Deny Whatever You Want: Dual-Deniable Public-Key Encryption

Zhiyuan An[1,2] and Fangguo Zhang[1,2](✉)

[1] School of Computer Science and Engineering, Sun Yat-sen University,
Guangzhou 510006, China
anzhy@mail2.sysu.edu.cn,isszhfg@mail.sysu.edu.cn

[2] Guangdong Provincial Key Laboratory of Information Security Technology,
Guangzhou 510006, China

Abstract. We introduce an enhanced requirement of deniable public key encryption that we call *dual-deniability*. It asks that a sender who is coerced should be able to produce fake randomness, which can explain the target ciphertext as the encryption of any alternative message under any valid key she/he desires to deny. Compared with the original notion of deniability (Canetti et al. in CRYPTO '97, hereafter named message-deniability), this term further provides a shield for the anonymity of the receiver against coercion attacks.

We first give a formal definition of dual-deniability, along with its weak-mode variant. For conceptual understanding, we then show dual-deniability implies semantic security and anonymity against CPA, separates full robustness, and even contradicts key-less or mixed robustness, while is (constructively) implied by key-deniability and full robustness with a minor assumption for bits encryption. As for the availability of dual-deniability, our main scheme is a generic approach from ciphertext-simulatable PKE, where we devise a subtle multi-encryption schema to hide the true message within random masking ciphertexts under plan-ahead setting. Further, we leverage the weak model to present a more efficient scheme having negligible detection probability and constant ciphertext size. Besides, we revisit the notable scheme (Sahai and Waters in STOC '14) and show it is inherently dual-deniable. Finally, we extend the Boneh-Katz transform to capture CCA security, deriving dual-deniable and CCA-secure PKE from any selectively dual-deniable IBE under multi-TA setting. Overall our work mounts the feasibility of anonymous messaging against coercion attacks.

Keywords: Deniable encryption · Anonymity· Key privacy · Generic Construction · CCA security

1 Introduction

Deniable Encryption, introduced by Canetti et al. [10], is a seemingly contradictory primitive, which allows a sender or receiver to freely produce fake but convincing random coins interpreting the original ciphertext into any alternative message, and so makes post-coercion or bribery attacks meaningless. Compared with the common semantic security under chosen-plaintext attacks, deniability is a stronger requirement of data privacy against more hostile threats. In this sense, deniable public key encryption is a key tool for building adaptively secure cryptosystems, e.g., incoercible electronic voting [11, 16], multiparty computation [10], and searchable encryption [15].

Anonymous Encryption. Besides the concern of data security, anonymity (or key privacy) is also of crucial importance when protecting user's privacy across Internet applications. In the area of PKE, anonymity ensures that a valid ciphertext leaks nothing about the public key used to create it, i.e., the receiver is anonymous from the view of any CPA/CCA adversary. This concept was first formalized by Bellare et al. [4], where they also termed it as *indistinguishability of keys* (IK). A hand-in-hand requirement with anonymity is *robustness* [1], which keeps clear of miscommunication on anonymous encryption by guaranteeing the ciphertexts against being valid under different recipients' keys. Namely, one can decrypt the ciphertext using her/his secret key, and learn that it is indeed for her/himself iff the decryption result is not a failure.

Current Progress. There have been many advances towards the two directions. The seminal work [10] introduced several deniable flavors such as sender, receiver, or bi-deniability. Following work further extends the functionalities: O'Neill et al. [33] considered non-interactive bi-deniability; Sahai and Waters [35] introduced public deniability where faking does not require the original randomness; Canetti et al. [13] further considered full deniability where both parties are simultaneously coerced; Agrawal et al. [2] also studied the concept of deniable fully homomorphic encryption. For anonymity, a sufficient condition starting from semantic security is given in [28]. It has also been proved in [4, 27, 31] that some known PKE schemes, e.g., ElGamal, Cramer-Shoup, and Kyber KEM, satisfy IK-CPA/CCA. Besides, some serial work [1, 23, 26, 27] also provided several generic or scheme-oriented transforms that confer robustness for anonymity.

However, to the best of our knowledge, all the existing works on deniability only scope data privacy, and none of the known anonymous encryption schemes is resilient to coercion attacks. Below we elaborate more on this gap.

Motivation. Let's reconsider the issue in [10]: some adversary *Eve* has the power to approach a sender *Alice* after seeing a ciphertext was transmitted from her host, and demand to see all the private information: the plaintext, public key (of receiver *Bob*), and random coins used for encryption (assuming that *Alice* is unable to erase these records). The known message-deniable PKE can only protect the data privacy under such setting, i.e., *Alice* could produce fake randomness related to fake plaintext but has to honestly reveal the used public key, and so *Eve* learns that it is *Bob* who has communicated with *Alice*.

We stress that such enhanced coercion attacks bring much damage to the anonymous systems, e.g., the notable cryptocurrency framework Zcash [5] or auction systems [36], as the security of all these applications highly depends on the assumption that no one can identify the receiver of some involved ciphertext from the valid users. Note that applying key-private encryption cannot help, since it only enables the ciphertext to be unlinkable from the receiver (public key), but has no guarantee of the indistinguishability between the sender's identities. In this way, a coercer can still trace the anonymous ciphertext back to its initiator, and then force *Alice* to disclose *Bob*. Even worse, *Alice* may later be bribed and so will volunteer to sell out *Bob*. The above unsatisfactory situation motivates us to think about the following question:

Can we achieve deniability regarding both data and key privacy?

Our Contributions. We tackle the above problem positively, our contributions are summarized as follows.

- We formalize a new primitive: *dual-deniable* public key encryption, where the sender can deny both the plaintext and the public key of a ciphertext. Then we explore its relations with other security notions (Fig. 1), showing it implies IND/IK-CPA, but contradicts key-less or mixed robustness, and can be built from key deniability plus full robustness with a minor assumption.
- We present two generic constructions of dual-deniable public key encryption:
  - The first is the main scheme, being a transform from any ciphertext-simulatable PKE and one-way function, under plan-ahead setting.
  - The second is a simplified variant of the first concerning a weak model, having negligible detection probability and constant ciphertext size.
- We prove the known $i\mathcal{O}$-based scheme is inherently dual-deniable, and provide an extended BK conversion of dual-deniable and IK/IND-CCA PKE, from any dual-deniable IBE.

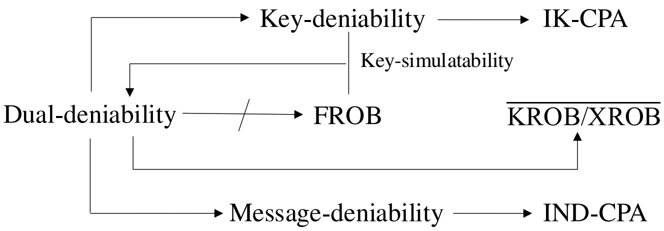

Fig. 1. Relations among notions. $\overline{\text{A}}$ is the negative of A; Arrow $\rightarrow$ (resp., barred arrow $\nrightarrow$, extended arrow $\mapsto$) is an implication (resp., separation, constructive implication).

Overview of Our Techniques. In the following, we provide more technical details of our contributions.

Concept of Dual-Deniability. Dual-deniable public key encryption (DDPKE) supports a cooking algorithm $\mathsf{Fake}(\mathsf{pk}, m, r, \mathsf{pk}^*, m^*) \rightarrow r^*$, by which the sender can claim a real ciphertext ct of m under pk using coins r as the encryption of m^* under pk^* using r^*. The coercer can rerun the encryption to verify such fake confession of ct, while ct can still be correctly decrypted using sk. For provable security, we require the following computational indistinguishability (Def. 10):

$$D_0 := (\mathsf{pk}^*, m^*, r, \mathsf{Enc}(\mathsf{pk}^*, m^*; r)) \overset{\mathsf{c}}{\approx} D_1 := (\mathsf{pk}^*, m^*, r^*, \mathsf{Enc}(\mathsf{pk}, m; r)),$$

where the left is the honest encryption of m^* under pk^*, and the right is the fake opening of the ciphertext for m under pk. Also, we adopt the *plan-ahead* setting widely considered in [2,10,33], where the fake pair (pk^*, m^*) is determined at the beginning of encryption (Def. 12). Namely, algorithm Enc (resp., Fake) extends to include such pair as auxiliary input (resp., output), leading to a relaxed demand of indistinguishability between the fake opening of $\mathsf{Enc}(\mathsf{pk}, m, \mathsf{pk}^*, m^*; r)$ and the honest one of $\mathsf{Enc}(\mathsf{pk}^*, m^*, \mathsf{pk}^{**}, m^{**}; r)$ w.r.t. randomly sampled $(\mathsf{pk}^{**}, m^{**})$ from public key set $\mathcal{P} := \{\mathsf{pk}, \mathsf{pk}^*\}$ and message space $\mathcal{M}$.

Relations among Relevant Notions. We first argue that dual-deniability implies both IND-CPA and IK-CPA (Prop. 1). The intuition is that coercion attacks can be seen as an enhancement of CPA, in that the adversary $\mathcal{A}$ learns not only the target public keys the messages, but also the randomness used in encryption.

Then we move to *robustness* concerning ciphertext collisions (Def. 4). Note dual-deniability functionally makes for "equivocal" ciphertexts being simultaneously the encryptions of m under pk and m^* under pk^*, so rejects *key-less* or *mixed robustness* (Prop. 2), which exactly disallows collisions on distinct encryption materials. Yet, dual-deniability brings no impact on decryption phase, and so is consistent with *full robustness* asking a ciphertext not to be decryptable by distinct keys, though it does not further imply this inability (Prop. 3).

Finally we explore building dual-deniability from other notions. One natural idea is that it equals to the combination of two biased properties – message and key deniability, e.g., first deny the message via $r^*_{\mathtt{M}} \leftarrow_{\$} \mathsf{Fake}_{\mathtt{M}}(\mathsf{pk}, m, r, m^*)$ and then the key via $r^*_{\mathtt{K}} \leftarrow_{\$} \mathsf{Fake}_{\mathtt{K}}(\mathsf{pk}, m^*, r^*_{\mathtt{M}}, \mathsf{pk}^*)$. However, since the two notions only ensure the deniability using real randomness r, we cannot further expect $r^*_{\mathtt{K}}$ to be indistinguishable from an already fake coin $r^*_{\mathtt{M}}$. Another attempt is to encrypt twice in a KEM manner, by encrypting m using a one-time key pair $(\mathsf{pk}_{\mathtt{M}}, \mathsf{sk}_{\mathtt{M}})$ of an MDPKE, and encrypting $\mathsf{sk}_{\mathtt{M}}$ using $\mathsf{pk}_{\mathtt{K}}$ of a KDPKE. But it also fails in that the sender has to honestly reveal $\mathsf{sk}_{\mathtt{M}}$ as KDPKE is only key-deniable, and so lying about the first cipher is impossible.

These failures somehow inspire us to encrypt the message in a way that the "plaintext" is independent of m. The feasible solution comes to "encrypting one bit" within the decryption result by use of different keys. That is, ciphertext of 1 is encrypting 1 under $\mathsf{pk}_{\mathtt{K}}$, which is decryptable under $\mathsf{sk}_{\mathtt{K}}$; and that of 0 is encrypting 1 under random $\overline{\mathsf{pk}}_{\mathtt{K}} \neq \mathsf{pk}_{\mathtt{K}}$, which would be undecryptable under $\mathsf{sk}_{\mathtt{K}}$ if *full robustness* – the impossibility of decryption collisions, is provided. Delicately, such fresh $\overline{\mathsf{pk}}_{\mathtt{K}}$ can be freely sampled once the underlying KDPKE also delivers key-simulatability [20]. Finally, faking between the two types of encryption is available by switching the used public key via key-deniability. See Prop. 4 for the details of these reductions.

Generic Construction of Dual-Deniability. Despite the adoptive model, it is hard to build DDPKE from the methods effective in message-deniable designs. E.g., in [10], description of a translucent set (TS) serves as a public key. But elements of a TS when working with a real key are only supposed to be indistinguishable from the uniform set over $\{0, 1\}^\lambda$, and (as far as we know) cannot further be explained as the ones of another TS served as a fake key, which, however, is the core task for key deniability. The only hope appears in a recent plan-ahead scheme by An et al. [3] using ciphertext-simulatable PKE (Def. 5), which admits oblivious sampling $\mathsf{OEnc}(;r_{\mathsf{O}})$ of a ciphertext c and inverse sampling IEnc of randomness r_{O} from any c. The framework of [3] (Fig. 2) works as follows. Its overall ciphertext ct of m consists of an OWF tag $\sigma = \mathcal{H}(u)$, and n sub-ciphertexts $\{c_i\}_i$ having two types according to the pattern of a random bit-string $\boldsymbol{s}$: for $\boldsymbol{s}[i] = 0$, c_i is randomly sampled from OEnc; for $\boldsymbol{s}[i] = 1$, c_i is a real encryption of arbitrary $m_i||u$ under pk, except for the sole index t mapped from $\boldsymbol{s}$ it embeds $m_t := m$. The receiver decrypts each c_i, verifies σ to recover $\boldsymbol{s}[i]$, and so locates t. When coerced, the sender interprets c_t as an oblivious element by invoking IEnc, then flips $\boldsymbol{s}[t]$ to reveal a fake $\boldsymbol{s}^*$ and t^* (mapped from $\boldsymbol{s}^*$), such that ct can be explained as the encryption of the plan-ahead fake message $m^* := m_{t^*}$. As a result, the detecting probability is scaled by $\Delta(\boldsymbol{s}, \boldsymbol{s}^*) = \Theta(\frac{1}{\sqrt{n}})$ (Thm. 1 of [3]).

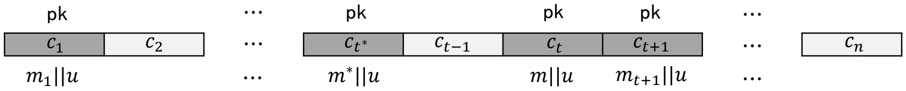

Fig. 2. Encryption style of [3] under random $\boldsymbol{s}$ over $\{0,1\}^n$. t and t^* are uniquely determined by $\boldsymbol{s}$. $\{c_i\}_{\boldsymbol{s}[i]=0}$ (lightgray) are sampled from OEnc; $\{c_i\}_{\boldsymbol{s}[i]=1}$ (drakgray) encrypt random $m_i||u$ under pk, except m_t/m_{t^*} is the real/fake message m/m^*.

We are going to lift this suite up to the dual-deniable mode. We begin with an unsuccessful attempt, which essentially leads to the final secure construction. Observe that $\{c_i\}$ except c_t carry no information of the real message m. Thus, c_t itself is functionally enough to support correct decryption, assuming all the other c_i would be termed as invalid under sk. Note that type-0 c_i already mismatches with σ with the design in [3]. We can further replace the rest type-1 $\{c_i\}$ with those encrypted under pk^*, which would also be inconsistent with σ thanks to *one-wayness* of $\mathcal{H}$ as well as *semantic security* of the used PKE. Moreover, the deniability of key seems available: the sender still interprets c_t as a type-0 cipher and honestly reveals all the other randomness, aiming to explain ct as the overall encryption of m_{t^*} under pk^*. However, such falsehood is indeed *distinguishable* to a coercer. This is because now the distribution of fake opening D_1 only includes one key pk^*; while the view of honest opening D_0 may contain both pk^* and pk with probability $1/2$, since in such case the auxiliary encryption randomness pk^{**}, which is used to encrypt the auxiliary type-1 $\{c_i\}$, is randomly sampled from $\mathcal{P} := \{\mathsf{pk}, \mathsf{pk}^*\}$.

Intuitively, in order to get rid of the above difference, we shall mess up those masking sub-ciphertexts $\{c_i\}_{\boldsymbol{s}[i]=1 \wedge i \neq t, t^*}$, i.e., make them encrypt random m_i using pk or pk^* with equal probability. Also, we need to enable the receiver to identify c_t as before. To these effects, we arrange all the type-1 masks after the index t, such that c_t will instead be the very first valid sub-ciphertext under

sk. But this change naturally requires t (resp., t^*) to always be the first bit-1 of $\boldsymbol{s}$ (resp., $\boldsymbol{s}^*$), which is impossible for a random $\boldsymbol{s}$. Fortunately, under this twisted rule of decryption, we actually no longer need the seed $\boldsymbol{s}$ to locate t or c_t. In particular, we switch to directly sample the target index $t \leftarrow_\$ [n-1]$, and simply set the plan-ahead fake index $t^* = t+1$. Further, all the obliviously-sampled (type-0) elements are placed before t. A brief scope on this new fit of $\{c_i\}$ is illustrated in Fig. 3. Upon coercion, the sender still explains c_t as an oblivious cipher and reveals t^* as well as the other real randomness, so to dishonestly open ct as the ciphertext of m^* under pk^* using fake auxiliary input $(\mathsf{pk}^{**}, m^{**}) := (\mathsf{pk}_{t^*+1}, m_{t^*+1})$, a random pair from $\mathcal{P} \times \mathcal{M}$. In this way, the view of fake opening only differs from that of real opening in the distribution of $t^* \in [2, n]$, whose distance from t is clearly $\frac{1}{n-1}$. Finally, recall that DDPKE implies MDPKE, then our construction can also be seen as an improved MDPKE with better detection probability than the baseline in [3].

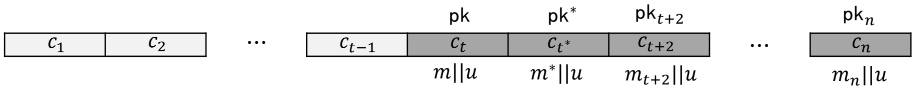

Fig. 3. Encryption style of our scheme under random t over $[n-1]$. $t^* = t+1$. $\{c_i\}_{i<t}$ (lightgray) are sampled from OEnc; $\{c_i\}_{i \geq t}$ (drakgray) encrypt random $m_i||u$ under random pk_i, except $(\mathsf{pk}_t, m_t)/(\mathsf{pk}_{t^*}, m_{t^*})$ is the real/fake pair $(\mathsf{pk}, m)/(\mathsf{pk}^*, m^*)$.

More Efficient Transform under Weak Model. Now we consider a weak flavor of deniability introduced in [10], which allows an extra encryption algorithm DEnc. The sender can first run DEnc and later equivocate that Enc is invoked (Def. 10).

Below we show how to simplify the above design in such weak model. Originally, we have to evaluate the difference between D_0 and D_1 within the fixed algorithm Enc. That distance depends heavily on the offset of the fake index t^*, which induces a polynomial factor n. But here, by use of two spare encryption algorithms, we can freely explain an execution of DEnc using t as one of Enc using t^*, without having any requirements on their similarities. Namely, the parameter n can be minimized to 2, leading $t = 1$ and $t^* = 2$. Concerning these constants, DEnc plays the role of the standard encryption:

$$c_1 \leftarrow_\$ \mathcal{E}.\mathsf{Enc}(\mathsf{pk}, m||u),\ c_2 \leftarrow_\$ \mathcal{E}.\mathsf{Enc}(\mathsf{pk}^*, m^*||u);$$

while Enc serves as fake opening by "flipping" c_1 as an obliviously sampled one:

$$c_1 \leftarrow_\$ \mathcal{E}.\mathsf{OEnc}(\mathsf{ik}),\ c_2 \leftarrow_\$ \mathcal{E}.\mathsf{Enc}(\mathsf{pk}, m||u).$$

In this way, the detecting probability only depends on the distinguishability between a real encryption and an obliviously sampled one, which, is negligible by *ciphertext-simulatability* of the underlying PKE.

Revisit of iO-based Scheme. [35] have presented a notable message-deniable scheme based on indistinguishability obfuscation ($i\mathcal{O}$), where the authors also introduced a special feature of deniability called *public explanation*, asking convincing randomness r^* of ct can be generated from $\mathsf{Fake}(\mathsf{pk}, m, \mathsf{ct})$ without the original coin r that creates ct. [35] pointed out that message-deniability is implied by such property plus IND-CPA, we further prove that dual-deniability is also in hand with the help of IK-CPA. The intuition behind is r^* is independent of r, so we can embed the CPA ciphertext into the dual-deniable challenge. Namely, start with the honest case $(r, c_0 = \mathsf{Enc}(\mathsf{pk}^*, m^*; r))$, first hop to (r^*, c_0) where $r^* \leftarrow_{\$} \mathsf{Fake}(\mathsf{pk}^*, m^*, c_0)$, by the security of public explanation the two hybrids are indistinguishable; then move to $(r^*, c'_1 = \mathsf{Enc}(\mathsf{pk}, m^*; r))$, the indistinguishability comes from IK-CPA over c_0 and c'_1; finally arrive at the fake case $(r^*, c_1 = \mathsf{Enc}(\mathsf{pk}, m; r))$, and the indistinguishability comes from IND-CPA over c'_1 and c_1. Since in [35] the scheme was proved to be IND-CPA (IK-CPA trivially follows) and publicly explainable, it is then inherently dual-deniable. Below we summarize in Table. 1 the caveats of the above three DDPKE constructions.

Scheme	Methods	Message space	Detecting probability	Model	Setting
Ours-I	CSPKE	$\{0,1\}^\ell$	$1/\mathsf{poly}(\lambda)$	Standard	Plan-ahead
Ours-II	CSPKE	$\{0,1\}^\ell$	$\mathsf{negl}(\lambda)$	Weak	Plan-ahead
[35]	$i\mathcal{O}$	$\{0,1\}$	$\mathsf{negl}(\lambda)$	Standard	Normal

Table 1. Caveats of the involved DDPKE schemes. λ is the security parameter, ℓ is a function of λ associated with the underlying CSPKE.

CCA-secure Conversions. We finally consider capturing both dual-deniability and CCA security. One may expect that CCA security of our two schemes naturally comes from that of the used PKE. However, as we expound in §.6.1, it is wrong in that the adversary can query the decryption oracle with specific ciphertext reassembled from some c_i of the challenge ct, so to learn the questioned b. Thus, we turn to achieving the two properties in one shot. Although there are well-studied paradigms for CCA security, many (e.g., Fujisaki-Okamoto [24], Naor-Yung [32]) take auxiliary matching operations on the plaintext and so reject any sense of deniability. The only survivor is the IBE-based framework [7, 12], whose extra element is just a MAC for the validity of the ciphertext, and so denying the encryption contents does not influence this check.

