Tree-Structured Composition of Homomorphic Encryption: How to Weaken Underlying Assumptions

Koji Nuida*†, Goichiro Hanaoka*, Takahiro Matsuda*

* National Institute of Advanced Industrial Science and Technology (AIST)
{k.nuida, hanaoka-goichiro, t-matsuda}@aist.go.jp
† Japan Science and Technology Agency (JST), PRESTO

November 20, 2014

Abstract

Cryptographic primitives based on infinite families of progressively weaker assumptions have been proposed by Hofheinz–Kiltz and by Shacham (the n-Linear assumptions) and by Escala et al. (the Matrix Diffie–Hellman assumptions). All of these assumptions are extensions of the decisional Diffie–Hellman (DDH) assumption. In contrast, in this paper, we construct (additive) homomorphic encryption (HE) schemes based on a new infinite family of assumptions extending the *decisional Composite Residuosity (DCR) assumption*. This is the first result on a primitive based on an infinite family of progressively weaker assumptions not originating from the DDH assumption. Our assumptions are indexed by *rooted trees*, and provides a completely different structure compared to the previous extensions of the DDH assumption.

Our construction of a HE scheme is generic; based on a tree structure, we recursively combine copies of building-block HE schemes associated to each leaf of the tree (e.g., the Paillier cryptosystem, for our DCR-based result mentioned above). Our construction for depth-one trees utilizes the "share-then-encrypt" multiple encryption paradigm, modified appropriately to ensure security of the resulting HE schemes. We prove several separations between the CPA security of our HE schemes based on different trees; for example, the existence of an adversary capable of breaking *all* schemes based on depth-one trees, does *not* imply an adversary against our scheme based on a depth-*two* tree (within a computational model analogous to the generic group model). Moreover, based on our results, we give an example which reveals a type of "non-monotonicity" for security of generic constructions of cryptographic schemes and their building-block primitives; if the building-block primitives for a scheme are replaced with other ones secure under *stronger* assumptions, it may happen that the resulting scheme becomes secure under a *weaker* assumption than the original.

Keywords: Homomorphic encryption, Composite Residuosity assumption, tree-shaped assumption family, generic construction

1 Introduction

In modern cryptology, cryptographic primitives based on as *weak assumptions* as possible have been intensively studied. In particular, several primitives based on *infinite* families of assumptions, which are either one-dimensional (the n-Linear assumptions [20, 32], including the decisional Diffie–Hellman (DDH) and the Decision Linear assumptions as special cases) or two-dimensional (the Matrix Diffie–Hellman assumptions [12]) extensions of the DDH assumption, were proposed. Assumptions in each family have (non-)implication relations; for example, the ability to break the n-Linear assumption with a larger n implies the ability to break all assumptions with a smaller n, but the converse does *not* hold (proven in the generic group model [33]).

Here we point out that, *all the previous work on such constructions of primitives are based on extensions of the Diffie–Hellman (DH) assumptions.* Our present work aims at constructing, possibly by a different approach, primitives based on an infinite family of assumptions which are extensions of different standard assumptions.

1.1 Our Contributions

In this paper, we develop the first construction of primitives based on an infinite family of assumptions which are extensions of standard assumptions other than the DH assumption, namely the *Composite Residuosity* (*CR*) *assumption.* More precisely, starting from an (additive) homomorphic encryption (HE) scheme Π, we construct new HE schemes by combining copies of Π in various ways. Hence, our result is a *generic construction* rather than concrete constructions as in the previous work [12, 20, 32]; the CR-based construction is derived by choosing the Paillier cryptosystem [30] as the building-block scheme Π. We also extend the construction to a more general case that a new HE scheme is obtained by combining *different* building-block HE schemes with common plaintext space.

Our construction is *recursive*, indexed by a *rooted tree*; it is completely different from the previous "line-shaped" [20, 32] and "matrix-shaped" [12] constructions. Each copy of the building-block scheme is associated to a leaf of the tree; the scheme at a (non-leaf) vertex is constructed by combining the schemes at the child vertices; and finally our proposed scheme is obtained as the scheme at the root of the tree.

Essence of our construction. As an example, we consider the case that ℓ copies of the Paillier cryptosystem is combined to obtain an HE scheme associated to the parent vertex. Our construction is based on the existing "share-then-encrypt" multiple encryption paradigm (see e.g., [10]), where the easiest ℓ-out-of-ℓ secret sharing (i.e., the secret is the sum of shares) is used in order to simplify our analysis. To encrypt $m \in \mathbb{Z}/n\mathbb{Z}$, we first divide it into random shares $s_1, \dots, s_\ell \in \mathbb{Z}/n\mathbb{Z}$, and then encrypt each s_i by the i-th copy of the Paillier cryptosystem. One may naively expect that this idea would improve the security, since to learn information on m, it would be necessary to learn information on all of $s_1, \dots, s_\ell$ by breaking the ℓ ciphertext components simultaneously.

Now we in fact need to be extra careful, since we are dealing with *homomorphic* encryption. Namely, if the base elements of the ciphertext space $(\mathbb{Z}/n^2\mathbb{Z})^\times$ involved in the public key of each copy of the Paillier cryptosystem are *equal*, then the adversary can merge the ℓ ciphertext components into a *single* ciphertext of plaintext m for the Paillier cryptosystem by using its additive homomorphic property (i.e., recovering the secret m from the shares $s_1, \dots, s_\ell$ homomorphically). Consequently, breaking the new scheme is not more difficult than breaking the Paillier cryptosystem, which is not desirable.

Therefore, we must use *different* base elements $g_1, \dots, g_\ell$ for the ℓ components, each being a part of a public key for a copy of the Paillier cryptosystem. On the other hand, the other part n of the public key for the Paillier cryptosystem must be *common* for the ℓ components. However, such a separated treatment of individual parts of a public key is not suitable to generalize to a generic (black-box) construction.

To resolve the problem, we re-interpret each base element g_i as a ciphertext of a different plaintext with a *fixed* base element g (say, $g = 1 + n \bmod n^2$), and re-interpret the encryption of s_i with base element g_i as a "rerandomized scalar multiplication" of s_i to g_i; i.e., if $g_i = g^{a_i} \cdot {r_i}^n$ is a ciphertext of $a_i \in \mathbb{Z}/n\mathbb{Z}$, then the encryption result ${g_i}^{s_i} \cdot {r'_i}^n$ with base element g_i is a (rerandomized) ciphertext of $s_i \cdot a_i$ (since ${g_i}^{s_i} \cdot {r'_i}^n = g^{s_i \cdot a_i} \cdot ({r_i}^{s_i} \cdot r'_i)^n$). Hence, the original public key (n, g_i) for the Paillier cryptosystem at each component is converted to the pair of a *common* public key (n, g) (which can be used in a black-box manner) and a *ciphertext* g_i by the common public key. This enables us to extend the construction to other building-block HE scheme, provided it is also endowed with "scalar multiplication" and "rerandomization" functionalities (the resulting HE scheme also has these additional functionalities, therefore the recursive construction is indeed possible).

(Non-)implication relations. We prove several separation relations between the underlying assumptions for the CPA security of our proposed HE schemes indexed by different trees. For the case that the schemes are constructed from a single building-block scheme, first we prove that, for the assumptions indexed by trees of depth one with $\ell \geq 1$ leaves, the assumption with smaller ℓ implies that with larger ℓ but the converse does *not* hold. It is analogous to the relations of the n-Linear assumptions. This also implies that our new assumptions are strictly weaker than the assumption for the building-block scheme, since the latter is in fact equivalent to the assumption with $\ell = 1$.

Moreover, we prove that, even if *all* the assumptions indexed by the trees of depth one are broken, it does *not* immediately imply that the assumption indexed by a tree $T^{\S}$ of depth *two* is efficiently breakable (see Example 2 in Section 5.1 for the definition of $T^{\S}$). Hence, our assumption family indeed has beyond one-dimensional degrees of freedom.

When the building-block scheme is the Paillier cryptosystem, the strength of our new "tree-shaped" assumptions are all lying strictly between the Computational Composite Residuosity (CCR) and the Decisional Composite Residuocity (DCR) assumptions. Hence, our result reveals an interesting fact that there are infinitely many assumptions, having the rich variety, between the closely related CCR and DCR assumptions.

Our computational model for non-implications. The non-implication relations for our assumptions above are proven in a new computational model, which is an analogy of the generic group model [33] with modifications made in order to deal with separations between *generic constructions* of primitives. Our computational model is a variant of the Boolean circuit model (see e.g., Section 1.2.4.1 of [14]), where we can treat *black-box* elements of the ciphertext space (as well as ordinary bits), and each circuit involves gates for *black-box* computations on ciphertexts via homomorphic functionalities (as well as ordinary gates for bit operations). We emphasize that plaintexts are expressed by bit sequences (rather than black-box elements) and *any* (efficient) operations on the bit sequences expressing plaintexts are allowed, for making the computational model reasonably powerful.

For example, when the building-block scheme is the Paillier cryptosystem, our computational model has strong enough functionality to be comparable to the generic group model on the ciphertext space $(\mathbb{Z}/n^2\mathbb{Z})^\times$ of the Paillier cryptosystem; see Remark 2 in Section 6.

Application: "Non-monotonicity" of combined security. By using our result, we construct HE schemes $\Pi_1, \dots, \Pi_4$ satisfying the following: The assumptions for Π_1 and Π_2 are strictly *stronger* (within the computational model mentioned above) than Π_3 and Π_4, respectively; but conversely, the assumption for our proposed HE scheme that combines Π_1 and Π_2 is strictly *weaker* than that combining Π_3 and Π_4. See Section 8. This suggests that the precise strength of our new assumptions may be further weaker than evaluated in this paper. It also gives an insight that, in a generic construction of a cryptographic primitive, the security is in general inherited *not monotonically* from the building blocks.

1.2 Related Work

In the previous work by Escala et al. [12] mentioned above, they proposed several primitives based on their assumptions, but did *not* propose HE schemes. On the other hand, the framework for HE schemes by Armknecht et al. [1] does not entirely cover our class of HE schemes, and their ElGamal-like HE schemes based on the n-Linear assumptions are much different from ours. This shows the independent significance of our work.

One may feel that our generic construction has a flavor similar to the "robust combiners" for several kinds of primitives (e.g., [2, 4, 9, 18, 19, 26]), where the constructed scheme is secure provided *at least one* of the building-block schemes is secure (or to the *quantitative* security amplification such as Yao's XOR lemma, cf., [15, 21, 22, 23, 24]). We emphasize that *our proposed scheme can be secure even when all of the building-block schemes are insecure*, which is also a noteworthy feature of our construction.

1.3 Organization of This Paper

In Section 2, we summarize some notations, terminology and basic definitions used in this paper. In Section 3, we define the class of HE schemes considered in this paper, and give an equivalent but simplified notion of the CPA security for these HE schemes. In Section 4, we show some instances of the HE schemes in the literature. In Section 5, we construct our proposed HE schemes and show some implication relations for the CPA security between them. We give our main non-implication relations for the CPA security in Section 7, using the computational model in Section 6. Finally, in Section 8, we present an example of the non-monotonicity of security in generic constructions of cryptographic primitives.

2 Preliminaries

In this paper, k denotes the security parameter unless otherwise specified. We say that a quantity $\varepsilon \geq 0$ is *negligible*, if $\varepsilon = k^{-\omega(1)}$; and ε is *overwhelming*, if $1 - \varepsilon$ is negligible. For probability distributions $\mathcal{D}_1$ and $\mathcal{D}_2$, we write $\mathcal{D}_1 \sim \mathcal{D}_2$ to mean that $\mathcal{D}_1$ and $\mathcal{D}_2$ are identical, while we say that $\mathcal{D}_1$ and $\mathcal{D}_2$ are *statistically close*, if their statistical distance $\frac{1}{2}\sum_x |\Pr[x \leftarrow \mathcal{D}_1] - \Pr[x \leftarrow \mathcal{D}_2]|$ is negligible. We say that a finite set X is *samplable* (respectively, *approximately samplable*), if there exists a probabilistic polynomial-time (PPT) algorithm with output distribution identical (respectively, statistically close) to the uniform distribution on X. Let an expression "$x \leftarrow_R X$" mean that an element x is chosen from a set X uniformly at random.

We recall the syntax for public key encryption schemes and their security notion discussed in this paper.

Definition 1 (Public key encryption)**.** We say that $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ is a *public key encryption* (*PKE*) *scheme*, if it consists of the following three algorithms:

- The PPT algorithm Gen outputs a pair $(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^k)$ of public key pk and secret key sk. Finite sets $\mathcal{M}$ and $\mathcal{C}$ of plaintexts and ciphertexts, respectively, are also associated to pk.
- The PPT algorithm Enc outputs a ciphertext $c \leftarrow \mathsf{Enc}(\mathsf{pk}, m)$ in $\mathcal{C}$ of plaintext $m \in \mathcal{M}$ under public key pk.
- The algorithm Dec, with a secret key sk and a ciphertext $c \in \mathcal{C}$ as inputs, outputs either an element of $\mathcal{M}$ or a "failure symbol" $\perp \not\in \mathcal{M}$.

Let $\mathcal{C}_m \subset \mathcal{C}$ be a set of valid ciphertexts of plaintext $m \in \mathcal{M}$ (under a given public key pk), which is supposed to satisfy

$$\Pr[m \leftarrow \mathsf{Dec}(\mathsf{sk}, c)] = 1 \text{ for every } c \in \mathcal{C}_m \enspace .$$

Then Π is supposed to satisfy (*perfect*) *correctness*:

$$c \in \mathcal{C}_m \text{ for any } c \leftarrow \mathsf{Enc}(\mathsf{pk}, m) \text{ with } m \in \mathcal{M} \enspace .$$

We often omit the symbols pk and sk for public and secret keys unless it causes ambiguity.

Definition 2 (CPA security)**.** We say that a PKE scheme Π is *CPA secure*, if for any PPT adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$, the advantage $\mathrm{Adv}_{\mathcal{A}}(k) := |\Pr[b = b^*] - 1/2|$ of $\mathcal{A}$ for a game defined by the following procedure is negligible:

$$\begin{aligned}&[(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^k);\;\; (m_0, m_1, \mathsf{state}) \leftarrow \mathcal{A}_1(\mathsf{pk});\\ &\qquad b^* \leftarrow_R \{0,1\};\;\; c^* \leftarrow \mathsf{Enc}(\mathsf{pk}, m_{b^*});\;\; b \leftarrow \mathcal{A}_2(\mathsf{pk}, c^*, \mathsf{state})] \enspace .\end{aligned}$$

3 Our Class of Homomorphic Encryption

In this section, we formalize the class of HE schemes with some additional functionalities mentioned in the introduction. We call such an HE scheme a *rerandomizable module-homomorphic encryption scheme* (*RMHE scheme*, in short).[1] We give the definition:

Definition 3 (RMHE schemes)**.** Let $(\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$ be a PKE scheme, Add and Mult be polynomial-time deterministic algorithms[2] and Rerand be a PPT algorithm, where

- $\mathsf{Add}(\mathsf{pk}, c_1, c_2)$ outputs a ciphertext from public key pk and ciphertexts c_1, c_2,
- $\mathsf{Mult}(\mathsf{pk}, m, c)$ outputs a ciphertext from pk, plaintext m and ciphertext c,
- $\mathsf{Rerand}(\mathsf{pk}, c)$ outputs a ciphertext from pk and ciphertext c.

Then we say that $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{Add}, \mathsf{Mult}, \mathsf{Rerand})$ is an *RMHE scheme*, if the following conditions are satisfied, where pk is any public key (see Section 2 for notations):

[1] The term "module-homomorphic" is inspired by the notion of "module" in the area of abstract algebra, which is a set endowed with addition and scalar multiplication analogously to vector spaces.

[2] The arguments in this paper can be easily extended to the case of probabilistic algorithms.

- The plaintext space $\mathcal{M}$ is a finite commutative ring with efficiently computable ring operations, and both $\mathcal{M}$ and its subset $\mathcal{M}^\times$ of invertible elements are samplable.[3]
- We have $\mathsf{Add}(\mathsf{pk}, c_1, c_2) \in \mathcal{C}_{m_1+m_2}$ for any $c_1 \in \mathcal{C}_{m_1}$ and $c_2 \in \mathcal{C}_{m_2}$.
- We have $\mathsf{Mult}(\mathsf{pk}, m, c') \in \mathcal{C}_{m \cdot m'}$ for any $m \in \mathcal{M}$ and $c' \in \mathcal{C}_{m'}$.
- For any $c \in \mathcal{C}_m$, $\mathsf{Rerand}(\mathsf{pk}, c)$ outputs an element of $\mathcal{C}_m$ and its output distribution is identical[4] to the output distribution of $\mathsf{Enc}(\mathsf{pk}, m)$.

The Paillier cryptosystem is an RMHE scheme with $\mathsf{Add}(c_1, c_2) = c_1 \cdot c_2$, $\mathsf{Mult}(m, c') = c'^m$ and $\mathsf{Rerand}(c) = \mathsf{Add}(c, \mathsf{Enc}(0))$. See Section 4 for other existing examples.

For RMHE schemes, the CPA security is in fact not weakened even by restricting the two challenge plaintexts by the adversary to pairs of 0 and a uniformly random element. Precisely, first we give the following definition:

Definition 4 (ZPA security)**.** We say that a PKE scheme Π is *zero plaintext attack* (*ZPA*) *secure*, if for any PPT adversary $\mathcal{A}$, the advantage $\mathrm{Adv}_{\mathcal{A}}(k) := |\Pr[b = b^*] - 1/2|$ of $\mathcal{A}$ in the following procedure is negligible:

$$[(\mathsf{pk}, \mathsf{sk}) \leftarrow \mathsf{Gen}(1^k);\ \ m_0 := 0;\ \ m_1 \leftarrow_R \mathcal{M};\ \ b^* \leftarrow_R \{0,1\};\ \ c^* \leftarrow \mathsf{Enc}(\mathsf{pk}, m_{b^*});\ \ b \leftarrow \mathcal{A}(\mathsf{pk}, c^*)]\ .$$

Then we give the following result, whose proof is analogous to the CPA security for the ElGamal cryptosystem [11] under the DDH assumption:

Lemma 1. *An RMHE scheme Π is CPA secure if and only if it is ZPA secure.*

Proof. Since the CPA security implies the ZPA security by definition, we show that Π is CPA secure if it is ZPA secure.

Let $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ be any PPT CPA adversary for Π. We convert it efficiently into a ZPA adversary $\mathcal{A}^\dagger$ for Π in the following manner. Given a challenge (pk, c^*) for $\mathcal{A}^\dagger$ with $c^* \leftarrow \mathsf{Enc}(m_{b^*})$ as in the ZPA game, the algorithm $\mathcal{A}^\dagger$ first generates $(m'_0, m'_1, \mathsf{state}) \leftarrow \mathcal{A}_1(\mathsf{pk})$. Secondly, $\mathcal{A}^\dagger$ generates $j \leftarrow_R \{0,1\}$ and $c' \leftarrow \mathsf{Rerand}(\mathsf{Add}(c^*, \mathsf{Enc}(m'_j)))$. Finally, $\mathcal{A}^\dagger$ generates $b_j \leftarrow \mathcal{A}_2(\mathsf{pk}, c', \mathsf{state})$ and outputs $b := b_j \,\mathsf{XOR}\, j$.

We have $c' \sim \mathsf{Enc}(m'_j)$ when $b^* = 0$, while $c' \sim \mathsf{Enc}(m_1 + m'_j) \sim \mathsf{Enc}(m^\dagger)$ where $m^\dagger \leftarrow_R \mathcal{M}$ when $b^* = 1$ (since m_1 is uniformly random and independent of m'_j). This implies that the distribution of the input for $\mathcal{A}$ executed in $\mathcal{A}^\dagger$ for the case $b^* = 0$ is identical to the CPA game, while the input for $\mathcal{A}$ in the case $b^* = 1$ is independent of j. Therefore, we have

$$\begin{aligned}\mathrm{Adv}_{\mathcal{A}}(k) &= \left|\Pr[b_j = j \mid b^* = 0] - \frac{1}{2}\right| = \left|2\Pr[b_j = j \wedge b^* = 0] - \frac{1}{2}\right| \\ &= \left|2\left(\Pr[b_j \,\mathsf{XOR}\, j = b^*] - \frac{1}{2}\right) - 2\Pr[(b_j, j) \in \{(0,1),(1,0)\} \wedge b^* = 1] + \frac{1}{2}\right|\end{aligned}$$

[3]Our results can be naturally extended to the cases that $\mathcal{M}$ and $\mathcal{M}^\times$ are approximately samplable.

[4]The property "identical" can in fact be relaxed to "statistically close"; due to this fact, our class of HE schemes here is not entirely included in the class of "group homomorphic encryption" studied in [1].

and this is not larger (by the triangle inequality) than

$$\begin{aligned}
&2\mathrm{Adv}_{\mathcal{A}^\dagger}(k) + \left|\Pr[(b_j,j) \in \{(0,1),(1,0)\} \mid b^* = 1] - \frac{1}{2}\right| \\
&= 2\mathrm{Adv}_{\mathcal{A}^\dagger}(k) + \left|\frac{1}{2}\left(\Pr[b_j = 0 \mid b^* = 1] + \Pr[b_j = 1 \mid b^* = 1]\right) - \frac{1}{2}\right| \\
&= 2\mathrm{Adv}_{\mathcal{A}^\dagger}(k) + \left|\frac{1}{2} - \frac{1}{2}\right| = 2\mathrm{Adv}_{\mathcal{A}^\dagger}(k) \ .
\end{aligned}$$

Hence, $\mathrm{Adv}_{\mathcal{A}}(k)$ is negligible if $\mathrm{Adv}_{\mathcal{A}^\dagger}(k)$ is negligible, concluding the proof of Lemma 1. □

Owing to Lemma 1, instead of the CPA security, we study the ZPA security for RMHE schemes, which is defined by a *non-interactive* game, in order to simplify our argument.

Remark 1. The operation Rerand for RMHE schemes plays a crucial role in Lemma 1. For example, if we modify the Paillier cryptosystem (supposed to be CPA secure) in a way that the new encryption algorithm with plaintext 1 uses no randomness, then the modified scheme satisfies the conditions for RMHE schemes except the existence of Rerand; it is still ZPA secure since the probability that $1 \in \mathcal{M}$ is chosen as the random challenge plaintext is negligible; but the scheme is no longer CPA secure, since the unique fresh ciphertext of challenge plaintext $1 \in \mathcal{M}$ is now easily recognizable.

4 Examples of RMHE Schemes

In this section, we summarize some existing HE schemes in the literature, which indeed satisfy the conditions for RMHE schemes.

4.1 Paillier Cryptosystem and Its Variants

Here we summarize the construction of the Damgård–Jurik cryptosystem [7] which is a generalization of the Paillier cryptosystem [30]. In fact, we describe a simplified (without loss of security) version of the Damgård–Jurik cryptosystem given in the same paper [7], which also includes a simplified version of the Paillier cryptosystem. See the original papers for some omitted details.

The Damgård–Jurik cryptosystem is parameterized by a publicly known positive integer s, where the choice $s = 1$ yields the Paillier cryptosystem. Its public key is the product $n = pq$ of two different large random primes p, q of the same bit length. The corresponding secret key is $\lambda := \mathrm{lcm}(p-1, q-1)$. The plaintext space is $\mathcal{M} := \mathbb{Z}/n^s\mathbb{Z}$. A ciphertext of $m \in \mathcal{M}$ is given by $c := (1+n)^m r^{n^s} \bmod n^{s+1}$ with $r \leftarrow_R (\mathbb{Z}/n^{s+1}\mathbb{Z})^\times$. We define $\mathsf{Add}(\mathsf{pk}, c, c') = c \cdot c'$, $\mathsf{Mult}(\mathsf{pk}, m, c') := c'^m$ and $\mathsf{Rerand}(\mathsf{pk}, c) := c \cdot r^{n^s}$ with $r \leftarrow_R (\mathbb{Z}/n^{s+1}\mathbb{Z})^\times$.[5] Then a straightforward argument shows that all the conditions for RMHE schemes are indeed satisfied.

[5] We note that, in order to make the exponentiation c'^m for the Mult operation well-defined, here we regard the plaintext m as an *integer* rather than a residue class in the ring $\mathbb{Z}/n^s\mathbb{Z}$. The same remark is also applied to the Okamoto–Uchiyama cryptosystem below.

4.2 Okamoto–Uchiyama Cryptosystem

The Okamoto–Uchiyama cryptosystem [28] has a similar structure to the Paillier cryptosystem (see the original paper for some omitted details). Its public key is (n, g, h), where $n = p^2q$ is a composite integer with p, q being two different large random primes of the same bit length, $g \leftarrow_R (\mathbb{Z}/n\mathbb{Z})^\times$ with $g^p \neq 1 \bmod p^2$, and $h := g^n \in (\mathbb{Z}/n\mathbb{Z})^\times$. The corresponding secret key is (p, q). The plaintext space is $\mathcal{M} := \mathbb{Z}/n\mathbb{Z}$. A ciphertext of $m \in \mathcal{M}$ is given by $c := g^m h^r \in (\mathbb{Z}/n\mathbb{Z})^\times$ with $r \leftarrow_R \{0, 1, \dots, n-1\}$. We define $\mathsf{Add}(\mathsf{pk}, c, c') := c \cdot c'$, $\mathsf{Mult}(\mathsf{pk}, m, c') := c'^m$ and $\mathsf{Rerand}(\mathsf{pk}, c) := c \cdot h^r$ with $r \leftarrow_R \{0, 1, \dots, n-1\}$. Then a straightforward argument shows that all the conditions for RMHE schemes are indeed satisfied.

