M samplable in time $f(k)$ and R computable in time $f(k)$. Then, an encryption scheme Π is called* secure in the sense of CNM*-ATK *if for all polynomial p and all probabilistic adversary A bounded by time $p(k)$, $\mathsf{Adv}_{\Pi,A}^{CNM^*\text{-}ATK}(k)$ is negligible.*

In the indistinguishability-based formulation of non-malleability introduced in [4], an adversary A is a triple of algorithms, $A = (A_1, A_2, A_3)$, corresponding to three stages of an attack. The first stage adversary A_1 takes as input the public key pk and outputs two plaintexts x_0 and x_1 such that $|x_0| = |x_1|$, together with state information s_1 for A_2. Next, one of the two plaintexts x_0 and x_1, say x_b ($b \in \{0, 1\}$), is chosen at random and then encrypted to give a challenge ciphertext y. The second stage adversary A_2 takes as input the challenge ciphertext y and the state information s_1 and outputs a sequence $\boldsymbol{y}$ of ciphertexts such that $y \notin \boldsymbol{y}$, together with state information s_2 for A_3. Then, $\boldsymbol{y}$ is decrypted to give $\boldsymbol{x}$. The third stage adversary A_3 takes as input the sequence $\boldsymbol{x}$ and the state information s_2 and outputs $d \in \{0, 1\}$, where A is considered successful if $d = b$ and $\perp \notin \boldsymbol{x}$ hold. Also, A is supposed to have access to the decryption oracle $\mathcal{D}_{sk}(\cdot)$ depending on the attack model PCAX; namely, A has no access to $\mathcal{D}_{sk}(\cdot)$ for PCA0, only A_1 has access to $\mathcal{D}_{sk}(\cdot)$ for PCA1 and all A_1, A_2 and A_3 have access to $\mathcal{D}_{sk}(\cdot)$ for PCA2, where A_2 and A_3 are prohibited from asking the challenge ciphertext y to $\mathcal{D}_{sk}(\cdot)$ for the last case. Note that queries to the decryption oracle are more powerful than parallel chosen ciphertext queries because (i) the former can be adaptive and (ii) the former including an invalid ciphertext does not result in failure of an adversary even under the valid ciphertext condition. A formal definition of IND*-PCAX is described below.

Definition 3 (IND*-PCAX [4]). *Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be an encryption scheme and $A = (A_1, A_2, A_3)$ be an adversary attacking Π. For $k \in \mathbb{N}$ and* PCAX $\in \{\mathrm{PCA0}, \mathrm{PCA1}, \mathrm{PCA2}\}$*, define the advantage of A by*

$$\mathsf{Adv}^{IND^*\text{-}PCAX}_{\Pi,A}(k) = 2\Pr[\mathsf{Expt}^{IND^*\text{-}PCAX}_{\Pi,A}(k) : w = 1] - 1,$$

where

$\mathsf{Experiment}\ \mathsf{Expt}^{\mathrm{IND}^*\text{-}\mathrm{PCAX}}_{\Pi,A}(k)$
$(pk, sk) \leftarrow \mathcal{K}(1^k)$; $(x_0, x_1, s_1) \leftarrow A_1^{\mathcal{O}_1}(pk)$; $b \leftarrow \{0,1\}$; $y \leftarrow \mathcal{E}_{pk}(x_b)$
$(\boldsymbol{y}, s_2) \leftarrow A_2^{\mathcal{O}_2}(x_0, x_1, s_1, y)$; $\boldsymbol{x} \leftarrow \mathcal{D}_{sk}(\boldsymbol{y})$; $d \leftarrow A_3^{\mathcal{O}_2}(\boldsymbol{x}, s_2)$
$\mathsf{if}\ d = b \wedge \perp \notin \boldsymbol{x}\ \mathsf{then}\ w \leftarrow 1$
$\mathsf{else}\ w \leftarrow 0$

Here, A is supposed to be legitimate as in Definition 1, and $\mathcal{O}_1$ and $\mathcal{O}_2$ are oracles given by

$$\begin{array}{llll} \mathcal{O}_1 = \varepsilon(\cdot) & \text{and} & \mathcal{O}_2 = \varepsilon(\cdot) & \textit{if PCAX} = \mathrm{PCA0}, \\ \mathcal{O}_1 = \mathcal{D}_{sk}(\cdot) & \text{and} & \mathcal{O}_2 = \varepsilon(\cdot) & \textit{if PCAX} = \mathrm{PCA1}, \\ \mathcal{O}_1 = \mathcal{D}_{sk}(\cdot) & \text{and} & \mathcal{O}_2 = \mathcal{D}_{sk}(\cdot) & \textit{if PCAX} = \mathrm{PCA2}, \end{array}$$

respectively, where $\varepsilon(\cdot)$ denotes the empty function as before. Then, an encryption scheme Π is called secure in the sense of IND*-PCAX *if for all polynomial p and all probabilistic adversary A runnable in time $p(k)$, $\mathsf{Adv}^{IND^*\text{-}PCAX}_{\Pi,A}(k)$ is negligible.*

3 Separation between simulation-based and comparison-based formulations

Let X and Y be security notions for encryption schemes. In order to show the separation $X \not\Longrightarrow Y$, it is necessary to show that there exists an encryption scheme which is secure in the sense of X but not secure in the sense of Y. Since the existence of computationally secure encryption schemes has not been proved, it is standard to show the separation by modifying an encryption scheme Π to another encryption scheme Π' so that if Π is X-secure, then Π' is still X-secure but not Y-secure.[11] The proofs in this paper follow this standard.

To prove the separation between SNM* and CNM*, we modify an encryption scheme $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ to $\Pi' = (\mathcal{K}', \mathcal{E}', \mathcal{D}')$ so that the modified decryption algorithm $\mathcal{D}'$ has an additional "option" which gives no (absolute) advantage to an adversary and a simulator. More precisely, $\mathcal{E}'$ takes a plaintext x and outputs a ciphertext $(0, \mathcal{E}_{pk}(x))$, and $\mathcal{D}'$ takes a ciphertext (a, y) and outputs $\mathcal{D}_{sk}(y)$ if $a = 0$ or x equals a specific string (say, 0), otherwise $\perp$. It can be seen from

[11] Since the existence of computationally secure private key encryption schemes is equivalent to that of one-way functions, we may show separations for private key encryption schemes by assuming the latter (see e.g. [19]).

this definition of $\mathcal{D}'$ that there is no advantage to choose the option $a \neq 0$. An SNM* simulator may not choose this option, while a CNM* adversary can force the "random guess" to choose the option so as to take relative advantage against it. We are now ready to show the separation between SNM* and CNM*.

Theorem 1. *SNM*-CCA1 $\not\Longrightarrow$ CNM*-CPA.*

Proof. Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be an encryption scheme. By using Π, let us construct another encryption scheme $\Pi' = (\mathcal{K}', \mathcal{E}', \mathcal{D}')$ as

Algorithm $\mathcal{K}'(1^k)$	Algorithm $\mathcal{E}'_{pk}(x)$	Algorithm $\mathcal{D}'_{sk}((a,y))$
$(pk, sk) \leftarrow \mathcal{K}(1^k)$	$y \leftarrow \mathcal{E}_{pk}(x)$	$x \leftarrow \mathcal{D}_{sk}(y)$
return (pk, sk)	return $(0, y)$	if $a = 0$ then return x
		else if $x = 0$ then return x
		else return $\perp$

It can be seen from this definition that for a and x such that $x \leftarrow \mathcal{D}'_{sk}((a,y))$ for some y,

$$x \neq \perp \wedge a \neq 0 \iff x \neq \perp \wedge a \neq 0 \wedge x = 0. \tag{3}$$

Then, the theorem follows from Lemmas 1 and 2, provided in appendix A. □

Lemmas 1 and 2 claim that (a) If Π is SNM*-CCA1, then so is Π' and (b) Π' is not CNM*-CPA, respectively. To show the first lemma (a), we construct an adversary A attacking Π and a relation R for A by using an adversary A' attacking Π' and a relation R' for A', respectively. The construction of A from A' is straightforward except for the case where A'_2 outputs a sequence $\boldsymbol{y}'$ of ciphertexts which contains a component (a', y) such that $a' \neq 0$ and y is a challenge ciphertext for A (we note that challenge ciphertexts for A' have the form $(0, y)$). In fact, A_2 can generate a sequence $\boldsymbol{y}$ of ciphertexts by simply ignoring the first component a' of each component (a', y') of $\boldsymbol{y}'$. On the other hand, in the exceptional case mentioned above, the sequence $\boldsymbol{y}$ generated as above contains the challenge ciphertext y, which violates the legitimate condition (ii) (see Definition 1 for this condition). This violation can be avoided as follows. Let $\boldsymbol{a}'$ be a sequence given by simply ignoring the second component y' of each component (a', y') of $\boldsymbol{y}'$. For $i \in [|\boldsymbol{a}'|]$, if $\boldsymbol{a}'_i \neq 0$, then A_2 replaces $\boldsymbol{y}_i$ with z such that $z \neq y$, and then concatenates an encryption $\mathcal{E}_{pk}(\boldsymbol{a}')$ of the position $\boldsymbol{a}'$ of this replacement to the sequence $\boldsymbol{y}$ of ciphertexts. Then, the relation R can replace $\mathcal{D}_{sk}(z)$ at the position of the above replacement with the specific string 0. Here, note that (3) implies that a ciphertext (a, y) with $a \neq 0$ is valid only if $x = \mathcal{D}_{sk}(y) = 0$. It thus follows that the advantage of A is no less than that of A'. Furthermore, we can show the second lemma (b) by considering a CNM*-CPA adversary which simply transforms a challenge ciphertext $(0, y)$ to $(1, y)$ and outputs it as a component of $\boldsymbol{y}$. In fact, if the adversary outputs the message distribution $M = \{0, 1\}$ and the relation R defined by $R(x, \boldsymbol{x}) = 1$ iff $\boldsymbol{x} = (x)$, then the adversary is successful if and only if $x_1 = 0$ (which occurs with probability $\frac{1}{2}$), while the "random guess" is successful if and only if $x_0 = x_1 = 0$

(which occurs with probability $\frac{1}{4}$), and so the former has the advantage $\frac{1}{4}$ against the latter. Detailed proofs of these lemmas are described in appendix A. We note that the above proof of Theorem 1 applies to a larger message space M if and only if its cardinality $|M|$ is upper-bounded by a polynomial of k.