Recall that the Boneh-Katz (BK) transform [7] takes the master key of IBE as the public key of PKE. Then the anonymity of such fashion hinges on the key privacy of IBE master keys, and so we step to formalize dual-deniability of IBE under the setting of multiple trusted authorities (Def. 14). Further, we show how to promote the BK transform to the dual-deniable setting, i.e., gaining dual-deniable and CCA-secure PKE from any selectively multi-TA dual-deniable IBE. Accordingly, Such IBE can be derived by applying our first mechanism to any ciphertext-simulatable multi-TA IBE.

Other Related Works. Besides the notion of message-deniability, Canetti et al. [10] also formalized receiver-deniability, where the receiver can generate a fake key decrypting the target ciphertext to a fake message; O'Neill et al. [33] further utilized lattice-based bi-translucent sets and simulatable encryption [17] to build weak bi-deniable schemes; Caro et al. put forth the deniable FE and presented a receiver-deniable scheme based on $i\mathcal{O}$ and delayed trapdoor circuits [14]; Apart from the bit-related encryption schema, An et al. [3] also examined some side-channel ((e.g., power or timing) attacks against deniability; By use of $i\mathcal{O}$, Canetti et al. [13] achieved fully interactive deniability, where the bribed parties' claims can be inconsistent; Agrawal et al. [2] resort to bootstrapping for obliviously generating homomorphic ciphertexts, they then introduced the deniable FHE with some instantiations from classic LWE; Coladangelo et al. [18] explored message deniability where the encryption is performed within a quantum program, and also proposed a construction assuming the quantum hardness of LWE.

For theoretical lower bounds, Bendlin et al. [6] showed that non-interactive receiver-deniable PKE with negligible detecting probability cannot have polynomial key size; Dachman-Soled [19] proved that sender deniability with super-polynomial security cannot be derived from black-box use of simulatable PKE.

Organization. In the forthcoming sections, we first recall some necessary preliminaries in §.2. Then §.3 presents the model of DDPKE and its relations to the relevant notions. §.4 proposes a generic approach of plan-ahead DDPKE from ciphertext-simulatable PKE. §.5 provides a more efficient framework under the weak model and revisits the existing $i\mathcal{O}$-based scheme. §.6 describes a general transform of CCA-secure DDPKE from any DDIBE with multi-TA setting.

2 Preliminaries

Notations. Let $\lambda \in \mathbb{N}$ denote the security parameter. Function $f(\lambda)$ is negligible (resp., polynomial) if it is $\mathcal{O}(\lambda^{-c})$ for all $c > 0$ (resp., $\mathcal{O}(\lambda^{c})$ for some constant $c > 0$), and is denoted as $\mathsf{negl}(\lambda)$ (resp., $\mathsf{poly}(\lambda)$). $f(\lambda)$ is abbreviated as f where clear from context. Denote by $y \leftarrow_{\$} \mathcal{F}(x)$ that a randomized algorithm $\mathcal{F}$ outputs y on input x, and by $y := \mathcal{F}(x;r)$ that specifies the randomness r of $\mathcal{F}$. Let integer set $[n] := \{1, \ldots, n\}$ and $[m,n] := \{m, \ldots, n\}$. Let bold lower-case letters, e.g., $\boldsymbol{s}$, be a bit-string. Denote by $x \leftarrow_{\$} \mathcal{X}$ sampling uniformly at random from finite set $\mathcal{X}$, and by $y \leftarrow_{\$} D$ sampling over the distribution D. Denote the statistical distance between $y_1 \leftarrow_{\$} D_1$ and $y_2 \leftarrow_{\$} D_2$ over $\mathcal{X}$ as $\Delta(y_1, y_2) = \frac{1}{2} \sum_{x \in \mathcal{X}} |D_1(x) - D_2(x)|$.

2.1 Model of Public Key Encryption

Definition 1 (PKE). *A PKE scheme $\mathcal{E}$ for message space $\mathcal{M}$ consists of four polynomial-time algorithms $\langle \mathsf{Gen}, \mathsf{KGen}, \mathsf{Enc}, \mathsf{Dec} \rangle$ with the following interfaces:*

- $\mathsf{Gen}(1^{\lambda})$*: On input the security parameter λ, the probabilistic initial key generation algorithm returns an initial key* ik.
- $\mathsf{KGen}(\mathsf{ik})$*: On input an initial key* ik*, the probabilistic key generation algorithm returns a public/secret key pair* $(\mathsf{pk}, \mathsf{sk})$.
- $\mathsf{Enc}(\mathsf{pk}, m)$*: On input a public key* pk *and a message m, the probabilistic encryption algorithm returns a ciphertext* ct.

- $\mathsf{Dec}(\mathsf{sk}, \mathsf{ct})$*: On input a secret key* sk *and a ciphertext* ct*, the deterministic decryption algorithm returns a message* $m \in \mathcal{M}$ *or* $\perp$ *to declare a failure.*

Remark 1. Following [4], the initial key ik is introduced to argue *anonymity* between different pk; W.l.o.g., we assume pk (resp., sk) always contains the corresponding ik (resp., pk), and each pk associates with the same $\mathcal{M} \subseteq \{0,1\}^*$.

Correctness. $\mathcal{E}$ is said to be *correct* if, for all security parameter $\lambda \in \mathbb{N}$, initial key $\mathsf{ik} \leftarrow_{\$} \mathsf{Gen}(1^\lambda)$, key pair $(\mathsf{pk}, \mathsf{sk}) \leftarrow_{\$} \mathsf{KGen}(\mathsf{ik})$, and message $m \in \mathcal{M}$, it holds that $\mathbb{P}\left[\mathsf{Dec}\big(\mathsf{sk}, \mathsf{Enc}(\mathsf{pk}, m)\big) = m\right] = 1 - \mathsf{negl}(\lambda)$.

Below we review the basic security notions of PKE, where we denote the usual term "IND-CPA/CCA" as IM-CPA/CCA, mainly to emphasize that they are about the privacy of messages.

Definition 2 (CCA Security). *$\mathcal{E}$ is IM-CCA (resp., IK-CCA) if for all PPT adversary $\mathcal{A} := (\mathcal{A}_1, \mathcal{A}_2)$, the absolute difference of probability of returning 1 between experiment $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{IM}\text{-}0}$ (resp., $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{IK}\text{-}0}$) and $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{IM}\text{-}1}$ (resp., $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{IK}\text{-}1}$) is negligible.*

```
Experiment: $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{IM}\text{-}b}(1^\lambda)$

$\mathsf{ik} \leftarrow_{\$} \mathsf{Gen}(1^\lambda)$.
$(\mathsf{pk}, \mathsf{sk}) \leftarrow_{\$} \mathsf{KGen}(\mathsf{ik})$.
$(m_0, m_1, \mathsf{st}) \leftarrow_{\$} \mathcal{A}_1^{\mathcal{D}_{\mathsf{sk}}(\cdot)}(\mathsf{pk})$.
$\mathsf{ct} \leftarrow_{\$} \mathsf{Enc}(\mathsf{pk}, m_b)$.
$b' \leftarrow_{\$} \mathcal{A}_2^{\mathcal{D}_{\mathsf{sk}}(\neg\mathsf{ct})}(\mathsf{ct}, \mathsf{st})$.
Return $b'$.
```

```
Experiment: $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{IK}\text{-}b}(1^\lambda)$

$\mathsf{ik} \leftarrow_{\$} \mathsf{Gen}(1^\lambda)$.
$(\mathsf{pk}_0, \mathsf{sk}_0) \leftarrow_{\$} \mathsf{KGen}(\mathsf{ik})$, $(\mathsf{pk}_1, \mathsf{sk}_1) \leftarrow_{\$} \mathsf{KGen}(\mathsf{ik})$.
$(m, \mathsf{st}) \leftarrow_{\$} \mathcal{A}_1^{\mathcal{D}_{\mathsf{sk}_0}(\cdot), \mathcal{D}_{\mathsf{sk}_1}(\cdot)}(\mathsf{pk}_0, \mathsf{pk}_1)$.
$\mathsf{ct} \leftarrow_{\$} \mathsf{Enc}(\mathsf{pk}_b, m)$.
$b' \leftarrow_{\$} \mathcal{A}_2^{\mathcal{D}_{\mathsf{sk}_0}(\neg\mathsf{ct}), \mathcal{D}_{\mathsf{sk}_1}(\neg\mathsf{ct})}(\mathsf{ct}, \mathsf{st})$.
Return $b'$.
```

In the above games, st contains all the internal states of $\mathcal{A}_1$ in find phase, $m_0, m_1 \in \mathcal{M}$ and $|m_0| = |m_1|$, decryption oracle $\mathcal{D}_{\mathsf{sk}}$ on input ct outputs $m := \mathsf{Dec}(\mathsf{sk}, \mathsf{ct})$. Definitions of IM/IK-CPA take only one change: $\mathcal{A}$ has no access to any decryption oracles. We say a PKE scheme $\mathcal{E}$ is *CCA-secure* (resp., *CPA-secure*) if it is both IM-CCA (resp., IM-CPA) and IK-CCA (resp., IK-CPA).

2.2 Message-Deniability and Robustness of PKE

We recall the notion – *deniability of messages* introduced in [10], which is relative to the encryption randomness of Enc over space $\mathcal{R}_{\mathsf{E}}$, and in particular introduces an additional PPT algorithm Fake for producing fake randomness:

- $\mathsf{Fake}(\mathsf{pk}, m, r, m^*)$: On input a public key pk, a message m, randomness r of the original encryption, and a fake message m^*, return fake randomness r^*.

Accordingly, it should be satisfied that for any maliciously-selected $m, m^* \in \mathcal{M}$, the fake opening (pk, m^*, r^*) of ciphertext ct_1 actually for m is indistinguishable from the honest opening (pk, m^*, r) of ct_0 exactly for m^*.

Definition 3 (Deniability of Messages). *$\mathcal{E}$ satisfies deniability of messages if for all PPT adversary $\mathcal{A} := (\mathcal{A}_1, \mathcal{A}_2)$, the absolute difference of probability of returning 1 between experiment $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{MD}\text{-}1}$ and $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{MD}\text{-}0}$ is negligible.*

```
Experiment: $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{MD}\text{-}b}(1^\lambda)$

$\mathsf{ik} \leftarrow_\$ \mathsf{Gen}(1^\lambda)$, $(\mathsf{pk}, \mathsf{sk}) \leftarrow_\$ \mathsf{KGen}(\mathsf{ik})$.
$(m, m^*, \mathsf{st}) \leftarrow_\$ \mathcal{A}_1(\mathsf{pk})$.
$r \leftarrow_\$ \mathcal{R}_\mathsf{E}, r^* \leftarrow_\$ \mathsf{Fake}(\mathsf{pk}, m, r, m^*)$.
For $b = 0$, $D_0 := (r, \mathsf{Enc}(\mathsf{pk}, m^*; r))$.
For $b = 1$, $D_1 := (r^*, \mathsf{Enc}(\mathsf{pk}, m; r))$.
$b' \leftarrow_\$ \mathcal{A}_2(D_b, \mathsf{st})$.
Return $b'$.
```

Remark 2. In this work, we also consider a common relaxed condition where the non-trivial advantage of $\mathcal{A}$ is instead an *inverse polynomial* [2,10,33]. We say $\mathcal{E}$ is $f(\lambda)$-*deniable* if $f(\lambda) = \frac{1}{\mathsf{poly}(\lambda)}$, and omit the notation f when $f(\lambda) = \mathsf{negl}(\lambda)$.

Then we recall *key-less* robustness (KROB), *mixed* robustness (XROB), and *full* robustness (FROB) w.r.t. different types of ciphertext collisions. Combination of the three notions implies any other known flavors of robustness [1,23,31].

Definition 4 (KROB, XROB, FROB). *$\mathcal{E}$ satisfies key-less (resp., mixed, full) robustness if for all PPT adversary $\mathcal{A}$, the probability of returning 1 for experiment $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{KROB}}$ (resp., $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{XROB}}$, $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{FROB}}$) is negligible,*

```
Experiment: $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{KROB}}(1^\lambda)$

$\mathsf{ik} \leftarrow_\$ \mathsf{Gen}(1^\lambda)$.
$(m_0, m_1, \mathsf{pk}_0, \mathsf{pk}_1, r_0, r_1) \leftarrow_\$ \mathcal{A}(\mathsf{ik})$.
$\mathsf{ct}_0 := \mathsf{Enc}(\mathsf{pk}_0, m_0; r_0)$.
$\mathsf{ct}_1 := \mathsf{Enc}(\mathsf{pk}_1, m_1; r_1)$.
Return $(\mathsf{pk}_0 \neq \mathsf{pk}_1) \wedge (\mathsf{ct}_0 = \mathsf{ct}_1 \neq \perp)$.
```

```
Experiment: $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{XROB}}(1^\lambda)$

$\mathsf{ik} \leftarrow_\$ \mathsf{Gen}(1^\lambda)$.
$(m_0, \mathsf{pk}_0, r_0, \mathsf{sk}_1) \leftarrow_\$ \mathcal{A}(\mathsf{ik})$.
$\mathsf{ct}_0 := \mathsf{Enc}(\mathsf{pk}_0, m_0; r_0)$.
$m_1 := \mathsf{Dec}(\mathsf{sk}_1, \mathsf{ct}_0)$.
Return $(\mathsf{pk}_0 \neq \mathsf{pk}_1) \wedge (m_0 \neq \perp)$
          $\wedge (m_1 \neq \perp)$.
```

```
Experiment: $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{FROB}}(1^\lambda)$

$\mathsf{ik} \leftarrow_\$ \mathsf{Gen}(1^\lambda)$.
$(\mathsf{ct}, \mathsf{sk}_0, \mathsf{sk}_1) \leftarrow_\$ \mathcal{A}(\mathsf{ik})$.
$m_0 := \mathsf{Dec}(\mathsf{sk}_0, \mathsf{ct})$, $m_1 := \mathsf{Dec}(\mathsf{sk}_1, \mathsf{ct})$.
Return $(\mathsf{pk}_0 \neq \mathsf{pk}_1) \wedge (m_0 \neq \perp) \wedge (m_1 \neq \perp)$.
```

where it is implicitly required that $m_0 \in \mathcal{M}$ and $m_1 \in \mathcal{M}$.

2.3 Ciphertext(Key)-Simulatable Public Key Encryption

Simulatable PKE [20,30] has been explored for a while to construct deniable systems [19,33]. Below we recall its relaxed variant called ciphertext-simulatability [3], which admits oblivious sampling (OEnc) from the ciphertext space, as well as the inverting sampling (IEnc) that interprets a real ciphertext as an obliviously sampled one by use of the original encryption materials.

Formally, a ciphertext-simulatable PKE scheme $\mathcal{E}$ consists of universal algorithms $\langle \mathsf{Gen}, \mathsf{KGen}, \mathsf{Enc}, \mathsf{Dec} \rangle$ plus two PPT algorithms $\langle \mathsf{OEnc}, \mathsf{IEnc} \rangle$, and the sampling randomness space w.r.t. OEnc is specified as $\mathcal{R}_{\mathsf{O}}$.

- $\mathsf{OEnc}(\mathsf{ik}; r_{\mathsf{O}})$: On input an initial key ik, use randomness $r_{\mathsf{O}} \leftarrow_{\$} \mathcal{R}_{\mathsf{O}}$ to sample a simulated ciphertext ct.
- $\mathsf{IEnc}(\mathsf{pk}, m, r_{\mathsf{E}})$: On input a public key pk, a message m, and randomness r_{E}, return simulated randomness r_{O}^*.

Definition 5 (Ciphertext-Simulatability). *$\mathcal{E}$ is ciphertext-simulatable if for all PPT adversary $\mathcal{A} := (\mathcal{A}_1, \mathcal{A}_2)$, $\mathsf{ik} \leftarrow_{\$} \mathsf{Gen}(1^\lambda)$, $(\mathsf{pk}, \mathsf{sk}) \leftarrow_{\$} \mathsf{KGen}(\mathsf{ik})$, $(m, \mathsf{st}) \leftarrow_{\$} \mathcal{A}_1(\mathsf{pk})$, $(r_{\mathsf{E}}, r_{\mathsf{O}}) \leftarrow_{\$} \mathcal{R}_{\mathsf{E}} \times \mathcal{R}_{\mathsf{O}}, \mathsf{ct}_0 := \mathsf{Enc}(\mathsf{pk}, m; r_{\mathsf{E}}), \mathsf{ct}_1 := \mathsf{OEnc}(\mathsf{ik}; r_{\mathsf{O}})$, it holds*

$$\Big| \mathbb{P}[\mathcal{A}_2(\mathsf{st}, \mathsf{ct}_0, \mathsf{IEnc}(\mathsf{pk}, m, r_{\mathsf{E}})) = 1] - \mathbb{P}[\mathcal{A}_2(\mathsf{st}, \mathsf{ct}_1, r_{\mathsf{O}}) = 1] \Big| = \mathsf{negl}(\lambda),$$

where we require $m \in \mathcal{M}$ and the probabilities are taken over the randomness of $\mathsf{ik}, (\mathsf{pk}, \mathsf{sk}), r_{\mathsf{E}}, r_{\mathsf{O}}$, and the one internally used by $\mathcal{A}$.

Note that algorithm OEnc only takes the initial key ik as input, so ciphertext-simulatability implies both IM-CPA and IK-CPA. Most of the known PKE schemes have been proved to meet such property, e.g., ElGamal and Cramer-Shoup instantiated with simulatable groups [21], anonymous RSA-OAEP [4], and Kyber [8]. See [4, 17, 20, 21, 27, 30, 33] for more details.

Regarding security proof of Prop. 4, we also require key-simulatability, another relaxed notion from the original simulatability [20]. Instead of ciphertexts, such property concerns the sampling of public keys, which delivers oblivious sampling (OKGen) from the public key space w.r.t. an initial key ik, as well as the inverting sampling (IKGen) that comes up with the sampling randomness k associated with a public key pk, such that $\mathsf{OKGen}(\mathsf{ik}; \mathsf{IKGen}(\mathsf{pk})) = \mathsf{pk}$. Formally, a key-simulatable PKE scheme $\mathcal{E}$ consists of universal algorithms $\langle \mathsf{Gen}, \mathsf{KGen}, \mathsf{Enc}, \mathsf{Dec} \rangle$ plus PPT algorithms $\langle \mathsf{OKGen}, \mathsf{IKGen} \rangle$, where we specify the key sampling randomness space as $\mathcal{R}_{\mathsf{K}}$.

- $\mathsf{OKGen}(\mathsf{ik}; k)$: On input an initial key ik, use randomness $k \leftarrow_{\$} \mathcal{R}_{\mathsf{K}}$ to generate a simulated public key pk.
- $\mathsf{IKGen}(\mathsf{pk})$: On input a public key pk, return simulated randomness k^*.

Definition 6 (Key-Simulatability). *$\mathcal{E}$ is key-simulatable if for all PPT adversary $\mathcal{A}$, $\mathsf{ik} \leftarrow_{\$} \mathsf{Gen}(1^\lambda)$, $(\mathsf{pk}, \mathsf{sk}) \leftarrow_{\$} \mathsf{KGen}(\mathsf{ik})$, $k \leftarrow_{\$} \mathcal{R}_{\mathsf{K}}$, it holds*

$$\Big| \mathbb{P}[\mathcal{A}(\mathsf{pk}, \mathsf{IKGen}(\mathsf{pk})) = 1] - \mathbb{P}[\mathcal{A}(\mathsf{OKGen}(\mathsf{ik}; k), k) = 1] \Big| = \mathsf{negl}(\lambda).$$

Note that IKGen does not require the randomness used in KGen as input. Thus, the above definition ensures that whatever knowledge one could learn from sampling a key pk using OKGen, one could also derive from only pk itself.

2.4 Message Authentication Code and Commitment Schemes

A simplified MAC scheme consists of two polynomial-time algorithms $\langle \mathsf{Mac}, \mathsf{Vry} \rangle$: probabilistic $\mathsf{Mac}(\mathsf{k}, m)$ returns an authentication code *tag* of m under secret key k; deterministic $\mathsf{Vry}(\mathsf{k}, m, tag)$ returns $0/1$ as the verification result of *tag* w.r.t. k and m. We require that for all $\mathsf{k} \in \{0,1\}^\lambda$ and $m \in \{0,1\}^*$, $\mathsf{Vry}(\mathsf{k}, m, \mathsf{Mac}(\mathsf{k}, m)) = 1$. Below we define the *one-time strong unforgeability* of MAC.

Definition 7 (One-time Strong Unforgeability). *A MAC scheme is one-time strongly unforgeable if for all PPT adversary* $\mathcal{A} := (\mathcal{A}_1, \mathcal{A}_2)$, $(m, \mathsf{st}) \leftarrow_\$ \mathcal{A}_1(1^\lambda)$, $\mathsf{k} \leftarrow_\$ \{0,1\}^\lambda$, $tag \leftarrow_\$ \mathsf{Mac}(\mathsf{k}, m)$, *and* $(m^*, tag^*) \leftarrow_\$ \mathcal{A}_2(tag, \mathsf{st})$, *it holds that*

$$\mathbb{P}\big[\mathsf{Vry}(\mathsf{k}, m^*, tag^*) = 1 \ \wedge\ (m^*, tag^*) \neq (m, tag)\big] = \mathsf{negl}(\lambda).$$

Then we move to "weak" commitment scheme [7], a 3-tuple of PPT algorithms $\langle \mathsf{CGen}, \mathsf{Samp}, \mathsf{Open} \rangle$. $\mathsf{CGen}(1^\lambda)$ returns a public parameter par; $\mathsf{Samp}(\mathsf{par})$ returns a triple (m, com, dec) with $m \in \{0,1\}^\lambda$ and $dec \in \{0,1\}^{\beta(\lambda)}$; $\mathsf{Open}(\mathsf{par}, com, dec)$ returns a message $m \in \{0,1\}^\lambda$ or $\perp$. We require that for all $\mathsf{par} \leftarrow_\$ \mathsf{CGen}(1^\lambda)$ and $(m, com, dec) \leftarrow_\$ \mathsf{Com}(\mathsf{par})$, $\mathsf{Open}(\mathsf{par}, com, dec) = m$. Below we define computational *hiding* and *biding* properties of a weak commitment scheme.

