Chosen Ciphertext Security via Point Obfuscation

Takahiro Matsuda and Goichiro Hanaoka

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

Abstract. In this paper, we show two new constructions of chosen ciphertext secure (CCA secure) public key encryption (PKE) from general assumptions. The key ingredient in our constructions is an obfuscator for point functions with multi-bit output (MBPF obfuscators, for short), that satisfies some (average-case) indistinguishability-based security, which we call AIND security, in the presence of hard-to-invert auxiliary input. Specifically, our first construction is based on a chosen plaintext secure PKE scheme and an MBPF obfuscator satisfying the AIND security in the presence of computationally hard-to-invert auxiliary input. Our second construction is based on a lossy encryption scheme and an MBPF obfuscator satisfying the AIND security in the presence of statistically hard-to-invert auxiliary input. To clarify the relative strength of AIND security, we show the relations among security notions for MBPF obfuscators, and show that AIND security with computationally (resp. statistically) hard-to-invert auxiliary input is implied by the average-case virtual black-box (resp. virtual grey-box) property with the same type of auxiliary input. Finally, we show that a lossy encryption scheme can be constructed from an obfuscator for point functions (point obfuscator) that satisfies re-randomizability and a weak form of composability in the worst-case virtual grey-box sense. This result, combined with our second generic construction and several previous results on point obfuscators and MBPF obfuscators, yields a CCA secure PKE scheme that is constructed *solely* from a re-randomizable and composable point obfuscator. We believe that our results make an interesting bridge that connects CCA secure PKE and program obfuscators, two seemingly isolated but important cryptographic primitives in the area of cryptography.

Keywords: public key encryption, lossy encryption, key encapsulation mechanism, chosen ciphertext security, point obfuscation.

1 Introduction

1.1 Background and Motivation

One of the fundamental research themes in cryptography is to clarify what the minimal assumptions to realize various kinds of cryptographic primitives are, and up to now, a number of relationships among primitives have been investigated and established. Clarifying these relationships gives us a lot of insights for how to construct and/or prove the security of cryptographic primitives, enables us to understand the considered primitives more deeply, and leads to systematizing the research area in cryptography.

In this paper, we focus on the constructions of public key encryption (PKE) schemes secure against chosen ciphertext attacks (CCA) [54, 29] from general cryptographic assumptions. CCA secure PKE is one of the most important cryptographic primitives that has been intensively studied, due to its resilience against practical attacks such as [10], and its implication to many useful security notions, such as non-malleability [29] and universal composability [18].

The first successful result regarding this line of research is the construction by Dolev, Dwork, and Naor [29] that uses a chosen plaintext secure (CPA secure) PKE scheme and a non-interactive zero-knowledge proof. Since these two primitives can be constructed from (an enhanced variant of) trapdoor permutations (TDP) [35], CCA secure PKE can be constructed solely from TDPs. Canetti, Halevi, and Katz [20] showed that CCA secure PKE can be constructed from an identity-based encryption (IBE). It was later shown that in fact, a weaker primitive called tag-based encryption suffices [45]. Peikert and Waters [53] showed that CCA secure PKE can be constructed from any lossy trapdoor function (TDF), and subsequent works showed that injective TDFs with weaker properties suffice: injective TDFs secure for correlated inputs [55], slightly lossy TDFs [49], adaptive one-way TDFs [46], and adaptive one-way relations [59]. (CPA secure) PKE schemes with additional security/functional properties have also turned out to be useful for constructing CCA secure PKE: Hemenway and Ostrovsky [40] showed that we can construct CCA secure PKE in several ways from homomorphic encryption with appropriate properties. The same authors [41] also showed that CCA secure PKE can be constructed from a lossy encryption scheme [6] if the plaintext space is larger than the randomness space (the results of [40, 41] achieve CCA secure PKE via lossy TDFs [53]). Hohenberger, Lewko, and Waters [42] showed that if one has a PKE scheme which satisfies the notion called detectable CCA security, which is somewhere between CCA1 and CCA2 security, then using it one can construct a CCA secure PKE scheme. Myers and Shelat [50] showed how to construct a CCA secure PKE scheme that can encrypt plaintexts with arbitrary length from a CCA secure one with 1-bit plaintext space. Lin and Tessaro [47] showed how to amplify weak CCA security into ordinary one. Very recently, Dachman-Soled [25] constructs CCA secure PKE from PKE satisfying (standard model) plaintext-awareness together with some additional property.

The main purpose of this work is to show that a different kind of cryptographic primitives is also useful for achieving CCA secure PKE. Specifically, we add new recipes for the construction of CCA secure PKE, based on the techniques and results from program obfuscation [3] for the very simple classes of functions, point functions and point functions with multi-bit output. Despite the tremendous efforts, it is not known whether it is possible to construct CCA secure PKE only from CPA secure one (in fact, a partial negative result is known [33]). Clarifying new classes of primitives that serve as building blocks is important for tackling this problem. In particular, it has been shown that there is no black-box construction of IBE and a TDF from (CCA secure) PKE [11, 34] and thus to tackle the CPA-to-CCA problem, the attempts to construct IBE or the above TDF-related primitives from a CPA secure PKE scheme seem hopeless (though there is a possibility that some non-black-box construction exists). Our new constructions based on (multi-bit) point obfuscators do not seem to be covered by these negative results, and thus potentially it could serve as a new target for building CCA secure PKE.

1.2 Our Contribution

In this paper, we show two new constructions of CCA secure PKE schemes from general cryptographic assumptions, using the techniques and results from program obfuscation [3]. We actually construct CCA secure key encapsulation mechanisms (KEMs) [24], where a KEM is a "PKE"-part of hybrid encryption that encrypts a random "session-key" for symmetric key encryption (SKE). By combining a CCA secure KEM with a CCA secure SKE scheme, one obtains a full-fledged CCA secure PKE scheme [24]. The key ingredient in our constructions is an obfuscator for point functions with multi-bit output (MBPF obfuscators) [48, 19, 27, 37, 21, 7], that satisfies a kind of average-case indistinguishability-based security in the presence of "hard-to-invert" auxiliary inputs. The formal definition of this security notion is given in Section 3. For brevity, we call it AIND security.

Our first construction in Section 4.1 is based on a CPA secure PKE scheme and an MBPF obfuscator satisfying the above mentioned AIND security in the presence of computationally hard-to-invert auxiliary input. Our second construction in Section 4.2 is based on a lossy encryption scheme [6] and an MBPO satisfying the above mentioned AIND security in the presence of statistically hard-to-invert auxiliary input. Interestingly, the first and the second constructions are in fact exactly the same, and we show two different security analyses from different assumptions on building blocks. These two constructions add new recipes into the current picture of the constructions of CCA secure PKE schemes/KEMs from general cryptographic assumptions.

In order to clarify where these AIND security definitions for MBPF obfuscators are placed, in Section 5 we show that AIND security with computationally (resp. statistically) hard-to-invert auxiliary inputs is implied by the (average-case) virtual black-box property [3] (resp. virtual grey-box property [7]) in the presence of the same auxiliary inputs. Besides these, we show the relations among several related worst-/average-case virtual black-/grey-box properties under several types of auxiliary inputs, and summarize them in Fig. 2, which we believe is useful for further research on this topic and might be of independent interest.

Finally, in Section 6, we show that a lossy encryption scheme can be constructed from an obfuscator for point functions (point obfuscator) that satisfies re-randomizability [7] and a weak form of composability [48, 19, 7] in the worst-case virtual grey-box sense. This result, combined with our second generic construction and the results on composable point obfuscators with the virtual grey-box property in [7], shows that a CCA secure PKE scheme can be constructed *solely* from a point obfuscator which is re-randomizable and composable.

We believe that our results make an interesting bridge that connects CCA secure PKE and program obfuscators,[1] two seemingly isolated but important primitives in the area of cryptography, and hope that our results motivate further studies on them.

[1] Recently, Sahai and Waters [57] (among others) showed how to construct CCA secure PKE using *indistinguishability obfuscation*. We explain the difference with our results in Section 1.4.

1.3 Overview of Techniques

Our proposed constructions of KEMs are based on the "witness-recovering" technique [53, 55, 50, 42] in which a part of randomness used to generate a ciphertext is somehow embedded into the ciphertext itself, and is later recovered in the decryption process for checking the validity of the ciphertext by re-encryption. What we believe is novel in our constructions is how to implement this mechanism of witness-recovering by using an MBPF obfuscator with an appropriate security property.

Let $\mathcal{I}_{\alpha\to\beta}$ denote an MBPF such that $\mathcal{I}_{\alpha\to\beta}(x) = \beta$ if $x = \alpha$ and $\perp$ otherwise, and let MBPO denotes an MBPF obfuscator which takes an MBPF $\mathcal{I}_{\alpha\to\beta}$ as input, and outputs an obfuscated circuit $\mathtt{DL}$ for $\mathcal{I}_{\alpha\to\beta}$. ("$\mathtt{DL}$" stands for "digital locker," the name due to [19].) Let $\Pi = (\mathsf{PKG}, \mathsf{Enc}, \mathsf{Dec})$ be a PKE scheme, where PKG, Enc, and Dec are the key generation, the encryption, and the decryption algorithms of Π, respectively.

Below we give a high level idea behind our main proposed constructions in Section 4 by explaining how the "toy" version of our constructions $\Pi' = (\mathsf{PKG}', \mathsf{Enc}', \mathsf{Dec}')$, constructed using Π and MBPO, is proved $\mathtt{CCA1}$ secure based on the assumptions that Π is $\mathtt{CPA}$ secure and that MBPO satisfies the virtual black-box property with respect to dependent auxiliary input [36]. (As mentioned earlier, in this paper we actually construct KEMs rather than PKE schemes, but the intuition for our results are captured by the explanation here.) A public/secret key pair (PK, SK) of Π' is of the form $PK = (pk_1, pk_2)$, $SK = (sk_1, sk_2)$, where each (pk_i, sk_i) is an independently generated key pair by running PKG. To encrypt a plaintext m under PK, Enc' first picks a random string $\alpha \in \{0,1\}^k$ (where k is the security parameter) and two randomness r_1 and r_2 for Enc, and computes a ciphertext C in the following way:

$$C = (c_1, c_2, \mathtt{DL}) = \Big(\mathsf{Enc}(pk_1, (m\|\alpha); r_1), \mathsf{Enc}(pk_2, (m\|\alpha); r_2), \mathsf{MBPO}(\mathcal{I}_{\alpha\to(r_1\|r_2)})\Big)$$

where "$\|$" denotes the concatenation of strings, and "$\mathsf{Enc}(pk, m; r)$" means to encrypt the plaintext m under the public key pk using the randomness r. To decrypt C, we first decrypt c_1 by using sk_1 to obtain $(m\|\alpha)$, then run $\mathtt{DL}(\alpha)$ to recover $(r_1\|r_2)$. Finally, m is returned if $c_i = \mathsf{Enc}(pk_i, (m\|\alpha); r_i)$ holds for both $i = 1, 2$, and otherwise we reject C. Here, it should be noted that due to the symmetric roles of pk_1 and pk_2 and the validity check by re-encryption performed in Dec', we can also decrypt C using sk_2, so that the decryption result of C using sk_1 and that using sk_2 always agree.

Now, recall the interface of a $\mathtt{CCA1}$ adversary $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$, where $\mathcal{A}_1$ and $\mathcal{A}_2$ represent an adversary's algorithm before and after the challenge, respectively. $\mathcal{A}_1$ is firstly given a public key PK, and can start using the decryption oracle $\mathsf{Dec}'(SK, \cdot)$. After that, $\mathcal{A}_1$ terminates with output two plaintexts (m_0, m_1) and some state information st that is passed to $\mathcal{A}_2$. $\mathcal{A}_2$ is given st and the challenge ciphertext $C^* = (c_1^*, c_2^*, \mathtt{DL}^*)$ which is an encryption of m_b (where b is the challenge bit), and outputs a bit as its guess for b.

