Indistinguishability Obfuscation versus Point Obfuscation with Auxiliary Input

Christina Brzuska[1] Arno Mittelbach[2]

[1]Tel Aviv University, Israel
[2]Darmstadt University of Technology, Germany
brzuska@post.tau.ac.il arno.mittelbach@cased.de

Abstract. In a recent celebrated breakthrough, Garg et al. (FOCS 2013) gave the first candidate for so-called indistinguishability obfuscation (iO) thereby reviving the interest in obfuscation for a general purpose. Since then, iO has been used to advance numerous sub-areas of cryptography. While indistinguishability obfuscation is a general purpose obfuscation scheme, several obfuscators for specific functionalities have been considered. In particular, special attention has been given to the obfuscation of so-called *point functions* that return zero everywhere, except for a single point α. A strong variant is point obfuscation with auxiliary input (AIPO), which allows an adversary to learn some non-trivial auxiliary information about the obfuscated point α (Goldwasser, Tauman-Kalai; FOCS, 2005).

Multi-bit point functions are a strengthening of point functions, where on α, the point function returns a string β instead of 1. Multi-bit point functions with auxiliary input (MB-AIPO) have been constructed by Canetti and Bitansky (Crypto 2010) and have been used by Matsuda and Hanaoka (TCC 2014) to construct CCA-secure public-key encryption schemes and by and Bitansky and Paneth (TCC 2012) to construct three-round weak zero-knowledge protocols for NP.

In this paper we present both positive and negative results. We show that if indistinguishability obfuscation exists, then MB-AIPO does not. Towards this goal, we build on techniques by Brzuska, Farshim and Mittelbach (Crypto 2014) who use indistinguishability obfuscation as a means of attacking a large class of assumptions from the Universal Computational Extractor framework (Bellare et al; Crypto 2013). On the positive side we introduce a weak version of MB-AIPO which we deem to be outside the reach of our impossibility result. We prove that this weak version of MB-AIPO suffices to construct a public-key encryption scheme that is secure even if the adversary can learn an arbitrary leakage function of the secret key, as long as the secret key remains computationally hidden. Thereby, we strengthen a result by Canetti et al. (TCC 2010) that showed a similar connection in the symmetric-key setting.

Keywords. Indistinguishability obfuscation, differing-inputs obfuscation, point function obfuscation, multi-bit point function obfuscation, auxiliary input obfuscation, leakage resilient PKE

Contents

1 Introduction

The obfuscation of a program should hide its inner workings while preserving the functionality of the program. Inspired by heuristic code-obfuscation techniques [CTL97], obfuscation turned into a major research area of cryptography due to its manifold applications.

The formal definition of Virtual Black-Box Obfuscation (VBB) demands that an obfuscated program is as good as a black-box that provides the same input-output behaviour as the program. Since the seminal paper of Barak et al. [BGI+01, BGI+12], we know that this strong notion of obfuscation is generally not achievable. In particular, Barak et al. show that so-called *point functions* cannot be VBB-obfuscated when "paired" with a particularly chosen second function.

A (multi-output-bit-)point function $p_{\alpha,\beta}$ maps all strings to 0, except for a single point α which the function maps to the string β. The second function is a *test function* $\mathcal{T}_{\alpha,\beta}$ that takes as input a circuit C and tests whether $C(\alpha)$ is equal to β. Now, if an adversary is given access to two oracles that compute $p_{\alpha,\beta}$ and $\mathcal{T}_{\alpha',\beta'}$ then it cannot check whether the two functions "match", i.e., whether $(\alpha', \beta') = (\alpha, \beta)$. In turn, when given a circuit C that computes $p_{\alpha,\beta}$, the adversary can run $\mathcal{T}_{\alpha,\beta}$ on C and simply check whether $\mathcal{T}_{\alpha,\beta}(C)$ returns 1. Hence, the obfuscation of $p_{\alpha,\beta}$ and the obfuscation of $\mathcal{T}_{\alpha,\beta}$ leak more information than two oracles for $p_{\alpha,\beta}$ and $\mathcal{T}_{\alpha,\beta}$ establishing a counterexample for VBB obfuscation.

As Barak et al. [BGI+01, BGI+12] showed that general VBB-obfuscation is impossible, follow-up work did not aim for general obfuscation techniques but rather focused on obfuscating *specific* functionalities such as re-encryption [HRsV07] and encrypted signatures [Had10]. Perhaps surprisingly, a sequence of works also achieved positive results for VBB obfuscation of point functions [Can97, CMR98, Fis99, Wee05, CD08, CKVW10, BC10, BP12].