Definition 8 (Security of Commitment). *A commitment scheme satisfies hiding if for all PPT adversary* $\mathcal{A}$, $\mathsf{par} \leftarrow_\$ \mathsf{CGen}(1^\lambda)$, $b \leftarrow_\$ \{0,1\}$, $m_0 \leftarrow_\$ \{0,1\}^\lambda$, *and* $(m_1, com, dec) \leftarrow_\$ \mathsf{Samp}(\mathsf{par})$, *it holds that*

$$\Big| \mathbb{P}[\ b' = b \mid b' \leftarrow_\$ \mathcal{A}(\mathsf{par}, com, m_b)\ ] - 1/2 \Big| = \mathsf{negl}(\lambda),$$

and binding if for all PPT adversary $\mathcal{A}, \mathsf{par} \leftarrow_\$ \mathsf{CGen}(1^\lambda)$, *and* $(m, com, dec) \leftarrow_\$ \mathsf{Samp}(\mathsf{par})$, *it holds that*

$$\mathbb{P}\big[\mathsf{Open}(\mathsf{par}, com, dec') \notin \{m, \perp\} \mid dec' \leftarrow_\$ \mathcal{A}(\mathsf{par}, m, com, dec)\big] = \mathsf{negl}(\lambda).$$

3 Defining Dual-Deniable Public Key Encryption

Message deniability can be seen as a feasible shield over data privacy in context of active attacks. However, there are lots of applications (e.g., authenticated key exchange, anonymous credentials, or electronic auction [9,22,36]) for secure messaging where key privacy is also requested.

As mentioned in §.1, we already have some countermeasures like *anonymous* encryption [4] under the CPA/CCA setting, which ensures the indistinguishability between ciphertexts under different public keys. But they are no longer effective if more advanced coercion/bribery attacks are applied. Thus, we are motivated to formalize the similar notion of message deniability concerning key privacy, named *key-deniability* (KD), where a sender being forced to reveal the public key and the randomness used for encryption, can convincingly provide fake randomness to claim any valid receiver with whom the sender pretends that it has communicated. We further endow the coerced sender with the freedom to interpret the ciphertext as any message under any public key, and we call this property *dual-deniability*. Obviously, this property is the most desirable one in practice, as it allows deniability to the greatest extent.

We begin with defining *key-deniable* and *dual-deniable* PKE, then show some theoretical relations between the involved security notions, and finally formalize a relaxed variant of DDPKE under *plan-ahead* setting.

3.1 Key-Deniability and Dual-Deniability

Key deniability supports opening a ciphertext ct under a dishonest (fake) public key pk^*, i.e., a fake randomness r^* associated with pk^* can be generated by use of the real encryption materials (pk, m, r) of ct, as long as pk^* shares the same ik (i.e., the same PKI authority) of the real key pk.

Formally, a KDPKE scheme $\mathcal{E} := \langle \mathsf{Gen}, \mathsf{KGen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Fake} \rangle$ has the first four universal algorithms, while algorithm Fake is converted as follows:

- $\mathsf{Fake}(\mathsf{pk}, m, r, \mathsf{pk}^*)$: On input a public key pk, a message m, randomness r of the original encryption, and a fake public key pk^*, return fake randomness r^*.

Analogous to Def. 3, *key-deniability* requires the indistinguishability between an honest opening under the fake key pk^* and the fake opening under the real key pk, which are captured as the following codes.

Definition 9 (Deniability of Keys). *$\mathcal{E}$ satisfies deniability of keys if for all PPT adversary $\mathcal{A} := (\mathcal{A}_1, \mathcal{A}_2)$, the absolute difference of probability of returning 1 between experiment $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{KD}\text{-}1}$ and $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{KD}\text{-}0}$ is negligible.*

Experiment*:* $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{KD}\text{-}b}(1^\lambda)$

```
$\mathsf{ik} \leftarrow_{\$} \mathsf{Gen}(1^\lambda)$, $(\mathsf{pk}, \mathsf{sk}) \leftarrow_{\$} \mathsf{KGen}(\mathsf{ik})$, $(\mathsf{pk}^*, \mathsf{sk}^*) \leftarrow_{\$} \mathsf{KGen}(\mathsf{ik})$.
$(m, \mathsf{st}) \leftarrow_{\$} \mathcal{A}_1(\mathsf{pk}, \mathsf{pk}^*)$.
$r \leftarrow_{\$} \mathcal{R}_{\mathsf{E}}, r^* \leftarrow_{\$} \mathsf{Fake}(\mathsf{pk}, m, r, \mathsf{pk}^*)$.
For $b = 0$, $D_0 := (r, \mathsf{Enc}(\mathsf{pk}^*, m; r))$.
For $b = 1$, $D_1 := (r^*, \mathsf{Enc}(\mathsf{pk}, m; r))$.
$b' \leftarrow_{\$} \mathcal{A}_2(D_b, \mathsf{st})$.
Return $b'$.
```

Combining the two one-sided notions, we are ready to present the model of DDPKE, where a sender can opt for whatever she/he wants to reveal. Namely, algorithm Fake now takes into consideration both fake message and public key:

- $\mathsf{Fake}(\mathsf{pk}, m, r, \mathsf{pk}^*, m^*)$: On input a public key pk, a message m, randomness r of the original encryption, and a fake public key pk^* and message m^*, return fake randomness r^*.

Weak DDPKE. We also adapt the weak model [10] to dual-deniability, where the sender can first encrypt under an alternative algorithm DEnc and later claim that it has executed Enc. The newly-introduced encryption algorithm DEnc has an unchanged interface except a different encryption randomness space $\mathcal{R}_{\mathsf{DE}}$.

Definition 10 ((Weak) Dual-Deniability). *A DDPKE scheme satisfies dual-deniability (resp., weak dual-deniability) if for all PPT adversary $\mathcal{A} := (\mathcal{A}_1, \mathcal{A}_2)$, the absolute difference of probability of returning 1 between experiment $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{DD}\text{-}1}$ and $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{DD}\text{-}0}$ (resp., $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{wDD}\text{-}1}$ and $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{wDD}\text{-}1}$) is negligible.*

```
Experiment: $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{DD}\text{-}b}(1^\lambda)$

$\mathsf{ik} \leftarrow_{\$} \mathsf{Gen}(1^\lambda)$.
$(\mathsf{pk}, \mathsf{sk}), (\mathsf{pk}^*, \mathsf{sk}^*) \leftarrow_{\$} \mathsf{KGen}(\mathsf{ik})$.
$(m, m^*, \mathsf{st}) \leftarrow_{\$} \mathcal{A}_1(\mathsf{pk}, \mathsf{pk}^*)$.
$r \leftarrow_{\$} \mathcal{R}_{\mathsf{E}}, r^* \leftarrow_{\$} \mathsf{Fake}(\mathsf{pk}, m, r, \mathsf{pk}^*, m^*)$.
For $b = 0, D_0 := (r, \mathsf{Enc}(\mathsf{pk}^*, m^*; r))$.
For $b = 1, D_1 := (r^*, \mathsf{Enc}(\mathsf{pk}, m; r))$.
$b' \leftarrow_{\$} \mathcal{A}_2(D_b, \mathsf{st})$.
Return $b'$.
```

```
Experiment: $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{wDD}\text{-}b}(1^\lambda)$

$\mathsf{ik} \leftarrow_{\$} \mathsf{Gen}(1^\lambda)$.
$(\mathsf{pk}, \mathsf{sk}), (\mathsf{pk}^*, \mathsf{sk}^*) \leftarrow_{\$} \mathsf{KGen}(\mathsf{ik})$.
$(m, m^*, \mathsf{st}) \leftarrow_{\$} \mathcal{A}_1(\mathsf{pk}, \mathsf{pk}^*)$.
$r \leftarrow_{\$} \mathcal{R}_{\mathsf{E}}, r' \leftarrow_{\$} \mathcal{R}_{\mathsf{DE}}$.
$r^* \leftarrow_{\$} \mathsf{Fake}(\mathsf{pk}, m, r', \mathsf{pk}^*, m^*)$.
For $b = 0, D_0 := (r, \mathsf{Enc}(\mathsf{pk}^*, m^*; r))$.
For $b = 1, D_1 := (r^*, \mathsf{DEnc}(\mathsf{pk}, m; r'))$.
$b' \leftarrow_{\$} \mathcal{A}_2(D_b, \mathsf{st})$.
Return $b'$.
```

Naturally, for weak DDPKE, the other related security requirements should hold w.r.t. both Enc and DEnc. We give more discussions on the theoretical necessity and practical applications of weak DDPKE at App. A.

3.2 Relations between Deniability and Other Notions

In §.2, we review several primary notions for the consistency and privacy of PKE. Now we explore the relations (implications or separations) between these and deniable notions.

(Weak) Deniability implies CPA security. The intuition is that coercion attacks can be seen as a proactive variant of CPA, where the ability of the adversary is enhanced to approach the entire information of the target ciphertext including the used message, public key, and the internal randomness.

Proposition 1. *Suppose a PKE scheme $\mathcal{E}$ is (weakly) dual-deniable (resp., key-deniable, message-deniable), then it is CPA-secure (resp., IK-CPA, IM-CPA).*

Proof. We begin with the arguments for dual-deniability. First consider the implication of IK-CPA, suppose there is a CPA adversary $\mathcal{A}$ succeeds in $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{IK}\text{-}b}$ with non-negligible advantage ϵ, then we can build a PPT algorithm $\mathcal{B}$ that breaks *dual-deniability* of $\mathcal{E}$ with also non-negligible advantage ϵ. Let $\mathsf{ik} \leftarrow_{\$} \mathsf{Gen}(1^\lambda)$, $(\mathsf{pk}, \mathsf{sk}) \leftarrow_{\$} \mathsf{KGen}(\mathsf{ik})$, $(\mathsf{pk}^*, \mathsf{sk}^*) \leftarrow_{\$} \mathsf{KGen}(\mathsf{ik})$. First $\mathcal{B}$ forwards $(\mathsf{pk}, \mathsf{pk}^*)$ to $\mathcal{A}$, with which $\mathcal{A}$ picks $m \in \mathcal{M}$ and submits it to $\mathcal{B}$. Then $\mathcal{B}$ sends (m, m) to the challenger, getting back an instance $D_b := (r_b, \mathsf{ct}_b)$. Finally, $\mathcal{B}$ passes ct_b to $\mathcal{A}$ and simply relays $\mathcal{A}$'s answer to the challenger. The non-negligible advantage of $\mathcal{B}$ derives from the fact that $\mathsf{ct}_b = \mathsf{Enc}(\mathsf{pk}^*, m; r)$ for $b = 0$ or $\mathsf{Enc}(\mathsf{pk}, m; r)$ for $b = 1$, which is exactly the corresponding challenge of the IK-CPA game. The implication of IM-CPA is nearly the same as above, so we omit the detailed argument for brevity.

Then, we move to the weak-mode cases. We are going to prove that both $\langle \mathsf{Gen}, \mathsf{KGen}, \mathsf{Enc}, \mathsf{Dec} \rangle$ and $\langle \mathsf{Gen}, \mathsf{KGen}, \mathsf{DEnc}, \mathsf{Dec} \rangle$ are IM-CPA if $\mathcal{E}$ is weakly dual-deniable. To argue that $\mathsf{ct}_0 \leftarrow_{\$} \mathsf{Enc}(\mathsf{pk}, m_0)$ and $\mathsf{ct}_1 \leftarrow_{\$} \mathsf{Enc}(\mathsf{pk}, m_1)$ are indistinguishable, we take $\mathsf{pk} = \mathsf{pk}^* := \mathsf{pk}$, $m := m_0$ and $m^* := m_1$, then by *weak dual-deniability* we have ct_1 is indistinguishable from $\mathsf{ct}_0' = \mathsf{DEnc}(\mathsf{pk}, m_0)$. Next, we further change $m = m^* := m_0$, again by *weak dual-deniability* we have ct_0' is indistinguishable from ct_0. Combining these analysis, we have $\mathsf{ct}_1 \overset{\mathsf{c}}{\approx} \mathsf{ct}_0$. The same techniques can be used to prove that $\langle \mathsf{Gen}, \mathsf{KGen}, \mathsf{DEnc}, \mathsf{Dec} \rangle$ is IM-CPA.

It remains to prove that the two sub-schemes are also IK-CPA. To show $\mathsf{ct}_0 \leftarrow_{\$} \mathsf{Enc}(\mathsf{pk}_0, m)$ and $\mathsf{ct}_1 \leftarrow_{\$} \mathsf{Enc}(\mathsf{pk}_1, m)$ are indistinguishable, we take $\mathsf{pk} := \mathsf{pk}_0$, $\mathsf{pk}^* := \mathsf{pk}_1$, and $m = m^* := m$, then by *weak dual-deniability* we have ct_1 is indistinguishable from $\mathsf{ct}_0' = \mathsf{DEnc}(\mathsf{pk}_0, m)$. Further, we change $\mathsf{pk}^* := \mathsf{pk}_0$, again by *weak dual-deniability* we have ct_0' is indistinguishable from ct_0. Combining these hops, we have $\mathsf{ct}_1 \overset{\mathsf{c}}{\approx} \mathsf{ct}_0$. The same techniques can be used to prove that $\langle \mathsf{Gen}, \mathsf{KGen}, \mathsf{DEnc}, \mathsf{Dec} \rangle$ is IK-CPA. □

Dual (Key)-Deniability contradicts KROB and XROB. This statement comes from that dual (key)-deniability enables an encryption of m under pk to be explained as one of m^* under pk^*, essentially leading to an encryption collision.

Proposition 2. *Suppose a PKE scheme $\mathcal{E}$ satisfies deniability of keys or dual-deniability, then it is neither key-less robust nor mixed robust.*

Proof. We first show how an adversary $\mathcal{A}$ can produce the tuple $(m_0, m_1, \mathsf{pk}_0, \mathsf{pk}_1, r_0, r_1)$, for which the game $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{KROB}}$ returns 1 with overwhelming probability. $\mathcal{A}$ honestly runs $\mathsf{ik} \leftarrow_{\$} \mathsf{Gen}(1^\lambda)$, $(\mathsf{pk}_0, \mathsf{sk}_0)$, $(\mathsf{pk}_1, \mathsf{sk}_1) \leftarrow_{\$} \mathsf{KGen}(\mathsf{ik})$, if $\mathsf{pk}_0 = \mathsf{pk}_1$ which only occurs with negligible probability by the CPA security (induced by dual-deniability) of $\mathcal{E}$, resamples these keys. Next, $\mathcal{A}$ samples $m_0 \leftarrow_{\$} \mathcal{M}$, $r_0 \leftarrow_{\$} \mathcal{R}_{\mathsf{E}}$, and generates $\mathsf{ct}_0 := \mathcal{E}.\mathsf{Enc}(\mathsf{pk}_0, m_0; r_0)$. Then $\mathcal{A}$ samples another $m_1 \leftarrow_{\$} \mathcal{M}$, and invokes $\mathcal{E}.\mathsf{Fake}(\mathsf{pk}_0, m_0, r_0, \mathsf{pk}_1, m_1)$ to get r_1. By the *dual-deniability* of $\mathcal{E}$, it holds that $\mathbb{P}[\mathcal{E}.\mathsf{Enc}(\mathsf{pk}_1, m_1; r_1) = \mathsf{ct}_0] = 1 - \mathsf{negl}(\lambda)$, thus $(m_0, m_1, \mathsf{pk}_0, \mathsf{pk}_1, r_0, r_1)$ is a successful output of $\mathcal{A}$ against the KROB of $\mathcal{E}$.

Then we construct a tuple $(m_0, \mathsf{pk}_0, r_0, \mathsf{ct}_1, \mathsf{sk}_1)$ for which game $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{XROB}}$ always returns 1. $\mathcal{A}$ honestly runs $\mathsf{ik} \leftarrow_{\$} \mathsf{Gen}(1^\lambda)$, $(\mathsf{pk}_0, \mathsf{sk}_0)$, $(\mathsf{pk}_1, \mathsf{sk}_1) \leftarrow_{\$} \mathsf{KGen}(\mathsf{ik})$. Next, $\mathcal{A}$ samples $m_1 \leftarrow_{\$} \mathcal{M}$, $r_1 \leftarrow_{\$} \mathcal{R}_{\mathsf{E}}$, and generates $\mathsf{ct}_1 := \mathcal{E}.\mathsf{Enc}(\mathsf{pk}_1, m_1; r_1)$. $\mathcal{A}$ further samples $m_0 \leftarrow_{\$} \mathcal{M}$ and $r_0 \leftarrow_{\$} \mathcal{E}.\mathsf{Fake}(\mathsf{pk}_1, m_1, r_1, \mathsf{pk}_0, m_0)$. By *dual-deniability* and *correctness* of $\mathcal{E}$, it holds that $\mathbb{P}[\mathcal{E}.\mathsf{Enc}(\mathsf{pk}_0, m_0; r_0) = \mathsf{ct}_1] = 1 - \mathsf{negl}(\lambda)$ and $\mathbb{P}[\mathcal{E}.\mathsf{Dec}(\mathsf{sk}_1, \mathsf{ct}_1) = m_1] = 1 - \mathsf{negl}(\lambda)$, thus $(m_0, \mathsf{pk}_0, r_0, \mathsf{sk}_1)$ is a successful output of $\mathcal{A}$ against the XROB of $\mathcal{E}$. □

Dual-Deniability separates FROB. Finally, we shall demonstrate a separation result, i.e., dual-deniability does not imply FROB. To this effect, it suffices to show that there is a dual-deniable PKE scheme that is not FROB, which can be derived from the $i\mathcal{O}$-based framework $i\mathcal{DE}$ in [35].

Proposition 3. *Suppose that there exists indistinguishability obfuscation, then we can build a PKE scheme that is dual-deniable but not fully robust.*

Proof. First, we will later in §.5.2 prove that the message-deniable construction $i\mathcal{DE}$ is also dual-deniable. Besides, the decryption procedure of $i\mathcal{DE}$ adopts the exact secret key and decryption algorithm of its underlying PKE scheme. Thus, we can use a non-FROB PKE (e.g., ElGamal) to obtain a non-FROB and dual-deniable instance of $i\mathcal{DE}$, as required by the separation. □

However, we stress that dual-deniability does not further reject FROB. In particular, we can also use an FROB PKE (e.g., the one proposed in [23]) to earn an instance of $i\mathcal{DE}$ being both FROB and dual-deniable. This is abstractly because though dual-deniability destroys the one-to-one map between the encryption and ciphertext, it brings no impact on the decryption procedure, thus giving hope for arguing the robustness on decryption consistency.

3.3 Relations within Deniable Notions

Dual-deniability trivially implies deniability of messages or keys, by restricting the game to return $\perp$ whenever $m \neq m^*$ or $\mathsf{pk} \neq \mathsf{pk}^*$. However, we fail to prove the reverse statement that it can be derived from the two partial notions. One natural attempt is to encrypt twice in a KEM manner, i.e., encrypt m as $\mathsf{ct}_\mathtt{M}$ using a one-time key pair $(\mathsf{pk}_\mathtt{M}, \mathsf{sk}_\mathtt{M})$ from an MDPKE, and encrypt $\mathsf{sk}_\mathtt{M}$ as $\mathsf{ct}_\mathtt{K}$ using the long-term key $\mathsf{pk}_\mathtt{K}$ from a KDPKE. However, the fatal fault is that one has to honestly reveal $\mathsf{sk}_\mathtt{M}$ since KDPKE is only key-deniable, and so any fake of $\mathsf{ct}_\mathtt{M}$ can be trivially caught as the coercer itself can decrypt $\mathsf{ct}_\mathtt{M}$ via $\mathsf{sk}_\mathtt{M}$.

Then we explore combining key deniability with other properties to achieve dual-deniability, and it is *robustness* that works. In more detail, we devise a subtle bit-encryption fashion such that the message is hidden in the decryption result (success or failure). Namely, ciphertext of 1 encrypts under pk; ciphertext of 0 encrypts under another fresh $\overline{\mathsf{pk}} \neq \mathsf{pk}$. In this way, a receiver obtains bit 0/1 by testing if the ciphertext can be decrypted under sk, this is rightly ensured by full robustness of the used PKE. The remaining issue is how to obtain a fresh $\overline{\mathsf{pk}}$. In real-world applications, the sender can easily take $\overline{\mathsf{pk}}$ as the public key of another system user. Yet, for theoretical sake, we need the used PKE to also meet *key-simulatability* (Def. 6), which admits two helper algorithms $\langle \mathsf{OKGen}, \mathsf{IKGen} \rangle$ for sampling public keys.

Proposition 4. *Suppose there is a fully robust and key-simulatable KDPKE scheme $\mathcal{E}$ for bits, then we can build a DDPKE scheme $\mathcal{DE}$ for bits.*

Proof. The construction of $\mathcal{DE}$ is as follows, where we denote by $\mathcal{R}_\mathsf{E}$ (resp., $\mathcal{R}_\mathsf{K}$) the encryption (resp., oblivious public key sampling) randomness space of $\mathcal{E}$.

- $\mathsf{Gen}(1^\lambda)$: Return $\mathsf{ik} \leftarrow_\$ \mathcal{E}.\mathsf{Gen}(1^\lambda)$.
- $\mathsf{KGen}(\mathsf{ik})$: Return $(\mathsf{pk}, \mathsf{sk}) \leftarrow_\$ \mathcal{E}.\mathsf{KGen}(\mathsf{ik})$.
- $\mathsf{Enc}(\mathsf{pk}, b; \mathfrak{R})$: Sample $\mathfrak{R} := (k, r) \leftarrow_\$ \mathcal{R}_\mathsf{K} \times \mathcal{R}_\mathsf{E}$, and conduct the following two branches for generating ct:
  - For $b = 0$, $\mathsf{ct} := \mathcal{E}.\mathsf{Enc}(\overline{\mathsf{pk}}, 1; r)$ where $\overline{\mathsf{pk}} := \mathcal{E}.\mathsf{OKGen}(\mathsf{ik}; k)$.

  - For $b = 1$, $\mathsf{ct} := \mathcal{E}.\mathsf{Enc}(\mathsf{pk}, 1; r)$.
- $\mathsf{Dec}(\mathsf{sk}, \mathsf{ct})$: $d := \mathcal{E}.\mathsf{Dec}(\mathsf{sk}, \mathsf{ct})$, return 1 if $d = 1$, or 0 otherwise.
- $\mathsf{Fake}(\mathsf{pk}, b, \mathfrak{R}, \mathsf{pk}^*, b^*)$: Retrieve $\overline{\mathsf{pk}} := \mathcal{E}.\mathsf{OKGen}(\mathsf{ik}; k)$, set $k^* := k$, and perform the following branches for generating r^* of $\mathfrak{R}^*$:
  - For $b^* = 0$, if $b = 0$, set $r^* := r$; else, sample $r^* \leftarrow_\$ \mathcal{E}.\mathsf{Fake}(\mathsf{pk}, 1, r, \overline{\mathsf{pk}})$.
  - For $b^* = 1$, if $b = 0$, sample $r^* \leftarrow_\$ \mathcal{E}.\mathsf{Fake}(\overline{\mathsf{pk}}, 1, r, \mathsf{pk}^*)$; else, sample $r^* \leftarrow_\$ \mathcal{E}.\mathsf{Fake}(\mathsf{pk}, 1, r, \mathsf{pk}^*)$.

