

Non-malleability under Selective Opening Attacks: Implication and Separation

Zhengan Huang¹, Shengli Liu¹, Xianping Mao¹, and Kefei Chen^{2,3}

1. Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China

2. School of Science, Hangzhou Normal University, Hangzhou 310036, China

3. State Key Laboratory of Mathematical Engineering and Advanced Computing, Wuxi 214000, China

zhahuang.sjtu@gmail.com, {slliu, maexp, kfchen}@sjtu.edu.cn^{*}

Abstract. We formalize the security notions of non-malleability under selective opening attacks (NM-SO security) in two approaches: the indistinguishability-based approach and the simulation-based approach. We explore the relations between NM-SO security notions and the known selective opening security notions, and the relations between NM-SO security notions and the standard non-malleability notions.

Keywords: public-key encryption, non-malleability, selective opening attack.

1 Introduction

Non-malleability. The basic goal of public-key encryption (PKE) schemes is to guarantee the privacy of messages. The universally accepted formalization for this is semantic security proposed in [10], which requires that it be infeasible to learn any useful information of the message from the ciphertext. However, some cryptographic applications in a complex setting suggest that non-malleability is necessary. Non-malleability (NM), introduced by Dolev, Dwork and Naor [8] in 1991, requires that given a challenge ciphertext, it be infeasible to generate ciphertexts whose decryptions are related to the decryption of the challenge ciphertext. Nowadays, two main kinds of formalizations (indistinguishability-based [5] and simulation-based [8]) of non-malleability are widely accepted, especially the first one. (Actually, there is another formalization of non-malleability, comparison-based non-malleability [1][5].) Similar to semantic security, the formal security definitions of indistinguishability-based non-malleability (IND-NM) and simulation-based non-malleability (SIM-NM) consider all the three kinds of standard attacks: chosen-plaintext attacks (CPA), non-adaptive chosen-ciphertext attacks (CCA1) [18] and adaptive chosen-ciphertext attacks (CCA2) [20][8][9]. The combination of SIM-NM, IND-NM and CPA, CCA1, CCA2 gives six specific security notions (e.g., IND-NM-CPA security). The relations among these six security notions were figured out in [5][19].

Selective opening attacks. In Eurocrypt 2009, Bellare et al. [4] introduced the notion of selective opening security (SOA security) for sender corruptions. Roughly speaking, selective opening attack (for sender corruptions) is as follows: n senders encrypt their own messages with the public key of a single receiver. The adversary can corrupt some of these senders, by opening their ciphertexts, i.e., obtaining their messages and the random coins which were used during the encryption. The goal of SOA security is to guarantee the privacy of the unopened messages.

^{*} An extended abstract of this paper will be published in the proceedings of the 13th International Conference on Applied Cryptography and Network Security (ACNS 2015). This is the full version.

In [4], Bellare et al. presented two SOA security notions, the indistinguishability-based one (IND-SO) and the simulation-based one (SIM-SO). Later, Hemenway et al. [14] introduced the notions of IND-SO-CCA1/CCA2 security and SIM-SO-CCA1/CCA2 security. Over the years, several PKE schemes were proposed and proved to possess SOA security [11][14][12][15]. The relations between IND-SO-CPA security and SIM-SO-CPA security were clarified by Böhl et al. [3]. Bellare et al. [2] separated IND-CPA (even IND-CCA2) and SIM-SO-CPA security. Recently, Hofheinz and Rupp [17] showed a separation between IND-CCA2 and IND-SO-CCA2 security, and a “partial” equivalence between IND-CPA and IND-SO-CPA security.

To the best of our knowledge, how to formalize non-malleability under selective opening attacks remains elusive. Very recently, Hofheinz and Rupp referred to “NM-SO-CPA security” in [17]. But they did not present any formal definition.

Our contributions. This paper focuses on security notions and their relations. We first formalize the notion of simulation-based non-malleability under selective opening attacks (SIM-NM-SO), and the notion of indistinguishability-based non-malleability under selective opening attacks (IND-NM-SO). We figure out the relations among SIM-NM-SO-CPA(/CCA1/CCA2) security, IND-NM-SO-CPA(/CCA1/CCA2) security, SIM/IND-SO-CPA(/CCA1/CCA2) security and non-malleability security SIM/IND-NM-CPA(/CCA1/CCA2). Specifically, our results are as follows (see Figure 1). Below, we use $\text{SEC1} \Rightarrow \text{SEC2}$ to indicate that SEC1 implies SEC2, and $\text{SEC1} \not\Rightarrow \text{SEC2}$ to indicate the existence of some PKE scheme achieving SEC1 but not SEC2, for any two security notions SEC1 and SEC2.

1. *NM-SO versus SO*:
 - (a) *Simulation-based* (Section 4):
 - i. “SIM-NM-SO-ATK $\stackrel{\Rightarrow}{\not\Leftarrow}$ SIM-SO-ATK”, for any $\text{ATK} \in \{\text{CPA}, \text{CCA1}, \text{CCA2}\}$.
 - ii. For those PKE schemes having an invertible decryption algorithm (Definition 8), if the range of its decryption algorithm is recognizable, “SIM-SO-CCA2 \Leftrightarrow SIM-NM-SO-CCA2”.
 - (b) *Indistinguishability-based* (Section 5):
 - i. “IND-NM-SO-CPA $\stackrel{\Leftarrow}{\not\Rightarrow}$ IND-SO-CCA1”.
 - ii. “IND-NM-SO-CCA1/CPA $\stackrel{\Rightarrow}{\not\Leftarrow}$ IND-SO-CCA1/CPA”, but “IND-NM-SO-CCA2 \Leftrightarrow IND-SO-CCA2”.
2. *NM-SO versus NM*:
 - (a) *Simulation-based* (Section 6):
 - i. “SIM-NM-SO-ATK $\stackrel{\Rightarrow}{\not\Leftarrow}$ SIM-NM-ATK”, for any $\text{ATK} \in \{\text{CPA}, \text{CCA1}, \text{CCA2}\}$. In fact, we have a stronger result: “SIM-NM-CCA2 $\not\Rightarrow$ SIM-NM-SO-CPA”, which suggests “SIM-NM-ATK' $\not\Rightarrow$ SIM-NM-SO-ATK''”, for any $\text{ATK}', \text{ATK}'' \in \{\text{CPA}, \text{CCA1}, \text{CCA2}\}$.
 - (b) *Indistinguishability-based* (Section 7):
 - i. “IND-NM-SO-ATK \Rightarrow IND-NM-ATK”, for any $\text{ATK} \in \{\text{CPA}, \text{CCA1}, \text{CCA2}\}$.
 - ii. “IND-NM-CCA2 $\not\Rightarrow$ IND-NM-SO-CCA2”, and “IND-NM-SO-CPA $\not\Rightarrow$ IND-NM-CCA1”.
3. *SIM-NM-SO versus IND-NM-SO* (Section 8):
 “IND-NM-SO-ATK $\not\Rightarrow$ SIM-NM-SO-ATK”, for any $\text{ATK} \in \{\text{CCA1}, \text{CCA2}\}$. In fact, we have a stronger result: “IND-NM-SO-CCA2 $\not\Rightarrow$ SIM-NM-SO-CCA1”.

Based on the relations that we obtained, (in Section 9) we conclude that some known PKE schemes have already obtained SIM-NM-SO-CCA2 or IND-NM-SO-CCA2 security. More specifically, the NC-CCA2 secure encryption scheme proposed by Fehr et al. [11] is SIM-NM-SO-CCA2

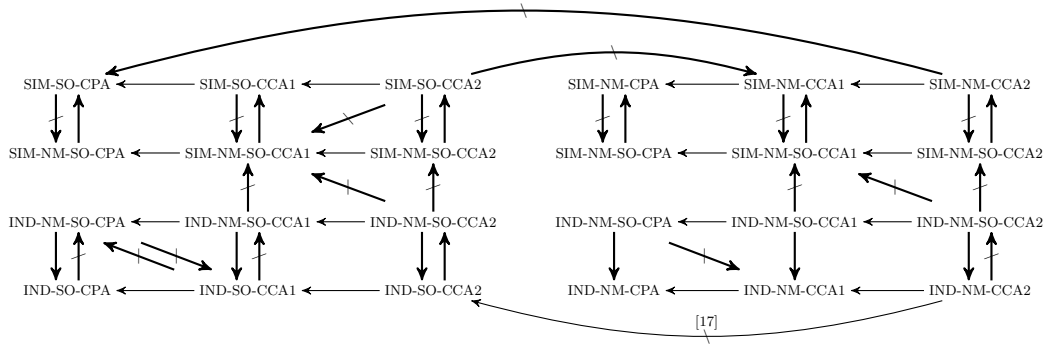


Fig. 1. Relations among SO-NM securities, SO securities and NM securities.

secure; Any IND-SO-CCA2 secure encryption scheme (e.g., [14][12]) is IND-NM-SO-CCA2 secure.

Techniques for the implications. For two main non-trivial implication results, we provide their high-level descriptions of the reasonings here.

- For our contribution 1.(a).ii., the key point is how to construct a SIM-NM-SO-CCA2 simulator S_{NS} from a SIM-SO-CCA2 simulator S . Given S 's output out_S , if it is a valid message, S_{NS} can simply generate a ciphertext by encrypting it, such that the decryption of S_{NS} 's output equals out_S . The barrier is that when out_S is not a valid message, this method doesn't work. To overcome this issue, we apply the idea from [19], assuming that there is an algorithm F recovering ciphertexts from decrypted messages. Under this assumption, S_{NS} can use F to recover a ciphertext from out_S , if out_S falls into the range of decrypted messages. However, this method fails if out_S does not belong to the range of the decryption algorithm Dec . This problem can be solved by assuming that the range of the decryption algorithm Dec is recognizable. With the recognizable property of Dec , SIM-SO-CCA2 security ensures that S 's output out_S is almost always in the range of Dec as long as the SIM-SO-CCA2 adversary's final output is in the range.
- For our contribution 2.(a).i., the key point is constructing a SIM-NM-ATK simulator S_N from a SIM-NM-SO-ATK simulator S_{NS} . Note that S_{NS} has the ability, which S_N doesn't, to ask an opening query. To overcome this issue, we consider a special "half-uniform" message distribution (see Definition 9), which consists of two independent distributions and the second is a uniform one. Correspondingly, the challenge message vector generated from this specific distribution also consists of two parts. If S_{NS} outputs a "half-uniform" distribution and asks to open the uniform part, S_N can always answer it on its own by returning a uniformly chosen message vector. However, S_N still cannot deal with a misbehaved S_{NS} which outputs other distributions or it does not open the uniform part. To solve this problem, we construct a behaved SIM-NM-SO-ATK adversary A_{NS} , which always outputs a half-uniform distribution and asks to open the uniform part, and then SIM-NM-SO-ATK security guarantees S_{NS} is behaved, except with negligible probability.

Observations for the separations. Some of our separation results can be seen as extensions of [1][19][13]. Most of these separations are based on the following observations. Let's look at

the SIM-based notions first. A SIM-NM security notion requires that the decryptions of both of the adversary's and the simulator's outputs be indistinguishable. Note that a non-NM security notion only requires that their outputs be indistinguishable. We can provide a uniformly distributed string, which leads to a special ciphertext (e.g., decrypted to sk), to the adversary through the decryption oracle. It is hard for any SIM-NM simulator to generate such a ciphertext, since it has no access to the decryption oracle. This feature can be used to separate some SIM-based NM and non-NM security notions (in a SOA or non-SOA setting). For the IND-based notions, note that even under CPA attacks, an IND-NM adversary can make a *one-time* parallel decryption query *after* receiving the challenge ciphertext. This feature can be used to separate some IND-based NM and non-NM security notions (in a SOA or non-SOA setting).

Open question. The primary open question is to figure out the relations between SIM-NM-SO and IND-NM-SO security notions. The barriers we encounter are as follows. For NM security notions, there is always a parallel decryption process *after* the adversary receiving the challenge ciphertext. This fact makes the relation between these two notions (even under CPA attacks) similar to that between SIM-SO-CCA2 and IND-SO-CCA2 security. Besides that, we also need to deal with the aforementioned issue, i.e., the SIM-NM-SO simulator's output always contains a ciphertext vector.

2 Preliminaries

Notations. Throughout this paper, we use κ as the security parameter, and ϵ as the empty string. For $n \in \mathbb{N}^+$, let $[n]$ denote the set $\{1, 2, \dots, n\}$. Let U_n denote a uniform distribution over $\{0, 1\}^n$. For a finite set \mathcal{S} , let $s \leftarrow \mathcal{S}$ denote the process of sampling an element s uniformly at random from \mathcal{S} . For a probabilistic algorithm A , let \mathcal{R}_A denote the randomness space of A . We let $y \leftarrow A(x; R)$ denote the process of running A on input x and inner randomness $R \in \mathcal{R}_A$, and outputting y . We write $y \leftarrow A(x)$ for $y \leftarrow A(x; R)$ with uniformly chosen $R \in \mathcal{R}_A$. If A 's running time is polynomial in κ , we say that A is a probabilistic polynomial-time (PPT) algorithm. For two sequences of random variables $X = \{X_\kappa\}_{\kappa \in \mathbb{N}}$ and $Y = \{Y_\kappa\}_{\kappa \in \mathbb{N}}$, if for any PPT algorithm D , $|\Pr[D(X_\kappa, 1^\kappa) = 1] - \Pr[D(Y_\kappa, 1^\kappa) = 1]|$ is negligible in κ , we say that X and Y are computationally indistinguishable (denoted by $X \stackrel{c}{\approx} Y$).

We use boldface letters for vectors. For a vector \mathbf{m} (resp. a finite set \mathcal{S}), we let $|\mathbf{m}|$ (resp. $|\mathcal{S}|$) denote the length of the vector (resp. the size of the set). For a set $I = \{i_1, i_2, \dots, i_{|I|}\} \subseteq [|\mathbf{m}|]$, let $\mathbf{m}[I] = (\mathbf{m}[i_1], \mathbf{m}[i_2], \dots, \mathbf{m}[i_{|I|}])$. We write $m \in \mathbf{m}$ to denote $m \in \{\mathbf{m}[i] \mid i \in [|\mathbf{m}|]\}$, extending the set membership notation to vectors.

Public-key encryption. A public-key encryption (PKE) scheme is a tuple of algorithms $\text{PKE} = (\text{Gen}, \text{Enc}, \text{Dec})$. The key generation algorithm Gen takes a security parameter κ as input and outputs a public/secret key pair (pk, sk) . The encryption algorithm Enc takes a public key pk and a message m as input, and outputs a ciphertext c . The decryption algorithm Dec takes a secret key sk and a ciphertext c as input, and outputs a message m or a failure symbol \perp . For correctness, we require that for $(pk, sk) \leftarrow \text{Gen}(1^\kappa)$ and $c \leftarrow \text{Enc}(pk, m)$, $\text{Dec}(sk, c) = m$ with overwhelming probability.

For simplicity, we write $\text{Enc}(pk, \mathbf{m}) := (\text{Enc}(pk, \mathbf{m}[1]), \text{Enc}(pk, \mathbf{m}[2]), \dots, \text{Enc}(pk, \mathbf{m}[|\mathbf{m}|]))$. Note that for every $i \in [|\mathbf{m}|]$, we use a fresh random coin $\mathbf{r}[i]$ during the encryption of $\mathbf{m}[i]$.