The seemingly contradiction is resolved by considering *auxiliary information* about the obfuscated circuit. Goldwasser and Tauman-Kalai introduced the concept of auxiliary information within the study of obfuscation [GK05]. With this concept we can rephrase the counterexample by Barak et al. [BGI+01] by considering the "test function" $\mathcal{T}_{\alpha,\beta}$ to be auxiliary information about the obfuscated point function $p_{\alpha,\beta}$. Thus, one way of stating their result is to say that no VBB obfuscation for point functions exist, when arbitrary auxiliary information is allowed.

The works of Wee [Wee05], Hofheinz et al. [HMLS07], and Canetti et al. [CD08, CKVW10] achieve (variants of) VBB obfuscation for point functions. In their respective models for VBB, they disallow the leakage of auxiliary information about (α, β) and hence, the impossibility result [BGI+01] and variants thereof [HMLS07] do not apply to their respective settings.

In turn, Canetti [Can97], Dodis, Tauman-Kalai and Lovett [DKL09], Bitansky and Canetti [BC10] as well as Bitansky and Paneth [BP12] allow for auxiliary information about the obfuscated point to leak, but use a weaker notion of obfuscation and thereby avoid the impossibility result. Bitansky and Paneth provide a clean treatment of auxiliary inputs and introduce the notion of *point obfuscation with auxiliary input* secure against unpredictable distributions (AIPO) which we adopt in this paper.

Point functions vs. point functions with multi-bit output. When considering point function obfuscation, we need to make a clear distinction between plain point functions such as p_x which map every input to 0 except for the single input x that is mapped to 1 and point functions with multi-bit output (MBPF) such as $p_{x,m}$ where input x is mapped to string m. Although very similar, obfuscation schemes for MBPFs are seemingly harder to construct than obfuscation schemes for plain point functions. Indeed, Canetti and Dakdouk initiated the study of obfuscation for MBPFs and showed that such obfuscation schemes are closely related to *composable* obfuscation schemes for plain point functions [CD08]. They show that obfuscators for MBPFs exist if, and only if, composable obfuscators for plain point functions exist. Moreover, they show that composability is a

non-trivial property. Both of these results carry over to obfuscation in the presence of hard-to-invert auxiliary information.

INDISTINGUISHABILITY OBFUSCATION. Simultaneously to constructing task-specific obfuscation schemes, the quest for general obfuscators continued, and in a celebrated breakthrough [GGH+13], Garg, Gentry, Halevi, Raykova, Sahai and Waters presented a candidate construction for indistinguishability obfuscation (iO). The notion of indistinguishability obfuscation is weaker than VBB-obfuscation and says intuitively that, for any two circuits that compute the same function, their obfuscations are indistinguishable. As Goldwasser and Rothblum [GR07] establish, this seemingly weak notion of obfuscation is actually the *best possible* notion of obfuscation. And indeed, the work by Garg et al. [GGH+13] inspired simultaneous breakthroughs for hard problems in several sub-areas of cryptography [SW13, BCP14, ABG+13, GGHR14, HSW14, BZ13, BST13] such as functional encryption, deniable encryption, two-round secure multi-party computation, full-domain hash, poly-many hardcore bits for any one-way function and more.

CONTRIBUTION. In this paper we give both positive and negative results. We show that the existence of indistinguishability obfuscation *contradicts* the positive results for multi-bit point function obfuscation in the presence of hard-to-invert auxiliary information (MB-AIPO) [BP12, MH14]. That is, if indistinguishability obfuscation exists, then MB-AIPO does not exist and the assumptions in [BP12, MH14] are false. Or, equivalently, if MB-AIPO exists, then indistinguishability obfuscation does not exist and all candidate assumptions are false [GGH+13, PST13, GLSW14]. However, given the current advancements in the understanding of indistinguishability obfuscation—for example, Gentry et al. [GLSW14] show in a very recent work that iO can be based on the Multilinear Subgroup Elimination Assumption thereby giving the first construction based on an instance-independent assumption—we consider the existence of iO to be more likely.

In summary, we derive the following negative results.

Theorem [informal]. *If indistinguishability obfuscation exists, then composable AIPO, VBB multi-bit point obfuscation secure with auxiliary inputs and average-case MB-AIPO do not exist.*

To show our impossibility result, we construct an obfuscation of the test function $\mathcal{T}_{\alpha,\beta}$ that Barak et al. used to establish their impossibility result for VBB obfuscation [BGI+01, BGI+12]. The hardness resides in proving that the obfuscation of $\mathcal{T}_{\alpha,\beta}$ indeed hides all partial information about α and β. Towards this goal, we build on techniques developed by Brzuska, Farshim and Mittelbach [BFM14] who show a similar 1-out-of-2 result, namely that indistinguishability and a large class of assumptions of the Universal Computational Extractor Framework (UCE) [BHK13] are mutually exclusive. Namely, we obfuscate the test function via indistinguishability obfuscation and prove that it is indistinguishable from an obfuscation of the zero circuit, the circuit that returns 0 on all inputs. As the zero circuit does not contain any information about α and β, we argue that also an obfuscation of the test function $\mathcal{T}_{\alpha,\beta}$ hides α and β computationally.