4 Separation between semantic security and ciphertext indistinguishability

We begin with showing the separation between SNM* and IND*. For this purpose, we modify an encryption scheme $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ to $\Pi' = (\mathcal{K}', \mathcal{E}', \mathcal{D}')$ so that the modified decryption algorithm $\mathcal{D}'$ has an additional "option" which makes an adversary and a simulator fail with probability at least $\frac{1}{2}$. More precisely, $\mathcal{E}'$ takes a plaintext x and outputs a ciphertext $(0, \mathcal{E}_{pk}(ux))$ with u being a random bit, and $\mathcal{D}'$ takes a ciphertext (a, y) and outputs $\hat{x}$ if $a = 0$ or $\hat{u} = 0$, otherwise $\perp$, where we have introduced $\hat{u}$ and $\hat{x}$ to denote the first bit and the remaining bits of $\mathcal{D}_{sk}(y)$, respectively (i.e. $\mathcal{D}_{sk}(y) = \hat{u}\hat{x}$ with $|\hat{u}| = 1$). It can be seen from this definition of $\mathcal{D}'$ that an adversary and a simulator fail with probability at least $\Pr[\hat{u} = 1] = \frac{1}{2}$ if they choose the option $a \neq 0$. Hence, there is no advantage for an IND* adversary with a message distribution whose support consists of two elements x_0 and x_1 to choose this option, while an SNM* adversary may take advantage from this option by choosing a message distribution M whose support consists of more than two elements. We are now ready to show the separation between SNM* and IND*.

Theorem 2. *IND*-PCA1 $\not\Rightarrow$ SNM*-CPA.*

Proof. Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be an encryption scheme. By using Π, let us construct another encryption scheme $\Pi' = (\mathcal{K}', \mathcal{E}', \mathcal{D}')$ as

```
Algorithm K'(1^k)
  (pk, sk) ← K(1^k)
  return (pk, sk)

Algorithm E'_pk(x)
  u ← {0,1}
  y ← E_pk(ux)
  return (0, y)

Algorithm D'_sk((a, y))
  x' ← D_sk(y)
  if |x'| = 0 then return ⊥
  else parse x' as ux with |u| = 1
  if a = 0 then return x
  else if u = 0 then return x
       else return ⊥
```

Then, the theorem follows from Lemmas 3 and 4, provided in appendix B. □

Lemmas 3 and 4 claim that (a) If Π is IND*-PCA1, then so is Π' and (b) Π' is not SNM*-CPA, respectively. To show the first lemma (a), we construct an adversary A attacking Π by using an adversary A' attacking Π' as before. Again, the construction of A from A' is straightforward except for the case where A'_2 outputs a sequence $\boldsymbol{y}'$ of ciphertexts which contains a component (a', y) such that $a' \neq 0$ and y is a challenge ciphertext for A. Let us thus describe the construction for the exceptional case. Receiving two plaintexts x_0 and x_1 from A'_1, A_1 generates two plaintexts vx_0 and $\bar{v}x_1$ with v being a random bit, where $\bar{v}$ denotes the

inversion of v. Note that vx_0 and $\bar{v}x_1$ can be expressed as $vx_0 = (v \oplus 0)x_0$ and $\bar{v}x_1 = (v \oplus 1)x_1$, respectively, and the distributions of $(v \oplus b)x_b$ and ux_b are identical if b, v and u are independent random bits. If we consider v as a guess of b, then $(v \oplus b)x_b$ has the form $0x_b$ if the guess is correct (i.e. $v = b$), otherwise it has the form $1x_b$ and so $\mathcal{D}'_{sk}$ returns $\perp$. Now, A_2 generates a sequence $\boldsymbol{y}$ of ciphertexts by simply ignoring the first component a' of each component (a', y') of $\boldsymbol{y}'$. Next, A_2 replaces y in $\boldsymbol{y}$ with z such that $z \neq y$, and then includes the position of this replacement in the state information s_2 for A_3. Finally, A_3 replaces $\mathcal{D}_{sk}(z)$ at the position of the above replacement with x_v. It can be seen from this construction that A can completely simulate the view of A' if the guess is correct, while if the guess is not correct, then A' always fails because $\perp \in \mathcal{D}_{sk}(\boldsymbol{y}')$. It thus follows that the advantage of A' is upper-bounded by that of A. Furthermore, we can show the second lemma (b) by considering an SNM*-CPA adversary which simply transforms a challenge ciphertext $(0, y)$ to $(1, y)$ and outputs it as a component of $\boldsymbol{y}$. In fact, if the adversary outputs a message distribution whose support consists of more than two (say, $M = \{0,1\}^2$), then for the relation R given by $R(x, \boldsymbol{x}, M, s_2) = 1$ iff $M = \{0,1\}^2 \wedge x = \boldsymbol{x}_1$, the adversary is successful if and only if $u = 0$ (which occurs with probability $\frac{1}{2}$), while the simulator is successful with probability at most $\frac{1}{4}$, and so the former has the advantage at least $\frac{1}{4}$ against the latter. Detailed proofs of these lemmas are described in appendix B.

In the above proof, it is essential that an IND* adversary has to output a message distribution whose support consists of exactly two elements, but an SNM* adversary is free of such restriction on a message distribution. This may motivate us to consider simulation-based and comparison-based formulations of semantic security against parallel chosen-ciphertext attacks, SSS*-PCAX and CSS*-PCAX, because semantic security is commonly formulated without such restriction on a message distribution, and so may be (potentially) stronger than IND*-PCAX. We first provide a formulation of SSS*-PCAX, which is just a combination of the definitions of semantic security [16] and parallel chosen-ciphertext attacks [4], and leave a formulation of CSS*-PCAX in appendix C.

We refer to section 1 for an informal explanation of the formulations of semantic security and to section 2.1 for remarks on the related formulations, and here make a few remarks. In the formulation of SSS*-PCAX, the function F is computable in polynomial-time, and takes as input not only the plaintext x but also M and s_3. As before, the reason why F takes input M is for a fair comparison between the adversary A and its simulator S. We use that F can take input side information s_3 in the proof of Proposition 1, provided in appendix C. The third stage adversary A_3 and its simulator S_3 output v and side information s_3 for F, where A and S are considered successful if $F(x, M, s_3) = v$ and $\perp \notin \boldsymbol{x}$ hold. A formal definition of SSS*-PCAX is described below.

Definition 4 (SSS*-PCAX). *Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be an encryption scheme and F be a function. Let $A = (A_1, A_2, A_3)$ be an adversary attacking Π and $S = (S_1, S_2, S_3)$ be its simulator. For $k \in \mathbb{N}$ and* $\mathrm{PCAX} \in \{\mathrm{PCA0}, \mathrm{PCA1}, \mathrm{PCA2}\}$,

define the advantage of A against S by

$$\mathsf{Adv}^{SSS^*\text{-}PCAX}_{\Pi,F,A,S}(k) = \Pr[\mathsf{Expt}^{SSS^*\text{-}PCAX\text{-}1}_{\Pi,F,A}(k) : w = 1] - \Pr[\mathsf{Expt}^{SSS^*\text{-}PCAX\text{-}0}_{\Pi,F,S}(k) : w = 1],$$

where

Experiment $\mathsf{Expt}^{\mathrm{SSS^*\text{-}PCAX\text{-}1}}_{\Pi,F,A}(k)$	Experiment $\mathsf{Expt}^{\mathrm{SSS^*\text{-}PCAX\text{-}0}}_{\Pi,F,S}(k)$
$(pk, sk) \leftarrow \mathcal{K}(1^k)$	$(pk, sk) \leftarrow \mathcal{K}(1^k)$
$(M, s_1) \leftarrow A_1^{\mathcal{O}_1}(pk)$	$(M, s_1) \leftarrow S_1(pk)$
$x \leftarrow M$; $y \leftarrow \mathcal{E}_{pk}(x)$	$x \leftarrow M$
$(\boldsymbol{y}, s_2) \leftarrow A_2^{\mathcal{O}_2}(s_1, y)$; $\boldsymbol{x} \leftarrow \mathcal{D}_{sk}(\boldsymbol{y})$	$(\boldsymbol{y}, s_2) \leftarrow S_2(s_1)$; $\boldsymbol{x} \leftarrow \mathcal{D}_{sk}(\boldsymbol{y})$
$(v, s_3) \leftarrow A_3^{\mathcal{O}_2}(\boldsymbol{x}, s_2)$	$(v, s_3) \leftarrow S_3(\boldsymbol{x}, s_2)$
if $F(x, M, s_3) = v \wedge \perp \notin \boldsymbol{x}$ then $w \leftarrow 1$	if $F(x, M, s_3) = v \wedge \perp \notin \boldsymbol{x}$ then $w \leftarrow 1$
else $w \leftarrow 0$	else $w \leftarrow 0$

Here, A is supposed to be legitimate as in Definition 1, and $\mathcal{O}_1$ and $\mathcal{O}_2$ are defined as in Definition 3. Then, an encryption scheme Π is called secure in the sense of SSS*-PCAX *if for all polynomial p, all probabilistic adversary A bounded by time $p(k)$ and all function F computable in time $p(k)$, there exist a polynomial $p'(k)$ and a simulator S bounded by time $p'(k)$ such that $\mathsf{Adv}^{SSS^*\text{-}ATK}_{\Pi,R,A,S}(k)$ is negligible.*

It turns out that SSS*-PCAX and CSS*-PCAX are equivalent to SNM*-ATK and CNM*-ATK, respectively (see appendix C). Hence, it follows from this equivalence, together with Theorem 2, that (simulation-based) semantic security and ciphertext indistinguishability separate against the weaker parallel chosen-ciphertext attacks under the valid ciphertext condition.