4.3 Goldwasser–Micali Cryptosystem and Its Variants

The ciphertexts in some other known HE schemes, such as the Goldwasser–Micali cryptosystem [16, 17], the Benaloh cryptosystem [3, 13] and the Naccache–Stern cryptosystem [27], have similar structures as the Paillier and the Okamoto–Uchiyama cryptosystems. For example, a ciphertext in the Goldwasser–Micali cryptosystem of a plaintext $m \in \mathcal{M} := \{0, 1\}$ is of the form $c = y^2 x^m \bmod N$ where $N = pq$ is an RSA integer, x is an integer with Legendre symbols $\left(\frac{x}{p}\right) = \left(\frac{x}{q}\right) = -1$, and $y \leftarrow_R (\mathbb{Z}/N\mathbb{Z})^\times$. On the other hand, given an integer $r \geq 2$, a ciphertext in the Benaloh cryptosystem (more precisely, its corrected version in [13]) of a plaintext $m \in \mathcal{M} := \mathbb{Z}/r\mathbb{Z}$ is of the form $c = y^m u^r \bmod n$ where $n = pq$, p and q are large primes with the property that r divides $p - 1$, r and $(p-1)/r$ are relatively prime and r and $q-1$ are relatively prime, $y \in (\mathbb{Z}/n\mathbb{Z})^\times$ satisfying that $y = g^\alpha \bmod p$ for a generator g of $(\mathbb{Z}/p\mathbb{Z})^\times$ and some α which is coprime to r, and $u \leftarrow_R (\mathbb{Z}/n\mathbb{Z})^\times$. The ciphertexts in the Naccache–Stern cryptosystem also have a similar structure, where the range of plaintexts can be much larger than the two cryptosystems above, owing to the use of Chinese Remainder Theorem. By virtue of the similarity of ciphertext structures, an argument similar to the previous two examples shows that these cryptosystems are also RMHE schemes (provided, for the case of the Benaloh cryptosystem, that a factorization of r is known and $(\mathbb{Z}/r\mathbb{Z})^\times$ is approximately samplable).

4.4 "Lifted" ElGamal Cryptosystem and Its Variant

It is known that the ElGamal cryptosystem [11], which is originally a *multiplicative* HE scheme, can be used as an additive HE scheme by regarding the *exponents* of group elements (rather than group elements themselves) as the plaintexts, as long as such an exponent is efficiently computable from a given group element. Usually, the efficient computability of exponents is guaranteed by restricting the range of plaintexts. However, when an HE scheme is used in our generic construction, the plaintexts vary over the full range of exponents. Consequently, to be used in our generic construction, the underlying group for the cryptosystem should be a *Trapdoor Discrete Log* (*TDL*) *group* (see e.g., [25, 29, 31]) in order to make the decryption always efficient.

For future reference, here we describe the additive homomorphic "lifted" ElGamal cryptosystem by assuming the existence of a TDL group, denoted by G. Let g be a generator of G, and let d denote the order of g. A public key consists of G, g and $h := g^x$ with $x \leftarrow_R \mathbb{Z}/d\mathbb{Z}$. The corresponding secret key is x and the trapdoor information for G. The plaintext space is $\mathcal{M} := \mathbb{Z}/d\mathbb{Z}$. A ciphertext of $m \in \mathcal{M}$ is given by $c = (c_1, c_2) := (g^r, h^r g^m)$ with $r \leftarrow_R \mathbb{Z}/d\mathbb{Z}$, which can be efficiently decrypted by computing ${c_1}^{-x} \cdot c_2 = g^m$ first and then recovering the exponent m by the TDL property for G. We define $\mathsf{Add}(\mathsf{pk}, c, c') := (c_1 \cdot c'_1, c_2 \cdot c'_2)$, $\mathsf{Mult}(\mathsf{pk}, m, c') := ({c'_1}^m, {c'_2}^m)$ and $\mathsf{Rerand}(\mathsf{pk}, c) := (c_1 g^r, c_2 h^r)$ with $r \leftarrow_R \mathbb{Z}/d\mathbb{Z}$. Then a straightforward argument shows that all the conditions for RMHE schemes are indeed satisfied. We note that the same argument can be applied to the Damgård's variant [6] of the ElGamal cryptosystem.

5 Our Construction of Homomorphic Encryption

In Section 5.1, we describe our proposed construction of an RMHE scheme, denoted by $\Gamma(T)$, indexed by a rooted tree T to which some building-block RHME schemes Π are associated. (The scheme $\Gamma(T)$ for the "smallest" tree T becomes equivalent to Π.) Then in Section 5.2, we show that, when the tree T is converted to a "larger" tree (see Theorem 1 for the precise meaning), the ZPA (hence the CPA) security for the resulting scheme is at least as difficult to break as the original scheme. Some cases where the latter scheme becomes *strictly* more difficult to break than the former will be studied in later sections.

5.1 Construction

Let $\Pi = (\mathsf{Gen}_\Pi, \mathsf{Enc}_\Pi, \mathsf{Dec}_\Pi, \mathsf{Add}_\Pi, \mathsf{Mult}_\Pi, \mathsf{Rerand}_\Pi)$ denote an RMHE scheme with plaintext space $\mathcal{M}_\Pi$ and ciphertext space $\mathcal{C}_\Pi$. Here we construct an RMHE scheme $\Gamma(T)$ corresponding to a rooted tree T, where a building-block RMHE scheme $\Pi(v) = \Pi(T; v)$ is associated to each leaf v of T. In a special case that all the building-block schemes are the same RMHE scheme Π, we sometimes write $\Gamma(T)$ as $\Gamma(T; \Pi)$ to specify the choice of the building-block scheme. When T is a trivial rooted tree consisting of the root $r = r(T)$ only, we define $\Gamma(T) := \Pi(T; r)$. For non-trivial trees T, we define $\Gamma(T)$ recursively.

In our proposed construction, the collection of the building-block schemes has a requirement. Roughly speaking, the building-block schemes must have a common plaintext space. Here we note that the plaintext space of each scheme is in general dependent on the choice of its public key, therefore the requirement should also be dependent on the distributions of public keys for these building-block schemes. To make it clear, we introduce the following condition for the building-block schemes of our proposed RMHE schemes:

Definition 5 (Combinable schemes)**.** We say that a finite non-empty set $\mathfrak{S}$ of RMHE schemes is *combinable*, if there exists a polynomial-time samplable random variable $\mathsf{Key}_{\mathfrak{S}} = \mathsf{Key}_{\mathfrak{S}}(1^k)$ on the set of the tuples of key pairs $(\mathsf{PK}_\Pi, \mathsf{SK}_\Pi)$ for $\Pi \in \mathfrak{S}$ satisfying the following conditions, where for each non-empty subset $\mathfrak{S}' \subset \mathfrak{S}$, $\mathsf{Key}_{\mathfrak{S}'}$ denotes the restriction of $\mathsf{Key}_{\mathfrak{S}}$ to the components $(\mathsf{PK}_{\Pi'}, \mathsf{SK}_{\Pi'})$ indexed by $\Pi' \in \mathfrak{S}'$:

- We have $\mathsf{Key}_{\{\Pi\}} \sim \mathsf{Gen}_\Pi$ for each $\Pi \in \mathfrak{S}$.[6] [7]
- For any pair of non-empty subsets $\mathfrak{S}' \subset \mathfrak{S}'' \subset \mathfrak{S}$, there exists a PPT algorithm $\mathsf{ExpandKey}_{\mathfrak{S}' \to \mathfrak{S}''}$ with the property that the distribution of $(\mathsf{PK}_\Pi)_{\Pi \in \mathfrak{S}''}$ given by $(\mathsf{PK}_\Pi, \mathsf{SK}_\Pi)_{\Pi \in \mathfrak{S}'} \leftarrow \mathsf{Key}_{\mathfrak{S}'}$ and $(\mathsf{PK}_\Pi)_{\Pi \in \mathfrak{S}'' \setminus \mathfrak{S}'} \leftarrow \mathsf{ExpandKey}_{\mathfrak{S}' \to \mathfrak{S}''}((\mathsf{PK}_\Pi)_{\Pi \in \mathfrak{S}'})$ is identical to the distribution of $(\mathsf{PK}_\Pi)_{\Pi \in \mathfrak{S}''}$ given by $\mathsf{Key}_{\mathfrak{S}''}$.[8]

[6] Namely, $\mathsf{Key}_{\mathfrak{S}}$ is a joint distribution with marginal distributions Gen_Π, $\Pi \in \mathfrak{S}$.

[7] The following construction is easily extendable to a slightly more general case that the two distributions are statistically close. The same also holds for the other parts of the definition.

[8] Intuitively, given public keys (but *not* secret keys) for some schemes in $\mathfrak{S}$, public keys for other schemes in $\mathfrak{S}$ can be efficiently sampled with the correct conditional probability; this property will be required in the security proofs below.

- For any $(\mathsf{PK}_\Pi, \mathsf{SK}_\Pi)_{\Pi \in \mathfrak{S}}$ generated by $\mathsf{Key}_{\mathfrak{S}}$, the plaintext spaces $\mathcal{M}_\Pi$ associated to the public key PK_Π are common for all $\Pi \in \mathfrak{S}$.

We note that this (somewhat technical) definition indeed covers both of the following two important cases:

- There is only a single building-block scheme in $\mathfrak{S}$.
- For each security parameter, the possibility of the plaintext space for each $\Pi \in \mathfrak{S}$ is unique and it is common for all Π. Now $\mathsf{Key}_{\mathfrak{S}}$ can be the combination of *independent* distributions Gen_Π for $\Pi \in \mathfrak{S}$, and the construction of $\mathsf{ExpandKey}$ is obvious.

In the following arguments, unless otherwise specified, we suppose that $\mathfrak{S}$ is a combinable set of RMHE schemes and the RMHE scheme $\Pi(v) = \Pi(T; v)$ associated to a leaf v of a tree T is a member of $\mathfrak{S}$. We define $\mathfrak{S}[T]$ to be the set of $\Pi(T; v)$ for all leaves v of T (we do *not* assume that $\mathfrak{S}[T] = \mathfrak{S}$, i.e., not all members in $\mathfrak{S}$ are always used as the building-block schemes in each construction).

We describe our proposed construction of RMHE schemes. Let $V = V(T)$ and $E = E(T)$ be the vertex set and the edge set of the tree T. For $e \in E$, let $\mathsf{top}(e)$ and $\mathsf{bot}(e)$ denote the vertices of e closer to and farther from the root r, respectively. Let $v_1 \to v_2$ denote the edge e with $\mathsf{top}(e) = v_1$ and $\mathsf{bot}(e) = v_2$. For $v \in V$, let T_v denote the subtree of T with root v, let $v_\downarrow$ denote the set of the child vertices of v, and let v_* denote the last element of $v_\downarrow$ (in a fixed ordering). Then we recursively construct our RMHE scheme $\Gamma(T) = (\mathsf{Gen}_T, \mathsf{Enc}_T, \mathsf{Dec}_T, \mathsf{Add}_T, \mathsf{Mult}_T, \mathsf{Rerand}_T)$ as follows (see Figure 1), where, for each $v \in V$, we write

$$\begin{aligned} \mathsf{pk}_{\wedge v} &:= (\mathsf{PK}_\Pi)_{\Pi \in \mathfrak{S}[T_v]} \cup (\mathsf{PK}_e)_{e \in E(T_v)} \enspace , \\ \mathsf{sk}_{\wedge v} &:= (\mathsf{SK}_\Pi)_{\Pi \in \mathfrak{S}[T_v]} \cup (\mathsf{SK}_e)_{e \in E(T_v)} \enspace . \end{aligned}$$

$\mathsf{Gen}_T(1^k)$. The algorithm generates $(\mathsf{PK}_\Pi, \mathsf{SK}_\Pi)_{\Pi \in \mathfrak{S}[T]} \leftarrow \mathsf{Key}_{\mathfrak{S}[T]}(1^k)$ where $\mathsf{Key}_{\mathfrak{S}[T]}$ is the random variable introduced in Definition 5, and sets $\mathcal{M} := \mathcal{M}_\Pi$. Then for $e \in E$, the algorithm generates $\mathsf{SK}_e \leftarrow_R \mathcal{M}^\times$ and $\mathsf{PK}_e \leftarrow \mathsf{Enc}_{\mathsf{bot}(e)}(\mathsf{pk}_{\wedge \mathsf{bot}(e)}, \mathsf{SK}_e)$, where we abbreviate Enc_{T_v} to Enc_v for $v \in V$ (we also use similar abbreviations for other algorithms). The output of the algorithm is the pair $(\mathsf{pk}, \mathsf{sk}) := (\mathsf{pk}_{\wedge r}, \mathsf{sk}_{\wedge r})$.

$\mathsf{Enc}_T(\mathsf{pk}, m)$ ($m \in \mathcal{M}$). The algorithm generates $s_v \leftarrow_R \mathcal{M}$ for each $v \in r_\downarrow \setminus \{r_*\}$, and sets $s_{r_*} := m - \sum_{u \in r_\downarrow \setminus \{r_*\}} s_u$. Then for each $v \in r_\downarrow$, the algorithm generates $c_v \leftarrow \mathsf{Rerand}_v(\mathsf{pk}_{\wedge v}, \mathsf{Mult}_v(\mathsf{pk}_{\wedge v}, s_v, \mathsf{PK}_{r \to v}))$. Now the output is $c := (c_v)_{v \in r_\downarrow}$.

$\mathsf{Dec}_T(\mathsf{sk}, c)$ ($c = (c_v)_{v \in r_\downarrow}$). The algorithm first generates $t_v \leftarrow \mathsf{Dec}_v(\mathsf{sk}_{\wedge v}, c_v)$ for each $v \in r_\downarrow$. If $t_v = \perp$ or $\mathsf{SK}_{r \to v} \not\in \mathcal{M}^\times$ for some v, then the output is $\perp$. Otherwise, the output is $\sum_{v \in r_\downarrow} t_v / \mathsf{SK}_{r \to v} \in \mathcal{M}$.

$\mathsf{Add}_T(\mathsf{pk}, c, c')$. The algorithm generates $c''_v \leftarrow \mathsf{Add}_v(\mathsf{pk}_{\wedge v}, c_v, c'_v)$ for each $v \in r_\downarrow$. Then the output is $c'' = (c''_v)_{v \in r_\downarrow}$.

$\mathsf{Mult}_T(\mathsf{pk}, m, c')$. The algorithm generates $c''_v \leftarrow \mathsf{Mult}_v(\mathsf{pk}_{\wedge v}, m, c'_v)$ for each $v \in r_\downarrow$. Then the output is $c'' = (c''_v)_{v \in r_\downarrow}$.

$\mathsf{Rerand}_T(\mathsf{pk}, c)$. The algorithm generates $s_v \leftarrow_R \mathcal{M}$ for each $v \in r_\downarrow \setminus \{r_*\}$, and sets $s_{r_*} := -\sum_{u \in r_\downarrow \setminus \{r_*\}} s_u$. Then for each $v \in r_\downarrow$, the algorithm generates $c'_v \leftarrow \mathsf{Rerand}_v(\mathsf{pk}_{\wedge v}, \mathsf{Add}_v(\mathsf{pk}_{\wedge v}, c_v, \mathsf{Mult}_v(\mathsf{pk}_{\wedge v}, s_v, \mathsf{PK}_{r \to v})))$. The output is $c' := (c'_v)_{v \in r_\downarrow}$.

Key generation $\mathsf{Gen}_T(1^k)$

$(\mathsf{PK}_\Pi, \mathsf{SK}_\Pi)_{\Pi \in \mathfrak{S}[T]} \leftarrow \mathsf{Key}_{\mathfrak{S}[T]}(1^k)$, $\mathcal{M} := \mathcal{M}_\Pi$

$$\left(\begin{array}{l} \mathsf{pk}_{\wedge v} := (\mathsf{PK}_\Pi)_{\Pi \in \mathfrak{S}[T_v]} \cup (\mathsf{PK}_e)_{e \in E(T_v)} \\ \mathsf{sk}_{\wedge v} := (\mathsf{SK}_\Pi)_{\Pi \in \mathfrak{S}[T_v]} \cup (\mathsf{SK}_e)_{e \in E(T_v)} \end{array} (v \in V) \right)$$

For $e \in E$: $\mathsf{SK}_e \leftarrow_R \mathcal{M}^\times$, $\mathsf{PK}_e \leftarrow \mathsf{Enc}_{\mathsf{bot}(e)}(\mathsf{pk}_{\wedge \mathsf{bot}(e)}, \mathsf{SK}_e)$ $\rightsquigarrow$ Output $(\mathsf{pk}, \mathsf{sk}) := (\mathsf{pk}_{\wedge r}, \mathsf{sk}_{\wedge r})$

Encryption $\mathsf{Enc}_T(\mathsf{pk}, m)$, $m \in \mathcal{M}$

For each $v \in r_\downarrow$:
- $s_v \leftarrow_R \mathcal{M}$ (if $v \neq r_*$)
- $s_v := m - \sum_{u \neq r_*} s_u$ (if $v = r_*$)
- $c_v \leftarrow \mathsf{Rerand}_v(\mathsf{pk}_{\wedge v}, \mathsf{Mult}_v(\mathsf{pk}_{\wedge v}, s_v, \mathsf{PK}_{r \to v}))$

Output $c := (c_v)_{v \in r_\downarrow}$

Decryption $\mathsf{Dec}_T(\mathsf{sk}, c)$, $c = (c_v)_{v \in r_\downarrow} \in \mathcal{C}$

For each $v \in r_\downarrow$:
- $t_v \leftarrow \mathsf{Dec}_v(\mathsf{sk}_{\wedge v}, c_v)$
- If $t_v = \perp$ or $\mathsf{SK}_{r \to v} \notin \mathcal{M}^\times$, then output $\perp$

Output $\sum_{v \in r_\downarrow} t_v / \mathsf{SK}_{r \to v}$

Addition $\mathsf{Add}_T(\mathsf{pk}, c, c')$, $c, c' \in \mathcal{C}$

For each $v \in r_\downarrow$: $c''_v \leftarrow \mathsf{Add}_v(\mathsf{pk}_{\wedge v}, c_v, c'_v)$

Output $c'' := (c''_v)_{v \in r_\downarrow}$

Scalar multiplication $\mathsf{Mult}_T(\mathsf{pk}, m, c')$, $m \in \mathcal{M}$, $c' \in \mathcal{C}$

For each $v \in r_\downarrow$: $c''_v \leftarrow \mathsf{Mult}_v(\mathsf{pk}_{\wedge v}, m, c'_v)$

Output $c'' := (c''_v)_{v \in r_\downarrow}$

Rerandomization $\mathsf{Rerand}_T(\mathsf{pk}, c)$, $c \in \mathcal{C}$

$$\left. \begin{array}{l} \text{For each } v \in r_\downarrow\text{: } s_v \leftarrow_R \mathcal{M} \text{ (if } v \neq r_*\text{)}, \ s_v := -\sum_{u \neq r_*} s_u \text{ (if } v = r_*\text{)} \\ \quad c'_v \leftarrow \mathsf{Rerand}_v(\mathsf{pk}_{\wedge v}, \mathsf{Add}_v(\mathsf{pk}_{\wedge v}, c_v, \mathsf{Mult}_v(\mathsf{pk}_{\wedge v}, s_v, \mathsf{PK}_{r \to v}))) \end{array} \right\} \rightsquigarrow \text{Output } c' := (c'_v)_{v \in r_\downarrow}$$

Figure 1: Recursive construction of our proposed RMHE scheme $\Gamma(T)$ (here r is the root of the tree $T = (V, E)$; r_* is the last element of the set $r_\downarrow$ of the child vertices of r; some subscripts "T_v" of algorithms are abbreviated to v; we set $\Gamma(T_v) := \Pi(v) \in \mathfrak{S}$ for any leaf v of T; and we define $\mathfrak{S}[T]$ to be the set of all $\Pi(v)$ for leaves v of T)

We say that a tuple of plaintexts is a *share set* of $m \in \mathcal{M}$, if their sum is m. We note that the tuples $(s_v)_v$ in the definitions of Enc_T and Rerand_T are uniformly random share sets of m and of 0, respectively. Then a straightforward argument shows the following property:

Proposition 1. *The scheme $\Gamma(T)$ is an RMHE scheme, where the set $\mathcal{C}_{T,m}$ of valid ciphertexts of plaintext m in $\Gamma(T)$ is defined recursively to be the union of the direct product $\prod_{v \in r_\downarrow} \mathcal{C}_{T_v, m_v \cdot \mathsf{SK}_{r \to v}}$ over all share sets $(m_v)_{v \in r_\downarrow}$ of m (see above for the terminology). Moreover, by using the notations in Figure 1, we have $c_v \sim \mathsf{Enc}_v(\mathsf{pk}_{\wedge v}, s_v \cdot \mathsf{SK}_{r \to v})$ for outputs of Enc_T, and $c'_v \sim \mathsf{Enc}_v(\mathsf{pk}_{\wedge v}, \mathsf{Dec}_v(\mathsf{sk}_{\wedge v}, c_v) + s_v \cdot \mathsf{SK}_{r \to v})$ for outputs of Rerand_T.*

Proof. The claim is obvious when T is the trivial tree; now $\Gamma(T) = \Pi(T; r)$. Hence, we consider the case of non-trivial trees T only. The condition for the plaintext space is implied by that for the building-block RMHE schemes.

We use induction on the depth of T. First, the properties $c_v \sim \mathsf{Enc}_v(\mathsf{pk}_{\wedge v}, s_v \cdot \mathsf{SK}_{r \to v})$ for outputs of Enc_T and $c'_v \sim \mathsf{Enc}_v(\mathsf{pk}_{\wedge v}, \mathsf{Dec}_v(\mathsf{sk}_{\wedge v}, c_v) + s_v \cdot \mathsf{SK}_{r \to v})$ for outputs of Rerand_T follow from the properties of the RMHE schemes $\Gamma(T_v)$. Now for the algorithm Enc_T, $(s_v)_{v \in r_\downarrow}$ is a uniformly random share set of m, therefore the outputs of Enc_T belong to the set $\mathcal{C}_{T,m}$ defined in the statement. On the other hand, for the algorithm $\mathsf{Dec}_T(\mathsf{sk}, c)$ for $c = (c_v)_{v \in r_\downarrow} \in \mathcal{C}_{T,m}$ with $c_v \in \mathcal{C}_{T_v, m_v \cdot \mathsf{SK}_{r \to v}}$ for each v, we have $t_v = m_v \cdot \mathsf{SK}_{r \to v}$ and $\mathsf{Sk}_{r \to v} \in \mathcal{M}^\times$, therefore the output is $\sum_{v \in r_\downarrow} m_v = m$. Hence the correctness holds.

By the structures of the valid ciphertext spaces $\mathcal{C}_{T,m}$ specified in the statement, the conditions for the algorithms Add_T and Mult_T follow from the properties for the RMHE schemes $\Gamma(T_v)$ and the fact that the component-wise addition of share sets $(m_v)_{v \in r_\downarrow}$ and

$(m'_v)_{v \in r_\downarrow}$ of m and m', respectively, is a share set of $m + m'$, and the component-wise scalar multiplication $(m \cdot m'_v)_{v \in r_\downarrow}$ for a share set $(m'_v)_{v \in r_\downarrow}$ of m' by $m \in \mathcal{M}$ is a share set of $m \cdot m'$. Moreover, for the algorithm $\mathsf{Rerand}_T(\mathsf{pk}, c)$ for $c = (c_v)_{v \in r_\downarrow} \in \mathcal{C}_{T,m}$ with $c_v \in \mathcal{C}_{T_v, m_v \cdot \mathsf{SK}_{r \to v}}$ for each v, we have $c'_v \sim \mathsf{Enc}_v(\mathsf{pk}_{\wedge v}, (m_v + s_v) \cdot \mathsf{SK}_{r \to v})$ by the properties of the RMHE schemes $\Gamma(T_v)$. Since $(m_v)_{v \in r_\downarrow}$ is a share set of m and $(s_v)_{v \in r_\downarrow}$ is a uniformly random share set of 0, it follows that $(m_v + s_v)_{v \in r_\downarrow}$ is a uniformly random share set of m, therefore we have $c' \sim \mathsf{Enc}_T(\mathsf{pk}, m)$ by the argument in the previous paragraph. This completes the proof of Proposition 1. □

Example 1. We consider the case of the tree, denoted by T_ℓ, of depth one consisting of the root r, ℓ leaves $v_1, \dots, v_\ell$ and ℓ edges $r \to v_1, \dots, r \to v_\ell$ ($\ell \geq 1$). A public key pk for $\Gamma(T_\ell)$ consists of a public key PK_Π for each $\Pi \in \mathfrak{S}[T_\ell]$ and ℓ ciphertexts $\mathsf{PK}_{r \to v_j} \leftarrow \mathsf{Enc}_{\Pi(j)}(\mathsf{PK}_{\Pi(j)}, \mathsf{SK}_{r \to v_j})$ in $\Pi(j)$ with $\mathsf{SK}_{r \to v_j} \leftarrow_R \mathcal{M}^\times$ ($j = 1, \dots, \ell$), where we abbreviate $\Pi(v_j)$ to $\Pi(j)$.

To encrypt $m \in \mathcal{M}$, we choose $s_1, \dots, s_{\ell-1} \leftarrow_R \mathcal{M}$ and generate

$$c_{v_j} \leftarrow \begin{cases} \mathsf{Rerand}_{\Pi(j)}(\mathsf{Mult}_{\Pi(j)}(s_j, \mathsf{PK}_{r \to v_j})) & \text{for } j = 1, \dots, \ell - 1 \ , \\ \mathsf{Rerand}_{\Pi(\ell)}(\mathsf{Mult}_{\Pi(\ell)}(m - \sum_{i=1}^{\ell-1} s_i, \mathsf{PK}_{r \to v_\ell})) & \text{for } j = \ell \ . \end{cases}$$

Therefore, a ciphertext c consists of ℓ ciphertexts $c_{v_1}, \dots, c_{v_\ell}$ in the building-block RMHE schemes. The homomorphic operations in $\Gamma(T_\ell)$ are made from those in the building-block schemes for the ℓ components.