Intriguingly, it seems that our negative results do not carry over to the setting of obfuscating plain point functions in the presence of auxiliary information, that is, to plain AIPO (assuming they are not composable). As an analogy, consider the impossibility result by Barak et al. [BGI+01, BGI+12]. Also here, it seems crucial that the point function $p_{\alpha,\beta}$ has a multi-bit output β. Imagine that $\mathcal{T}_{\alpha}$ takes the circuit C as input and returns 1 if and only if $C(\alpha) = 1$, then an adversary could perform binary search and recover α, even when only given access to $\mathcal{T}_{\alpha}$ and p_{α} as oracles. Therefore, the impossibility does not carry through to standard point functions.

On the positive side we show ways to work around our impossibility result. Firstly, note that Canetti et al. [CKVW10] introduce weaker versions of MB-AIPO that are not affected by our negative results. In particular, they use these weaker notions to build a symmetric-key encryption scheme that is secure in the presence of hard to invert leakage about the key. We strengthen their result insofar, as we present a notion that lies between their weaker versions of MB-AIPO and full MB-AIPO.

Our weak notion of MB-AIPO requires that the auxiliary information computationally hides the point x even when given the corresponding point value m for some multi-bit point function $p_{x,m}$. This restriction seems sufficient to bypass our impossibility result and, intriguingly, assuming AIPO and iO, we give a construction that achieves this notion of weak MB-AIPO. Finally, we use our weak MB-AIPO construction to build a public-key encryption scheme which is leakage resilient in the presence of hard-to-invert leakage of the key.

Conclusion and future work. We showed that MB-AIPO as used in [BP12, MH14] and iO are mutually exclusive. It remains to investigate whether the positive results in [BP12, MH14] can be salvaged through weaker notions of MB-AIPO or, perhaps, when combining AIPO and iO in a similar way as we did in Section 4. At a first glance, our weakened notion of MB-AIPO does not seem to suffice for the applications in [BP12, MH14], but it remains to study whether other weak variants of MB-AIPO can be used.

On the other hand, one might also ask whether our negative result can be extended to showing that AIPO and iO are mutually exclusive. Currently, we do not know whether this is possible, but such a result seems to require different techniques than the ones we use. Our result implies that differing-inputs obfuscation (diO) and MB-AIPO are mutually exclusive. Perhaps, using different techniques, one might be able to show that diO and AIPO are mutually exclusive, for example, by showing that we can instantiate the special-purpose obfuscator by Garg et al. [GGHW13] using AIPO.

We hope that our work sparks further interest in studying the connections between iO/diO on the one hand and notions of (multi-bit) point obfuscation on the other hand.

2 Preliminaries

Notation. We denote by $\lambda \in \mathbb{N}$ the security parameter, which all algorithms get implicitly and in unary representation 1^λ. By $\{0,1\}^\ell$ we denote the set of all bit-strings of length ℓ, and by $\{0,1\}^*$ the set of all bit-strings of finite length. The length of x is denoted by $|x|$. If $x, y \in \{0,1\}^*$ are two bit strings of the same length, then we denote their inner product over $\mathbb{GF}(2)$ by $\langle x, y\rangle$. For a finite set X, we denote the action of sampling x uniformly at random from X by $x \leftarrow\!\!\$\ X$, and denote the cardinality of X by $|X|$. Algorithms are assumed to be randomized, unless otherwise stated. We call a algorithm efficient or PPT if it runs in time polynomial in the security parameter. All algorithms may be randomized, unless explicitly stated differently. If $\mathcal{A}$ is randomized then by $y \leftarrow \mathcal{A}(x;r)$ we denote that $\mathcal{A}$ is run on input x and with random coins r and produced output y. If no randomness is specified, then we assume that $\mathcal{A}$ is run with freshly sampled uniform random coins, and write this as $y \leftarrow\!\!\$\ \mathcal{A}(x)$. We often refer to algorithms, or tuples of algorithms, as adversaries. If E is an event then we denote by $\Pr[E]$ its probability and if X is a random variable, we denote its expectation by $\mathbb{E}[X]$. We say a function $\mathrm{negl}(\lambda)$ is negligible if $\mathrm{negl}(\lambda) \in \lambda^{-\omega(1)}$. We say a function poly is polynomial if $\mathrm{poly} \in \lambda^{\mathcal{O}(1)}$.