Corollary 1. *IND*-PCA1 $\not\Longrightarrow$ SSS*-PCA0.*

5 Concluding remarks

It may be natural to consider the possibility that the results of this work extend to the private key setting. For this extension, it is necessary that a simulator S has access to the encryption oracle (see the proof of Lemma 1, provided in appendix A). It can be seen from the construction of S which runs the key generation algorithm $\mathcal{K}$ (see e.g. the proof of CNM $\Longrightarrow$ SNM in [4]) that the strength of security does not change even if S has access to the decryption oracle. Hence, it would be of interest to consider the strength and validity of the simulation-based non-malleability in which S has access to the encryption oracle in the private key setting. Furthermore, the proof of SSS* $\Longrightarrow$ SNM* (see Proposition 1, provided in appendix C) requires that the function F in semantic security takes as input side information s_3. We note that it does not affect the strength of security whether or not the relation R in non-malleability takes as input side information s_2 (see e.g. the proof of SNM $\Longrightarrow$ IND in [4]). Hence, it would also be of interest to consider the strength of the simulation-based semantic security in which F is not given s_2.

Acknowledgements

The author is grateful to reviewers for helpful comments.

References

1. Akavia, A., Gentry, C., Halevi, S., Vald, M.: Achievable cca2 relaxation for homomorphic encryption. In: TCC. pp. 70–99 (2022)
2. Bellare, M., Desai, A., Jokipii, E., Rogaway, P.: A concrete security treatment of symmetric encryption: Analysis of the des modes of operation. In: FOCS. pp. 394–403 (1997)
3. Bellare, M., Desai, A., Pointcheval, D., Rogaway, P.: Relations among notions of security for public-key encryption schemes. In: CRYPTO. pp. 26–45 (1998), https://eprint.iacr.org/1998/021
4. Bellare, M., Sahai, A.: Non-malleable encryption: Equivalence between two notions, and an indistinguishability-based characterization. In: CRYPTO. pp. 519–536 (1999), Full version available at https://eprint.iacr.org/2006/228
5. Bos, J., Ducas, L., Kiltz, E., Lepoint, T., Lyubashevsky, V., Schanck, J.M., Schwabe, P., Seiler, G., Stehle, D.: Crystals - kyber: A cca-secure module-lattice-based kem. In: EuroS&P. pp. 353–367 (2018). https://doi.org/10.1109/EuroSP.2018.00032
6. Canetti, R., Varia, M.: Non-malleable obfuscation. In: TCC. pp. 73–90 (2009)
7. Coretti, S., Dodis, Y., Tackmann, B., Venturi, D.: Non-malleable encryption: Simpler, shorter, stronger. In: TCC. pp. 306–335 (2016)
8. Coretti, S., Maurer, U., Tackmann, B., Venturi, D.: From single-bit to multi-bit public-key encryption via non-malleable codes. In: TCC. pp. 532–560 (2015)
9. Di Crescenzo, G., Ishai, Y., Ostrovsky, R.: Non-interactive and non-malleable commitment. In: STOC. pp. 141–150 (1998)
10. Dodis, Y., Halevi, S., Wichs, D.: Security with functional re-encryption from cpa. In: Rothblum, G., Wee, H. (eds.) TCC. pp. 279–305 (2023)
11. Dolev, D., Dwork, C., Naor, M.: Non-malleable cryptography. SIAM Journal on Computing **30**(2), 391–437 (2000)
12. Dziembowski, S., Pietrzak, K., Wichs, D.: Non-malleable codes. In: TCC. pp. 434–452 (2010), https://eprint.iacr.org/2009/608
13. Goldreich, O.: A uniform-complexity treatment of encryption and zero-knowledge. Journal of Cryptology **6**(1), 21–53 (1993)
14. Goldreich, O.: Foundations of Cryptography: Volume 2, Basic Applications. Cambridge University Press (2009)
15. Goldreich, O., Lustig, Y., Naor, M.: On chosen ciphertext security of multiple encryptions. Cryptology ePrint Archive, Paper 2002/089 (2002), https://eprint.iacr.org/2002/089
16. Goldwasser, S., Micali, S.: Probabilistic encryption. Journal of Computer and System Sciences **28**(2), 270–299 (1984)
17. Goyal, V., Kumar, A.: Non-malleable secret sharing. In: STOC. pp. 685–698 (2018), https://eprint.iacr.org/2018/316
18. Herranz, J., Hofheinz, D., Kiltz, E.: Some (in)sufficient conditions for secure hybrid encryption. Information and Computation **208**(11), 1243–1257 (2010), Full version available at https://eprint.iacr.org/2006/265
19. Katz, J., Yung, M.: Characterization of security notions for probabilistic private-key encryption. Journal of Cryptology **19**(1), 67–95 (2006)

20. Lindell, Y.: How to Simulate It – A Tutorial on the Simulation Proof Technique, pp. 277–346. Springer (2017)
21. Micali, S., Rackoff, C., Sloan, B.: The notion of security for probabilistic cryptosystems. SIAM Journal on Computing **17**(2), 412–426 (1988)
22. Naor, M., Yung, M.: Public-key cryptosystems provably secure against chosen ciphertext attacks. In: STOC. pp. 427–437 (1990)
23. Pass, R., shelat, a., Vaikuntanathan, V.: Construction of a non-malleable encryption scheme from any semantically secure one. In: CRYPTO. pp. 271–289 (2006)
24. Pass, R., Shelat, A., Vaikuntanathan, V.: Relations among notions of non-malleability for encryption. In: ASIACRYPT. pp. 519–535 (2007)
25. Rackoff, C., Simon, D.R.: Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attack. In: CRYPTO. pp. 433–444 (1991)
26. Sahai, A.: Non-malleable non-interactive zero knowledge and adaptive chosen-ciphertext security. In: FOCS. pp. 543–553 (1999)
27. Watanabe, Y., Shikata, J., Imai, H.: Equivalence between semantic security and indistinguishability against chosen ciphertext attacks. In: PKC. pp. 71–84 (2002)

A Proofs of lemmas for Theorem 1

Lemma 1. *If Π is SNM*-CCA1, then so is Π'.*

Proof. Let p be a polynomial of k. Let R' be a relation computable in time $p(k)$ and $A' = (A'_1, A'_2)$ be a legitimate SNM*-CCA1 adversary attacking Π', bounded by time $p(k)$ (see Definition 1 for an adversary *bounded by time* $p(k)$). By using A' and R', let us construct an SNM*-CCA1 adversary $A = (A_1, A_2)$ attacking Π and a relation R as

Algorithm $A_1^{\mathcal{D}_{sk}}(pk)$	Algorithm $A_2(s'_1, y)$	Relation $R(x, \hat{\boldsymbol{x}}, M, s_2)$
$(M, s_1, s_2) \leftarrow A_1'^{\mathcal{D}'_{sk}}(pk)$	$\boldsymbol{y}' \leftarrow A'_2(s_1, (0, y))$	if $\lvert\hat{\boldsymbol{x}}\rvert$ is odd then return 0
$x' \leftarrow M; L \leftarrow \lvert x'\rvert + 1$	$\boldsymbol{a}' \leftarrow \boldsymbol{y}'_{:1}$	parse $\hat{\boldsymbol{x}}$ as $\boldsymbol{x}\lvert\lvert\boldsymbol{a}'$ with $\lvert\boldsymbol{x}\rvert = \lvert\boldsymbol{a}'\rvert$
$z \leftarrow \mathcal{E}_{pk}(0^L)$	$\boldsymbol{y} \leftarrow (\boldsymbol{a}' = 0 \ ? \ \boldsymbol{y}'_{:2} : z)$	$\tilde{\boldsymbol{x}} \leftarrow (\boldsymbol{a}' = 0 \ ? \ \boldsymbol{x} : 0)$
$s'_1 \leftarrow (s_1, z, pk)$	$\hat{\boldsymbol{y}} \leftarrow \boldsymbol{y}\lvert\lvert\mathcal{E}_{pk}(\boldsymbol{a}')$	return $R'(x, \tilde{\boldsymbol{x}}, M, s_2)$
return (M, s'_1, s_2)	return $\hat{\boldsymbol{y}}$	

(see (1) and (2) for the notations $\boldsymbol{x}_{:j}$ and $(\boldsymbol{c} = \boldsymbol{d} \ ? \ \boldsymbol{a} : \boldsymbol{b})$, respectively), where the length L is chosen so that $\lvert 0^L\rvert > \lvert x\rvert$ for any output x of M (note that M outputs messages of a fixed length), which ensures that $\mathcal{E}_{pk}(0^L) \neq y$ with probability 1. Since A' is bounded by time $p(k)$ (and so every string output by A' has a length bounded by $p(k)$), R' is computable in time $p(k)$ and $\mathcal{E}_{pk}$ is polynomial-time, it follows that M is samplable in time $p(k)$ and A and R are polynomial-time. Moreover, A can be seen legitimate as follows: the condition (i) follows from that A' is legitimate and the condition (iii) from that A_2 has no oracle access to $\mathcal{D}_{sk}$; since $(0, y) \notin \boldsymbol{y}'$ and so $\forall i((\boldsymbol{y}'_{:2})_i = y \implies \boldsymbol{a}'_i \neq 0)$, we have $y \notin \boldsymbol{y} = (\boldsymbol{a}' = 0 \ ? \ \boldsymbol{y}'_{:2} : z)$, from which the condition (ii) follows. We note that A_1 can answer queries from A'_1 by using her own oracle $\mathcal{D}_{sk}$ to compute $\mathcal{D}'_{sk}$.