The key observation is that $\mathcal{A}_2$ can be seen as an adversary for the MBPF obfuscator MBPO, by regarding $(\mathsf{st}, c_1^*, c_2^*)$ as an auxiliary input z about the obfuscated circuit $\mathtt{DL}^*$ of the MBPF $\mathcal{I}_{\alpha^*\to(r_1^*\|r_2^*)}$. Then, if MBPO satisfies the virtual black-box property with respect to dependent auxiliary input [36], there exists a simulator $\mathcal{S}$ that takes only $z = (\mathsf{st}, c_1^*, c_2^*)$ as input, has oracle access to $\mathcal{I}_{\alpha^*\to(r_1^*\|r_2^*)}$, and has the property that

$\mathcal{A}$'s success probability (in guessing b) is negligibly close to the probability that $\mathcal{S}$ succeeds in guessing b. (For convenience, let us call the latter probability "$\mathcal{S}$'s success probability," although $\mathcal{S}$ is not a CCA1 adversary and thus its task is not to guess a challenge bit.) This means that if $\mathcal{S}$'s success probability is close to $1/2$, then so is $\mathcal{A}$'s success probability, which will prove the CCA1 security of Π'.

To show that $\mathcal{S}$'s success probability is close to $1/2$, we consider the hypothetical experiment for $\mathcal{S}$ in which the auxiliary input z is generated so that decryption queries from $\mathcal{A}_1$ are answered using sk_2, and both c_1^* and c_2^* are an encryption of a fixed value (say, $0^{|m_0|+k}$). Since z contains no information on b and α^*, in this hypothetical experiment $\mathcal{S}$'s success probability is exactly $1/2$ and the probability that $\mathcal{S}$ makes the query α^* (which is chosen randomly) is negligible. Next, we make the experiment closer to the actual $\mathcal{S}$'s experiment, by changing c_1^* into an encryption of $(m_b\|\alpha^*)$. By the CPA security regarding pk_1, $\mathcal{S}$'s success probability as well as the probability of $\mathcal{S}$ making the query α^* is negligibly close to those in the hypothetical experiment. Then, we further modify the previous experiment by changing c_2^* into an encryption of $(m_b\|\alpha^*)$, but this time we use sk_1 for answering $\mathcal{A}_1$'s queries. Notice that this is exactly the actual experiment for $\mathcal{S}$. As mentioned above, switching sk_2 to/from sk_1 for answering $\mathcal{A}_1$'s queries does not affect $\mathcal{A}_1$'s behavior, and thus again by the CPA security regarding pk_2, $\mathcal{S}$'s success probability is negligibly close to $1/2$ and the probability that $\mathcal{S}$ makes the query α^* is negligible. Then, by the virtual black-box property of MBPO with auxiliary input, $\mathcal{A}$'s original success probability is negligibly close to $1/2$, meaning that $\mathcal{A}$ has negligible advantage in breaking the CCA1 security of the scheme Π'.

The above completes a proof sketch of how Π' is proved CCA1 secure. By encrypting a random K, Π' can be used as a CCA1 secure KEM. Our proposed CCA2 secure KEMs are obtained by applying several optimizations and enhancement to this KEM:

- Firstly, we do not need the full virtual black-box property with auxiliary input of [36]. As mentioned earlier, an indistinguishability-based definition in the presence of only "hard-to-invert" auxiliary input is sufficient for a similar argument to work.
- Secondly, we need not include a plaintext into each of c_i. Instead, we pick a randomness $K \in \{0,1\}^k$ used as a plaintext of a KEM, and include this K into the output of the MBPF, i.e now we obfuscate the MBPF $\mathcal{I}_{\alpha\to(r_1\|r_2\|K)}$. (This is the actual our basic construction whose formal description and security proof are given in the full version.)
- Lastly, note that the above construction cannot be proved to be CCA2 secure as it is. In particular, the obfuscated circuit DL could be malleable. To deal with this issue, instead of the Naor-Yung-style double encryption [52], we employ the Dolev-Dwork-Naor-style multiple encryption [29] together with the technique of the "unduplicatable set selection" [56]. Unlike the classical method of using a one-time signature scheme, we implement the technique using a universal one-way hash function (UOWHF) [51], where a hash value of the obfuscated circuit DL is used as a "selector" of the public key components. Another issue is that the second stage adversary $\mathcal{A}_2$ in the CCA2 experiment can also make decryption queries, and thus the above explained idea of replacing $\mathcal{A}_2$ with a simulator $\mathcal{S}$ does not work. However, our indistinguishability-based security definition for MBPF obfuscators enables us to

directly work with an original CCA2 adversary, and we can avoid considering how a simulator deal with the queries from $\mathcal{A}_2$. For more details, see Section 4.

1.4 Related Work: Program Obfuscation

Roughly speaking, an obfuscator is an algorithm that takes a program (e.g. Turing machine or circuit) as input, and outputs another program with the same functionality, but otherwise "unintelligible."

After the impossibility of general-purpose program obfuscation satisfying the nowadays standard security notion called *virtual black-box* property shown in the seminal work by Barak et. al. [3], several subsequent works extended the impossibility in various other settings [36, 58, 38, 7]. The other line of research pursues possibilities of obfuscating a specific class of functions. Before 2013, most known positive results were about obfuscation for point functions and their variants, e.g. [48, 58, 19, 22, 7]. Relaxing the security requirements to "average-case" in which a program is sampled according to some distribution, several more complex tasks have been shown to be obfuscatable, such as proximity testing [28] and cryptographic tasks such as re-encryption [43].

Since the first candidates of a cryptographic multilinear map have been proposed in 2013 [30, 23], the research field of (cryptographic) obfuscation has drastically changed and accelerated. Brakerski and Rothblum [14] showed how to construct an obfuscator for conjunctions from graded encoding schemes [30, 23], and the same authors showed a further extension [13]. Most recently, they showed a general-purpose obfuscator satisfying a virtual black-box property in an idealized model called the generic graded encoded scheme model [15]. Barak et al. [2] studied obfuscation for a class of functions called *evasive functions* which in particular includes point functions (with multi-bit output). A series of works [32, 57, 44, 31] (and many other recent works) have shown that a general-purpose obfuscator satisfying a security notion weaker than the virtual black-box property, called *indistinguishability obfuscator*, which seems to be too weak to be useful, is in fact surprisingly powerful and can be used as a building block for constructing a various kinds of cryptographic primitives. Garg et al. [32] constructed the first candidate of general-purpose indistinguishability obfuscation. A security notion stronger than indistinguishability obfuscation, called *differing-inputs obfuscation* (a.k.a. *extractability obfuscation* [12]), has also been shown to be quite powerful and useful [1, 12].

Among a number of recent fascinating results, especially relevant to our work is the work by Sahai and Waters [57] who showed (among several other primitives) how to construct CCA secure PKE from an indistinguishability obfuscator (and a one-way function). Although our work and [57] have the common property that both works build CCA secure PKE using techniques and results from obfuscation, our use of obfuscators and that of [57] are quite different: We use an obfuscator for a specific class of functions, point functions and MBPFs, while [57] uses an obfuscator for all polynomial-sized circuits. Furthermore, the indistinguishability-based security notion for MBPF obfuscators used in our main result is about randomly chosen MBPFs, while that used in [57] is for the worst-case choice of circuits (that compute the same functions). We would also like to stress that our work and [57] were done concurrently and independently.

1.5 Paper Organization

The rest of the paper is organized as follows: In Section 2 (and Appendix A) we review the basic notations and definitions of primitives. In Section 3, we introduce the formal definitions of our new indistinguishability-based security notions for MBPF obfuscators. In Section 4, we show our main results: two CCA secure KEMs using a MBPF obfuscator. In Section 5, we investigate relations between our new security notions and other notions for MBPF obfuscators. In Section 6, we show how to construct a lossy encryption scheme from a point obfuscator with re-randomizability and composability. In Section 7, we discuss some issues on the MBPF obfuscators that we use.

2 Preliminaries

Here, we review the basic notation and the definitions for lossy encryption [6] and (cryptographic) obfuscation. The definitions for standard cryptographic primitives that are not given here are given in Appendix A, which include PKE, KEMs, and UOWHFs.

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

2.1 Lossy Encryption

Definition 1. *A tuple of PPTAs $\Pi = (\mathsf{PKG}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{LKG})$ is said to be an* ϵ-lossy encryption *scheme*[2] *if the following properties are satisfied:*

- ***(Syntax)*** *$(\mathsf{PKG}, \mathsf{Enc}, \mathsf{Dec})$ constitutes a PKE scheme. The algorithm LKG is called a lossy key generation algorithm, which takes 1^k as input, and outputs a "lossy" public key pk.*
- ***(Indistinguishability of ordinary/lossy keys)*** *For all PPTAs $\mathcal{A}$, $\mathsf{Adv}^{\mathtt{KEY}}_{\Pi,\mathcal{A}}(k) := 2 \cdot |\Pr[\mathsf{Expt}^{\mathtt{KEY}}_{\Pi,\mathcal{A}}(k) = 1] - 1/2|$ is negligible, where the experiment $\mathsf{Expt}^{\mathtt{KEY}}_{\Pi,\mathcal{A}}(k)$ is defined as follows:*

$$[\ (pk_0, sk) \leftarrow \mathsf{PKG}(1^k);\ pk_1 \leftarrow \mathsf{LKG}(1^k);\ b \leftarrow \{0,1\};\ b' \leftarrow \mathcal{A}(pk_b);$$
$$\mathrm{Return}\ (b' \stackrel{?}{=} b)\ ].$$

[2] In this paper, we consider the "exact security"-style definition for lossy encryption and CPA secure PKE. This is to quantify the "hardness" of inverting an auxiliary input functions used in the security definitions of MBPF obfuscators. For details, see Section 3.

– ***(Statistical lossiness)*** *For all computationally unbounded algorithms $\mathcal{A}$ and for all sufficiently large $k \in \mathbb{N}$ it holds that $\mathsf{Adv}^{\mathtt{LOS\text{-}CPA}}_{\Pi,\mathcal{A}}(k) := 2 \cdot |\Pr[\mathsf{Expt}^{\mathtt{LOS\text{-}CPA}}_{\Pi,\mathcal{A}}(k) = 1] - 1/2| \leq \epsilon(k)$, where the experiment $\mathsf{Expt}^{\mathtt{LOS\text{-}CPA}}_{\Pi,\mathcal{A}}(k)$ is defined in the same way as the ordinary $\mathtt{CPA}$ experiment $\mathsf{Expt}^{\mathtt{CPA}}_{\Pi,\mathcal{A}}(k)$ except that the public key pk is generated as $pk \leftarrow \mathsf{LKG}(1^k)$. We call ϵ* lossiness.

2.2 Obfuscation for Circuits and Worst-Case Security Definitions

Here, we recall the definition of circuit obfuscations, following the definitions given in [3, 48, 36, 8]. In the following, by $\mathcal{C}$ we denote an ensemble $\{\mathcal{C}_k\}_{k\in\mathbb{N}}$, where $\mathcal{C}_k$ is a collection of circuits whose input length is k and whose size is bounded by some polynomial of k.

Definition 2. *We say that a PPTA* Obf *is an obfuscator for $\mathcal{C}$ if it satisfies the following:*

– ***(Functionality)*** *For every $k \in \mathbb{N}$ and every $C \in \mathcal{C}_k$, a circuit output from $\mathsf{Obf}(C)$ computes the same function as C.*
– ***(Polynomial blowup)*** *There exists a polynomial $p = p(k) > 0$ such that for every $k \in \mathbb{N}$ and every $C \in \mathcal{C}_k$, the size of a circuit output from $\mathsf{Obf}(C)$ is bounded by p.*

Note that Definition 2 is only about the functionality requirements of obfuscators.

Next, we recall the security definitions for "worst-case" choice of circuits.: The *virtual black-box property* is due to Barak et al. [3], the *virtual black-box property with (dependent) auxiliary input* is due to Goldwasser and Kalai [36], and *virtual "grey"-box (with (dependent) auxiliary input)* is due to Bitansky and Canetti [7].