Decryption oracles. For simplicity, we will use the notations $\mathcal{O}_1(\cdot)$ and $\mathcal{O}_2(\cdot)$ in all the security notions throughout the paper. In a chosen-plaintext attack (CPA), both the oracles $\mathcal{O}_1(\cdot)$ and $\mathcal{O}_2(\cdot)$ always return ϵ . In a non-adaptive chosen-ciphertext attack (CCA1), $\mathcal{O}_1(\cdot) = \text{Dec}(sk, \cdot)$, and $\mathcal{O}_2(\cdot)$ still returns ϵ whatever it is queried. In an adaptive chosen-ciphertext attack (CCA2), both $\mathcal{O}_1(\cdot)$ and $\mathcal{O}_2(\cdot)$ are $\text{Dec}(sk, \cdot)$, with the only exception that $\mathcal{O}_2(\cdot)$ returns ϵ when queried on a ciphertext appeared in the challenge ciphertext vector.

Non-malleability for encryption. The first definition of non-malleability for encryption was proposed by Dolev, Dwork and Naor [8][9] in 1991. Their definition is simulation-based. Several years later, comparison-based and indistinguishability-based definitions of non-malleability were proposed [1][5], and their relations were explored in [5][19]. We recall the simulation/indistinguishability-based definitions in [19] as follows.

Definition 1 (SIM-NM security). A public-key encryption scheme $\text{PKE} = (\text{Gen}, \text{Enc}, \text{Dec})$ is SIM-NM-ATK secure, if for any stateful PPT adversary $A = (A_1, A_2)$, there is a stateful PPT simulator $S = (S_1, S_2)$, such that

$$\text{Exp}_{\text{PKE}, A}^{\text{SIM-NM-ATK-Real}}(\kappa) \stackrel{c}{\approx} \text{Exp}_{\text{PKE}, S}^{\text{SIM-NM-ATK-Ideal}}(\kappa),$$

where $\text{ATK} \in \{\text{CPA}, \text{CCA1}, \text{CCA2}\}$, $\text{Exp}_{\text{PKE}, A}^{\text{SIM-NM-ATK-Real}}(\kappa)$ and $\text{Exp}_{\text{PKE}, S}^{\text{SIM-NM-ATK-Ideal}}(\kappa)$ are defined in Table 1.

Definition 2 (IND-NM security). A public-key encryption scheme $\text{PKE} = (\text{Gen}, \text{Enc}, \text{Dec})$ is IND-NM-ATK secure, if for any stateful PPT adversary $A = (A_1, A_2, A_3)$, its advantage $\text{Adv}_{\text{PKE}, A}^{\text{IND-NM-ATK}}(\kappa)$ is negligible, where $\text{ATK} \in \{\text{CPA}, \text{CCA1}, \text{CCA2}\}$. Here

$$\text{Adv}_{\text{PKE}, A}^{\text{IND-NM-ATK}}(\kappa) := |\Pr[\text{Exp}_{\text{PKE}, A}^{\text{IND-NM-ATK-1}}(\kappa) = 1] - \Pr[\text{Exp}_{\text{PKE}, A}^{\text{IND-NM-ATK-0}}(\kappa) = 1]|,$$

where the experiment $\text{Exp}_{\text{PKE}, A}^{\text{IND-NM-ATK-}b}(\kappa)$ ($b \in \{0, 1\}$) is defined in Table 1, and we require that in the experiment, $|\mathbf{m}_0| = |\mathbf{m}_1|$, and $|\mathbf{m}_0[i]| = |\mathbf{m}_1[i]|$ for any $i \in [|\mathbf{m}_0|]$.

Remark 1. Note that in Definition 1 and Definition 2, we do not require that $|\mathbf{y}| = |\mathbf{m}|$ or $|\mathbf{y}| = |\mathbf{m}_b|$. We also note that the ciphertexts contained in \mathbf{y} may be invalid, i.e., $\perp \in \mathbf{x}$. According to [19], these two definitions are stronger than the versions which require that \mathbf{y} must be valid ciphertexts.

Selective opening security for encryption. Simulation-based and indistinguishability-based selective opening security notions were presented by Bellare et al. [4] in Eurocrypt 2009. We follow [4][14][3] for the definition.

Definition 3 (SIM-SO security [3]). A public-key encryption scheme $\text{PKE} = (\text{Gen}, \text{Enc}, \text{Dec})$ is SIM-SO-ATK secure, if for any stateful PPT adversary $A = (A_1, A_2, A_3)$, there is a stateful PPT simulator $S = (S_1, S_2, S_3)$, such that

$$\text{Exp}_{\text{PKE}, A}^{\text{SIM-SO-ATK-Real}}(\kappa) \stackrel{c}{\approx} \text{Exp}_{\text{PKE}, S}^{\text{SIM-SO-ATK-Ideal}}(\kappa),$$

where $\text{ATK} \in \{\text{CPA}, \text{CCA1}, \text{CCA2}\}$, $\text{Exp}_{\text{PKE}, A}^{\text{SIM-SO-ATK-Real}}(\kappa)$ and $\text{Exp}_{\text{PKE}, S}^{\text{SIM-SO-ATK-Ideal}}(\kappa)$ are defined in Table 1.

Table 1. SIM-NM, SIM-SO, IND-NM and IND-SO experiments

SIM-NM experiment:	
$\text{Exp}_{\text{PKE},A}^{\text{SIM-NM-ATK-Real}}(\kappa):$ $(pk, sk) \leftarrow \text{Gen}(1^\kappa)$ $(\mathcal{M}, s) \leftarrow A_1^{\mathcal{O}_1(\cdot)}(pk)$ $\mathbf{m} \leftarrow \mathcal{M}$ $\mathbf{c} \leftarrow \text{Enc}(pk, \mathbf{m})$ $(\mathbf{y}, \sigma) \leftarrow A_2^{\mathcal{O}_2(\cdot)}(\mathbf{c}, s)$ For $i \in [\mathbf{y}]$, If $\mathbf{y}[i] \in \mathbf{c}$, then $\mathbf{x}[i] := \text{COPY}$ else, $\mathbf{x}[i] := \text{Dec}(sk, \mathbf{y}[i])$ return $(\mathcal{M}, \mathbf{m}, \mathbf{x}, \sigma)$	$\text{Exp}_{\text{PKE},S}^{\text{SIM-NM-ATK-Ideal}}(\kappa):$ $(pk, sk) \leftarrow \text{Gen}(1^\kappa)$ $(\mathcal{M}, s) \leftarrow S_1(pk)$ $\mathbf{m} \leftarrow \mathcal{M}$ $(\mathbf{y}, \sigma) \leftarrow S_2(s)$ For $i \in [\mathbf{y}]$, If $\mathbf{y}[i] = \text{COPY}$, then $\mathbf{x}[i] := \text{COPY}$ else, $\mathbf{x}[i] := \text{Dec}(sk, \mathbf{y}[i])$ return $(\mathcal{M}, \mathbf{m}, \mathbf{x}, \sigma)$
SIM-SO experiment:	
$\text{Exp}_{\text{PKE},A}^{\text{SIM-SO-ATK-Real}}(\kappa):$ $(pk, sk) \leftarrow \text{Gen}(1^\kappa)$ $(\mathcal{M}, s_1) \leftarrow A_1^{\mathcal{O}_1(\cdot)}(pk)$ $\mathbf{m} \leftarrow \mathcal{M}$ $\mathbf{r} \leftarrow (\mathcal{R}_{\text{Enc}})^{ \mathbf{m} }$ $\mathbf{c} \leftarrow \text{Enc}(pk, \mathbf{m}; \mathbf{r})$ $(I, s_2) \leftarrow A_2^{\mathcal{O}_2(\cdot)}(\mathbf{c}, s_1)$ $\text{out}_A \leftarrow A_3^{\mathcal{O}_3(\cdot)}(\mathbf{m}[I], \mathbf{r}[I], s_2)$ return $(\mathcal{M}, \mathbf{m}, I, \text{out}_A)$	$\text{Exp}_{\text{PKE},S}^{\text{SIM-SO-ATK-Ideal}}(\kappa):$ $(\mathcal{M}, s_1) \leftarrow S_1(1^\kappa)$ $\mathbf{m} \leftarrow \mathcal{M}$ $(I, s_2) \leftarrow S_2(s_1)$ $\text{out}_s \leftarrow S_3(\mathbf{m}[I], s_2)$ return $(\mathcal{M}, \mathbf{m}, I, \text{out}_s)$
IND-NM experiment:	IND-SO experiment:
$\text{Exp}_{\text{PKE},A}^{\text{IND-NM-ATK-}b}(\kappa):$ $(pk, sk) \leftarrow \text{Gen}(1^\kappa)$ $(\mathbf{m}_0, \mathbf{m}_1, s) \leftarrow A_1^{\mathcal{O}_1(\cdot)}(pk)$ $\mathbf{c} \leftarrow \text{Enc}(pk, \mathbf{m}_b)$ $(\mathbf{y}, \sigma) \leftarrow A_2^{\mathcal{O}_2(\cdot)}(\mathbf{c}, s)$ For $i \in [\mathbf{y}]$, If $\mathbf{y}[i] \in \mathbf{c}$, then $\mathbf{x}[i] := \text{COPY}$ else, $\mathbf{x}[i] := \text{Dec}(sk, \mathbf{y}[i])$ $b' \leftarrow A_3^{\mathcal{O}_3(\cdot)}(\mathbf{x}, \sigma)$ return b'	$\text{Exp}_{\text{PKE},A}^{\text{IND-SO-ATK-}b}(\kappa):$ $(pk, sk) \leftarrow \text{Gen}(1^\kappa)$ $(\mathcal{M}, \text{Resamp}_{\mathcal{M}}, s_1) \leftarrow A_1^{\mathcal{O}_1(\cdot)}(pk)$ $\mathbf{m}_0 \leftarrow \mathcal{M}$ $\mathbf{r} \leftarrow (\mathcal{R}_{\text{Enc}})^{ \mathbf{m}_0 }$ $\mathbf{c} \leftarrow \text{Enc}(pk, \mathbf{m}_0; \mathbf{r})$ $(I, s_2) \leftarrow A_2^{\mathcal{O}_2(\cdot)}(\mathbf{c}, s_1)$ $\mathbf{m}_1 \leftarrow \text{Resamp}_{\mathcal{M}}(I, \mathbf{m}_0[I])$ $b' \leftarrow A_3^{\mathcal{O}_3(\cdot)}(\mathbf{m}_b, \mathbf{r}[I], s_2)$ return b'

For indistinguishability-based selective opening (IND-SO) security notion, we restrict message distributions to be *efficiently re-samplable*. In [3], the IND-SO security notion with this restriction is called “weak” IND-SO security, and the one without this restriction is called “full”. But there is no PKE achieving full IND-SO-CPA security yet.

Definition 4 (Efficiently re-samplable). A message distribution \mathcal{M} is *efficiently re-samplable*, if there is a PPT algorithm $\text{Resamp}_{\mathcal{M}}$, such that for any \mathbf{m} sampled from \mathcal{M} and any subset $I \subseteq [|\mathbf{m}|]$, $\text{Resamp}_{\mathcal{M}}(I, \mathbf{m}[I])$ samples from $\mathcal{M}|_{I, \mathbf{m}[I]}$, i.e., $\mathbf{m}' \leftarrow \text{Resamp}_{\mathcal{M}}(I, \mathbf{m}[I])$ is sampled from the distribution \mathcal{M} , conditioned on $\mathbf{m}'[I] = \mathbf{m}[I]$.

Definition 5 (IND-SO security). A public-key encryption scheme $\text{PKE} = (\text{Gen}, \text{Enc}, \text{Dec})$ is *IND-SO-ATK secure*, if for any stateful PPT adversary $A = (A_1, A_2, A_3)$, its advantage $\text{Adv}_{\text{PKE},A}^{\text{IND-SO-ATK}}(\kappa)$ is negligible, where $\text{ATK} \in \{\text{CPA}, \text{CCA1}, \text{CCA2}\}$. Here

$$\text{Adv}_{\text{PKE},A}^{\text{IND-SO-ATK}}(\kappa) := |\Pr[\text{Exp}_{\text{PKE},A}^{\text{IND-SO-ATK-1}}(\kappa) = 1] - \Pr[\text{Exp}_{\text{PKE},A}^{\text{IND-SO-ATK-0}}(\kappa) = 1]|,$$

where the experiment $\text{Exp}_{\text{PKE},A}^{\text{IND-SO-ATK-}b}(\kappa)$ ($b \in \{0, 1\}$) is defined in Table 1.

3 Non-malleability under selective opening attack

In this section, we formalize non-malleability under selective opening attacks for PKE. We consider simulation-based and indistinguishability-based formalizations of this security, which we call SIM-NM-SO security and IND-NM-SO security, respectively.

Simulation-based selective opening non-malleability. The simulation-based notion of non-malleability under selective opening attacks combines SIM-NM security and SIM-SO security. Informally, a SIM-NM-SO-ATK adversary is a SIM-NM-ATK adversary being allowed to make an additional selective opening query. Similarly, the related simulator is also allowed to make an opening query. The formal definition is as follows.

Definition 6 (SIM-NM-SO security). *A public-key encryption scheme $\text{PKE} = (\text{Gen}, \text{Enc}, \text{Dec})$ is SIM-NM-SO-ATK secure, if for any stateful PPT adversary $A = (A_1, A_2, A_3)$, there is a stateful PPT simulator $S = (S_1, S_2, S_3)$, such that*

$$\text{Exp}_{\text{PKE},A}^{\text{SIM-NM-SO-ATK-Real}}(\kappa) \stackrel{c}{\approx} \text{Exp}_{\text{PKE},S}^{\text{SIM-NM-SO-ATK-Ideal}}(\kappa),$$

where $\text{ATK} \in \{\text{CPA}, \text{CCA1}, \text{CCA2}\}$, $\text{Exp}_{\text{PKE},A}^{\text{SIM-NM-SO-ATK-Real}}(\kappa)$ and $\text{Exp}_{\text{PKE},S}^{\text{SIM-NM-SO-ATK-Ideal}}(\kappa)$ are defined as follows:

Exp_{PKE,A}^{SIM-NM-SO-ATK-Real}(κ):

```

(pk, sk) ← Gen(1κ)
(M, s1) ← A1O1(·)(pk)
m ← M
r ← (REnc)|m|
c ← Enc(pk, m; r)
(I, s2) ← A2O2(·)(c, s1)
(y, σ) ← A3O3(·)(m[I], r[I], s2)
For i ∈ [|y|],
    If y[i] ∈ c, then x[i] := COPY
    else, x[i] := Dec(sk, y[i])
return (M, m, I, x, σ)
    
```

Exp_{PKE,S}^{SIM-NM-SO-ATK-Ideal}(κ):

```

(pk, sk) ← Gen(1κ)
(M, s1) ← S1(pk)
m ← M
(I, s2) ← S2(s1)
(y, σ) ← S3(m[I], s2)
For i ∈ [|y|],
    If y[i] = COPY, then x[i] := COPY
    else, x[i] := Dec(sk, y[i])
return (M, m, I, x, σ)
    
```

Indistinguishability-based selective opening non-malleability. The indistinguishability-based notion of non-malleability under selective opening attacks is also a combination of IND-NM security and IND-SO security. However, there are some subtleties in this combination. First, as the notion of IND-SO security, we require that every message distribution outputted by the adversary should be *efficiently re-samplable*. Second, in this combination, an adversary should be allowed to make two special oracle queries, a selective opening query and a parallel decryption query. In the following formal definition, we allow the adversary to decide the order of these two oracle queries. More specifically, the adversary can make these two queries at any time after receiving the vector of challenge ciphertexts, but only once for each oracle. Note that we require the adversary *has to* make these two oracle queries, since the “challenge bit” b is given through the opening oracle $\text{Open}_{b,\mathcal{M},\mathbf{m}_0,\mathbf{r}}(\cdot)$. The formal definition is as follows.