It is now convenient to consider the experiment $\mathsf{Expt}_1(k)$ defined by

$\mathsf{Experiment}\ \mathsf{Expt}_1(k)$
$(pk, sk) \leftarrow \mathcal{K}(1^k)$; $(M, s_1) \leftarrow A_1^{\prime \mathcal{D}'_{sk}}(pk)$; $x, x' \leftarrow M$; $y \leftarrow \mathcal{E}_{pk}(x)$
$L \leftarrow |x'| + 1$; $z \leftarrow \mathcal{E}_{pk}(0^L)$; $(\boldsymbol{y}', s_2) \leftarrow A'_2(s_1, (0, y))$
$\boldsymbol{a}' \leftarrow \boldsymbol{y}'_{:1}$; $\boldsymbol{y} \leftarrow (\boldsymbol{a}' = 0\ ?\ \boldsymbol{y}'_{:2} : z)$; $\boldsymbol{x}' \leftarrow \mathcal{D}'_{sk}(\boldsymbol{y}')$; $\boldsymbol{x} \leftarrow \mathcal{D}_{sk}(\boldsymbol{y})$; $\tilde{\boldsymbol{x}} \leftarrow (\boldsymbol{a}' = 0\ ?\ \boldsymbol{x} : 0)$

and to introduce, for an event E, the short-hand notation $p_1(E) = \Pr[\mathsf{Expt}_1(k) : E]$. Since $\boldsymbol{x} = (\boldsymbol{a}' = 0\ ?\ \boldsymbol{x}' : 0^L)$ and $\tilde{\boldsymbol{x}} = (\boldsymbol{a}' = 0\ ?\ \boldsymbol{x} : 0)$, we have

$$\perp \in \boldsymbol{x} \iff \perp \in \tilde{\boldsymbol{x}} \implies \perp \in \boldsymbol{x}'. \tag{4}$$

Moreover, since $\tilde{\boldsymbol{x}}$ can be written as $\tilde{\boldsymbol{x}} = (\boldsymbol{a}' = 0\ ?\ \boldsymbol{x}' : 0)$, we have $\boldsymbol{x}' = \tilde{\boldsymbol{x}} \iff \forall i(\boldsymbol{a}'_i = 0 \vee \boldsymbol{x}'_i = 0)$. It thus follows from (3) that

$$\begin{aligned}
\perp \notin \boldsymbol{x}' &\iff \forall i(\boldsymbol{x}'_i \neq \perp \wedge (\boldsymbol{a}'_i = 0 \vee \boldsymbol{a}'_i \neq 0)) \\
&\iff \forall i(\boldsymbol{x}'_i \neq \perp \wedge (\boldsymbol{a}'_i = 0 \vee (\boldsymbol{a}'_i \neq 0 \wedge \boldsymbol{x}'_i = 0))) \\
&\iff \forall i(\boldsymbol{x}'_i \neq \perp \wedge (\boldsymbol{a}'_i = 0 \vee \boldsymbol{x}'_i = 0)) \\
&\iff \perp \notin \boldsymbol{x}' \wedge \boldsymbol{x}' = \tilde{\boldsymbol{x}}.
\end{aligned}$$

Therefore,

$$\begin{aligned}
&\Pr[\mathsf{Expt}^{\text{SNM}^*\text{-CCA1-1}}_{\Pi,R,A}(k) : w = 1] \\
&= p_1(R(x, \boldsymbol{x}, M, (s_2)||\boldsymbol{a}') = 1 \wedge \perp \notin \boldsymbol{x}) \\
&= p_1(R'(x, \tilde{\boldsymbol{x}}, M, s_2) = 1 \wedge \perp \notin \tilde{\boldsymbol{x}}) \\
&\geq p_1(R'(x, \tilde{\boldsymbol{x}}, M, s_2) = 1 \wedge \perp \notin \boldsymbol{x}') \\
&= p_1(R'(x, \tilde{\boldsymbol{x}}, M, s_2) = 1 \wedge \perp \notin \boldsymbol{x}' \wedge \boldsymbol{x}' = \tilde{\boldsymbol{x}}) \\
&= p_1(R'(x, \boldsymbol{x}', M, s_2) = 1 \wedge \perp \notin \boldsymbol{x}' \wedge \boldsymbol{x}' = \tilde{\boldsymbol{x}}) \\
&= p_1(R'(x, \boldsymbol{x}', M, s_2) = 1 \wedge \perp \notin \boldsymbol{x}') \\
&= \Pr[\mathsf{Expt}^{\text{SNM}^*\text{-CCA1-1}}_{\Pi',R',A'}(k) : w = 1],
\end{aligned}$$

where the inequality follows from (4).

It follows from Definition 1 that if Π is secure in the sense of SNM*-CCA1, then there exist a polynomial p' and a simulator $S = (S_1, S_2)$ of the above adversary A, bounded by time $p'(k)$, such that $\mathsf{Adv}^{\text{SNM}^*\text{-CCA1}}_{\Pi,R,A,S}(k)$ is negligible. By using such S, let us next construct a simulator $S' = (S'_1, S'_2)$ of A' as[12]

$\mathsf{Algorithm}\ S'_1(pk')$
$(pk, sk) \leftarrow \mathcal{K}(1^k)$; $(M, s_1, s_2) \leftarrow S_1(pk)$
$\hat{\boldsymbol{y}} \leftarrow S_2(s_1)$; $\hat{\boldsymbol{x}} \leftarrow \mathcal{D}_{sk}(\hat{\boldsymbol{y}})$
$\mathsf{if}\ |\hat{\boldsymbol{x}}|\ \mathsf{is\ odd\ then\ return}\ (M, ((), \varepsilon))$
$\mathsf{parse}\ \hat{\boldsymbol{x}}\ \mathsf{as}\ \boldsymbol{x}||\boldsymbol{a}'\ \mathsf{with}\ |\boldsymbol{x}| = |\boldsymbol{a}'|$
$\tilde{\boldsymbol{x}} \leftarrow (\boldsymbol{a}' = 0\ ?\ \boldsymbol{x} : 0)$; $\boldsymbol{y}' \leftarrow \mathcal{E}_{pk'}(\tilde{\boldsymbol{x}})$
$\mathsf{return}\ (M, \boldsymbol{y}')$

$\mathsf{Algorithm}\ S'_2(\boldsymbol{y}')$
$\mathsf{return}\ \boldsymbol{y}'$

[12] It can be seen from this construction that the strength of security does not change even if S is given the secret key/decryption oracle, and S can be one stage. We note that the construction of S which runs $\mathcal{K}$ was given in [4].

Since S is bounded by time $p'(k)$ and $\mathcal{K}$, $\mathcal{E}_{pk'}$ and $\mathcal{D}_{sk}$ are polynomial-time, it follows that M is samplable in time $p'(k)$ and S' is polynomial-time. It can also be seen from the above construction of S' and R that

$$\Pr[\mathsf{Expt}^{\text{SNM}^*\text{-CCA1-0}}_{\Pi',R',S'}(k) : w = 1] \geq \Pr[\mathsf{Expt}^{\text{SNM}^*\text{-CCA1-0}}_{\Pi,R,S}(k) : w = 1]$$

(where equality holds if and only if S' always fails when $|\hat{\boldsymbol{x}}|$ is odd), and so

$$\mathsf{Adv}^{\text{SNM}^*\text{-CCA1}}_{\Pi',R',A',S'}(k) \leq \mathsf{Adv}^{\text{SNM}^*\text{-CCA1}}_{\Pi,R,A,S}(k).$$

Consequently, if Π is secure in the sense of SNM-CCA1, then $\mathsf{Adv}^{\text{SNM}^*\text{-CCA1}}_{\Pi,R,A,S}(k)$ is negligible, and so is $\mathsf{Adv}^{\text{SNM-CCA1}}_{\Pi',R',A',S'}(k)$. This completes the proof. □

Lemma 2. *Π' is not CNM*-CPA.*

Proof. Let $A = (A_1, A_2)$ be a CNM*-CPA adversary attacking Π' defined by

$\mathsf{Algorithm}\ A_1(pk)$	$\mathsf{Algorithm}\ A_2(s,(0,y))$
$\quad\mathsf{return}\ (\{0,1\},\varepsilon)$	$\quad\mathsf{return}\ (R,((1,y)))$

where the relation R output by A_2 is given by

$\mathsf{Relation}\ R(x,\boldsymbol{x})$
$\quad\mathsf{if}\ \boldsymbol{x} = (x)\ \mathsf{then\ return}\ 1$
$\quad\mathsf{else\ return}\ 0$