OBFUSCATION. Obfuscation has been extensively used within cryptography and it comes in many different flavors. In the following section we present the various definitions that we use in this paper. We start by recalling the strongest definition of virtual black-box (VBB) obfuscation with auxiliary inputs due to [BGI+01, GK05, BGI+12].

Definition 2.1 (Worst-case obfuscator with auxiliary input)**.** *A* PPT $\mathcal{O}$ *is a worst-case obfuscator with auxiliary input for an ensemble* $\mathcal{C} = \{\mathcal{C}_\lambda\}_{\lambda\in\mathbb{N}}$ *of families of poly-size circuits if it satisfies:*

- **Functionality.** *For any* $\lambda \in \mathbb{N}$ *and* $C \in \mathcal{C}_\lambda$, $\mathcal{O}(C)$ *is a circuit which computes the same function as* C*.*
- **Polynomial slowdown.** *For any* $\lambda \in \mathbb{N}$ *and* $C \in \mathcal{C}_\lambda$, $|\mathcal{O}(C)| \leq \mathrm{poly}(|C|)$*.*
- **Virtual black-box.** *For any* PPT *adversary* $\mathcal{A}$ *there is a* PPT *simulator* Sim *such that for all sufficiently large* $\lambda \in \mathbb{N}, C \in \mathcal{C}_\lambda$ *and* $z \in \{0,1\}^{\mathsf{poly}(\lambda)}$*:*

$$\left|\Pr[\mathcal{A}(z,\mathcal{O}(C))=1] - \Pr\left[\mathsf{Sim}^C(z,1^{|C|})=1\right]\right| \leq \mathrm{negl}(\lambda)$$

where the probability is taken over the coins of $\mathcal{A}$*,* Sim *and* $\mathcal{O}$*.*

INDISTINGUISHABILITY OBFUSCATION. While VBB obfuscation as defined above provably does not exist [BGI+01] for all circuits, weaker notions such as *indistinguishability obfuscation* may well do. VBB obfuscation requires that for any PPT adversary given the code of some functionality (and some auxiliary input) there exists a PPT simulator that given only black-box access to the functionality (and as input the same auxiliary input) produces a computationally indistinguishable distribution. An indistinguishability obfuscation (iO) scheme, on the other hand, only ensures that the obfuscations of any two functionally equivalent circuits are computationally indistinguishable. Indistinguishability obfuscation was originally proposed by Barak et al. [BGI+01] as a potential weakening of virtual-black-box obfuscation. We recall the definition from [GGH+13].

Definition 2.2. *A* PPT *algorithm* iO *is called an* indistinguishability obfuscator *for a circuit class* $\{\mathcal{C}_\lambda\}_{\lambda\in\mathbb{N}}$ *if the following conditions are satisfied:*

- **Correctness.** *For all security parameters* $\lambda \in \mathbb{N}$*, for all* $C \in \mathcal{C}_\lambda$*, and for all inputs* x *we have that*

$$\Pr\left[C'(x) = C(x) : C' \leftarrow_{\$} \mathsf{iO}(1^\lambda, C)\right] = 1 \ .$$

- **Security.** *For any* PPT *distinguisher* $\mathcal{D}$*, for all pairs of circuits* $C_0, C_1 \in \mathcal{C}_\lambda$ *such that* $C_0(x) = C_1(x)$ *on all inputs* x *the following distinguishing advantage is negligible:*

$$\left|\Pr\left[\mathcal{D}(1^\lambda, \mathsf{iO}(1^\lambda, C_1)) = 1\right] - \Pr\left[\mathcal{D}(1^\lambda, \mathsf{iO}(1^\lambda, C_0)) = 1\right]\right| \leq \mathrm{negl}(\lambda) \ .$$

DIFFERING-INPUTS OBFUSCATION. Differing-inputs obfuscation is closely related to indistinguishability obfuscation and also goes back to the seminal paper of Barak et al. [BGI+01, BGI+12]. While indistinguishability obfuscation requires circuits to be identical on all inputs, differing-inputs obfuscation intuitively says that if a distinguisher can tell apart two obfuscated circuits then one can efficiently extract a value on which the circuits differ. We here follow the definition of Ananth et al. [ABG+13] and Boyle et al. [BCP14] and first define the notion of *differing-inputs circuits* which in turn are then used to define differing-inputs obfuscation.