Definition 7 (IND-NM-SO security). A public-key encryption scheme $\text{PKE} = (\text{Gen}, \text{Enc}, \text{Dec})$ is IND-NM-SO-ATK secure, if for any stateful PPT adversary $A = (A_1, A_2)$, its advantage $\text{Adv}_{\text{PKE}, A}^{\text{IND-NM-SO-ATK}}(\kappa)$ is negligible, where $\text{ATK} \in \{\text{CPA}, \text{CCA1}, \text{CCA2}\}$. Here

$$\text{Adv}_{\text{PKE}, A}^{\text{IND-NM-SO-ATK}}(\kappa) := |\Pr[\text{Exp}_{\text{PKE}, A}^{\text{IND-NM-SO-ATK-1}}(\kappa) = 1] - \Pr[\text{Exp}_{\text{PKE}, A}^{\text{IND-NM-SO-ATK-0}}(\kappa) = 1]|,$$

where the experiment $\text{Exp}_{\text{PKE}, A}^{\text{IND-NM-SO-ATK-}b}(\kappa)$ ($b \in \{0, 1\}$) and the related oracles are defined as follows. In experiment $\text{Exp}_{\text{PKE}, A}^{\text{IND-NM-SO-ATK-}b}(\kappa)$, we require that adversary A_2 access to both oracles $\text{Open}_{b, \mathcal{M}, \mathbf{m}_0, \mathbf{r}}(\cdot)$ and $P_{sk, \mathbf{c}}(\cdot)$ just once respectively.

<p>Exp_{PKE, A}^{IND-NM-SO-ATK-<i>b</i>}(κ):</p> <p>$(pk, sk) \leftarrow \text{Gen}(1^\kappa)$ $(\mathcal{M}, \text{Resamp}_{\mathcal{M}, s_1}) \leftarrow A_1^{\mathcal{O}_1(\cdot)}(pk)$ $\mathbf{m}_0 \leftarrow \mathcal{M}$ $\mathbf{r} \leftarrow (\mathcal{R}_{\text{Enc}})^{ \mathbf{m}_0 }$ $\mathbf{c} \leftarrow \text{Enc}(pk, \mathbf{m}_0; \mathbf{r})$ $b' \leftarrow A_2^{\text{Open}_{b, \mathcal{M}, \mathbf{m}_0, \mathbf{r}(\cdot)}, P_{sk, \mathbf{c}}(\cdot), \mathcal{O}_2(\cdot)}(\mathbf{c}, s_1)$ return b'</p>	<p>Oracle $\text{Open}_{b, \mathcal{M}, \mathbf{m}_0, \mathbf{r}}(I)$:</p> <p>$\mathbf{m}_1 \leftarrow \text{Resamp}_{\mathcal{M}}(I, \mathbf{m}_0[I])$ return $(\mathbf{m}_b, \mathbf{r}[I])$</p> <p>Oracle $P_{sk, \mathbf{c}}(\mathbf{y})$:</p> <p>For $i \in [\mathbf{y}]$, If $\mathbf{y}[i] \in \mathbf{c}$, then $\mathbf{x}[i] := \text{COPY}$ else, $\mathbf{x}[i] := \text{Dec}(sk, \mathbf{y}[i])$ return \mathbf{x}</p>
---	--

Remark 2. In [11][3], the notions of traditional selective opening security were generalized to a new version, where the adversary is allowed to make multiple opening queries adaptively. SIM-NM-SO security and IND-NM-SO security can also be naturally generalized to the similar notions. In this paper, for simplicity, when we talk about selective opening attack (i.e., SIM/IND-SO security or SIM/IND-NM-SO security), we just consider the adversaries making one round of opening query. However, all the results investigated in this paper can be extended to the generalized notions.

4 Relations between SIM-NM-SO securities and SIM-SO securities

In this section, we explore the relations between SIM-NM-SO securities and SIM-SO securities, showing that SIM-NM-SO-ATK security is strictly stronger than SIM-SO-ATK security, for any $\text{ATK} \in \{\text{CPA}, \text{CCA1}, \text{CCA2}\}$.

SIM-NM-SO-ATK \Rightarrow SIM-SO-ATK. We provide a high-level description of the reasoning here.

Given any SIM-SO-ATK adversary $A = (A_1, A_2, A_3)$ for an encryption scheme PKE, we construct a SIM-NM-SO-ATK adversary A' (in Table 2). If $\text{Exp}_{\text{PKE}, A'}^{\text{SIM-NM-SO-ATK-Real}}(\kappa) := (\mathcal{M}, \mathbf{m}, I, \mathbf{x}, \sigma)$, then $\text{Exp}_{\text{PKE}, A}^{\text{SIM-SO-ATK-Real}}(\kappa) = (\mathcal{M}, \mathbf{m}, I, \sigma)$. SIM-NM-SO-ATK security guarantees that there is a simulator S' with respect to A' , such that $\text{Exp}_{\text{PKE}, S'}^{\text{SIM-NM-SO-ATK-Ideal}}(\kappa) \stackrel{c}{\approx} \text{Exp}_{\text{PKE}, A'}^{\text{SIM-NM-SO-ATK-Real}}(\kappa)$, i.e., $(\mathcal{M}_{S'}, \mathbf{m}_{S'}, I_{S'}, \mathbf{x}_{S'}, \sigma_{S'}) \stackrel{c}{\approx} (\mathcal{M}, \mathbf{m}, I, \mathbf{x}, \sigma)$. Hence, $(\mathcal{M}_{S'}, \mathbf{m}_{S'}, I_{S'}, \sigma_{S'}) \stackrel{c}{\approx} (\mathcal{M}, \mathbf{m}, I, \sigma)$. Based on S' , we can construct a SIM-SO-ATK simulator S (in Table 2), such that $\text{Exp}_{\text{PKE}, S}^{\text{SIM-SO-ATK-Ideal}}(\kappa) := (\mathcal{M}_{S'}, \mathbf{m}_{S'}, I_{S'}, \sigma_{S'})$. Hence, we have the following theorem.

Theorem 1. (SIM-NM-SO-ATK \Rightarrow SIM-SO-ATK). For any $\text{ATK} \in \{\text{CPA}, \text{CCA1}, \text{CCA2}\}$, SIM-NM-SO-ATK security implies SIM-SO-ATK security.

Table 2. Constructions of adversary $A' = (A'_1, A'_2, A'_3)$ and simulator $S = (S_1, S_2, S_3)$

$A'_1{}^{\mathcal{O}_1(\cdot)}(pk):$ $(\mathcal{M}, s_1) \leftarrow A'_1{}^{\mathcal{O}_1(\cdot)}(pk)$ return (\mathcal{M}, s_1)	$A'_2{}^{\mathcal{O}_2(\cdot)}(\mathbf{c}, s_1):$ $(I, s_2) \leftarrow A'_2{}^{\mathcal{O}_2(\cdot)}(\mathbf{c}, s_1)$ return (I, s_2)	$A'_3{}^{\mathcal{O}_2(\cdot)}(\mathbf{m}[I], \mathbf{r}[I], s_2):$ $out_A \leftarrow A'_3{}^{\mathcal{O}_2(\cdot)}(\mathbf{m}[I], \mathbf{r}[I], s_2)$ $\mathbf{y} := \mathbf{c}, \sigma := out_A$ return (\mathbf{y}, σ)
$S_1(1^\kappa):$ $(pk, sk) \leftarrow \text{Gen}(1^\kappa)$ $(\mathcal{M}, s_1) \leftarrow S'_1(pk)$ return (\mathcal{M}, s_1)	$S_2(s_1):$ $(I, s_2) \leftarrow S'_2(s_1)$ return (I, s_2)	$S_3(\mathbf{m}[I], s_2):$ $(\mathbf{y}, \sigma) \leftarrow S'_3(\mathbf{m}[I], s_2)$ $out_S := \sigma$ return out_S

SIM-SO-ATK $\not\Rightarrow$ **SIM-NM-SO-ATK**. Now we show that SIM-SO security is strictly weaker than SIM-NM-SO-ATK security. Formally, we have the following theorem.

Theorem 2. (SIM-SO-ATK $\not\Rightarrow$ SIM-NM-SO-ATK). *For any $ATK \in \{CPA, CCA1, CCA2\}$, there is a SIM-SO-ATK secure PKE scheme, which is not SIM-NM-SO-ATK secure.*

We prove this theorem with two counterexamples.

In the case of $ATK = CPA$, we consider the Goldwasser-Micali probabilistic encryption scheme (the GM scheme) [10]. In [4], Bellare et al. pointed out that the GM scheme is SIM-SO-CPA secure. We claim that the GM scheme is not SIM-NM-SO-CPA secure because of its homomorphic property. Roughly speaking, let the challenge ciphertext vector \mathbf{c} be generated from a random message vector \mathbf{m} . We can construct an adversary A who encrypts bit 0 to obtain a ciphertext y' , and then outputs $\mathbf{y} := (y' \cdot \mathbf{c}[i])_{i \in [n]} \neq \mathbf{c}$. Obviously, the decryption of \mathbf{y} is $\mathbf{x} := (0 \oplus \mathbf{m}[i])_{i \in [n]} = \mathbf{m}$. However, no PPT simulator S can output a ciphertext vector \mathbf{y} satisfying $\mathbf{x} = \mathbf{m}$, since \mathbf{m} was uniformly chosen and no information about \mathbf{m} is leaked to S except the opened messages.

In the case of $ATK \in \{CCA1, CCA2\}$, we show a counterexample as follows. The main idea of our counterexample is similar to that in [19]. Let $\text{PKE} = (\text{Gen}, \text{Enc}, \text{Dec})$ be an encryption scheme. We construct a new scheme $\widetilde{\text{PKE}} = (\widetilde{\text{Gen}}, \widetilde{\text{Enc}}, \widetilde{\text{Dec}})$:

To prove that $\widetilde{\text{PKE}}$ is not SIM-NM-SO-CCA1/CCA2 secure, consider the adversary A : A obtains θ by querying the decryption oracle on input $(c, 0, 1^\kappa)$, and outputs a ciphertext whose decryption is \perp . Notice that any PPT simulator S has no information about the uniformly chosen θ , since it cannot access to the decryption oracle. So the probability that the simulator outputs a ciphertext whose decryption is \perp is negligible. Consider the distinguisher D : On input $(\mathcal{M}, \mathbf{m}, I, \mathbf{x}, \sigma)$, return 1 if and only if $\perp \in \mathbf{x}$. Then D can distinguish $\text{Exp}_{\widetilde{\text{PKE}}, A}^{\text{SIM-NM-SO-CCA1/CCA2-Real}}(\kappa)$ and $\text{Exp}_{\widetilde{\text{PKE}}, S}^{\text{SIM-NM-SO-CCA1/CCA2-Ideal}}(\kappa)$. Hence, $\widetilde{\text{PKE}}$ is not SIM-NM-SO-CCA1/CCA2 secure. Now, what remains is to prove the SIM-SO-CCA1/CCA2 security of $\widetilde{\text{PKE}}$, which is guaranteed by PKE 's SIM-SO-CCA1/CCA2 security. The formal proof will be given in Appendix A.

Remark 3. The aforementioned analysis actually shows that $\widetilde{\text{PKE}}$ is not SIM-NM-SO-CCA1 secure, *even if* PKE is SIM-SO-CCA2 secure. So we have a stronger conclusion: “SIM-SO-CCA2 $\not\Rightarrow$ SIM-NM-SO-CCA1”, and a similar analysis gives “SIM-SO-CCA2 $\not\Rightarrow$ SIM-NM-CCA1”.

Remark 4. Since SIM-SO-CPA security implies IND-SO-CPA security, the GM scheme is also IND-SO-CPA secure. Due to the same reason, we will find that the GM scheme is not IND-NM-SO-CPA secure. In other words, the GM scheme is an example which is SIM/IND-SO-CPA

Table 3. $\widetilde{\text{PKE}} = (\widetilde{\text{Gen}}, \widetilde{\text{Enc}}, \widetilde{\text{Dec}})$

$\widetilde{\text{Gen}}(1^\kappa)$:	$\widetilde{\text{Enc}}(\widetilde{pk}, m)$:	$\widetilde{\text{Dec}}(\widetilde{sk}, \widetilde{c})$:
$(pk, sk) \leftarrow \text{Gen}(1^\kappa)$	$c \leftarrow \text{Enc}(pk, m)$	Parse $\widetilde{c} = (c, b, \vartheta)$
$\theta \leftarrow \{0, 1\}^\kappa$	return $\widetilde{c} := (c, 1, 0^\kappa)$	If $b = 0$ and $\vartheta = 1^\kappa$, then return θ
$\widetilde{pk} := pk$		If $b = 0$ and $\vartheta = \theta$, then return \perp
$\widetilde{sk} := (sk, \theta)$		If $b = 1$ and $\vartheta = 0^\kappa$, set $m = \text{Dec}(sk, c)$
return $(\widetilde{pk}, \widetilde{sk})$		If $m = \perp$, then return 0; else, return m
		Otherwise, return 0

secure, but meanwhile SIM/IND-NM-SO-CPA insecure.

A note on SIM-NM-SO-CCA2. In [19], Pass et al. specified a special condition (i.e., the message space and the range of the decryption algorithm are identical), under which IND-NM-CCA1/CCA2 security and SIM-NM-CCA1/CCA2 security are equivalent. Interestingly, we find that under this condition, if the range of the decryption algorithm is recognizable (i.e., roughly speaking, there is a polynomial-time algorithm, which can determine whether an element is in the range of the decryption algorithm), then SIM-SO-CCA2 security implies SIM-NM-SO-CCA2 security (i.e., these two security notions are equivalent).

Below we recall the special condition proposed in [19], which we name “invertible decryption”.

Definition 8 (Invertible decryption). *Let $\text{PKE} = (\text{Gen}, \text{Enc}, \text{Dec})$ be a PKE scheme. Dec is invertible if there exists a PPT algorithm F , such that for any ciphertext c , $\text{Dec}(sk, F(pk, \text{Dec}(sk, c))) = \text{Dec}(sk, c)$, where $(pk, sk) \leftarrow \text{Gen}(1^\kappa)$.*

Theorem 3. *If a SIM-SO-CCA2 secure PKE scheme has an invertible decryption algorithm, and the range of the decryption algorithm is recognizable in polynomial time, then the scheme is also SIM-NM-SO-CCA2 secure.*

Proof. Let $\text{PKE} = (\text{Gen}, \text{Enc}, \text{Dec})$ be a SIM-SO-CCA2 secure encryption scheme, such that it has an inverting algorithm F , and the range of Dec is recognizable. Now we prove PKE is SIM-NM-SO-CCA2 secure.

For any PPT adversary $A = (A_1, A_2, A_3)$ attacking PKE in the sense of SIM-NM-SO-CCA2, we construct a PPT adversary $A' = (A'_1, A'_2, A'_3)$ attacking PKE in the sense of SIM-SO-CCA2 as follows.

Receiving a public key pk , A'_1 runs A_1 on the input of pk . For any decryption query c' asked by A_1 , A'_1 sends c' to its own decryption oracle, and then returns the answer to A_1 . At some point, A_1 returns a message distribution \mathcal{M} . Then, A'_1 outputs \mathcal{M} to the challenger.

On the other side, the challenger samples $\mathbf{m} \leftarrow \mathcal{M}$ and $\mathbf{r} \leftarrow (\mathcal{R}_{\text{Enc}})^{|\mathbf{m}|}$, and generates $\mathbf{c}^* \leftarrow \text{Enc}(pk, \mathbf{m}; \mathbf{r})$.

Receiving \mathbf{c}^* from the challenger, A'_2 runs A_2 on the input of \mathbf{c}^* . For any decryption query c' asked by A_2 , A'_2 answers with its own decryption oracle as before (of course, both A_2 and A'_2 are not allowed to query $c' \in \mathbf{c}^*$). At some point, A_2 returns a subset $I \subset [|\mathbf{c}^*|]$. Then, A'_2 outputs I to the challenger.