Example 2. We consider the case of the tree, denoted by $T^\S$, of depth two with five vertices $r = r(T^\S)$, v_1, v_2, v_3, v_4 and four edges $r \to v_1$, $r \to v_2$, $v_2 \to v_3$ and $v_2 \to v_4$. For the four components of a public key pk for $\Gamma(T^\S)$ other than public keys PK_Π for $\Pi \in \mathfrak{S}[T^\S]$, we have $\mathsf{PK}_e \leftarrow \mathsf{Enc}_\Pi(\mathsf{PK}_\Pi, \mathsf{SK}_e)$ with $\mathsf{SK}_e \leftarrow_R \mathcal{M}^\times$ for $e = (r \to v_1)$, $(v_2 \to v_3)$ and $(v_2 \to v_4)$, where $\Pi = \Pi(v_1)$, $\Pi(v_3)$ and $\Pi(v_4)$ for the three choices of the edge e, respectively. To generate the remaining component $\mathsf{PK}_{r \to v_2} = (\mathsf{PK}^{(1)}_{r \to v_2}, \mathsf{PK}^{(2)}_{r \to v_2})$ of pk, we choose $s_{\mathrm{pk}} \leftarrow_R \mathcal{M}$ and generate

$$\begin{aligned} \mathsf{PK}^{(1)}_{r \to v_2} &\leftarrow \mathsf{Rerand}_{\Pi(v_3)}(\mathsf{Mult}_{\Pi(v_3)}(s_{\mathrm{pk}}, \mathsf{PK}_{v_2 \to v_3})) \ , \\ \mathsf{PK}^{(2)}_{r \to v_2} &\leftarrow \mathsf{Rerand}_{\Pi(v_4)}(\mathsf{Mult}_{\Pi(v_4)}(\mathsf{SK}_{r \to v_2} - s_{\mathrm{pk}}, \mathsf{PK}_{v_2 \to v_4})) \end{aligned}$$

as in the definition of $\mathsf{Enc}_{T_{v_2}}$ (where we omit the symbols PK_Π). Summarizing, pk consists of PK_Π for each $\Pi \in \mathfrak{S}[T^\S]$ and five ciphertexts in the building-block RMHE schemes.

To encrypt $m \in \mathcal{M}$, we first choose $s_1 \leftarrow_R \mathcal{M}$ and generate

$$c_{v_1} \leftarrow \mathsf{Rerand}_\Pi(\mathsf{Mult}_\Pi(s_1, \mathsf{PK}_{r \to v_1})) \ .$$

To generate the other component c_{v_2}, we choose $s_2 \leftarrow_R \mathcal{M}$ and generate

$$\begin{aligned} c_{v_3} &\leftarrow \mathsf{Rerand}_\Pi(\mathsf{Add}_\Pi(\mathsf{Mult}_\Pi(m - s_1, \mathsf{PK}^{(1)}_{r \to v_2}), \mathsf{Mult}_\Pi(s_2, \mathsf{PK}_{v_2 \to v_3}))) \ , \\ c_{v_4} &\leftarrow \mathsf{Rerand}_\Pi(\mathsf{Add}_\Pi(\mathsf{Mult}_\Pi(m - s_1, \mathsf{PK}^{(2)}_{r \to v_2}), \mathsf{Mult}_\Pi(-s_2, \mathsf{PK}_{v_2 \to v_4}))) \ . \end{aligned}$$

Then we have $c_{v_2} = (c_{v_3}, c_{v_4})$, therefore a ciphertext c of plaintext m consists of three ciphertexts c_{v_1}, c_{v_3} and c_{v_4} in the building-block schemes. The homomorphic operations in $\Gamma(T^\S)$ are also decomposed into combinations of homomorphic operations in the building-block schemes for the three components of ciphertexts.

5.2 Security Implications for Different Trees

Here we study some implication relations of the ZPA (hence the CPA) security for $\Gamma(T)$ between different trees T. For the purpose, we define some transformations of the trees $T = (V, E)$, where we also concern the correspondences between building-block RMHE schemes in $\mathfrak{S}$ and the leaves of T. Let $L = L(T)$ denote the set of leaves of T, and let r be the root of T. Then the transformations for the trees are defined as follows:

$\mathrm{Fork}_v(T)$ **(for** $v \in V \setminus L$**):** Add a new edge $e^\dagger$ with $\mathsf{top}(e^\dagger) = v$; now $\mathsf{bot}(e^\dagger)$ is a new leaf of the resulting tree. Moreover, associate an RMHE scheme $\Pi(\mathsf{bot}(e^\dagger))$ to the new leaf $\mathsf{bot}(e^\dagger)$.

$\mathrm{Divide}_{v,v'}(T)$ **(for** $(v \to v') \in E$**):** Add a new vertex $v^\dagger$ between v and v'.

$\mathrm{Grow}(T)$**:** Add a new root $r^\dagger$, i.e., r is the unique child vertex of $r^\dagger$ in the new tree.

We show below that the ZPA security becomes at least as difficult to break as the original situation when the transformation Fork_v is applied to the tree T, while the other transformations $\mathrm{Divide}_{v,v'}$ and Grow do not change the difficulty to break the ZPA security.

We note that, when the parent vertex of $v' \in V$, denoted here by $v'^\uparrow$, is a child vertex of $v \in V$ and v' is the unique child vertex of $v'^\uparrow$, the inverse transformation of $\mathrm{Divide}_{v,v'}$ can be applied to T; it concatenates the two edges $v \to v'^\uparrow$ and $v'^\uparrow \to v'$ to form a new edge $v \to v'$. Similarly, when the root r of T has a unique child vertex r', the inverse transformation of Grow can be applied to T; it removes r and makes r' the new root.

We define the relation $T \preceq T'$ for trees which means that T can be converted to T' by a (possibly empty) sequence of transformations of the form Fork_v, $\mathrm{Divide}_{v,v'}$, $\mathrm{Divide}_{v,v'}^{-1}$, Grow or Grow^{-1}. We emphasize that the assignments of building-block schemes to the leaves are relevant to the definition of the relation. We also note that, since the five kinds of transformations above do not remove any leaf of the tree, it follows that if $T \preceq T'$, then any leaf v of T is also a leaf of T' and we have $\Pi(T; v) = \Pi(T'; v)$. To compare the ZPA security for $\Gamma(T)$ and $\Gamma(T')$, we give the following lemma, which implies that a random challenge in the ZPA game for $\Gamma(T)$ can be efficiently converted to a random challenge in the ZPA game for $\Gamma(T')$:

Lemma 2. *For each* $\Phi \in \{\mathrm{Fork}_v, \mathrm{Divide}_{v,v'}, \mathrm{Divide}_{v,v'}^{-1}, \mathrm{Grow}, \mathrm{Grow}^{-1}\}$ *and any* T *to which the transformation* Φ *is applicable, there exists a PPT transformation* $\varphi_{\Phi,T}$ *of pairs of public keys and ciphertexts, not using the secret keys for* $\Gamma(T)$*, satisfying the following:*

For a public key pk *of* $\Gamma(T)$ *following the distribution* $\mathsf{Gen}_T(1^k)$ *and* $c \leftarrow \mathsf{Enc}_T(\mathsf{pk}, m)$ *with* $m = 0$ *or* $m \leftarrow_R \mathcal{M}$*, respectively,* $(\mathsf{pk}', c') \leftarrow \varphi_{\Phi,T}(\mathsf{pk}, c)$ *satisfies the followings:*

- pk' *is a public key of* $\Gamma(\Phi(T))$ *and has the same component* $\mathsf{PK}_{\Pi(v)}$ *as* pk *for each leaf* v *of* T *(hence the same plaintext space).*
- *The distribution of* pk' *is identical to the distribution of the first component of the output of* $\mathsf{Gen}_{\Phi(T)}(1^k)$*.*
- *The distribution of* c' *is identical to the output distribution of* $\mathsf{Enc}_{\Phi(T)}(\mathsf{pk}', m)$*, where* $m = 0$ *or* $m \leftarrow_R \mathcal{M}$ *as above.*

Proof. We prove the claim, together with the following auxiliary property: For each fixed m as in the statement, we have $c' \sim \mathsf{Enc}_{\Phi(T)}(\mathsf{pk}', m \cdot \sigma)$ where $\sigma \in \mathcal{M}^\times$ may depend on pk but is independent of m, and c' is computable from pk', c and σ only. We note that, if it holds, then $m \cdot \sigma \sim m$ for each case of $m = 0$ and $m \leftarrow_R \mathcal{M}$, therefore the third condition in the statement follows.

When $\Phi = \mathrm{Grow}$, pk' is correctly generated from pk by adding a component $\mathsf{PK}_{r^\dagger \to r} \leftarrow \mathsf{Enc}_T(\mathsf{pk}, \mathsf{SK}_{r^\dagger \to r})$ with $\mathsf{SK}_{r^\dagger \to r} \leftarrow_R \mathcal{M}^\times$, and we define $c' := (c)$, i.e., c' consists of a unique component c. Then the claim holds, with $\sigma = (\mathsf{SK}_{r^\dagger \to r})^{-1}$. Conversely, when $\Phi = \mathrm{Grow}^{-1}$, pk' is correctly generated by removing the component $\mathsf{PK}_{r^\dagger \to r}$ from pk, and c' is set to be the unique component of c, which satisfies the claim, with $\sigma = \mathsf{SK}_{r^\dagger \to r}$.

For the other Φ, we use induction on the depth of v. When $\Phi = \mathrm{Fork}_v$ and $v = r$, pk' is given by first generating an additional public key component $\mathsf{PK}_{\Pi(\mathsf{bot}(e^\dagger))}$ for $\Pi(\mathsf{bot}(e^\dagger))$ by using the algorithm $\mathsf{ExpandKey}_{\mathfrak{S}' \to \mathfrak{S}''}$ given in Definition 5 with $\mathfrak{S}' := \mathfrak{S}[T]$ and $\mathfrak{S}'' := \mathfrak{S}' \cup \{\Pi(\mathsf{bot}(e^\dagger))\}$, and then adding a component $\mathsf{PK}_{e^\dagger} \leftarrow \mathsf{Enc}_{\Pi(\mathsf{bot}(e^\dagger))}(\mathsf{PK}_{\Pi(\mathsf{bot}(e^\dagger))}, \mathsf{SK}_{e^\dagger})$ where $\mathsf{SK}_{e^\dagger} \leftarrow_R \mathcal{M}^\times$. On the other hand, c' is given by first converting c to a ciphertext of m for $\Gamma(\Phi(T))$ by adding a component $c_{\mathsf{bot}(e^\dagger)} \leftarrow \mathsf{Enc}_{\Pi(\mathsf{bot}(e^\dagger))}(\mathsf{PK}_{\Pi(\mathsf{bot}(e^\dagger))}, 0)$ and then rerandomizing the resulting ciphertext for $\Gamma(\Phi(T))$. Hence the claim holds, with $\sigma = 1$.

For the remaining cases, let w be the unique element of $r_\downarrow$ with $v \in V(T_w)$ when $\Phi = \mathrm{Fork}_v$ (and $v \neq r$) and $v' \in V(T_w)$ when $\Phi \in \{\mathrm{Divide}_{v,v'}, \mathrm{Divide}_{v,v'}^{-1}\}$. We set $(w', \Phi') := (v^\dagger, \mathrm{Grow})$ if $\Phi = \mathrm{Divide}_{v,v'}$ and $v = r$ (now $w = v'$); set $(w', \Phi') := (v', \mathsf{Grow}^{-1})$ if $\Phi = \mathrm{Divide}_{v,v'}^{-1}$ and $v = r$ (now $w = v'^{\uparrow}$); and set $(w', \Phi') := (w, \Phi)$ otherwise (now $v \in V(T_w)$). Let sk' denote the secret key after the transformation. To generate pk', we first convert the pair $(\mathsf{pk}_{\wedge w}, \mathsf{PK}_{r \to w})$ to $(\mathsf{pk}'_{\wedge w'}, \mathsf{PK}'_{r \to w'}) \leftarrow \varphi_{\Phi', T_w}(\mathsf{pk}_{\wedge w}, \mathsf{PK}_{r \to w})$. By the induction hypothesis, we have $\mathsf{PK}'_{r \to w'} \sim \mathsf{Enc}_{w'}(\mathsf{pk}'_{\wedge w'}, \mathsf{SK}_{r \to w} \cdot \sigma)$ with $\sigma \in \mathcal{M}^\times$ associated to φ_{Φ', T_w}. We also convert $\mathsf{PK}_{r \to u}$ for each $u \in r_\downarrow \setminus \{w\}$ to $\mathsf{PK}'_{r \to u} \leftarrow \mathsf{Rerand}_u(\mathsf{pk}_{\wedge u}, \mathsf{Mult}_u(\mathsf{pk}_{\wedge_u}, \sigma, \mathsf{PK}_{r \to u}))$; then we have $\mathsf{PK}'_{r \to u} \sim \mathsf{Enc}_u(\mathsf{pk}_{\wedge u}, \mathsf{SK}_{r \to u} \cdot \sigma)$. This choice of pk' corresponds to $\mathsf{SK}'_{r \to w'} := \mathsf{SK}_{r \to w} \cdot \sigma$ and $\mathsf{SK}'_{r \to u} := \mathsf{SK}_{r \to u} \cdot \sigma$. On the other hand, to generate c', we convert the component c_w of c to the second component $c'_{w'}$ of $\varphi_{\Phi', T_w}(\mathsf{pk}_{\wedge w}, c_w)$ computed from $\mathsf{pk}'_{\wedge w}$, c_w and σ, and set $c'_u \leftarrow \mathsf{Rerand}_u(\mathsf{pk}_{\wedge u}, \mathsf{Mult}_u(\mathsf{pk}_{\wedge u}, \sigma, c_u))$ for each $u \in r_\downarrow \setminus \{w\}$. By the induction hypothesis, when $c_u \in \mathcal{C}_{T_u, m_u \cdot \mathsf{SK}_{r \to u}}$ for each $u \in r_\downarrow$, we have

$$
\begin{aligned}
c'_{w'} &\sim \mathsf{Enc}_{w'}(\mathsf{pk}'_{\wedge w'}, m_w \cdot \mathsf{SK}_{r \to w} \cdot \sigma) \sim \mathsf{Enc}_{w'}(\mathsf{pk}'_{\wedge w'}, m_w \cdot \mathsf{SK}'_{r \to w'}) \ , \\
c'_u &\sim \mathsf{Enc}_u(\mathsf{pk}'_{\wedge u}, m_u \cdot \mathsf{SK}_{r \to u} \cdot \sigma) \sim \mathsf{Enc}_u(\mathsf{pk}'_{\wedge u}, m_u \cdot \mathsf{SK}'_{r \to u}) \text{ for } u \neq w \ .
\end{aligned}
$$

This implies that $c' \sim \mathsf{Enc}_{\Phi(T)}(\mathsf{pk}', m)$, therefore the claim holds, where $\sigma = 1$. □

By applying Lemma 2 repeatedly, if $T \preceq T'$, then an input for an adversary in the ZPA game for $\Gamma(T)$ can be efficiently converted to a correctly distributed input for an adversary in the ZPA game for $\Gamma(T')$, therefore an attack to break $\Gamma(T)$ can be reduced to an attack breaking $\Gamma(T')$. This implies the following result:

Theorem 1. *If $\Gamma(T)$ is ZPA secure and $T \preceq T'$, then $\Gamma(T')$ is ZPA secure as well.*

In particular, since we have $T_v \preceq T$ for any leaf v of T, the argument above implies the following property, which means that the underlying assumption for the ZPA security of $\Gamma(T)$ is at least as weak as the *logical OR* of those for the building-block schemes:

Theorem 2. *If* at least one *of the RMHE schemes $\Pi \in \mathfrak{S}[T]$ is ZPA secure, then $\Gamma(T)$ is ZPA secure as well.*

Moreover, we consider the case of a single building-block scheme; $\mathfrak{S} = \{\Pi\}$. We use the notations T_ℓ with $\ell \geq 1$ for the trees of depth one as in Example 1, and let T_0 denote the trivial tree. In this case, we have $T_{\ell+1} = \mathrm{Fork}_{r(T_\ell)}(T_\ell)$ for $\ell \geq 1$, and $T_1 = \mathrm{Grow}(T_0)$, therefore $T_1 \preceq T_0 \preceq T_1 \preceq T_2 \preceq \cdots$. Hence, by Theorem 1, we have the following result:

Theorem 3. *Suppose that $\mathfrak{S} = \{\Pi\}$. Then for any $\ell \geq 1$, the ZPA security for $\Gamma(T_\ell; \Pi)$ implies the ZPA security for $\Gamma(T_{\ell+1}; \Pi)$. Moreover, the ZPA security for $\Gamma(T_1; \Pi)$ is equivalent to the ZPA security for Π.*

6 Computational Model for Non-Implication Results

In this section, in order to discuss *non-implication* relations between the ZPA security for our RMHE schemes $\Gamma(T)$ with different trees T, we introduce a computational model to formalize a class of "natural" reductions between the ZPA adversaries for these schemes. Our model is a variant of the Boolean circuit model (see e.g., Section 1.2.4.1 of [14]) with a flavor of the generic group model [33], associated to the building-block RMHE schemes Π in a combinable set $\mathfrak{S}$. Each circuit in the model represents a ZPA adversary for some scheme $\Gamma(T)$, called an *outer scheme*, and it internally uses oracles that break the ZPA security of other schemes $\Gamma(T')$, called *inner schemes*. We note that each challenge in the ZPA game for any scheme $\Gamma(T)$ is composed of a public key for each building-block scheme $\Pi \in \mathfrak{S}[T]$ and a number of ciphertexts for these schemes.

As usual, each circuit C in the computational model is an acyclic data flow. Each node is given some objects from its incoming edges as its local inputs (or a part of the input for C, if it is an input (source) node), computes its local output (if it is either an input node or the unique output (sink) node, then there is no local computation), and then sends its copies to the outgoing edges (or it is the output of C, if the node is the output node).

In the model, the possible data types of the objects are *bit* and *ciphertext*; the latter is further classified into Π-*ciphertext* for each $\Pi \in \mathfrak{S}$. Each object of Π-ciphertext-type is a *black-box* object, which can only be generated or modified via internal nodes corresponding to the functionalities of the building-block scheme Π, and only be viewed by internal nodes corresponding to the ZPA oracles for the inner schemes (see below for the details). We emphasize that *plaintexts are represented by collections of bits*, and *any* (efficient) operation on plaintexts, which may be non-algebraic, is allowed in the model. (On the other hand, the public key PK_Π for each $\Pi \in \mathfrak{S}$ involved in each challenge in the ZPA game is made implicit for simplifying the description.) Each edge of a circuit is assigned one of the data types, and it can carry the corresponding kind of objects only.

In a circuit C in the model, each input node is given either a ciphertext for some building-block scheme Π (which is a component of a challenge for the outer scheme) or a uniformly random bit (which represents the internal randomness for C). On the other hand, there exists a unique output node and it has a unique incoming edge, which is of bit-type (i.e., the output of C is a single bit). Moreover, the types of the internal nodes are one of the followings, where $\Pi \in \mathfrak{S}$:

$\mathrm{Enc}_\Pi(m; r)$, $\mathrm{Add}_\Pi(c, c')$, $\mathrm{Mult}_\Pi(m, c')$, $\mathrm{Rerand}_\Pi(c; r)$: Here m is a plaintext (expressed by a bit sequence);[9] c and c' are ciphertexts for Π; and r is a bit sequence. The output is of Π-ciphertext-type and it is the same as the corresponding algorithms for Π with public key PK_Π, where r (if it exists) is used as the internal randomness of the algorithm.

$\mathrm{AND}(b, b')$, $\mathrm{OR}(b, b')$, $\mathrm{NOT}(b)$: These nodes behave as the ordinary bit operations.

$\mathrm{Switch}_\Pi(c, c'; b)$: For two ciphertexts c, c' for Π, the output is c if $b = 0$, and c' if $b = 1$.

[9] For a technical reason, if the bit sequence does not represent a valid plaintext, then the outputs of Enc_Π and Mult_Π nodes are defined to be a random ciphertext of a uniformly random plaintext.

$\mathcal{O}_{T'}(\vec{c})$**:** The input $\vec{c}$ is a collection of ciphertexts for some schemes $\Pi \in \mathfrak{S}$, which (together with the public keys PK_Π for these Π) forms a challenge for an inner scheme $\Gamma(T')$. This node represents an oracle (outputting a bit) that breaks the ZPA security for $\Gamma(T')$.

Remark 2. When Π is the Paillier cryptosystem, multiplication of $h_1, h_2 \in G := (\mathbb{Z}/n^2\mathbb{Z})^\times$, inverse of $h \in G$ and random sampling on G can be computed in our model by $\mathrm{Add}_\Pi(h_1, h_2)$, $\mathrm{Mult}_\Pi(-1, h)$ and $\mathrm{Enc}_\Pi(m; r)$ with uniformly random m and r. This suggests that our model has reasonably strong functionality comparable to the generic group model on G, hence it is worthy to study the relations between the security of our proposed schemes on the model (note that the results in Section 5.2 can indeed be described within our model).

7 Main Result: Security *Non*-Implications

In this section, we study non-implication relations between the ZPA security for our RMHE schemes $\Gamma(T)$ with different trees T. Here we assume that the building-block schemes $\Pi \in \mathfrak{S}$ satisfy the following two technical conditions on the plaintext spaces; we note that these are indeed satisfied by the Paillier cryptosystem (unless it is totally broken):

Assumption 1 The ratio $|\mathcal{M}^\times|/|\mathcal{M}|$ is overwhelming (hence $1/|\mathcal{M}|$ is negligible).

Assumption 2 It is computationally hard to find an element of $\mathcal{M} \setminus (\mathcal{M}^\times \cup \{0\})$.[10]

Then we have the following, which is the main theorem of this paper:

Theorem 4. *Let T^{out} be any tree. Under Assumptions 1 and 2, there are no polynomial-time constructible circuit families $C = (C_k)_{k \geq 1}$ in the model in Section 6 with the following property: If the (not necessarily polynomial-time computable) oracle $\mathcal{O}_{T^{\mathrm{in}}}$ has a non-negligible advantage as a ZPA adversary for $\Gamma(T^{\mathrm{in}})$ for* every *tree T^{in} of depth one with $T^{\mathrm{out}} \not\preceq T^{\mathrm{in}}$, then C is also a ZPA adversary for $\Gamma(T^{\mathrm{out}})$ with non-negligible advantage.*

Intuitively, Theorem 4 says that, even if the ZPA security of $\Gamma(T^{\mathrm{in}})$ for *all* trees T^{in} as in the statement are broken, it *cannot* be efficiently converted (by a "natural" reduction algorithm described in our model in Section 6) to an adversary that breaks $\Gamma(T^{\mathrm{out}})$.

Now we start to describe the proof of Theorem 4. The proof is divided into several steps as shown in the following subsections.

7.1 Restriction of Possibilities of the Outer Scheme

At the beginning of the proof, we show that, to prove the theorem, it is sufficient to consider the cases $T^{\mathrm{out}} = T_\ell$ ($\ell \geq 1$) and $T^{\mathrm{out}} = T^\S$ (see Examples 1 and 2 in Section 5.1 for the definitions of T_ℓ and $T^\S$).

We suppose that, for a tree T', we have $T' \preceq T^{\mathrm{out}}$ and $T' \not\preceq T^{\mathrm{in}}$ for any depth-one tree T^{in} with $T^{\mathrm{out}} \not\preceq T^{\mathrm{in}}$. Now the set of depth-one trees T^{in} with $T^{\mathrm{out}} \not\preceq T^{\mathrm{in}}$ is not changed when T^{out} is replaced with T'. In this case, Theorem 4 for the tree T^{out} is implied by that for the tree T'; if a circuit family C as in the statement exists for the case of T^{out}, then by Theorem 1, it can be efficiently converted to another circuit family as in the statement for the case of T'. From now, we show that $T' = T_\ell$ or $T' = T^\S$ indeed satisfies the condition.

[10] For the case of the Paillier cryptosystem with $\mathcal{M} = \mathbb{Z}/n\mathbb{Z}$, since $\mathcal{M} \setminus \mathcal{M}^\times = \{a \in \mathcal{M} \mid \gcd(a, n) > 1\}$, any element of $\mathcal{M} \setminus (\mathcal{M}^\times \cup \{0\})$ yields the factorization of n, which reveals the secret key.

First we note that the condition above is preserved by applying the transformations $\Phi = \mathrm{Divide}_{v,v'}^{-1}$ and $\Phi = \mathrm{Grow}^{-1}$, since now $T' \preceq \Phi(T') \preceq T'$ by the definition of $\preceq$. Hence, by applying these transformations in advance, we assume without loss of generality that these transformations cannot be applied to T^{out}, that is, every non-leaf vertex of T^{out} has at least two child vertices. If T^{out} is a trivial tree, then we may consider $T_1 = \mathrm{Grow}(T^{\mathrm{out}})$ instead of T^{out}, since $T^{\mathrm{out}} \preceq \mathrm{Grow}(T^{\mathrm{out}}) \preceq T^{\mathrm{out}}$. Now T^{out} is of the desired form T_ℓ if it has depth one; from now, we consider the other case that T^{out} has depth at least two.

By the condition on the depth, T^{out} has at least one non-leaf vertex of depth one, say v'_2. By the condition above, v'_2 has at least two child vertices, say v'_3 and v'_4. On the other hand, the root of T^{out} also has at least two child vertices; let v'_1 be its child vertex other than v'_2. Now by the definition of $\preceq$, we have $T^{\S} \preceq T^{\mathrm{out}}$, where for each leaf v_i of $T^{\S}$ with $i \in \{1,3,4\}$, one of the building-block schemes associated to some leaf of $(T^{\mathrm{out}})_{v'_i}$ is associated to v_i. Moreover, we have $T^{\S} \not\preceq T^{\mathrm{in}}$ for any depth-one tree T^{in} by the definition of $\preceq$. This shows that $T' = T^{\S}$ indeed satisfies the condition in this case.

Hence we have shown that, to prove Theorem 4, we may assume without loss of generality that $T^{\mathrm{out}} = T_\ell$ or $T^{\mathrm{out}} = T^{\S}$. We use the notations in Examples 1 and 2 for T_ℓ and $T^{\S}$, and we often abbreviate v_j to j.

7.2 Construction of the Oracles

From now, we assume that a circuit family $C = (C_k)_{k \geq 1}$ as in the statement exists, and deduce a contradiction. First, we determine the oracles $\mathcal{O}_{T^{\mathrm{in}}}$ involved in C concretely.