Definition 2.3 (Differing-Inputs Circuits)**.** *A circuit family $\{\mathcal{C}_\lambda\}_{\lambda\in\mathbb{N}}$ together with a sample algorithm $(C_0, C_1, z) \leftarrow\!\!\$\ \mathsf{Sam}(1^\lambda)$ which samples $C_0, C_1 \in \mathcal{C}_\lambda$ is said to be a* differing-inputs *family if for all* PPT *algorithms $\mathcal{A}$ there is a negligible function* negl *such that:*

$$\Pr\Big[C_0(x) \neq C_1(x) : (C_0, C_1, z) \leftarrow\!\!\$\ \mathsf{Sam}(1^\lambda), x \leftarrow\!\!\$\ \mathcal{A}(1^\lambda, C_0, C_1, z)\Big] \leq \mathrm{negl}(\lambda)$$

Definition 2.4 (Differing-Inputs Obfuscation)**.** *A* PPT *algorithm* diO *is a* differing-inputs obfuscator *for a differing-inputs family $(\{\mathcal{C}_\lambda\}, \mathsf{Sam})$ if the following holds:*

- **Correctness.** *For all security parameters $\lambda \in \mathbb{N}$, for all $C \in \mathcal{C}_\lambda$, and for all inputs x we have that*
$$\Pr\Big[C'(x) = C(x) : C' \leftarrow\!\!\$\ \mathsf{diO}(1^\lambda, C)\Big] = 1\ .$$
- **Security.** *For any* PPT *distinguisher $\mathcal{D}$, for any $(C_0, C_1, z) \leftarrow\!\!\$\ \mathsf{Sam}(1^\lambda)$ the following distinguishing advantage is negligible:*
$$\Big|\Pr\Big[\mathcal{D}(1^\lambda, \mathsf{diO}(1^\lambda, C_1), z) = 1\Big] - \Pr\Big[\mathcal{D}(1^\lambda, \mathsf{diO}(1^\lambda, C_0), z) = 1\Big]\Big| \leq \mathrm{negl}(\lambda)\ .$$

Boyle, Chung and Pass [BCP14] show that any general indistinguishability obfuscator is also an differing-inputs obfuscator for certain classes of circuits. That is, any indistinguishability obfuscator for all circuits in $\mathcal{P}/\mathrm{poly}$ is a also a differing-inputs obfuscator for circuits that differ on at most polynomially many inputs. We recall their Theroem:

Theorem 2.5 ([BCP14])**.** *Let* iO *be an indistinguishability obfuscator for $\mathcal{P}/\mathrm{poly}$. Let $(\{\mathcal{C}_\lambda\}, \mathsf{Sam})$ be a differing-inputs family for which there exists a polynomial $d : \mathbb{N} \to \mathbb{N}$, such that for all $\lambda \in \mathbb{N}$ and all pairs $C_0, C_1 \in \mathcal{C}_\lambda$ it holds that $|\{x : C_0(x) \neq C_1(x)\}| \leq d(\lambda)$. Then* iO *is a differing-inputs obfuscator for $(\{\mathcal{C}_\lambda\}, \mathsf{Sam})$.*

Point obfuscation. Besides the general purpose indistinguishability obfuscator we consider obfuscators for the specific class of so-called point functions. A point function p_x for some value $x \in \{0,1\}^*$ is defined as

$$p_x(s) := \begin{cases} 1 & \text{if } s = x \\ \perp & \text{o/w} \end{cases}$$

In this paper, we consider a variant of point function obfuscators under auxiliary input which was first formalized by Canetti [Can97]. We here give the definition from [BP12]. The first definition formalizes unpredictable distributions which are in turn used to define obfuscators for point functions.

Definition 2.6 (Unpredictable Distribution)**.** *A distribution ensemble $\mathcal{D} = \{D_\lambda = (Z_\lambda, X_\lambda)\}_{\lambda\in\mathbb{N}}$, on pairs of strings is unpredictable if no poly-size circuit family can predict X_λ from Z_λ. That is, for every poly-size circuit family $\{C_\lambda\}_{\lambda\in\mathbb{N}}$ and for all large enough λ:*

$$\Pr_{(z,x) \leftarrow\!\!\$\ D_n}[C_\lambda(z) = x] \leq \mathrm{negl}(\lambda)$$

Definition 2.7 (Auxiliary input point obfuscation for unpredictable distributions (AIPO))**.** *A* PPT *algorithm* AIPO *is a* point obfuscator for unpredictable distributions *if it satisfies the functionality and polynomial slowdown requirements as in Definition 2.1, and the following secrecy property: for any unpredictable distribution $\mathcal{D} = \{D_\lambda = (Z_\lambda, X_\lambda)\}_{\lambda\in\mathbb{N}}$ over $\{0,1\}^{\mathrm{poly}(\lambda)} \times \{0,1\}^\lambda$ it holds for any* PPT *algorithms $\mathcal{A}$ that there exists a negligible function* negl *such that:*

$$\Big|\Pr_{(z,x) \leftarrow\!\!\$\ D_\lambda}\Big[\mathcal{A}(1^\lambda, \mathsf{AIPO}(x), z) = 1\Big] - \Pr_{z \leftarrow\!\!\$\ Z_\lambda, u \leftarrow\!\!\$\ \{0,1\}^\lambda}\Big[\mathcal{A}(1^\lambda, \mathsf{AIPO}(u), z) = 1\Big]\Big| \leq \mathrm{negl}(\lambda)$$