Receiving $\mathbf{m}[I]$ and $\mathbf{r}[I]$, A'_3 runs A_3 on the input of $\mathbf{m}[I]$ and $\mathbf{r}[I]$. For any decryption query c' asked by A_3 , A'_3 answers it as before. At last, A_3 returns its final output (\mathbf{y}, σ) . Then, A'_3 generates \mathbf{x} (where $|\mathbf{x}| = |\mathbf{y}|$) as follows: For $i = 1, 2, \dots, |\mathbf{y}|$, if $\mathbf{y}[i] \notin \mathbf{c}^*$, submit $\mathbf{y}[i]$ to A' 's

decryption oracle and denote the decryption by $\mathbf{x}[i]$; if $\mathbf{y}[i] \in \mathbf{c}^*$, set that $\mathbf{x}[i] := \text{COPY}$. Finally, A'_3 outputs $out_{A'} := (\mathbf{x}, \sigma)$.

That is the description of adversary A' .

Notice that A' perfectly simulates the real experiment $\text{Exp}_{\text{PKE},A}^{\text{SIM-NM-SO-ATK-Real}}(\kappa)$ for A . Hence,

$$\text{Exp}_{\text{PKE},A'}^{\text{SIM-SO-CCA2-Real}}(\kappa) = (\mathcal{M}, \mathbf{m}, I, out_{A'}) = (\mathcal{M}, \mathbf{m}, I, \mathbf{x}, \sigma) = \text{Exp}_{\text{PKE},A}^{\text{SIM-NM-SO-CCA2-Real}}(\kappa). \quad (1)$$

Since PKE is SIM-SO-CCA2 secure, there is a PPT simulator $S' = (S'_1, S'_2, S'_3)$ such that

$$\text{Exp}_{\text{PKE},S'}^{\text{SIM-SO-CCA2-Ideal}}(\kappa) \stackrel{c}{\approx} \text{Exp}_{\text{PKE},A'}^{\text{SIM-SO-CCA2-Real}}(\kappa). \quad (2)$$

Now, based on S' , we construct a simulator $S = (S_1, S_2, S_3)$ in the sense of SIM-NM-SO-CCA2.

Receiving a public key pk , S_1 runs S'_1 on the input of 1^κ . Then S_1 outputs the $\mathcal{M}_{S'}$ returned by S'_1 .

On the other side, the challenger samples $\mathbf{m}_{S'} \leftarrow \mathcal{M}_{S'}$, without returning anything to S .

Later, S'_2 outputs a subset $I_{S'}$. S_2 outputs $I_{S'}$ to the challenger.

Upon receiving $\mathbf{m}_{S'}[I_{S'}]$, S_3 runs S'_3 on the input of $\mathbf{m}_{S'}[I_{S'}]$, obtaining S'_3 's final output $out_{S'}$. After parsing $out_{S'} = (\mathbf{x}_{S'}, \sigma_{S'})$, S_3 checks whether there is some $i_0 \in [|\mathbf{x}_{S'}|]$ such that $\mathbf{x}_{S'}[i_0] \neq \text{COPY}$ and meanwhile $\mathbf{x}_{S'}[i_0]$ is not in the range of Dec . It is feasible to check that in polynomial time since the range of Dec is recognizable. If there is such an i_0 , then S_3 aborts by outputting a random string. Otherwise, S_3 generates \mathbf{y}_S (where $|\mathbf{y}_S| = |\mathbf{x}_{S'}|$) as follows: For $i = 1, 2, \dots, |\mathbf{y}_S|$, if $\mathbf{x}_{S'}[i] = \text{COPY}$, then set $\mathbf{y}_S[i] = \text{COPY}$; otherwise, generate $\mathbf{y}_S[i] \leftarrow \text{F}(pk, \mathbf{x}_{S'}[i])$. After that, S_3 outputs $(\mathbf{y}_S, \sigma_{S'})$.

That is the description of simulator S .

Let bad denote the event that S aborts. If bad does not occur, then for any $j \in [|\mathbf{x}_{S'}|]$ such that $\mathbf{x}_{S'}[j] \neq \text{COPY}$, there is some ciphertext \hat{c}_j (not has to be valid), such that $\text{Dec}(sk, \hat{c}_j) = \mathbf{x}_{S'}[j]$. We have $\text{Dec}(sk, \mathbf{y}_S[j]) = \text{Dec}(sk, \text{F}(pk, \text{Dec}(sk, \hat{c}_j))) = \text{Dec}(sk, \hat{c}_j) = \mathbf{x}_{S'}[j]$. In this case,

$$\begin{aligned} \text{Exp}_{\text{PKE},S}^{\text{SIM-NM-SO-CCA2-Ideal}}(\kappa) &= (\mathcal{M}_{S'}, \mathbf{m}_{S'}, I_{S'}, \mathbf{x}_{S'}, \sigma_{S'}) \\ &= (\mathcal{M}_{S'}, \mathbf{m}_{S'}, I_{S'}, out_{S'}) \\ &= \text{Exp}_{\text{PKE},S'}^{\text{SIM-SO-CCA2-Ideal}}(\kappa). \end{aligned}$$

So for any PPT algorithm D ,

$$|\Pr[D(\text{Exp}_{\text{PKE},S}^{\text{SIM-NM-SO-CCA2-Ideal}}(\kappa)) = 1] - \Pr[D(\text{Exp}_{\text{PKE},S'}^{\text{SIM-SO-CCA2-Ideal}}(\kappa)) = 1]| \leq \Pr[\text{bad}].$$

Notice that if $\Pr[\text{bad}]$ is negligible, then we have

$$\text{Exp}_{\text{PKE},S}^{\text{SIM-NM-SO-CCA2-Ideal}}(\kappa) \stackrel{c}{\approx} \text{Exp}_{\text{PKE},S'}^{\text{SIM-SO-CCA2-Ideal}}(\kappa). \quad (3)$$

Combining equations (1), (2) and (3) gives

$$\text{Exp}_{\text{PKE},A}^{\text{SIM-NM-SO-CCA2-Real}}(\kappa) \stackrel{c}{\approx} \text{Exp}_{\text{PKE},S}^{\text{SIM-NM-SO-CCA2-Ideal}}(\kappa).$$

Hence, what remains is to prove that $\Pr[\text{bad}]$ is negligible. We consider the following distinguisher D' :

Algorithm $D'(\mathcal{M}, \mathbf{m}, I, out)$:

Parse $out = (\mathbf{x}, \sigma)$

For $i \in [|\mathbf{x}|]$,

If $\mathbf{x}[i] \neq \text{COPY}$ and $\mathbf{x}[i]$ is not in the range of Dec, then return 1

Return 0

It is obvious that $\Pr[D'(\text{Exp}_{\text{PKE}, A'}^{\text{SIM-SO-CCA2-Real}}(\kappa)) = 1] = 0$, and $\Pr[D'(\text{Exp}_{\text{PKE}, S'}^{\text{SIM-SO-CCA2-Ideal}}(\kappa)) = 1] = \Pr[\text{bad}]$. In other words,

$$\Pr[\text{bad}] = |\Pr[D'(\text{Exp}_{\text{PKE}, A'}^{\text{SIM-SO-CCA2-Real}}(\kappa)) = 1] - \Pr[D'(\text{Exp}_{\text{PKE}, S'}^{\text{SIM-SO-CCA2-Ideal}}(\kappa)) = 1]|.$$

Hence, equation (2) guarantees that $\Pr[\text{bad}]$ is negligible. So we finish the proof of Theorem 3. \square

5 Relations between IND-NM-SO securities and IND-SO securities

In this section, we explore the relations between IND-NM-SO securities and IND-SO securities. First of all, for any $\text{ATK} \in \{\text{CPA}, \text{CCA1}, \text{CCA2}\}$, an IND-NM-SO-ATK adversary is more powerful than an IND-SO-ATK adversary in that it can make an additional query to oracle $P_{sk}(\cdot)$. Intuitively, IND-NM-SO-ATK security implies IND-SO-ATK security. Further more, any IND-SO-CCA2 adversary A is able to access to the decryption oracle after receiving the challenge ciphertext vector. So providing A the ability to make a parallel decryption query yields no additional power. The above analysis results in the following theorem.

Theorem 4. (IND-NM-SO-ATK \Rightarrow IND-SO-ATK, IND-NM-SO-CCA2 \Leftrightarrow IND-SO-CCA2). *For any $\text{ATK} \in \{\text{CPA}, \text{CCA1}, \text{CCA2}\}$, IND-NM-SO-ATK security implies IND-SO-ATK security. Further more, if $\text{ATK} = \text{CCA2}$, these two securities are equivalent.*

IND-NM-SO-CPA $\not\Leftarrow$ IND-SO-CCA1. Formally, we have the following theorem. This is an direct extension of the conclusion in [1]. So we just provide a high-level description of the reasoning here.

Theorem 5. (IND-NM-SO-CPA $\not\Leftarrow$ IND-SO-CCA1). *There is an IND-SO-CCA1 secure PKE scheme, which is not IND-NM-SO-CPA secure; vice versa.*

The direction \Leftarrow . Note that after receiving the challenge ciphertext, the IND-SO-CCA1 adversary cannot access to the decryption oracle, but the IND-NM-SO-CPA adversary still can make a parallel decryption query. Based on this observation, any PKE scheme, achieving IND-SO-CCA1 but not IND-SO-CCA2 security, might be used as a counterexample. The following scheme PKE' , with message space $\{0, 1\}^\kappa$, is from [1]. If the basic scheme $\text{PKE} = (\text{Gen}, \text{Enc}, \text{Dec})$ is IND-SO-CCA1 secure, then we can prove that PKE' is IND-SO-CCA1 secure but not IND-NM-SO-CPA secure. The formal proof is in Appendix B.

The direction \Rightarrow . Note that an IND-NM-SO-CPA adversary can make just a one-time decryption query (although it is parallel), but an IND-SO-CCA1 adversary can query the decryption oracle polynomial times. Based on this observation, we provide a PKE scheme PKE'' , which is identical to the scheme PKE in Section 4, except that during the decryption, roughly, the decryption algorithm returns the original secret key sk instead of the special symbol \perp , in the case of " $b = 0$ and $\vartheta = \theta$ ". The analysis is similar to that in Section 4. The IND-SO-CCA1

Table 4. $\text{PKE}' = (\text{Gen}', \text{Enc}', \text{Dec}')$

$\text{Gen}'(1^\kappa)$: $(pk, sk) \leftarrow \text{Gen}(1^\kappa)$ $pk' := pk$ $sk' := sk$ return (pk', sk')	$\text{Enc}'(pk', m)$: $c_1 \leftarrow \text{Enc}(pk, m)$ $c_2 \leftarrow \text{Enc}(pk, \bar{m})$ (Note: \bar{m} is the bitwise complement of m) return $c := (c_1, c_2)$	$\text{Dec}'(sk', c)$: Parse $c = (c_1, c_2)$ $m = \text{Dec}(sk, c_1)$ return m
--	--	--

adversary can obtain θ by querying the decryption oracle on input $(c, 0, 1^\kappa)$, so it can obtain the original sk by querying on $(c, 0, \theta)$. Hence, PKE'' is not IND-SO-CCA1 secure. However, the IND-NM-SO-CPA adversary cannot make any other decryption query after the the parallel decryption query. Notice that θ is uniformly chosen, so PKE'' can be proved IND-NM-SO-CPA secure. The formal proof is in Appendix B.

Remark 5. Since IND-SO-CCA1 (resp. IND-NM-SO-CCA1) security implies IND-SO-CPA (resp. IND-NM-SO-CPA) security, we have the following corollary.

Corollary 1. (IND-SO-CPA/CCA1 $\not\Rightarrow$ IND-NM-SO-CPA/CCA1). *IND-SO-CPA/CCA1 security is strictly weaker than IND-NM-SO-CPA/CCA1 security.*

6 Relations between SIM-NM-SO securities and SIM-NM securities

SIM-NM-SO-ATK \Rightarrow SIM-NM-ATK. Compared with the conclusion that “SIM-NM-SO-ATK \Rightarrow SIM-SO-ATK”, this conclusion is not that obvious. That is because, compared with the SIM-NM-SO-ATK adversary, although the SIM-NM-ATK adversary is less powerful (i.e., not allowed to make any opening query), the corresponding simulator also has less information (i.e., not allowed to make any opening query) about the message vector. Formally, we have the following theorem.

Theorem 6. (SIM-NM-SO-ATK \Rightarrow SIM-NM-ATK). *For any $\text{ATK} \in \{\text{CPA}, \text{CCA1}, \text{CCA2}\}$, SIM-NM-SO-ATK security implies SIM-NM-ATK security.*

For convenience, we firstly define a special message distribution, and then turn to the formal proof.

Definition 9 ((n_1, n_2)-half-uniform distribution). *A distribution \mathcal{M} is (n_1, n_2)-half-uniform, if it satisfies the following three properties: (1) For any $\mathbf{m} \leftarrow \mathcal{M}$, $|\mathbf{m}| = 2n_1$; (2) $\mathcal{M} = \mathcal{M}_A || (U_{n_2})^{n_1}$, where \mathcal{M}_A is independent of $(U_{n_2})^{n_1}$; (3) The description of \mathcal{M} consists of two descriptions (i.e., \mathcal{M}_A and $(U_{n_2})^{n_1}$).*

Remark 6. In Table 1, every “ \mathcal{M} ” returned by A_1 or S_1 actually stands for the *description* of message distribution \mathcal{M} . The above property (3) requires that receiving a description of an (n_1, n_2)-half-uniform distribution \mathcal{M} , any one can efficiently extract the description of the related distribution \mathcal{M}_A .

Proof. We prove that “SIM-NM-SO-CCA2 security \Rightarrow SIM-NM-CCA2 security”. The proof in the case of CPA/CCA1 is similar, which we will omit here.

Let $\text{PKE} = (\text{Gen}, \text{Enc}, \text{Dec})$ be a SIM-NM-SO-CCA2 secure encryption scheme. For any PPT adversary $A = (A_1, A_2)$ attacking PKE in the sense of SIM-NM-CCA2, we construct a PPT adversary $A' = (A'_1, A'_2, A'_3)$ attacking PKE in the sense of SIM-NM-SO-CCA2 as follows.

Receiving a public key pk , A'_1 runs A_1 on the input of pk . For any decryption query c' asked by A_1 , A'_1 sends c' to its own decryption oracle, and then returns the answer to A_1 . At some point, A_1 returns a message distribution \mathcal{M}_A . Without loss of generality, we assume that all the message vectors sampled from \mathcal{M}_A have the same size (denoted by n_1), i.e., for any $\mathbf{m}_A \leftarrow \mathcal{M}_A$, $|\mathbf{m}_A| = n_1$. Then, A'_1 outputs $\mathcal{M}_{A'} := \mathcal{M}_A || (U_{n_2})^{n_1}$, where n_2 is also an integer polynomial in κ .

On the other side, the challenger chooses $\mathbf{m}_{A'} \leftarrow \mathcal{M}_{A'}$ (i.e., samples $\mathbf{m}_A \leftarrow \mathcal{M}_A$, $\mathbf{m}_U \leftarrow (U_{n_2})^{n_1}$, and sets $\mathbf{m}_{A'} := \mathbf{m}_A || \mathbf{m}_U$) and $\mathbf{r} \leftarrow (\mathcal{R}_{\text{Enc}})^{2n_1}$, and generates $\mathbf{c}^* \leftarrow \text{Enc}(pk, \mathbf{m}_{A'}; \mathbf{r})$.