Definition 3. *We say that an obfuscator* Obf *for $\mathcal{C}$ satisfies:*

– *the* worst-case virtual black-box property *(*$\mathtt{WVB}$ *security, for short), if for every PPTA $\mathcal{A}$ (adversary) and every positive polynomial $q = q(k)$, there exists a PPTA $\mathcal{S}$ (simulator) such that for all sufficiently large $k \in \mathbb{N}$ and all circuits $C \in \mathcal{C}_k$, it holds that*

$$|\Pr[\mathcal{A}(1^k, \mathsf{Obf}(C)) = 1] - \Pr[\mathcal{S}^C(1^k) = 1]| \leq 1/q,$$

– *the* worst-case virtual black-box property w.r.t. auxiliary input *(*$\mathtt{WVB\text{-}AI}$ *security, for short), if for every PPTA $\mathcal{A}$ and every positive polynomials $q = q(k)$ and $\ell = \ell(k)$, there exists a PPTA $\mathcal{S}$ such that all sufficiently large $k \in \mathbb{N}$, all circuits $C \in \mathcal{C}_k$, and all strings $z \in \{0,1\}^\ell$, it holds that*

$$|\Pr[\mathcal{A}(1^k, z, \mathsf{Obf}(C)) = 1] - \Pr[\mathcal{S}^C(1^k, z) = 1]| \leq 1/q,$$

where the probabilities are over the randomness consumed by Obf*, $\mathcal{A}$, and $\mathcal{S}$.*

Furthermore, we define the worst-case virtual grey-box property *(*$\mathtt{WVG}$ *security), and the* worst-case virtual grey-box property w.r.t. auxiliary input *(*$\mathtt{WVG\text{-}AI}$ *security) of* Obf*, in the same way as the definitions for the corresponding virtual black-box properties, except that we replace "a PPTA $\mathcal{S}$" in each definition with "a computationally unbounded algorithm $\mathcal{S}$ that makes only polynomially many queries."*

Note that in the above definitions, the simulator $\mathcal{S}$ can depend on the polynomial q which represents the "quality of simulation." Wee [58] refers to the simulators of this type as a "weak simulator."

We also define (t-)composability of obfuscations [48, 19, 7, 21]. Following [8], we only define the composability in the grey-box (WVG) notion, using a computationally unbounded simulator, which is sufficient for our purpose in this paper.

Definition 4. *([7]) Let $t = t(k) > 0$ be a polynomial. We say that an obfuscator* Obf *for $\mathcal{C}$ satisfies* t-composability*, if for every PPTA $\mathcal{A}$ and a positive polynomial $q = q(k)$, there exists a computationally unbounded algorithm $\mathcal{S}$ that makes only polynomially many queries, such that for all sufficiently large $k \in \mathbb{N}$ and for all circuits $C_1, \ldots, C_t \in \mathcal{C}_k$, it holds that:*

$$|\Pr[\mathcal{A}(1^k, \mathsf{Obf}(C_1), \ldots, \mathsf{Obf}(C_t)) = 1] - \Pr[\mathcal{S}^{C_1,\ldots,C_t}(1^k) = 1]| \leq 1/q,$$

where the probabilities are over the randomness consumed by Obf*, $\mathcal{A}$, and $\mathcal{S}$.*

Notations for Point Obfuscators and MBPF Obfuscators. Let $\mathcal{X}$ be a finite set, $t \in \mathbb{N}$, $\alpha \in \mathcal{X}$, and $\beta \in \{0,1\}^t$. A *point function* $\mathcal{I}_\alpha$ and a *multi-bit point function* (MBPF) $\mathcal{I}_{\alpha\to\beta}$ are functions defined as follows:

$$\mathcal{I}_\alpha(x) = \begin{cases} \top & \text{if } x = \alpha \\ \bot & \text{otherwise} \end{cases} \quad \text{and} \quad \mathcal{I}_{\alpha\to\beta}(x) = \begin{cases} \beta & \text{if } x = \alpha \\ \bot & \text{otherwise} \end{cases}$$

We refer to α and β as the *point address* and the *point value*, respectively.

In this paper, we will only consider circuits for computing point functions/MBPFs with the properties that (1) the description is given in some canonical form and thus there is a one-to-one correspondence between a point address/value and the circuit for computing the point function/MBPF, and (2) the description of the circuits reveals the point address/value in the clear. Hereafter, we will identify a point function and an MBPF with circuits that compute them (with the above mentioned properties).

For an ensemble $\mathcal{X} = \{\mathcal{X}_k\}_{k\in\mathbb{N}}$, where each $\mathcal{X}_k$ is a set, we denote by $\mathsf{PF}(\mathcal{X})$ the ensemble of point functions $\{\mathcal{I}_\alpha\}_{\alpha\in\mathcal{X}_k}$. Similarly, for $\mathcal{X}$ and a polynomial t, we denote by $\mathsf{MBPF}(\mathcal{X}, t)$ the ensemble MBPFs $\{\mathcal{I}_{\alpha\to\beta}\}_{\alpha\in\mathcal{X}_k,\beta\in\{0,1\}^t}$.

Hereafter, we refer to an obfuscator for point functions as a *point obfuscator* and will denote it by PO. Furthermore, we refer to an obfuscator for MBPFs as an *MBPF obfuscator* and will denote it by MBPO. Moreover, we call an ensemble $\mathcal{X} = \{\mathcal{X}_k\}_{k\in\mathbb{N}}$ a *"domain ensemble" (for point functions and MBPFs)* if (1) for all $k \in \mathbb{N}$, each element of $\mathcal{X}_k$ is k-bit, (2) $|\mathcal{X}_k|$ is superpolynomially large in k (and thus $1/|\mathcal{X}_k|$ is negligible), and (3) we can efficiently sample an element from $\mathcal{X}_k$ uniformly at random.

Concrete Instantiations of a Composable Point Obfuscator and an MBPF Obfuscator. In Appendix B, we recall the concrete construction of a point obfuscator due to the results [17, 7], which is originally proposed by Canetti [17] as a perfectly one-way function and is later shown to be t-composable under the t-strong vector decision Diffie-Hellman (t-SVDDH) assumption [7], which is a stronger variant of the decisional Diffie-Hellman (DDH) assumption. There, we also recall the construction of an MBPF obfuscator based on a composable point obfuscator [19, 7].

3 New Security Definitions for MBPF Obfuscators

In this section, we introduce and formalize the new security notions for MBPF obfuscators that we call *average-case indistinguishability w.r.t. (computationally/statistically) partially uninvertible auxiliary input*, which will play a central role in our proposed KEMs given in Section 4. This security definition requires that obfuscated circuits of MBPFs hide the point values on average, even in the presence of "dependent" auxiliary inputs [36, 27], as long as the auxiliary input has some "hard-to-invert" property.

In the following, we formally define what we mean by "hard-to-invert" auxiliary input in Section 3.1. Then, in Section 3.2, we define the new indistinguishability-based notions.

For notational convenience, in this section, $\mathcal{X}$ will always denote a domain ensemble $\{\mathcal{X}_k\}_{k\in\mathbb{N}}$, and $t = t(k) > 0$ be a polynomial that will be used for MBPF obfuscators for $\mathsf{MBPF}(\mathcal{X}, t)$, and do not introduce them in each definition.

3.1 Auxiliary Input Functions and Partial Uninvertibility

For MBPF obfuscators, we will consider the average-case security in the presence of "dependent" auxiliary input [36] that depends on the description of an MBPF $\mathcal{I}_{\alpha\to\beta}$ being obfuscated. We will capture this by a probabilistic function ai that takes as input the point address/value pair $(\alpha, \beta) \in \mathcal{X}_k \times \{0,1\}^t$. Furthermore, we consider the (average-case) "partial uninvertibility" of the function ai. That is, given z output by $\mathsf{ai}(\alpha, \beta)$ for a randomly chosen (α, β), it is hard to find α. We consider computational and statistical partial uninvertibility.

Definition 5. *Let $\delta : \mathbb{N} \to [0,1]$, and let $\mathsf{ai} : \mathcal{X}_k \times \{0,1\}^t \to \{0,1\}^*$ be a (possibly probabilistic) two-input function. We say that ai is a* δ-computationally (resp. δ-statistically) partially uninvertible auxiliary input function (δ-`cPUAI` (resp. δ-`sPUAI`) function, for short) *if (1) it is efficiently computable, and (2) for all PPTAs (resp. computationally unbounded algorithms) $\mathcal{F}$ and for all sufficiently large $k \in \mathbb{N}$, it holds that $\mathsf{Adv}^{\texttt{P-Inv}}_{\mathsf{ai},\mathcal{F}}(k) := \Pr[\mathsf{Expt}^{\texttt{P-Inv}}_{\mathsf{ai},\mathcal{F}}(k) = 1] - 1/|\mathcal{X}_k| \le \delta(k)$,[3] where the experiment $\mathsf{Expt}^{\texttt{P-Inv}}_{\mathsf{ai},\mathcal{F}}(k)$ is defined as follows:*

$$[\ \alpha \leftarrow \mathcal{X}_k;\ \beta \leftarrow \{0,1\}^t;\ z \leftarrow \mathsf{ai}(\alpha,\beta);\ \alpha' \leftarrow \mathcal{F}(1^k, z);\ \mathrm{Return}\ (\alpha' \stackrel{?}{=} \alpha)\ ].$$

Furthermore, we say that ai is ℓ-bounded if the output length of ai is bounded by $\ell = \ell(k)$.

3.2 Average-Case Indistinguishability of Point Values with Auxiliary Input

In our proposed KEM constructions, what we need for an MBPF obfuscator is that it hides the point value "on average," in the presence of auxiliary input that is *simltaneously* dependent on the point address and the point value. This indistinguishability-based definition, formalized below, enables us to avoid using simulator-based security notions, and helps to make the security analyses of our proposed constructions simpler.

[3] Here, the subtraction of $1/|\mathcal{X}_k|$ is to offset the trivial success probability by a random guess.

Definition 6. *Let $\delta : \mathbb{N} \to [0,1]$. We say that an MBPF obfuscator* MBPO *satisfies* average-case indistinguishability w.r.t. δ-computationally (resp. δ-statistically) partially uninvertible auxiliary input *(* $\texttt{AIND}$-δ-$\texttt{cPUAI}$ *(resp.* $\texttt{AIND}$-δ-$\texttt{sPUAI}$*) secure, for short), if for all PPTAs $\mathcal{A}$ and all δ-$\texttt{cPUAI}$ (resp. δ-$\texttt{sPUAI}$) functions* ai, $\mathsf{Adv}^{\texttt{AIND-AI}}_{\mathsf{MBPO},\mathsf{ai},\mathcal{A}}(k) := 2 \cdot |\Pr[\mathsf{Expt}^{\texttt{AIND-AI}}_{\mathsf{MBPO},\mathsf{ai},\mathcal{A}}(k) = 1] - 1/2|$ *is negligible, where the experiment* $\mathsf{Expt}^{\texttt{AIND-AI}}_{\mathsf{MBPO},\mathsf{ai},\mathcal{A}}(k)$ *is defined as follows:*

$$[\, \alpha \leftarrow \mathcal{X}_k;\ \beta_0, \beta_1 \leftarrow \{0,1\}^t;\ z \leftarrow \mathsf{ai}(\alpha, \beta_0);\ b \leftarrow \{0,1\};$$
$$\texttt{DL} \leftarrow \mathsf{MBPO}(\mathcal{I}_{\alpha \to \beta_b});\ b' \leftarrow \mathcal{A}(1^k, z, \texttt{DL});\ \mathrm{Return}\ (b' \stackrel{?}{=} b)\,].$$

In the experiment, $\texttt{DL}$ stands for a "digital locker" (the name is due to [19]).

The following is a simple fact that in order for the new definitions to be meaningful, δ has to be a negligible function. (The proof is given in the full version.)

Lemma 1. *Let $\delta : \mathbb{N} \to [0,1]$. If δ is non-negligible, then an MBPF obfuscator cannot be* $\texttt{AIND}$-δ-$\texttt{sPUAI}$ *secure (and hence it cannot be* $\texttt{AIND}$-δ-$\texttt{cPUAI}$ *secure, either).*

4 Chosen Ciphertext Security via MBPF Obfuscation

In this section, we show our main results: two constructions of $\texttt{CCA2}$ secure KEMs. The first and second constructions are given in Sections 4.1 and 4.2, respectively. We also explain several extensions applicable to our proposed constructions in Section 4.3.

4.1 First Construction

Let $\Pi = (\mathsf{PKG}, \mathsf{Enc}, \mathsf{Dec})$ be a PKE scheme with the plaintext space $\{0,1\}^k$, the public key length $\ell_{\texttt{PK}}(k)$, the randomness length $\ell_{\texttt{R}}(k)$, and the ciphertext length $\ell_{\texttt{C}}(k)$ (where the definitions of these are given in Appendix A). We define $t(k) = k \cdot \ell_{\texttt{R}}(k) + k$ and $t'(k) = k \cdot \ell_{\texttt{PK}}(k) + k \cdot \ell_{\texttt{C}}(k) + k$. Let $\mathcal{X} = \{\mathcal{X}_k\}_{k \in \mathbb{N}}$ be a domain ensemble such that each element in $\mathcal{X}_k$ is of length k, and let MBPO be an MBPF obfuscator for $\mathsf{MBPF}(\mathcal{X}, t)$. Furthermore, let $\mathcal{H} = (\mathsf{HKG}, \mathsf{H})$ be a UOWHF. Then we construct the proposed KEM $\Gamma = (\mathsf{KKG}, \mathsf{Encap}, \mathsf{Decap})$ as in Fig. 1.