For $b = 1$, *correctness* of $\mathcal{DE}$ trivially follows from that of $\mathcal{E}$; For $b = 0$, we have by *key-simulatability* and IK-CPA (induced by *key-deniability*) of $\mathcal{E}$, the randomly sampled key $\overline{\mathsf{pk}} \neq \mathsf{pk}$ with overwhelming probability, and then *correctness* follows from *full robustness* of $\mathcal{E}$. Below we further attest *dual-deniability.*

Claim 1. *$\mathcal{DE}$ is dual-deniable assuming $\mathcal{E}$ is key-deniable and key-simulatable.*

Proof. The intuition is that k takes no side information except the fresh key $\overline{\mathsf{pk}}$, due to *key-simulatability* of $\mathcal{E}$, and the fake r^* is indistinguishable from the honest r by *key-deniability* of $\mathcal{E}$. Below we expound on these reductions.

Let $\mathsf{ik} \leftarrow_\$ \mathsf{Gen}(1^\lambda)$, $(\mathsf{pk}, \mathsf{sk}), (\mathsf{pk}^*, \mathsf{sk}^*) \leftarrow_\$ \mathsf{KGen}(\mathsf{ik})$, $(b, b^*, \mathsf{st}) \leftarrow_\$ \mathcal{A}_1(\mathsf{pk}, \mathsf{pk}^*)$, the goal of $\mathcal{A}_2$ is to distinguish between the following two games:

Game 0. This is the honest opening case $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{DD}\text{-}0}$ w.r.t. the encryption of b^* under pk^*, $\mathcal{A}_2$ is actually given

$$D_{\mathsf{H}} = \big(\mathfrak{R} := (k, r), \mathsf{ct}_0 := \mathcal{E}.\mathsf{Enc}(\mathsf{pk}_{\mathsf{H}}, 1; r), \mathsf{st}\big),$$

where $\mathfrak{R}$ is sampled w.r.t. (pk^*, b^*) as in Enc, and pk_{H} is assigned as pk^* for $b^* = 1$ or $\mathcal{E}.\mathsf{OKGen}(\mathsf{ik}; k)$ for $b^* = 0$.

Game 1. This is the fake opening case $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{DD}\text{-}1}$ w.r.t. the encryption of b, the view of $\mathcal{A}_2$ changes into:

$$D_{\mathsf{F}} = \big(\mathfrak{R}^* := (k^*, r^*), \mathsf{ct}_1 := \mathcal{E}.\mathsf{Enc}(\mathsf{pk}_{\mathsf{F}}, 1; r), \mathsf{st}\big),$$

where $(k, r) \leftarrow_\$ \mathcal{R}_{\mathsf{K}} \times \mathcal{R}_{\mathsf{E}}$, pk_{F} is taken as pk for $b = 1$ or $\mathcal{E}.\mathsf{OKGen}(\mathsf{ik}; k)$ for $b = 0$, and $\mathfrak{R}^*$ is sampled from $\mathsf{Fake}(\mathsf{pk}, b, (k, r), \mathsf{pk}^*, b^*)$.

Now we consider the four possible values of (b, b^*) chosen by $\mathcal{A}_1$:

– $(b, b^*) = (1, 1)$, the outside view of $(D_{\mathsf{H}}, D_{\mathsf{F}})$ is:

$$\big(k, r, \mathsf{ct}_0 := \mathcal{E}.\mathsf{Enc}(\mathsf{pk}^*, 1; r)\big),\ \big(k, r^*, \mathsf{ct}_1 := \mathcal{E}.\mathsf{Enc}(\mathsf{pk}, 1; r)\big),$$

where $(k, r) \leftarrow_\$ \mathcal{R}_{\mathsf{K}} \times \mathcal{R}_{\mathsf{E}}$ and $r^* \leftarrow_\$ \mathcal{E}.\mathsf{Fake}(\mathsf{pk}, 1, r, \mathsf{pk}^*)$. Note that k is just a random element over $\mathcal{R}_{\mathsf{K}}$. Thus, $(D_{\mathsf{H}}, D_{\mathsf{F}})$ can be seen as a valid instance pair w.r.t. $m = 1$ and $(\mathsf{pk}, \mathsf{pk}^*)$ for the key-deniable game against $\mathcal{E}$.

– $(b, b^*) = (0, 1)$, the outside view of $(D_{\mathsf{H}}, D_{\mathsf{F}})$ is:

$$\big(k, r, \mathsf{ct}_0 := \mathcal{E}.\mathsf{Enc}(\mathsf{pk}^*, 1; r)\big),\ \big(k, r^*, \mathsf{ct}_1 := \mathcal{E}.\mathsf{Enc}(\overline{\mathsf{pk}}, 1; r)\big),$$

where $(k, r) \leftarrow_\$ \mathcal{R}_{\mathsf{K}} \times \mathcal{R}_{\mathsf{E}}$ and $r^* \leftarrow_\$ \mathcal{E}.\mathsf{Fake}(\overline{\mathsf{pk}}, 1, r, \mathsf{pk}^*)$ for a fresh public key $\overline{\mathsf{pk}} := \mathcal{E}.\mathsf{OKGen}(\mathsf{ik}; k)$. Note that by *key-simulatability* of $\mathcal{E}$, $\overline{\mathsf{pk}}$ is computationally indistinguishable from the one randomly sampled by $\mathcal{E}.\mathsf{KGen}(\mathsf{ik})$, and k is further computationally indistinguishable from the one simulated by

$\mathcal{E}.\mathsf{IKGen}(\overline{\mathsf{pk}})$, which leaks no extra information except $\overline{\mathsf{pk}}$ itself. Thus, $(D_\mathsf{H}, D_\mathsf{F})$ can also be seen as a valid instance pair w.r.t. $m = 1$ and $(\overline{\mathsf{pk}}, \mathsf{pk}^*)$ for the key-deniable game against $\mathcal{E}$.

- $(b, b^*) = (1, 0)$, the outside view of $(D_\mathsf{H}, D_\mathsf{F})$ is:
$$\big(k, r, \mathsf{ct}_0 := \mathcal{E}.\mathsf{Enc}(\overline{\mathsf{pk}}, 1; r)\big),\ \big(k, r^*, \mathsf{ct}_1 := \mathcal{E}.\mathsf{Enc}(\mathsf{pk}, 1; r)\big),$$
where $(k, r) \leftarrow_\$ \mathcal{R}_\mathsf{K} \times \mathcal{R}_\mathsf{E}$ and $r^* \leftarrow_\$ \mathcal{E}.\mathsf{Fake}(\mathsf{pk}, 1, r, \overline{\mathsf{pk}})$ for a fresh public key $\overline{\mathsf{pk}} := \mathcal{E}.\mathsf{OKGen}(\mathsf{ik}; k)$. By the same arguments as the above sub-case, we have that $(D_\mathsf{H}, D_\mathsf{F})$ can be seen as a valid instance pair w.r.t. $m = 1$ and $(\mathsf{pk}, \overline{\mathsf{pk}})$ for the key-deniable game against $\mathcal{E}$.
- $(b, b^*) = (0, 0)$, the outside view of $(D_\mathsf{H}, D_\mathsf{F})$ is:
$$\big(k, r, \mathsf{ct}_0 := \mathcal{E}.\mathsf{Enc}(\overline{\mathsf{pk}}, 1; r)\big),\ \big(k, r, \mathsf{ct}_1 := \mathcal{E}.\mathsf{Enc}(\overline{\mathsf{pk}}, 1; r)\big),$$
where $(k, r) \leftarrow_\$ \mathcal{R}_\mathsf{K} \times \mathcal{R}_\mathsf{E}$ and $\overline{\mathsf{pk}} := \mathcal{E}.\mathsf{OKGen}(\mathsf{ik}; k)$. It is easy to see that now D_H and D_F share the very same distribution.

Taken together, we conclude that *Game* 0 and 1 are computationally indistinguishable if $\mathcal{E}$ is both key-deniable and key-simulatable, so the claim holds. □

3.4 Plan-Ahead DDPKE

To allow effective constructions, the literature of MDPKE [2, 10, 33] has always served under *plan-ahead* setting. In context of DDPKE, it requires the sender to determine the fake materials at the time of encryption as follows:

- the fake message m^* from the supporting message space $\mathcal{M}$ of the system;
- the fake key pk^* from the current available key set under the same ik associated with the real key pk, i.e., the bulletin board certificated by the same PKI associated with pk, which shall be denoted by $\mathcal{P}_\mathsf{ik}$ for simplicity.

Both $\mathcal{M}$ and $\mathcal{P}_\mathsf{ik}$ are taken as implicit inputs of sender-related algorithms Enc and Fake under such setting. Accordingly, the fake randomness r^* can only be generated for these *plan-ahead* elements. Although limited, such setting is sufficient in many cases if the sender just wants to mask the real plaintext/receiver with some known messages/identities, e.g., transaction dates or voter lists (see more detailed justifications at App. A).

Under this mode, we follow the routine of [10, 33] to extend the encryption algorithm Enc w.r.t. standard DDPKE by taking as auxiliary input a fake pair (pk^*, m^*):

- $\mathsf{Enc}(\mathsf{pk}, m, \mathsf{pk}^*, m^*; r)$: On inputting a public key pk and message m, a fake public key pk^* and message m^*, use randomness $r \leftarrow_\$ \mathcal{R}_\mathsf{E}$ to generate a ciphertext ct.

In turn, for standard DDPKE, the fake algorithm Fake should also return an auxiliary pair $(\mathsf{pk}^{**}, m^{**})$ in addition to the original fake randomness r^*. On the other hand, regarding weak DDPKE, only algorithm DEnc takes as auxiliary input a fake pair (pk^*, m^*), while algorithm Enc keeps unchanged. Thus, the fake algorithm Fake (associated with Enc) also maintains the inteface quo. Finally, we make slight changes for the related security notions as follows:

Correctness asks that for all security parameter $\lambda \in \mathbb{N}$, initial key $\mathsf{ik} \leftarrow_{\$} \mathsf{Gen}(1^\lambda)$, real key pair $(\mathsf{pk}, \mathsf{sk}) \leftarrow_{\$} \mathsf{KGen}(\mathsf{ik})$, real message $m \in \mathcal{M}$, fake key and message $(\mathsf{pk}^*, m^*) \in \{0,1\}^* \times \{0,1\}^*$, and encryption randomness $r \leftarrow_{\$} \mathcal{R}_{\mathsf{E}}$, it holds that

$$\mathbb{P}\left[\mathsf{Dec}\big(\mathsf{sk}, \mathsf{Enc}(\mathsf{pk}, m, \mathsf{pk}^*, m^*; r)\big) = m\right] = 1 - \mathsf{negl}(\lambda),$$

where the encryption algorithm switches to DEnc with randomness space $\mathcal{R}_{\mathsf{DE}}$ for weak mode.

CPA/CCA, *KROB*, and *XROB* additionally ask the adversary to submit the auxiliary encryption material (pk^*, m^*), while *FROB* remains unchanged. To be less repetitive, below we only give the definitions of IK-CCA and XROB w.r.t. plan-ahead DDPKE and refer to App. B for other adaptive ones.

Definition 11 (IK-CCA and XROB of Plan-Ahead DDPKE). *A plan-ahead DDPKE is IK-CCA if for all PPT adversary $\mathcal{A} := (\mathcal{A}_1, \mathcal{A}_2)$, the absolute difference of probability of returning 1 between experiment $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{pIK}\text{-}0}$ and $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{pIK}\text{-}1}$ is negligible, and is XROB if for all PPT adversary $\mathcal{A}$, the probability of returning 1 for experiment $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{pXROB}}$ is negligible.*

```
Experiment: $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{pIK}\text{-}b}(1^\lambda)$

$\mathsf{ik} \leftarrow_{\$} \mathsf{Gen}(1^\lambda)$.
$(\mathsf{pk}_0, \mathsf{sk}_0) \leftarrow_{\$} \mathsf{KGen}(\mathsf{ik}), (\mathsf{pk}_1, \mathsf{sk}_1) \leftarrow_{\$} \mathsf{KGen}(\mathsf{ik})$.
$\mathcal{P}_{\mathsf{ik}} := \{\mathsf{pk}_0, \mathsf{pk}_1\}$.
$(m, \mathsf{pk}^*, m^*, \mathsf{st}) \leftarrow_{\$} \mathcal{A}_1^{\mathcal{D}_{\mathsf{sk}_0}(\cdot), \mathcal{D}_{\mathsf{sk}_1}(\cdot)}(\mathsf{pk}_0, \mathsf{pk}_1)$.
$\mathsf{ct} := \mathsf{Enc}(\mathsf{pk}_b, m, \mathsf{pk}^*, m^*)$.
$b' \leftarrow_{\$} \mathcal{A}_2^{\mathcal{D}_{\mathsf{sk}_0}(\neg\mathsf{ct}), \mathcal{D}_{\mathsf{sk}_1}(\neg\mathsf{ct})}(\mathsf{ct}, \mathsf{st})$.
Return $b'$.
```

```
Experiment: $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{pXROB}}(1^\lambda)$

$\mathsf{ik} \leftarrow_{\$} \mathsf{Gen}(1^\lambda)$.
$(m_0, \mathsf{pk}_0, m_0^*, \mathsf{pk}_0^*, r_0, \mathsf{sk}_1) \leftarrow_{\$} \mathcal{A}(\mathsf{ik})$.
$\mathsf{ct}_0 := \mathsf{Enc}(\mathsf{pk}_0, m_0, \mathsf{pk}_0^*, m_0^*; r_0)$.
$m_1 := \mathsf{Dec}(\mathsf{sk}_1, \mathsf{ct}_0)$.
Return $(\mathsf{pk}_0 \neq \mathsf{pk}_1) \wedge (m_0 \neq \perp)$
          $\wedge (m_1 \neq \perp)$.
```

Note that in the left codes, pk^* is not included in $\mathcal{P}_{\mathsf{ik}}$, this is because pk^* may be maliciously generated by $\mathcal{A}_1$ and so invalid under ik.

Plan-ahead dual-deniability invokes Enc with extra input (pk^*, m^*) for $b = 1$, and random pair $(\mathsf{pk}^{**}, m^{**})$ over $\mathcal{P}_{\mathsf{ik}} \times \mathcal{M}$ for $b = 0$; while weak notion only shifts the case of $b = 1$ by invoking DEnc with extra pair (pk^*, m^*).

Definition 12 (Plan-Ahead (Weak) Dual-Deniability). *A plan-ahead DDPKE satisfies dual-deniability (resp., weak dual-deniability) if for all PPT adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$, the absolute difference of probability of returning 1 between experiment $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{pDD}\text{-}1}$ and $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{pDD}\text{-}0}$ (resp., $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{pwDD}\text{-}1}$ and $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{pwDD}\text{-}1}$) is negligible.*

```
Experiment: $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{pDD}\text{-}b}(1^\lambda)$

$\mathsf{ik} \leftarrow_\$ \mathsf{Gen}(1^\lambda)$.
$(\mathsf{pk}, \mathsf{sk}), (\mathsf{pk}^*, \mathsf{sk}^*) \leftarrow_\$ \mathsf{KGen}(\mathsf{ik})$.
$\mathcal{P}_{\mathsf{ik}} := \{\mathsf{pk}, \mathsf{pk}^*\}$.
$(m, m^*, \mathsf{st}) \leftarrow_\$ \mathcal{A}_1(\mathsf{pk}, \mathsf{pk}^*)$.
$\mathsf{pk}^{**} \leftarrow_\$ \mathcal{P}_{\mathsf{ik}}, m^{**} \leftarrow_\$ \mathcal{M}, r \leftarrow_\$ \mathcal{R}_{\mathsf{E}}$.
$R := (\mathsf{pk}^{**}, m^{**}, r)$.
$R^* \leftarrow_\$ \mathsf{Fake}(\mathsf{pk}, m, r, \mathsf{pk}^*, m^*)$.
$D_0 := (R, \mathsf{Enc}(\mathsf{pk}^*, m^*, \mathsf{pk}^{**}, m^{**}; r))$.
$D_1 := (R^*, \mathsf{Enc}(\mathsf{pk}, m, \mathsf{pk}^*, m^*; r))$.
$b' \leftarrow_\$ \mathcal{A}_2(D_b, \mathsf{st})$.
Return $b'$.
```

```
Experiment: $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{pwDD}\text{-}b}(1^\lambda)$

$\mathsf{ik} \leftarrow_\$ \mathsf{Gen}(1^\lambda)$.
$(\mathsf{pk}, \mathsf{sk}), (\mathsf{pk}^*, \mathsf{sk}^*) \leftarrow_\$ \mathsf{KGen}(\mathsf{ik})$.
$\mathcal{P}_{\mathsf{ik}} := \{\mathsf{pk}, \mathsf{pk}^*\}$.
$(m, m^*, \mathsf{st}) \leftarrow_\$ \mathcal{A}_1(\mathsf{pk}, \mathsf{pk}^*)$.
$r \leftarrow_\$ \mathcal{R}_{\mathsf{E}}, r' \leftarrow_\$ \mathcal{R}_{\mathsf{DE}}$.
$r^* \leftarrow_\$ \mathsf{Fake}(\mathsf{pk}, m, r', \mathsf{pk}^*, m^*)$.
$D_0 := (r, \mathsf{Enc}(\mathsf{pk}^*, m^*; r))$.
$D_1 := (r^*, \mathsf{DEnc}(\mathsf{pk}, m, \mathsf{pk}^*, m^*; r'))$.
$b' \leftarrow_\$ \mathcal{A}_2(D_b, \mathsf{st})$.
Return $b'$.
```

Finally, for plan-ahead setting, it is trivial to verify that Prop. 2-4 still hold, while Prop. 1 is no longer satisfied due to the relaxed requirement of deniability.

4 Generic Construction of Plan-Ahead DDPKE

In this section, we move to the construction of DDPKE, where we provide a generic approach under plan-ahead setting by leveraging ciphertext-simulatable PKE. We begin with an abstract overview of the main techniques, then give the formal description of our scheme as well as its security analysis.

4.1 Roadmap of The Scheme

The abstract idea of our design is to hide the encryption $\boldsymbol{c}$ of the true message $\boldsymbol{m}$ under pk randomly within $n = \mathsf{poly}(\lambda)$ sub-ciphertexts $\{\boldsymbol{c}_i\}_{i\in[n]}$, some of which are obliviously sampled and others are masking encryptions of random $\boldsymbol{m}_i$ under random pk_i. Then, *ciphertext-simulatability* enables the sender to explain $\boldsymbol{c}$ as an oblivious ciphertext, and safely open another *plan-ahead* fake encryption to fool the coercer.

With this blueprint in mind, the goal of decryption is essentially to identify the target $\boldsymbol{c}$ from $\{\boldsymbol{c}_i\}$ using sk. This is achieved by the following two steps:

- To separate $\boldsymbol{c}$ from oblivious $\boldsymbol{c}_i$, we make $\boldsymbol{c}$ encrypt $\boldsymbol{m}$ plus a random label $\boldsymbol{u}$, and further include the OWF tag $\sigma := \mathcal{H}(\boldsymbol{u})$ in the overall ciphertext. In this way, *one-wayness* of $\mathcal{H}$ ensures that any $\boldsymbol{u}'$ decrypted from an oblivious cipher would be invalid, i.e., not match with σ. Accordingly, the other masking encryptions shall also deliver both $\boldsymbol{m}_i$ and $\boldsymbol{u}$ for fake but consistent opening with σ later.
- To locate $\boldsymbol{c}$ from masking $\boldsymbol{c}_i$, we place $\boldsymbol{c}$ at a random index $t \in [n-1]$ and all the masking ciphertexts after the index t. Further, all the oblivious elements are arranged before the index t. Then, the receiver can decode $\{\boldsymbol{c}_i\}$ in sequence and take as decryption result the very first valid message, which is exactly the one decrypted from $\boldsymbol{c}_t$ by the above approach.

In a sum, the encryption procedure features under $t \leftarrow_{\$} [n-1]$ as follows (Fig. 3): $\boldsymbol{c}_i$ is sampled from $\mathsf{OEnc}(\mathsf{ik}; r_i)$ for $i < t$; $\boldsymbol{c}_t := \mathsf{Enc}(\mathsf{pk}, \boldsymbol{m}||\boldsymbol{u}; r_t)$; $\boldsymbol{c}_{t+1}$ encrypts the *plan-ahead* fake message $\boldsymbol{m}^*$ (and $\boldsymbol{u}$) with the *plan-ahead* fake key pk^*; $\boldsymbol{c}_i := \mathsf{Enc}(\mathsf{pk}_i, \boldsymbol{m}_i||\boldsymbol{u}; r_i)$ using masking materials $(\mathsf{pk}_i, \boldsymbol{m}_i) \leftarrow_{\$} \mathcal{P}_{\mathsf{ik}} \times \mathcal{M}$ for $i > t+1$.

Finally, upon coercion, the sender just inversely samples an oblivious randomness r_t^* from $\mathsf{IEnc}(\mathsf{pk}, \boldsymbol{m}, r_t)$, and reveals r_t^* as well as $t^* = t+1$, the rest randomness $(\boldsymbol{u}, \{r_i\}_{i \neq t}, \{\mathsf{pk}_i, \boldsymbol{m}_i\}_{i > t^*+1})$, and the fake plan-ahead input $(\mathsf{pk}^{**}, \boldsymbol{m}^{**}) := (\mathsf{pk}_{t^*+1}, \boldsymbol{m}_{t^*+1})$. Since r_t^* is indistinguishable from r_t by *ciphertext-simulatability* of the underlying PKE, such fake opening of $(\{\boldsymbol{c}_i\}, \sigma)$ concerning $(\mathsf{pk}, \boldsymbol{m})$ distributes almost the same as an honest one concerning $(\mathsf{pk}^*, \boldsymbol{m}^*)$, except for the seed t^* over $[2, n]$ and t over $[n-1]$, whose distance is clearly bounded by $\mathcal{O}(\frac{1}{n})$.