Upon receiving \mathbf{c}^* from the challenger, A'_2 outputs $I_{A'} := \{n_1 + 1, n_1 + 2, \dots, 2n_1\}$ as its opening query.

Receiving $\mathbf{m}_{A'}[I_{A'}]$ and $\mathbf{r}[I_{A'}]$ from the challenger, A'_3 parses $\mathbf{c}^* = \mathbf{c}_A || \mathbf{c}_U$, such that $|\mathbf{c}_A| = |\mathbf{c}_U| = n_1$. Then, A'_3 runs A_2 on the input of \mathbf{c}_A . For any decryption query c' asked by A_2 , if $c' \notin \mathbf{c}_U$, A'_3 answers this query with its own decryption oracle; otherwise, A'_3 answers this query with $\mathbf{m}_{A'}[I_{A'}]$, since $\mathbf{m}_{A'}[I_{A'}] = \mathbf{m}_U$. Finally, receiving A_2 's final output (\mathbf{y}, σ) , A'_3 generates its own ciphertext vector \mathbf{y}' , where $|\mathbf{y}'| = |\mathbf{y}|$, as follows: For $i \in [|\mathbf{y}'|]$,

- If $\mathbf{y}[i] \in \mathbf{c}_U$ and meanwhile $\mathbf{y}[i] \notin \mathbf{c}_A$, then A'_3 recovers the decryption of $\mathbf{y}[i]$ (denoted by $\mathbf{x}[i]$) from $\mathbf{m}_{A'}[I_{A'}] = \mathbf{m}_U$, and generates $\mathbf{y}'[i] \leftarrow \text{Enc}(pk, \mathbf{x}[i])$ such that $\mathbf{y}'[i] \notin \mathbf{c}_A \cup \mathbf{c}_U$. It is easy for A'_3 to generate such a $\mathbf{y}'[i]$, since PKE is a probabilistic encryption scheme achieving SIM-NM-SO-CCA2 security.
- Otherwise, set that $\mathbf{y}'[i] := \mathbf{y}[i]$.

A'_3 returns (\mathbf{y}', σ) as its final output.

That is the description of adversary A' .

Note that A' perfectly simulates the real experiment $\text{Exp}_{\text{PKE}, A}^{\text{SIM-NM-CCA2-Real}}(\kappa)$ for A , and the decryptions of \mathbf{y}' and \mathbf{y} are identical (denoted by \mathbf{x}). We have that

$$\begin{aligned} \text{Exp}_{\text{PKE}, A'}^{\text{SIM-NM-SO-CCA2-Real}}(\kappa) &= (\mathcal{M}_{A'}, \mathbf{m}_{A'}, I_{A'}, \mathbf{x}, \sigma) \\ &= (\mathcal{M}_A || (U_{n_2})^{n_1}, \mathbf{m}_A || \mathbf{m}_U, \{n_1 + 1, n_1 + 2, \dots, 2n_1\}, \mathbf{x}, \sigma), \end{aligned} \quad (4)$$

and

$$\text{Exp}_{\text{PKE}, A}^{\text{SIM-NM-CCA2-Real}}(\kappa) = (\mathcal{M}_A, \mathbf{m}_A, \mathbf{x}, \sigma). \quad (5)$$

Since PKE is SIM-NM-SO-CCA2 secure, there is a PPT simulator $S' = (S'_1, S'_2, S'_3)$, such that

$$\text{Exp}_{\text{PKE}, S'}^{\text{SIM-NM-SO-CCA2-Ideal}}(\kappa) \stackrel{c}{\approx} \text{Exp}_{\text{PKE}, A'}^{\text{SIM-NM-SO-CCA2-Real}}(\kappa). \quad (6)$$

Now, based on S' , we show a simulator $S = (S_1, S_2)$ in the sense of SIM-NM-CCA2.

Receiving a public key pk , S_1 obtains a message distribution $\mathcal{M}_{S'}$ by running S'_1 on the input of pk . After receiving an opening query $I_{S'}$ from S'_2 , S_1 runs as follows: If $\mathcal{M}_{S'}$ is not (n_1, n_2) -half-uniform (for some n_1, n_2 that are both polynomial in κ), or $I_{S'} \neq \{n_1 + 1, n_1 + 2, \dots, 2n_1\}$ for the n_1 determined by $\mathcal{M}_{S'}$, then S aborts (with S_1 outputting $\mathcal{M}_S = U_{n_3}$ for some integer n_3 , and S_2 outputting randomly chosen (\mathbf{y}_U, σ_U)); Otherwise, S_1 parses $\mathcal{M}_{S'} = \mathcal{M}_S || (U_{n_2})^{n_1}$, and outputs \mathcal{M}_S to the challenger.

On the other side, the challenger samples $\mathbf{m}_S \leftarrow \mathcal{M}_S$, without returning anything to S .

S_2 samples $\mathbf{m}_U \leftarrow (U_{n_2})^{n_1}$, and runs S'_3 on the input of \mathbf{m}_U . Finally, S_2 outputs S'_3 's final output $(\mathbf{y}_{S'}, \sigma_{S'})$.

That is the description of simulator S .

Let bad denote the event that S aborts, and $\mathbf{x}_{S'}$ denote the decryption of $\mathbf{y}_{S'}$.

Then, *when bad does not occur*, both of the following equations hold,

$$\text{Exp}_{\text{PKE},S'}^{\text{SIM-NM-SO-CCA2-Ideal}}(\kappa) = (\mathcal{M}_S \parallel (U_{n_2})^{n_1}, \mathbf{m}_S \parallel \mathbf{m}_U, \{n_1 + 1, n_1 + 2, \dots, 2n_1\}, \mathbf{x}_{S'}, \sigma_{S'}), \quad (7)$$

$$\text{Exp}_{\text{PKE},S}^{\text{SIM-NM-CCA2-Ideal}}(\kappa) = (\mathcal{M}_S, \mathbf{m}_S, \mathbf{x}_{S'}, \sigma_{S'}). \quad (8)$$

Hence, for any PPT distinguisher D , we denote its advantage by

$$\begin{aligned} \text{Adv}_D &:= |\Pr[D(\text{Exp}_{\text{PKE},A}^{\text{SIM-NM-CCA2-Real}}(\kappa)) = 1] - \Pr[D(\text{Exp}_{\text{PKE},S}^{\text{SIM-NM-CCA2-Ideal}}(\kappa)) = 1]| \\ &= |\Pr[D(\text{Exp}_{\text{PKE},A}^{\text{SIM-NM-CCA2-Real}}(\kappa)) = 1] - \Pr[D(\text{Exp}_{\text{PKE},S}^{\text{SIM-NM-CCA2-Ideal}}(\kappa)) = 1 | \neg \text{bad}] \cdot \Pr[\neg \text{bad}] \\ &\quad - \Pr[D(\text{Exp}_{\text{PKE},S}^{\text{SIM-NM-CCA2-Ideal}}(\kappa)) = 1 | \text{bad}] \cdot \Pr[\text{bad}]| \\ &\leq |\Pr[D(\text{Exp}_{\text{PKE},A}^{\text{SIM-NM-CCA2-Real}}(\kappa)) = 1] - \Pr[D(\text{Exp}_{\text{PKE},S}^{\text{SIM-NM-CCA2-Ideal}}(\kappa)) = 1 | \neg \text{bad}] \cdot \Pr[\neg \text{bad}]| \\ &\quad + \Pr[\text{bad}]. \end{aligned}$$

To bound the inequality, we present the following two lemmas and postpone their proofs.

Lemma 1. $|\Pr[D(\text{Exp}_{\text{PKE},A}^{\text{SIM-NM-CCA2-Real}}(\kappa)) = 1] - \Pr[D(\text{Exp}_{\text{PKE},S}^{\text{SIM-NM-CCA2-Ideal}}(\kappa)) = 1 | \neg \text{bad}] \cdot \Pr[\neg \text{bad}]|$ is negligible.

Lemma 2. $\Pr[\text{bad}]$ is negligible.

Hence, Adv_D is negligible. So we conclude that

$$\text{Exp}_{\text{PKE},A}^{\text{SIM-NM-CCA2-Real}}(\kappa) \stackrel{c}{\approx} \text{Exp}_{\text{PKE},S}^{\text{SIM-NM-CCA2-Ideal}}(\kappa),$$

which means that PKE is SIM-NM-CCA2 secure.

So what remains is to prove Lemma 1 and Lemma 2.

Proof. (of Lemma 1)

Based on the aforementioned D , we show an algorithm D' , distinguishing $\text{Exp}_{\text{PKE},A'}^{\text{SIM-NM-SO-CCA2-Real}}(\kappa)$ and $\text{Exp}_{\text{PKE},S'}^{\text{SIM-NM-SO-CCA2-Ideal}}(\kappa)$, described in Table 5.

Combining equations (4), (5), (7) and (8), it is not hard to see that D' has the following properties:

- $D'(\text{Exp}_{\text{PKE},A'}^{\text{SIM-NM-SO-CCA2-Real}}(\kappa)) = D(\text{Exp}_{\text{PKE},A}^{\text{SIM-NM-CCA2-Real}}(\kappa))$.
- If bad does not occur, then $D'(\text{Exp}_{\text{PKE},S'}^{\text{SIM-NM-SO-CCA2-Ideal}}(\kappa)) = D(\text{Exp}_{\text{PKE},S}^{\text{SIM-NM-CCA2-Ideal}}(\kappa))$.
- If bad occurs, then $D'(\text{Exp}_{\text{PKE},S'}^{\text{SIM-NM-SO-CCA2-Ideal}}(\kappa)) = 0$.

Let $\text{Adv}_{D'}$ denote D' 's advantage. So we have that

$$\begin{aligned} \text{Adv}_{D'} &:= |\Pr[D'(\text{Exp}_{\text{PKE},A'}^{\text{SIM-NM-SO-CCA2-Real}}(\kappa)) = 1] - \Pr[D'(\text{Exp}_{\text{PKE},S'}^{\text{SIM-NM-SO-CCA2-Ideal}}(\kappa)) = 1]| \\ &= |\Pr[D'(\text{Exp}_{\text{PKE},A'}^{\text{SIM-NM-SO-CCA2-Real}}(\kappa)) = 1] \\ &\quad - \Pr[D'(\text{Exp}_{\text{PKE},S'}^{\text{SIM-NM-SO-CCA2-Ideal}}(\kappa)) = 1 | \neg \text{bad}] \cdot \Pr[\neg \text{bad}]| \\ &= |\Pr[D(\text{Exp}_{\text{PKE},A}^{\text{SIM-NM-CCA2-Real}}(\kappa)) = 1] \\ &\quad - \Pr[D(\text{Exp}_{\text{PKE},S}^{\text{SIM-NM-CCA2-Ideal}}(\kappa)) = 1 | \neg \text{bad}] \cdot \Pr[\neg \text{bad}]| \end{aligned}$$

Equation (6) guarantees that $\text{Adv}_{D'}$ is negligible. So we finish the proof of Lemma 1. \square

Table 5. Algorithm D' and Algorithm D''

Algorithm $D'(\mathcal{M}', \mathbf{m}', I', \mathbf{x}', \sigma')$:	Algorithm $D''(\mathcal{M}', \mathbf{m}', I', \mathbf{x}', \sigma')$:
If \mathcal{M}' is not (n_1, n_2) -half-uniform (for some n_1, n_2 that are both polynomial in κ), return 0	If \mathcal{M}' is not (n_1, n_2) -half-uniform (for some n_1, n_2 that are both polynomial in κ), return 1
If $I' \neq \{n_1 + 1, n_1 + 2, \dots, 2n_1\}$ for the n_1 determined by $\mathcal{M}_{S'}$, return 0	If $I' \neq \{n_1 + 1, n_1 + 2, \dots, 2n_1\}$ for the n_1 determined by $\mathcal{M}_{S'}$, return 1
Parse $\mathcal{M}' = \mathcal{M} \parallel (U_{n_2})^{n_1}$ and $\mathbf{m}' = \mathbf{m} \parallel \mathbf{m}_U$	Return 0
Return $D(\mathcal{M}, \mathbf{m}, \mathbf{x}', \sigma')$	

Proof. (of Lemma 2)

Note that **bad** occurs if and only if $\mathcal{M}_{S'}$ is not an (n_1, n_2) -half-uniform distribution, or $I_{S'} \neq \{n_1 + 1, n_1 + 2, \dots, 2n_1\}$ for the n_1 determined by $\mathcal{M}_{S'}$. Hence, whether **bad** occurs can be check in polynomial time. Consider the PPT algorithm D'' described in Table 5.

From equation (4), it is easy to see that $\Pr[D''(\text{Exp}_{\text{PKE}, A'}^{\text{SIM-NM-SO-CCA2-Real}}(\kappa)) = 1] = 0$.

We also notice that $\Pr[D''(\text{Exp}_{\text{PKE}, S'}^{\text{SIM-NM-SO-CCA2-Ideal}}(\kappa)) = 1] = \Pr[\text{bad}]$. Then, we have

$$\Pr[\text{bad}] = |\Pr[D''(\text{Exp}_{\text{PKE}, A'}^{\text{SIM-NM-SO-CCA2-Real}}(\kappa)) = 1] - \Pr[D''(\text{Exp}_{\text{PKE}, S'}^{\text{SIM-NM-SO-CCA2-Ideal}}(\kappa)) = 1]|.$$

Therefore, equation (6) guarantees that $\Pr[\text{bad}]$ is negligible. \square

\square

Remark 7. We can also prove Theorem 6 by simply constructing a “non-opening” SIM-NM-SO-ATK adversary, which is a copy of the SIM-NM-ATK adversary, and using the related SIM-NM-SO-ATK simulator as the SIM-NM-ATK simulator. Hence, our aforementioned proof actually shows that even considering constrained SIM-NM-SO-ATK adversary (i.e., “opening” adversary), Theorem 6 still holds. We note that all the simulation-based security notions (e.g., SIM-SO-ATK security) in this paper are described as “ $\text{Exp}_{\text{PKE}, A}^{\text{SIM-XX-ATK-Real}}(\kappa) \stackrel{c}{\approx} \text{Exp}_{\text{PKE}, S}^{\text{SIM-XX-ATK-Ideal}}(\kappa)$ ”, free of “relation R ”. For formal definitions of SIM-NM-ATK security (resp. SIM-SO-ATK security) defined with “relation R ”, we refer the readers to the papers [5] (resp. [4]). We note that if considering the simulation-based security notions described with “relation R ”, the conclusion of Theorem 6 might need to be reconsidered.

SIM-NM-ATK $\not\Rightarrow$ SIM-NM-SO-ATK. We will show that the IND-CCA2 secure Cramer-Shoup scheme [6][7] (the CS scheme) is SIM-NM-CCA2 secure. But the CS scheme is not SIM-SO-CPA secure [2]. According to Theorem 1, it is not SIM-NM-SO-CPA secure either. Consequently, “SIM-NM-ATK' $\not\Rightarrow$ SIM-NM-SO-ATK'”, for any $\text{ATK}', \text{ATK}'' \in \{\text{CPA}, \text{CCA1}, \text{CCA2}\}$.

To show that the CS scheme is SIM-NM-CCA2 secure, we use the following two facts: (1) For any PKE scheme having an invertible decryption algorithm, it is IND-NM-CCA2 secure iff it is SIM-NM-CCA2 secure [19, Theorem 6]. (2) IND-CCA2 security is equivalent to IND-NM-CCA2 security, since the parallel decryption query provides no additional ability to the adversary in the case of CCA2. So what remains is to show that the CS scheme has an invertible decryption algorithm. Let (Enc, Dec) denote the corresponding encryption/decryption algorithms. Following the notations of [7], any valid ciphertext ψ of the CS scheme has the form $\psi := (a, \hat{a}, c, d) \in G^4$, the message space is G , and the range of Dec is $G \cup \{\text{reject}\}$, where G is a group of prime order q (see [7]). We construct an inverting algorithm F as follows: On input $(pk, \text{Dec}(sk, \psi))$, if $\text{Dec}(sk, \psi) \in G$, then F runs $\text{Enc}(pk, \text{Dec}(sk, \psi))$ and returns the generated ciphertext; If $\text{Dec}(sk, \psi) = \text{reject}$, then F returns an arbitrary ciphertext not in G^4 .