Useful Properties of Γ. To show the $\texttt{CCA2}$ security of the proposed KEM Γ, it is useful to note the following two simple properties, which are both due to the validity check performed in the last step of Decap (and the correctness of the underlying PKE scheme Π). The first property states that in order to generate a valid ciphertext, an obfuscated circuit $\texttt{DL}$ cannot be copied from other valid ciphertexts.

Lemma 2. *Let (PK, SK) be a key pair output by $\mathsf{KKG}(1^k)$, and $C = (c_1, \ldots, c_k, \texttt{DL})$ be a ciphertext output by $\mathsf{Encap}(PK)$. Then, for any ciphertext $C' = (c'_1, \ldots, c'_k, \texttt{DL}')$ satisfying $\texttt{DL}' = \texttt{DL}$ and $(c'_1, \ldots, c'_k) \neq (c_1, \ldots, c_k)$, it holds that $\mathsf{Decap}(SK, C') = \perp$.*

$\mathsf{KKG}(1^k)$:	$\mathsf{Decap}(SK, C)$:
$\kappa \leftarrow \mathsf{HKG}(1^k)$	Parse SK as $(\{sk_i^{(j)}\}_{i\in[k],j\in\{0,1\}}, \kappa)$
$(pk_i^{(j)}, sk_i^{(j)}) \leftarrow \mathsf{PKG}(1^k)$ for $(i,j) \in [k] \times \{0,1\}$	Parse C as $(c_1, \ldots, c_k, \mathtt{DL})$
$PK \leftarrow (\{pk_i^{(j)}\}_{i\in[k],j\in\{0,1\}}, \kappa)$	$h \leftarrow \mathsf{H}_\kappa(\mathtt{DL})$
$SK \leftarrow (\{sk_i^{(j)}\}_{i\in[k],j\in\{0,1\}}, \kappa)$	View h as $(h_1 \| \ldots \| h_k) \in \{0,1\}^k$
Return (PK, SK)	$\alpha \leftarrow \mathsf{Dec}(sk_1^{(h_1)}, c_1)$
$\mathsf{Encap}(PK)$:	If $\alpha = \perp$ then return $\perp$
Parse PK as $(\{pk_i^{(j)}\}_{i\in[k],j\in\{0,1\}}, \kappa)$	$\beta \leftarrow \mathtt{DL}(\alpha)$
$\alpha \leftarrow \mathcal{X}_k$; $\beta \leftarrow \{0,1\}^t$	If $\beta = \perp$ then return $\perp$
$\mathtt{DL} \leftarrow \mathsf{MBPO}(\mathcal{I}_{\alpha\to\beta})$	Parse β as $(r_1, \ldots, r_k, K)$ $\in (\{0,1\}^{\ell_R})^k \times \{0,1\}^k$
$h \leftarrow \mathsf{H}_\kappa(\mathtt{DL})$	If $\forall i \in [k] : \mathsf{Enc}(pk_i^{(h_i)}, \alpha; r_i) = c_i$ then return K else return $\perp$
View h as $(h_1 \| \ldots \| h_k) \in \{0,1\}^k$	
Parse β as $(r_1, \ldots, r_k, K) \in (\{0,1\}^{\ell_R})^k \times \{0,1\}^k$	
$c_i \leftarrow \mathsf{Enc}(pk_i^{(h_i)}, \alpha; r_i)$ for $i \in [k]$	
$C \leftarrow (c_1, \ldots, c_k, \mathtt{DL})$	
Return (C, K)	

Fig. 1. The proposed CCA2 secure KEM Γ.

The second property is the existence of the "alternative" decapsulation algorithm AltDecap. For a k-bit string $h^* = (h_1^* \| \ldots \| h_k^*) \in \{0,1\}^k$ and a key pair (PK, SK) output by $\mathsf{KKG}(1^k)$, where $SK = (\{sk_i^{(j)}\}_{i\in[k],j\in\{0,1\}}, \kappa)$, we define the "alternative" secret key $\widehat{SK}_{h^*}$ associated with h^* by $\widehat{SK}_{h^*} = (h^*, PK, \{sk_i^{(1-h_i^*)}\}_{i\in[k]})$. AltDecap takes an "alternative" secret key $\widehat{SK}_{h^*}$ and a ciphertext $C = (c_1, \ldots, c_k, \mathtt{DL})$ as input, and runs as follows:

$\mathsf{AltDecap}(\widehat{SK}_{h^*}, C)$**:** First check if $\mathsf{H}_\kappa(\mathtt{DL}) = h^*$, and return $\perp$ if this is the case. Otherwise, let $h = \mathsf{H}_\kappa(\mathtt{DL})$ and let $\ell \in [k]$ be the smallest index such that $h_\ell = 1 - h_\ell^*$, where h_ℓ is the ℓ-th bit of h. (Note that such ℓ must exist because $h \neq h^*$ in this case.) Run in exactly the same way as $\mathsf{Decap}(SK, C)$, except that it executes $\mathsf{Dec}(sk_\ell^{(1-h_\ell^*)}, c_\ell)$ in the fifth step, instead of executing $\mathsf{Dec}(sk_1^{(h_1)}, c_1)$.

Regarding AltDecap, the following lemma holds due to the symmetric role of each of $sk_i^{(j)}$ and the validity check of each c_i by re-encryption performed at the last step.

Lemma 3. *Let $h^* \in \{0,1\}^k$ be a string, (PK, SK) be a key pair output by $\mathsf{KKG}(1^k)$, and $\widehat{SK}_{h^*}$ be an alternative secret key defined as above. Then, for any ciphertext $C = (c_1, \ldots, c_k, \mathtt{DL})$ (which could be outside the range of $\mathsf{Encap}(PK)$) satisfying $\mathsf{H}_\kappa(\mathtt{DL}) \neq h^*$, it holds that $\mathsf{Decap}(SK, C) = \mathsf{AltDecap}(\widehat{SK}_{h^*}, C)$.*

The formal proofs of Lemmas 2 and 3 are given in the full version.

CCA2 *Security of Γ.* The security of Γ is guaranteed by the following theorem. (The formal proof is given in the full version.)

Theorem 1. *Assume that Π is ϵ-CPA secure with negligible ϵ, $\mathcal{H}$ is a UOWHF, and* MBPO *is* AIND-δ-cPUAI *secure with $\delta(k) \geq k\epsilon(k)$. Then, the KEM Γ constructed as in Fig. 1 is* CCA2 *secure.*

Proof Sketch of Theorem 1. Let $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ be any PPTA adversary that attacks the $\mathtt{CCA2}$ security of the KEM Γ. Consider the following sequence of games: (Here, the values with asterisk (*) represent those related to the challenge ciphertext for $\mathcal{A}$.)

Game 1: This is the experiment $\mathsf{Expt}^{\mathtt{CCA2}}_{\Gamma,\mathcal{A}}(k)$ itself. Without loss of generality, we generate the challenge ciphertext $C^* = (c_1^*, \ldots, c_k^*, \mathtt{DL}^*)$ and the challenge session-key K_b^* for $\mathcal{A}$, where b is the challenge bit for $\mathcal{A}$, before running $\mathcal{A}_1$. (Note that this modification does not affect $\mathcal{A}$'s behavior.)

Game 2: Same as Game 1, except that all decapsulation queries $C = (c_1, \ldots, c_k, \mathtt{DL})$ satisfying $\mathtt{DL} = \mathtt{DL}^*$ are answered with $\perp$.

Game 3: Same as Game 2, except that all decapsulation queries $C = (c_1, \ldots, c_k, \mathtt{DL})$ satisfying $\mathsf{H}_\kappa(\mathtt{DL}) = h^* = \mathsf{H}_\kappa(\mathtt{DL}^*)$ are answered with $\perp$.

Game 4: Same as Game 3, except that all decapsulation queries C are answered with $\mathsf{AltDecap}(\widehat{SK}_{h^*}, C)$, where $\widehat{SK}_{h^*}$ is the alternative secret key corresponding to (PK, SK) and $h^* = \mathsf{H}_\kappa(\mathtt{DL}^*) \in \{0,1\}^k$.

Game 5: Same as Game 4, except that $\mathtt{DL}^*$ is replaced with an obfuscation of the MBPF $\mathcal{I}_{\alpha^* \to \beta'}$ with an independently chosen random value $\beta' \in \{0,1\}^t$. That is, the step "$\mathtt{DL}^* \leftarrow \mathsf{MBPO}(\mathcal{I}_{\alpha^* \to \beta^*})$" is replaced with the steps "$\beta' \leftarrow \{0,1\}^t$; $\mathtt{DL}^* \leftarrow \mathsf{MBPO}(\mathcal{I}_{\alpha^* \to \beta'})$." (Note that each r_i^* and K_1^* are still generated from β^*.)

For $i \in [5]$, let S_i be the event that $\mathcal{A}$ succeeds in guessing the challenge bit (i.e. $b' = b$ occurs) in Game i. Using the above notation, $\mathcal{A}$'s $\mathtt{CCA2}$ advantage can be calculated as follows:

$$\mathsf{Adv}^{\mathtt{CCA2}}_{\Gamma,\mathcal{A}}(k) = 2 \cdot |\Pr[\mathsf{S}_1] - \frac{1}{2}| \leq 2 \cdot \sum_{i \in [4]} |\Pr[\mathsf{S}_i] - \Pr[\mathsf{S}_{i+1}]| + 2 \cdot |\Pr[\mathsf{S}_5] - \frac{1}{2}|. \quad (1)$$

To complete the proof, it remains to upperbound the right hand side of the above inequality (1).

Firstly, notice that the difference between Game 1 and Game 2 is only in how $\mathcal{A}$'s decapsulation query $C = (c_1, \ldots, c_k, \mathtt{DL})$ satisfying $\mathtt{DL} = \mathtt{DL}^*$ is answered. (It is answered with $\perp$ in Game 2, while it may be answered with some value that is not $\perp$ in Game 1.) However, due to Lemma 2, the only ciphertext C that contains $\mathtt{DL}^*$ and can be decapsulated to some value that is not $\perp$ is the challenge ciphertext C^* itself, and $\mathcal{A}_2$ is not allowed to ask it. Furthermore, since $\mathtt{DL}^*$ is information-theoretically hidden from $\mathcal{A}_1$'s view, the probability of $\mathcal{A}_1$ making a decapsulation query containing $\mathtt{DL}^*$ is negligible. Hence, the oracles behave almost identically in both Game 1 and Game 2, which shows that $|\Pr[\mathsf{S}_1] - \Pr[\mathsf{S}_2]|$ is negligible.

Next, notice that $|\Pr[\mathsf{S}_2] - \Pr[\mathsf{S}_3]|$ can be upperbounded by the probability of $\mathcal{A}$ making a decapsulation query $C = (c_1, \ldots, c_k, \mathtt{DL})$ satisfying $\mathsf{H}_\kappa(\mathtt{DL}) = h^* = \mathsf{H}_\kappa(\mathtt{DL}^*)$ and $\mathtt{DL} \neq \mathtt{DL}^*$ (because Game 2 and Game 3 proceed identically without such a query), but it is easy to see that this probability is negligible due to the security of the UOWHF $\mathcal{H}$.

It is also easy to see that $\Pr[\mathsf{S}_3] = \Pr[\mathsf{S}_4]$, because the behavior of the oracle in Game 3 and that in Game 4, are identical due to Lemma 3.