Obfuscation for point functions with multi-bit output. While point functions only return a single bit, a point function with multi-bit output (MBPF) $p_{x,m}$ for values $x, m \in \{0,1\}^*$ is defined as

$$p_{x,m}(s) := \begin{cases} m & \text{if } s = x \\ \perp & \text{o/w} \end{cases}$$

For an MBPF $p_{x,m}$ we call x the point address and m the point value. Similar to AIPO we define MB-AIPO via an unpredictable distribution where the distribution outputs a tuple (x, m) (defining a point function $p_{x,m}$) together with auxiliary information z. We require that it should be computationally infeasible to recover the point address x given auxiliary information z. Thus, in the MBPF setting we define the unpredictable distribution as $\mathcal{D} = \{D_\lambda = (Z_\lambda, X_\lambda, M_\lambda)\}_{\lambda\in\mathbb{N}}$ but still require that the point address (aka., x) remains hidden given the auxiliary input. From an MB-AIPO obfuscator we now require that the obfuscation of $p_{x,m}$ is indistinguishable from an obfuscation with a changed point address m' where m' is chosen uniformly at random. Intuitively this captures that the obfuscation does not reveal any information about the point value.

Definition 2.8 (Auxiliary input point obfuscation for unpredictable distributions (MB-AIPO))**.** *A* PPT *algorithm* MB-AIPO *is a* multi-bit point obfuscator for unpredictable distributions *if it satisfies the functionality and polynomial slowdown requirements as in Definition 2.1, and the following secrecy property: for any unpredictable distribution $\mathcal{D} = \{D_\lambda = (Z_\lambda, X_\lambda, M_\lambda)\}_{\lambda\in\mathbb{N}}$ over $\{0,1\}^{\mathrm{poly}(\lambda)} \times \{0,1\}^\lambda \times \{0,1\}^\lambda$ it holds for any* PPT *algorithms $\mathcal{A}$ that*

$$\Pr_{(z,x,m)\leftarrow\$\ D_\lambda}\Big[\mathcal{A}(1^\lambda, \mathsf{MB\text{-}AIPO}(x,m), z) = 1\Big] - \Pr_{\substack{(z,x,m)\leftarrow\$\ D_\lambda,\\ m'\leftarrow\$\{0,1\}^\lambda}}\Big[\mathcal{A}(1^\lambda, \mathsf{MB\text{-}AIPO}(x,m'), z) = 1\Big]$$

is negligible in λ.

Remark. Similarly, to the presented definition we could define MB-AIPO by requiring that the obfuscation of point function $p_{x,m}$ is indistinguishable from an obfuscation of $p_{x',m'}$. That is instead of only adapting the point value we could conceive a notion in which the honest obfuscation should be indistinguishable from one where both the point address and the point value are chosen uniformly at random. However, we show in Section 3 that, assuming indistinguishability obfuscation exists, neither of the above definitions can be met.

Average-case point obfuscation and statistical unpredictability. The notions for point obfuscation as defined above can be regarded as worst-case notions. In many applications an average-case notion where the point address is chosen uniformly at random could be sufficient. Indeed Matsuda and Hanaoka [MH14] recently presented constructions of CCA-secure public-key encryption schemes based on an average-case notion of MB-AIPO where the point address is sampled uniformly at random. MH14 denote the resulting average case MB-AIPO notion by AIND-δ-cPUAI.

A second avenue to weaken the security requirements of point obfuscators is to require that the auxiliary input needs to hide the point address statistically. We call unpredictable distributions for which this is the case *statistically unpredictable*.

3 IO implies the impossibility of MB-AIPO

In the following we present our negative result, namely that indistinguishability obfuscation and multi-bit point function obfuscation in the presence of auxiliary information (MB-AIPO) are mutually exclusive. We discuss implications of our result in Section 3.2.