When $T^{\mathrm{in}} = T_{\ell'}$ ($\ell' \geq 1$), we write $\Pi_j := \Pi(v_j)$ for $1 \leq j \leq \ell'$ if it is not confusing. Then a challenge in the ZPA game for $\Gamma(T_{\ell'})$ consists of a public key pk and a challenge ciphertext $c^* = (c^*_1, \dots, c^*_{\ell'})$, where pk consists of a public key PK_{Π_j} and a ciphertext $\mathsf{PK}_{r \to j} \in \mathcal{C}_{\Pi_j, \mathsf{SK}_{r \to j}}$ for $1 \leq j \leq \ell'$, and $c^*_j \in \mathcal{C}_{\Pi_j, s_j \cdot \mathsf{SK}_{r \to j}}$ for each $1 \leq j \leq \ell'$, where $(s_j)_{j=1}^{\ell'}$ is a share set of the challenge plaintext m_{b^*}. Now we have

$$m_{b^*} = \sum_{j=1}^{\ell'} \frac{s_j \cdot \mathsf{SK}_{r \to j}}{\mathsf{SK}_{r \to j}} = \frac{\sum_{j=1}^{\ell'} \left(s_j \cdot \mathsf{SK}_{r \to j} \cdot \prod_{1 \leq j' \leq \ell', j' \neq j} \mathsf{SK}_{r \to j'} \right)}{\mathsf{SK}_{r \to 1} \cdots \mathsf{SK}_{r \to \ell'}}$$

$$= \frac{\sum_{j=1}^{\ell'} \left(\mathsf{Dec}_{\Pi_j}(c^*_j) \cdot \prod_{1 \leq j' \leq \ell', j' \neq j} \mathsf{Dec}_{\Pi_{j'}}(\mathsf{PK}_{r \to j'}) \right)}{\mathsf{SK}_{r \to 1} \cdots \mathsf{SK}_{r \to \ell'}} \quad ,$$

therefore we have $m_{b^*} = 0$ (which is equivalent to $b^* = 0$ except a negligible probability[11]) if and only if the numerator of the right-hand side is zero.

Based on the observation, we introduce the following polynomial in the variables $Z_1, \dots, Z_{\ell'}$ and $Z'_1, \dots, Z'_{\ell'}$:

$$F_{\ell'} := \sum_{j=1}^{\ell'} \left(Z_j \cdot Z'_1 \cdots Z'_{j-1} Z'_{j+1} \cdots Z'_{\ell'} \right) \quad .$$

Then we have $m_{b^*} = 0$ if and only if the value $F_{\ell'}(\mathsf{Dec}_{\Pi_j}(c^*_j); \mathsf{Dec}_{\Pi_j}(\mathsf{PK}_{r \to j}))$ of $F_{\ell'}$ given by substituting $\mathsf{Dec}_{\Pi_j}(c^*_j)$ to Z_j and $\mathsf{Dec}_{\Pi_j}(\mathsf{PK}_{r \to j})$ to Z'_j for each $1 \leq j \leq \ell'$ is equal to zero. Now we define the oracle $\mathcal{O}_{T^{\mathrm{in}}}$ as follows: $\mathcal{O}_{T^{\mathrm{in}}}$ outputs 0 if $F_{\ell'}(\mathsf{Dec}_{\Pi_j}(c^*_j); \mathsf{Dec}_{\Pi_j}(\mathsf{PK}_{r \to j})) = 0$, and outputs 1 otherwise. Then by the argument above, $\mathcal{O}_{T^{\mathrm{in}}}$ has a non-negligible advantage as a ZPA adversary for $\Gamma(T^{\mathrm{in}})$. Note that $\mathcal{O}_{T^{\mathrm{in}}}$ is in general not polynomial-time

[11] Note that $\Pr[m_{b^*} = 0 \mid b^* = 1] = 1/|\mathcal{M}|$, which is negligible by Assumption 1.

computable (i.e., the computation of the values of $\mathsf{Dec}_{\Pi_j}(c_j^*)$ and $\mathsf{Dec}_{\Pi_j}(\mathsf{PK}_{r\to j})$ would need brute-force attacks), but it is indeed allowed in the statement of the theorem. Hence, this oracle $\mathcal{O}_{T^{\mathrm{in}}}$ indeed satisfies the desired condition.

7.3 Overall Strategy: Hybrid Argument

Here we summarize the overall strategy for the remaining proof. First, for each security parameter k, we choose an ordering of the nodes in the circuit C_k with the property that there are no paths in C_k from a node appearing later to a node appearing earlier (with respect to the ordering). Note that this can be done in polynomial time, since C itself is polynomial-time constructible (in particular, the number of nodes in C_k is polynomially bounded). Let ρ denote the total number of the oracle nodes $\mathcal{O}_{T^{\mathrm{in}}}$ in C_k; hence ρ is polynomially bounded as well.

From now, we construct a sequence of circuits $C_k^0 := C_k, C_k^1, \dots, C_k^\rho$ recursively, in the following manner. To construct $C_k^{\rho'}$ for each $\rho' = 1, 2, \dots, \rho$, we modify the previous circuit $C_k^{\rho'-1}$ by replacing the ρ'-th oracle node $\mathcal{O}_{T^{\mathrm{in}}}$ with another node $\mathcal{O}'_{T^{\mathrm{in}}}$ determined later. By the definition, the input distribution for the $\mathcal{O}_{T^{\mathrm{in}}}$ in $C_k^{\rho'-1}$ is identical to that for the $\mathcal{O}'_{T^{\mathrm{in}}}$ in $C_k^{\rho'}$; we construct $\mathcal{O}'_{T^{\mathrm{in}}}$ in such a way that *the output distribution of the $\mathcal{O}'_{T^{\mathrm{in}}}$ is statistically close to that of the $\mathcal{O}_{T^{\mathrm{in}}}$*. This implies that the output distributions of $C_k^{\rho'-1}$ and $C_k^{\rho'}$ are also statistically close; hence, since ρ is polynomially bounded, the output distributions of $C_k = C_k^0$ and C_k^ρ are statistically close as well. We also show that *the output of C_k^ρ is independent of the challenge bit b^* in the ZPA game for* $\Gamma(T^{\mathrm{out}})$; this implies that C_k^ρ has zero advantage as a ZPA adversary for $\Gamma(T^{\mathrm{out}})$, therefore the advantage of C_k is negligible by the argument above. This is a contradiction, which will complete the proof of Theorem 4. This is the outline of our proof.

In the remaining part of the proof, we sometimes associate a superscript "in" or "out" to an object related to the inner schemes $\Gamma(T^{\mathrm{in}})$ or the outer scheme $\Gamma(T^{\mathrm{out}})$, respectively, when we want to clarify which of them the object is associated.

7.4 Expressions of Plaintexts for the Ciphertexts

Before constructing $\mathcal{O}'_{T^{\mathrm{in}}}$ mentioned above, we give some preliminary argument on the behaviors of plaintexts corresponding to the ciphertexts in the circuit C_k.

Let $\mathcal{X}$ denote the set of plaintexts appearing in the whole construction of the challenge in the ZPA game for $\Gamma(T^{\mathrm{out}})$ which is the input for C_k (see below for examples). Then for each ciphertext for some building-block scheme which is a component of the input for C_k, denoted here by γ^{out}, the plaintext for γ^{out} is a polynomial in elements of $\mathcal{X}$, denoted by $\mathcal{F}[\gamma^{\mathrm{out}}]$. More precisely, we have the following:

- For the case $T^{\mathrm{out}} = T_{\ell^{\mathrm{out}}}$ (see Example 1 for the notations for the tree T_ℓ), $\mathcal{X}$ consists of $\mathsf{SK}_{r\to j}$ for $1 \le j \le \ell^{\mathrm{out}}$, s_j for $1 \le j \le \ell^{\mathrm{out}} - 1$, and m_{b^*}. We have

$$\mathcal{F}[\mathsf{PK}^{\mathrm{out}}_{r\to j}] = \mathsf{SK}_{r\to j} \text{ for } 1 \le j \le \ell^{\mathrm{out}} \enspace ,$$

$$\mathcal{F}[c_j^{*\mathrm{out}}] = s_j \cdot \mathsf{SK}_{r\to j} \text{ for } 1 \le j \le \ell^{\mathrm{out}} - 1 \enspace ,$$

$$\mathcal{F}[c_\ell^{*\mathrm{out}}] = (m_{b^*} - \sum_{j=1}^{\ell^{\mathrm{out}}-1} s_j) \cdot \mathsf{SK}_{r\to \ell^{\mathrm{out}}} \enspace .$$

- For the case $T^{\mathrm{out}} = T^{\S}$ (see Example 2 for the notations for the tree $T^{\S}$), $\mathcal{X}$ consists of $\mathsf{SK}_{r\to 1}$, $\mathsf{SK}_{r\to 2}$, $\mathsf{SK}_{2\to 3}$, $\mathsf{SK}_{2\to 4}$, s_{pk}, s_1, s_2 and m_{b^*}. We have

$$\mathcal{F}[\mathsf{PK}^{\mathrm{out}}_{r\to 1}] = \mathsf{SK}_{r\to 1} \ ,$$

$$\mathcal{F}[\mathsf{PK}^{\mathrm{out}(1)}_{r\to 2}] = s_{\mathrm{pk}} \cdot \mathsf{SK}_{2\to 3} \ ,$$

$$\mathcal{F}[\mathsf{PK}^{\mathrm{out}(2)}_{r\to 2}] = (\mathsf{SK}_{r\to 2} - s_{\mathrm{pk}}) \cdot \mathsf{SK}_{2\to 4} \ ,$$

$$\mathcal{F}[\mathsf{PK}^{\mathrm{out}}_{2\to 3}] = \mathsf{SK}_{2\to 3} \ ,$$

$$\mathcal{F}[\mathsf{PK}^{\mathrm{out}}_{2\to 4}] = \mathsf{SK}_{2\to 4} \ ,$$

$$\mathcal{F}[c_1^{*\mathrm{out}}] = s_1 \cdot \mathsf{SK}_{r\to 1} \ ,$$

$$\mathcal{F}[c_3^{*\mathrm{out}}] = ((m_{b^*} - s_1)s_{\mathrm{pk}} + s_2) \cdot \mathsf{SK}_{2\to 3} \ ,$$

$$\mathcal{F}[c_4^{*\mathrm{out}}] = ((m_{b^*} - s_1)(\mathsf{SK}_{r\to 2} - s_{\mathrm{pk}}) - s_2) \cdot \mathsf{SK}_{2\to 4} \ .$$

From now, we associate to each edge of C_k of ciphertext-type a collection $\vec{m} = (m[\widetilde{\gamma}^{\mathrm{out}}])_{\widetilde{\gamma}^{\mathrm{out}}}$ of elements $m[\widetilde{\gamma}^{\mathrm{out}}] \in \mathcal{M}$, which we call a *coefficient vector*. Here, the index $\widetilde{\gamma}^{\mathrm{out}}$ is either a ciphertext component γ^{out} of the input for C_k as above, or a symbol "const" (which means "constant term"). Given a coefficient vector $\vec{m}$, we define an element $\vec{m} \cdot \mathcal{F}$ of $\mathcal{M}$ by

$$\vec{m} \cdot \mathcal{F} := \sum_{\widetilde{\gamma}^{\mathrm{out}}} m[\widetilde{\gamma}^{\mathrm{out}}] \cdot \mathcal{F}[\widetilde{\gamma}^{\mathrm{out}}] = \sum_{\gamma^{\mathrm{out}}} m[\gamma^{\mathrm{out}}] \cdot \mathcal{F}[\gamma^{\mathrm{out}}] + m[\mathsf{const}] \ ,$$

where we set $\mathcal{F}[\mathsf{const}] := 1$ for simplifying the notation. We define the coefficient vectors recursively in such a way that the following holds:

Lemma 3. *Let $\vec{m}$ be a coefficient vector associated to an edge of C_k of type* ciphertext*. Then the plaintext corresponding to the ciphertext carried by the edge is equal to $\vec{m} \cdot \mathcal{F}$. Moreover, if it is a ciphertext for buliding-block scheme Π, then we have $m[\gamma^{\mathrm{out}}] = 0$ for any component γ^{out} of the input for C_k which is a ciphertext for a building-block scheme other than Π.*

To define the coefficient vectors $\vec{m}$ recursively, first, if the edge is an outgoing edge of an input node corresponding to a component γ^{out} of the input for C_k, then we define $\vec{m}$ by $m[\gamma^{\mathrm{out}}] := 1$ and $m[\widetilde{\gamma}^{\mathrm{out}}] := 0$ for any $\widetilde{\gamma}^{\mathrm{out}} \neq \gamma^{\mathrm{out}}$. Then we have $\vec{m} \cdot \mathcal{F} = \mathcal{F}[\gamma^{\mathrm{out}}]$, therefore Lemma 3 holds for this case by the definition of $\mathcal{F}[\gamma^{\mathrm{out}}]$. For the remaining cases, the definition is as follows:

- For the case that the edge is an outgoing edge of an Enc_{Π} node with input $(m'; r)$, we define $\vec{m}$ by $m[\mathsf{const}] := m'$ and $m[\gamma^{\mathrm{out}}] := 0$ for any γ^{out}.[12] Then we have $\vec{m} \cdot \mathcal{F} = m'$, therefore Lemma 3 holds for this case.

[12]As mentioned in Section 6, if the bit sequence m' does not represent a correct plaintext, then the output of the node is defined to be a random ciphertext of a uniformly random plaintext. Accordingly, we define $\vec{m}$ in this case by $m[\mathsf{const}] \leftarrow_R \mathcal{M}$ and $m[\gamma^{\mathrm{out}}] := 0$ for any γ^{out}; now Lemma 3 holds for this case.

- For the case that the edge is an outgoing edge of an Add_Π node with input (c', c''), suppose that coefficient vectors $\vec{m}'$ and $\vec{m}''$ are associated to the incoming edges for the two input components c' and c'', respectively, and Lemma 3 holds for these incoming edges. Then we define $\vec{m}$ to be the component-wise addition of $\vec{m}'$ and $\vec{m}''$. Now we have $\vec{m} \cdot \mathcal{F} = \vec{m}' \cdot \mathcal{F} + \vec{m}'' \cdot \mathcal{F}$, while c' and c'' are ciphertexts of plaintexts $\vec{m}' \cdot \mathcal{F}$ and $\vec{m}'' \cdot \mathcal{F}$, respectively, by the choice of $\vec{m}'$ and $\vec{m}''$. Hence Lemma 3 holds for this case (note that the latter part of the claim is also satisfied, since c' and c'' are also ciphertexts for the same scheme Π).

- For the case that the edge is an outgoing edge of a Mult_Π node with input (m', c''), suppose that a coefficient vector $\vec{m}''$ is associated to the incoming edge for the input component c'', and Lemma 3 holds for this incoming edge. Then we define $\vec{m}$ to be the scalar multiplication $m' \cdot \vec{m}''$ to the vector $\vec{m}''$ by m'.[13] Now we have $\vec{m} \cdot \mathcal{F} = m' \cdot (\vec{m}'' \cdot \mathcal{F})$, while c'' is a ciphertext of plaintext $\vec{m}'' \cdot \mathcal{F}$ by the choice of $\vec{m}''$. Hence Lemma 3 holds for this case (note that the latter part of the claim is also satisfied, since c' is also a ciphertext for the same scheme Π).

- For the case that the edge is an outgoing edge of a Rerand_Π node with input $(c'; r)$, suppose that a coefficient vector $\vec{m}'$ is associated to the incoming edge for the input component c', and Lemma 3 holds for this incoming edge. Then we define $\vec{m}$ by $\vec{m} := \vec{m}'$; now Lemma 3 holds for this case, since the algorithm Rerand_Π does not change the plaintext corresponding to the ciphertext.

- For the case that the edge is an outgoing edge of a Switch_Π node with input $(c', c''; b)$, suppose that coefficient vectors $\vec{m}'$ and $\vec{m}''$ are associated to the incoming edges for the two input components c' and c'', respectively, and Lemma 3 holds for these incoming edges. Then we define $\vec{m}$ by $\vec{m} := \vec{m}'$ if $b = 0$ and $\vec{m} := \vec{m}''$ if $b = 1$; now Lemma 3 holds for this case.

Summarizing the arguments above, it follows that Lemma 3 holds. We also note that the overhead of the computational cost to calculate the coefficient vectors for all those edges, in addition to the original execution of the circuit C_k, is polynomially bounded, since the process above to determine the coefficient vector for a new edge is efficient.

7.5 Definition of the Auxiliary Oracles

To proceed the recursive construction of $C_k^{\rho'}$ for $\rho' = 1, 2, \dots, \rho$, we describe the construction of the new oracle $\mathcal{O}'^{(\rho')} = \mathcal{O}'_{T^{\mathrm{in}}}$ which replaces the ρ'-th oracle node $\mathcal{O}_{T^{\mathrm{in}}}$ in $C_k^{\rho'-1}$. We construct these oracles $\mathcal{O}'_{T^{\mathrm{in}}}$ in such a way that the following holds:

Lemma 4. *In the circuit C_k^{ρ}, the coefficient vector associated to each edge of ciphertext-type is independent of the values of the plaintexts in the set $\mathcal{X}$ defined above, and the bit carried by each edge of bit-type is independent of the values of the plaintexts in $\mathcal{X}$.*

Once Lemma 4 is proven, the bit carried by the incoming edge of the output node, which is the output of C_k^{ρ}, is independent of the values of the plaintexts in $\mathcal{X}$; in particular, it is independent of the challenge plaintext m_{b^*}. This implies that the output of C_k^{ρ} is independent of the challenge bit b^* in the ZPA game for $\Gamma(T^{\mathrm{out}})$, as desired.

[13] By the same reason as the case of Enc_Π node, if the bit sequence m' does not represent a correct plaintext, then we define $\vec{m}$ by $m[\mathsf{const}] \leftarrow_R \mathcal{M}$ and $m[\gamma^{\mathrm{out}}] := 0$ for any γ^{out}.

To prove Lemma 4, first we note that for each node of C_k^ρ other than the oracle nodes, if the claim of Lemma 4 holds for all the incoming edges, then the claim also holds for the outgoing edges by the definitions of the nodes and the coefficient vectors. Therefore, it suffices to show the following property:

Lemma 5. *For each oracle node $\mathcal{O}'_{T^{\mathrm{in}}}$ in C_k^ρ, if the claim of Lemma 4 holds for any outgoing edge of every node in C_k^ρ which precedes $\mathcal{O}'_{T^{\mathrm{in}}}$ in C_k^ρ (with respect to the ordering of nodes specified in Section 7.3), then the claim of Lemma 4 also holds for any outgoing edge of $\mathcal{O}'_{T^{\mathrm{in}}}$.*

First we note that, by the choice of the ordering of nodes mentioned in Lemma 5, any incoming edge for the node $\mathcal{O}'_{T^{\mathrm{in}}}$ is an outgoing edge of a node which precedes $\mathcal{O}'_{T^{\mathrm{in}}}$ with respect to the ordering of nodes. Therefore, in the situation of Lemma 5, the coefficient vector associated to each incoming edge for the node $\mathcal{O}'_{T^{\mathrm{in}}}$ is independent of the values of the plaintexts in $\mathcal{X}$. Moreover, if $\mathcal{O}'_{T^{\mathrm{in}}}$ is the ρ'-th oracle node ($1 \leq \rho' \leq \rho$), then the input distribution for $\mathcal{O}'_{T^{\mathrm{in}}}$ in C_k^ρ is identical to that in $C_k^{\rho'}$. From now, we focus on the circuit $C_k^{\rho'-1}$ and construct the oracle $\mathcal{O}'_{T^{\mathrm{in}}}$ which replaces the ρ'-th oracle node $\mathcal{O}_{T^{\mathrm{in}}}$ in $C_k^{\rho'-1}$. Let $\ell^{\mathrm{in}} \geq 1$ denote the number of leaves of T^{in}; $T^{\mathrm{in}} = T_{\ell^{\mathrm{in}}}$.

In the setting, $\mathcal{O}_{T^{\mathrm{in}}}$ is a ZPA adversary for $\Gamma(T_{\ell^{\mathrm{in}}})$; let $c_j^{*\mathrm{in}}$ and $\mathsf{PK}^{\mathrm{in}}_{r \to j}$ ($1 \leq j \leq \ell^{\mathrm{in}}$) denote the components of the input for $\mathcal{O}_{T^{\mathrm{in}}}$ mentioned in Section 7.2. For $\gamma^{\mathrm{in}} = c_j^{*\mathrm{in}}$ and $\gamma^{\mathrm{in}} = \mathsf{PK}^{\mathrm{in}}_{r \to j}$, let $\vec{m}[\gamma^{\mathrm{in}}] = (m[\gamma^{\mathrm{in}}; \widetilde{\gamma}^{\mathrm{out}}])_{\widetilde{\gamma}^{\mathrm{out}}}$ denote the coefficient vector associated to the incoming edge corresponding to the input component γ^{in} for $\mathcal{O}_{T^{\mathrm{in}}}$. Then by Lemma 3, the plaintext for the ciphertext γ^{in} is $\vec{m}[\gamma^{\mathrm{in}}] \cdot \mathcal{F}$. Therefore, $\mathcal{O}_{T^{\mathrm{in}}}$ outputs 0 if the value of the polynomial Φ in the elements of $\mathcal{X}$ defined by

$$\Phi := F_{\ell^{\mathrm{in}}}(\vec{m}[c_j^{*\mathrm{in}}] \cdot \mathcal{F}; \vec{m}[\mathsf{PK}^{\mathrm{in}}_{r \to j}] \cdot \mathcal{F})$$

(see Section 7.2 for the definition of the polynomial $F_{\ell^{\mathrm{in}}}$) becomes 0 when the actual values of plaintexts in $\mathcal{X}$ are substituted, and $\mathcal{O}_{T^{\mathrm{in}}}$ outputs 1 otherwise.

We define $\mathcal{O}'_{T^{\mathrm{in}}}$ in such a way that it outputs 0 if $\Phi = 0$ *as a polynomial* (i.e., all the coefficients of Φ are zero), and outputs 1 otherwise. Now note that the coefficients of Φ are determined from the coefficient vectors $\vec{m}[\gamma^{\mathrm{in}}]$. Hence, by the argument above, *the coefficients of* Φ *are independent on the values of plaintexts in* $\mathcal{X}$; therefore *the output of* $\mathcal{O}'_{T^{\mathrm{in}}}$ *is also independent of the values of plaintexts in* $\mathcal{X}$. This proves Lemma 5, which implies as mentioned above that the output of the circuit C_k^ρ is independent of the challenge bit b^* in the ZPA game for $\Gamma(T^{\mathrm{out}})$. We also note that the polynomial Φ can be computed in polynomial time; indeed, each coefficient vector is efficiently computable, while Φ involves a constant number (depending solely on T^{out}) of variables and has a polynomially bounded degree (since ℓ^{in} is polynomially bounded, owing to the property that C_k is polynomial-time constructible), therefore Φ has only a polynomially many terms. Hence, $\mathcal{O}'_{T^{\mathrm{in}}}$ can be computed in polynomial time.

7.6 Evaluation of Statistical Distances: Preliminaries

By the results above, our remaining task is to show that the statistical distance between the output distributions of $\mathcal{O}_{T^{\mathrm{in}}}$ and $\mathcal{O}'_{T^{\mathrm{in}}}$ is bounded by a negligible value, which is common for all $\mathcal{O}_{T^{\mathrm{in}}}$. First we note that, if $\mathcal{O}'_{T^{\mathrm{in}}}$ outputs 0 (i.e., $\Phi = 0$ as a polynomial) for a given input, then $\mathcal{O}_{T^{\mathrm{in}}}$ also outputs 0 (i.e., the value of Φ becomes zero when the actual values of plaintexts in $\mathcal{X}$ is substituted) for the same input. Therefore, it suffices to bound the probability that $\mathcal{O}'_{T^{\mathrm{in}}}$ outputs 1 but $\mathcal{O}_{T^{\mathrm{in}}}$ outputs 0, i.e., Φ is a non-zero polynomial but its value becomes zero.

We note that, by the definition of $\Gamma(T^{\mathrm{out}})$, the value of each plaintext in $\mathcal{X}$ is chosen uniformly at random from either $\mathcal{M}$ or $\mathcal{M}^{\times}$. Now, to bound the probability above, we present the following lemma:

Lemma 6. *Let $g(t_1, \dots, t_n)$ be any non-zero polynomial of degree $\deg(g)$ over the ring $\mathcal{M}$ in n variables $t_1, \dots, t_n$. Then the number of zeroes $\vec{a} = (a_1, \dots, a_n)$ of g, i.e., $\vec{a} \in \mathcal{M}^n$ with $g(a_1, \dots, a_n) = 0$, is at most $n \cdot \deg(g)|\mathcal{M}|^{n-1}(|\mathcal{M}| - |\mathcal{M}^{\times}|)$.*

Proof. First we consider the case $n = 1$. By the polynomial remainder theorem, we can decompose $g(t)$ as $g(t) = (t - a_1) \cdots (t - a_d)h(t)$ in such a way that $0 \le d \le \deg(g)$, $a_1, \dots, a_d \in \mathcal{M}$ and the non-zero polynomial $h(t)$ over $\mathcal{M}$ has no zeroes in $\mathcal{M}$. Then for each zero $b \in \mathcal{M}$ of g, we have $(b - a_1) \cdots (b - a_d)h(b) = 0$, while $h(b) \ne 0$ by the choice of h. Therefore, at least one of $b - a_i$ is not invertible in $\mathcal{M}$. This implies that the number of such b is at most $d(|\mathcal{M}| - |\mathcal{M}^{\times}|) \le \deg(g)(|\mathcal{M}| - |\mathcal{M}^{\times}|)$, as desired.