To show the upperbound of $|\Pr[\mathsf{S}_4] - \Pr[\mathsf{S}_5]|$, we need to use the $\mathtt{AIND}$-δ-$\mathtt{cPUAI}$ security of MBPO. We therefore first specify the auxiliary input function that we are

going to consider. Define the probabilistic function $\mathsf{ai}_{\Gamma} : \mathcal{X}_k \times \{0,1\}^t \to \{0,1\}^{t'}$ which takes $(\alpha, \beta) \in \mathcal{X}_k \times \{0,1\}^t$ as input, and computes $z = (\{pk_i\}_{i\in[k]}, c_1^*, \ldots, c_k^*, K^*) \in \{0,1\}^{t'}$ in the following way:

$$\begin{aligned}\mathsf{ai}_{\Gamma}(\alpha,\beta) : [\ &(pk_i, sk_i) \leftarrow \mathsf{PKG}(1^k) \text{ for } i \in [k];\\ &\text{Parse } \beta \text{ as } (r_1^*, \ldots, r_k^*, K^*) \in (\{0,1\}^{\ell_{\mathtt{R}}})^k \times \{0,1\}^k;\\ &c_i^* \leftarrow \mathsf{Enc}(pk_i, \alpha; r_i^*) \text{ for } i \in [k];\ \text{Return } z \leftarrow (\{pk_i\}_{i\in[k]}, c_1^*, \ldots, c_k^*, K^*)\ ].\end{aligned}$$

Note that ai_{Γ} is efficiently computable. Furthermore, due to the ϵ-CPA security of the underlying PKE scheme Π and the security of the k-repetition construction Π^k (which is $(k\epsilon)$-CPA secure based on the ϵ-CPA security of Π)[4], it is straightforward to see that ai_{Γ} is $(k\epsilon)$-computationally partially uninvertible (in particular, in the P-Inv experiment regarding ai_{Γ}, each r_i^* is a uniformly chosen randomness (independently of any other values), and thus we can rely on the CPA security of Π). In the full proof, we will show that there exists a PPTA $\mathcal{B}_o$ such that $\mathsf{Adv}^{\mathtt{AIND\text{-}AI}}_{\mathsf{MBPO},\mathsf{ai}_{\Gamma},\mathcal{B}_o}(k) = |\Pr[\mathsf{S}_4] - \Pr[\mathsf{S}_5]|$: $\mathcal{B}_o$ takes an auxiliary input $z = (\{pk_i\}_{i\in[k]}, c_1^*, \ldots, c_k^*, K^*) \leftarrow \mathsf{ai}_{\Gamma}(\alpha, \beta_0)$ and an obfuscated circuit $\mathtt{DL}^*$ which is either computed as $\mathsf{MBPO}(\mathcal{I}_{\alpha\to\beta_0})$ or $\mathsf{MBPO}(\mathcal{I}_{\alpha\to\beta_1})$ as input. $\mathcal{B}_o$ will generate $\mathcal{A}$'s challenge ciphertext C^* based on the auxiliary input z and the obfuscated ciphertext $\mathtt{DL}^*$ that it receives, and generates the remaining key materials, which enables $\mathcal{B}_o$ to generate the alternative key $\widehat{SK}_{h^*}$, and thus using $\mathsf{AltDecap}$, $\mathcal{B}_o$ can perfectly simulate the decryption oracle in Game 4 (and Game 5) for $\mathcal{A}$. Here, by regarding α, β_0, and β_1 in $\mathcal{B}_o$'s experiment as α^*, β^*, and β' (in Game 4 and Game 5), respectively, $\mathcal{B}_o$ will simulate the whole of Game 4 or Game 5 perfectly for $\mathcal{A}$ depending on the value of $\mathcal{B}$'s challenge bit, and we can derive $\mathsf{Adv}^{\mathtt{AIND\text{-}AI}}_{\mathsf{MBPO},\mathsf{ai}_{\Gamma},\mathcal{B}_o}(k) = |\Pr[\mathsf{S}_4] - \Pr[\mathsf{S}_5]|$. But here, since ai_{Γ} is a $(k\epsilon)$-cPUAI function and $\delta(k) \geq k\epsilon(k)$, the AIND-$\delta$-cPUAI security of MBPO guarantees that $|\Pr[\mathsf{S}_4] - \Pr[\mathsf{S}_5]|$ is negligible.

Finally, observe that in Game 5, the "real" session-key K_1^* is independent of the challenge ciphertext C^* and thus the challenge session-key K_b^* (together with other values available to $\mathcal{A}$ in Game 5) is distributed identically regardless of $\mathcal{A}$'s challenge bit b. This implies $\Pr[\mathsf{S}_5] = 1/2$.

Therefore, the right hand side of the inequality (1) is shown to be negligible, which implies that Γ is CCA2 secure. □

4.2 Second Construction

In the first construction shown above, we used an ordinary CPA secure PKE scheme for Π. Now, we consider the construction of the KEM Γ in which Π is replaced with a lossy encryption scheme. Π now has the lossy key generation algorithm LKG, and thus is of the form $\Pi = (\mathsf{PKG}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{LKG})$. (The lossy key generation algorithm LKG is actually not used in the construction, and is used only in the security proof.) Because of this change, we can now relax the requirement for the MBPF obfuscator

[4] Here, by "k-repetition construction" Π^k we mean the PKE scheme in which a public key consists of k independently generated public keys of Π, and a ciphertext consists of k ciphertexts of a same plaintext.

MBPO to be secure in the presence of only statistically partially uninvertible auxiliary input. This result is captured by the following theorem. (The formal proof is given in the full version.)

Theorem 2. *Assume Π is an ϵ-lossy encryption scheme with negligible ϵ, $\mathcal{H}$ is a UOWHF, and* MBPO *is* AIND-δ-sPUAI *secure with $\delta(k) \geq k\epsilon(k)$. Then, the KEM Γ constructed as in Fig. 1 is* CCA2 *secure.*

Proof Sketch of Theorem 2. The proof proceeds very similarly to that of Theorem 1. The main difference is that we consider an additional game between Game 4 and Game 5 (say, Game 4.5), in which we generate all public keys for $\{pk_i^{(h_i^*)}\}_{i\in[k]}$ by using the lossy key generation algorithm $\mathsf{LKG}(1^k)$, instead of $\mathsf{PKG}(1^k)$, where h_i^* is the i-th bit of $h^* = \mathsf{H}_\kappa(\mathtt{DL}^*)$. Then the difference between a CCA2 adversary $\mathcal{A}$'s success probability in Game 4 and that in Game 4.5 can be bounded to be negligible by the indistinguishability of keys of the k-repetition lossy encryption scheme Π^k. In particular, the corresponding secret keys $\{sk_i^{(h_i^*)}\}_{i\in[k]}$ are already not used in Game 4, and the reduction algorithm (for distinguishing ordinary/lossy public keys of Π^k) need not use them. Correspondingly to the above, in order to show that the difference between $\mathcal{A}$'s success probability in Game 4.5 and that in Game 5 is negligible, we will use the AIND-δ-sPUAI security of MBPO, with the auxiliary input $\mathsf{ai}'_\Gamma : \mathcal{X}_k \times \{0,1\}^t \to \{0,1\}^{t'}$ that is defined in the same way as ai_Γ used in the proof of Theorem 1 except that the public keys $\{pk_i\}_{i\in[k]}$ are generated by executing the lossy key generation algorithm $\mathsf{LKG}(1^k)$. Since the keys $\{pk_i\}_{i\in[k]}$ are generated from LKG, due to ϵ-lossiness of the lossy encryption scheme Π and $(k\epsilon)$-lossiness of the k-repetition construction Π^k (where $(k\epsilon)$-lossiness of Π^k based on ϵ-lossiness of Π can be shown by a standard hybrid argument), we can easily see that ai'_Γ is a $(k\epsilon)$-sPUAI function. The rest of the proof proceeds identically to that of Theorem 1. □

4.3 Extensions

A-priori Fixed and Bounded-length Auxiliary Input Functions. Note that for both of our proposed constructions, the auxiliary input functions under which the building block MBPF obfuscator MBPO needs to be secure, are dependent only on the building block PKE/lossy encryption scheme Π, which is fixed when Π is fixed. In particular, MBPO is required to satisfy AIND-δ-cPUAI (and AIND-δ-sPUAI) security only for t'-bounded δ-cPUAI (and δ-sPUAI) functions with $t'(k) = k \cdot \ell_{\mathsf{PK}}(k) + k \cdot \ell_{\mathsf{C}}(k) + k$. This a-priori bounded output length for auxiliary input functions might make it easier to achieve AIND-δ-cPUAI (and AIND-δ-sPUAI) secure MBPF obfuscators. We note that a similar observation on the possibility of weakening the requirement regarding auxiliary inputs by bounding the output length is also given in [9].

Using MBPF Obfuscators with Short Point Values. In our constructions, the MBPF obfuscator MBPO needs to obfuscate an MBPF $\mathcal{I}_{\alpha\to\beta}$ whose point value β is relatively long (which consists of k randomness $\{r_i\}_{i\in[k]}$ and a k-bit string K). For our first construction, however, we can shorten the length of a point value of MBPFs that need to be obfuscated by utilizing a pseudorandom generator (PRG). More specifically, let

$G : \{0,1\}^k \to \{0,1\}^t$ be a PRG (where $t(k) = k \cdot \ell_{\mathtt{R}}(k) + k$). Then instead of picking $\{r_i\}_{i\in[k]}$ and $K \in \{0,1\}^k$ uniformly at random, these values can be generated from a short seed $s \in \{0,1\}^k$ by $\beta = (r_1\| \ldots \|r_k\|K) \leftarrow G(s)$, and now we only need to obfuscate $\mathcal{I}_{\alpha\to s}$, instead of $\mathcal{I}_{\alpha\to\beta}$. However, this modification is at the cost of a stronger requirement for AIND-δ-cPUAI security of MBPO. That is, now δ has to be large enough to incorporate the security of the used PRG. Specifically, if the PRG is ϵ_g-secure, then it is required that $\delta \geq k\epsilon + \epsilon_g$ (where a PRG is said to be ϵ-secure if all PPTA adversaries have at most advantage $\epsilon = \epsilon(k)$ in distinguishing a pseudorandom value from a truly random value for all sufficiently large $k \in \mathbb{N}$). We note that this idea of using a PRG does not work for our second construction, because we cannot use a pseudorandom string as a randomness in the encryption algorithm of a lossy encryption scheme. Using a pseudorandomness violates the statistical lossiness property in general.

A Simpler Construction with CCA1 *Security.* We can show that a simpler variant of the proposed construction which employs the Naor-Yung-style double encryption [52] (instead of the Dolev-Dwork-Naor-style multiple encryption), leads a CCA1 secure KEM. This KEM is partly explained in Introduction, and we will show the details in the full version. Interestingly, unlike our CCA2 secure constructions, in the proof of this CCA1 secure variant, we need to use an auxiliary input function that internally runs (a part of) a CCA1 adversary, and thus its output length cannot be a-priori bounded.

5 Relations among Security Notions for MBPF Obfuscators

In this section, we investigate the relations between our new indistinguishability-based security notions for MBPF obfuscators, AIND-δ-cPUAI/sPUAI, and the worst-/average-case virtual black-/grey-box properties in the presence of auxiliary inputs. For the average-case virtual black-/grey-box properties, we consider the auxiliary input functions defined in Section 3.1, and show that our new security notions are implied by the average-case virtual black-/grey-box properties with the same type of auxiliary inputs.

We first formally define the average-case virtual black-/grey-box properties. For notational convenience, for an MBPF obfuscator MBPO, a probabilistic algorithm $\mathcal{M}$ whose output is restricted to be a bit, and a two-input probabilistic function $\mathsf{ai} : \mathcal{X}_k \times \{0,1\}^t \to \{0,1\}^*$, we define the following three experiments:

$\mathsf{Expt}^{\mathtt{Real}}_{\mathsf{MBPO},\mathsf{ai},\mathcal{M}}(k)$:	$\mathsf{Expt}^{\mathtt{Sim}}_{\mathsf{ai},\mathcal{M}}(k)$:	$\mathsf{Expt}^{\mathtt{s\text{-}Sim}}_{\mathsf{ai},\mathcal{M}}(k)$:
$\alpha \leftarrow \mathcal{X}_k$	$\alpha \leftarrow \mathcal{X}_k$	$\alpha \leftarrow \mathcal{X}_k$
$\beta \leftarrow \{0,1\}^t$	$\beta \leftarrow \{0,1\}^t$	$\beta \leftarrow \{0,1\}^t$
$z \leftarrow \mathsf{ai}(\alpha,\beta)$	$z \leftarrow \mathsf{ai}(\alpha,\beta)$	$z \leftarrow \mathsf{ai}(\alpha,\beta)$
$\mathtt{DL} \leftarrow \mathsf{MBPO}(\mathcal{I}_{\alpha\to\beta})$	Return $b \leftarrow \mathcal{M}^{\mathcal{I}_{\alpha\to\beta}}(1^k, z)$	Return $b \leftarrow \mathcal{M}(1^k, z)$
Return $b \leftarrow \mathcal{M}(1^k, z, \mathtt{DL})$		

(Note that in $\mathsf{Expt}^{\mathtt{s\text{-}Sim}}_{\mathsf{ai},\mathcal{M}}(k)$, the algorithm $\mathcal{M}$ does not have access to any oracle.)