7 Relations between IND-NM-SO securities and IND-NM securities

In this section, we explore the relations between IND-NM-SO securities and IND-NM securities. Our conclusions are as follows.

Theorem 7. (IND-NM-CCA2 $\not\Rightarrow$ IND-NM-SO-CCA2). *There is an IND-NM-CCA2 secure PKE scheme, which is not IND-NM-SO-CCA2 secure.*

Theorem 8. (IND-NM-SO-ATK \Rightarrow IND-NM-ATK). *For any $ATK \in \{CPA, CCA1, CCA2\}$, IND-NM-SO-ATK security implies IND-NM-ATK security.*

Notice that IND-NM-CCA2 (resp. IND-NM-SO-CCA2) security is equivalent to IND-CCA2 (resp. IND-SO-CCA2) security, so Theorem 7 is directly from [17], which separated IND-CCA2 security and IND-SO-CCA2 security.

The conclusion of Theorem 8 is not surprising at all. Intuitively, compared with the adversary considered in the notion of IND-NM security, the one considered in the notion of IND-NM-SO security is similar but more powerful. One subtlety here is that the ways that message vectors are sampled in these two notions are different. Due to space limitations, we provide the proof of this theorem in Appendix C.

Remark 8. In Section 5, we have showed that “IND-NM-SO-CPA $\not\Rightarrow$ IND-SO-CCA1” by utilizing scheme PKE” as a counterexample. With a similar analysis, it is easy to see that PKE” is not IND-NM-CCA1 secure. So we conclude that “IND-NM-SO-CPA $\not\Rightarrow$ IND-NM-CCA1”.

8 Relations between SIM-NM-SO securities and IND-NM-SO securities

In this section, we explore the relations between SIM-NM-SO securities and IND-NM-SO securities. Formally, we have the following conclusion. Its proof is similar to that of Theorem 2 and [19, Theorem 4], so we just provide a sketch here.

Theorem 9. (IND-NM-SO-CCA1/CCA2 $\not\Rightarrow$ SIM-NM-SO-CCA1/CCA2). *For any $ATK \in \{CCA1, CCA2\}$, there is an IND-NM-SO-ATK secure PKE scheme, which is not SIM-NM-SO-ATK secure.*

Proof. (Sketch) Let $PKE = (\text{Gen}, \text{Enc}, \text{Dec})$ be an IND-NM-SO-CCA1/CCA2 secure encryption scheme. We construct the scheme $\widetilde{PKE} = (\widetilde{\text{Gen}}, \widetilde{\text{Enc}}, \widetilde{\text{Dec}})$ described in Table 3. Note that in Section 4, we have shown that \widetilde{PKE} is not SIM-NM-SO-CCA1/CCA2 secure, and the reasoning there does not involve the security of the basic scheme PKE. So here we just need to prove that \widetilde{PKE} achieves IND-NM-SO-CCA1/CCA2 security.

For any PPT adversary \widetilde{A} attacking \widetilde{PKE} in the sense of IND-NM-SO-CCA1/CCA2 with non-negligible advantage, roughly speaking, we construct a PPT adversary A attacking PKE (in the sense of IND-NM-SO-CCA1/CCA2) as follows: Receiving the public key, A chooses $\theta \leftarrow \{0, 1\}^\kappa$, and uses this θ and its own decryption oracle to answer \widetilde{A} ’s decryption queries. A outputs the same message distribution \mathcal{M} as \widetilde{A} does, transforms any component $\mathbf{c}[i]$ of its own challenge ciphertext vector into $(\mathbf{c}[i], 1, 0^\kappa)$ to get a modified challenge ciphertext vector and passes the modified one to \widetilde{A} . A uses its own opening oracle to answer \widetilde{A} ’s opening query. Finally, A returns \widetilde{A} ’s final output. Notice that A perfectly simulates the IND-NM-SO-CCA1/CCA2 experiment (about \widetilde{PKE}) for \widetilde{A} . So A ’s advantage is also non-negligible, contradicting the assumption. \square

Remark 9. Note that \widetilde{PKE} is not SIM-NM-SO-CCA1 secure, *even if PKE is IND-NM-SO-CCA2 secure*. So we actually have a stronger conclusion: “IND-NM-SO-CCA2 $\not\Rightarrow$ SIM-NM-SO-CCA1”.

9 Constructions

Fortunately, there are some known selective opening secure PKE schemes achieving SIM/IND-NM-SO securities. Details are as follows.

SIM-NM-SO-CCA2 secure construction. The Fehr-Hofheinz-Kiltz-Wee encryption scheme (the FHKW scheme) is SIM-SO-CCA2 secure [11][15][16]. We claim that the decryption algorithm of the FHKW scheme is invertible, and the range of the decryption algorithm is recognizable. Hence, according to Theorem 3, the FHKW scheme is SIM-NM-SO-CCA2 secure. Our claim is justified as follows.

According to [11], any valid ciphertext of the FHKW scheme has the form (X_1, \dots, X_L, T) , and the message space is $\{0, 1\}^L$. For any ciphertext of the form (X_1, \dots, X_L, T) , where $X_i \in \mathcal{X}$ and $T \in \mathcal{XT}$, its decryption is an L -bit string. Since \mathcal{X} and \mathcal{XT} are both efficiently recognizable, any invalid ciphertext (X_1, \dots, X_L, T) (i.e., $X_i \notin \mathcal{X}$ for some i , or $T \notin \mathcal{XT}$) will be decrypted to \perp . In other words, the range of the decryption algorithm is $\{0, 1\}^L \cup \{\perp\}$, which is recognizable. As to the special inverting algorithm F , we construct it as follows: Let (Enc, Dec) denote the encryption/decryption algorithms of the FHKW scheme. For any ciphertext c , we have that $\text{Dec}(sk, c) \in \{0, 1\}^L \cup \{\perp\}$. If $\text{Dec}(sk, c) \in \{0, 1\}^L$, F runs $\text{Enc}(pk, \text{Dec}(sk, c))$ and returns the generated ciphertext; If $\text{Dec}(sk, c) = \perp$, F returns an arbitrary ciphertext (X_1, \dots, X_L, T) where $X_i \notin \mathcal{X}$ or $T \notin \mathcal{XT}$.

IND-NM-SO-CCA2 secure construction. According to Theorem 4, IND-NM-SO-CCA2 security is equivalent to IND-SO-CCA2 security. So any IND-SO-CCA2 secure encryption scheme (e.g. the PKE scheme constructed from all-but-many lossy trapdoor functions [12]) meets IND-NM-SO-CCA2 security.

Acknowledgments. The authors would like to thank the anonymous reviewers of ACNS 2015 for their helpful comments. This work is funded by the Specialized Research Fund for the Doctoral Program of Higher Education under Grant 20110073110016, the Scientific Innovation Projects of Shanghai Education Committee under Grant 12ZZ021, and the National Natural Science Foundation of China under Grants 61170229, 61373153, 61133014 and 61472114.

References

1. M. Bellare, A. Desai, D. Pointcheval and P. Rogaway. Relations among notions of security for public key encryption schemes. In: CRYPTO 1998. LNCS, vol. 1462, Springer, Heidelberg (1998)
2. M. Bellare, R. Dowsley, B. Waters and S. Yilek. Standard security does not imply security against selective-opening. In: Eurocrypt 2012. LNCS, vol. 7237, pp. 645-662. Springer, Heidelberg (2012)
3. F. Böhl, D. Hofheinz and D. Kraschewski. On definitions of selective opening security. In: PKC 2012. LNCS, vol. 7293, pp. 522-539. Springer (2012)
4. M. Bellare, D. Hofheinz and S. Yilek. Possibility and impossibility results for encryption and commitment secure under selective opening. In: Eurocrypt 2009. LNCS, vol. 5479, pp. 1-35. Springer, Heidelberg (2009)
5. M. Bellare and A. Sahai. Non-malleable encryption: Equivalence between two notions, and an indistinguishability-based characterization. In: CRYPTO 1999. LNCS, vol. 1666, pp. 519-536. Springer, Heidelberg (1999)
6. R. Cramer and V. Shoup. A practical public key cryptosystem provably secure against adaptive chosen ciphertext attack. In: CRYPTO 1998. Springer Berlin Heidelberg, 1998: 13-25.
7. R. Cramer and V. Shoup. Design and analysis of practical public-key encryption schemes secure against adaptive chosen ciphertext attack. In: SIAM Journal on Computing, 2003, 33(1): 167-226.

8. D. Dolev, C. Dwork and M. Naor. Non-Malleable Cryptography. In 23rd ACM Symposium on the Theory of Computing, 1991, pages 542-552.
9. D. Dolev, C. Dwork and M. Naor. Nonmalleable cryptography. SIAM J. Comput. 30(2), 391-437 (2000)
10. S. Goldwasser and S. Micali. Probabilistic encryption. In: Journal of computer and system sciences 28.2: 270-299 (1984)
11. S. Fehr, D. Hofheinz, E. Kiltz and H. Wee. Encryption schemes secure against chosen-ciphertext selective opening attacks. In: Eurocrypt 2010. LNCS, vol. 6110, pp. 381-402. Springer, Heidelberg (2010)
12. D. Hofheinz. All-but-many lossy trapdoor functions. In: Eurocrypt 2012. LNCS, vol. 7237, pp. 209-227. Springer, Heidelberg (2012)
13. J. Huang and X. Lai. Revisiting key schedules diffusion in relation with round functions diffusion. In: Designs Codes & Cryptography 73.1(2014):85-103.
14. B. Hemenway, B. Libert, R. Ostrovsky and D. Vergnaud. Lossy encryption: Constructions from general assumptions and efficient selective opening chosen ciphertext security. In: Asiacrypt 2011. LNCS. Springer (2011)
15. Z. Huang, S. Liu and B. Qin. Sender-Equivocable Encryption Schemes Secure against Chosen-Ciphertext Attacks Revisited. In: PKC 2013, pp. 369-385. Springer Berlin Heidelberg, 2013.
16. Z. Huang, S. Liu, B. Qin and K. Chen. Fixing the Sender-Equivocable Encryption Scheme in Eurocrypt 2010. In: Intelligent Networking and Collaborative Systems (INCoS), 2013 5th International Conference on. IEEE, 2013: 366-372.
17. D. Hofheinz and A. Rupp. Standard versus Selective Opening Security: Separation and Equivalence Results. In: TCC 2014. LNCS, vol. 8349, pp. 591-615. Springer, Heidelberg (2014)
18. M. Naor and M. Yung. Public-key cryptosystems provably secure against chosen ciphertext attacks. In: Proceedings of the twenty-second annual ACM symposium on Theory of computing, pp. 427-437. ACM, 1990.
19. R. Pass, A. Shelat and V. Vaikuntanathan. Relations among notions of non-malleability for encryption. In: ASIACRYPT 2007. pp. 519-535. Springer Berlin Heidelberg, 2007.
20. C. Rackoff and D. R. Simon. Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attack. In: CRYPTO 1991, pp. 433-444. Springer Berlin Heidelberg, 1992.

Appendix A. Proof of Theorem 2

Proof. (of Theorem 2.)

We prove that “SIM-SO-CCA2 security $\not\Rightarrow$ SIM-NM-SO-CCA2 security” as follows. The proof in the case of CCA1 is similar, and that in the case of CPA is obvious, which we will omit here.

Assuming that PKE is SIM-SO-CCA2 secure, for the new scheme $\widetilde{\text{PKE}}$ described in Section 4, we prove the following two lemmas. Lemma 3 claims that $\widetilde{\text{PKE}}$ achieves SIM-SO-CCA2 security. Lemma 4 claims that $\widetilde{\text{PKE}}$ is SIM-NM-SO-CCA2 insecure. Hence, we finish the proof of Theorem 2.

Lemma 3. $\widetilde{\text{PKE}}$ is SIM-SO-CCA2 secure.

Proof. (of Lemma 3)

For any PPT adversary $\widetilde{A} = (\widetilde{A}_1, \widetilde{A}_2, \widetilde{A}_3)$ attacking $\widetilde{\text{PKE}}$ in the sense of SIM-SO-CCA2, we show a PPT adversary $A = (A_1, A_2, A_3)$ attacking PKE in the sense of SIM-SO-CCA2 as follows.

Receiving a public key pk , A_1 samples $\theta \leftarrow \{0, 1\}^\kappa$, and sends $\widetilde{pk} = pk$ to \widetilde{A}_1 . For any decryption query $\widetilde{c} = (c, b, \vartheta)$ asked by \widetilde{A}_1 , A_1 answers like this:

Case 1. If $b = 0$ and $\vartheta = 1^\kappa$, then A_1 returns θ to \widetilde{A}_1 .

Case 2. If $b = 0$ and $\vartheta = \theta$, then A_1 returns \perp to \widetilde{A}_1 .

Case 3. If $b = 1$ and $\vartheta = 0^\kappa$, then A_1 sends c to its own decryption oracle. After receiving $m = \text{Dec}(sk, c)$, A_1 checks whether $m = \perp$. If $m = \perp$, then A_1 returns 0 to \widetilde{A}_1 ; else, it returns m .

Case 4. If (b, ϑ) is not in any of the aforementioned forms, then A_1 returns 0 to \widetilde{A}_1 .

When \widetilde{A}_1 finishes all its decryption queries, it returns a message distribution \mathcal{M} . Then A_1 outputs \mathcal{M} as its final output.

On the other side, the challenger chooses $\mathbf{m} \leftarrow \mathcal{M}$ and $\mathbf{r} \leftarrow (\mathcal{R}_{\text{Enc}})^{|\mathbf{m}|}$, and generates $\mathbf{c}^* \leftarrow \text{Enc}(pk, \mathbf{m}; \mathbf{r})$.

Upon receiving \mathbf{c}^* from the challenger, A_2 sets $\widetilde{\mathbf{c}}^*$ such that $|\widetilde{\mathbf{c}}^*| = |\mathbf{c}^*|$ and $\widetilde{\mathbf{c}}^*[i] := (\mathbf{c}^*[i], 1, 0^\kappa)$ for any $i \in [|\widetilde{\mathbf{c}}^*|]$. Then, A_2 sends $\widetilde{\mathbf{c}}^*$ to \widetilde{A}_2 . For any decryption query $\widetilde{c} = (c, b, \vartheta)$ asked by \widetilde{A}_2 , A_2 answers as A_1 does, with the only exception that if $\widetilde{c} \in \widetilde{\mathbf{c}}^*$, then A_2 returns ϵ . When \widetilde{A}_2 finishes all its decryption queries, it outputs a subset $I \subseteq [|\widetilde{\mathbf{c}}^*|]$. A_2 outputs I to the challenger.

Upon receiving $\widetilde{\mathbf{m}}[I]$ and $\mathbf{r}[I]$ from the challenger, \widetilde{A}_3 passes them to \widetilde{A}_3 . For any decryption query asked by \widetilde{A}_3 , A_3 answers as A_2 does. Finally, \widetilde{A}_3 returns $out_{\widetilde{A}}$. Then A_3 outputs $out_A := out_{\widetilde{A}}$.

That is the description of adversary A .