From now, we consider the case $n \ge 2$. We focus on the variable t_n and take the coefficient $g^{\dagger}$ of the highest power of t_n in g, therefore $g^{\dagger}$ is a non-zero polynomial over $\mathcal{M}$ in variables $t_1, \dots, t_{n-1}$ having degree at most $\deg(g)$. By induction on n, the number of $\vec{a} \in \mathcal{M}^{n-1}$ satisfying $g^{\dagger}(a_1, \dots, a_{n-1}) = 0$ is at most $\deg(g)(n-1)|\mathcal{M}|^{n-2}(|\mathcal{M}| - |\mathcal{M}^{\times}|)$. On the other hand, for each $(a_1, \dots, a_{n-1}) \in \mathcal{M}^{n-1}$ with $g^{\dagger}(a_1, \dots, a_{n-1}) \ne 0$, the non-zero polynomial $g(a_1, \dots, a_{n-1}, t_n)$ over $\mathcal{M}$ in a single variable t_n of degree at most $\deg(g)$ has at most $\deg(g)(|\mathcal{M}| - |\mathcal{M}^{\times}|)$ zeroes. Therefore, the number of zeroes $\vec{a} \in \mathcal{M}^n$ of g satisfying $g^{\dagger}(a_1, \dots, a_{n-1}) \ne 0$ is at most $|\mathcal{M}|^{n-1} \cdot \deg(g)(|\mathcal{M}| - |\mathcal{M}^{\times}|)$. Hence the number of zeroes of g is at most

$$\begin{aligned} &|\mathcal{M}| \cdot \deg(g)(n-1)|\mathcal{M}|^{n-2}(|\mathcal{M}| - |\mathcal{M}^{\times}|) + |\mathcal{M}|^{n-1} \cdot \deg(g)(|\mathcal{M}| - |\mathcal{M}^{\times}|) \\ &\qquad\qquad \le \deg(g) \cdot n \cdot |\mathcal{M}|^{n-1}(|\mathcal{M}| - |\mathcal{M}^{\times}|) \ , \end{aligned}$$

concluding the proof of Lemma 6. □

By Lemma 6, when a uniformly random element of either $\mathcal{M}$ or $\mathcal{M}^{\times}$ is substituted into each variable t_j of a polynomial g in the statement of Lemma 6, the probability that the value of g becomes zero is not larger than

$$\begin{aligned} \frac{n \cdot \deg(g)|\mathcal{M}|^{n-1}(|\mathcal{M}| - |\mathcal{M}^{\times}|)}{|\mathcal{M}^{\times}|^n} &= n \cdot \deg(g) \left(\frac{|\mathcal{M}|}{|\mathcal{M}^{\times}|} \right)^{n-1} \left(\frac{|\mathcal{M}|}{|\mathcal{M}^{\times}|} - 1 \right) \\ &= n \cdot \deg(g) \frac{1 - |\mathcal{M}^{\times}|/|\mathcal{M}|}{(1 - (1 - |\mathcal{M}^{\times}|/|\mathcal{M}|))^n} \ , \end{aligned}$$

which is negligible if both n and $\deg(g)$ are polynomially bounded (since $1 - |\mathcal{M}^{\times}|/|\mathcal{M}|$ is negligible by Assumption 1).

Now note that the number of variables in Φ (i.e., $|\mathcal{X}|$) and the degree of Φ are both polynomially bounded, since C_k is polynomial-time constructible. Then by the previous paragraph, in the case of the challenge bit $b^* = 1$ (i.e., m_{b^*} is uniformly random), if $\Phi \ne 0$ as a polynomial, then the probability that its value becomes zero is negligible.[14] On the other hand, in the other case $b^* = 0$ (i.e., $m_{b^*} = 0$), if the polynomial $\Phi' := \Phi|_{m_{b^*}=0}$ is non-zero, then the probability that its value becomes zero is negligible as well. Therefore, the remaining task is to show the following: *The probability that $\Phi \ne 0$ but $\Phi' = \Phi|_{m_{b^*}=0} = 0$ as polynomials is negligible.*

[14] Here we used the fact that the coefficients of Φ are independent of the values of plaintexts in $\mathcal{X}$.

7.7 Properties of Polynomials for Plaintexts

To evaluate the probability specified above, here we investigate some properties of the polynomials $\mathcal{F}[\gamma^{\mathrm{out}}]$ introduced in Section 7.4, which are used in the construction of Φ.

Here we note that the study of polynomials over $\mathcal{M}$ *for the case that* $\mathcal{M}$ *is a field* is much easier than the general case, mainly due to the fact that the polynomial ring (with a finite number of variables) over a field is a unique factorization domain (UFD), hence any irreducible polynomial g is also a prime polynomial, i.e., if g divides the product $f_1 f_2$ of two polynomials, then g also divides one of f_1 and f_2 (see e.g., [5]). In order to reduce the argument in the general case to the special case that $\mathcal{M}$ is a field, we fix a maximal ideal $\mathfrak{m}$ of the (finite commutative) ring $\mathcal{M}$ and let $\mathbb{F} := \mathcal{M}/\mathfrak{m}$ be the quotient ring, which is now a finite field by the maximality of $\mathfrak{m}$ (see e.g., [5]). Let $\varphi\colon \mathcal{M} \to \mathbb{F}$ denote the quotient map. Note that, for any $m \in \mathcal{M}^{\times}$, $\varphi(m)$ is also an invertible element of $\mathbb{F}$, hence $\varphi(m) \neq 0$. We emphasize that it is *not* required in the following argument that such $\mathbb{F} = \mathcal{M}/\mathfrak{m}$ and φ are efficiently computable. For any coefficient vector $\vec{m}$, let $\varphi(\vec{m})$ denote the vector obtained by taking the image of every component of $\vec{m}$ by φ. Moreover, for any polynomial Ψ over $\mathcal{M}$, let $\varphi(\Psi)$ denote the polynomial over $\mathbb{F}$ given by applying the map φ to every coefficient of Ψ.

In our argument below, the (ir)reducibility of polynomials of the form $\varphi(\vec{m} \cdot \mathcal{F})|_{m_{b^*}=0}$ plays a key role. First, we present the following lemma:

Lemma 7. *Let $t_1, \dots, t_n$ be distinct variables, and let $f_0, f_1, \dots, f_n$ be polynomials over the field $\mathbb{F}$ which do not involve the variables $t_1, \dots, t_n$. If $f := f_0 + \sum_{j=1}^{n} f_j t_j$ is reducible (i.e., having a divisor which is not a scalar multiple of itself), then the polynomials $f_0, \dots, f_n$ have a non-constant common divisor.*

Proof. Since f is reducible, we have $f = h_0 \cdot h_1$ for some non-constant polynomials h_0 and h_1. For each index $1 \leq j \leq n$ with $f_j \neq 0$, the variable t_j has degree one in f by the assumption. Since f is a polynomial over the *field* $\mathbb{F}$, it follows that either h_0 or h_1, say h_{i_j}, has degree zero with respect to t_j, and the other polynomial h_{1-i_j} has degree one with respect to t_j. Now the coefficient of t_j in f, which is $f_j \neq 0$ by the assumption, is a multiple of h_{i_j}. Therefore, by the assumption on f_j, the polynomial h_{i_j} does not involve the variables $t_1, \dots, t_n$; while h_{1-i_j} involves the variable t_j as above. This implies that the index $i_j \in \{0, 1\}$ must be common for all $1 \leq j \leq n$ with $f_j \neq 0$, therefore the non-constant polynomial h_{i_j} is a divisor of every $f_1, \dots, f_n$. Moreover, since f is a multiple of h_{i_j}, now h_{i_j} is also a divisor of $f - \sum_{j=1}^{n} f_j t_j = f_0$. Hence Lemma 7 holds. □

We introduce an auxiliary terminology; we say that a coefficient vector $\vec{m}$ is *invertible*, if every non-zero component of $\vec{m}$ is invertible in $\mathcal{M}$. Now we introduce the following classification of non-zero invertible coefficient vectors $\vec{m}$, which will be used in our argument:

Type I The polynomial $\varphi(\vec{m} \cdot \mathcal{F})|_{m_{b^*}=0}$ is reducible.

Type II For some non-zero invertible coefficient vector $\vec{m}'$, the polynomial $\varphi(\vec{m} \cdot \mathcal{F})|_{m_{b^*}=0}$ divides $\varphi(\vec{m}' \cdot \mathcal{F})|_{m_{b^*}=0}$ but is not a scalar multiple of $\varphi(\vec{m}' \cdot \mathcal{F})|_{m_{b^*}=0}$.

Type III Otherwise.

Then we have the following:

Lemma 8. *The coefficient vectors $\vec{m}$ of types I and II are as listed in Tables 1 and 2.*

Table 1: Coefficient vectors $\vec{m}$ of type I and type II in Lemma 8, when $T^{\mathrm{out}} = T_{\ell^{\mathrm{out}}}$ (here the last column means that every component $m[\widetilde{\gamma}^{\mathrm{out}}]$ of $\vec{m}$ with index $\widetilde{\gamma}^{\mathrm{out}}$ not listed there is equal to zero)

type		indices $\widetilde{\gamma}^{\mathrm{out}}$ for possibly non-zero components
I	I-i ($\ell^{\mathrm{out}} \geq 2$, $1 \leq i \leq \ell^{\mathrm{out}}$)	$c_i^{*\mathrm{out}}$, $\mathsf{PK}_{r\to i}^{\mathrm{out}}$ ($m[c_i^{*\mathrm{out}}] \neq 0$)
	I-0 ($\ell^{\mathrm{out}} = 2$)	$c_1^{*\mathrm{out}}$, $\mathsf{PK}_{r\to 1}^{\mathrm{out}}$, $c_2^{*\mathrm{out}}$, $\mathsf{PK}_{r\to 2}^{\mathrm{out}}$ ($\varphi(m[c_1^{*\mathrm{out}}]) = -\alpha\varphi(m[c_2^{*\mathrm{out}}]) \neq 0$ and $\varphi(m[\mathsf{PK}_{r\to 1}^{\mathrm{out}}]) = \alpha\varphi(m[\mathsf{PK}_{r\to 2}^{\mathrm{out}}])$ for some $\alpha \in \mathbb{F} \setminus \{0\}$)
II	($\ell^{\mathrm{out}} \geq 2$)	$\mathsf{PK}_{r\to i}^{\mathrm{out}}$ ($1 \leq i \leq \ell^{\mathrm{out}}$, $m[\mathsf{PK}_{r\to i}^{\mathrm{out}}] \neq 0$)
	const ($m[\mathsf{const}] \neq 0$)	
	($\ell^{\mathrm{out}} = 2$)	$\mathsf{PK}_{r\to 1}^{\mathrm{out}}$, $\mathsf{PK}_{r\to 2}^{\mathrm{out}}$ ($m[\mathsf{PK}_{r\to 1}^{\mathrm{out}}], m[\mathsf{PK}_{r\to 2}^{\mathrm{out}}] \neq 0$)

Table 2: Coefficient vectors $\vec{m}$ of type I and type II in Lemma 8, when $T^{\mathrm{out}} = T^{\S}$ (here the last column means that every component $m[\widetilde{\gamma}^{\mathrm{out}}]$ of $\vec{m}$ with index $\widetilde{\gamma}^{\mathrm{out}}$ not listed there is equal to zero)

type		indices $\widetilde{\gamma}^{\mathrm{out}}$ for possibly non-zero components
I	I-1	$\mathsf{PK}_{r\to 1}^{\mathrm{out}}$, $c_1^{*\mathrm{out}}$ ($m[c_1^{*\mathrm{out}}] \neq 0$)
	I-2	$\mathsf{PK}_{2\to 3}^{\mathrm{out}}$, $\mathsf{PK}_{r\to 2}^{\mathrm{out}(1)}$, $c_3^{*\mathrm{out}}$ ($m[\mathsf{PK}_{r\to 2}^{\mathrm{out}(1)}] \neq 0$ or $m[c_3^{*\mathrm{out}}] \neq 0$)
	I-3	$\mathsf{PK}_{2\to 4}^{\mathrm{out}}$, $\mathsf{PK}_{r\to 2}^{\mathrm{out}(2)}$, $c_4^{*\mathrm{out}}$ ($m[\mathsf{PK}_{r\to 2}^{\mathrm{out}(2)}] \neq 0$ or $m[c_4^{*\mathrm{out}}] \neq 0$)
II	$\mathsf{PK}_e^{\mathrm{out}}$ ($e = (r \to 1)$, $(2 \to 3)$ or $(2 \to 4)$, $m[\mathsf{PK}_e^{\mathrm{out}}] \neq 0$)	
	const ($m[\mathsf{const}] \neq 0$)	

Proof. Since $\vec{m}$ is invertible, we have $\varphi(m[\widetilde{\gamma}^{\mathrm{out}}]) \neq 0$ for any non-zero component $m[\widetilde{\gamma}^{\mathrm{out}}]$ of $\vec{m}$. First we note that, if $m[\mathsf{const}]$ is the only non-zero component of $\vec{m}$, then $\vec{m}$ is of type II (since $\vec{m} \cdot \mathcal{F}$ is a constant), as listed in Tables 1 and 2. From now, we consider the other case that some component of $\vec{m}$ other than $m[\mathsf{const}]$ is non-zero. In this case, if $\vec{m}$ is of type II, then the coefficient vector $\vec{m}'$ in the definition of type II is of type I.

We divide the argument into the following three cases.

Case 1: $T^{\mathrm{out}} = T_2$**.** In this case, we have

$$\begin{aligned}\vec{m} \cdot \mathcal{F}|_{m_{b^*}=0} = m[\mathsf{const}] &+ m[c_1^{*\mathrm{out}}] \cdot s_1 \cdot \mathsf{SK}_{r\to 1} + m[\mathsf{PK}_{r\to 1}^{\mathrm{out}}] \cdot \mathsf{SK}_{r\to 1} \\ &- m[c_2^{*\mathrm{out}}] \cdot s_1 \cdot \mathsf{SK}_{r\to 2} + m[\mathsf{PK}_{r\to 2}^{\mathrm{out}}] \cdot \mathsf{SK}_{r\to 2} \enspace .\end{aligned}$$

If $\vec{m}$ is of type I, then by Lemma 7 applied to the variables $\mathsf{SK}_{r\to 1}$ and $\mathsf{SK}_{r\to 2}$, the three polynomials $\varphi(m[\mathsf{const}])$, $\varphi(m[c_1^{*\mathrm{out}}]) \cdot s_1 + \varphi(m[\mathsf{PK}_{r\to 1}^{\mathrm{out}}])$ and $-\varphi(m[c_2^{*\mathrm{out}}]) \cdot s_1 + \varphi(m[\mathsf{PK}_{r\to 2}^{\mathrm{out}}])$ have a non-constant common divisor (hence each polynomial is not constant unless it is zero). In particular, $\varphi(m[\mathsf{const}])$ must be zero, and for each $j \in \{1,2\}$, we have $\varphi(m[\mathsf{PK}_{r\to j}^{\mathrm{out}}]) = 0$ whenever $\varphi(m[c_j^{*\mathrm{out}}]) = 0$.

For each $j \in \{1,2\}$, if $\varphi(m[c_j^{*\mathrm{out}}]) = \varphi(m[\mathsf{PK}_{r\to j}^{\mathrm{out}}]) = 0$, then we have $\varphi(m[{c_{3-j}^{*}}^{\mathrm{out}}]) \neq 0$

since $\vec{m}$ is non-zero. This case is listed as "type I-i" with $i = 3 - j$ in Table 1; note that, in such a case, $\varphi(\vec{m} \cdot \mathcal{F})|_{m_{b^*}=0}$ is indeed reducible, therefore $\vec{m}$ is indeed of type I.

For the other case that $\varphi(m[c_j^{*\mathrm{out}}]) \neq 0$ for any $j \in \{1, 2\}$, $\varphi(m[c_1^{*\mathrm{out}}]) \cdot s_1 + \varphi(m[\mathsf{PK}_{r\to 1}^{\mathrm{out}}])$ and $-\varphi(m[c_2^{*\mathrm{out}}]) \cdot s_1 + \varphi(m[\mathsf{PK}_{r\to 2}^{\mathrm{out}}])$ are of degree one and have a non-constant common divisor. This is possible only when these two polynomials are constant multiple of each other. This case is listed as "type I-0" in Table 1; note that, in such a case, $\varphi(\vec{m} \cdot \mathcal{F})|_{m_{b^*}=0}$ is indeed reducible, therefore $\vec{m}$ is indeed of type I.

Moreover, as mentioned above, if $\vec{m}$ is of type II, then $\varphi(\vec{m} \cdot \mathcal{F})|_{m_{b^*}=0}$ is a divisor of $\varphi(\vec{m}' \cdot \mathcal{F})|_{m_{b^*}=0}$ for some $\vec{m}'$ of type I. Now the result above on type I implies that the possibilities of $\vec{m}$ are as listed in Table 1. Hence, the claim holds for Case 1.

Case 2: $T^{\mathrm{out}} = T_{\ell^{\mathrm{out}}}$ with $\ell^{\mathrm{out}} \neq 2$. In this case, we have

$$\begin{aligned}\vec{m} \cdot \mathcal{F}|_{m_{b^*}=0} = m[\mathsf{const}] + \sum_{j=1}^{\ell^{\mathrm{out}}-1} (m[c_j^{*\mathrm{out}}] \cdot s_j \cdot \mathsf{SK}_{r\to j} + m[\mathsf{PK}_{r\to j}^{\mathrm{out}}] \cdot \mathsf{SK}_{r\to j}) \\ - m[c_{\ell^{\mathrm{out}}}^{*}{}^{\mathrm{out}}] \cdot \left(\sum_{j=1}^{\ell^{\mathrm{out}}-1} s_j \right) \cdot \mathsf{SK}_{r\to \ell^{\mathrm{out}}} + m[\mathsf{PK}_{r\to \ell^{\mathrm{out}}}^{\mathrm{out}}] \cdot \mathsf{SK}_{r\to \ell^{\mathrm{out}}} \enspace .\end{aligned}$$

If $\vec{m}$ is of type I, then by Lemma 7 applied to the variables $\mathsf{SK}_{r\to j}$ for $1 \leq j \leq \ell^{\mathrm{out}}$, the polynomials $\varphi(m[\mathsf{const}])$, $\varphi(m[c_j^{*\mathrm{out}}]) \cdot s_j + \varphi(m[\mathsf{PK}_{r\to j}^{\mathrm{out}}])$ for $1 \leq j \leq \ell^{\mathrm{out}} - 1$, and $-\varphi(m[c_{\ell^{\mathrm{out}}}^{*}{}^{\mathrm{out}}]) \cdot \sum_{j=1}^{\ell^{\mathrm{out}}-1} s_j + \varphi(m[\mathsf{PK}_{r\to \ell^{\mathrm{out}}}^{\mathrm{out}}])$ have a non-constant common divisor (hence each polynomial is not constant unless it is zero). This does not happen when $\ell^{\mathrm{out}} = 1$, since $\vec{m}$ is non-zero. From now, we consider the case $\ell^{\mathrm{out}} > 2$. In this case, the argument above implies that $m[\mathsf{const}] = 0$ and only one of the remaining ℓ^{out} polynomials above, say j-th with $1 \leq j \leq \ell^{\mathrm{out}}$, is non-zero. Now we have $\varphi(m[c_j^{*\mathrm{out}}]) \neq 0$; this case is listed as "type I-i" with $i = j$ in Table 1. Note that, in such a case, $\varphi(\vec{m} \cdot \mathcal{F})|_{m_{b^*}=0}$ is indeed reducible, therefore $\vec{m}$ is indeed of type I.

Moreover, as mentioned above, if $\vec{m}$ is of type II, then $\varphi(\vec{m} \cdot \mathcal{F})|_{m_{b^*}=0}$ is a divisor of $\varphi(\vec{m}' \cdot \mathcal{F})|_{m_{b^*}=0}$ for some $\vec{m}'$ of type I. Now the result above on type I implies that the possibilities of $\vec{m}$ are as listed in Table 1. Hence, the claim holds for Case 2.

Case 3: $T^{\mathrm{out}} = T^{\S}$. In this case, we have

$$\begin{aligned}\vec{m} \cdot \mathcal{F}|_{m_{b^*}=0} &= m[\mathsf{const}] + \big(m[\mathsf{PK}_{r\to 1}^{\mathrm{out}}] + m[c_1^{*\mathrm{out}}] \cdot s_1\big) \cdot \mathsf{SK}_{r\to 1} \\ &+ \big(m[\mathsf{PK}_{2\to 3}^{\mathrm{out}}] + m[\mathsf{PK}_{r\to 2}^{\mathrm{out}(1)}] \cdot s_{\mathrm{pk}} + m[c_3^{*\mathrm{out}}] \cdot (-s_1 \cdot s_{\mathrm{pk}} + s_2)\big) \cdot \mathsf{SK}_{2\to 3} \\ &+ \big(m[\mathsf{PK}_{2\to 4}^{\mathrm{out}}] + m[\mathsf{PK}_{r\to 2}^{\mathrm{out}(2)}] \cdot (\mathsf{SK}_{r\to 2} - s_{\mathrm{pk}}) + m[c_4^{*\mathrm{out}}](-s_1 \cdot (\mathsf{SK}_{r\to 2} - s_{\mathrm{pk}}) - s_2)\big) \cdot \mathsf{SK}_{2\to 4} \enspace .\end{aligned}$$

If $\vec{m}$ is of type I, then by applying Lemma 7 to the variables $\mathsf{SK}_{r\to 1}$, $\mathsf{SK}_{2\to 3}$ and $\mathsf{SK}_{2\to 4}$ similarly to Case 2 above, it follows that $m[\mathsf{const}] = 0$, precisely one of the three polynomials $m[\mathsf{PK}_{r\to 1}^{\mathrm{out}}] + m[c_1^{*\mathrm{out}}] \cdot s_1$, $m[\mathsf{PK}_{2\to 3}^{\mathrm{out}}] + m[\mathsf{PK}_{r\to 2}^{\mathrm{out}(1)}] \cdot s_{\mathrm{pk}} + m[c_3^{*\mathrm{out}}] \cdot (-s_1 \cdot s_{\mathrm{pk}} + s_2)$ and $m[\mathsf{PK}_{2\to 4}^{\mathrm{out}}] + m[\mathsf{PK}_{r\to 2}^{\mathrm{out}(2)}] \cdot (\mathsf{SK}_{r\to 2} - s_{\mathrm{pk}}) + m[c_4^{*\mathrm{out}}](-s_1 \cdot (\mathsf{SK}_{r\to 2} - s_{\mathrm{pk}}) - s_2)$ is non-zero, and the non-zero polynomial is not constant. These cases are listed in Table 2, where "type I-1", "type I-2" and "type I-3" correspond to the cases that the first, the second and the third polynomials above are non-zero, respectively. Note that, in such a case, $\varphi(\vec{m} \cdot \mathcal{F})|_{m_{b^*}=0}$ is indeed reducible, therefore $\vec{m}$ is indeed of type I.

Moreover, as mentioned above, if $\vec{m}$ is of type II, then $\varphi(\vec{m} \cdot \mathcal{F})|_{m_{b^*}=0}$ is a divisor of $\varphi(\vec{m}' \cdot \mathcal{F})|_{m_{b^*}=0}$ for some $\vec{m}'$ of type I. Now the result above on type I implies that the possibilities of $\vec{m}$ are as listed in Table 2. Hence, the claim holds for Case 3.

This completes the proof of Lemma 8. $\square$

By using Lemma 8, we give another key property in our argument below:

Lemma 9. *Let $\vec{m}$ be a non-zero invertible coefficient vector, which is either of type I except type I-0 for $T^{\text{out}} = T_2$ (see Table 1), or of type III. Let $\widetilde{\gamma}_0^{\text{out}}$ be an index with $m[\widetilde{\gamma}_0^{\text{out}}] \neq 0$. Then for any collection of non-zero invertible coefficient vectors $\vec{m}^{(i)}$, $1 \leq i \leq n$, which is not of type I-0 for $\Gamma^{\text{out}} = \Gamma_2$, if $\varphi(\vec{m} \cdot \mathcal{F})|_{m_{b^*}=0}$ divides $\prod_{i=1}^{n} \varphi(\vec{m}^{(i)} \cdot \mathcal{F})|_{m_{b^*}=0}$, then there exists an index $1 \leq h \leq n$ satisfying that, for each index $\widetilde{\gamma}^{\text{out}}$,*

$$\varphi\left(m^{(h)}[\widetilde{\gamma}^{\text{out}}] - \frac{m^{(h)}[\widetilde{\gamma}_0^{\text{out}}]}{m[\widetilde{\gamma}_0^{\text{out}}]} \cdot m[\widetilde{\gamma}^{\text{out}}]\right) = 0 \ .$$

Proof. First, we consider the case that $\vec{m}$ is of type III. Then the polynomial $\varphi(\vec{m}\cdot\mathcal{F})|_{m_{b^*}=0}$ is irreducible (hence is a prime polynomial, as mentioned above), therefore it divides some $\varphi(\vec{m}^{(h)}\cdot\mathcal{F})|_{m_{b^*}=0}$. Since $\vec{m}$ is not of type II, we have $\varphi(\vec{m}^{(h)}\cdot\mathcal{F})|_{m_{b^*}=0} = \gamma\cdot\varphi(\vec{m}\cdot\mathcal{F})|_{m_{b^*}=0}$ for some $\gamma \in \mathbb{F}$, therefore $\varphi(m^{(h)}[\widetilde{\gamma}_0^{\text{out}}]/m[\widetilde{\gamma}_0^{\text{out}}]) = \gamma$ and $\varphi(m^{(h)}[\widetilde{\gamma}^{\text{out}}]) = \gamma \cdot \varphi(m[\widetilde{\gamma}^{\text{out}}])$ for every index $\widetilde{\gamma}^{\text{out}}$. Hence the claim holds in this case.

From now, we consider the other case that $\vec{m}$ is of type I except type I-0 for $T^{\text{out}} = T_2$. First, we suppose that $T^{\text{out}} = T_{\ell^{\text{out}}}$, $\ell^{\text{out}} \geq 2$. Let the type of $\vec{m}$ be type I-i. Set

$$f := \begin{cases} \varphi(m[c_i^{*\text{out}}])s_i + \varphi(m[\mathsf{PK}_{r\to i}^{\text{out}}]) & \text{if } 1 \leq i \leq \ell^{\text{out}} - 1 \ , \\ -\varphi(m[c_{\ell^{\text{out}}}^{*}{}^{\text{out}}] \sum_{j=1}^{\ell^{\text{out}}-1} s_j + \varphi(m[\mathsf{PK}_{r\to\ell^{\text{out}}}^{\text{out}}]) & \text{if } i = \ell^{\text{out}} \ . \end{cases}$$

Then f is an irreducible (hence prime) polynomial and is a non-constant divisor of $\varphi(\vec{m}\cdot\mathcal{F})|_{m_{b^*}=0}$ (see Table 1). Therefore, f divides some $\varphi(\vec{m}^{(h)}\cdot\mathcal{F})|_{m_{b^*}=0}$. Now by the shape of these polynomials, f is not a scalar multiple of $\varphi(\vec{m}^{(h)}\cdot\mathcal{F})|_{m_{b^*}=0}$, therefore $\vec{m}^{(h)}$ is of type I. Since neither $\vec{m}$ nor $\vec{m}^{(h)}$ is of type I-0 for $T^{\text{out}} = T_2$ by the assumption, the existence of such a common divisor f implies (by Table 1) that $\varphi(\vec{m}^{(h)}\cdot\mathcal{F})|_{m_{b^*}=0}$ must be a scalar multiple of $\varphi(\vec{m}\cdot\mathcal{F})|_{m_{b^*}=0}$. Now the claim follows by the same argument as the previous paragraph.