4.2 Details of The Scheme

Let $n = \mathsf{poly}(\lambda)$, $\mathcal{H}$ be a secure OWF over $\{0,1\}^{h(\lambda)}$, $\mathcal{E}$ be a ciphertext-simulatable PKE with message space $\{0,1\}^{\ell(\lambda)+h(\lambda)}$, encryption (resp., oblivious ciphertext sampling) randomness space $\mathcal{R}_{\mathsf{E}}$ (resp., $\mathcal{R}_{\mathsf{O}}$). Our DDPKE scheme $\mathcal{DE}$ for message space $\mathcal{M} := \{0,1\}^{\ell(\lambda)}$ under plan-ahead setting proceeds as follows:

- $\mathsf{Gen}(1^\lambda)$: Return $\mathsf{ik} \leftarrow_{\$} \mathcal{E}.\mathsf{Gen}(1^\lambda)$.
- $\mathsf{KGen}(\mathsf{ik})$: Return $(\mathsf{pk}, \mathsf{sk}) \leftarrow_{\$} \mathcal{E}.\mathsf{KGen}(\mathsf{ik})$.
- $\mathsf{Enc}(\mathsf{pk}, \boldsymbol{m}, \mathsf{pk}^*, \boldsymbol{m}^*; \mathfrak{R})$: Take as input the designated pair $(\mathsf{pk}, \boldsymbol{m})$ and the auxiliary fake pair $(\mathsf{pk}^*, \boldsymbol{m}^*)$, conduct the following:
  1. First sample the internal randomness $\mathfrak{R}$ as follows:
     ① Sample $\boldsymbol{u} \leftarrow_{\$} \{0,1\}^h, t \leftarrow_{\$} [n-1]$.
     ② $\forall i \in [t+2, n]$, sample $\mathsf{pk}_i \leftarrow_{\$} \mathcal{P}_{\mathsf{ik}}$ and $\boldsymbol{m}_i \leftarrow_{\$} \mathcal{M}$.
     ③ $\forall i \in [n]$, sample the other encryption randomness as follows:
       – If $i < t$, sample $r_i \leftarrow_{\$} \mathcal{R}_{\mathsf{O}}$.
       – Else, sample $r_i \leftarrow_{\$} \mathcal{R}_{\mathsf{E}}$.
     ④ Set $\mathfrak{R} := \left(\boldsymbol{u}, t, \{\mathsf{pk}_i, \boldsymbol{m}_i\}_{i \in [t+2,n]}, \{r_i\}_{i \in [n]}\right)$.
  2. $\forall i \in [n]$, produce the sub-ciphertext $\boldsymbol{c}_i$ as follows.
     ① For $i < t$, generate an oblivious encryption $\boldsymbol{c}_i := \mathcal{E}.\mathsf{OEnc}(\mathsf{ik}; r_i)$.
     ② For $i = t$, generate a true encryption $\boldsymbol{c}_i := \mathcal{E}.\mathsf{Enc}(\mathsf{pk}, \boldsymbol{m}||\boldsymbol{u}; r_i)$.
     ③ For $i = t+1$, generate a fake encryption $\boldsymbol{c}_i := \mathcal{E}.\mathsf{Enc}(\mathsf{pk}^*, \boldsymbol{m}^*||\boldsymbol{u}; r_i)$.
     ④ For $i > t+1$, generate a masking encryption $\boldsymbol{c}_i := \mathcal{E}.\mathsf{Enc}(\mathsf{pk}_i, \boldsymbol{m}_i||\boldsymbol{u}; r_i)$.
  3. Set the OWF tag $\sigma := \mathcal{H}(\boldsymbol{u})$, and return $\mathsf{ct} := \left(\{\boldsymbol{c}_i\}_{i \in [n]}, \sigma\right)$.
- $\mathsf{Dec}(\mathsf{sk}, \mathsf{ct})$: Initialize $\mathsf{succ} := 0$, and perform the following to iteratively decode $\boldsymbol{c}_i$ for $i \in [n]$:
  1. Decrypt $\boldsymbol{m}'||\boldsymbol{u}' := \mathcal{E}.\mathsf{Dec}(\mathsf{sk}, \boldsymbol{c}_i)$.
  2. If $\mathcal{H}(\boldsymbol{u}') = \sigma$, set $\mathsf{succ} := 1$, stop and return $\boldsymbol{m}'$.
  3. Move to $i = i + 1$.

  After the iteration, if $\mathsf{succ} = 0$, return $\perp$.

- $\mathsf{Fake}(\mathsf{pk}, \boldsymbol{m}, \mathfrak{R}, \mathsf{pk}^*, \boldsymbol{m}^*)$: Upon input the real pair $(\mathsf{pk}, \boldsymbol{m})$, randomness $\mathfrak{R}$, and the fake pair $(\mathsf{pk}^*, \boldsymbol{m}^*)$, do the following:
  1. Set $\boldsymbol{u}^* := \boldsymbol{u}$, $t^* := t + 1$, and $(\mathsf{pk}^{**}, \boldsymbol{m}^{**}) := (\mathsf{pk}_{t^*+1}, \boldsymbol{m}_{t^*+1})$.
  2. $\forall i \in [n]$, perform the following steps.
     ① If $i = t$, sample $r_i^* \leftarrow_\$ \mathcal{E}.\mathsf{IEnc}(\mathsf{pk}, \boldsymbol{m}||\boldsymbol{u}, r_i)$.
     ② Else, set $r_i^* := r_i$. Further if $i > t^* + 1$, set $(\mathsf{pk}_i^*, \boldsymbol{m}_i^*) := (\mathsf{pk}_i, \boldsymbol{m}_i)$.
     Set $\mathfrak{R}^* := \left(\boldsymbol{u}^*, t^*, \{\mathsf{pk}_i^*, \boldsymbol{m}_i^*\}_{i\in[t^*+2,n]}, \{r_i^*\}_{i\in[n]}\right)$.
  3. Return $(\mathsf{pk}^{**}, \boldsymbol{m}^{**}, \mathfrak{R}^*)$.

Theorem 1. *Suppose that $\mathcal{E}$ is correct and $\mathcal{H}$ is one-way, then $\mathcal{DE}$ is correct under plan-ahead setting.*

Proof. We shall show that it is always the sub-ciphertext $\boldsymbol{c}_t$ that triggers the "stop" condition within the iteration of Dec, and so the true message $\boldsymbol{m}' = \boldsymbol{m}$ is returned. Below we consider two decryption cases w.r.t. oblivious or normal $\boldsymbol{c}_i$.

1. For $i < t$ (event $\mathtt{E}_1$), we shall demonstrate the following fact.

Claim 2. $\mathbb{P}[\ \mathcal{H}(\boldsymbol{u}') = \sigma |\ \mathtt{E}_1\ ] = \mathbf{Adv}_{\mathcal{H}}^{\mathsf{OWF}}$.

Proof. For event $\mathtt{E}_1$, $\boldsymbol{c}_i$ is obliviously sampled from $\mathcal{E}.\mathsf{OEnc}(\mathsf{ik}; r_i)$ using $r_i \leftarrow_\$ \mathcal{R}_\mathsf{O}$, and so the distribution for such $\boldsymbol{c}_i$ is independent of that for σ, which itself is a uniform evaluation on $\mathcal{H}$ over $\{0,1\}^h$. Hence, if $\mathcal{H}(\boldsymbol{u}') = \sigma$, we can obtain a trivial algorithm breaking the one-wayness of $\mathcal{H}$. Namely, given a challenge σ of $\mathcal{H}$ from the challenger, honestly sample $\mathsf{ik} \leftarrow_\$ \mathcal{E}.\mathsf{Gen}(1^\lambda)$, $(\mathsf{pk}, \mathsf{sk}) \leftarrow_\$ \mathcal{E}.\mathsf{KGen}(\mathsf{ik})$, $r \leftarrow_\$ \mathcal{R}_\mathsf{O}$, then generate a helper ciphertext $\boldsymbol{c} := \mathcal{E}.\mathsf{OEnc}(\mathsf{ik}; r)$ and decrypt $\boldsymbol{c}$ using sk to obtain $\boldsymbol{m}'||\boldsymbol{u}'$, finally output $\boldsymbol{u}'$ as a preimage of σ.

2. For $i \geq t$, note that $\boldsymbol{c}_t$ is a true encryption of $\boldsymbol{m}$ under pk. Then, by *correctness* of $\mathcal{E}$, we have $\boldsymbol{m}'$ is always equal to $\boldsymbol{m}$ when succ encounters to be t. Hence, $\boldsymbol{c}_t$ will first trigger the "stop" condition within $[t, n]$.

From the above analysis, we conclude that, if $i < t$, no $\boldsymbol{c}_i$ triggers the "stop" condition; otherwise, $\boldsymbol{c}_t$ first triggers. Thus, the iteration always stops at $i = t$ and so returns the correct $\boldsymbol{m}' = \boldsymbol{m}$. □

Remark 3. One may observe from Thm. 1 that the use of OWF tag σ implicitly provides $\mathcal{E}$ with some sense of robustness. Actually, we can directly apply a robust and ciphertext-simulatable PKE to build DDPKE, which can be seen as a more general version of $\mathcal{DE}$. Yet, we keep the current scheme here as it is more succinct and easy to follow, and provide a variant of $\mathcal{DE}$ at App. C.

4.3 Security Analysis

Below we prove the security of $\mathcal{DE}$ under plan-ahead setting.

Theorem 2. *Suppose that $\mathcal{E}$ is CPA-secure, then $\mathcal{DE}$ is CPA-secure under plan-ahead setting.*

Proof. We prove IK-CPA of $\mathcal{DE}$ under plan-ahead setting, proof of IM-CPA is similar. Suppose that a CPA adversary $\mathcal{A}$ against $\mathcal{DE}$ succeeds in distinguishing $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{pIK}\text{-}b}$ with non-negligible advantage ϵ, then we can build a PPT algorithm $\mathcal{B}$ that breaks IK-CPA of $\mathcal{E}$ also with advantage ϵ.

Let $\mathsf{ik} \leftarrow_{\$} \mathcal{E}.\mathsf{Gen}(1^\lambda)$, $(\mathsf{pk}_0, \mathsf{sk}_0), (\mathsf{pk}_1, \mathsf{sk}_1) \leftarrow_{\$} \mathcal{E}.\mathsf{KGen}(\mathsf{ik})$, given $(\mathsf{pk}_0, \mathsf{pk}_1)$, $\mathcal{B}$ forwards them to $\mathcal{A}$ and interacts with $\mathcal{A}$ as follows:

- **Challenge.** $\mathcal{A}$ returns $(\boldsymbol{m}, \mathsf{pk}^*, \boldsymbol{m}^*)$ to $\mathcal{B}$, with which $\mathcal{B}$ samples $\boldsymbol{u} \leftarrow_{\$} \{0,1\}^h$ and submits $\boldsymbol{m}||\boldsymbol{u}$. Then the challenger samples $b \leftarrow_{\$} \{0,1\}, r \leftarrow_{\$} \mathcal{R}_{\mathsf{E}}$, and outputs the challenge ciphertext $\boldsymbol{c} := \mathcal{E}.\mathsf{Enc}(\mathsf{pk}_b, \boldsymbol{m}||\boldsymbol{u}; r)$. Finally, $\mathcal{B}$ performs as $\mathcal{DE}.\mathsf{Enc}$ to produce a trick ciphertext for $\mathcal{A}$:
  1. Sample $t \leftarrow_{\$} [n-1]$, $r_i \leftarrow_{\$} \mathcal{R}_{\mathsf{O}}$ for $i \in [t-1]$, and $r_i \leftarrow_{\$} \mathcal{R}_{\mathsf{E}}$ for $i \in [t, n]$.
  2. $\forall i \in [n]$, generate the sub-ciphertext $\boldsymbol{c}_i$ as follows:
     ① If $i < t$, produce $\boldsymbol{c}_i := \mathcal{E}.\mathsf{OEnc}(\mathsf{ik}; r_i)$.
     ② Else if $i = t$, set $\boldsymbol{c}_i := \boldsymbol{c}$.
     ③ Else if $i = t+1$, produce $\boldsymbol{c}_i := \mathcal{E}.\mathsf{Enc}(\mathsf{pk}^*, \boldsymbol{m}^*||\boldsymbol{u}; r_i)$.
     ④ Else, set $\boldsymbol{c}_i := \mathcal{E}.\mathsf{Enc}(\mathsf{pk}_i, \boldsymbol{m}_i||\boldsymbol{u}; r_i)$ with $(\mathsf{pk}_i, \boldsymbol{m}_i) \leftarrow_{\$} \mathcal{P}_{\mathsf{ik}} \times \mathcal{M}$, where $\mathcal{P}_{\mathsf{ik}} := \{\mathsf{pk}_0, \mathsf{pk}_1\}$.
  3. Set $\sigma := \mathcal{H}(\boldsymbol{u})$ and return $\mathsf{ct} := (\{\boldsymbol{c}_1\}_{i\in[n]}, \sigma)$ to $\mathcal{A}$.
- **Guess.** $\mathcal{A}$ outputs a guess bit $b' \in \{0,1\}$, $\mathcal{B}$ also outputs b' as the guess of b.

By the above construction, $\mathcal{B}$ provides a perfect simulation of the plan-ahead IK-CPA game. In particular, if the challenger picks $b = 0$ (resp., $b = 1$), $\mathcal{A}$ is exactly in $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{pIK}\text{-}0}$ (resp., $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{pIK}\text{-}1}$). Thus, the fact that $\mathcal{A}$ wins with non-trivial probability ϵ implies that $\mathcal{B}$'s advantage of breaking IK-CPA of $\mathcal{E}$ is also ϵ. □

Theorem 3. *Suppose that $\mathcal{E}$ is ciphertext-simulatable, then $\mathcal{DE}$ is $\left(\frac{1}{n-1} + \mathsf{negl}(\lambda)\right)$-dual-deniable under plan-ahead setting.*

Proof. The intuition is that the honest and fake opening of a ciphertext for $\mathcal{DE}$ only differs in the distributions of (t, r_t) and (t^*, r_t^*). The distance $\Delta(t, t^*) = \frac{1}{n-1}$ since t (resp., t^*) is uniformly random over $[n-1]$ (resp., $[2, n]$). In addition, r_t^* is invert-sampled from $\mathcal{E}.\mathsf{IEnc}$ using r_t, and so computationally indistinguishable from r_t by *ciphertext-simulatability* of $\mathcal{E}$. Below we expound on these reductions via some hybrid games of $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{pDD}\text{-}b}$.

Game 0. This is the honest opening case $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{pDD}\text{-}0}$ w.r.t. the encryption of $\boldsymbol{m}^*$ under pk^*, $\mathcal{A}_2$ is actually given

$$D_{\mathsf{G}_0} = \left(\mathsf{pk}^{**}, \boldsymbol{m}^{**}, \mathfrak{R}, \mathsf{ct}_0 := \mathsf{Enc}(\mathsf{pk}^*, \boldsymbol{m}^{\boldsymbol{*}}, \mathsf{pk}^{**}, \boldsymbol{m}^{**}; \mathfrak{R}), \mathsf{st}\right),$$

where $\mathsf{ik} \leftarrow_{\$} \mathsf{Gen}(1^\lambda)$, $(\mathsf{pk}, \mathsf{sk}), (\mathsf{pk}^*, \mathsf{sk}^*) \leftarrow_{\$} \mathsf{KGen}(\mathsf{ik})$, $(\boldsymbol{m}, \boldsymbol{m}^*, \mathsf{st}) \leftarrow_{\$} \mathcal{A}_1(\mathsf{pk}, \mathsf{pk}^*)$. Besides, the auxiliary pair $(\mathsf{pk}^{**}, \boldsymbol{m}^{**})$ and randomness $\mathfrak{R}$ are sampled as follows:

1. Sample $\mathsf{pk}^{**} \leftarrow_{\$} \mathcal{P}_{\mathsf{ik}}$ where $\mathcal{P}_{\mathsf{ik}} := \{\mathsf{pk}, \mathsf{pk}^*\}$, and $\boldsymbol{m}^{**} \leftarrow_{\$} \mathcal{M}$.
2. Sample $\boldsymbol{u} \leftarrow_{\$} \{0,1\}^h, t \leftarrow_{\$} [n-1]$.
3. $\forall i \in [t+2, n]$, sample $\mathsf{pk}_i \leftarrow_{\$} \mathcal{P}_{\mathsf{ik}}$ and $\boldsymbol{m}_i \leftarrow_{\$} \mathcal{M}$.
4. $\forall i \in [n]$, if $i < t$, sample $r_i \leftarrow_{\$} \mathcal{R}_{\mathsf{O}}$; otherwise, sample $r_i \leftarrow_{\$} \mathcal{R}_{\mathsf{E}}$.
5. Set $\mathfrak{R} := \left(\boldsymbol{u}, t, \{\mathsf{pk}_i, \boldsymbol{m}_i\}_{i\in[t+2,n]}, \{r_i\}_{i\in[n]}\right)$.

Game 1. This game changes the sampling way of t, the view from $\mathcal{A}_2$ becomes

$$D_{\mathsf{G}_1} = \left(\mathsf{pk}^{**}, \boldsymbol{m}^{**}, \mathfrak{R}^{'}, \mathsf{ct}_1 := \mathsf{Enc}(\mathsf{pk}^*, \boldsymbol{m}^*, \mathsf{pk}^{**}, \boldsymbol{m}^{**}; \mathfrak{R}^{'}), \mathsf{st}\right),$$

where the randomness $\mathfrak{R}^{'}$ is sampled as follows:

1. Sample $\mathsf{pk}^{**} \leftarrow_\$ \mathcal{P}_{\mathsf{ik}}$ where $\mathcal{P}_{\mathsf{ik}} := \{\mathsf{pk}, \mathsf{pk}^*\}$, and $\boldsymbol{m}^{**} \leftarrow_\$ \mathcal{M}$.
2. Sample $\boldsymbol{u} \leftarrow_\$ \{0,1\}^h, t \leftarrow_\$ [n-1]$, set $t' = t+1$.
3. $\forall i \in [t'+2, n]$, sample $\mathsf{pk}_i \leftarrow_\$ \mathcal{P}_{\mathsf{ik}}$ and $\boldsymbol{m}_i \leftarrow_\$ \mathcal{M}$.
4. $\forall i \in [n]$, if $i < t'$, sample $r_i \leftarrow_\$ \mathcal{R}_{\mathsf{O}}$; otherwise, sample $r_i \leftarrow_\$ \mathcal{R}_{\mathsf{E}}$.
5. Set $\mathfrak{R}' := \left(\boldsymbol{u}, t', \{\mathsf{pk}_i, \boldsymbol{m}_i\}_{i\in[t'+2,n]}, \{r_i\}_{i\in[n]}\right)$.

Note that D_{G_1} differs from D_{G_0} only in the distribution of t'. Recall that t is uniform random over $[n-1]$, and so $t' = t+1$ is uniform random over $[2,n]$. Hence, it holds that $\Delta(t,t') = \frac{1}{n-1}$, and further $\left|\mathbb{P}[b'_{\mathsf{G}_1} = 1] - \mathbb{P}[b'_{\mathsf{G}_0} = 1]\right| \leq \frac{1}{n-1}$.

Game 2. This is the fake opening case $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{pDD\text{-}1}}$ w.r.t. the encryption of $\boldsymbol{m}$ under pk, the view of $\mathcal{A}_2$ changes into

$$D_{\mathsf{G}_2} = \left(\mathsf{pk}^{**}, \boldsymbol{m}^{**}, \mathfrak{R}^*, \mathsf{ct}_2 := \mathsf{Enc}(\mathsf{pk}, \boldsymbol{m}, \mathsf{pk}^*, \boldsymbol{m}^*; \mathfrak{R}), \mathsf{st}\right),$$

where the real randomness $\mathfrak{R} = \left(\boldsymbol{u}, t, \{\mathsf{pk}_i, \boldsymbol{m}_i\}_{i\in[t+2,n]}, \{r_i\}_{i\in[n]}\right)$ is sampled in the same way as that in *Game* 0, and the fake auxiliary pair $(\mathsf{pk}^{**}, \boldsymbol{m}^{**})$ and randomness $\mathfrak{R}^*$ are sampled from $\mathsf{Fake}(\mathsf{pk}, \boldsymbol{m}, \mathfrak{R}, \mathsf{pk}^*, \boldsymbol{m}^*)$ as follows:

1. Set $(\mathsf{pk}^{**}, \boldsymbol{m}^{**}) := (\mathsf{pk}_{t+2}, \boldsymbol{m}_{t+2})$.
2. Set $\boldsymbol{u}^* := \boldsymbol{u}$ and $t^* := t+1$.
3. $\forall i \in [n] \setminus \{t\}$, set $r_i^* := r_i$. Further if $i > t^*+1$, set $(\mathsf{pk}_i^*, \boldsymbol{m}_i^*) := (\mathsf{pk}_i, \boldsymbol{m}_i)$.
4. Sample $r_t^* \leftarrow_\$ \mathcal{E}.\mathsf{IEnc}(\mathsf{pk}, \boldsymbol{m}||\boldsymbol{u}, r_t)$.
5. Set $\mathfrak{R}^* := \left(\boldsymbol{u}^*, t^*, \{\mathsf{pk}_i^*, \boldsymbol{m}_i^*\}_{i\in[t^*+2,n]}, \{r_i^*\}_{i\in[n]}\right)$.

By *ciphertext-simulatability* of $\mathcal{E}$, the sub-ciphertext $\boldsymbol{c}_{2,t} := \mathcal{E}.\mathsf{Enc}(\mathsf{pk}, \boldsymbol{m}; r_t)$ of ct_2 can be explained as $\mathcal{E}.\mathsf{Enc}(\mathsf{ik}; r_t^*)$. Thus, the overall ct_2 can be explained as $\mathsf{Enc}(\mathsf{pk}^*, \boldsymbol{m}^*, \mathsf{pk}^{**}, \boldsymbol{m}^{**}; \mathfrak{R}^*)$. In particular, we shall show that D_{G_2} and D_{G_1} are computationally indistinguishable by the following observations:

- $(\mathsf{pk}^{**}, \boldsymbol{m}^{**})$ in both *Game* 1 and 2 is uniformly random over $\mathcal{P}_{\mathsf{ik}} \times \mathcal{M}$.
- Both $\boldsymbol{u}$ and $\boldsymbol{u}^*$ are uniformly random over $\{0,1\}^h$.
- Both t' and t^* are uniformly random over $[2,n]$ since $t \leftarrow_\$ [n-1]$.
- For $i \in [t^*+2, n]$ (resp., $[t'+2, n]$), $(\mathsf{pk}_i^*, \boldsymbol{m}_i^*)$ (resp., $(\mathsf{pk}_i, \boldsymbol{m}_i)$) is uniformly random over $\mathcal{P}_{\mathsf{ik}} \times \mathcal{M}$.
- For $i \in [t-1]$, both r_i^* and r_i are uniformly random over $\mathcal{R}_{\mathsf{O}}$.
- For $i \in [t+1, n]$, both r_i^* and r_i are uniformly random over $\mathcal{R}_{\mathsf{E}}$.
- For $i = t$, $r_t \leftarrow_\$ \mathcal{R}_{\mathsf{O}}$, $\boldsymbol{c}_{1,t} := \mathcal{E}.\mathsf{OEnc}(\mathsf{ik}; r_t)$ in D_{G_1}; while $r_t \leftarrow_\$ \mathcal{R}_{\mathsf{E}}$, $\boldsymbol{c}_{2,t} := \mathcal{E}.\mathsf{Enc}(\mathsf{pk}, \boldsymbol{m}; r_t)$, $r_t^* := \mathcal{E}.\mathsf{IEnc}(\mathsf{pk}, \boldsymbol{m}, r_t)$ in D_{G_2}. Then by *ciphertext-simulatability* of $\mathcal{E}$, we have $(r_t, \boldsymbol{c}_{1,t})$ and $(r_t^*, \boldsymbol{c}_{2,t})$ are computationally indistinguishable from the view of $\mathcal{A}_2$, even if it can learn both pk and $\boldsymbol{m}$.