3.1 IO and MB-AIPO are mutually exclusive

Multi-bit point obfuscation with auxiliary inputs is a powerful primitive and has been used to construct CCA-secure encryption schemes [MH14] and to circumvent black-box impossibility results for three-round weak zero-knowledge protocols for $\mathcal{NP}$ [BP12]. Our following result intuitively says that, if indistinguishability obfuscation exists, then MB-AIPOs (as defined in Definition 2.8) cannot exist. The result remains valid even if we consider average case MB-AIPOs (where point address x is chosen uniformly at random). Technically our result builds on techniques used by Brzuska, Farshim and Mittelbach (BFM; [BFM14]). BFM show a similar "one-out-of-two" result, namely that if indistinguishability obfuscation exists, then certain kinds of UCE-secure hash functions (a hash function security notion recently introduced in [BHK13]) cannot exist [BFM14]. BFM use an indistinguishability obfuscation of a specially constructed circuit which can be shown to hide (computationally) its inner workings but which is sufficient to later distinguish within the security game. UCEs are, in some sense, similar to AIPOs as the admissibility of adversaries is decided in a setting which is not quite the same as the setting into which the adversary is placed in the security game. For AIPOs, an adversary is admissible if its first part implements an unpredictable distribution. In the actual security game, the second adversary, however, does not only get as output the (unpredictable) leakage of the first adversary but some additional information provided by the game. The idea is now to output a circuit which, given this additional information will break the security property but which without it leaks no information whatsoever. Showing that an indistinguishability obfuscation hides a certain value is usually the crux in proofs involving iO. For this, we construct a new technique which may be of independent interest and is given as Lemma 3.2.

Theorem 3.1. *If indistinguishability obfuscation exists, then average-case obfuscation for multi-bit point functions secure under auxiliary input (MB-AIPO) does not exist.*

To prove Theorem 3.1 we use indistinguishability obfuscation to construct an unpredictable distribution $\mathcal{B}_1$ together with an adversary $\mathcal{B}_2$ that, given leakage from the unpredictable distribution can distinguish between point obfuscations from the unpredictable distribution and point obfuscations from the uniform distribution.

We first give the unpredictable distribution $\mathcal{B}_1$ which takes as input the security parameter 1^λ and outputs two values x, z together with some leakage L. Here leakage L will be the indistinguishability obfuscation of a predicate circuit that takes as input a description of a function f, evaluates the function on a hard-coded value x, runs the result through a pseudo-random generator PRG and finally compares this result with some hard-coded value y. That is, we consider the circuit

$$C[x, y] := \mathsf{iO}\left(\mathsf{PRG}(\mathsf{uT}(\cdot, x)) = y\right),$$

where uT denotes a universal Turing machine.

To formally define the unpredictable distribution, let n and m be two polynomials and let $\mathsf{PRG} : \{0,1\}^{n(\lambda)} \to \{0,1\}^{2n(\lambda)}$ be a pseudo-random generator with stretch 2. Note that we do not need to additionally assume the existence of PRGs as AIPOs (and in particular MB-AIPOs) already imply one-way functions.[1] Let, furthermore, $\mathsf{uT}(\cdot, x)$ be a universal Turing machine that on input a description of a function f outputs $f(x)$. We define the unpredictable distribution as $((x, z), L)$

[1] Canetti et al. [CKVW10] show that multi-bit point function obfuscation is tightly related to symmetric encryption and that MB-AIPO implies the existence of (leakage-resilient) IND-CPA symmetric encryption schemes.

computed as:

$$
\begin{aligned}
&z \leftarrow_{\$} \{0,1\}^{n(\lambda)} \\
&y \leftarrow \mathsf{PRG}(z) \\
&x \leftarrow_{\$} \{0,1\}^{m(\lambda)} \\
&L \leftarrow_{\$} \mathsf{iO}\left(\mathsf{PRG}(\mathsf{uT}(\cdot, x)) = y\right) \\
&\textbf{output: } ((x,z), L)
\end{aligned}
$$

We now present the adversary $\mathcal{B}_2$ that, given the leakage from $\mathcal{B}_1$, breaks the security of the multi-bit point obfuscator. We will then argue that $\mathcal{B}_1$, indeed, implements an unpredictable distribution. Adversary $\mathcal{B}_2$ gets values p and L as input, where p is either a point obfuscation of $I_{x,z}$ sampled according to $\mathcal{B}_1$ or an obfuscation for $I_{u,v}$ for uniformly random values u, v. Adversary $\mathcal{B}_2$ computes $L(p)$ and outputs the result. If p is an obfuscation of $I_{x,z}$, then $\mathcal{B}_2$ computes the predicate function

$$\mathsf{PRG}(I_{x,z}(x)) = y$$

where y was computed as $\mathsf{PRG}(z)$. Thus, it will always output 1. If, however, p is an obfuscation of $I_{u,v}$, then with overwhelming probability over the choice of v and u, $\mathcal{B}_2$ returns 0. It follows that $(\mathcal{B}_1, \mathcal{B}_2)$ break the security of the obfuscator with overwhelming probability. We now prove that $(\mathcal{B}_1, \mathcal{B}_2)$ is also a valid pair of adversaries, that is, that $\mathcal{B}_1$ is an unpredictable distribution. In the following lemma, we show that, under the assumption of indistinguishability obfuscation, the leakage computed by $\mathcal{B}_1$ is indistinguishable from an obfuscated zero circuit, the circuit that returns 0 on all inputs. As the zero circuit does not leak any information about y, the leakage is unpredicable.