Definition 7. *We say that an MBPF obfuscator* MBPO *satisfies*

– *the* average-case virtual black-box property w.r.t. δ-computationally (resp. δ-statistically) partially uninvertible auxiliary input *(*AVB-δ-cPUAI *(resp.* AVB-δ-sPUAI*) security, for short), if for every PPTA $\mathcal{A}$ and all positive polynomials $q =$*

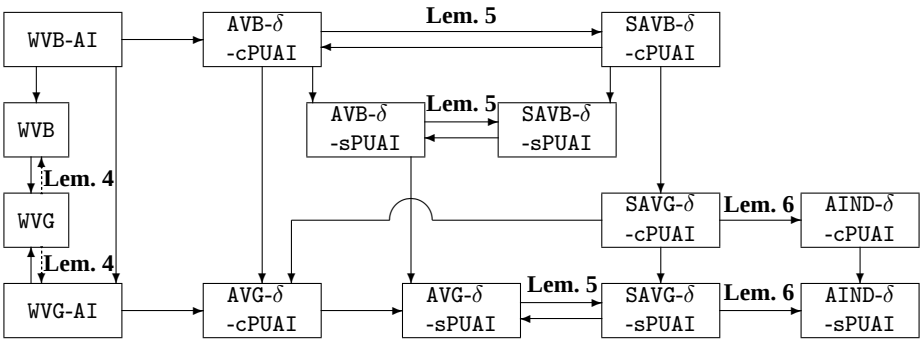

Fig. 2. Relations among security notions for MBPF obfuscators defined in this paper. The arrow "$\mathtt{X} \to \mathtt{Y}$" indicates that $\mathtt{X}$-security implies $\mathtt{Y}$-security. The dotted arrows indicate the implications that hold only for the non-uniform setting in which an adversary (and a simulator) are non-uniform algorithms. In the figure, δ is a negligible function.

$q(k)$ *and* $\ell = \ell(k)$*, there exists a PPTA* $\mathcal{S}$ *such that for every* ℓ*-bounded* δ-cPUAI *(resp.* δ-sPUAI*) function* ai *and all sufficiently large* $k \in \mathbb{N}$*, it holds that*

$$\mathsf{Adv}^{\text{A-MBPO-AI}}_{\mathsf{MBPO},\mathsf{ai},\mathcal{A},\mathcal{S}}(k) := |\Pr[\mathsf{Expt}^{\mathtt{Real}}_{\mathsf{MBPO},\mathsf{ai},\mathcal{A}}(k) = 1] - \Pr[\mathsf{Expt}^{\mathtt{Sim}}_{\mathsf{ai},\mathcal{S}}(k) = 1]| \leq 1/q.$$

– *the* strong average-case virtual black-box property w.r.t. δ-computationally (resp. δ-statistically) partially uninvertible auxiliary input *(*SAVB-δ-cPUAI *(resp.* SAVB-δ-sPUAI*) security, for short), if for every PPTA* $\mathcal{A}$ *and all positive polynomials* $q = q(k)$ *and* $\ell = \ell(k)$*, there exists a PPTA* $\mathcal{S}$ *such that for every* ℓ*-bounded* δ-cPUAI *(resp.* δ-sPUAI*) function* ai *and all sufficiently large* $k \in \mathbb{N}$*, it holds that*

$$\mathsf{Adv}^{\text{SA-MBPO-AI}}_{\mathsf{MBPO},\mathsf{ai},\mathcal{A},\mathcal{S}}(k) := |\Pr[\mathsf{Expt}^{\mathtt{Real}}_{\mathsf{MBPO},\mathsf{ai},\mathcal{A}}(k) = 1] - \Pr[\mathsf{Expt}^{\mathtt{s\text{-}Sim}}_{\mathsf{ai},\mathcal{S}}(k) = 1]| \leq 1/q.$$

Furthermore, we define the (strong) average-case virtual grey-box property w.r.t. δ-computationally (resp. δ-statistically) partially uninvertible auxiliary input *(*($\mathtt{S}$)AVG-δ-cPUAI *(resp.* ($\mathtt{S}$)AVG-δ-sPUAI*) security for short) for an MBPF obfuscator* MBPO*, in the same way as the definitions for the corresponding virtual black-box properties, except that we replace "a PPTA* $\mathcal{S}$*" in each definition with "a computationally unbounded algorithm* $\mathcal{S}$ *that makes only polynomially many queries."*

Now, we show the relations among security notions, which are summarized in Fig. 2. Most of the relations are obvious. Namely, the virtual black-box properties always imply the virtual grey-box properties for the same class of auxiliary inputs. Furthermore, WVB-AI security implies AVB-δ-cPUAI security for arbitrary (not necessarily negligible) δ, and AVB-δ-cPUAI security implies AVB-δ-sPUAI security because the class of δ-sPUAI functions are smaller than the class of δ-cPUAI functions for the same δ. Moreover, by definition, for both $\mathtt{X} \in \{\delta\text{-}\mathtt{cPUAI}, \delta\text{-}\mathtt{sPUAI}\}$, SAVB-X and SAVG-X imply AVB-X and AVG-X, respectively, because the former notions consider simulators that do not make any oracle queries and thus can also be used as simulator for the latter.

In the following, we show the implications of the non-trivial directions. The following equivalence is due to the result by Bitansky and Canetti [7]. (Note that the following

results are only for non-uniform PPTA adversaries, while our default notions in this paper are with respect to uniform PPTA adversaries.)

Lemma 4. *([8, Propositions 8.3 and A.3]) For MBPF obfuscators,* WVB *security for non-uniform PPTA adversaries with non-uniform PPTA simulators,* WVG *security for non-uniform PPTA adversaries, and* WVG-AI *security for PPTA non-uniform adversaries, are equivalent.*

The following is useful for showing the implication to the AIND security notions that we will show later.

Lemma 5. *Let $\delta : \mathbb{N} \to [0,1]$ be a negligible function. For MBPF obfuscators, for both* X $\in \{\delta$-cPUAI$, \delta$-sPUAI$\}$*,* AVB-X *security and* SAVB-X *security are equivalent. Furthermore,* AVG-δ-sPUAI *security and* SAVG-δ-sPUAI *security are equivalent.*

Intuition. For both cPUAI and sPUAI cases, the implication from the latter to the former is trivial by definition. The implications of the opposite directions can be shown because the partial uninvertibility of an auxiliary input function guarantees that a simulator cannot find the point address of the MBPF being obfuscated and thus having oracle access to an MBPF does not give much advantage. The computational uninvertibility and statistical uninvertibility naturally correspond to the uninvertibility of auxiliary input functions against a PPTA simulator and that against a computationally unbounded simulator, respectively.

Finally, the following implications clarify that AIND notions introduced in Section 3.2 are indeed implied by the average-case virtual black-box/grey-box properties.

Lemma 6. *Let $\delta : \mathbb{N} \to [0,1]$ be a negligible function. For both* X $\in \{\delta$-cPUAI$, \delta$-sPUAI$\}$*, if an MBPF obfuscator is* SAVG-X *secure, then it is* AIND-X *secure.*

Intuition. This lemma is shown by considering a hybrid experiment in which a (computationally unbounded) simulator $\mathcal{S}$ (due to SAVG-δ-cPUAI/sPUAI security) is given only an auxiliary input $\mathsf{ai}(\alpha, \beta)$ (for randomly chosen (α, β)) as input, and outputs a bit.; By the SAVG-δ-cPUAI/sPUAI security, for both cases $b \in \{0,1\}$, the probability that an adversary (attacking the AIND-δ-cPUAI/sPUAI security) on input $\mathsf{ai}(\alpha, \beta_0)$ and $\mathsf{MBPO}(\mathcal{I}_{\alpha \to \beta_b})$ (for randomly chosen α, β_0, β_1) outputs 1 can be shown to be negligibly close to the probability that the simulator $\mathcal{S}$ outputs 1 in the hybrid experiment, which proves the lemma.

6 Lossy Encryption from Re-randomizable Point Obfuscation

In this section, we show that a re-randomizable point obfuscator yields a lossy encryption scheme. We first recall the definition of re-randomizability [7].

Definition 8. *([7]) Let $\mathcal{X} = \{\mathcal{X}_k\}_{k \in \mathbb{N}}$ be a domain ensemble and let* PO *be a point obfuscator for* $\mathsf{PF}(\mathcal{X})$ *whose randomness space is $\{0,1\}^{\ell(k)}$. We say that* PO *is* re-randomizable *if there exists a PPTA* ReRand *(called the re-randomization algorithm) such that for all $k \in \mathbb{N}$, all $\alpha \in \mathcal{X}_k$, and for all $r \in \{0,1\}^{\ell}$, the distribution of* $\mathsf{ReRand}(\mathsf{PO}(\mathcal{I}_\alpha; r))$ *and the distribution of* $\mathsf{PO}(\mathcal{I}_\alpha)$ *are identical.*

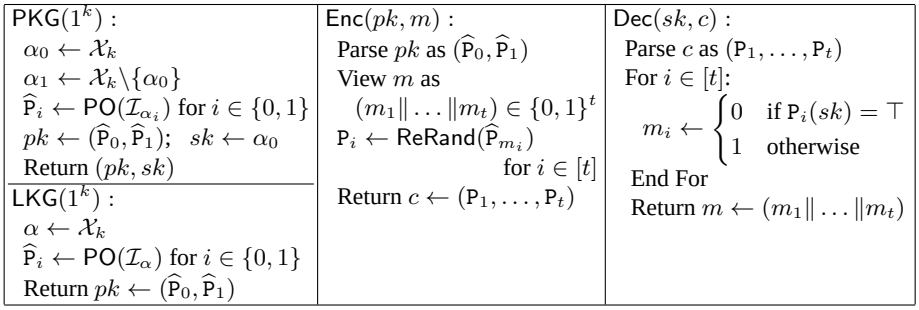

Fig. 3. Lossy encryption from a re-randomizable point obfuscator.

We note that the point obfuscator based on the perfect one-way hash function by Canneti [17] is re-randomizable. (We review the construction in Appendix B.)

Now, we formally describe our proposed lossy encryption scheme. Let $\mathcal{X} = \{\mathcal{X}_k\}_{k \in \mathbb{N}}$ be a domain ensemble, and let PO be a re-randomizable point obfuscator for $\mathsf{PF}(\mathcal{X})$ with the re-randomization algorithm ReRand, and let $t = t(k) > 0$ be a polynomial. Then we construct a lossy encryption scheme $\Pi = (\mathsf{PKG}, \mathsf{Enc}, \mathsf{Dec}, \mathsf{LKG})$ whose plaintext space is $\{0,1\}^t$ as in Fig. 3.

Our construction is inspired partly by the construction of a PKE scheme from a re-randomizable point obfuscator due to Bitansky and Canetti [7], and partly by the construction of lossy encryption from a re-randomizable encryption scheme due to Hemenway et al. [39]. The following theorem guarantees that Π constructed as above is indeed a lossy encryption scheme. (The formal proof is given in the full version.)

Theorem 3. *If* PO *is re-randomizable and* 2*-composable, then* Π *constructed as in Fig. 3 is a* 0*-lossy encryption scheme.*

Intuition. Theorem 3 is shown by using the equivalence of t-composability and *t-distributional indistinguishability for coordinate-wise well-spread (CWS) distributions*, established by Bitansky and Canetti [8]. The latter property roughly states that if $(\alpha_1, \dots, \alpha_t)$ are chosen from a distribution so that each α_i has high min-entropy (but α_i's could be arbitrarily correlated), $(\mathsf{PO}(\alpha_1), \dots, \mathsf{PO}(\alpha_t))$ is computationally indistinguishable from $(\mathsf{PO}(u_1), \dots, \mathsf{PO}(u_t))$ where each u_i is chosen uniformly at random (the formal definition appears in the full version). This property can be used to show the indistinguishability of keys, which is easy to see due to the design of PKG and LKG. Moreover, note that a lossy key consists of a pair of obfuscated circuits of point functions with a same point address. Therefore, due to the re-randomizability, an encryption of any plaintext have identical distribution, which implies 0-statistical lossiness.