With these facts, we have that all the elements of D_{G_2} are indistinguishable from those of D_{G_1}. Therefore we have

$$\left|\mathbb{P}[b'_{\mathsf{G}_1} = 1] - \mathbb{P}[b'_{\mathsf{G}_0} = 1]\right| \leq \mathbf{Adv}_{\mathcal{E}}^{\mathsf{CS}} = \mathsf{negl}(\lambda).$$

Further, by combining all the above analysis, we arrive at the desired result:

$$\left|\mathbb{P}[b' = 1 \mid \mathbf{Exp}_{\mathcal{A}}^{\mathsf{pDD\text{-}1}}] - \mathbb{P}[b' = 1 \mid \mathbf{Exp}_{\mathcal{A}}^{\mathsf{pDD\text{-}0}}]\right| \leq \frac{1}{n-1} + \mathsf{negl}(\lambda).$$

□

5 Achieving Negligible Detection Probability

In this section, we forward to build DDPKE with negligible detection advantage, where we present a more efficient construction under the weak model and show an existing $i\mathcal{O}$-based scheme is inherently dual-deniable.

5.1 New Weak-mode Construction

Weak-mode DDPKE contains two encryption algorithms Enc and DEnc (Def. 10), and a sender can first run DEnc and later claim she has invoked Enc. We shall simplify the framework $\mathcal{DE}$ in §.4 by use of this flexible feature.

Recall that $\mathcal{DE}$ hides $\boldsymbol{m}$ at a random $t \in [n-1]$ and reveals $t^* = t+1$ as the fake coin used for Enc. Thus, the detection advantage depends heavily on the difference between t and t^*), which is scaled by the length n of the overall ciphertext. Now, by also switching the encryption algorithm, we can explain the index t used in DEnc as t^* used in Enc, without the requirement of indistinguishability between the two randomness. Then n can be minimized to be a constant 2, i.e., $\boldsymbol{m}$ is always hidden at $t = 1$ and $\boldsymbol{m}^*$ is encrypted at $t^* = 2$. Accordingly, DEnc plays the role of the usual encryption by conducting:

$$\boldsymbol{c}_1 \leftarrow_{\$} \mathcal{E}.\mathsf{Enc}(\mathsf{pk}, \boldsymbol{m}||\boldsymbol{u}),\ \boldsymbol{c}_2 \leftarrow_{\$} \mathcal{E}.\mathsf{Enc}(\mathsf{pk}^*, \boldsymbol{m}^*||\boldsymbol{u});$$

while Enc serves as fake opening by “flipping” $\boldsymbol{c}_1$ as an obliviously sampled one:

$$\boldsymbol{c}_1 \leftarrow_{\$} \mathcal{E}.\mathsf{OEnc}(\mathsf{ik}),\ \boldsymbol{c}_2 \leftarrow_{\$} \mathcal{E}.\mathsf{Enc}(\mathsf{pk}, \boldsymbol{m}||\boldsymbol{u}).$$

In this way, the difference between DEnc and Enc only depends on the distance between a real encryption and an obliviously sampled one, which becomes negligible by *ciphertext-simulatability* of the underlying PKE.

Based on the above insights, we present the weakly dual-deniable scheme $w\mathcal{DE}$ under plan-ahead setting, where we adopt the notations in §.4.2.

- $\mathsf{Gen}(1^\lambda)$: Return $\mathsf{ik} \leftarrow_{\$} \mathcal{E}.\mathsf{Gen}(1^\lambda)$.
- $\mathsf{KGen}(\mathsf{ik})$: Return $(\mathsf{pk}, \mathsf{sk}) \leftarrow_{\$} \mathcal{E}.\mathsf{KGen}(1^\lambda)$.
- $\mathsf{Enc}(\mathsf{pk}, \boldsymbol{m}; \mathfrak{R})$: On input the public key pk and message $\boldsymbol{m}$, run as follows:
  1. Sample $\mathfrak{R} := (\boldsymbol{u}, r_1, r_2) \leftarrow_{\$} \{0,1\}^h \times \mathcal{R}_{\mathsf{O}} \times \mathcal{R}_{\mathsf{E}}$.
  2. Generate an oblivious encryption $\boldsymbol{c}_1 := \mathcal{E}.\mathsf{OEnc}(\mathsf{ik}; r_1)$.
  3. Generate a true encryption $\boldsymbol{c}_2 := \mathcal{E}.\mathsf{Enc}(\mathsf{pk}, \boldsymbol{m}||\boldsymbol{u}; r_2)$.
  4. Set $\sigma := \mathcal{H}(\boldsymbol{u})$ and return $\mathsf{ct} := (\boldsymbol{c}_1, \boldsymbol{c}_2, \sigma)$.
- $\mathsf{DEnc}(\mathsf{pk}, \boldsymbol{m}, \mathsf{pk}^*, \boldsymbol{m}^*; \mathfrak{R})$: On input the designated pair $(\mathsf{pk}, \boldsymbol{m})$ and the auxiliary pair $(\mathsf{pk}^*, \boldsymbol{m}^*)$, run as follows:

1. Sample $\mathfrak{R} := (\boldsymbol{u}, r_1, r_2) \leftarrow_{\$} \{0,1\}^h \times \mathcal{R}_{\mathsf{E}} \times \mathcal{R}_{\mathsf{E}}$.
2. Generate a true encryption $\boldsymbol{c}_1 := \mathcal{E}.\mathsf{Enc}(\mathsf{pk}, \boldsymbol{m}||\boldsymbol{u}; r_1)$.
3. Generate a fake encryption $\boldsymbol{c}_2 := \mathcal{E}.\mathsf{Enc}(\mathsf{pk}^*, \boldsymbol{m}^*||\boldsymbol{u}; r_2)$.
4. Set $\sigma := \mathcal{H}(\boldsymbol{u})$ and return $\mathsf{ct} := (\boldsymbol{c}_1, \boldsymbol{c}_2, \sigma)$.

- $\mathsf{Dec}(\mathsf{sk}, \mathsf{ct})$: Initialize $\mathsf{succ} := 0$ and do the following for $i \in [2]$:
  1. Decrypt $\boldsymbol{m}'||\boldsymbol{u}' := \mathcal{E}.\mathsf{Dec}(\mathsf{sk}, \boldsymbol{c}_i)$.
  2. If $\mathcal{H}(\boldsymbol{u}') = \sigma$, set $\mathsf{succ} := 1$, stop and return $\boldsymbol{m}'$.
  3. Move to $i = i + 1$.

  After the iteration, if $\mathsf{succ} = 0$, return $\perp$.
- $\mathsf{Fake}(\mathsf{pk}, \boldsymbol{m}, \mathfrak{R}, \mathsf{pk}^*, \boldsymbol{m}^*)$: Upon input the real pair $(\mathsf{pk}, \boldsymbol{m})$, randomness $\mathfrak{R}$, and the fake pair $(\mathsf{pk}^*, \boldsymbol{m}^*)$, perform as follows:
  1. Generate $r_1^* \leftarrow_{\$} \mathcal{E}.\mathsf{IEnc}(\mathsf{pk}, \boldsymbol{m}||\boldsymbol{u}, r_1)$ and set $(\boldsymbol{u}^*, r_2^*) = (\boldsymbol{u}, r_2)$.
  2. Return $\mathfrak{R}^* := (\boldsymbol{u}^*, r_1^*, r_2^*)$.

Theorem 4. *Suppose that $\mathcal{E}$ is correct and $\mathcal{H}$ is one-way, then $w\mathcal{DE}$ is correct under plan-ahead setting.*

Proof. First prove *correctness* of DEnc. Since $\boldsymbol{c}_1$ is the true encryption of $\boldsymbol{m}||\boldsymbol{u}$ under pk, by *correctness* of $\mathcal{E}$ we have the output of $\mathsf{Dec}(\mathsf{sk}, \boldsymbol{c}_1)$ is exactly $\boldsymbol{m}||\boldsymbol{u}$, which satisfies $\mathcal{H}(\boldsymbol{u}) = \sigma$. Thus, Dec will return the correct $\boldsymbol{m}$ at $i = 1$.

To prove *correctness* of Enc, note that now $\boldsymbol{c}_1$ is obliviously sampled from $\mathcal{E}.\mathsf{OEnc}$, then by Claim 2 and the one-wayness of $\mathcal{H}$, we have $\mathcal{H}(\boldsymbol{u}') = \sigma$ with only negligible probability. Thus the decryption will go to $i = 2$ where $\boldsymbol{c}_2$ is the true encryption of $\boldsymbol{m}||\boldsymbol{u}$ under pk, then by *correctness* of $\mathcal{E}$ we have that now $\mathcal{H}(\boldsymbol{u}') = \sigma$ holds and so the correct $\boldsymbol{m}' = \boldsymbol{m}$ is returned. □

Theorem 5. *Suppose that $\mathcal{E}$ is CPA-secure, then $w\mathcal{DE}$ is CPA secure w.r.t. both encryption algorithms under plan-ahead setting.*

Proof. We prove IK-CPA of $w\mathcal{DE}$ under *plan-ahead* setting, proof of IM-CPA is similar. First prove the case of $w\mathcal{DE}^{\mathsf{T}} := \langle \mathsf{Gen}, \mathsf{KGen}, \mathsf{Enc}, \mathsf{Dec} \rangle$. Given a CPA adversary $\mathcal{A}$ against $w\mathcal{DE}^{\mathsf{T}}$ with non-negligible advantage ϵ in distinguishing $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{pIK}\text{-}b}$, we build a PPT algorithm $\mathcal{B}$ breaking IK-CPA of $\mathcal{E}$ also with advantage ϵ. Let $\mathsf{ik} \leftarrow_{\$} \mathcal{E}.\mathsf{Gen}(1^\lambda)$, $(\mathsf{pk}_0, \mathsf{sk}_0), (\mathsf{pk}_1, \mathsf{sk}_1) \leftarrow_{\$} \mathcal{E}.\mathsf{KGen}(\mathsf{ik})$, given $(\mathsf{pk}_0, \mathsf{pk}_1)$, $\mathcal{B}$ forwards them to $\mathcal{A}$ and interacts with $\mathcal{A}$ as follows:

- **Challenge.** $\mathcal{A}$ returns $\boldsymbol{m}$ to $\mathcal{B}$, with which $\mathcal{B}$ samples $\boldsymbol{u} \leftarrow_{\$} \{0,1\}^h$ and submits $\boldsymbol{m}||\boldsymbol{u}$. Then the challenger samples $b \leftarrow_{\$} \{0,1\}$, $r \leftarrow_{\$} \mathcal{R}_{\mathsf{E}}$, and outputs the challenge ciphertext $\boldsymbol{c} := \mathcal{E}.\mathsf{Enc}(\mathsf{pk}_b, \boldsymbol{m}||\boldsymbol{u}; r)$. Finally, $\mathcal{B}$ performs as $w\mathcal{DE}.\mathsf{Enc}$ to produce a trick ciphertext for $\mathcal{A}$:
  1. Sample $r_1 \leftarrow_{\$} \mathcal{R}_{\mathsf{O}}$, generate an oblivious encryption $\boldsymbol{c}_1 := \mathcal{E}.\mathsf{OEnc}(\mathsf{ik}; r_1)$.
  2. Set $\sigma = \mathcal{H}(\boldsymbol{u})$ and return $\mathsf{ct} := (\boldsymbol{c}_1, \boldsymbol{c}, \sigma)$ to $\mathcal{A}$.
- **Guess.** $\mathcal{A}$ outputs a guess bit $b' \in \{0,1\}$, $\mathcal{B}$ also outputs b' as the guess of b.

By the above construction, $\mathcal{B}$ provides a perfect simulation of the IK-CPA game under plan-ahead setting. Moreover, if the challenger takes $b = 0$, $\mathcal{A}$ is exactly in $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{pIK\text{-}0}}$; otherwise, it is in $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{pIK\text{-}1}}$. Thus, the fact that $\mathcal{A}$ wins with non-trivial probability ϵ implies that $\mathcal{B}$'s advantage also ϵ.

The proof of IK-CPA under $w\mathcal{DE}^{\mathsf{F}} := \langle \mathsf{Gen}, \mathsf{KGen}, \mathsf{DEnc}, \mathsf{Dec} \rangle$ takes the same techniques, except that $\mathcal{A}_1$ submits $\boldsymbol{m}$ as well as an auxiliary pair $(\mathsf{pk}^*, \boldsymbol{m}^*)$. Also, $\mathcal{B}$ performs as $w\mathcal{DE}.\mathsf{DEnc}$ to produce a trick ciphertext for $\mathcal{A}$ as follows:

1. Sample $r_2 \leftarrow_{\$} \mathcal{R}_{\mathsf{E}}$, generate a fake encryption $\boldsymbol{c}_2 := \mathcal{E}.\mathsf{Enc}(\mathsf{pk}^*, \boldsymbol{m}^* || \boldsymbol{u}; r_2)$.
2. Set $\sigma = \mathcal{H}(\boldsymbol{u})$ and return $\mathsf{ct} := (\boldsymbol{c}, \boldsymbol{c}_2, \sigma)$.

In this way, it still holds that if the challenger takes $b = 0$, $\mathcal{A}$ is in $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{pIK\text{-}0}}$; otherwise, it is in $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{pIK\text{-}1}}$, which completes the reduction. □

Theorem 6. *Suppose that $\mathcal{E}$ is ciphertext-simulatable, then $w\mathcal{DE}$ is weakly dual-deniable under plan-ahead setting.*

Proof. Consider $\mathcal{A}_2$'s view in games $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{pwDD\text{-}1}}$ and $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{pwDD\text{-}0}}$ with $\mathsf{ik} \leftarrow_{\$} \mathsf{Gen}(1^\lambda)$, $(\mathsf{pk}, \mathsf{sk}), (\mathsf{pk}^*, \mathsf{sk}^*) \leftarrow_{\$} \mathsf{KGen}(\mathsf{ik})$, $(\boldsymbol{m}, \boldsymbol{m}^*, \mathsf{st}) \leftarrow_{\$} \mathcal{A}_1(\mathsf{pk}, \mathsf{pk}^*)$.

- For the honest case, it is $D_0 := (\mathfrak{R}, \mathsf{ct}_0 := \mathsf{Enc}(\mathsf{pk}^*, \boldsymbol{m}^*; \mathfrak{R}), \mathsf{st})$, where $\mathfrak{R} := (\boldsymbol{u}, r_1, r_2) \leftarrow_{\$} \{0,1\}^h \times \mathcal{R}_{\mathsf{O}} \times \mathcal{R}_{\mathsf{E}}$.
- For the fake case, it is $D_1 := (\mathfrak{R}^*, \mathsf{ct}_1 := \mathsf{DEnc}(\mathsf{pk}, \boldsymbol{m}, \mathsf{pk}^*, \boldsymbol{m}^*; \mathfrak{R}'), \mathsf{st})$, where the involved randomness $\mathfrak{R}'$ and $\mathfrak{R}^*$ are sampled as follows:
  1. Sample $\mathfrak{R}' = (\boldsymbol{u}, r_1, r_2) \leftarrow_{\$} \{0,1\}^h \times \mathcal{R}_{\mathsf{E}} \times \mathcal{R}_{\mathsf{E}}$.
  2. Sample $r_1^* \leftarrow_{\$} \mathcal{E}.\mathsf{IEnc}(\mathsf{pk}, \boldsymbol{m}||\boldsymbol{u}, r_1)$, set $\mathfrak{R}^* := (\boldsymbol{u}, r_1^*, r_2)$.

Parse $\mathsf{ct}_0 = (\boldsymbol{c}_{0,1}, \boldsymbol{c}_{0,2}, \sigma_0)$ and $\mathsf{ct}_1 = (\boldsymbol{c}_{1,1}, \boldsymbol{c}_{1,2}, \sigma_1)$, we can easily obtain the following observations. First, both $\boldsymbol{c}_{0,2}$ and $\boldsymbol{c}_{1,2}$ are true encryptions of $\boldsymbol{m}^*||\boldsymbol{u}$ under pk^* using randomness r_2. Besides, note $\boldsymbol{c}_{0,1} = \mathcal{E}.\mathsf{OEnc}(\mathsf{ik}; \mathsf{r}_1)$ and $\boldsymbol{c}_{1,1} = \mathcal{E}.\mathsf{Enc}(\mathsf{pk}, \boldsymbol{m}; r_1)$, then by *ciphertext-simulatability* of $\mathcal{E}$, $(r_1, \boldsymbol{c}_{0,1})$ and $(r_1^*, \boldsymbol{c}_{1,1})$ are computationally indistinguishable even if $\mathcal{A}_2$ can learn $(\mathsf{pk}, \boldsymbol{m})$. Based on these facts, it holds $\left|\mathbb{P}[b' = 1|\mathbf{Exp}_{\mathcal{A}}^{\mathsf{pwDD\text{-}1}}] - \mathbb{P}[b' = 1|\mathbf{Exp}_{\mathcal{A}}^{\mathsf{pwDD\text{-}0}}]\right| \leq \mathbf{Adv}_{\mathcal{E}}^{\mathsf{CS}}$. □

5.2 Revisit of iO-based Construction

In this subsection, we show that dual-deniability is implied by another flavor of deniability – *public explanation* [35] plus CPA security. Roughly speaking, public explanation supports generating fake randomness r^* with no need of the original encryption materials. Having this property in hand, to prove the indistinguishability of fake opening (r^*, ct_1) and honest encryption (r, ct_0), it suffices to further require the indistinguishability of the ciphertexts, which is exactly CPA security. In particular, the $i\mathcal{O}$-based scheme $i\mathcal{DE}$ in [35] has been proved to satisfy such deniability and IM-CPA (IK-CPA and even CCA security trivially follow from similar arguments), and so is inherently dual-deniable.

We begin with recalling the notion of public explanation, which is relative to the following adapted syntax of algorithm Fake:

- $\mathsf{Fake}(\mathsf{ct}, \mathsf{pk}^*, m^*)$: On input a ciphertext ct and a pair (pk^*, m^*), return fake randomness r^*.

Despite using notations of (pk^*, m^*), public explanation only achieves indistinguishability of fake and real randomness under the *same* key and message.

Definition 13 (Indistinguishability of Explanation). *A PKE is explanation-indistinguishable if for all PPT adversary $\mathcal{A} := (\mathcal{A}_1, \mathcal{A}_2)$, the absolute difference of probability of returning 1 between $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{IoE\text{-}1}}$ and $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{IoE\text{-}0}}$ is negligible.*

```
Experiment: $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{IoE}\text{-}b}(1^\lambda)$
$\mathsf{ik} \leftarrow_\$ \mathsf{Gen}(1^\lambda)$, $(\mathsf{pk}, \mathsf{sk}) \leftarrow_\$ \mathsf{KGen}(\mathsf{ik})$.
$(m, \mathsf{st}) \leftarrow_\$ \mathcal{A}_1(\mathsf{pk})$.
$r \leftarrow_\$ \mathcal{R}_{\mathsf{E}}, \mathsf{ct} := \mathsf{Enc}(\mathsf{pk}, m; r)$, $r^* \leftarrow_\$ \mathsf{Fake}(\mathsf{ct}, \mathsf{pk}, m)$.
For $b = 0 : D_0 := (\mathsf{ct}, r)$.
For $b = 1 : D_1 := (\mathsf{ct}, r^*)$.
$b' \leftarrow_\$ \mathcal{A}_2(D_b, \mathsf{st})$.
Return $b'$.
```

Theorem 7. *Suppose that a PKE scheme $\mathcal{E}$ is explanation-indistinguishable and CPA-secure, then it is also dual-deniable.*

Proof. The proof can be seen as an extension of the special case for message-deniability discussed in [35]. We proceed through several hybrid games.

Game 0. This is the opening encryption case $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{DD\text{-}0}}$, and $\mathcal{A}_2$ is given $D_{\mathsf{G}_0} = (r, \mathsf{ct}_0)$, where $r \leftarrow_\$ \mathcal{R}_{\mathsf{E}}$ and $\mathsf{ct}_0 := \mathsf{Enc}(\mathsf{pk}^*, m^*; r)$.

Game 1. This game returns a fake randomness $r^* \leftarrow_\$ \mathsf{Fake}(\mathsf{ct}_0, \mathsf{pk}^*, m^*)$, now the distribution from $\mathcal{A}_2$'s view is $D_{\mathsf{G}_1} = (r^*, \mathsf{ct}_0)$. Since $\mathcal{E}$ is *explanation-indistinguishable*, we have $\left|\mathbb{P}[b'_{\mathsf{G}_1} = 1] - \mathbb{P}[b'_{\mathsf{G}_0} = 1]\right| \leq \mathbf{Adv}_{\mathcal{E}}^{\mathsf{IoE}}$.

Game 2. This game moves to encrypt under pk, the distribution from $\mathcal{A}$'s view becomes $D_{\mathsf{G}_2} = (r^*, \mathsf{ct}_2)$, where $\mathsf{ct}_2 := \mathsf{Enc}(\mathsf{pk}, m^*; r)$ and $r^* \leftarrow_\$ \mathsf{Fake}(\mathsf{ct}_2, \mathsf{pk}^*, m^*)$. If $\mathcal{A}_2$ can distinguish between *Game* 1 and 2, we can build a PPT algorithm $\mathcal{B}$ to break IK-CPA of $\mathcal{E}$. Such reduction essentially comes from the fact that Fake under public explanation does not need the original randomness.