On the other hand, we suppose that $T^{\text{out}} = T^{\S}$. Let the type of $\vec{m}$ be type I-i. Set

$$f := \begin{cases} \varphi(m[\mathsf{PK}_{r\to1}^{\text{out}}]) - \varphi(m[c_1^{*\text{out}}])\cdot s_1 & \text{if } i = 1 \ , \\ \varphi(m[\mathsf{PK}_{2\to3}^{\text{out}}]) + \varphi(m[\mathsf{PK}_{r\to2}^{\text{out}(1)}])\cdot s_{\text{pk}} + \varphi(m[c_3^{*\text{out}}])\cdot(-s_1\cdot s_{\text{pk}} + s_2) & \text{if } i = 2 \ , \\ \varphi(m[\mathsf{PK}_{2\to4}^{\text{out}}]) + \varphi(m[\mathsf{PK}_{r\to2}^{\text{out}(2)}])\cdot(\mathsf{SK}_{r\to2} - s_{\text{pk}}) & \\ \qquad +\varphi(m[c_4^{*\text{out}}])(-s_1\cdot(\mathsf{SK}_{r\to2} - s_{\text{pk}}) - s_2) & \text{if } i = 3 \ . \end{cases}$$

Then f is a non-constant divisor of $\varphi(\vec{m}\cdot\mathcal{F})|_{m_{b^*}=0}$. Now the same argument as the previous paragraph (using Table 2 instead of Table 1) implies that some $\varphi(\vec{m}^{(h)}\cdot\mathcal{F})|_{m_{b^*}=0}$ must be a scalar multiple of $\varphi(\vec{m}\cdot\mathcal{F})|_{m_{b^*}=0}$, therefore the claim holds. Hence the proof of Lemma 9 is concluded. $\square$

7.8 Evaluation of the Probability: Overall Strategy

We come back to the evaluation of the probability that $\Phi \neq 0$ but $\Phi' = \Phi|_{m_{b^*}=0} = 0$ as polynomials, using the results above. The overall strategy is as follows: We construct a PPT algorithm $\mathcal{D}$ of the following form. The input for $\mathcal{D}$ is the security parameter k, and its output is one of $\top$, $\bot$ and an element of $\mathcal{M} \setminus (\mathcal{M}^{\times} \cup \{0\})$. For some subroutines $\mathcal{D}'_1, \ldots, \mathcal{D}'_\rho$ specified later, the algorithm $\mathcal{D}$ proceeds as follows:

1. Generate a challenge for the ZPA game for $\Gamma(T^{\mathrm{out}})$.
2. Choose an index $1 \leq \rho' \leq \rho$ uniformly at random.
3. Calculate the collection, denoted by $\vec{\vec{m}}$, of coefficient vectors $\vec{m}[\gamma^{\mathrm{in}}] = (m[\gamma^{\mathrm{in}}; \widetilde{\gamma}^{\mathrm{out}}])_{\widetilde{\gamma}^{\mathrm{out}}}$ associated to the input components γ^{in} for the ρ'-th oracle $\mathcal{O}'_{T^{\mathrm{in}}}$, by emulating the circuit $\mathcal{C}_k^{\rho'}$ with the challenge above as input.
4. Execute the subroutine $\mathcal{D}'_{\rho'}$ with input $(\mathsf{pk}, \vec{\vec{m}})$, and output the output of $\mathcal{D}'_{\rho'}$.

We will construct the subroutines $\mathcal{D}'_{\rho'}$ to satisfy the following conditions:

1. If $\mathcal{D}'_{\rho'}$ outputs $\top$, then $\Phi = 0$ as a polynomial.
2. If $\mathcal{D}'_{\rho'}$ outputs $\bot$, then $\Phi|_{m_{b^*}=0} \neq 0$ as a polynomial.

Then the probability that $\Phi \neq 0$ but $\Phi|_{m_{b^*}=0} = 0$ as polynomials, considered in the case of the ρ'-th oracles $\mathcal{O}_{T^{\mathrm{in}}}$ and $\mathcal{O}'_{T^{\mathrm{in}}}$, does not exceed the probability that $\mathcal{D}'_{\rho'}$ outputs an element of $\mathcal{M} \setminus (\mathcal{M}^{\times} \cup \{0\})$, hence does not exceed ρ times the probability that $\mathcal{D}$ outputs an element of $\mathcal{M} \setminus (\mathcal{M}^{\times} \cup \{0\})$ (in particular, the bound is independent of ρ'). Moreover, the latter probability is negligible, by Assumption 2 and the assumption here that $\mathcal{D}$ is PPT. Since ρ is polynomially bounded, it will follow that the probability that $\Phi \neq 0$ but $\Phi|_{m_{b^*}=0} = 0$ as polynomials is negligible, which will complete the proof of Theorem 4.

From now, we define the subroutine $\mathcal{D}' = \mathcal{D}'_{\rho'}$ as follows, where we set $T^{\mathrm{in}} = T_{\ell^{\mathrm{in}}}$:

1. If $m[\gamma^{\mathrm{in}}; \widetilde{\gamma}^{\mathrm{out}}] \in \mathcal{M} \setminus (\mathcal{M}^{\times} \cup \{0\})$ for some indices γ^{in} and $\widetilde{\gamma}^{\mathrm{out}}$, then output the element of $\mathcal{M} \setminus (\mathcal{M}^{\times} \cup \{0\})$ and halt.
2. If $\vec{m}[\mathsf{PK}^{\mathrm{in}}_{r \to j}]$ is the zero vector $\vec{0}$ for some $1 \leq j \leq \ell^{\mathrm{in}}$, then:
   (a) If $\vec{m}[\mathsf{PK}^{\mathrm{in}}_{r \to j'}] \neq \vec{0}$ for every $1 \leq j' \leq \ell^{\mathrm{in}}$ with $j' \neq j$ and $\vec{m}[c_j^{*\mathrm{in}}] \neq \vec{0}$, then output $\bot$ and halt.
   (b) Otherwise, output $\top$ and halt.
3. For each $1 \leq j \leq \ell^{\mathrm{in}}$, if $m[\mathsf{PK}^{\mathrm{in}}_{r \to j}; \widetilde{\gamma}^{\mathrm{out}}] \neq 0$ for some index $\widetilde{\gamma}^{\mathrm{out}} \neq \mathsf{const}$, then choose such an index $\widetilde{\gamma}_j^{\mathrm{out}} := \widetilde{\gamma}^{\mathrm{out}}$; otherwise, set $\widetilde{\gamma}_j^{\mathrm{out}} := \mathsf{const}$. Moreover, perform the replacement
$$\vec{m}[c_j^{*\mathrm{in}}] \leftarrow \frac{1}{m[\mathsf{PK}^{\mathrm{in}}_{r \to j}; \widetilde{\gamma}_j^{\mathrm{out}}]} \cdot \vec{m}[c_j^{*\mathrm{in}}] \, , \, \vec{m}[\mathsf{PK}^{\mathrm{in}}_{r \to j}] \leftarrow \frac{1}{m[\mathsf{PK}^{\mathrm{in}}_{r \to j}; \widetilde{\gamma}_j^{\mathrm{out}}]} \cdot \vec{m}[\mathsf{PK}^{\mathrm{in}}_{r \to j}] \ .$$
4. Initialize a set I by $I := \{1, 2, \dots, \ell^{\mathrm{in}}\}$, and set $\vec{m}^I[c_j^{*\mathrm{in}}] := \vec{m}[c_j^{*\mathrm{in}}]$ for each $j \in I$.
5. Repeat, until $|I|$ becomes 1, the following (referred to as the "first-level loop"):
   (a) If $m^I[c_j^{*\mathrm{in}}; \widetilde{\gamma}^{\mathrm{out}}] \in \mathcal{M} \setminus (\mathcal{M}^{\times} \cup \{0\})$ for some $j \in I$ and an index $\widetilde{\gamma}^{\mathrm{out}}$, then output the element of $\mathcal{M} \setminus (\mathcal{M}^{\times} \cup \{0\})$ and halt.
   (b) Repeat the following for each $j \in I$ (referred to as the "second-level loop"):

i. If for some index $\widetilde{\gamma}^{\mathrm{out}}$, we have

$$m[\mathsf{PK}^{\mathrm{in}}_{r\to h};\widetilde{\gamma}^{\mathrm{out}}] - m[\mathsf{PK}^{\mathrm{in}}_{r\to h};\widetilde{\gamma}^{\mathrm{out}}_j]\cdot m[\mathsf{PK}^{\mathrm{in}}_{r\to j};\widetilde{\gamma}^{\mathrm{out}}] \in \mathcal{M}\setminus(\mathcal{M}^{\times}\cup\{0\})$$

for some $h \in I \setminus \{j\}$, or

$$m^I[c^{*\mathrm{in}}_j;\widetilde{\gamma}^{\mathrm{out}}] - m^I[c^{*\mathrm{in}}_j;\widetilde{\gamma}^{\mathrm{out}}_j]\cdot m[\mathsf{PK}^{\mathrm{in}}_{r\to j};\widetilde{\gamma}^{\mathrm{out}}] \in \mathcal{M}\setminus(\mathcal{M}^{\times}\cup\{0\}) \ ,$$

then output this element of $\mathcal{M}\setminus(\mathcal{M}^{\times}\cup\{0\})$ and halt.

ii. If we have

$$\vec{m}[\mathsf{PK}^{\mathrm{in}}_{r\to h}] \neq m[\mathsf{PK}^{\mathrm{in}}_{r\to h};\widetilde{\gamma}^{\mathrm{out}}_j]\cdot\vec{m}[\mathsf{PK}^{\mathrm{in}}_{r\to j}] \text{ for every } h\in I\setminus\{j\}$$

and

$$\vec{m}^I[c^{*\mathrm{in}}_j] \neq m^I[c^{*\mathrm{in}}_j;\widetilde{\gamma}^{\mathrm{out}}_j]\cdot\vec{m}[\mathsf{PK}^{\mathrm{in}}_{r\to j}] \ ,$$

then replace j with the next element of I and repeat the second-level loop.

iii. If $\vec{m}^I[c^{*\mathrm{in}}_j] = m^I[c^{*\mathrm{in}}_j;\widetilde{\gamma}^{\mathrm{out}}_j]\cdot\vec{m}[\mathsf{PK}^{\mathrm{in}}_{r\to j}]$, then set $I' := I\setminus\{j\}$, choose an element $h\in I'$, set $\vec{m}^{I'}[c^{*}_{j'}{}^{\mathrm{in}}] := \vec{m}^I[c^{*}_{j'}{}^{\mathrm{in}}]$ for each $j'\in I'\setminus\{h\}$, set

$$\vec{m}^{I'}[c^{*\mathrm{in}}_h] := \vec{m}^I[c^{*\mathrm{in}}_h] + m^I[c^{*\mathrm{in}}_j;\widetilde{\gamma}^{\mathrm{out}}_j]\cdot\vec{m}[\mathsf{PK}^{\mathrm{in}}_{r\to h}] \ ,$$

perform the replacement $I \leftarrow I'$, and repeat the first-level loop.

iv. Set $I' := I\setminus\{j\}$ and choose an element $h\in I'$ satisfying $\vec{m}[\mathsf{PK}^{\mathrm{in}}_{r\to h}] = m[\mathsf{PK}^{\mathrm{in}}_{r\to h};\widetilde{\gamma}^{\mathrm{out}}_j]\cdot\vec{m}[\mathsf{PK}^{\mathrm{in}}_{r\to j}]$. Then set $\vec{m}^{I'}[c^{*}_{j'}{}^{\mathrm{in}}] := \vec{m}^I[c^{*}_{j'}{}^{\mathrm{in}}]$ for each $j'\in I'\setminus\{h\}$, set

$$\vec{m}^{I'}[c^{*\mathrm{in}}_h] := \vec{m}^I[c^{*\mathrm{in}}_h] + m[\mathsf{PK}^{\mathrm{in}}_{r\to h};\widetilde{\gamma}^{\mathrm{out}}_j]\cdot\vec{m}^I[c^{*\mathrm{in}}_j] \ ,$$

perform the replacement $I \leftarrow I'$, and repeat the first-level loop.

(c) Output $\perp$ and halt.

6. Let j be the unique element of I. If $m^I[c^{*\mathrm{in}}_j;\widetilde{\gamma}^{\mathrm{out}}]\in\mathcal{M}\setminus(\mathcal{M}^{\times}\cup\{0\})$ for some index $\widetilde{\gamma}^{\mathrm{out}}$, then output the element of $\mathcal{M}\setminus(\mathcal{M}^{\times}\cup\{0\})$ and halt.

7. If $\vec{m}^I[c^{*\mathrm{in}}_j] = \vec{0}$, then output $\top$ and halt.

8. Output $\perp$ and halt.

We evaluate the computational complexity of $\mathcal{D}'$. Each task in $\mathcal{D}'$ can be efficiently executed, and the number of tasks in $\mathcal{D}'$ is of order $O((\ell^{\mathrm{in}})^3)$ (note that the number of indices $\widetilde{\gamma}^{\mathrm{out}}$, which depends solely on T^{out}, is now a constant). Since ℓ^{in} is polynomially bounded, it follows that the computational complexity of $\mathcal{D}'$ is also bounded by a polynomial (common for all $\mathcal{D}' = \mathcal{D}'_{\rho'}$). Hence $\mathcal{D}$ is PPT, as desired.

7.9 Analysis of the Subroutine

Now the remaining task is to prove the above-mentioned relations between the output of $\mathcal{D}'$ and the polynomials Φ and $\Phi|_{m_{b^*}=0}$. For the purpose, it suffices to consider the case that $\mathcal{D}'$ *does not output an element of* $\mathcal{M}\setminus(\mathcal{M}^{\times}\cup\{0\})$.

In the present situation, $\mathcal{D}'$ does not halt at Step 1, therefore all the coefficient vectors $\vec{m}[\gamma^{\mathrm{in}}]$ are invertible. Now by the definition of Φ, we have $\Phi = 0$ (as a polynomial) if and only if

$$\sum_{j=1}^{\ell^{\mathrm{in}}} (\vec{m}[c_j^{*\mathrm{in}}] \cdot \mathcal{F}) \prod_{1 \leq h \leq \ell^{\mathrm{in}}\,;\, h \neq j} (\vec{m}[\mathsf{PK}_{r \to h}^{\mathrm{in}}] \cdot \mathcal{F}) = 0 \ , \tag{1}$$

while $\Phi|_{m_{b^*}=0} = 0$ if and only if

$$\sum_{j=1}^{\ell^{\mathrm{in}}} (\vec{m}[c_j^{*\mathrm{in}}] \cdot \mathcal{F}|_{m_{b^*}=0}) \prod_{1 \leq h \leq \ell^{\mathrm{in}}\,;\, h \neq j} (\vec{m}[\mathsf{PK}_{r \to h}^{\mathrm{in}}] \cdot \mathcal{F}|_{m_{b^*}=0}) = 0 \ . \tag{2}$$

For Step 2, if $1 \leq j \leq \ell^{\mathrm{in}}$ and $\vec{m}[\mathsf{PK}_{r \to j}^{\mathrm{in}}] = \vec{0}$, then (1) is equivalent to

$$(\vec{m}[c_j^{*\mathrm{in}}] \cdot \mathcal{F}) \prod_{1 \leq h \leq \ell\,;\, h \neq j} (\vec{m}[\mathsf{PK}_{r \to h}^{\mathrm{in}}] \cdot \mathcal{F}) = 0 \ ,$$

which implies that

$$\varphi(\vec{m}[c_j^{*\mathrm{in}}] \cdot \mathcal{F}) \prod_{1 \leq h \leq \ell\,;\, h \neq j} \varphi(\vec{m}[\mathsf{PK}_{r \to h}^{\mathrm{in}}] \cdot \mathcal{F}) = 0 \ .$$

This also implies (since the coefficient ring $\mathbb{F}$ is now a *field*) that either $\varphi(\vec{m}[c_j^{*\mathrm{in}}] \cdot \mathcal{F}) = 0$ or $\varphi(\vec{m}[\mathsf{PK}_{r \to h}^{\mathrm{in}}] \cdot \mathcal{F}) = 0$ for some $h \neq j$. By the shapes of polynomials $\mathcal{F}[\gamma^{\mathrm{out}}]$, it also follows that either $\varphi(\vec{m}[c_j^{*\mathrm{in}}]) = \vec{0}$ or $\varphi(\vec{m}[\mathsf{PK}_{r \to h}^{\mathrm{in}}]) = \vec{0}$ for some $h \neq j$. Moreover, since the coefficient vectors are all invertible as mentioned above, we have either $\vec{m}[c_j^{*\mathrm{in}}] = \vec{0}$ or $\vec{m}[\mathsf{PK}_{r \to h}^{\mathrm{in}}] = \vec{0}$ for some $h \neq j$, which now implies (1). Summarizing, in the present case, (1) is equivalent to that $\vec{m}[c_j^{*\mathrm{in}}] = \vec{0}$ or $\vec{m}[\mathsf{PK}_{r \to h}^{\mathrm{in}}] = \vec{0}$ for some $h \neq j$. The same argument also implies that, in the present case, (2) is also equivalent to that $\vec{m}[c_j^{*\mathrm{in}}] = \vec{0}$ or $\vec{m}[\mathsf{PK}_{r \to h}^{\mathrm{in}}] = \vec{0}$ for some $h \neq j$. Therefore, in Step 2, $\mathcal{D}'$ outputs $\top$ if and only if $\Phi = 0$, and $\mathcal{D}'$ outputs $\perp$ if and only if $\Phi|_{m_{b^*}=0} \neq 0$, as desired.

For Step 3, we note that whether (1) holds or not is preserved by the operations in the step, and the same also holds for (2). Therefore, owing to Step 3, we assume from now without loss of generality that

$$m[\mathsf{PK}_{r \to j}^{\mathrm{in}}; \widetilde{\gamma}_j^{\mathrm{out}}] = 1 \text{ for each } 1 \leq j \leq \ell^{\mathrm{in}} \ . \tag{3}$$

We study the behavior of the first-level loop recursively. Let I_0 denote the set of all $j \in I$ with $\widetilde{\gamma}_j^{\mathrm{out}} = \mathsf{const}$. We note that, for each $j \in I_0$, we have $m[\mathsf{PK}_{r \to j}^{\mathrm{in}}; \widetilde{\gamma}^{\mathrm{out}}] = 0$ for any $\widetilde{\gamma}^{\mathrm{out}} \neq \mathsf{const}$ and $m[\mathsf{PK}_{r \to j}^{\mathrm{in}}; \mathsf{const}] = 1$ owing to Step 3. Here we assume (as the recursion hypothesis) that, for the index set I in the first-level loop, we have $\Phi = 0$ if and only if

$$\prod_{j \in \{1,\dots,\ell^{\mathrm{in}}\} \setminus I} (\vec{m}[\mathsf{PK}_{r \to j}^{\mathrm{in}}] \cdot \mathcal{F}) \sum_{j \in I} (\vec{m}^I[c_j^{*\mathrm{in}}] \cdot \mathcal{F}) \prod_{j' \in I \setminus \{j\}} (\vec{m}[\mathsf{PK}_{r \to j'}^{\mathrm{in}}] \cdot \mathcal{F}) = 0 \ , \tag{4}$$

and $\Phi|_{m_{b^*}=0} = 0$ if and only if

$$\prod_{j \in \{1,\dots,\ell^{\mathrm{in}}\} \setminus I} (\vec{m}[\mathsf{PK}_{r \to j}^{\mathrm{in}}] \cdot \mathcal{F}|_{m_{b^*}=0}) \sum_{j \in I} (\vec{m}^I[c_j^{*\mathrm{in}}] \cdot \mathcal{F}|_{m_{b^*}=0}) \prod_{j' \in I \setminus \{j\}} (\vec{m}[\mathsf{PK}_{r \to j'}^{\mathrm{in}}] \cdot \mathcal{F}|_{m_{b^*}=0}) = 0 \ . \tag{5}$$

We note that the assumption is indeed satisfied at the initial choice of $I = \{1, \dots, \ell^{\mathrm{in}}\}$. Moreover, we also assume (as another recursion hypothesis) the following condition:

(*) For each $j \in I \setminus I_0$, let $\Pi(T^{\mathrm{in}}; j)$ denote the building-block scheme associated to the leaf j of T^{in}. Then we have $m^I[c_j^{*\mathrm{in}}; \gamma^{\mathrm{out}}] = 0$ and $m[\mathsf{PK}^{\mathrm{in}}_{r \to j}; \gamma^{\mathrm{out}}] = 0$ for any ciphertext component γ^{out} of the input for $C_k^{\rho'}$ with $\Pi(T^{\mathrm{out}}; \gamma^{\mathrm{out}}) \neq \Pi(T^{\mathrm{in}}; j)$, where $\Pi(T^{\mathrm{out}}; \gamma^{\mathrm{out}}) \in \mathfrak{S}$ denotes the building-block scheme satisfying that γ^{out} is a ciphertext for $\Pi(T^{\mathrm{out}}; \gamma^{\mathrm{out}})$.

By the latter part of Lemma 3, the assumption (*) is also satisfied at the initial choice of $I = \{1, \dots, \ell^{\mathrm{in}}\}$.

The key property in the analysis of $\mathcal{D}'$ is the following:

Lemma 10. *In the first-level loop, suppose that $|I| \geq 2$ and (5) is satisfied. Then the condition in Step 5(b)ii is not satisfied for some $j \in I$; hence, for the choice of j, the execution of the algorithm reaches Step 5(b)iii.*

Proof. First we note that, in the present case, the coefficient vectors $\vec{m}[\mathsf{PK}^{\mathrm{in}}_{r \to j}]$ for $1 \leq j \leq \ell^{\mathrm{in}}$ and $\vec{m}^I[c_j^{*\mathrm{in}}]$ for $j \in I$ are all invertible owing to Step 5a. Therefore, the condition (5) implies that

$$\sum_{j \in I} (\vec{m}^I[c_j^{*\mathrm{in}}] \cdot \mathcal{F}|_{m_{b^*}=0}) \prod_{j' \in I \setminus \{j\}} (\vec{m}[\mathsf{PK}^{\mathrm{in}}_{r \to j'}] \cdot \mathcal{F}|_{m_{b^*}=0}) = 0 \tag{6}$$

(note that $\varphi(\vec{m}[\mathsf{PK}^{\mathrm{in}}_{r \to j}]) \neq \vec{0}$ for any $j \in \{1, \dots, \ell^{\mathrm{in}}\} \setminus I$ by (3)). Moreover, now the coefficient vectors $\vec{m}^I[c_j^{*\mathrm{in}}] - m^I[c_j^{*\mathrm{in}}; \widetilde{\gamma}_j^{\mathrm{out}}] \cdot \vec{m}[\mathsf{PK}^{\mathrm{in}}_{r \to j}]$ and $\vec{m}[\mathsf{PK}^{\mathrm{in}}_{r \to h}] - m[\mathsf{PK}^{\mathrm{in}}_{r \to h}; \widetilde{\gamma}_j^{\mathrm{out}}] \cdot \vec{m}[\mathsf{PK}^{\mathrm{in}}_{r \to j}]$ for any $h \in I \setminus \{j\}$ are invertible owing to Step 5(b)i. Therefore, the claim is equivalent to the following; for some $j \in I$, we have

$$\varphi(\vec{m}^I[c_j^{*\mathrm{in}}]) = \varphi(m^I[c_j^{*\mathrm{in}}; \widetilde{\gamma}_j^{\mathrm{out}}]) \cdot \varphi(\vec{m}[\mathsf{PK}^{\mathrm{in}}_{r \to j}])$$

or

$$\varphi(\vec{m}[\mathsf{PK}^{\mathrm{in}}_{r \to h}]) = \varphi(m[\mathsf{PK}^{\mathrm{in}}_{r \to h}; \widetilde{\gamma}_j^{\mathrm{out}}]) \cdot \varphi(\vec{m}[\mathsf{PK}^{\mathrm{in}}_{r \to j}]) \text{ for some } h \in I \setminus \{j\} \ .$$

In the following argument, we use Lemma 8 and Lemma 9 (see also Tables 1 and 2).

Case 1: For some $j \in I$, $\vec{m}[\mathsf{PK}^{\mathrm{in}}_{r \to j}]$ is either of type I except type I-0 for $T^{\mathrm{out}} = T_2$ or of type III. By the shape of (6), the polynomial $\varphi(\vec{m}[\mathsf{PK}^{\mathrm{in}}_{r \to j}] \cdot \mathcal{F})|_{m_{b^*}=0}$ divides the product of $\varphi(\vec{m}^I[c_j^{*\mathrm{in}}] \cdot \mathcal{F})|_{m_{b^*}=0}$ and $\varphi(\vec{m}[\mathsf{PK}^{\mathrm{in}}_{r \to j'}] \cdot \mathcal{F})|_{m_{b^*}=0}$ over all $j' \in I \setminus \{j\}$. Now the claim above follows from Lemma 9 applied to $\widetilde{\gamma}_0{}^{\mathrm{out}} := \widetilde{\gamma}_j{}^{\mathrm{out}}$; recall (3).