CCA2 Secure PKE/KEM Based Solely on Re-randomizable, Composable Point Obfuscators. Recall that when considering non-uniform PPTA adversaries, WVB security (with non-uniform PPTA simulators), WVG security, and WVG-AI security for MBPF obfuscators are equivalent (see Lemma 4). Therefore, the WVG secure MBPF obfuscator for t-bit point values due to [19, 7] based on a $(t+1)$-composable point obfuscator can be used as an AIND-δ-sPUAI secure MBPF obfuscator (with any negligible δ). Note that if we

denote by ℓ the length of the randomness used by ReRand, then the randomness length $\ell_{\mathtt{R}}$ of the lossy encryption scheme Π for the k-bit plaintext space is $\ell_{\mathtt{R}}(k) = k \cdot \ell(k)$. Combining these results with our second generic construction, we obtain the following.

Theorem 4. *Assume there exists a point obfuscator which is (1) re-randomizable where* ReRand *uses $\ell(k)$-bit randomness, and (2) $(k^2 \cdot \ell(k) + k + 1)$-composable for non-uniform PPTA adversaries. Then there exists a* CCA2 *secure PKE scheme/KEM.*

7 Discussion

On Replacing MBPF Obfuscators with SKE. As has been clarified in several previous works [19, 27, 37, 21], there is a strong connection between MBPF obfuscators and SKE schemes. More specifically, an MBPF obfuscator can always be used as a SKE scheme. In order for the opposite direction to be true, among other things regarding security, it is necessary that a SKE scheme has the property called the *unique-key* property [27, 37, 21]. Therefore, a variant of our KEM Γ in Section 4 in which an MBPF obfuscator is replaced with a SKE scheme that has the unique-key property and satisfies the security that we call AIND-δ-cPUAI (and AIND-δ-sPUAI) security (which is defined similarly to that for MBPF obfuscator), can also be proved CCA2 secure.

Since the unique-key property is not satisfied by SKE schemes in general, it may be the case that a SKE scheme is in general a weaker primitive than an MBPF obfuscator, and is potentially easier to achieve. (Although a generic transformation of a SKE scheme into one that has this property was proposed in [21], we could not figure out whether this transformation preserves AIND-δ-cPUAI security and AIND-δ-sPUAI security.) Motivated by this observation, in the full version we will show another variant of the proposed KEM based on a SKE scheme without the unique-key property.

On the Difficulty of Achieving AIND-δ-cPUAI *Security.* We have shown that AIND-δ-sPUAI security is implied by the virtual grey-box properties (see Fig. 2), and thus by the results established by [19, 7] we can construct an AIND-δ-sPUAI secure MBPF obfuscator (or SKE) from any composable point obfuscator. Unfortunately, however, we could not come up with a natural assumption that is sufficient to realize an AIND-δ-cPUAI secure MBPF obfuscator, and we would like to leave it as an interesting open problem. In the full version, we will show that constructing it is at least as difficult as constructing a SKE scheme which is one-time chosen plaintext secure in the presence of computationally hard-to-invert leakage where leakage occurs only from a key. There, we will also show that the MBPF obfuscator by Lynn et al. [48] can be shown to be AIND-δ-cPUAI secure for any negligible δ. This at least suggests that it can be achieved under a strong assumption. We conjecture that the MBPF obfuscator by Lynn et al. can be shown to be AIND-δ-cPUAI secure for any negligible δ if we instantiate the random oracle as a family of hash functions satisfying (some version of) UCE security that is recently introduced by Bellare et al. [5].

We see that the difficulty of achieving AIND-δ-cPUAI security is that it allows a leakage from a random point address/value pair (α, β) (or a key/message pair in the case of SKE) that could be arbitrarily correlated, as long as partial uninvertibility is satisfied. This definition allows β to be (a part of) the source of the hardness of the partial

uninvertibility. For example, we could consider an auxiliary input function $\mathsf{ai}(\alpha, \beta)$ that returns an encryption of the "plaintext" α under the "key" β, using some SKE scheme, which will be a δ-$\mathtt{cPUAI}$ function under a reasonable assumption on the SKE scheme. This is quite different from a usual indistinguishability-based security definition (e.g. $\mathtt{CPA}$ security of a SKE scheme) in which a point value (or a message in SKE) is chosen by an adversary, and thus cannot be a source of hardness. This is one of the reasons why we cannot straightforwardly use the existing results on MBPF obfuscators/SKE [27, 21] (or a stronger primitive of PKE secure under hard-to-invert leakage [26]). We notice that the formulation of $\mathtt{AIND}$-δ-$\mathtt{cPUAI}$ security looks close to the security definition for deterministic encryption in the hard-to-invert auxiliary input setting [16], which considers a leakage from a plaintext (as opposed to a key). This setting is in some sense a "dual" of the settings that consider leakage only from a key. We also notice the similarity to the notion called security under *chosen distribution attacks* [4] that considers the security under a correlated leakage from a message and randomness simultaneously (this is a security notion for PKE but can be considered for SKE as well), but this does not consider a leakage from a key or leakage with computational uninvertibility. It would be worth clarifying further whether it is possible to leverage techniques from these various kinds of "leakage resilient" cryptography for achieving $\mathtt{AIND}$-δ-$\mathtt{cPUAI}$/$\mathtt{sPUAI}$ secure MBPF obfuscators/SKE schemes.

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

References

1. P. Ananth, D. Boneh, S. Garg, A. Sahai, and M. Zhandry. Differing-inputs obfuscation and Applications, 2013. http://eprint.iacr.org/2013/689.
2. B. Barak, N. Bitansky, R. Canetti, Y.T. Kalai, O. Paneth, and A. Sahai. Obfuscation for evasive functions, 2013. To appear in TCC 2014. http://eprint.iacr.org/2013/668.
3. B. Barak, O. Goldreich, R. Impagliazzo, S. Rudich, A. Sahai, S.P. Vadhan, and K. Yang. On the (im)possibility of obfuscating programs. In *CRYPTO 2001*, LNCS 2139, pp. 1–18, 2001.
4. M. Bellare, Z. Brakerski, M. Naor, T. Ristenpart, G. Segev, H. Shacham, and S. Yilek. Hedged public-key encryption: How to protect against bad randomness. In *ASIACRYPT 2009*, LNCS 5912, pp. 232–249, 2009.
5. M. Bellare, V.T. Hoang, and S. Keelveedhi. Instantiating random oracles via UCEs. In *CRYPTO 2013(2)*, LNCS 8043, pp. 398–415, 2013.
6. M. Bellare, D. Hofheinz, and S. Yilek. Possibility and impossibility results for encryption and commitment secure under selective opening. In *EUROCRYPT 2009*, LNCS 5479, pp. 1–35, 2009.
7. N. Bitansky and R. Canetti. On strong simulation and composable point obfuscation, 2010. Full version of [8]. http://eprint.iacr.org/2010/414.
8. N. Bitansky and R. Canetti. On strong simulation and composable point obfuscation. In *CRYPTO 2010*, LNCS 6223, pp. 520–537, 2010.
9. N. Bitansky and O. Paneth. Point obfuscation and 3-round zero-knowledge. In *TCC 2012*, LNCS 7194, pp. 190–208, 2012.

10. D. Bleichenbacher. Chosen ciphertext attacks against protocols based on the RSA encryption standard PKCS #1. In *CRYPTO 1998*, LNCS 1462, pp. 1–12, 1998.
11. D. Boneh, P.A. Papakonstantinou, C. Rackoff, Y. Vahlis, and B. Waters. On the impossibility of basing identity based encryption on trapdoor permutations. In *FOCS 2008*, pp. 283–292, 2008.
12. E. Boyle, K.-M. Chung, and R. Pass. On extractability obfuscation, 2013. http://eprint.iacr.org/2013/650.
13. Z. Brakerski and G.N. Rothblum. Black-box obfuscation for d-CNFs, 2013. To appear in ITCS 2014. http://eprint.iacr.org/2013/557.
14. Z. Brakerski and G.N. Rothblum. Obfuscating conjunctions. In *CRYPTO 2013(2)*, LNCS 8043, pp. 416–434, 2013.
15. Z. Brakerski and G.N. Rothblum. Virtual black-box obfuscation for all circuits via generic graded encoding, 2013. To appear in TCC 2014. http://eprint.iacr.org/2013/563.
16. Z. Brakerski and G. Segev. Better security for deterministic public-key encryption: The auxiliary-input setting. In *CRYPTO 2011*, 6841, pp. 543–560, 2011.
17. R. Canetti. Towards realizing random oracles: Hash functions that hide all partial information. In *CRYPTO 1997*, LNCS 1294, pp. 455–469, 1997.
18. R. Canetti. Universally composable security: A new paradigm for cryptographic protocols. In *FOCS 2001*, pp. 136–145, 2001.
19. R. Canetti and R.R. Dakdouk. Obfuscating point functions with multibit output. In *EUROCRYPT 2008*, LNCS 4965, pp. 489–508, 2008.
20. R. Canetti, S. Halevi, and J. Katz. Chosen-ciphertext security from identity-based encryption. In *EUROCRYPT 2004*, LNCS 3027, pp. 207–222, 2004.
21. R. Canetti, Y.T. Kalai, M. Varia, and D. Wichs. On symmetric encryption and point obfuscation. In *TCC 2010*, LNCS 5978, pp. 52–71, 2010.
22. R. Canetti, G.N. Rothblum, and M. Varia. Obfuscation of hyperplane membership. In *TCC 2010*, LNCS 5978, pp. 72–89, 2010.
23. J.S. Coron, T. Lepoint, and M. Tibouchi. Practical multilinear maps over the integers. In *CRYPTO 2013(1)*, LNCS 8042 of *LNCS*, 2013.
24. R. Cramer and V. Shoup. Design and analysis of practical public-key encryption schemes secure against adaptive chosen ciphertext attack. *SIAM J. Computing*, 33(1):167–226, 2003.
25. D. Dachman-Soled. A black-box construction of a CCA2 encryption scheme from a plaintext aware encryption scheme, 2013. http://eprint.iacr.org/2013/680.
26. Y. Dodis, S. Goldwasser, Y.T. Kalai, C. Peikert, and V. Vaikuntanathan. Public-key encryption with auxiliary inputs. In *TCC 2010*, LNCS 5978, pp. 361–381, 2010.
27. Y. Dodis, Y.T. Kalai, and S. Lovett. On cryptography with auxiliary input. In *STOC 2009*, pp. 621–630, 2009.
28. Y. Dodis and A. Smith. Correcting errors without leaking partial information. In *STOC 2005*, pp. 654–663, 2005.
29. D. Dolev, C. Dwork, and M. Naor. Non-malleable cryptography. In *STOC 1991*, pp. 542–552, 1991.
30. S. Garg, C. Gentry, and S. Halevi. Candidate multilinear maps from ideal lattices. In *EUROCRYPT 2013*, LNCS 7881, pp. 1–17, 2013.
31. S. Garg, C. Gentry, S. Halevi, and M. Raykova. Two-round secure MPC from indistinguishability obfuscation, 2013. To appear in TCC 2014. http://eprint.iacr.org/2013/601.
32. S. Garg, C. Gentry, S. Halevi, M. Raykova, A. Sahai, and B Waters. Candidate indistinguishability obfuscation and functional encryption for all curcuits, 2013. To appear in FOCS 2013. http://eprint.iacr.org/2013/451.
33. Y. Gertner, T. Malkin, and S. Myers. Towards a separation of semantic and CCA security for public key encryption. In *TCC 2007*, LNCS 4392, pp. 434–455, 2007.