It is obvious that

$$\text{Exp}_{\text{PKE}, A}^{\text{SIM-SO-CCA2-Real}}(\kappa) = \text{Exp}_{\text{PKE}, \widetilde{A}}^{\text{SIM-SO-CCA2-Real}}(\kappa) = (\mathcal{M}, \mathbf{m}, I, out_{\widetilde{A}}). \quad (9)$$

Since PKE is SIM-SO-CCA2 secure, there is a simulator $S = (S_1, S_2, S_3)$, such that

$$\text{Exp}_{\text{PKE}, S}^{\text{SIM-SO-CCA2-Ideal}}(\kappa) \stackrel{c}{\approx} \text{Exp}_{\text{PKE}, A}^{\text{SIM-SO-CCA2-Real}}(\kappa). \quad (10)$$

Note that in the sense of SIM-SO-CCA2, a simulator does not receive any public key or ciphertext, and is not allowed to ask any decryption query either. Hence, simulator S in the ideal experiment for PKE can be used as a simulator in the ideal experiment for PKE. Therefore, setting $\widetilde{S} := S$, we have that

$$\text{Exp}_{\text{PKE}, \widetilde{S}}^{\text{SIM-SO-CCA2-Ideal}}(\kappa) = \text{Exp}_{\text{PKE}, S}^{\text{SIM-SO-CCA2-Ideal}}(\kappa). \quad (11)$$

Combining equations (9), (10) and (11), we have that

$$\text{Exp}_{\text{PKE}, \widetilde{A}}^{\text{SIM-SO-CCA2-Real}}(\kappa) \stackrel{c}{\approx} \text{Exp}_{\text{PKE}, \widetilde{S}}^{\text{SIM-SO-CCA2-Ideal}}(\kappa).$$

Therefore, $\widetilde{\text{PKE}}$ is SIM-SO-CCA2 secure. \square

Lemma 4. $\widetilde{\text{PKE}}$ is not SIM-NM-SO-CCA2 secure.

Proof. (of Lemma 4)

Consider an adversary $B = (B_1, B_2, B_3)$ as follows: After receiving the public key \widetilde{pk} , B_1 makes a decryption query $(c', 0, 1^\kappa)$, where c' is an arbitrary element in the ciphertext space. Then B_1 will receive θ . In the end of the real experiment $\text{Exp}_{\text{PKE}, B, R}^{\text{SIM-NM-SO-CCA2-Real}}(\kappa)$, B_3 returns $((c', 0, \theta), \sigma)$ as B 's final output, where σ is an arbitrary string. Since the decryption of $(c', 0, \theta)$ is \perp , we have that $\text{Exp}_{\text{PKE}, B}^{\text{SIM-NM-SO-CCA2-Real}}(\kappa) = (\mathcal{M}, \mathbf{m}, \perp, I, \sigma)$ for some $\mathcal{M}, \mathbf{m}, I$.

However, for any PPT simulator S , in the ideal experiment $\text{Exp}_{\text{PKE}, S}^{\text{SIM-NM-SO-CCA2-Ideal}}(\kappa)$ S can not access to the decryption oracle, which means that S has no information about θ . So the probability that S outputs a ciphertext whose decryption is \perp is $\frac{1}{2^\kappa}$.

Consider the distinguisher D : On input $(\mathcal{M}, \mathbf{m}, \mathbf{x}, I, \sigma)$, return 1 if and only if $\perp \in \mathbf{x}$ and $|\mathbf{x}| = 1$. Then we have that

$$\Pr[D(\text{Exp}_{\text{PKE}, B}^{\text{SIM-NM-SO-CCA2-Real}}(\kappa)) = 1] = 1,$$

and for any PPT simulator S ,

$$\Pr[D(\text{Exp}_{\text{PKE}, S}^{\text{SIM-NM-SO-CCA2-Ideal}}(\kappa)) = 1] = \frac{1}{2^\kappa}.$$

Therefore, $\widetilde{\text{PKE}}$ is not SIM-NM-SO-CCA2 secure. □

□

Appendix B. Proof of Theorem 5

Proof. (of Theorem 5)

The direction \Leftarrow . Let's prove that IND-NM-SO-CPA \Leftarrow IND-SO-CCA1.

Let $\text{PKE} = (\text{Gen}, \text{Enc}, \text{Dec})$ be an IND-SO-CCA1 secure encryption scheme with message space $\{0, 1\}^\kappa$. For any $m \in \{0, 1\}^\kappa$, let \bar{m} denote its bitwise complement. Similarly, for any $\mathbf{m} \in \{0, 1\}^{\kappa \times n}$, let $\bar{\mathbf{m}}$ denote the bitwise complement of \mathbf{m} (i.e., for $i \in [n]$, $\bar{\mathbf{m}}[i]$ is the bitwise complement of $\mathbf{m}[i]$). We construct a new scheme $\text{PKE}' = (\text{Gen}', \text{Enc}', \text{Dec}')$ as follows.

$\text{Gen}'(1^\kappa):$ $(pk, sk) \leftarrow \text{Gen}(1^\kappa)$ $pk' := pk$ $sk' := sk$ return (pk', sk')	$\text{Enc}'(pk', m):$ $c_1 \leftarrow \text{Enc}(pk, m)$ $c_2 \leftarrow \text{Enc}(pk, \bar{m})$ return $c := (c_1, c_2)$	$\text{Dec}'(sk', c):$ Parse $c = (c_1, c_2)$ $m = \text{Dec}(sk, c_1)$ return m
---	--	---

Similar to the proof of Theorem 2, for the new scheme PKE' , we prove the following two lemmas. Lemma 5 claims that PKE' achieves IND-SO-CCA1 security. Lemma 6 claims that PKE' is IND-NM-SO-CPA insecure. Hence, we finish the proof of **The direction \Leftarrow** .

Lemma 5. PKE' is IND-SO-CCA1 secure.

Proof. (of Lemma 5)

For any PPT adversary $A' = (A'_1, A'_2, A'_3)$ attacking PKE' in the sense of IND-SO-CCA1, we show a PPT adversary $A = (A_1, A_2, A_3)$ attacking PKE in the sense of IND-SO-CCA1 as follows.

Receiving a public key pk , A_1 passes $pk' = pk$ to A'_1 . For any decryption query $c' := (c'_1, c'_2)$ asked by A'_1 , A_1 sends c'_1 to its own decryption oracle, and passes the message, returned from the oracle, to A'_1 . When A'_1 finishes all its decryption queries, it returns a message distribution \mathcal{M}' and a PPT algorithm $\text{Resamp}'_{\mathcal{M}'}(\cdot, \cdot)$. Without loss of generality, we assume that $n = |\mathbf{m}'|$ for any $\mathbf{m}' \leftarrow \mathcal{M}'$. A_1 sets the following new message distribution \mathcal{M} :

$$\mathbf{m} \leftarrow \mathcal{M} \text{ means that } \mathbf{m}' \leftarrow \mathcal{M}', \mathbf{m} := \mathbf{m}' \parallel \bar{\mathbf{m}}' .$$

A_1 also sets a new PPT algorithm $\text{Resamp}_{\mathcal{M}}(\cdot, \cdot)$ related to \mathcal{M} as follows:

$\text{Resamp}_{\mathcal{M}}(I, \mathbf{m}[I]):$
 Set $I' := I \cap [n]$
 For $i \in I \setminus I'$,
 if $i \bmod n \notin I$,
 set $\mathbf{m}[i \bmod n] = \overline{\mathbf{m}}[i]$ (i.e., the bitwise complement of $\mathbf{m}[i]$), and $I' = I' \cup \{i \bmod n\}$
 $\mathbf{x} \leftarrow \text{Resamp}'_{\mathcal{M}'}(I', \mathbf{m}[I'])$
 Set $\mathbf{m}_{resm} := \mathbf{x} \parallel \overline{\mathbf{x}}$
 Return \mathbf{m}_{resm}

Then, A_1 outputs $(\mathcal{M}, \text{Resamp}_{\mathcal{M}}(\cdot, \cdot))$.

On the other side, the challenger samples $b \leftarrow \{0, 1\}$, $\mathbf{m}_0 \leftarrow \mathcal{M}$, $\mathbf{r} \leftarrow (\mathcal{R}_{\text{Enc}})^{2n}$, and generates a challenge ciphertext vector, $\mathbf{c} \leftarrow \text{Enc}(pk, \mathbf{m}_0; \mathbf{r})$.

Receiving \mathbf{c} from the challenger, A_2 sets that $\mathbf{c}^* := ((\mathbf{c}[i], \mathbf{c}[i+n])_{i \in [n]})$ and sends \mathbf{c}^* to A'_2 . The latter returns a subset $I' \subseteq [n]$. A_2 sets that $I := I' \cup \{i+n \mid i \in I'\}$, and outputs I .

Upon receiving a message vector \mathbf{m}_b and $\mathbf{r}[I]$ from the challenger, A_3 parses $\mathbf{m}_b := \mathbf{m}_b^{(1)} \parallel \mathbf{m}_b^{(2)}$, where $|\mathbf{m}_b^{(1)}| = |\mathbf{m}_b^{(2)}| = n$. A_3 also sets that $\mathbf{r}'[I'] := ((\mathbf{r}[i], \mathbf{r}[i+n])_{i \in I'})$. Finally, A_3 sends $(\mathbf{m}_b^{(1)}, \mathbf{r}'[I'])$ to A'_3 , and outputs what A'_3 returns.

That is the construction of adversary A .

Note that A perfectly simulates the experiment $\text{Exp}_{\text{PKE}', A'}^{\text{IND-SO-CCA1-}b}(\kappa)$ ($b \in \{0, 1\}$) for A' , and A succeeds in outputting $b' = b$ if and only if A' also succeeds. So we have that

$$\text{Adv}_{\text{PKE}', A}^{\text{IND-SO-CCA1}}(\kappa) = \text{Adv}_{\text{PKE}', A'}^{\text{IND-SO-CCA1}}(\kappa).$$

□

Lemma 6. PKE' is not IND-NM-SO-CPA secure.

Proof. (of Lemma 6)

We show a PPT adversary $B = (B_1, B_2)$, attacking PKE' in the sense of IND-NM-SO-CPA, as follows.

At first, receiving a public key pk' , B_1 returns a uniform distribution \mathcal{M} over $\{0, 1\}^{2\kappa}$ (since \mathcal{M} is a uniform distribution, the related PPT algorithm $\text{Resamp}_{\mathcal{M}}(\cdot, \cdot)$ is obvious, which we omit here). The challenger chooses $\mathbf{m}_0 \leftarrow \mathcal{M}$, generates a challenge ciphertext vector $\mathbf{c} \leftarrow \text{Enc}'(pk', \mathbf{m}_0)$, and sends \mathbf{c} to B_2 . Specifically,

$$\mathbf{c} = (\mathbf{c}[1], \mathbf{c}[2]) := ((\mathbf{c}[1]^1, \mathbf{c}[1]^2), (\mathbf{c}[2]^1, \mathbf{c}[2]^2)),$$

where $(\mathbf{c}[i]^1, \mathbf{c}[i]^2) = \text{Enc}'(pk', \mathbf{m}_0[i]) = (\text{Enc}(pk', \mathbf{m}_0[i]), \text{Enc}(pk', \overline{\mathbf{m}_0}[i]))$ for $i \in [2]$.

For the opening query, B_2 outputs $I := \{1\}$. The challenger samples $b \leftarrow \{0, 1\}$, $\mathbf{m}_1[2] \leftarrow \{0, 1\}^\kappa$, sets that

$$\mathbf{m}_1 = (\mathbf{m}_1[1], \mathbf{m}_1[2]) := (\mathbf{m}_0[1], \mathbf{m}_1[2]),$$

and returns \mathbf{m}_b and the randomness, $\mathbf{r}[1]$, used in the process of $\text{Enc}'(pk', \mathbf{m}_0[1])$.

For the parallel decryption query, B_2 outputs $c' := (\mathbf{c}[2]^2, \mathbf{c}[2]^1)$. Note that c' is the encryption of $\overline{\mathbf{m}_0}[2]$. So if $c' \notin \mathbf{c}$, B_2 will receive $\overline{\mathbf{m}_0}[2]$. Then B_2 can recover $\mathbf{m}_0[2]$ and compare it with $\mathbf{m}_b[2]$. If $c' \in \mathbf{c}$, B_2 will receive ϵ . Finally, B_2 outputs 0 if and only if B_2 receives $\overline{\mathbf{m}_0}[2]$ and $\mathbf{m}_0[2] = \mathbf{m}_b[2]$.

That is the construction of adversary B .

Since $c' \in \mathbf{c}$ if and only if $\overline{\mathbf{m}_0}[2] = \mathbf{m}_0[1]$, we have $\Pr[c' \in \mathbf{c}] = \frac{1}{2^\kappa}$.

In the case of $c' \notin \mathbf{c}$, it is easy to see that when $b = 0$, B_2 will definitely output 0; when $b = 1$, B_2 will output 1 except for the case $\mathbf{m}_0[2] = \mathbf{m}_1[2]$.

Hence, B 's advantage

$$\begin{aligned}
 \mathbf{Adv}_{\text{PKE}', B}^{\text{IND-NM-SO-CPA}}(\kappa) &= |\Pr[\text{Exp}_{\text{PKE}', B}^{\text{IND-NM-SO-CPA-1}}(\kappa) = 1] - \Pr[\text{Exp}_{\text{PKE}', B}^{\text{IND-NM-SO-CPA-0}}(\kappa) = 1]| \\
 &= |(\Pr[\text{Exp}_{\text{PKE}', B}^{\text{IND-NM-SO-CPA-1}}(\kappa) = 1 \mid c' \in \mathbf{c}] \cdot \Pr[c' \in \mathbf{c}] \\
 &\quad + \Pr[\text{Exp}_{\text{PKE}', B}^{\text{IND-NM-SO-CPA-1}}(\kappa) = 1 \mid c' \notin \mathbf{c}] \cdot \Pr[c' \notin \mathbf{c}]) \\
 &\quad - (\Pr[\text{Exp}_{\text{PKE}', B}^{\text{IND-NM-SO-CPA-0}}(\kappa) = 1 \mid c' \in \mathbf{c}] \cdot \Pr[c' \in \mathbf{c}] \\
 &\quad + \Pr[\text{Exp}_{\text{PKE}', B}^{\text{IND-NM-SO-CPA-0}}(\kappa) = 1 \mid c' \notin \mathbf{c}] \cdot \Pr[c' \notin \mathbf{c}])| \\
 &= |(\frac{1}{2^\kappa} \cdot 1 + (1 - \frac{1}{2^\kappa})^2) - (\frac{1}{2^\kappa} \cdot 1 + (1 - \frac{1}{2^\kappa}) \cdot 0)| \\
 &= (1 - \frac{1}{2^\kappa})^2,
 \end{aligned}$$

which is overwhelming. \square

The direction \nRightarrow . Let's prove that $\text{IND-NM-SO-CPA} \nRightarrow \text{IND-SO-CCA1}$.

Let $\text{PKE} = (\text{Gen}, \text{Enc}, \text{Dec})$ be an IND-NM-SO-CPA secure encryption scheme. We construct a new scheme $\text{PKE}'' = (\text{Gen}'', \text{Enc}'', \text{Dec}'')$ as follows.