Case 2: For every $j \in I$, $\vec{m}[\mathsf{PK}^{\mathrm{in}}_{r \to j}]$ is of type II except the last row in Table 1 (for $T^{\mathrm{out}} = T_2$). By Tables 1, 2 and (3), for each $j \in I$, we have $m[\mathsf{PK}^{\mathrm{in}}_{r \to j}; \widetilde{\gamma}_j{}^{\mathrm{out}}] = 1$ and $m[\mathsf{PK}^{\mathrm{in}}_{r \to j}; \widetilde{\gamma}^{\mathrm{out}}] = 0$ for every $\widetilde{\gamma}^{\mathrm{out}} \neq \widetilde{\gamma}_j{}^{\mathrm{out}}$. Moreover, if $T^{\mathrm{out}} = T_{\ell^{\mathrm{out}}}$ with $\ell^{\mathrm{out}} \neq 2$, then $\widetilde{\gamma}_j{}^{\mathrm{out}}$ is either const (i.e., $\varphi(\vec{m}[\mathsf{PK}^{\mathrm{in}}_{r \to j}] \cdot \mathcal{F})|_{m_{b^*}=0} = 1$) or $\mathsf{PK}^{\mathrm{out}}_{r \to i_j}$ for some $1 \leq i_j \leq \ell^{\mathrm{out}}$ (i.e., $\varphi(\vec{m}[\mathsf{PK}^{\mathrm{in}}_{r \to j}] \cdot \mathcal{F})|_{m_{b^*}=0} = \mathsf{SK}_{r \to i_j}$); while if $T^{\mathrm{out}} = T^{\S}$, then $\widetilde{\gamma}_j{}^{\mathrm{out}}$ is either const (i.e., $\varphi(\vec{m}[\mathsf{PK}^{\mathrm{in}}_{r \to j}] \cdot \mathcal{F})|_{m_{b^*}=0} = 1$) or $\mathsf{PK}^{\mathrm{out}}_{e_j}$ for some $e_j \in \{(r \to 1), (2 \to 3), (2 \to 4)\}$ (i.e., $\varphi(\vec{m}[\mathsf{PK}^{\mathrm{in}}_{r \to j}] \cdot \mathcal{F})|_{m_{b^*}=0} = \mathsf{SK}_{e_j}$). Hence, the claim holds if $\widetilde{\gamma}_j{}^{\mathrm{out}} = \widetilde{\gamma}_{j'}{}^{\mathrm{out}}$ for some distinct $j, j' \in I$. From now, we consider the other case that all $\widetilde{\gamma}_j{}^{\mathrm{out}}$ for $j \in I$ are different (in particular, $|I_0| \leq 1$). We have the following two cases.

Case 2-1: $T^{\mathrm{out}} = T^{\S}$. In this case, for each $j \in I \setminus I_0$, (6) implies that SK_{e_j} divides the product of $\varphi(\vec{m}^I[c_j^{*\mathrm{in}}] \cdot \mathcal{F})|_{m_{b^*}=0}$ and $\varphi(\vec{m}[\mathsf{PK}^{\mathrm{in}}_{r \to j'}] \cdot \mathcal{F})|_{m_{b^*}=0}$ over all $j' \in I \setminus \{j\}$. Since the indices $\widetilde{\gamma}_{j'}{}^{\mathrm{out}}$ for $j' \in I$ are all different, it follows from the shapes of $\varphi(\vec{m}[\mathsf{PK}^{\mathrm{in}}_{r\to j'}] \cdot \mathcal{F})|_{m_{b^*}=0}$ mentioned above that SK_{e_j} divides $\varphi(\vec{m}^I[c_j^{*\mathrm{in}}] \cdot \mathcal{F})|_{m_{b^*}=0}$. This implies that $\varphi(\vec{m}^I[c_j^{*\mathrm{in}}] \cdot \mathcal{F})|_{m_{b^*}=0} = \Xi_j \cdot \mathsf{SK}_{e_j}$, where Ξ_j is defined as

$$\Xi_j = \begin{cases} \varphi(m^I[c_j^{*\mathrm{in}}; \mathsf{PK}^{\mathrm{out}}_{r\to 1}]) - \varphi(m^I[c_j^{*\mathrm{in}}; c_1^{*\mathrm{out}}]) \cdot s_1 & \text{if } e_j = (r \to 1) \ , \\ \varphi(m^I[c_j^{*\mathrm{in}}; \mathsf{PK}^{\mathrm{out}}_{2\to 3}]) + \varphi(m^I[c_j^{*\mathrm{in}}; \mathsf{PK}^{\mathrm{out}(1)}_{r\to 2}]) \cdot s_{\mathrm{pk}} & \\ \qquad +\varphi(m^I[c_j^{*\mathrm{in}}; c_3^{*\mathrm{out}}]) \cdot (-s_1 \cdot s_{\mathrm{pk}} + s_2) & \text{if } e_j = (2 \to 3) \ , \\ \varphi(m^I[c_j^{*\mathrm{in}}; \mathsf{PK}^{\mathrm{out}}_{2\to 4}]) + \varphi(m^I[c_j^{*\mathrm{in}}; \mathsf{PK}^{\mathrm{out}(2)}_{r\to 2}]) \cdot (\mathsf{SK}_{r\to 2} - s_{\mathrm{pk}}) & \\ \qquad +\varphi(m^I[c_j^{*\mathrm{in}}; c_4^{*\mathrm{out}}])(-s_1 \cdot (\mathsf{SK}_{r\to 2} - s_{\mathrm{pk}}) - s_2) & \text{if } e_j = (2 \to 4) \ . \end{cases}$$

If $I_0 = \emptyset$, then the equality (6) implies that

$$\sum_{j\in I} \Xi_j \cdot \mathsf{SK}_{e_j} \prod_{j' \in I \setminus \{j\}} \mathsf{SK}_{e_{j'}} = \left(\sum_{j\in I} \Xi_j \right) \prod_{j \in I} \mathsf{SK}_{e_j} = 0 \ ,$$

therefore we have $\sum_{j\in I} \Xi_j = 0$. On the other hand, if I_0 consists of a unique element, say j_0, then the equality (6) implies that

$$\begin{aligned} &\sum_{j\in I\setminus\{j_0\}} \Xi_j \cdot \mathsf{SK}_{e_j} \prod_{j' \in I \setminus \{j, j_0\}} \mathsf{SK}_{e_{j'}} + \vec{m}^I[c_{j_0}^{*\ \mathrm{in}}] \cdot \mathcal{F}|_{m_{b^*}=0} \prod_{j' \in I \setminus \{j_0\}} \mathsf{SK}_{e_{j'}} \\ &= \left(\sum_{j\in I\setminus\{j_0\}} \Xi_j + \vec{m}^I[c_{j_0}^{*\ \mathrm{in}}] \cdot \mathcal{F}|_{m_{b^*}=0} \right) \prod_{j \in I \setminus \{j_0\}} \mathsf{SK}_{e_j} = 0 \ , \end{aligned}$$

therefore we have $\sum_{j\in I\setminus\{j_0\}} \Xi_j + \vec{m}^I[c_{j_0}^{*\ \mathrm{in}}] \cdot \mathcal{F}|_{m_{b^*}=0} = 0$. In any case, by the shape of each polynomial, the equality above holds only when Ξ_j is constant for some $j \in I \setminus I_0$. Now the claim holds for the $j \in I$. Hence the claim holds in Case 2-1.

Case 2-2: $T^{\mathrm{out}} = T_{\ell^{\mathrm{out}}}$**.** In this case, for each $j \in I \setminus I_0$, (6) implies that $\mathsf{SK}_{r\to i_j}$ divides the product of $\varphi(\vec{m}^I[c_j^{*\mathrm{in}}] \cdot \mathcal{F})|_{m_{b^*}=0}$ and $\varphi(\vec{m}[\mathsf{PK}^{\mathrm{in}}_{r\to j'}] \cdot \mathcal{F})|_{m_{b^*}=0}$ over all $j' \in I \setminus \{j\}$. Since now the indices $i_{j'}$ for $j' \in I$ are all different, it follows that $\mathsf{SK}_{r\to i_j}$ divides $\varphi(\vec{m}^I[c_j^{*\mathrm{in}}] \cdot \mathcal{F})|_{m_{b^*}=0}$. By the shape of the polynomial, this implies that $\varphi(\vec{m}^I[c_j^{*\mathrm{in}}] \cdot \mathcal{F})|_{m_{b^*}=0} = \Xi_j \mathsf{SK}_{r\to i_j}$, where Ξ_j is defined as

$$\Xi_j = \begin{cases} \varphi(m^I[c_j^{*\mathrm{in}}; c_{i_j}^{*\ \mathrm{out}}]) s_{i_j} + \varphi(m^I[c_j^{*\mathrm{in}}; \mathsf{PK}^{\mathrm{out}}_{r\to i_j}]) & \text{if } i_j \neq \ell^{\mathrm{out}} \ , \\ -\varphi(m^I[c_j^{*\mathrm{in}}; c_{\ell^{\mathrm{out}}}^{*\ \mathrm{out}}]) \sum_{h=1}^{\ell^{\mathrm{out}}-1} s_h + \varphi(m^I[c_j^{*\mathrm{in}}; \mathsf{PK}^{\mathrm{out}}_{r\to \ell^{\mathrm{out}}}]) & \text{if } i_j = \ell^{\mathrm{out}} \ . \end{cases}$$

If $I_0 = \emptyset$, then the equality (6) implies that

$$\sum_{j\in I} \Xi_j \mathsf{SK}_{r\to i_j} \prod_{j' \in I \setminus \{j\}} \mathsf{SK}_{r\to i_{j'}} = \left(\sum_{j\in I} \Xi_j \right) \prod_{j \in I} \mathsf{SK}_{r\to i_j} = 0 \ ,$$

therefore we have $\sum_{j\in I} \Xi_j = 0$. On the other hand, if I_0 consists of a unique element, say j_0, then the equality (6) implies that

$$\begin{aligned} &\sum_{j\in I\setminus\{j_0\}} \Xi_j \mathsf{SK}_{r\to i_j} \prod_{j' \in I \setminus \{j, j_0\}} \mathsf{SK}_{r\to i_{j'}} + \varphi(\vec{m}^I[c_{j_0}^{*\ \mathrm{in}}] \cdot \mathcal{F})|_{m_{b^*}=0} \prod_{j' \in I \setminus \{j_0\}} \mathsf{SK}_{r\to i_{j'}} \\ &= \left(\sum_{j\in I\setminus\{j_0\}} \Xi_j + \varphi(\vec{m}^I[c_{j_0}^{*\ \mathrm{in}}] \cdot \mathcal{F})|_{m_{b^*}=0} \right) \prod_{j \in I \setminus \{j_0\}} \mathsf{SK}_{r\to i_j} = 0 \ , \end{aligned}$$

therefore we have $\sum_{j \in I \setminus \{j_0\}} \Xi_j + \varphi(\vec{m}^I[c^{*\,\mathrm{in}}_{j_0}] \cdot \mathcal{F})|_{m_{b^*}=0} = 0$.

In any case, if Ξ_j is not constant for every $j \in I \setminus I_0$, then by the shape of each polynomial, the terms of polynomials Ξ_j with $j \in I \setminus I_0$ involving any of the variables $s_1, \dots, s_{\ell^{\mathrm{out}}-1}$ should be cancelled within the sum $\sum_{j \in I \setminus I_0} \Xi_j$. Since all the indices i_j are different, it follows that $i_j = \ell^{\mathrm{out}}$ for some $j \in I \setminus I_0$, and for each $1 \leq h \leq \ell^{\mathrm{out}} - 1$, the non-zero coefficient $-\varphi(m^I[c^{*\mathrm{in}}_j; c^{*}_{\ell^{\mathrm{out}}}{}^{\mathrm{out}}])$ of s_h in Ξ_j is cancelled by the (non-zero) coefficient $\varphi(m^I[c^{*\,\mathrm{in}}_{j_h}; c^{*\mathrm{out}}_h])$ of s_h in some Ξ_{j_h} with $j_h \in I \setminus (I_0 \cup \{j\})$ and $i_{j_h} = h$. We note that j and these j_h are all different, since $i_j = \ell^{\mathrm{out}}$ and $i_{j_h} = h$ are all different, too. Now by the condition (*), we have $\Pi(T^{\mathrm{out}}; c^{*}_{\ell^{\mathrm{out}}}{}^{\mathrm{out}}) = \Pi(T^{\mathrm{in}}; j)$ and $\Pi(T^{\mathrm{out}}; c^{*\mathrm{out}}_h) = \Pi(T^{\mathrm{in}}; j_h)$ for any $1 \leq h \leq \ell^{\mathrm{out}} - 1$. This means that, for each leaf of $T^{\mathrm{out}} = T_{\ell^{\mathrm{out}}}$, the building-block scheme associated to the leaf is equal to the one associated to a leaf of $T^{\mathrm{in}} = T_{\ell^{\mathrm{in}}}$, and the latter leaves are all different. This implies that $T^{\mathrm{out}} \preceq T^{\mathrm{in}}$, which contradicts the condition for T^{in} in the statement of Theorem 4.

By the previous paragraph, it follows that Ξ_j is constant for some $j \in I \setminus I_0$. Now the claim holds for the $j \in I$. Hence the claim holds in Case 2-2, therefore in Case 2.

Case 3: $T^{\mathrm{out}} = T_2$, and the case is different from Cases 1 and 2. Now for each $j \in I$, $\vec{m}[\mathsf{PK}^{\mathrm{in}}_{r \to j}]$ is either of type I-0 (see Table 1) or of type II. Moreover, there is some $j_0 \in I$ for which $\vec{m}[\mathsf{PK}^{\mathrm{in}}_{r \to j_0}]$ is either of type I-0 or as in the last row of type II in Table 1.

If $\vec{m}[\mathsf{PK}^{\mathrm{in}}_{r \to j_0}]$ is of type I-0, then by Table 1, we have $\varphi(m[\mathsf{PK}^{\mathrm{in}}_{r \to j_0}; c^{*\mathrm{out}}_1]) \neq 0$ and $\varphi(m[\mathsf{PK}^{\mathrm{in}}_{r \to j_0}; c^{*\mathrm{out}}_2]) \neq 0$. On the other hand, if $\vec{m}[\mathsf{PK}^{\mathrm{in}}_{r \to j_0}]$ is as in the last row of type II in Table 1, then we have $\varphi(m[\mathsf{PK}^{\mathrm{in}}_{r \to j_0}; \mathsf{PK}^{\mathrm{out}}_{r \to 1}]) \neq 0$ and $\varphi(m[\mathsf{PK}^{\mathrm{in}}_{r \to j_0}; \mathsf{PK}^{\mathrm{out}}_{r \to 2}]) \neq 0$. In any case, the condition (*) implies that, for each of the two leaves v of $T^{\mathrm{out}} = T_2$, the building-block scheme $\Pi(T^{\mathrm{out}}; v)$ associated to the leaf is the same as the building-block scheme $\Pi(T^{\mathrm{in}}; j_0)$ associated to the leaf j_0 in T^{in}.

Now if $I \setminus I_0$ contains some element $j \neq j_0$, then we have $\varphi(m[\mathsf{PK}^{\mathrm{in}}_{r \to j}; \widetilde{\gamma}_j{}^{\mathrm{out}}]) \neq 0$ and $\widetilde{\gamma}_j{}^{\mathrm{out}} \neq \mathsf{const}$, therefore the condition (*) implies that $\Pi(T^{\mathrm{in}}; j) = \Pi(T^{\mathrm{out}}; v) = \Pi(T^{\mathrm{in}}; j_0)$. By the result of the previous paragraph, this implies that $T^{\mathrm{out}} \preceq T^{\mathrm{in}}$, contradicting the condition in the statement of Theorem 4. Hence, we have $I \setminus I_0 = \{j_0\}$. Moreover, if I_0 has two or more elements, then the claim holds for any $j \in I_0$. From now, we consider the other case that $|I_0| \leq 1$. Since $|I| \geq 2$ by the halting condition of the first-level loop, it follows that $|I_0| = 1$ and $|I| = 2$; let j_1 denote the unique element of I_0.

Now the equality (6) implies that

$$\varphi(\vec{m}^I[c^{*\,\mathrm{in}}_{j_0}] \cdot \mathcal{F})|_{m_{b^*}=0} + \varphi(\vec{m}^I[c^{*\,\mathrm{in}}_{j_1}] \cdot \mathcal{F})|_{m_{b^*}=0} \cdot \varphi(\vec{m}[\mathsf{PK}^{\mathrm{in}}_{r \to j_0}] \cdot \mathcal{F})|_{m_{b^*}=0} = 0 \enspace .$$

By the shape of $\varphi(\vec{m}[\mathsf{PK}^{\mathrm{in}}_{r \to j_0}] \cdot \mathcal{F})|_{m_{b^*}=0}$ mentioned above, both $\mathsf{SK}_{r \to 1}$ and $\mathsf{SK}_{r \to 2}$ have degree one in $\varphi(\vec{m}[\mathsf{PK}^{\mathrm{in}}_{r \to j_0}] \cdot \mathcal{F})|_{m_{b^*}=0}$, while both $\mathsf{SK}_{r \to 1}$ and $\mathsf{SK}_{r \to 2}$ have degree at most one in $\varphi(\vec{m}^I[c^{*\,\mathrm{in}}_{j_0}] \cdot \mathcal{F})|_{m_{b^*}=0}$. This implies that both $\mathsf{SK}_{r \to 1}$ and $\mathsf{SK}_{r \to 2}$ have degree zero in $\varphi(\vec{m}^I[c^{*\,\mathrm{in}}_{j_1}] \cdot \mathcal{F})|_{m_{b^*}=0}$, i.e., $\varphi(\vec{m}^I[c^{*\,\mathrm{in}}_{j_1}] \cdot \mathcal{F})|_{m_{b^*}=0}$ is constant as well as $\varphi(\vec{m}^I[\mathsf{PK}^{\mathrm{in}}_{r \to j_1}] \cdot \mathcal{F})|_{m_{b^*}=0}$. Hence the claim holds for the $j_1 \in I$, therefore the claim holds in Case 3.

This completes the proof of Lemma 10. $\square$

By Lemma 10, if the algorithm $\mathcal{D}'$ outputs $\perp$ at Step 5c in the first-level loop with the index set I, then the condition in (5) is not satisfied. By the assumption given before Lemma 10, this implies that $\Phi|_{m_{b^*}=0} \neq 0$, therefore $\Phi \neq 0$, as desired. From now, we consider the other case that the execution of the second-level loop with index set I reaches Step 5(b)iii with index $j \in I$.

First, we consider the case that the condition $\vec{m}^I[c_j^{*\mathrm{in}}] = m^I[c_j^{*\mathrm{in}}; \widetilde{\gamma}_j{}^{\mathrm{out}}] \cdot \vec{m}[\mathsf{PK}_{r\to j}^{\mathrm{in}}]$ in Step 5(b)iii is satisfied. Then for the element $h \in I'$ as in Step 5(b)iii, the polynomial $\sum_{p\in I}(\vec{m}^I[c_p^{*\mathrm{in}}] \cdot \mathcal{F}|_{m_{b^*}=0}) \prod_{p'\in I\setminus\{p\}}(\vec{m}[\mathsf{PK}_{r\to p'}^{\mathrm{in}}] \cdot \mathcal{F}|_{m_{b^*}=0})$ is equal to

$$\begin{aligned}
& m^I[c_j^{*\mathrm{in}}; \widetilde{\gamma}_j{}^{\mathrm{out}}] \cdot \vec{m}[\mathsf{PK}_{r\to j}^{\mathrm{in}}] \cdot \mathcal{F}|_{m_{b^*}=0} \prod_{h'\in I\setminus\{j\}} \vec{m}[\mathsf{PK}_{r\to h'}^{\mathrm{in}}] \cdot \mathcal{F}|_{m_{b^*}=0} \\
&\quad + \left(\vec{m}^{I'}[c_h^{*\mathrm{in}}] - m^I[c_j^{*\mathrm{in}}; \widetilde{\gamma}_j{}^{\mathrm{out}}] \cdot \vec{m}[\mathsf{PK}_{r\to h}^{\mathrm{in}}]\right) \cdot \mathcal{F}|_{m_{b^*}=0} \prod_{h'\in I\setminus\{h\}} \vec{m}[\mathsf{PK}_{r\to h'}^{\mathrm{in}}] \cdot \mathcal{F}|_{m_{b^*}=0} \\
&\quad + \sum_{j'\in I\setminus\{j,h\}} \vec{m}^{I'}[c_{j'}^{*\mathrm{in}}] \cdot \mathcal{F}|_{m_{b^*}=0} \prod_{h'\in I\setminus\{j'\}} \vec{m}[\mathsf{PK}_{r\to h'}^{\mathrm{in}}] \cdot \mathcal{F}|_{m_{b^*}=0} \\
&= \vec{m}[\mathsf{PK}_{r\to j}^{\mathrm{in}}] \cdot \mathcal{F}|_{m_{b^*}=0} \cdot \sum_{j'\in I'} \vec{m}^{I'}[c_{j'}^{*\mathrm{in}}] \cdot \mathcal{F}|_{m_{b^*}=0} \prod_{h'\in I'\setminus\{j'\}} \vec{m}[\mathsf{PK}_{r\to h'}^{\mathrm{in}}] \cdot \mathcal{F}|_{m_{b^*}=0} \enspace,
\end{aligned} \tag{7}$$

therefore the left-hand side of (5) for the index set I' is equal to that for the index set I. Similarly, the left-hand side of (4) for the case of the set I' is equal to that for the case of the set I. Moreover, the condition (*) for $\vec{m}^{I'}[c_h^{*\mathrm{in}}]$ is implied by the condition (*) for $\vec{m}^I[c_h^{*\mathrm{in}}]$ and $\vec{m}[\mathsf{PK}_{r\to h}^{\mathrm{in}}]$; the indices at which the components become zero due to the condition (*) are common for $\vec{m}^I[c_h^{*\mathrm{in}}]$ and $\vec{m}[\mathsf{PK}_{r\to h}^{\mathrm{in}}]$. Hence, the assumption given before Lemma 10 is also satisfied for the case of index set I'.

Secondly, we consider the case that the execution of the second-level loop with index set I and index $j \in I$ reaches Step 5(b)iv. Let h be an element of $I' = I \setminus \{j\}$ as in Step 5(b)iv. Then $\sum_{p\in I} \vec{m}^I[c_p^{*\mathrm{in}}] \cdot \mathcal{F}|_{m_{b^*}=0} \prod_{p'\in I\setminus\{p\}} \vec{m}[\mathsf{PK}_{r\to p'}^{\mathrm{in}}] \cdot \mathcal{F}|_{m_{b^*}=0}$ is equal to

$$\begin{aligned}
& \left(\vec{m}^{I'}[c_h^{*\mathrm{in}}] - m[\mathsf{PK}_{r\to h}^{\mathrm{in}}; \widetilde{\gamma}_j{}^{\mathrm{out}}] \cdot \vec{m}^I[c_j^{*\mathrm{in}}]\right) \cdot \mathcal{F}|_{m_{b^*}=0} \prod_{h'\in I\setminus\{h\}} \vec{m}[\mathsf{PK}_{r\to h'}^{\mathrm{in}}] \cdot \mathcal{F}|_{m_{b^*}=0} \\
&\quad + (\vec{m}^I[c_j^{*\mathrm{in}}] \cdot \mathcal{F}|_{m_{b^*}=0}) \cdot m[\mathsf{PK}_{r\to h}^{\mathrm{in}}; \widetilde{\gamma}_j{}^{\mathrm{out}}] \cdot (\vec{m}[\mathsf{PK}_{r\to j}^{\mathrm{in}}] \cdot \mathcal{F}|_{m_{b^*}=0}) \prod_{h'\in I\setminus\{j,h\}} \vec{m}[\mathsf{PK}_{r\to h'}^{\mathrm{in}}] \cdot \mathcal{F}|_{m_{b^*}=0} \\
&\quad + \sum_{j'\in I\setminus\{j,h\}} \vec{m}^I[c_{j'}^{*\mathrm{in}}] \cdot \mathcal{F}|_{m_{b^*}=0} \prod_{h'\in I\setminus\{j'\}} \vec{m}[\mathsf{PK}_{r\to h'}^{\mathrm{in}}] \cdot \mathcal{F}|_{m_{b^*}=0} \\
&= \vec{m}[\mathsf{PK}_{r\to j}^{\mathrm{in}}] \cdot \mathcal{F}|_{m_{b^*}=0} \cdot \sum_{j'\in I'} \vec{m}^{I'}[c_{j'}^{*\mathrm{in}}] \cdot \mathcal{F}|_{m_{b^*}=0} \prod_{h'\in I'\setminus\{j'\}} \vec{m}[\mathsf{PK}_{r\to h'}^{\mathrm{in}}] \cdot \mathcal{F}|_{m_{b^*}=0} \enspace,
\end{aligned} \tag{8}$$

therefore the left-hand side of (5) for the index set I' is equal to that for the index set I. Similarly, the left-hand side of (4) for the case of the set I' is equal to that for the case of the set I. Moreover, for the condition (*), if $h \in I \setminus I_0$, then the component of $\vec{m}[\mathsf{PK}_{r\to h}^{\mathrm{in}}] = m[\mathsf{PK}_{r\to h}^{\mathrm{in}}; \widetilde{\gamma}_j{}^{\mathrm{out}}] \cdot \vec{m}[\mathsf{PK}_{r\to j}^{\mathrm{in}}]$ at some index $\widetilde{\gamma}^{\mathrm{out}} \neq \mathsf{const}$ is non-zero, therefore $\vec{m}[\mathsf{PK}_{r\to j}^{\mathrm{in}}]$ has the same property, hence $j \not\in I_0$, and $m[\mathsf{PK}_{r\to h}^{\mathrm{in}}; \widetilde{\gamma}_j{}^{\mathrm{out}}] \neq 0$. Now the property $j \not\in I_0$ implies that $\widetilde{\gamma}_j{}^{\mathrm{out}} \neq \mathsf{const}$, therefore we have $\Pi(T^{\mathrm{in}}; j) = \Pi(T^{\mathrm{out}}; \widetilde{\gamma}_j{}^{\mathrm{out}}) = \Pi(T^{\mathrm{in}}; h)$. This implies that the indices at which the components become zero due to the condition (*) are common for $\vec{m}^I[c_h^{*\mathrm{in}}]$ and $\vec{m}^I[c_j^{*\mathrm{in}}]$. Hence, the assumption given before Lemma 10 is also satisfied for the case of index set I'.