34. Y. Gertner, T. Malkin, and O. Reingold. On the impossibility of basing trapdoor functions on trapdoor predicates. In *FOCS 2001*, pp. 126–135, 2001.
35. O. Goldreich and R.D. Rothblum. Enhancements of trapdoor permutations. *J. of Cryptology*, 26(3):484–512, 2013.
36. S. Goldwasser and Y.T. Kalai. On the impossibility of obfuscation with auxiliary input. In *FOCS 2005*, pp. 553–562, 2005.
37. S. Goldwasser, Y.T. Kalai, C. Peikart, and V. Vaikuntanathan. Robustness of the learning with errors assumption. In *ICS 2010*, pp. 230–240, 2010.
38. S. Goldwasser and G.N. Rothblum. On best-possible obfuscation. In *TCC 2007*, LNCS 4392, pp. 194–213, 2007.
39. B. Hemenway, B. Libert, R. Ostrovsky, and D. Vergnaud. Lossy encryption: Constructions from general assumptions and efficient selective opening chosen ciphertext security. In *ASIACRYPT 2011*, LNCS 7073, pp. 70–88, 2011.
40. B. Hemenway and R. Ostrovsky. On homomorphic encryption and chosen-ciphertext security. In *PKC 2012*, LNCS 7293, pp. 52–65, 2012.
41. B. Hemenway and R. Ostrovsky. Building lossy trapdoor functions from lossy encryption. In *ASIACRYPT 2013*, LNCS 8270, pp. 241–260, 2013.
42. S. Hohenberger, A. Lewko, and B. Waters. Detecting dangerous queries: A new approach for chosen ciphertext security. In *EUROCRYPT 2012*, LNCS 7237, pp. 663–681, 2012.
43. S. Hohenberger, G.N. Rothblum, A. Shelat, and V. Vaikuntanathan. Securely obfuscating re-encryption. In *TCC 2007*, LNCS 4392, pp. 233–252, 2007.
44. S. Hohenberger, A. Sahai, and B. Waters. Replacing a random oracle: Full domain hash from indistinguishability obfuscation, 2013. http://eprint.iacr.org/2013/509.
45. E. Kiltz. Chosen-ciphertext security from tag-based encryption. In *TCC 2006*, LNCS 3876, pp. 581–600, 2006.
46. E. Kiltz, P. Mohassel, and A. O'Neill. Adaptive trapdoor functions and chosen-ciphertext security. In *EUROCRYPT 2010*, LNCS 6110, pp. 673–692, 2010.
47. H. Lin and S. Tessaro. Amplification of chosen-ciphertext security. In *EUROCRYPT 2013*, LNCS 7881, pp. 503–519, 2013.
48. B. Lynn, M. Prabhakaran, and A. Sahai. Positive results and techniques for obfuscation. In *EUROCRYPT 2004*, LNCS 3027, pp. 20–39, 2004.
49. P. Mol and S. Yilek. Chosen-ciphertext security from slightly lossy trapdoor functions. In *PKC 2010*, LNCS 6056, pp. 296–311, 2010.
50. S. Myers and A. Shelat. Bit encryption is complete. In *FOCS 2009*, pp. 607–616, 2009.
51. M. Naor and M. Yung. Universal one-way hash functions and their cryptographic applications. In *STOC 1989*, pp. 33–43, 1989.
52. M. Naor and M. Yung. Public-key cryptosystems provably secure against chosen ciphertext attacks. In *STOC 1990*, pp. 427–437, 1990.
53. C. Peikert and B. Waters. Lossy trapdoor functions and their applications. In *STOC 2008*, pp. 187–196, 2008.
54. C. Rackoff and D.R. Simon. Non-interactive zero-knowledge proof of knowledge and chosen ciphertext attack. In *CRYPTO 1991*, LNCS 576, pp. 433–444, 1992.
55. A. Rosen and G. Segev. Chosen-ciphertext security via correlated products. In *TCC 2009*, LNCS 5444, pp. 419–436, 2009.
56. A. Sahai. Non-malleable non-interactive zero knowledge and adaptive chosen-ciphertext security. In *FOCS 1999*, pp. 543–553, 1999.
57. A. Sahai and B. Waters. How to use indistinguishability obfuscation: Deniable encryption, and more, 2013. http://eprint.iacr.org/2013/454.
58. H. Wee. On obfuscating point functions. In *STOC 2005*, pp. 523–532, 2005.
59. H. Wee. Efficient chosen-ciphertext security via extractable hash proofs. In *CRYPTO 2010*, LNCS 6223, pp. 314–332, 2010.

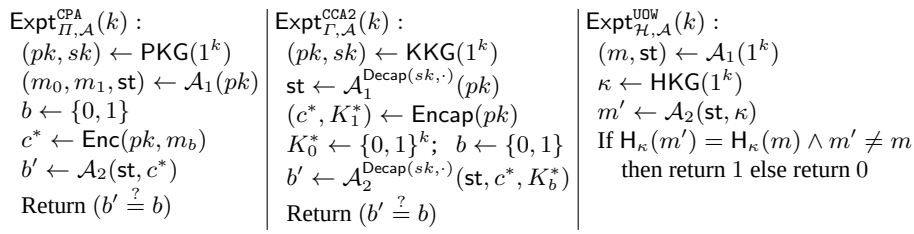

Fig. 4. The CPA security experiment for a PKE scheme Π (left), the CCA2 security experiment for a KEM Γ (center), and the security experiment for a UOWHF $\mathcal{H}$ (right).

A Basic Cryptographic Primitives

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

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

where Dec is a deterministic algorithm, (pk, sk) is a public/secret key pair, and c is a ciphertext of a plaintext m under pk. We require for all $k \in \mathbb{N}$, all (pk, sk) output by $\mathsf{PKG}(1^k)$, and all m, it holds that $\mathsf{Dec}(sk, \mathsf{Enc}(pk, m)) = m$.

We define the "*public key length*" $\ell_{\mathtt{PK}}(k)$ as the length of pk output by $\mathsf{PKG}(1^k)$. Moreover, if Enc can encrypt k-bit plaintexts (for security parameter k), we define the "*randomness length*" $\ell_{\mathtt{R}}(k)$ and the "*ciphertext length*" $\ell_{\mathtt{C}}(k)$, respectively, as the length of randomness used by Enc and the length of ciphertexts output from Enc.

We say that a PKE scheme Π is ϵ-CPA secure if for all PPTAs $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$ and for all sufficiently large $k \in \mathbb{N}$, it holds that $\mathsf{Adv}^{\mathtt{CPA}}_{\Pi,\mathcal{A}}(k) := 2 \cdot |\Pr[\mathsf{Expt}^{\mathtt{CPA}}_{\Pi,\mathcal{A}}(k) = 1] - 1/2| \leq \epsilon(k)$, where the experiment $\mathsf{Expt}^{\mathtt{CPA}}_{\Pi,\mathcal{A}}(k)$ is defined as in Fig. 4 (left). In the experiment, it is required that $|m_0| = |m_1|$.

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

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

where Decap is a deterministic algorithm, (pk, sk) is a public/secret key pair, and c is a ciphertext of a session-key $K \in \{0,1\}^k$ under pk. We require for all $k \in \mathbb{N}$, all (pk, sk) output by $\mathsf{KKG}(1^k)$, and all $(c, K) \leftarrow \mathsf{Encap}(pk)$, it holds that $\mathsf{Decap}(sk, c) = K$.

We say that a KEM Γ is CCA2 secure if for all PPTAs $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$, $\mathsf{Adv}^{\mathtt{CCA2}}_{\Gamma,\mathcal{A}}(k) := 2 \cdot |\Pr[\mathsf{Expt}^{\mathtt{CCA2}}_{\Gamma,\mathcal{A}}(k) = 1] - 1/2|$ is negligible, where the experiment $\mathsf{Expt}^{\mathtt{CCA2}}_{\Gamma,\mathcal{A}}(k)$ is defined as in Fig. 4 (center). In the experiment, $\mathcal{A}_2$ is not allowed to query c^*.

Universal One-Way Hash Function. We say that a pair of PPTAs $\mathcal{H} = (\mathsf{HKG}, \mathsf{H})$ is a universal one-way hash function (UOWHF) if the following two properties are satisfied:

$\mathsf{MBPO}(\mathcal{I}_{\alpha\to\beta})$:	$\mathcal{C}_{\mathtt{P}_0,\ldots,\mathtt{P}_t}(x)$:
$\mathtt{P}_0 \leftarrow \mathsf{PO}(\mathcal{I}_\alpha)$	If $\mathtt{P}_0(x) = \perp$ then return $\perp$
View β as $(\beta_1 \| \ldots \| \beta_t) \in \{0,1\}^t$	For $i \in [t]$:
$\alpha' \leftarrow \mathcal{X}_k \backslash \{\alpha\}$	$\beta_i \leftarrow \begin{cases} 1 & \text{if } \mathtt{P}_i(x) = \top \\ 0 & \text{otherwise} \end{cases}$
For $i \in [t]$:	End For
$\mathtt{P}_i \leftarrow \begin{cases} \mathsf{PO}(\mathcal{I}_\alpha) & \text{if } \beta_i = 1 \\ \mathsf{PO}(\mathcal{I}_{\alpha'}) & \text{otherwise} \end{cases}$	Return $\beta \leftarrow (\beta_1 \| \ldots \| \beta_t)$.
End For	
Return $\mathtt{DL} \leftarrow \mathcal{C}_{\mathtt{P}_0,\ldots,\mathtt{P}_t}$.	

Fig. 5. The construction of an MBPF obfuscator MBPO from a composable point obfuscator PO [19, 7]. MBPO takes an MBPF $\mathcal{I}_{\alpha\to\beta}$ as input, and returns a circuit $\mathtt{DL} = \mathcal{C}_{\mathtt{P}_0,\ldots,\mathtt{P}_t}$ that is described in the right column.

(1) On input 1^k, HKG outputs a hash-key κ. For any hash-key κ output from $\mathsf{HKG}(1^k)$, H defines an (efficiently computable) function of the form $\mathsf{H}_\kappa : \{0,1\}^* \to \{0,1\}^k$. (2) For all PPTAs $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$, $\mathsf{Adv}^{\mathtt{UOW}}_{\mathcal{H},\mathcal{A}}(k) := \Pr[\mathsf{Expt}^{\mathtt{UOW}}_{\mathcal{H},\mathcal{A}}(k) = 1]$ is negligible, where the experiment is defined as in Fig. 4 (right).

B Concrete Instantiations of Point/MBPF Obfuscators

Composable Point Obfuscator. Here we recall the point obfuscator due to Canetti [17] (which was originally introduced as a perfectly one-way hash function). Let $\mathbb{G}$ be a cyclic group with prime order p (where the size of p is determined by the security parameter k). Then, consider the following point obfuscator PO for $\mathsf{PF}(\mathbb{Z}_p)$:

$\mathsf{PO}(\mathcal{I}_\alpha)$**:** (where $\alpha \in \mathbb{Z}_p$) Pick a group element $r \leftarrow \mathbb{G}$ uniformly at random, and outputs the circuit $\mathcal{C}_{r,r^\alpha}(\cdot) : \mathbb{Z}_p \to \{\top, \perp\}$, where $\mathcal{C}_{A,B}$ is the circuit which takes $x \in \mathbb{Z}_p$ as input, and outputs $\top$ if $A^x = B$ and otherwise outputs $\perp$.

Bitansky and Canetti [7] showed that the above point obfuscator is t-composable, under a strong variant of the decisional Diffie-Hellman (DDH) assumption, called the t-strong vector DDH (t-SVDDH) assumption (see [7] for a formal definition).

We remark that as mentioned in [7], the point obfuscator based on the t-SVDDH assumption described here satisfies the re-randomizability in the sense of Definition 8. Specifically, we can just re-randomize two group elements in an obfuscated circuit output from PO without changing the point address.

$\mathtt{WVG}$ *Secure MBPF Obfuscator from Composable Point Obfuscator.* We recall the construction of an MBPF obfuscator based on a composable point obfuscator, due to Canetti and Dakdouk [19] and Bitansky and Canetti [7]. Let PO be a point obfuscator for $\mathsf{PF}(\mathcal{X})$ and let $t = t(k) > 0$ be a polynomial. Then an MBPF obfuscator MBPO for $\mathsf{MBPF}(\mathcal{X}, t)$ is constructed as in Fig. 5.

Based on the result of [19], Bitansky and Canetti [7] showed that if PO is $(t+1)$-composable, then the MBPF obfuscator MBPF constructed as in Fig. 5 is a $\mathtt{WVG}$ secure. By instantiating this conversion with the above mentioned point obfuscator, we obtain a $\mathtt{WVG}$ secure t-bit-output MBPF obfuscator under the $(t+1)$-SVDDH assumption.