It can be seen from this definition that M is samplable in time $O(1)$ and A is polynomial-time; moreover, since $|0| = |1|$ and $(0,y) \neq (1,y)$, A is legitimate. Since $\perp \notin \boldsymbol{x} \iff x_1 = 0$, it also follows that

$$\Pr[\mathsf{Expt}^{\text{CNM}^*\text{-ATK-1}}_{\Pi,A}(k) : w = 1] = \Pr[\mathsf{Expt}^{\text{CNM}^*\text{-ATK-1}}_{\Pi,A}(k) : x_1 = x_1 \wedge x_1 = 0] = \frac{1}{2},$$
$$\Pr[\mathsf{Expt}^{\text{CNM}^*\text{-ATK-0}}_{\Pi,A}(k) : w = 1] = \Pr[\mathsf{Expt}^{\text{CNM}^*\text{-ATK-0}}_{\Pi,A}(k) : x_0 = x_1 \wedge x_1 = 0] = \frac{1}{4},$$

and so

$$\mathsf{Adv}^{\text{CNM}^*\text{-ATK}}_{\Pi,A}(k) = \frac{1}{2} - \frac{1}{4} = \frac{1}{4},$$

which is not negligible. This completes the proof. □

B Proofs of lemmas for Theorem 2

Lemma 3. *If Π is IND*-PCA1, then so is Π'.*

Proof. Let p be a polynomial of k. Let $A' = (A'_1, A'_2, A'_3)$ be a legitimate IND*-PCA1 adversary attacking Π', bounded by time $p(k)$. By using A', let us construct an IND*-PCA1 adversary $A = (A_1, A_2, A_3)$ attacking Π as

$\mathsf{Algorithm}\ A_1^{\mathcal{D}_{sk}}(pk)$
$(x_0, x_1, s_1) \leftarrow A_1'^{\mathcal{D}'_{sk}}(pk)$
$v \leftarrow \{0,1\}$; $L \leftarrow |x_0| + 1$; $z \leftarrow \mathcal{E}_{pk}(0^L)$
$\mathsf{return}\ (vx_0, \bar{v}x_1, (s_1, z, x_v))$

$\mathsf{Algorithm}\ A_2(vx_0, \bar{v}x_1, (s_1, z, x_v), y)$
$(\boldsymbol{y}', s_2) \leftarrow A_2'(s_1, (0, y))$
$\boldsymbol{y} \leftarrow (\boldsymbol{y}'_{:2} = y\ ?\ z : \boldsymbol{y}'_{:2})$
$\mathsf{return}\ (\boldsymbol{y}, (s_2, \boldsymbol{y}'_{:2}, y, x_v))$

$\mathsf{Algorithm}\ A_3(\boldsymbol{x}, (s_2, \boldsymbol{y}'_{:2}, y, x_v))$
$\tilde{\boldsymbol{x}} \leftarrow (\boldsymbol{y}'_{:2} = y\ ?\ x_v : \boldsymbol{x})$; $d \leftarrow A_3'(\tilde{\boldsymbol{x}}, s_2)$
$\mathsf{return}\ d$

where $\bar{v}$ denotes the inversion of v. Since A' is bounded by time $p(k)$ and $\mathcal{E}_{pk}$ is polynomial-time, it follows that M is samplable in time $p(k)$ and A and R are also polynomial-time. Moreover, A can be seen legitimate as follows: the condition (i) follows from that A' is legitimate, the condition (ii) from that $\boldsymbol{y} = (\boldsymbol{y}'_{:2} = y\ ?\ z : \boldsymbol{y}'_{:2})$, where every component y has been replaced by z, and the condition (iii) from that A_2 has no oracle access to $\mathcal{D}_{sk}$. We note that A_1 can answer queries from A_1' by using her own oracle $\mathcal{D}_{sk}$ to compute $\mathcal{D}'_{sk}$.

It is now convenient to consider the experiment $\mathsf{Expt}_2(k)$ defined by

$\mathsf{Experiment}\ \mathsf{Expt}_2(k)$
$(pk, sk) \leftarrow \mathcal{K}(1^k)$; $(x_0, x_1, s_1) \leftarrow A_1'^{\mathcal{D}'_{sk}}(pk)$; $L \leftarrow |x_0| + 1$; $z \leftarrow \mathcal{E}_{pk}(0^L)$
$b, u, v \leftarrow \{0,1\}$; $y \leftarrow \mathcal{E}_{pk}((v \oplus b)x_b)$; $\hat{y} \leftarrow \mathcal{E}_{pk}(ux_b)$
$\boldsymbol{y}' \leftarrow A_2'(s_1, (0, y))$; $\boldsymbol{x}' \leftarrow \mathcal{D}'_{sk}(\boldsymbol{y}')$; $d' \leftarrow A_3'(\boldsymbol{x}', s_2)$
$\hat{\boldsymbol{y}} \leftarrow A_2'(s_1, (0, \hat{y}))$; $\hat{\boldsymbol{x}} \leftarrow \mathcal{D}'_{sk}(\hat{\boldsymbol{y}})$; $\hat{d} \leftarrow A_3'(\hat{\boldsymbol{x}}, s_2)$
$\boldsymbol{y} \leftarrow (\boldsymbol{y}'_{:2} = y\ ?\ z : \boldsymbol{y}'_{:2})$; $\boldsymbol{x} \leftarrow \mathcal{D}_{sk}(\boldsymbol{y})$; $\tilde{\boldsymbol{x}} \leftarrow (\boldsymbol{y}'_{:2} = y\ ?\ x_v : \boldsymbol{x})$; $d \leftarrow A_3'(\tilde{\boldsymbol{x}}, s_2)$

where we have used that vx_0 and $\bar{v}x_1$ can be expressed as $vx_0 = (v \oplus 0)x_0$ and $\bar{v}x_1 = (v \oplus 1)x_1$, respectively, and to introduce the short-hand notation $p_2(E) = \Pr[\mathsf{Expt}_2(k) : E]$, as before. We first note that the distributions of $(v \oplus b)x_b$ and ux_b are identical. Since $\boldsymbol{x} = (\boldsymbol{y}'_{:2} = y\ ?\ 0^L : \boldsymbol{x}')$, we have

$$\perp \in \boldsymbol{x} \implies \perp \in \boldsymbol{x}'.$$

It thus follows that

$$\begin{aligned}
\Pr[\mathsf{Expt}^{\mathrm{IND}^*\text{-PCA1}}_{\Pi,A}(k) : w = 1] &= p_2(d = b \wedge \perp \notin \boldsymbol{x}) \\
&\geq p_2(d = b \wedge \perp \notin \boldsymbol{x}') \\
&= p_2(d = b \wedge \perp \notin \boldsymbol{x}' \wedge v = b) \\
&\quad + p_2(d = b \wedge \perp \notin \boldsymbol{x}' \wedge v \neq b \wedge y \in \boldsymbol{y}'_{:2}) \\
&\quad + p_2(d = b \wedge \perp \notin \boldsymbol{x}' \wedge v \neq b \wedge y \notin \boldsymbol{y}'_{:2}).
\end{aligned}$$

We now estimate the above three terms in the right-hand side. To consider the first term, we begin with expressing $\boldsymbol{x}$ as $\boldsymbol{x} = (\boldsymbol{y}'_{:2} = y\ ?\ 0^L : \boldsymbol{x}')$. It can be seen from this expression that $\tilde{\boldsymbol{x}} = (\boldsymbol{y}'_{:2} = y\ ?\ x_v : \boldsymbol{x}) = (\boldsymbol{y}'_{:2} = y\ ?\ x_v : \boldsymbol{x}')$, and so

$$v = b \implies \tilde{\boldsymbol{x}} = (\boldsymbol{y}'_{:2} = y\ ?\ x_b : \boldsymbol{x}') = \boldsymbol{x}' \implies d = d'.$$

Hence, on noting that $v = b \iff v \oplus b = 0$ and $p_2(v \oplus b = 0) = p_2(u = 0) = \frac{1}{2}$, we have

$$\begin{aligned} p_2(d = b \wedge \bot \notin \boldsymbol{x}' \wedge v = b) &= p_2(d' = b \wedge \bot \notin \boldsymbol{x}' \wedge v = b) \\ &= p_2(d' = b \wedge \bot \notin \boldsymbol{x}' | v \oplus b = 0) p_2(v \oplus b = 0) \\ &= p_2(\hat{d} = b \wedge \bot \notin \hat{\boldsymbol{x}} | u = 0) p_2(u = 0) \\ &= p_2(\hat{d} = b \wedge \bot \notin \hat{\boldsymbol{x}} \wedge u = 0). \end{aligned}$$

To consider the second term, suppose that $y \in \boldsymbol{y}'_{:2}$, and let i be an index such that $\boldsymbol{y}'_i = (a, y)$. Then, since A' is legitimate, we have $(0, y) \notin \boldsymbol{y}'$ and so $a \neq 0$. Note here that $v \neq b \iff v \oplus b = 1$, and hence $\mathcal{D}_{sk}(y) = 1x_b$. It thus follows from the definition of $\mathcal{D}'_{sk}$ that $\boldsymbol{x}'_i = \mathcal{D}'_{sk}((a, y)) = \bot$. Similarly, if $\hat{y} \in \hat{\boldsymbol{y}}_{:2}$ and $u \neq 0$, then $\bot \in \hat{\boldsymbol{x}}$. Therefore,

$$p_2(d = b \wedge \bot \notin \boldsymbol{x}' \wedge v \neq b \wedge y \in \boldsymbol{y}'_{:2}) = p_2(\hat{d} = b \wedge \bot \notin \hat{\boldsymbol{x}} \wedge u \neq 0 \wedge \hat{y} \in \hat{\boldsymbol{y}}_{:2}) = 0.$$