By the result above, a recursive argument implies that the desired relations between the output of $\mathcal{D}'$ and the polynomials Φ and $\Phi|_{m_{b^*}=0}$ hold for the case that the execution of $\mathcal{D}'$ halts during the first-level loop; and, for the other case, after the first-level loop which is finished by achieving the halting condition $|I| = 1$, say $I = \{j\}$, we have $\Phi = 0$

if and only if

$$\left(\prod_{h\in\{1,\dots,\ell^{\mathrm{in}}\}\setminus\{j\}} \vec{m}[\mathsf{PK}^{\mathrm{in}}_{r\to h}]\cdot\mathcal{F}\right)\cdot(\vec{m}^I[c_j^{*\mathrm{in}}]\cdot\mathcal{F})=0\ , \tag{9}$$

and we have $\Phi|_{m_{b^*}=0}=0$ if and only if

$$\left(\prod_{h\in\{1,\dots,\ell^{\mathrm{in}}\}\setminus\{j\}} \vec{m}[\mathsf{PK}^{\mathrm{in}}_{r\to h}]\cdot\mathcal{F}|_{m_{b^*}=0}\right)\cdot(\vec{m}^I[c_j^{*\mathrm{in}}]\cdot\mathcal{F}|_{m_{b^*}=0})=0\ . \tag{10}$$

Moreover, since $\mathcal{D}'$ does not output an element of $\mathcal{M}\setminus(\mathcal{M}^\times\cup\{0\})$, the coefficient vector $\vec{m}^I[c_j^{*\mathrm{in}}]$ is invertible (see Step 6) as well as the vectors $\vec{m}[\mathsf{PK}^{\mathrm{in}}_{r\to h}]$. By the property, if (9) is satisfied, then we have either $\varphi(\vec{m}[\mathsf{PK}^{\mathrm{in}}_{r\to h}]\cdot\mathcal{F})=0$ for some $h\in\{1,\dots,\ell^{\mathrm{in}}\}\setminus\{j\}$ or $\varphi(\vec{m}^I[c_j^{*\mathrm{in}}]\cdot\mathcal{F})=0$, which also implies (by (3)) that $\varphi(\vec{m}^I[c_j^{*\mathrm{in}}])=\vec{0}$, hence $\vec{m}^I[c_j^{*\mathrm{in}}]=\vec{0}$. Conversely, the condition $\vec{m}^I[c_j^{*\mathrm{in}}]=\vec{0}$ implies (9). Therefore, we have $\Phi=0$ if and only if $\vec{m}^I[c_j^{*\mathrm{in}}]=\vec{0}$. The same argument (using (10) instead of (9)) also implies that we have $\Phi|_{m_{b^*}=0}=0$ if and only if $\vec{m}^I[c_j^{*\mathrm{in}}]=\vec{0}$. Hence, $\mathcal{D}'$ outputs $\top$ at Step 7 if and only if $\Phi=0$; and $\mathcal{D}'$ outputs $\perp$ at Step 8 if and only if $\Phi|_{m_{b^*}=0}\neq 0$.

Summarizing, we have proven that $\mathcal{D}'$ outputs $\top$ if and only if $\Phi=0$; and it outputs $\perp$ if and only if $\Phi|_{m_{b^*}=0}\neq 0$, as desired. This completes the proof of Theorem 4.

8 On Non-Monotonicity of Combined Security

In a generic construction of a cryptographic scheme from some building-block primitives, it would be naively expected in general that the superiority/inferiority of building-block primitives (in a certain sense) is monotonically inherited by the resulting scheme; namely, if a building-block primitive is superior to some other primitive, then the scheme constructed from the former primitive would also be superior to the one constructed from the latter primitive. In this section, based on our results in the previous sections, we construct an example which shows that, despite the natural expectation above, such a monotonicity in generic constructions does *not* always hold.

More precisely, in this section, we construct four RMHE schemes Π_1, Π_2, Π_3 and Π_4 with the following properties, where ZPA(Π) means the ZPA security for Π, ZPA(Π) $\to$ ZPA(Π') means that the ZPA security for Π implies the ZPA security for Π', and ZPA(Π) $\not\to$ ZPA(Π') means that the ZPA security for Π does not imply the ZPA security for Π' (here the non-implication relations are considered in our proposed computational model, which is used in our proof of Theorem 4):

- We have ZPA(Π_1) $\leftarrow$ ZPA(Π_3), ZPA(Π_1) $\not\to$ ZPA(Π_3), ZPA(Π_2) $\leftarrow$ ZPA(Π_4) and ZPA(Π_2) $\not\to$ ZPA(Π_4).
- We have ZPA($\Gamma(\Pi_1,\Pi_2)$) $\to$ ZPA($\Gamma(\Pi_3,\Pi_4)$) and ZPA($\Gamma(\Pi_1,\Pi_2)$) $\not\leftarrow$ ZPA($\Gamma(\Pi_3,\Pi_4)$), where $\Gamma(\Pi,\Pi')$ denotes the RMHE scheme obtained by our proposed construction indexed by the tree T_2 with two building-block schemes Π,Π' associated to the two leaves of T_2.

In other words, for our generic construction of RMHE schemes, when some building-block schemes (Π_3 and Π_4 above) require strictly *stronger* underlying assumptions than other building-block schemes (Π_1 and Π_2 above), the required assumption for the scheme constructed from the former building-block schemes can even be strictly *weaker* than the scheme constructed from the latter. Moreover, we also show that such an example can be constructed even in a way that Π_3 and Π_4 are absolutely *not* ZPA secure; now the scheme $\Gamma(\Pi_3, \Pi_4)$ generated from primitives Π_3, Π_4 which are known to be *insecure* is even more reliable than the scheme $\Gamma(\Pi_1, \Pi_2)$ generated from ordinary primitives Π_1, Π_2.

We explain the construction of the example. Let Π'_0 and Π'_1 be any combinable RMHE schemes with common plaintext space $\mathcal{M}$, which satisfies Assumption 1 in Section 7. Now we construct an RMHE scheme Λ_0 in the following manner: The key pair is generated by the joint key distribution for Π'_0 and Π'_1 introduced in Definition 5, and the plaintext space is $\mathcal{M}^2$. A ciphertext of $(m_0, m_1) \in \mathcal{M}^2$ is given by $(\mathsf{Enc}_{\Pi'_0}(m_0), \mathsf{Enc}_{\Pi'_1}(m_1))$, and the decryption is performed by decrypting each component of the ciphertext by the decryption algorithms of Π'_0 and Π'_1. The homomorphic operations are defined in a component-wise manner as follows:

$$\mathsf{Add}_{\Lambda_0}((c_0, c_1), (c'_0, c'_1)) := (\mathsf{Add}_{\Pi'_0}(c_0, c'_0), \mathsf{Add}_{\Pi'_1}(c_1, c'_1)) \ ,$$
$$\mathsf{Mult}_{\Lambda_0}((m_0, m_1), (c_0, c_1)) := (\mathsf{Mult}_{\Pi'_0}(m_0, c_0), \mathsf{Mult}_{\Pi'_1}(m_1, c_1)) \ ,$$
$$\mathsf{Rerand}_{\Lambda_0}(c_0, c_1) := (\mathsf{Rerand}_{\Pi'_0}(c_0), \mathsf{Rerand}_{\Pi'_1}(c_1)) \ .$$

We also define an RMHE scheme Λ_1 by exchanging the order of the two components in Λ_0; i.e., a ciphertext of (m_0, m_1) is given by $(\mathsf{Enc}_{\Pi'_1}(m_0), \mathsf{Enc}_{\Pi'_0}(m_1))$.

The following property is obvious by the definition:

Lemma 11. *In the setting, we have* $\mathrm{ZPA}(\Lambda_i) \to \mathrm{ZPA}(\Pi'_j)$ *for any* $i \in \{0,1\}$ *and* $j \in \{0,1\}$.

On the other hand, we consider the RMHE scheme $\Gamma(\Lambda_0, \Lambda_1)$ combining Λ_0 and Λ_1; we note that Λ_0 and Λ_1 are combinable, where the joint key distribution generates the same key pair for both Λ_0 and Λ_1. Namely, a public key for the scheme consists of

$$\mathsf{PK}_0 = (\mathsf{PK}_{0,0}, \mathsf{PK}_{0,1}) = \left(\mathsf{Enc}_{\Pi'_0}(\mathcal{U}(\mathcal{M}^\times)), \mathsf{Enc}_{\Pi'_1}(\mathcal{U}(\mathcal{M}^\times))\right) \ ,$$
$$\mathsf{PK}_1 = (\mathsf{PK}_{1,0}, \mathsf{PK}_{1,1}) = \left(\mathsf{Enc}_{\Pi'_1}(\mathcal{U}(\mathcal{M}^\times)), \mathsf{Enc}_{\Pi'_0}(\mathcal{U}(\mathcal{M}^\times))\right)$$

as well as public keys for Π'_0 and Π'_1, where $\mathcal{U}(X)$ denotes the uniform distribution on a set X. A ciphertext of $(m_0, m_1) \in \mathcal{M}^2$ for the scheme is a pair of ciphertexts

$$\left(\mathsf{Rerand}_{\Pi'_0}(\mathsf{Mult}_{\Pi'_0}(s_0, \mathsf{PK}_{0,0})), \mathsf{Rerand}_{\Pi'_1}(\mathsf{Mult}_{\Pi'_1}(s_1, \mathsf{PK}_{0,1}))\right) \text{ for } \Lambda_0 \ ,$$
$$\left(\mathsf{Rerand}_{\Pi'_1}(\mathsf{Mult}_{\Pi'_1}(s'_0, \mathsf{PK}_{1,0})), \mathsf{Rerand}_{\Pi'_0}(\mathsf{Mult}_{\Pi'_0}(s'_1, \mathsf{PK}_{1,1}))\right) \text{ for } \Lambda_1 \ ,$$

where s_0, s_1, s'_0 and s'_1 are randomly chosen from $\mathcal{M}$ in such a way that $m_0 = s_0 + s'_0$ and $m_1 = s_1 + s'_1$. Then we have the following:

Lemma 12. *We have* $\mathrm{ZPA}(\Pi'_0) \to \mathrm{ZPA}(\Gamma(\Lambda_0, \Lambda_1))$ *and* $\mathrm{ZPA}(\Pi'_1) \to \mathrm{ZPA}(\Gamma(\Lambda_0, \Lambda_1))$.

Proof. Given a PPT adversary $\mathcal{A}$ for the ZPA game for $\Gamma(\Lambda_0, \Lambda_1)$, we construct a PPT adversary $\mathcal{A}^\dagger$ for the ZPA game for Π'_0 in the following manner. Given a public key $\mathsf{pk}_{\Pi'_0}$ for Π'_0 and a challenge ciphertext c^* corresponding to the challenge bit b^* in the ZPA game for Π'_0, the algorithm $\mathcal{A}^\dagger$ first generates a public key $\mathsf{pk}_{\Pi'_1}$ for Π'_1 by using the algorithm $\mathsf{ExpandKey}$ in Definition 5 (associated to the combinable set $\{\Pi'_0, \Pi'_1\}$ of RMHE schemes), and generates the other two components PK_0 and PK_1 of a public key pk for $\Gamma(\Lambda_0, \Lambda_1)$ by

$$\mathsf{PK}_0 = (\mathsf{PK}_{0,0}, \mathsf{PK}_{0,1}) \leftarrow \left(\mathsf{Rerand}_{\Pi'_0}(\mathsf{Mult}_{\Pi'_0}(\mathcal{U}(\mathcal{M}^\times), c^*)), \mathsf{Enc}_{\Pi'_1}(\mathcal{U}(\mathcal{M}^\times))\right) \ ,$$
$$\mathsf{PK}_1 = (\mathsf{PK}_{1,0}, \mathsf{PK}_{1,1}) \leftarrow \left(\mathsf{Enc}_{\Pi'_1}(\mathcal{U}(\mathcal{M}^\times)), \mathsf{Rerand}_{\Pi'_0}(\mathsf{Mult}_{\Pi'_0}(\mathcal{U}(\mathcal{M}^\times), c^*))\right) \ .$$

Then the algorithm chooses $b^\dagger \leftarrow_R \{0,1\}$, and sets $(m_0, m_1) := (0,0) \in \mathcal{M}^2$ if $b^\dagger = 0$ and $(m_0, m_1) \leftarrow_R \mathcal{M}^2$ if $b^\dagger = 1$. Moreover, the algorithm generates $c^\dagger \leftarrow \mathsf{Enc}_{\Gamma(\Lambda_0,\Lambda_1)}((m_0, m_1))$, executes $\mathcal{A}$ with challenge input $c^\dagger$ and obtains its output bit b'. Finally, the algorithm outputs the bit $b := b^\dagger \,\mathsf{XOR}\, b' \,\mathsf{XOR}\, 1$.

We investigate the behavior of the algorithm $\mathcal{A}^\dagger$ above. In the case $b^* = 1$, since $|\mathcal{M}^\times|/|\mathcal{M}|$ is overwhelming (see Assumption 1 in Section 7), the distribution of c^* is statistically close to $c^{**} \leftarrow \mathsf{Enc}_{\Pi'_0}(\mathcal{U}(\mathcal{M}^\times))$. On the other hand, the plaintext for the ciphertext $\mathsf{Mult}_{\Pi'_0}(\mathcal{U}(\mathcal{M}^\times), c^{**})$ is uniformly random over $\mathcal{M}^\times$ and is independent of c^{**}. This implies that the distributions of PK_0 and PK_1 in the algorithm are statistically close to those in a correctly generated public key for $\Gamma(\Lambda_0, \Lambda_1)$. Moreover, since $b^* = 1$, we have $b = b^*$ if and only if $b' = b^\dagger$. Therefore, the difference between $|\Pr[b = b^* \mid b^* = 1] - 1/2|$ and $\mathrm{Adv}_{\mathcal{A}}(k)$ is negligible.

In the other case $b^* = 0$, both of $\mathsf{PK}_{0,0}$ and $\mathsf{PK}_{1,1}$ are ciphertexts of plaintext $0 \in \mathcal{M}$. Then, by choosing s_0, s_1, s'_0 and s'_1 as in the definition of the encryption for $\Gamma(\Lambda_0, \Lambda_1)$ described above, the distributions of the two components of $c^\dagger$ are identical to

$$\left(\mathsf{Enc}_{\Pi'_0}(0), \mathsf{Rerand}_{\Pi'_1}(\mathsf{Mult}_{\Pi'_1}(s_1, \mathsf{PK}_{0,1}))\right) \ ,$$
$$\left(\mathsf{Rerand}_{\Pi'_1}(\mathsf{Mult}_{\Pi'_1}(s'_0, \mathsf{PK}_{1,0})), \mathsf{Enc}_{\Pi'_0}(0)\right) \ .$$

Now the distributions of s_1 alone (not concerning s_0) and s'_0 alone (not concerning s'_1) are uniform on $\mathcal{M}$, *which are independent of the choice of* (m_0, m_1). This implies that the distribution of $c^\dagger$ is independent of $b^\dagger$, therefore we have

$$\Pr[b = b^* \mid b^* = 0] = \Pr[b' \neq b^\dagger \mid b^* = 0] = 1/2 \ .$$

By the results above, the advantage $\mathrm{Adv}_{\mathcal{A}^\dagger}(k)$ of $\mathcal{A}^\dagger$ is equal to

$$\begin{aligned}\mathrm{Adv}_{\mathcal{A}^\dagger}(k) = \left|\Pr[b = b^*] - \frac{1}{2}\right| &= \left|\frac{1}{2}\left(\Pr[b = b^* \mid b^* = 0] + \Pr[b = b^* \mid b^* = 1]\right) - \frac{1}{2}\right| \\ &= \frac{1}{2}\left|\Pr[b = b^* \mid b^* = 1] - \frac{1}{2}\right| \ ,\end{aligned}$$

which has a negligible difference from $\mathrm{Adv}_{\mathcal{A}}(k)$. Therefore, $\mathrm{Adv}_{\mathcal{A}}(k)$ is negligible whenever $\mathrm{Adv}_{\mathcal{A}^\dagger}(k)$ is negligible. Hence we have $\mathrm{ZPA}(\Pi'_0) \to \mathrm{ZPA}(\Gamma(\Lambda_0, \Lambda_1))$, and the other claim $\mathrm{ZPA}(\Pi'_1) \to \mathrm{ZPA}(\Gamma(\Lambda_0, \Lambda_1))$ follows from the same argument. This completes the proof of Lemma 12. □

Now, starting from any RMHE scheme Π satisfying Assumptions 1 and 2 in Section 7 (e.g., the Paillier cryptosystem), we set

$$\Pi'_0 := \Pi \, , \, \Pi'_1 := \Gamma(T_3; \Gamma(T_2; \Pi)) \ ,$$

and we define the four RMHE schemes $\Pi_1, \dots, \Pi_4$ by

$$\Pi_1 = \Pi_2 := \Gamma(T_2; \Pi) \, , \, \Pi_3 := \Lambda_0 \, , \, \Pi_4 := \Lambda_1 \ .$$

Then we have

$$\mathrm{ZPA}(\Pi_3) \to \mathrm{ZPA}(\Pi) \to \mathrm{ZPA}(\Pi_1)$$

(where we used Lemma 11 with $i = 0$ and $j = 0$ for the first step, and Theorem 1 for the second step), while we have $\mathrm{ZPA}(\Pi_1) \not\to \mathrm{ZPA}(\Pi)$ by Theorem 4, therefore

$$\mathrm{ZPA}(\Pi_1) \not\to \mathrm{ZPA}(\Pi_3) \ .$$

Similarly, we have

$$\mathrm{ZPA}(\Pi_4) \to \mathrm{ZPA}(\Pi) \to \mathrm{ZPA}(\Pi_2)$$

(where we used Lemma 11 with $i = 1$ and $j = 0$ for the first step), while we have $\mathrm{ZPA}(\Pi_2) \not\to \mathrm{ZPA}(\Pi)$ as above, therefore

$$\mathrm{ZPA}(\Pi_2) \not\to \mathrm{ZPA}(\Pi_4) \enspace .$$

Hence, these schemes satisfy the first condition for our example. On the other hand, we have

$$\mathrm{ZPA}(\Gamma(\Pi_1, \Pi_2)) \to \mathrm{ZPA}(\Pi'_1) \to \mathrm{ZPA}(\Gamma(\Pi_3, \Pi_4))$$

(where we used Theorem 1 for the first step, and Lemma 12 for the second step), while we have $\mathrm{ZPA}(\Pi'_1) \not\to \mathrm{ZPA}(\Gamma(\Pi_1, \Pi_2))$ by Theorem 4, therefore

$$\mathrm{ZPA}(\Gamma(\Pi_1, \Pi_2)) \not\leftarrow \mathrm{ZPA}(\Gamma(\Pi_3, \Pi_4)) \enspace .$$

Hence, these schemes satisfy the second condition for our example. This gives an example for the non-monotonicity in generic constructions as mentioned above.

Moreover, when we set Π'_0 to be a nonsense RMHE scheme whose encryption algorithm outputs the plaintext itself as the (obviously insecure) ciphertext, we can construct another example by setting

$$\Pi'_1 := \Gamma(T_3; \Pi) \, , \, \Pi_1 = \Pi_2 := \Pi \, , \, \Pi_3 := \Lambda_0 \, , \, \Pi_4 := \Lambda_1 \enspace .$$

In this case, Π_3 and Π_4 are not ZPA secure by the definition directly (or by Lemma 11), therefore the first condition for our example is automatically satisfied. On the other hand, the relations $\mathrm{ZPA}(\Gamma(\Pi_1, \Pi_2)) \to \mathrm{ZPA}(\Pi'_1) \to \mathrm{ZPA}(\Gamma(\Pi_3, \Pi_4))$ and $\mathrm{ZPA}(\Gamma(\Pi_1, \Pi_2)) \not\leftarrow \mathrm{ZPA}(\Gamma(\Pi_3, \Pi_4))$ are derived by the same argument as above, therefore the second condition for our example is also satisfied. Hence, as mentioned above, we can also construct a desired example in such a way that Π_3 and Π_4 are never ZPA secure.

Acknowledgments. The authors thank the members of Shin-Akarui-Angou-Benkyo-Kai for a fruitful discussion on the work, especially Shota Yamada for his idea inspiring a part of our results, Jacob C. N. Schuldt for his many discussions on the work and valuable comments on this paper, and Nuttapong Attrapadung, Keita Emura and Takashi Yamakawa for their precious comments on this paper. The authors also thank the anonymous referees for previous submissions of this paper for their detailed comments.

References

[1] F. Armknecht, S. Katzenbeisser and A. Peter, Group homomorphic encryption: Characterizations, impossibility results, and applications, *Des. Codes Cryptography*, vol.67, no.2, 2013, pp.209–232.

[2] C. A. Asmuth and G. R. Blakley, An efficient algorithm for constructing a cryptosystem which is harder to break than two other cryptosystems, *Comput. Math. Appl.*, vol.7, no.6, 1981, pp.447–450.

[3] J. Benaloh, Dense probabilistic encryption, in: *Proceedings of SAC 1994*, 1994, pp.120–128.

[4] D. Boneh and X. Boyen, On the impossibility of efficiently combining collision resistant hash functions, in: *Proceedings of CRYPTO 2006*, LNCS 4117, 2006, pp.570–583.

[5] P. M. Cohn, *An Introduction to Ring Theory*, Springer-Verlag, London (2000).

[6] I. Damgård, Towards practical public key systems secure against chosen ciphertext attacks, in: *Proceedings of CRYPTO 1991*, LNCS 576, 1992, pp.445–456.

[7] I. Damgård and M. Jurik, A generalisation, a simplification and some applications of Paillier's probabilistic public-key system, in: *Proceedings of PKC 2001*, LNCS 1992, 2001, pp.119–136.

[8] I. Damgård, M. Jurik and J. B. Nielsen, A generalization of Paillier's public-key system with applications to electronic voting, *Int. J. Inform. Security*, vol.9, no.6, 2010, pp.371–385.

[9] Y. Dodis and J. Katz, Chosen-ciphertext security of multiple encryption, in: *Proceedings of TCC 2005*, LNCS 3378, 2005, pp.188–209.

[10] Y. Dodis, J. Katz, S. Xu and M. Yung, Key-insulated public key cryptosystems, in: *Proceedings of EUROCRYPT 2002*, LNCS 2332, 2002, pp.65–82.

[11] T. ElGamal, A public key cryptosystem and a signature scheme based on discrete logarithms, *IEEE Trans. Inform. Theory*, vol.31, no.4, 1985, pp.469–472.

[12] A. Escala, G. Herold, E. Kiltz, C. Ràfols and J. Villar, An algebraic framework for Diffie–Hellman assumptions, in: *Proceedings of CRYPTO 2013*

[13] L. Fousse, P. Lafourcade and M. Alnuaimi, Benaloh's dense probabilistic encryption revisited, in: *Proceedings of AFRICACRYPT 2011*, LNCS 6737, 2011, pp.348–362.

[14] O. Goldreich, *Computational Complexity: A Conceptual Perspective*, Cambridge University Press, New York (2008).

[15] O. Goldreich, N. Nisan and A. Wigderson, On Yao's XOR-lemma, Electronic Colloquium on Computational Complexity, TR95-050, 1995, `http://eccc.hpi-web.de/report/1995/050/`

[16] S. Goldwasser and S. Micali, Probabilistic encryption and how to play mental poker keeping secret all partial information, in: *Proceedings of STOC 1982*, 1982, pp.365–377.

[17] S. Goldwasser and S. Micali, Probabilistic encryption, *J. Comput. Syst. Sci.*, vol.28, no.2, 1984, pp.270–299.

[18] D. Harnik, J. Kilian, M. Naor, O. Reingold and A. Rosen, On robust combiners for oblivious transfer and other primitives, in: *Proceedings of EUROCRYPT 2005*, LNCS 3494, 2005, pp.96–113.

[19] A. Herzberg, On tolerant cryptographic constructions, in: *Proceedings of CT-RSA 2005*, LNCS 3376, 2005, pp.172–190.

[20] D. Hofheinz and E. Kiltz, Secure hybrid encryption from weakened key encapsulation, in: *Proceedings of CRYPTO 2007*, LNCS 4622, 2007, pp.553–571.

[21] R. Impagliazzo, Hard-core distributions for somewhat hard problems, in: *Proceedings of FOCS 1995*, 1995, pp.538–545.

[22] R. Impagliazzo, R. Jaiswal, V. Kabanets and A. Wigderson, Uniform direct product theorems: Simplified, optimized, and derandomized, *SIAM J. Comput.*, vol.39, no.4, 2010, pp.1637–1665.

[23] R. Impagliazzo and A. Wigderson, P = BPP if E requires exponential circuits: Derandomizing the XOR lemma, in: *Proceedings of STOC 1997*, 1997, pp.220–229.

[24] L. A. Levin, One-way functions and pseudorandom generators, *Combinatorica*, vol.7, no.4, 1987, pp.357–363.

[25] U. M. Maurer and Y. Yacobi, Non-interactive public-key cryptography, in: *Proceedings of EUROCRYPT 1991*, LNCS 547, 1991, pp.498–507.

[26] R. Meier and B. Przydatek, On robust combiners for private information retrieval and other primitives, in: *Proceedings of CRYPTO 2006*, LNCS 4117, 2006, pp.555–569.

[27] D. Naccache and J. Stern, A new public key cryptosystem based on higher residues, in: *Proceedings of ACM CCS 1998*, 1998, pp.59–66.

[28] T. Okamoto and S. Uchiyama, A new public-key cryptosystem as secure as factoring, in: *Proceedings of EUROCRYPT 1998*, LNCS 1403, 1998, pp.308–318.

[29] T. Okamoto and S. Uchiyama, Security of an identity-based cryptosystem and the related reductions, in: *Proceedings of EUROCRYPT 1998*, LNCS 1403, 1998, pp.546–560.

[30] P. Paillier, Public-key cryptosystems based on composite degree residuosity classes, in: *Proceedings of EUROCRYPT 1999*, LNCS 1592, 1999, pp.223–238.

[31] P. Paillier, Trapdooring discrete logarithms on elliptic curves over rings, in: *Proceedings of ASIACRYPT 2000*, LNCS 1976, 2000, pp.573–584.

[32] H. Shacham, A Cramer–Shoup encryption scheme from the linear assumption and from progressively weaker linear variants, IACR ePrint Archive 2007/074, `http://eprint.iacr.org/2007/074`

[33] V. Shoup, Lower bounds for discrete logarithms and related problems, in: *Proceedings of EUROCRYPT 1997*, LNCS 1233, 1997, pp.256–266.

Contents