Given $(\mathsf{pk}_0, \mathsf{pk}_1)$, $\mathcal{B}$ forwards them to $\mathcal{A}$ and receives back a message m^*. Next, $\mathcal{B}$ submits m^* to the challenger, who flips a random $b \in \{0, 1\}$ and returns $\mathsf{ct}^* \leftarrow_\$ \mathsf{Enc}(\mathsf{pk}_b, m^*)$ to $\mathcal{B}$. Then $\mathcal{B}$ samples itself $r^* \leftarrow_\$ \mathsf{Fake}(\mathsf{ct}, \mathsf{pk}_0, m^*)$, and sends (r^*, ct^*) to $\mathcal{A}$. Finally, $\mathcal{B}$ output b' returned from $\mathcal{A}$ as the guess of b. Note that if $b = 0$, $\mathcal{A}$ is in *Game* 1; otherwise, $\mathcal{A}$ is in *Game* 2. Hence, $\mathcal{B}$ is a valid algorithm against IK-CPA of $\mathcal{E}$, and so we have $\left|\mathbb{P}[b'_{\mathsf{G}_2} = 1] - \mathbb{P}[b'_{\mathsf{G}_1} = 1]\right| \leq \mathbf{Adv}_{\mathcal{E}}^{\mathsf{IK\text{-}CPA}}$.

Game 3. This is the fake case $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{DD\text{-}1}}$, $\mathcal{A}$'s view becomes $D_{\mathsf{G}_2} = (r^*, \mathsf{ct}_3)$, where $\mathsf{ct}_3 := \mathsf{Enc}(\mathsf{pk}, m; r)$ and $r^* \leftarrow_\$ \mathsf{Fake}(\mathsf{ct}_3, \mathsf{pk}^*, m^*)$. By a similar argument, we have that any adversary being able to distinguish between *Game* 3 and 2, can also break IM-CPA of $\mathcal{E}$, and so $\left|\mathbb{P}[b'_{\mathsf{G}_3} = 1] - \mathbb{P}[b'_{\mathsf{G}_2} = 1]\right| \leq \mathbf{Adv}_{\mathcal{E}}^{\mathsf{IM\text{-}CPA}}$.

Combining all the analysis, we arrive at the final result that $\left|\mathbb{P}[b'=1|\mathbf{Exp}_{\mathcal{A}}^{\mathsf{DD\text{-}1}}]-\mathbb{P}[b'=1|\mathbf{Exp}_{\mathcal{A}}^{\mathsf{DD\text{-}0}}]\right| \leq \mathbf{Adv}_{\mathcal{E}}^{\mathsf{IoE}} + \mathbf{Adv}_{\mathcal{E}}^{\mathsf{IK\text{-}CPA}} + \mathbf{Adv}_{\mathcal{E}}^{\mathsf{IM\text{-}CPA}}$, so the theorem holds. □

6 Achieving CCA Security

When plugged into a protocol running in integrated environment, the underlying PKE module is always required to achieve CCA security. Thus, in this section, we move to build public key encryption being both *dual-deniable* and *CCA-secure*.

6.1 Warm-up: Some Failed Attempts

We first consider our two constructions presented above. One may expect that CCA security naturally follows from that of the used PKE. However, it is wrong in that the overall ciphertext ct of both schemes consists of multiple sub-ciphertexts $\{\boldsymbol{c}_i\}_i$. Then $\mathcal{A}$ can query the decryption oracle with a special $\mathsf{ct}_{\mathcal{A}}$ generated by replacing some $\boldsymbol{c}_i$ of the challenge ct, so to obtain some distinguishable knowledge. E.g., for scheme $w\mathcal{DE}$, $\mathcal{A}$ against IM-CPA of DEnc can query $\mathcal{D}_{\mathsf{sk}}$ with $(\boldsymbol{c}_1, \boldsymbol{c}_2^*, \sigma)$ where $\boldsymbol{c}_2^*$ is obliviously sampled, and the decryption result is just the questioned $\boldsymbol{m}_b$; also, $\mathcal{A}$ against IK-CPA of DEnc can query $\mathcal{D}_{\mathsf{sk}_0}$ with $(\boldsymbol{c}_1, \boldsymbol{c}_2^*, \sigma)$, check if m is returned, and then learn the questioned pk_b. Similar attacks can be applied to $w\mathcal{DE}$ w.r.t. algorithm Enc and scheme $\mathcal{DE}$.

Hence, we turn to capture CCA security by starting from dual-deniability. First we examine the well-studied paradigms for CCA security, seeing if they can confer deniability: the Fujisaki-Okamoto [24] or Naor-Yung [32] conversions take auxiliary operations on the plaintext (RO-based XOR or zero-knowledge proofs). As we currently do not know how to *deny* the input of a hash value or zero-knowledge proof, any false claim on the contents of the ciphertext will cause a mismatch with such elements. Fortunately, the IBE-based frameworks [7, 12] only introduce an authentication code of the ciphertext without any further check on the used message or key, thus bringing hope for denying these internal contents. In what follows, we show how to extend the BK transform [7], so to derive CCA-secure and dual-deniable PKE from dual-deniable IBE.

6.2 TA-related Dual-deniability for IBE

Canetti, Halevi, and Katz [12] first show how to derive IM-CCA PKE from any IM-CPA IBE. Boneh and Katz [7] further improve the efficiency by use of MAC and commitment instead of one-time signature. An IBE has syntax $\langle\mathsf{KGen}, \mathsf{KDer}, \mathsf{Enc}, \mathsf{Dec}\rangle$, where trusted authority (TA) runs $\mathsf{KGen}(1^\lambda)$ to output a master key pair $(\mathsf{mpk}, \mathsf{msk})$, and $\mathsf{KDer}(\mathsf{msk}, id)$ to output a user secret key sk_{id}.

Recall that both the CHK and BK fashion take mpk as the public key of a PKE. Thus, to argue key-deniability, we have to consider IBE under multi-TA setting [34], where an additional algorithm $\mathsf{Gen}(1^\lambda)$ outputs the initial key ik

shared by all the TAs. Analogous to Def. 10, we formalize dual (message and master key) deniability for IBE, whose fake algorithm performs as follows:

- $\mathsf{Fake}(\mathsf{mpk}, id, m, r, \mathsf{mpk}^*, m^*)$: On input a master public key mpk, identity id, message m, randomness r of the original encryption, and a fake master public key mpk^* and message m^*, return fake randomness r^*.

Definition 14 (TA-related Dual-Deniability of IBE). *A multi-TA IBE satisfies TA-related dual-deniability if for all PPT adversary $\mathcal{A} := (\mathcal{A}_1, \mathcal{A}_2)$, the absolute difference of probability of returning 1 between experiment $\mathbf{Exp}^{\mathsf{tDD\text{-}1}}_{\mathcal{A}}$ and $\mathbf{Exp}^{\mathsf{tDD\text{-}0}}_{\mathcal{A}}$ is negligible.*

```
Experiment: $\mathbf{Exp}^{\mathsf{tDD\text{-}b}}_{\mathcal{A}}(\lambda)$

$\mathsf{ik} \leftarrow_{\$} \mathsf{Gen}(1^\lambda), \mathsf{CorU} := \emptyset.$
$(\mathsf{mpk}, \mathsf{msk}), (\mathsf{mpk}^*, \mathsf{msk}^*) \leftarrow_{\$} \mathsf{KGen}(\mathsf{ik}).$
$(m, m^*, id, \mathsf{st}) \leftarrow_{\$} \mathcal{A}_1^{\mathcal{O}_U}(\mathsf{mpk}, \mathsf{mpk}^*).$
$r \leftarrow_{\$} \mathcal{R}_{\mathsf{E}}, r^* \leftarrow_{\$} \mathsf{Fake}(\mathsf{mpk}, id, m, r, \mathsf{mpk}^*, m^*).$
For $b = 0$, $D_0 = (r, \mathsf{Enc}(\mathsf{mpk}^*, id, m^*; r)).$
For $b = 1$, $D_1 = (r^*, \mathsf{Enc}(\mathsf{mpk}, id, m; r)).$
$b' \leftarrow_{\$} \mathcal{A}_2^{\mathcal{O}_U}(D_b, \mathsf{st}).$
If $id \notin \mathsf{CorU}$, return $b'$; Else, return $0$.
```

```
Oracle $\mathcal{O}_U(i, id)$
  $\mathsf{CorU} := \mathsf{CorU} \cup \{id\}.$
  If $i = 0, \mathsf{sk}_{id} \leftarrow_{\$} \mathsf{KDer}(\mathsf{msk}^*, id).$
  Else, $\mathsf{sk}_{id} \leftarrow_{\$} \mathsf{KDer}(\mathsf{msk}, id).$
  Return $\mathsf{sk}_{id}.$
```

Like [12], our conversion actually requires a weaker notion termed *selective* TA-related dual-deniability (s-TA-DD), where $\mathcal{A}$ specifies id at the beginning of the game. Such scheme can be built by applying the identity-based variant of the generic approach in §.3, i.e., by use of an IBE being both statically TA-anonymous [34] and ciphertext-simulatable, e.g., the multi-TA version of Gentry's IBE scheme [25], whose TA-related anonymity has been proved in [34] and ciphertext-simulatability can be achieved by use of simulatable groups [21].

6.3 Conversion of CCA Security with Dual-deniability

Now we are ready to present the enhanced BK framework, transforming any s-TA-DD IBE into a CCA-secure DDPKE. Let $\mathcal{I} = \langle \mathsf{Gen}, \mathsf{KGen}, \mathsf{KDer}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Fake} \rangle$ be a multi-TA IBE scheme with message space $\{0,1\}^{\ell+\beta}$ and encryption randomness space $\mathcal{R}_{\mathsf{E}}$, $\langle \mathsf{Mac}, \mathsf{Vry} \rangle$ be a deterministic MAC scheme, and $\langle \mathsf{CGen}, \mathsf{Samp}, \mathsf{Open} \rangle$ be a weak commitment scheme. Our generic construction $\mathcal{DE}^{\mathsf{CCA}}$ for message space $\mathcal{M} := \{0,1\}^{\ell}$ is described as follows.

- $\mathsf{Gen}(1^\lambda)$: $\mathsf{ik} \leftarrow_{\$} \mathcal{I}.\mathsf{Gen}(1^\lambda), \mathsf{par} \leftarrow_{\$} \mathsf{CGen}(1^\lambda)$, return $\mathsf{ik} := (\mathsf{ik}, \mathsf{par})$.
- $\mathsf{KGen}(\mathsf{ik})$: $(\mathsf{mpk}, \mathsf{msk}) \leftarrow_{\$} \mathcal{I}.\mathsf{KGen}(\mathsf{ik})$, return $(\mathsf{pk}, \mathsf{sk}) := (\mathsf{mpk}, \mathsf{msk})$.
- $\mathsf{Enc}(\mathsf{pk}, m; \mathfrak{R})$: Perform the following steps:
  1. Sample a commitment triple $(\mathsf{k}, com, dec) \leftarrow_{\$} \mathsf{Samp}(\mathsf{par})$ and $r \leftarrow_{\$} \mathcal{R}_{\mathsf{E}}$, the used randomness is $\mathfrak{R} := (com, dec, r)$.
  2. Produce $c := \mathcal{I}.\mathsf{Enc}(\mathsf{mpk}, com, m||dec; r)$ under identity com.
  3. Authenticate c under k to produce $tag := \mathsf{Mac}(\mathsf{k}, c)$.

4. Return $\mathsf{ct} := (com, c, tag)$.

- $\mathsf{Dec}(\mathsf{sk}, \mathsf{ct})$: Perform the following steps:
  1. Derive the secret key $\mathsf{sk}_{com} \leftarrow_{\$} \mathcal{I}.\mathsf{KDer}(\mathsf{msk}, com)$ of identity com.
  2. Decrypt $m||dec := \mathcal{I}.\mathsf{Dec}(\mathsf{sk}_{com}, c)$.
  3. Open com to obatin $\mathsf{k} := \mathsf{Open}(\mathsf{par}, com, dec)$.
  4. If $\mathsf{Vry}(\mathsf{k}, c, tag) = 1$, return m; otherwise, return $\perp$.
- $\mathsf{Fake}(\mathsf{pk}, m, \mathfrak{R}, \mathsf{pk}^*, m^*)$: Sample a fake IBE encryption randomness $r^* \leftarrow_{\$} \mathcal{I}.\mathsf{Fake}(\mathsf{mpk}, com, m||dec, r, \mathsf{mpk}^*, m^*||dec)$, and return $\mathfrak{R}^* := (com, dec, r^*)$.

Correctness of $\mathcal{DE}^{\mathsf{CCA}}$ trivially follows from that of $\mathcal{I}$, MAC, and commitment scheme. Besides, we have the following theorem regarding its security.

Theorem 8. *Suppose that the MAC is one-time strongly unforgeable, the commitment scheme is computationally binding and hiding, and $\mathcal{I}$ is s-TA-DD, then $\mathcal{DE}^{\mathsf{CCA}}$ is dual-deniable and CCA-secure.*

Proof. Intuitionally, dual-deniability follows from two facts: 1) $\mathfrak{R}$ and $\mathfrak{R}^*$ only differ in the IBE randomness r and r^*; 2) we can authenticate a valid ciphertext c of $\mathcal{I}$ without knowing the plaintext within c to obtain a valid ciphertext ct of $\mathcal{DE}^{\mathsf{CCA}}$. Intuition for CCA security is very similar to that of [7], the main idea is to show decryption queries (com, c, tag) bring no advantage for $\mathcal{A}$: if $com \neq com^*$, we can just deliver the secret key sk_{com} to $\mathcal{A}$; otherwise, the query will always be rejected thanks to the security of commitment and the unforgeability of MAC. Below we elaborate the two separate reductions.

Dual-deniability. Given any PPT adversary $\mathcal{A}$ attacking dual-deniability of $\mathcal{DE}^{\mathsf{CCA}}$, we construct a PPT algorithm $\mathcal{B}$ against *s-TA-DD* of $\mathcal{I}$ as follows.

$\mathcal{B}$ first samples $\mathsf{par} \leftarrow_{\$} \mathsf{CGen}(1^\lambda)$ and $(\mathsf{k}^*, com^*, dec^*) \leftarrow_{\$} \mathsf{Samp}(\mathsf{par})$, then declares $id := com^*$. Next, the challenger of $\mathcal{I}$ samples $\mathsf{ik} \leftarrow_{\$} \mathcal{I}.\mathsf{Gen}(1^\lambda)$, $(\mathsf{mpk}, \mathsf{msk}) \leftarrow_{\$} \mathcal{I}.\mathsf{KGen}(\mathsf{ik})$, $(\mathsf{mpk}^*, \mathsf{msk}^*) \leftarrow_{\$} \mathcal{I}.\mathsf{KGen}(\mathsf{ik})$ and gives $(\mathsf{mpk}, \mathsf{mpk}^*)$ to $\mathcal{B}$. Now $\mathcal{B}$ sets $\mathsf{pk} := \mathsf{mpk}, \mathsf{pk}^* := \mathsf{mpk}^*$ and forwards them as well as par to $\mathcal{A}$.

At this stage, $\mathcal{A}$ outputs two messages (m, m^*). Then $\mathcal{B}$ submits the pair $(m||dec^*, m^*||dec^*)$ to the challenger and gets back (r_b, c_b), then it computes $tag^* := \mathsf{Mac}(\mathsf{k}^*, c_b)$ and sends $\big((com^*, dec^*, r_b), (com^*, c_b, tag^*) \big)$ to $\mathcal{A}$. Finally, $\mathcal{B}$ outputs the same guess b' from $\mathcal{A}$. In this way, $\mathcal{B}$ provides a perfect simulation of $\mathbf{Exp}^{\mathsf{DD}\text{-}b}$ for $\mathcal{A}$, and so it will win the s-TA-DD game with the same probability of $\mathcal{A}$ in the dual-deniable game.

CCA security. We first prove IK-CCA of $\mathcal{DE}^{\mathsf{CCA}}$ using techniques in [7], proof of IM-CCA is similar. Below we use an IK-CCA adversary $\mathcal{A}$ to build another $\mathcal{B}$ against s-TA-DD of $\mathcal{I}$. Similarly, $\mathcal{B}$ first samples $\mathsf{par} \leftarrow_{\$} \mathsf{CGen}(1^\lambda)$ and $(\mathsf{k}^*, com^*, dec^*) \leftarrow_{\$} \mathsf{Samp}(\mathsf{par})$, then declares $id := com^*$. Next, the challenger of $\mathcal{I}$ samples $\mathsf{ik} \leftarrow_{\$} \mathcal{I}.\mathsf{Gen}(1^\lambda)$, $(\mathsf{mpk}, \mathsf{msk}) \leftarrow_{\$} \mathcal{I}.\mathsf{KGen}(\mathsf{ik})$, $(\mathsf{mpk}^*, \mathsf{msk}^*) \leftarrow_{\$} \mathcal{I}.\mathsf{KGen}(\mathsf{ik})$ and gives $(\mathsf{mpk}, \mathsf{mpk}^*)$ to $\mathcal{B}$. Now $\mathcal{B}$ sets $\mathsf{pk}_0 := \mathsf{mpk}, \mathsf{pk}_1 := \mathsf{mpk}^*$ and forwards them as well as par to $\mathcal{A}$. Then it interacts with $\mathcal{A}$ as follows:

- **Query.** $\mathcal{A}$ can make arbitrary query of the form $\big((com, c, tag), d \in \{0, 1\}\big)$ to the decryption oracle of msk_d. To simulate an answer, $\mathcal{B}$ asks the oracle $\mathcal{O}_U$

on (d, com) to obtain $\mathsf{sk}_{d,com}$. It then computes $m||dec := \mathcal{I}.\mathsf{Dec}(\mathsf{sk}_{d,com}, c)$, followed by $\mathsf{k} := \mathsf{Open}(\mathsf{par}, com, dec)$. Further if $\mathsf{Vry}(\mathsf{k}, c, tag) = 1$, $\mathcal{B}$ returns m to $\mathcal{A}$; otherwise, it returns $\bot$.

- **Challenge.** After polynomial queries, $\mathcal{A}$ outputs a message m. Then $\mathcal{B}$ submits $(m||dec^*, m||dec^*)$ and receives in return (r_b, c_b). Finally, $\mathcal{B}$ computes $tag^* := \mathsf{Mac}(\mathsf{k}^*, c_b)$ and forwards $\mathsf{ct}^* := (com^*, c_b, tag^*)$ to $\mathcal{A}$.
- **Adaptive Query.** $\mathcal{A}$ may continue to make decryption queries, but not on the challenge ciphertext itself. $\mathcal{B}$ answers as before, except for queries of the form $(com^*, \cdot, \cdot, \cdot)$, $\mathcal{B}$ just returns $\bot$.
- **Guess.** $\mathcal{A}$ finally outputs a guess b', $\mathcal{B}$ outputs the same guess b'.

We argue that the simulation $\mathcal{B}$ provides for $\mathcal{A}$ makes only one difference within a real execution of $\mathcal{A}$. That is, $\mathcal{A}$ queries (com^*, c, tag, d) and the response of the real decryption oracle is not $\bot$. Note that if such c is decrypted as some $m||dec$ and opening com^* using dec does not result in k^*, the challenger could use $\mathcal{A}$ to break the computationally *binding* property of the commitment scheme. The remaining case is that the retrieved k^* holds that $\mathsf{Vry}(\mathsf{k}^*, c, tag) = 1$. We term this event Forge and proceed to show it only happens with negligible probability. In fact, this case is very similar to that in the original proof of BK transform [7].

Claim 3. $\mathbb{P}[\mathsf{Forge}] \leq \mathbf{Adv}^{\mathsf{Hiding}}_{\mathsf{COM}} + \mathbf{Adv}^{\mathsf{OT\text{-}Unf}}_{\mathsf{MAC}}$.

Proof. Let Game 0 denote the real experiment where $\mathcal{A}$ interacts with the real decryption oracles. Consider the following hybrid games.

Game 1. The challenger uses (pk_b, com^*) to return an encryption c^* of $m||0^\beta$ instead of $m||dec^*$. Due to the selective *semantic security* of $\mathcal{I}$ (recall that for PKE dual-deniability implies semantic security, it is easy to see that the implication also holds for IBE), *Game* 1 and 0 are computationally indistinguishable from $\mathcal{A}$'s view.

Game 2. This game switches to generate $tag^* := \mathsf{Mac}(\mathsf{k}', c^*)$ under another fresh key $(\mathsf{k}', \cdot, \cdot) \leftarrow_\$ \mathsf{Samp}(\mathsf{par})$ instead of k^*, and also check decryption query $(com^*, \cdot, \cdot)$ with k'.

Based on the computationally *hiding* property of the commitment scheme, we argue that this change makes no difference to $\mathcal{A}$. Consider an adversary $\mathcal{F}$ against $\langle\mathsf{CGen}, \mathsf{Samp}, \mathsf{Open}\rangle$. Given $(\mathsf{par}, com, \mathsf{k}_b)$, $\mathcal{F}$ generates ik and $\{(\mathsf{mpk}_i, \mathsf{msk}_i)\}_{i\in\{0,1\}}$ under $\mathcal{I}$ on its own. Then it runs $\mathcal{A}$ as the real IK-CCA game does, except that for the challenge ciphertext, $\mathcal{F}$ sets $com^* := com, c^* \leftarrow_\$ \mathcal{I}.\mathsf{Enc}(\mathsf{mpk}_{b_\mathcal{F}}, com, m||0^\beta)$ where $b_\mathcal{F} \leftarrow_\$ \{0,1\}$ is sampled by $\mathcal{F}$ itself, and $tag^* := \mathsf{Mac}(\mathsf{k}_b, c^*)$. Now, if $b = 1$, $\mathcal{A}$ is in *Game* 1, otherwise $\mathcal{A}$ is in *Game* 2. So $\mathcal{A}$'s advantage in distinguishing between *Game* 1 and 2 is exactly $\mathcal{F}$'s advantage in breaking the hiding property of the concerned commitment scheme.

Finally, we have that in *Game* 2, both com^* and c_b are independent of tag^*, which is now an authentication code under k', and so event Forge means a successful attack to the one-time strong unforgeability of the underlying MAC scheme w.r.t. k'.

□

Acknowledgments. This work is supported by the National Key R&D Program of China under Grant (2022YFB2701500) and the National Natural Science Foundation of China (No. 62272491) and Guangdong Provincial Key Laboratory of Information Security Technology (No. 2023B1212060026).