To consider the third term, we begin with

$$y \notin \boldsymbol{y}'_{:2} \implies \tilde{\boldsymbol{x}} = (\boldsymbol{y}'_{:2} = y \ ? \ x_v : \boldsymbol{x}') = \boldsymbol{x}' \implies d = d'.$$

Hence, on noting that $v \neq b \iff v \oplus b = 1$, $u \neq 0 \iff u = 1$ and $p_2(v \oplus b = 1) = p_2(u = 1) = \frac{1}{2}$, we have

$$\begin{aligned} &p_2(d = b \wedge \bot \notin \boldsymbol{x}' \wedge v \neq b \wedge y \notin \boldsymbol{y}'_{:2}) \\ &= p_2(d' = b \wedge \bot \notin \boldsymbol{x}' \wedge v \neq b \wedge y \notin \boldsymbol{y}'_{:2}) \\ &= p_2(d' = b \wedge \bot \notin \boldsymbol{x}' \wedge y \notin \boldsymbol{y}'_{:2} | v \oplus b = 1) p_2(v \oplus b = 1) \\ &= p_2(\hat{d} = b \wedge \bot \notin \hat{\boldsymbol{x}} \wedge \hat{y} \notin \hat{\boldsymbol{y}}_{:2} | u = 1) p_2(u = 1) \\ &= p_2(\hat{d} = b \wedge \bot \notin \hat{\boldsymbol{x}} \wedge u \neq 0 \wedge \hat{y} \notin \hat{\boldsymbol{y}}_{:2}). \end{aligned}$$

Having estimated the three terms, we now combine these terms to give

$$\begin{aligned} \Pr[\mathsf{Expt}^{\text{IND}^*\text{-PCA1}}_{\Pi,A}(k) : w = 1] &\geq p_2(d = b \wedge \bot \notin \boldsymbol{x}') \\ &= p_2(\hat{d} = b \wedge \bot \notin \hat{\boldsymbol{x}}) \\ &= \Pr[\mathsf{Expt}^{\text{IND}^*\text{-PCA1}}_{\Pi',A'}(k) : w = 1], \end{aligned}$$

and hence

$$\mathsf{Adv}^{\text{IND}^*\text{-PCA1}}_{\Pi',A'}(k) \leq \mathsf{Adv}^{\text{IND}^*\text{-PCA1}}_{\Pi,A}(k).$$

Consequently, if Π is secure in the sense of IND*-PCA1, then $\mathsf{Adv}^{\text{IND}^*\text{-PCA1}}_{\Pi,A}(k)$ is negligible, and so is $\mathsf{Adv}^{\text{IND}^*\text{-PCA1}}_{\Pi',A'}(k)$. This completes the proof. □

Lemma 4. *Π' is not SNM*-CPA.*

Proof. Let $A = (A_1, A_2)$ be an SNM*-CPA adversary attacking Π' defined by

Algorithm $A_1(pk)$	Algorithm $A_2(s_1, (0, y))$
return $(\{0,1\}^2, \varepsilon, \varepsilon)$	return $((1, y))$

and let R be a relation defined by

Relation $R(x, \boldsymbol{x}, M, s_2)$
  if $M = \{0,1\}^2 \wedge x = \boldsymbol{x}_1$ then return 1
  else return 0

It can be seen from the above definition of A that M is samplable in time $O(1)$ and A is polynomial-time; it also follows from $|00| = |01| = |10| = |11|$ and $(0, y) \neq (1, y)$ that A is legitimate. Now, it follows from the construction of A that

$$\Pr[\mathsf{Expt}^{\text{SNM*-ATK-1}}_{\Pi',R,A}(k) : w = 1] = \Pr[\mathsf{Expt}^{\text{SNM*-ATK-1}}_{\Pi',R,A}(k) : u = 0] = \frac{1}{2},$$

where u is the random variable introduced in the definition of $\mathcal{E}'_{pk}$. On the other hand, S is given no information about the plaintext x, and hence the outputs from S are statistically independent of x. Consequently, since x is uniformly distributed on $\{0,1\}^2$, we have

$$\begin{aligned}\Pr[\mathsf{Expt}^{\text{SNM*-ATK-0}}_{\Pi',R,S}(k) : w = 1] &= \Pr[\mathsf{Expt}^{\text{SNM*-ATK-0}}_{\Pi',R,S}(k) : x = \boldsymbol{x}_1] \\ &\leq \frac{1}{|\{0,1\}^2|} = \frac{1}{4}\end{aligned}$$

(where equality holds if and only if S outputs $M = \{0,1\}^2$ and $\boldsymbol{y}$ such that $\mathcal{D}_{sk'}(\boldsymbol{y}_1) \in \{0,1\}^2$), and so

$$\mathsf{Adv}^{\text{SNM*-ATK}}_{\Pi',R,A,S}(k) \geq \frac{1}{4},$$

which is not negligible. This completes the proof. □

C Semantic security against parallel chosen-ciphertext attacks

In this appendix, we first provide a formal definition of comparison-based semantic security against parallel chosen-ciphertext attacks under the valid ciphertext condition, CSS*-PCAX, which is just a combination of the definitions of semantic security [16] and parallel chosen-ciphertext attacks [4], with reference to the comparison-based formulation of semantic security for the private key encryption [2] (see Definition 4 for the definition of the simulation-based semantic security SSS*-PCAX).

Definition 5 (CSS*-PCAX). *Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be an encryption scheme and $A = (A_1, A_2, A_3)$ be an adversary attacking Π. For $k \in \mathbb{N}$ and* PCAX $\in \{\mathrm{PCA0}, \mathrm{PCA1}, \mathrm{PCA2}\}$*, define the advantage of A by*

$$\mathsf{Adv}^{CSS^*\text{-}PCAX}_{\Pi,A}(k) = \Pr[\mathsf{Expt}^{CSS^*\text{-}PCAX\text{-}1}_{\Pi,A}(k) : w = 1] - \Pr[\mathsf{Expt}^{CSS^*\text{-}PCAX\text{-}0}_{\Pi,A}(k) : w = 1],$$

where

$\mathsf{Experiment}\ \mathsf{Expt}^{\mathrm{CSS^*\text{-}PCAX}\text{-}b}_{\Pi,A}(k)$
$(pk, sk) \leftarrow \mathcal{K}(1^k)$; $(M, s_1) \leftarrow A_1^{\mathcal{O}_1}(pk)$; $x_0, x_1 \leftarrow M$; $y \leftarrow \mathcal{E}_{pk}(x_1)$
$(\boldsymbol{y}, s_2) \leftarrow A_2^{\mathcal{O}_2}(s_1, y)$; $\boldsymbol{x} \leftarrow \mathcal{D}_{sk}(\boldsymbol{y})$; $(f, v) \leftarrow A_3^{\mathcal{O}_2}(\boldsymbol{x}, s_2)$
$\mathsf{if}\ f(x_b) = v \wedge \perp \notin \boldsymbol{x}\ \mathsf{then}\ w \leftarrow 1$
$\mathsf{else}\ w \leftarrow 0$

Here, A is supposed to be legitimate as in Definition 1, and $\mathcal{O}_1$ and $\mathcal{O}_2$ are defined as in Definition 3. Then, an encryption scheme Π is called secure in the sense of CSS*-PCAX *if for all polynomial p and all probabilistic adversary A runnable in time $p(k)$, $\mathsf{Adv}^{CSS^*\text{-}PCAX}_{\Pi,A}(k)$ is negligible.*

Again, note that in the above definition, $\mathsf{Expt}^{\mathrm{CSS^*\text{-}PCAX}\text{-}1}_{\Pi,A}$ (resp. $\mathsf{Expt}^{\mathrm{CSS^*\text{-}PCAX}\text{-}0}_{\Pi,A}$) denotes the experiment for an adversary (resp. "random guess").

Having provided formal definitions, we next show that SSS*-PCAX and CSS*-PCAX are equivalent to SNM*-ATK and CNM*-ATK, respectively. Again, the proof is rather straightforward, and we may refer to the proof of the equivalence among SNM, CNM and IND in [4]; for example, in order for relation R to run a probabilistic algorithm $A(x_1, \ldots; r)$, one can include the randomness r for the algorithm A in the side information s_2 for the relation R. A detailed proof of the equivalence is described below. (Here, it should be noted that the equivalence for (PCA2,CCA2) follows from the equivalence between IND-CCA2 and CNM-CCA2 [3].)

Proposition 1. *SSS*-PCAX $\iff$ SNM*-ATK and CSS*-PCAX $\iff$ CNM*-ATK for* (PCAX, ATK)$\in \{$(PCA0, CPA), (PCA1, CCA1), (PCA2, CCA2)$\}$.