$\text{Gen}''(1^\kappa)$:

$(pk, sk) \leftarrow \text{Gen}(1^\kappa)$
 $\theta \leftarrow \{0, 1\}^\kappa$
 $pk'' := pk$
 $sk'' := (sk, \theta)$
 return (pk'', sk'')

$\text{Dec}''(sk'', c'')$:

Parse $c'' = (c, b, \vartheta)$
 If $b = 0$ and $\vartheta = 1^\kappa$, then return θ
 If $b = 0$ and $\vartheta = \theta$, then return sk
 If $b = 1$ and $\vartheta = 0^\kappa$, then return $m = \text{Dec}(sk, c)$
 Otherwise, return COPY

$\text{Enc}''(pk'', m)$:

$c \leftarrow \text{Enc}(pk, m)$
 return $c'' := (c, 1, 0^\kappa)$

Similar to the proof of Theorem 2, assuming that PKE is IND-NM-SO-CPA secure, we prove the following two lemmas. Lemma 7 claims that PKE'' achieves IND-NM-SO-CPA security. Lemma 8 claims that PKE'' is IND-SO-CCA1 insecure. Hence, we finish the proof of **The direction \nRightarrow** .

Lemma 7. PKE'' is IND-NM-SO-CPA secure.

Proof. (of Lemma 7)

For any PPT adversary $A'' = (A''_1, A''_2)$ attacking PKE'' in the sense of IND-NM-SO-CPA , we show a PPT adversary $A = (A_1, A_2)$ attacking PKE in the sense of IND-NM-SO-CPA as follows.

Receiving a public key pk , A_1 samples $\theta \leftarrow \{0, 1\}^\kappa$, and sends $pk'' = pk$ to A''_1 . Then A_1 outputs the tuple $(\mathcal{M}, \text{Resamp}_{\mathcal{M}}(\cdot, \cdot))$ returned by A''_1 .

On the other side, the challenger samples $b \leftarrow \{0, 1\}$, $\mathbf{m}_0 \leftarrow \mathcal{M}$, $\mathbf{r} \leftarrow (\mathcal{R}_{\text{Enc}})^{|\mathbf{m}_0|}$, and generates a challenge ciphertext vector, $\mathbf{c} \leftarrow \text{Enc}(pk, \mathbf{m}_0; \mathbf{r})$.

After receiving \mathbf{c} from the challenger, A_2 generates a new ciphertext vector $\mathbf{c}'' := ((\mathbf{c}[i], 1, 0^\kappa)_{i \in [|\mathbf{c}|]})$, and passes \mathbf{c}'' to A_2'' . A_2 answers A_2'' 's opening query and parallel decryption query as follows:

- **Opening query:** Upon receiving $I \subseteq [|\mathbf{c}''|]$ from A_2'' , A_2 submits I to its opening oracle $\text{Open}_{b, \mathcal{M}, \mathbf{m}_0, \mathbf{r}}(\cdot)$. Then A_2 returns the received tuple $(\mathbf{m}_b, \mathbf{r}[I])$ to A_2'' .
- **Parallel decryption query:** Upon receiving \mathbf{y}'' , for $i \in [|\mathbf{y}''|]$, A_2 parses $\mathbf{y}''[i] = (y'_i, b_i, \vartheta_i)$, where $b_i \in \{0, 1\}$ and $\vartheta_i \in \{0, 1\}^\kappa$. A_2 generates a new decryption query \mathbf{y} , such that $|\mathbf{y}| = |\mathbf{y}''|$ and for $i \in [|\mathbf{y}''|]$: if $b_i = 1$ and $\vartheta_i = 0^\kappa$, then $\mathbf{y}[i] = y'_i$; otherwise, $\mathbf{y}[i] = \mathbf{c}[1]$. Then, A_2 submits \mathbf{y} as its own parallel decryption query. Receiving the decryption \mathbf{x} , A_2 resets \mathbf{x} as follows: For $i \in [|\mathbf{x}|]$, if $b_i = 0$ and $\vartheta_i = 1^\kappa$, then reset $\mathbf{x}[i] = \theta$. After that, A_2 sends \mathbf{x} to A_2'' .

Eventually, A_2 outputs the bit b' , returned by A_2'' , as its final output.

That is the construction of A . Now we analyze A 's advantage.

Let bad denote the event that in the parallel decryption query \mathbf{y}'' made by A'' , there exists some $i \in [|\mathbf{y}''|]$, such that $b_i = 0$ and $\vartheta_i = \theta$. Notice that for any $b \in \{0, 1\}$, the experiment simulated by A is identical to $\text{Exp}_{\text{PKE}'', A''}^{\text{IND-NM-SO-CPA-}b}(\kappa)$ except that bad occurs. Since θ is uniformly random chosen from $\{0, 1\}^\kappa$ by A , and that A'' have no information about θ before its parallel decryption query, the probability that bad occurs is $\Pr[\text{bad}] \leq \frac{l}{2^\kappa}$, where $l := |\mathbf{y}''|$.

Hence, we have that for any $b \in \{0, 1\}$,

$$|\Pr[\text{Exp}_{\text{PKE}', A}^{\text{IND-NM-SO-CPA-}b}(\kappa) = 1] - \Pr[\text{Exp}_{\text{PKE}'', A''}^{\text{IND-NM-SO-CPA-}b}(\kappa) = 1]| \leq \Pr[\text{bad}] \leq \frac{l}{2^\kappa}.$$

Because

$$\begin{aligned} \text{Adv}_{\text{PKE}'', A''}^{\text{IND-NM-SO-CPA}}(\kappa) &= |\Pr[\text{Exp}_{\text{PKE}'', A''}^{\text{IND-NM-SO-CPA-1}}(\kappa) = 1] - \Pr[\text{Exp}_{\text{PKE}'', A''}^{\text{IND-NM-SO-CPA-0}}(\kappa) = 1]| \\ &\leq |\Pr[\text{Exp}_{\text{PKE}'', A''}^{\text{IND-NM-SO-CPA-1}}(\kappa) = 1] - \Pr[\text{Exp}_{\text{PKE}', A}^{\text{IND-NM-SO-CPA-1}}(\kappa) = 1]| \\ &\quad + |\Pr[\text{Exp}_{\text{PKE}', A}^{\text{IND-NM-SO-CPA-1}}(\kappa) = 1] - \Pr[\text{Exp}_{\text{PKE}', A}^{\text{IND-NM-SO-CPA-0}}(\kappa) = 1]| \\ &\quad + |\Pr[\text{Exp}_{\text{PKE}', A}^{\text{IND-NM-SO-CPA-0}}(\kappa) = 1] - \Pr[\text{Exp}_{\text{PKE}'', A''}^{\text{IND-NM-SO-CPA-0}}(\kappa) = 1]| \\ &\leq \text{Adv}_{\text{PKE}', A}^{\text{IND-NM-SO-CPA}}(\kappa) + \frac{2l}{2^\kappa}, \end{aligned}$$

we have that

$$\text{Adv}_{\text{PKE}', A}^{\text{IND-NM-SO-CPA}}(\kappa) \geq \text{Adv}_{\text{PKE}'', A''}^{\text{IND-NM-SO-CPA}}(\kappa) - \frac{2l}{2^\kappa}.$$

□

Lemma 8. *PKE'' is not IND-SO-CCA1 secure.*

Proof. (of Lemma 8)

Consider an adversary $A = (A_1, A_2, A_3)$ as follows: After receiving the public key pk'' , A_1 makes a decryption query $(c', 0, 1^\kappa)$, where c' is an arbitrary element in the ciphertext space. After receiving θ , A_1 makes another decryption query $(c', 0, \theta)$. Then A_1 will receive the original secret key sk . With sk , the adversary can decrypt any challenge ciphertext vector \mathbf{c} on its own. So PKE'' is not IND-SO-CCA1 secure. □

□

Appendix C. Proof of Theorem 8

Proof. (of Theorem 8.)

We prove that “IND-NM-SO-CCA2 security \Rightarrow IND-NM-CCA2 security”. The proof in the case of CPA/CCA1 is similar, which we will omit here.

Let $\text{PKE} = (\text{Gen}, \text{Enc}, \text{Dec})$ be an arbitrary IND-NM-SO-CCA2 secure encryption scheme. For any PPT adversary $A = (A_1, A_2, A_3)$ attacking PKE in the sense of IND-NM-CCA2, we construct a PPT adversary $A' = (A'_1, A'_2)$, attacking PKE in the sense of IND-NM-SO-CCA2, as follows.

Receiving a public key pk , A'_1 runs A_1 on the input of pk . For any decryption query c' asked by A_1 , A'_1 sends it to its own decryption oracle, and then returns the answer to A_1 . At some point, A_1 finishes its decryption query, and returns two distinct message vectors $\mathbf{m}_0, \mathbf{m}_1$ with the same length. Let $n_1 := |\mathbf{m}_0| = |\mathbf{m}_1|$, and n_2 be an integer polynomial in κ . A'_1 sets that \mathcal{M} is a uniform distribution over the set $\{\mathbf{m}_0, \mathbf{m}_1\}$, and $\mathcal{M}' := \mathcal{M} \parallel (U_{n_2})^{n_1}$. A'_1 outputs \mathcal{M}' to the challenger. (The re-sampling algorithm $\text{Resamp}_{\mathcal{M}'}$ for \mathcal{M}' is obvious, so we omit it here.)

On the other side, the challenger chooses $\mathbf{m}'_0 \leftarrow \mathcal{M}'$ (i.e., samples $b_0 \leftarrow \{0, 1\}$, $\mathbf{m}_U \leftarrow (U_{n_2})^{n_1}$, and sets $\mathbf{m}'_0 := \mathbf{m}_{b_0} \parallel \mathbf{m}_U$) and $\mathbf{r} \leftarrow (\mathcal{R}_{\text{Enc}})^{2n_1}$, and generates $\mathbf{c}^* \leftarrow \text{Enc}(pk, \mathbf{m}'_0; \mathbf{r})$.

Upon receiving \mathbf{c}^* from the challenger, A'_2 outputs $I := \{n_1 + 1, n_1 + 2, \dots, 2n_1\}$ as its opening query.

The challenger chooses $\beta \leftarrow \{0, 1\}$ and $\mathbf{m}'_1 \leftarrow \mathcal{M}'|_{I, \mathbf{m}'_0[I]}$ (i.e., samples $b_1 \leftarrow \{0, 1\}$, and sets that $\mathbf{m}'_1 := \mathbf{m}_{b_1} \parallel \mathbf{m}'_0[I] = \mathbf{m}_{b_1} \parallel \mathbf{m}_U$).

Upon receiving $\mathbf{m}'_\beta = \mathbf{m}_{b_\beta} \parallel \mathbf{m}_U$ from the challenger, A'_2 parses $\mathbf{c}^* = \mathbf{c}_A \parallel \mathbf{c}_U$, such that $|\mathbf{c}_A| = |\mathbf{c}_U| = n_1$. Then, A'_2 runs A_2 on the input of \mathbf{c}_A . For any decryption query c' asked by A_2 , if $c' \notin \mathbf{c}_U$, A'_2 answers it with its own decryption oracle; otherwise, A'_2 answers it with \mathbf{m}_U . At some point, A_2 returns (\mathbf{y}, σ) . A'_2 submits \mathbf{y} to its own parallel decryption oracle $P_{sk}(\cdot)$, obtaining $\mathbf{x} = P_{sk}(\mathbf{y})$. Let $\mathcal{S} := \{i \in [|\mathbf{y}|] \mid \mathbf{y}[i] \in \mathbf{c}_U\}$. If $\mathcal{S} \neq \emptyset$, then for each $i \in \mathcal{S}$, reset $\mathbf{x}[i]$ with \mathbf{m}_U (i.e., for the index j such that $\mathbf{c}_U[j] = \mathbf{y}[i]$, set $\mathbf{x}[i] \leftarrow \mathbf{m}_U[j]$). Then, A'_2 runs A_3 on the input of (\mathbf{x}, σ) . For any decryption query asked by A_3 , A'_2 answers it as before. At some point, A_3 returns its final output b' . If $\mathbf{m}_{b'} = \mathbf{m}'_\beta[[n_1]]$, A'_2 outputs $\beta' = 0$; otherwise, A'_2 outputs $\beta' = 1$.

That is the description of adversary A' .

Now we analyze A' 's advantage.

For convenience, we define a “new” experiment. $\text{Exp}_{\text{PKE}, A}^{\text{IND-NM-ATK}}(\kappa)$ is identical to $\text{Exp}_{\text{PKE}, A}^{\text{IND-NM-ATK-}b}(\kappa)$ in Definition 2, except that b is uniformly chosen from $\{0, 1\}$ instead of being fixed before hand, and the final output of the experiment is $(b' = b)$, not b' . It is easy to see that A' advantage can be rewritten as

$$\text{Adv}_{\text{PKE}, A}^{\text{IND-NM-ATK}}(\kappa) := |2\Pr[\text{Exp}_{\text{PKE}, A}^{\text{IND-NM-ATK}}(\kappa) = 1] - 1|.$$

Before considering the probabilities, we point out some useful facts. Firstly, \mathbf{c}_A is the encryption of \mathbf{m}_{b_0} , no matter what the value of β is. Secondly, $\mathbf{m}'_\beta[[n_1]] = \mathbf{m}_{b_\beta}$, so $\mathbf{m}_{b'} = \mathbf{m}'_\beta[[n_1]]$ if and only if $b' = b_\beta$ (note that $\mathbf{m}_0 \neq \mathbf{m}_1$).

In the case of $\beta = 1$, we note that A has no information about b_1 , since $b_1 \leftarrow \{0, 1\}$ is independent of A 's view. Thus, the probability that $\mathbf{m}_{b'} = \mathbf{m}'_\beta[[n_1]]$ (i.e., $b' = b_1$) is $\frac{1}{2}$. Hence,

$$\Pr[\text{Exp}_{\text{PKE}, A'}^{\text{IND-NM-SO-CCA2-1}}(\kappa) = 1] = 1 - \frac{1}{2} = \frac{1}{2}.$$

In the case of $\beta = 0$, the probability $\mathbf{m}_{b'} = \mathbf{m}'_{\beta}[[n_1]]$ (i.e., $b' = b_0$) is actually the probability that A points out the encrypted message vector from the challenge ciphertext. In the case of $\beta = 1$, since A' perfectly simulates the experiment $\text{Exp}_{\text{PKE},A}^{\text{IND-NM-CCA2-}b_0}(\kappa)$ for A , and $b_0 \leftarrow \{0, 1\}$, we have that the probability $\mathbf{m}_{b'} = \mathbf{m}'_0[[n_1]]$ is equal to $\Pr[\text{Exp}_{\text{PKE},A}^{\text{IND-NM-CCA2}}(\kappa) = 1]$. In other words,

$$\Pr[\text{Exp}_{\text{PKE},A'}^{\text{IND-NM-SO-CCA2-0}}(\kappa) = 1] = 1 - \Pr[\text{Exp}_{\text{PKE},A}^{\text{IND-NM-CCA2}}(\kappa) = 1].$$

Therefore, A' 's advantage is

$$\begin{aligned} \mathbf{Adv}_{\text{PKE},A'}^{\text{IND-NM-SO-CCA2}}(\kappa) &= |\Pr[\text{Exp}_{\text{PKE},A'}^{\text{IND-NM-SO-CCA2-1}}(\kappa) = 1] - \Pr[\text{Exp}_{\text{PKE},A'}^{\text{IND-NM-SO-CCA2-0}}(\kappa) = 1]| \\ &= |1 - \Pr[\text{Exp}_{\text{PKE},A}^{\text{IND-NM-CCA2}}(\kappa) = 1] - \frac{1}{2}| \\ &= \frac{1}{2} \mathbf{Adv}_{\text{PKE},A}^{\text{IND-NM-CCA2}}(\kappa). \end{aligned}$$

□

Note. We provide the proof which applies to each case of $\text{ATK} \in \{\text{CPA}, \text{CCA1}, \text{CCA2}\}$. Actually, if we only consider the case of CCA2, the conclusion is obvious. Because IND-NM-SO-CCA2 security is equivalent to IND-SO-CCA2 security, and IND-NM-CCA2 security is equivalent to IND-CCA2 security.