References

1. Abdalla, M., Bellare, M., Neven, G.: Robust encryption. In: Micciancio, D. (ed.) TCC 2010. LNCS, vol. 5978, pp. 480–497. Springer (2010)
2. Agrawal, S., Goldwasser, S., Mossel, S.: Deniable fully homomorphic encryption from learning with errors. In: Malkin, T., Peikert, C. (eds.) CRYPTO 2021. LNCS, vol. 12826, pp. 641–670. Springer (2021)
3. An, Z., Tian, H., Chen, C., Zhang, F.: Deniable cryptosystems: Simpler constructions and achieving leakage resilience. In: Tsudik, G., Conti, M., Liang, K., Smaragdakis, G. (eds.) ESORICS 2023. LNCS, vol. 14344, pp. 24–44. Springer (2023)
4. Bellare, M., Boldyreva, A., Desai, A., Pointcheval, D.: Key-privacy in public-key encryption. In: Boyd, C. (ed.) ASIACRYPT 2001. LNCS, vol. 2248, pp. 566–582. Springer (2001)
5. Ben-Sasson, E., Chiesa, A., Garman, C., Green, M., Miers, I., Tromer, E., Virza, M.: Zerocash: Decentralized anonymous payments from bitcoin. In: S & P 2014. pp. 459–474. IEEE Computer Society (2014)
6. Bendlin, R., Nielsen, J.B., Nordholt, P.S., Orlandi, C.: Lower and upper bounds for deniable public-key encryption. In: Lee, D.H., Wang, X. (eds.) ASIACRYPT 2011. LNCS, vol. 7073, pp. 125–142. Springer (2011)
7. Boneh, D., Katz, J.: Improved efficiency for cca-secure cryptosystems built using identity-based encryption. In: Menezes, A. (ed.) CT-RSA 2005. LNCS, vol. 3376, pp. 87–103. Springer (2005)
8. Bos, J.W., Ducas, L., Kiltz, E., Lepoint, T., Lyubashevsky, V., Schanck, J.M., Schwabe, P., Seiler, G., Stehlé, D.: CRYSTALS - kyber: A cca-secure module-lattice-based KEM. In: EuroS&P. pp. 353–367. IEEE (2018)
9. Camenisch, J., Lysyanskaya, A.: An efficient system for non-transferable anonymous credentials with optional anonymity revocation. In: Pfitzmann, B. (ed.) EUROCRYPT 2001. LNCS, vol. 2045, pp. 93–118. Springer (2001)
10. Canetti, R., Dwork, C., Naor, M., Ostrovsky, R.: Deniable encryption. In: Jr., B.S.K. (ed.) CRYPTO 1997. vol. 1294, pp. 90–104. Springer (1997)
11. Canetti, R., Gennaro, R.: Incoercible multiparty computation (extended abstract). In: FOCS 1996. pp. 504–513. IEEE Computer Society (1996)
12. Canetti, R., Halevi, S., Katz, J.: Chosen-ciphertext security from identity-based encryption. In: Cachin, C., Camenisch, J. (eds.) EUROCRYPT 2004. LNCS, vol. 3027, pp. 207–222. Springer (2004)
13. Canetti, R., Park, S., Poburinnaya, O.: Fully deniable interactive encryption. In: Micciancio, D., Ristenpart, T. (eds.) CRYPTO 2020. LNCS, vol. 12170, pp. 807–835. Springer (2020)
14. Caro, A.D., Iovino, V., O'Neill, A.: Deniable functional encryption. In: Cheng, C., Chung, K., Persiano, G., Yang, B. (eds.) PKC 2016. LNCS, vol. 9614, pp. 196–222. Springer (2016)
15. Chi, P., Lei, C.: Audit-free cloud storage via deniable attribute-based encryption. IEEE Trans. Cloud Comput. **6**(2), 414–427 (2018)

16. Chillotti, I., Gama, N., Georgieva, M., Izabachène, M.: A homomorphic LWE based e-voting scheme. In: Takagi, T. (ed.) PQCrypto 2016. LNCS, vol. 9606, pp. 245–265. Springer (2016)
17. Choi, S.G., Dachman-Soled, D., Malkin, T., Wee, H.: Improved non-committing encryption with applications to adaptively secure protocols. In: Matsui, M. (ed.) ASIACRYPT 2009. LNCS, vol. 5912, pp. 287–302. Springer (2009)
18. Coladangelo, A., Goldwasser, S., Vazirani, U.V.: Deniable encryption in a quantum world. In: Leonardi, S., Gupta, A. (eds.) STOC 2022. pp. 1378–1391. ACM (2022)
19. Dachman-Soled, D.: On minimal assumptions for sender-deniable public key encryption. In: Krawczyk, H. (ed.) PKC 2014. LNCS, vol. 8383, pp. 574–591. Springer (2014)
20. Damgård, I., Nielsen, J.B.: Improved non-committing encryption schemes based on a general complexity assumption. In: Bellare, M. (ed.) CRYPTO 2000. LNCS, vol. 1880, pp. 432–450. Springer (2000)
21. Dent, A.W.: The cramer-shoup encryption scheme is plaintext aware in the standard model. In: Vaudenay, S. (ed.) EUROCRYPT 2006. LNCS, vol. 4004, pp. 289–307. Springer (2006)
22. Dowling, B., Hauck, E., Riepel, D., Rösler, P.: Strongly anonymous ratcheted key exchange. In: Agrawal, S., Lin, D. (eds.) ASIACRYPT 2022. LNCS, vol. 13793, pp. 119–150. Springer (2022)
23. Farshim, P., Libert, B., Paterson, K.G., Quaglia, E.A.: Robust encryption, revisited. In: Kurosawa, K., Hanaoka, G. (eds.) PKC 2013. LNCS, vol. 7778, pp. 352–368. Springer (2013)
24. Fujisaki, E., Okamoto, T.: Secure integration of asymmetric and symmetric encryption schemes. In: Wiener, M.J. (ed.) CRYPTO 1999. LNCS, vol. 1666, pp. 537–554. Springer (1999)
25. Gentry, C.: Practical identity-based encryption without random oracles. In: Vaudenay, S. (ed.) EUROCRYPT 2006. LNCS, vol. 4004, pp. 445–464. Springer (2006)
26. Géraud, R., Naccache, D., Rosie, R.: Robust encryption, extended. In: Matsui, M. (ed.) CT-RSA 2019. LNCS, vol. 11405, pp. 149–168. Springer (2019)
27. Grubbs, P., Maram, V., Paterson, K.G.: Anonymous, robust post-quantum public key encryption. In: Dunkelman, O., Dziembowski, S. (eds.) EUROCRYPT 2022. LNCS, vol. 13277, pp. 402–432. Springer (2022)
28. Halevi, S.: A sufficient condition for key-privacy. IACR Cryptol. ePrint Arch. pp. 1–3 (2005)
29. Kiayias, A., Tsiounis, Y., Yung, M.: Group encryption. In: Kurosawa, K. (ed.) ASIACRYPT 2007. LNCS, vol. 4833, pp. 181–199. Springer (2007)
30. Matsuda, T., Hanaoka, G.: Trading plaintext-awareness for simulatability to achieve chosen ciphertext security. In: Cheng, C., Chung, K., Persiano, G., Yang, B. (eds.) PKC 2016. LNCS, vol. 9614, pp. 3–34. Springer (2016)
31. Mohassel, P.: A closer look at anonymity and robustness in encryption schemes. In: Abe, M. (ed.) ASIACRYPT 2010. LNCS, vol. 6477, pp. 501–518. Springer (2010)
32. Naor, M., Yung, M.: Public-key cryptosystems provably secure against chosen ciphertext attacks. In: Ortiz, H. (ed.) STOC 1990. pp. 427–437. ACM (1990)
33. O'Neill, A., Peikert, C., Waters, B.: Bi-deniable public-key encryption. In: Rogaway, P. (ed.) CRYPTO 2011. LNCS, vol. 6841, pp. 525–542. Springer (2011)
34. Paterson, K.G., Srinivasan, S.: Security and anonymity of identity-based encryption with multiple trusted authorities. In: Galbraith, S.D., Paterson, K.G. (eds.) Pairing 2008. LNCS, vol. 5209, pp. 354–375. Springer (2008)
35. Sahai, A., Waters, B.: How to use indistinguishability obfuscation: deniable encryption, and more. In: Shmoys, D.B. (ed.) STOC 2014. pp. 475–484. ACM (2014)

36. Sako, K.: An auction protocol which hides bids of losers. In: Imai, H., Zheng, Y. (eds.) PKC 2000. LNCS, vol. 1751, pp. 422–432. Springer (2000)

A More Discussions on Weak/Plan-Ahead DDPKE

Weak Dual-Deniability. It has been proved in [6] that weak mode is necessary if negligible detection probability of message-deniability is desired for both sides, i.e., both *sender* and *receiver* deniability. We shall show that such mode is also necessary if negligible detection probability is required w.r.t. both sender-key-deniability and receiver-message-deniability, if key-simulatability (Def. 6) is also desired, e.g., for non-committing PKE [10, 20]. This comes from the following observations:

- First, [6] essentially shows that a non-interactive standard DDPKE for bits with polynomial secret key size cannot be both sender-message-deniable and receiver-deniable.
- Second, as described in Prop. 4, we can obtain standard sender-DDPKE (and so MDPKE) for bits from any standard key-simulatable sender-KDPKE being also FROB, for which the secret key is unchanged. Actually, the requirement of robustness can be relaxed to be weak robustness (Def. 16), a simplified notion introduced in [1]. Besides, although not formally attested, it is trivial to show that if the underlying scheme is receiver-message-deniable, the transformed one is also receiver-message-deniable.
- There exist some simple and generic constructions of weakly robust PKE from plain PKE [1, 31], for which the secret key is unchanged. Moreover, these conversions bring no impact on the degree of sender-key-deniability, receiver-message-deniability, and key-simulatability of the underlying PKE.

Combining the above facts, we arrive at the following proposition:

Proposition 5. *Suppose a PKE is key-simulatable and sender-key-deniable with the upper bound on the secret key size being γ, it can only be $\frac{1}{\omega(\gamma)}$-receiver-deniable.*

On the other hand, similar to the arguments in [33], when applying weak DDPKE schemes, we can make the implementation default to the normal algorithms, which do serve as an explicit system, and take the deniable algorithms as the “backup” procedures. Under coercion, the user can simply assert that they did not run those backup algorithms, though the coercer could have reason to believe — but not any legal evidence — that the deniable algorithms were actually invoked. This can be ensured by deploying these implicit branches on some undetectable devices, e.g., cloud-based libraries or distributed hosts.

Plan-Ahead Key-Deniability. We briefly show that plan-ahead dual/key-deniability is already enough in some anonymous applications. One example is when applying key deniable PKE in group encryption [29], where the sender can prepare an anonymous ciphertext designated to a member of some PKI group and convince an outsider that the ciphertext really “belongs to” that group. In deniable group encryption (a possible future work), the sender can just include another valid member along with the real one, and open the ciphertext to that “scapegoat”. Since all the communications within a group encryption system are anonymous, the coercer cannot get any information about the target set of the designated receivers, and thus the *plan-ahead* confession from the sender remains convincing in the view of the “blind” coercer. Another software application is for IP address hiding: an exposed host can encode the data packages via a plan-ahead KDPKE, such that when faced with invasion attacks, it can decode these packages towards a dummy destination and so protect the private routing tables.

Overall, plan-ahead dual-deniability is suitable for environments where the sender can easily obtain some auxiliary but valid public keys, and the coercer learns nothing of the real address book of that sender. In particular, compared with the $i\mathcal{O}$-based construction (§.5) under standard model, our plan-ahead scheme (§.4) is much more efficient in key size and allows weaker assumptions (DDH,RSA,LWE), giving hope for applying DD/KDPKE in real world after some further optimizations.

B Other Security Definitions of Plan-Ahead DDPKE

Following the syntax of Def. 11, below we formalize IK-CCA security and KROB w.r.t. plan-ahead DDPKE.

Definition 15 (IM-CCA and XROB of Plan-Ahead DDPKE). *A plan-ahead DDPKE is IM-CCA if for all PPT adversary $\mathcal{A} := (\mathcal{A}_1, \mathcal{A}_2)$, the absolute difference of probability of returning 1 between experiment $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{pIM\text{-}0}}$ and $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{pIM\text{-}1}}$ is negligible, and is KROB if for all PPT adversary $\mathcal{A}$, the probability of outputting 1 for experiment $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{pKROB}}$ is negligible,*

```
Experiment: $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{pIM}\text{-}b}(1^\lambda)$

$\mathsf{ik} \leftarrow_{\$} \mathsf{Gen}(1^\lambda)$.
$(\mathsf{pk}, \mathsf{sk}) \leftarrow_{\$} \mathsf{KGen}(\mathsf{ik})$, $\mathcal{P}_{\mathsf{ik}} := \{\mathsf{pk}\}$.
$(m_0, m_1, \mathsf{pk}^*, m^*, \mathsf{st}) \leftarrow_{\$} \mathcal{A}_1^{\mathcal{D}_{\mathsf{sk}}(\cdot)}(\mathsf{pk})$.
$\mathsf{ct} \leftarrow_{\$} \mathsf{Enc}(\mathsf{pk}, m_b, \mathsf{pk}^*, m^*)$.
$b' \leftarrow_{\$} \mathcal{A}_2^{\mathcal{D}_{\mathsf{sk}}(\neg\mathsf{ct})}(\mathsf{ct}, \mathsf{st})$.
Return $b'$.
```

```
Experiment: $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{pKROB}}(1^\lambda)$

$\mathsf{ik} \leftarrow_{\$} \mathsf{Gen}(1^\lambda)$.
$\big(\{m_i, m_i^*, \mathsf{pk}_i, \mathsf{pk}_i^*, r_i\}_{i\in\{0,1\}}\big) \leftarrow_{\$} \mathcal{A}(\mathsf{ik})$.
$\mathsf{ct}_0 := \mathsf{Enc}(\mathsf{pk}_0, m_0, \mathsf{pk}_0^*, m_0^*; r_0)$.
$\mathsf{ct}_1 := \mathsf{Enc}(\mathsf{pk}_1, m_1, \mathsf{pk}_1^*, m_1^*; r_1)$.
Return $(\mathsf{pk}_0 \neq \mathsf{pk}_1) \wedge (\mathsf{ct}_0 = \mathsf{ct}_1 \neq \perp)$.
```

C Another Look of the Construction in §.4

In the construction $\mathcal{DE}$ of DDPKE at §.4, we include in the overall ciphertext ct an OWF tag σ, mainly to enable the receiver to identify $\boldsymbol{c}_t$ from all the obliviously sampled $\boldsymbol{c}_i$. In other words, we make these oblivious elements undecryptable under sk. Naturally, we can also utilize *robustness* of PKE to achieve such task, and indeed it suffices to require a basic property – *weak robustness* (WROB):

Definition 16 (WROB). *A PKE scheme satisfies weak robustness if for all PPT adversary $\mathcal{A}$, the probability of returning 1 for experiment $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{WROB}}$ is negligible.*

```
Experiment: $\mathbf{Exp}_{\mathcal{A}}^{\mathsf{WROB}}(1^\lambda)$
$\mathsf{ik} \leftarrow_\$ \mathsf{Gen}(1^\lambda)$.
$(\mathsf{pk}_0, \mathsf{sk}_0) \leftarrow_\$ \mathsf{KGen}(\mathsf{ik}), (\mathsf{pk}_1, \mathsf{sk}_1) \leftarrow_\$ \mathsf{KGen}(\mathsf{ik})$.
$(m, b) \leftarrow_\$ \mathcal{A}(\mathsf{pk}_0, \mathsf{pk}_1)$.
$\mathsf{ct} \leftarrow_\$ \mathsf{Enc}(\mathsf{pk}_b, m), m' := \mathsf{Dec}(\mathsf{sk}_{1-b}, \mathsf{ct})$.
Return $m' \neq \perp$.
```

Note that both FROB and XROB imply WROB (Thm. 2 of [23]). Besides, the OWF-based encryption manner used in [3], i.e., encrypting $\boldsymbol{m}$ plus a random pad $\boldsymbol{r}$ as well as sending the OWF tag $\sigma := \mathcal{H}(\boldsymbol{u})$, is essentially also the transformation of WROB considered in [31]. Moreover, one can trivially verify that such extension of the original ciphertext brings no impact on *ciphertext-simulatability* of the underlying PKE. Similarly, it is easy to check that the conversion of FROB proposed in [31] also reserves *ciphertext-simulatability*. While another approach of FROB presented in [23] saves *ciphertext-simulatability* only if the used commitment scheme is also simulatable, which is true for the DLP-based commitments.

By applying the aforementioned techniques to the known ciphertext-simulatable PKE schemes (see §.2.2), we can obtain the desired building block of DDPKE. Below we give in Fig.4 the revised description of $\mathcal{DE}$, where the first two algorithms $\langle \mathsf{Gen}, \mathsf{KGen} \rangle$ are omitted as they are unchanged.

```
• $\mathsf{Enc}(\mathsf{pk}, \boldsymbol{m}, \mathsf{pk}^*, \boldsymbol{m}^*; \mathfrak{R})$
  1. Sample $\mathfrak{R}$ as follows:
     ① $t \leftarrow_\$ [n-1]$.
     ② $\forall i \in [t+2, n]$, $(\mathsf{pk}_i, \boldsymbol{m}_i) \leftarrow_\$ \mathcal{P}_{\mathsf{ik}} \times \mathcal{M}$.
     ③ $\forall i \in [t-1]$, $r_i \leftarrow_\$ \mathcal{R}_{\mathsf{O}}$.
     ④ $\forall i \in [t, n]$, $r_i \leftarrow_\$ \mathcal{R}_{\mathsf{E}}$.
    $\mathfrak{R} := \left(t, \{\mathsf{pk}_i, \boldsymbol{m}_i\}_{i \in [t+2,n]}, \{r_i\}_{i \in [n]}\right)$.
  2. $\forall i \in [n]$, produce each $\boldsymbol{c}_i$ as follows:
     ① $\boldsymbol{c}_i := \mathcal{E}.\mathsf{OEnc}(\mathsf{ik}; r_i)$ for $i < t$.
     ② $\boldsymbol{c}_t := \mathcal{E}.\mathsf{Enc}(\mathsf{pk}, \boldsymbol{m}; r_t)$.
     ③ $\boldsymbol{c}_{t+1} := \mathcal{E}.\mathsf{Enc}(\mathsf{pk}^*, \boldsymbol{m}^*; r_{t+1})$.
     ④ $\boldsymbol{c}_i := \mathcal{E}.\mathsf{Enc}(\mathsf{pk}_i, \boldsymbol{m}_i; r_i)$ for $i > t+1$.
  3. Return $\mathsf{ct} := \{\boldsymbol{c}_i\}_{i \in [n]}$.

• $\mathsf{Dec}(\mathsf{dsk}, \mathsf{dct})$
  Set $\mathsf{succ} := 0$, and do the following for $i \in [n]$:
    1. $\boldsymbol{m}' := \mathcal{E}.\mathsf{Dec}(\mathsf{sk}, \boldsymbol{c}_i)$.
    2. If $\boldsymbol{m}' \neq \perp$, set $\mathsf{succ} := 1$, stop and return $\boldsymbol{m}'$.
    3. Move to $i = i + 1$.
  If $\mathsf{succ} = 0$, return $\perp$.

• $\mathsf{Fake}(\mathsf{pk}, \boldsymbol{m}, \mathfrak{R}, \mathsf{pk}^*, \boldsymbol{m}^*)$
  1. Set $(t^*, \mathsf{pk}^{**}, \boldsymbol{m}^{**}) := (t+1, \mathsf{pk}_{t+2}, \boldsymbol{m}_{t+2})$.
  2. $r_t^* \leftarrow_\$ \mathcal{E}.\mathsf{IEnc}(\mathsf{pk}, \boldsymbol{m}, r_t)$.
  3. $\forall i \in [n] \setminus \{t\}$, $r_i^* := r_i$.
  4. $\forall i \in [t^* + 2]$, $(\mathsf{pk}_i^*, \boldsymbol{m}_i^*) := (\mathsf{pk}_i, \boldsymbol{m}_i)$.
 Return $\mathfrak{R}^* := \left(t^*, \{\mathsf{pk}_i^*, \boldsymbol{m}_i^*\}_{i \in [t^*+2,n]}, \{r_i^*\}_{i \in [n]}\right)$.
```

Fig. 4. Variant of scheme $\mathcal{DE}$ in §.4.

Claim 4. *Suppose that $\mathcal{E}$ is correct, WROB, and ciphertext-simulatable, then $\mathcal{DE}$ is correct, CPA-secure, and $\frac{1}{n-1}$-dual-deniable under plan-ahead setting.*

Proof. (Sketch). We begin with *correctness*. Based on the proof of Thm. 1, it suffices to show that all the $\{\boldsymbol{c}_i\}_{i\in[t-1]}$ are undecryptable under sk. This is equivalent to show that for random $\mathsf{ik} \leftarrow_{\$} \mathsf{Gen}(1^\lambda)$, $(\mathsf{pk}, \mathsf{sk}) \leftarrow_{\$} \mathsf{KGen}(\mathsf{ik})$, and $r \leftarrow_{\$} \mathcal{R}_{\mathsf{O}}$, it holds $\mathbb{P}[\mathsf{Dec}(\mathsf{sk}, \mathcal{E}.\mathsf{OEnc}(\mathsf{ik}; r)) \neq \perp] \leq \mathsf{negl}(\lambda)$. To this effect, we first argue that if the above inequation is false, it must instead hold $\mathbb{P}[\mathsf{Dec}(\mathsf{sk}, \mathcal{E}.\mathsf{Enc}(\mathsf{pk}', \boldsymbol{m}'; r')) \neq \perp] \geq \mathsf{negl}(\lambda)$ where $(\mathsf{pk}', \mathsf{sk}') \leftarrow_{\$} \mathsf{KGen}(\mathsf{ik})$ is another fresh key pair and $\boldsymbol{m}' \leftarrow_{\$} \mathcal{M}$, or we can directly build a PPT algorithm $\mathcal{B}$ against *ciphertext-simulatability* of $\mathcal{E}$. In other words, by *ciphertext-simulatability* of $\mathcal{E}$, if an obliviously sampled element is decryptable under a random secret key, a random encryption (under another public key) is also decryptable under the same secret key. However, this deduction further contradicts *weak robustness* of $\mathcal{E}$, which exactly disallows the successful decryption (of an honest ciphertext) under the *wrong* key. Taking together all the analysis, we have reduced *correctness* of $\mathcal{DE}$ to *correctness*, *ciphertext-simulatability*, and *WROB* of $\mathcal{E}$.

Proofs of *CPA-security* and *dual-deniability* are almost the same as the ones of Thm. 2 and 3, which are only based on *ciphertext-simulatability* of $\mathcal{E}$. □