Proof. (I) SSS*-PCAX $\Longrightarrow$ SNM*-ATK: Let p be a polynomial of k. Let R' be a relation computable in time $p(k)$ and $A' = (A'_1, A'_2)$ be a legitimate SNM*-ATK adversary attacking an encryption scheme $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$, bounded by time $p(k)$. By using A' and R', let us construct an SSS*-PCAX adversary $A = (A_1, A_2, A_3)$ attacking Π and a function F as

$\mathsf{Algorithm}\ A_1^{\mathcal{O}_1}(pk)$	$\mathsf{Algorithm}\ A_2^{\mathcal{O}_2}((s_1, s_2), y)$	$\mathsf{Algorithm}\ A_3^{\mathcal{O}_2}(s_2, \boldsymbol{x})$
$(M, s_1, s_2) \leftarrow A_1'^{\mathcal{O}_1}(pk)$	$\boldsymbol{y} \leftarrow A_2'^{\mathcal{O}_2}(s_1, y)$	$\boldsymbol{s} \leftarrow \boldsymbol{x}\|\|(s_2)$
$\mathsf{return}\ (M, (s_1, s_2))$	$\mathsf{return}\ (\boldsymbol{y}, s_2)$	$\mathsf{return}\ (1, \boldsymbol{s})$

$\mathsf{Function}\ F(x, M, \boldsymbol{s})$
$\mathsf{if}\ |\boldsymbol{s}| = 0\ \mathsf{then\ return}\ 0$
$\mathsf{else\ parse}\ \boldsymbol{s}\ \mathsf{as}\ \boldsymbol{x}||(s)\ \mathsf{with}\ |(s)| = 1$
$\mathsf{return}\ R'(x, \boldsymbol{x}, M, s)$

Since A' is bounded by time $p(k)$ and R' is computable in time $p(k)$, it follows that M is samplable in time $p(k)$ and A and F are also polynomial-time. Moreover, since A' is legitimate, A is also legitimate. We note that A can answer queries from A' by using her own oracle. It is now straightforward to see from the above construction of A and F that

$$\Pr[\mathsf{Expt}^{\text{SNM}^*\text{-ATK-1}}_{\Pi,R',A'}(k) : w = 1] = \Pr[\mathsf{Expt}^{\text{SSS}^*\text{-ATK-1}}_{\Pi,F,A}(k) : w = 1].$$

It follows from Definition 4 that if Π is secure in the sense of SSS*-PCAX, then there exist a polynomial p' and a simulator $S = (S_1, S_2, S_3)$ of the above adversary A, bounded by time $p'(k)$, such that $\mathsf{Adv}^{\text{SSS}^*\text{-PCAX}}_{\Pi,F,A,S}(k)$ is negligible. By using such S, let us next construct a simulator $S' = (S'_1, S'_2)$ of A' as

$\mathsf{Algorithm}\ S'_1(pk')$
$(pk, sk) \leftarrow \mathcal{K}(1^k)$; $(M, s_1) \leftarrow S_1(pk)$
$(\boldsymbol{y}, s_2) \leftarrow S_2(s_1)$; $\boldsymbol{x} \leftarrow \mathcal{D}_{sk}(\boldsymbol{y})$; $(v, \boldsymbol{s}) \leftarrow S_3(\boldsymbol{x}, s_2)$
$\mathsf{if}\ |\boldsymbol{s}| = 0\ \mathsf{then\ return}\ (M, ((), \varepsilon))$
$\mathsf{else\ parse}\ \boldsymbol{s}\ \mathsf{as}\ \boldsymbol{x}'||(s)\ \mathsf{with}\ |(s)| = 1$
$\boldsymbol{y}' \leftarrow \mathcal{E}_{pk'}(\boldsymbol{x}')$
$\mathsf{return}\ (M, \boldsymbol{y}', s)$

$\mathsf{Algorithm}\ S'_2(\boldsymbol{y}')$
$\mathsf{return}\ \boldsymbol{y}'$

Since S is bounded by time $p'(k)$ and $\mathcal{K}$, $\mathcal{E}_{pk'}$ and $\mathcal{D}_{sk}$ are polynomial-time, it follows that M is samplable in time $p'(k)$ and S' is also polynomial-time. Then, the above construction of S' and F gives that

$$\Pr[\mathsf{Expt}^{\text{SNM}^*\text{-ATK-0}}_{\Pi,R',S'}(k) : w = 1] \geq \Pr[\mathsf{Expt}^{\text{SSS}^*\text{-PCAX-0}}_{\Pi,F,S}(k) : w = 1]$$

(where equality holds if and only if S' always fails when $|\boldsymbol{s}| = 0$), and so

$$\mathsf{Adv}^{\text{SNM}^*\text{-ATK}}_{\Pi,R',A',S'}(k) \leq \mathsf{Adv}^{\text{SSS}^*\text{-PCAX}}_{\Pi,F,A,S}(k).$$

Consequently, if Π is secure in the sense of SSS*-PCAX, then $\mathsf{Adv}^{\text{SSS}^*\text{-PCAX}}_{\Pi,F,A,S}(k)$ is negligible, and so is $\mathsf{Adv}^{\text{SNM}^*\text{-ATK}}_{\Pi,R',A',S'}(k)$. This completes the proof of (I).

(II) SNM*-ATK $\Longrightarrow$ SSS*-PCAX: Let p be a polynomial of k. Let F' be a function computable in time $p(k)$ and $A' = (A'_1, A'_2, A'_3)$ be a legitimate SSS*-PCAX adversary attacking an encryption scheme $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$, bounded by time $p(k)$. By using A' and F', let us construct an SSS*-ATK adversary $A = (A_1, A_2)$ attacking Π and a relation R as[13]

$\mathsf{Algorithm}\ A_1^{\mathcal{D}_{sk}(\cdot)}(pk)$
$(M, s_1) \leftarrow A_1'^{\mathcal{D}_{sk}(\cdot)}(pk)$
$\mathsf{return}\ (M, s_1, \varepsilon)$

$\mathsf{Algorithm}\ A_2^{\mathcal{D}_{sk}(\cdot)}(s_1, y)$
$(\boldsymbol{y}, s_2) \leftarrow A_2'^{\mathcal{D}_{sk}(\cdot)}(s_1, y)$; $\boldsymbol{x} \leftarrow \mathcal{D}_{sk}(\boldsymbol{y})$
$(v, s_3) \leftarrow A_3'^{\mathcal{D}_{sk}(\cdot)}(s_2, \boldsymbol{x})$; $\boldsymbol{y}' \leftarrow \mathcal{E}_{pk}((v, s_3))$
$\mathsf{return}\ \boldsymbol{y}'$

[13] In this R, $\boldsymbol{x}'$ is parsed as $\boldsymbol{x}||(v)||(s_3)$ (which always gives $\boldsymbol{x} = ()$ for the output from A) for consistency with the weaker attack models.

Relation $R(x, \boldsymbol{x}', M, s_2)$
  if $|\boldsymbol{x}'| < 2$ then return 0
  else parse $\boldsymbol{x}'$ as $\boldsymbol{x}||(v)||(s_3)$ with $|(v)| = |(s_3)| = 1$
  if $F'(x, M, s_3) = v$ then return 1
  else return 0

for $(\mathrm{PCAX}, \mathrm{ATK}) = (\mathrm{PCA2}, \mathrm{CCA2})$, otherwise as

Algorithm $A_1^{\mathcal{O}_1}(pk)$
  $(M, s_1) \leftarrow A_1'^{\mathcal{O}_1}(pk)$
  return (M, s_1, ε)

Algorithm $A_2(s_1, y)$
  $(\boldsymbol{y}, s_2) \leftarrow A_2'(s_1, y)$
  choose randomness r for A_3'
  $\boldsymbol{y}' \leftarrow \boldsymbol{y}||\mathcal{E}_{pk}((r, s_2))$
  return $\boldsymbol{y}'$

Relation $R(x, \boldsymbol{x}', M, s_2)$
  if $|\boldsymbol{x}'| < 2$ then return 0
  else parse $\boldsymbol{x}'$ as $\boldsymbol{x}||(r)||(s_2)$ with $|(r)| = |(s_2)| = 1$
  $(v, s_3) \leftarrow A_3'(s_2, \boldsymbol{x}; r)$
  if $F'(x, M, s_3) = v$ then return 1
  else return 0

Since A' is bounded by time $p(k)$, F' is computable in time $p(k)$ and $\mathcal{D}_{sk}$ is polynomial-time, it follows that M is samplable in time $p(k)$ and A and R are also polynomial-time. Moreover, since A' is legitimate, A is also legitimate. We note that A can answer queries from A' by using her own oracle. It is now straightforward to see from the above construction of A and R that

$$\Pr[\mathsf{Expt}^{\mathrm{SSS^*\text{-}ATK\text{-}1}}_{\Pi,F',A'}(k) : w = 1] = \Pr[\mathsf{Expt}^{\mathrm{SNM^*\text{-}ATK\text{-}1}}_{\Pi,R,A}(k) : w = 1].$$

It follows from Definition 1 that if Π is secure in the sense of SNM*-ATK, then there exist a polynomial p' and a simulator $S = (S_1, S_2)$ of the above adversary A, bounded by time $p'(k)$, such that $\mathsf{Adv}^{\mathrm{SNM^*\text{-}ATK}}_{\Pi,R,A,S}(k)$ is negligible. By using such S, let us next construct a simulator $S' = (S_1', S_2', S_3')$ of A' as

Algorithm $S_1'(pk')$
  $(pk, sk) \leftarrow \mathcal{K}(1^k)$; $(M, s_1, s_2) \leftarrow S_1(pk)$
  $\boldsymbol{y}' \leftarrow S_2(s_1)$; $\boldsymbol{x}' \leftarrow \mathcal{D}_{sk}(\boldsymbol{y}')$
  if $|\boldsymbol{x}'| < 2$ then return $(M, ((), \varepsilon, \varepsilon))$
  else parse $\boldsymbol{x}'$ as $\boldsymbol{x}||(v)||(s_3)$ with $|(v)| = |(s_3)| = 1$
  $\boldsymbol{y} \leftarrow \mathcal{E}_{pk'}(\boldsymbol{x})$
  return $(M, (\boldsymbol{y}, v, s_3))$

Algorithm $S_2'((\boldsymbol{y}, v, s_3))$
  return $(\boldsymbol{y}, (v, s_3))$

Algorithm $S_3'((v, s_3), \boldsymbol{x})$
  return (v, s_3)

for $(\mathrm{PCAX}, \mathrm{ATK}) = (\mathrm{PCA2}, \mathrm{CCA2})$, otherwise as

$\mathsf{Algorithm}\ S'_1(pk')$
$(pk, sk) \leftarrow \mathcal{K}(1^k)$; $(M, s_1, s_2) \leftarrow S_1(pk)$
$\boldsymbol{y}' \leftarrow S_2(s_1)$; $\boldsymbol{x}' \leftarrow \mathcal{D}_{sk}(\boldsymbol{y}')$
$\mathsf{if}\ |\boldsymbol{x}'| < 2\ \mathsf{then\ return}\ (M, ((), \varepsilon, \varepsilon))$
$\mathsf{else\ parse}\ \boldsymbol{x}'\ \mathsf{as}\ \boldsymbol{x}||(r)||(s_2)\ \ \mathsf{with}\ |(r)| = |(s_2)| = 1$
$(v, s_3) \leftarrow A'_3(s_2, \boldsymbol{x}; r)$; $\boldsymbol{y} \leftarrow \mathcal{E}_{pk'}(\boldsymbol{x})$
$\mathsf{return}\ (M, (\boldsymbol{y}, v, s_3))$

$\mathsf{Algorithm}\ S'_2((\boldsymbol{y}, v, s_3))$
$\mathsf{return}\ (\boldsymbol{y}, (v, s_3))$

$\mathsf{Algorithm}\ S'_3((v, s_3), \boldsymbol{x})$
$\mathsf{return}\ (v, s_3)$

Since S is bounded by time $p'(k)$ and $\mathcal{K}$, $\mathcal{E}_{pk'}$ and $\mathcal{D}_{sk}$ are polynomial-time, it follows that M is samplable in time $p'(k)$ and S' is also polynomial-time. Then, the above construction of S' and R gives that

$$\Pr[\mathsf{Expt}^{\text{SSS}^*\text{-PCAX-0}}_{\Pi,F',S'}(k) : w = 1] \geq \Pr[\mathsf{Expt}^{\text{SNM}^*\text{-ATK-0}}_{\Pi,R,S}(k) : w = 1]$$

(where equality holds if and only if S' always fails when $|\boldsymbol{x}'| < 2$), and so

$$\mathsf{Adv}^{\text{SSS}^*\text{-PCAX}}_{\Pi,F',A',S'}(k) \leq \mathsf{Adv}^{\text{SNM}^*\text{-ATK}}_{\Pi,R,A,S}(k).$$

Consequently, if Π is secure in the sense of SNM*-ATK, then $\mathsf{Adv}^{\text{SNM}^*\text{-ATK}}_{\Pi,R,A,S}(k)$ is negligible, and so is $\mathsf{Adv}^{\text{SSS}^*\text{-PCAX}}_{\Pi,F',A',S'}(k)$. This completes the proof of (II).

(III) CSS*-PCAX $\Longrightarrow$ CNM*-ATK: Let p be a polynomial of k. Let $A' = (A'_1, A'_2)$ be a legitimate CNM*-ATK adversary attacking an encryption scheme $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$, bounded by time $p(k)$. By using A', let us construct an CSS*-PCAX adversary $A = (A_1, A_2, A_3)$ attacking Π as

$\mathsf{Algorithm}\ A_1^{\mathcal{O}_1}(pk)$
$(M, s_1) \leftarrow A_1'^{\mathcal{O}_1}(pk)$
$\mathsf{return}\ (M, s_1)$

$\mathsf{Algorithm}\ A_2^{\mathcal{O}_2}(s_1, y)$
$(\boldsymbol{y}, R) \leftarrow A_2'^{\mathcal{O}_2}(s_1, y)$
$\mathsf{return}\ (\boldsymbol{y}, R)$

$\mathsf{Algorithm}\ A_3^{\mathcal{O}_2}(R, \boldsymbol{x})$
$\mathsf{return}\ (F_{R,\boldsymbol{x}}, 1)$

where the function $F_{R,\boldsymbol{x}}$ output by A_3 is given by

$\mathsf{Function}\ F_{R,\boldsymbol{x}}(x)$
$\mathsf{return}\ R(x, \boldsymbol{x})$

Since A' is bounded by time $p(k)$, it follows that M is samplable in time $p(k)$ and A is also polynomial-time. Moreover, since A' is legitimate, A is also legitimate. We note that A can answer queries from A' by using her own oracle. It is now straightforward to see from the above construction of A that

$$\mathsf{Adv}^{\text{CNM}^*\text{-ATK}}_{\Pi,A}(k) = \mathsf{Adv}^{\text{CSS}^*\text{-PCAX}}_{\Pi,A'}(k).$$

Consequently, if Π is secure in the sense of CSS*-PCAX, then $\mathsf{Adv}^{\text{CSS}^*\text{-PCAX}}_{\Pi,A}(k)$ is negligible, and so is $\mathsf{Adv}^{\text{CNM}^*\text{-ATK}}_{\Pi,A'}(k)$. This completes the proof of (III).

(IV) CNM*-ATK $\Longrightarrow$ CSS*-PCAX: Let p be a polynomial of k. Let $A' = (A'_1, A'_2, A'_3)$ be a legitimate CSS*-PCAX adversary attacking an encryption scheme $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$, bounded by time $p(k)$. By using A', let us construct an SSS*-ATK adversary $A = (A_1, A_2)$ attacking Π as

```
Algorithm A_1^{O_1}(pk)
  (M, s_1) ← A_1'^{O_1}(pk)
  return (M, s_1)

Algorithm A_2(s_1, y)
  (y, s_2) ← A_2'(s_1, y)
  choose randomness r for A_3'
  return (y, R_{r,s_2})
```

where the relation R_{r,s_2} output by A_2 is given by

```
Relation R_{r,s_2}(x, x)
  (f, v) ← A_3'(s_2, x; r)
  if f(x) = v then return 1
  else return 0
```

Since A' is bounded by time $p(k)$, it follows that M is samplable in time $p(k)$ and A is also polynomial-time. Moreover, since A' is legitimate, A is also legitimate. We note that A can answer queries from A' by using her own oracle. It is now straightforward to see from the above construction of A and R that

$$\mathsf{Adv}^{\text{CSS}^*\text{-PCAX}}_{\Pi,A'}(k) = \mathsf{Adv}^{\text{CNM}^*\text{-ATK}}_{\Pi,A}(k).$$

Consequently, if Π is secure in the sense of CNM*-ATK, then $\mathsf{Adv}^{\text{CNM}^*\text{-ATK}}_{\Pi,A}(k)$ is negligible, and so is $\mathsf{Adv}^{\text{CSS}^*\text{-PCAX}}_{\Pi,A'}(k)$. This completes the proof of (IV), and the proposition follows. □

D Indistinguishability-based formulation equivalent to comparison-based non-malleability

In the private-key setting, a slightly modified indistinguishability-based formulation of non-malleability, denoted as IND†-PCAX in this paper, was introduced and shown to be equivalent to CNM*-ATK [19]. The difference between IND*-PCAX and IND†-PCAX is that an IND* adversary always fails if $\perp \in \boldsymbol{x}$, while the success of an IND† adversary is determined at random if $\perp \in \boldsymbol{x}$; namely, if $\perp \in \boldsymbol{x}$, then an IND† adversary succeeds with probability $\frac{1}{2}$ and fails with the same probability. A formal definition of IND†-PCAX is described below.

Definition 6 (IND†-PCAX [19]). *Let $\Pi = (\mathcal{K}, \mathcal{E}, \mathcal{D})$ be an encryption scheme and $A = (A_1, A_2)$ be an adversary attacking Π. For $k \in \mathbb{N}$ and* PCAX $\in \{\text{PCA0}, \text{PCA1}, \text{PCA2}\}$*, consider define the advantage of A by*

$$\mathsf{Adv}^{IND^\dagger\text{-}PCAX}_{\Pi,A}(k) = 2\Pr[\mathsf{Expt}^{IND^\dagger\text{-}PCAX}_{\Pi,A}(k) : w = 1] - 1,$$

where

```
Experiment Expt^{IND†-PCAX}_{Π,A}(k)
  (pk, sk) ← K(1^k); (x_0, x_1, s_1) ← A_1^{O_1}(pk); b ← {0,1}; y ← E_{pk}(x_b)
  (y, s_2) ← A_2^{O_2}(x_0, x_1, s_1, y); x ← D_{sk}(y); d ← A_3^{O_2}(x, s_2)
  if ⊥ ∈ x then w ← {0,1}
  else if d = b then w ← 1
       else w ← 0
```

Here, A is supposed to be legitimate as in Definition 1, and $\mathcal{O}_1$ and $\mathcal{O}_2$ are defined as in Definition 3. Then, an encryption scheme Π is called secure in the sense of IND†-PCAX *if for all polynomial p and all probabilistic adversary A runnable in time $p(k)$, $\mathsf{Adv}^{IND^\dagger\text{-}PCAX}_{\Pi,A}(k)$ is negligible.*

The proof of the equivalence between CNM* and IND† for the private key setting given in [19] straightforwardly applies to the public key setting, yielding the following proposition.

Proposition 2. *IND†-PCAX $\Longleftrightarrow$ CNM*-ATK for* $(\mathrm{PCAX}, \mathrm{ATK}) \in \{(\mathrm{PCA0}, \mathrm{CPA}), (\mathrm{PCA1}, \mathrm{CCA1}), (\mathrm{PCA2}, \mathrm{CCA2})\}$.