Lemma 3.2. *Let n, m be two polynomials. For $x \in \{0,1\}^{\lambda}$, let $\mathsf{uT}(\cdot, x)$ be a universal Turing machine that on input a description of a function $f \in \{0,1\}^{m(\lambda)}$ outputs $f(x)$.*

If $\mathsf{PRG} : \{0,1\}^{n(\lambda)} \to \{0,1\}^{2n(\lambda)}$ is a pseudo-random generator and iO is a secure indistinguishability obfuscator for all circuits in $\mathcal{P}/\mathrm{poly}$, then for all efficient PPT distinguishers Dist there exists a negligible function negl such that

$$\left|\Pr\left[\mathsf{Dist}\left(1^{\lambda}, \mathsf{iO}\left(\mathsf{PRG}(\mathsf{uT}(.,x)) = \mathsf{PRG}(y)\right)\right) = 1\right] - \Pr\left[\mathsf{Dist}(1^{\lambda}, \mathsf{iO}(\mathbf{0})) = 1\right]\right| \leq \mathrm{negl}(\lambda)$$

where the first probability is over the random choice of x and y and the coins of iO and Dist and the second probability is over the coins of iO and Dist.

Proof. We bound the distinguishing probability in Lemma 3.2 with the security of the PRG and the indistinguishability obfuscator iO. We consider the following hybrids (which are also depicted in Figure 1):

Game_1**:** The game chooses a random value z and computes $y \leftarrow \mathsf{PRG}(z)$. It constructs the predicate circuit $C_{x,y} \leftarrow (\mathsf{PRG}(\mathsf{uT}(.,x)) = \mathsf{PRG}(y))$ and an obfuscation $\tilde{C} = \leftarrow_{\$} \mathsf{iO}(C_{x,y})$. It then calls distinguisher Dist on input $\tilde{C}$ and outputs whatever Dist outputs.

Game_2**:** The game chooses a uniformly random value y. It constructs the predicate circuit $C_{x,y} \leftarrow (\mathsf{PRG}(\mathsf{uT}(.,x)) = y)$ and an obfuscation $\tilde{C} = \leftarrow_{\$} \mathsf{iO}(C_{x,y})$. It then calls distinguisher Dist on input $\tilde{C}$ and outputs whatever Dist outputs.

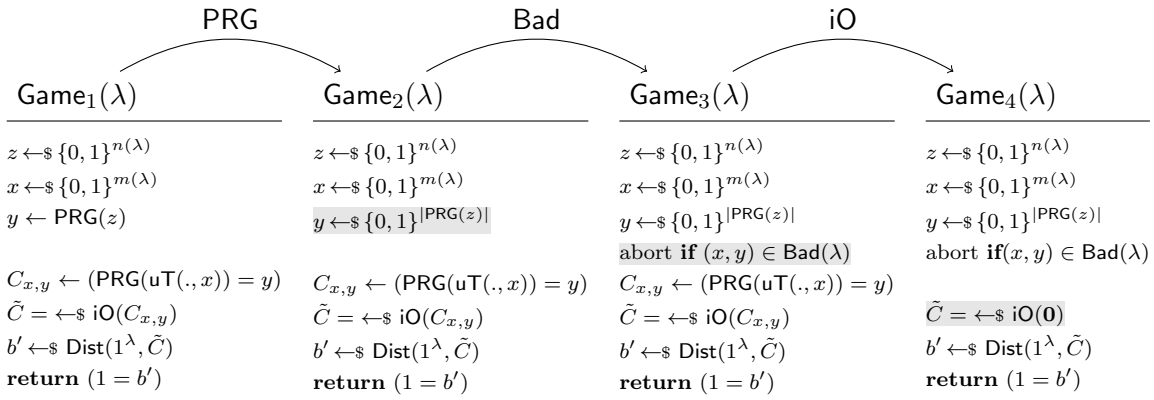

Figure 1: The hybrids for the proof of Claim 3.2. We have highlighted the changes between the games with a light-grey background.

Game$_3$: As before, except that the game terminates if there there exists a value $f \in \{0,1\}^{m(\lambda)}$ such that $C_{x,y}(f) = 1$. We denote this event by $(x,y) \in \mathsf{Bad}(\lambda)$ if for values (x,y) such a value f exists.

Game$_4$: As before, except that now an obfuscation of the constant zero-circuit $\tilde{C} \leftarrow\$ \mathsf{iO}(\mathbf{0})$ is constructed.

We bound the difference between Game_1 and Game_2 by the security of the PRG. Game_2 and Game_3 are, by the fundamental lemma of the game-playing technique [BR06], identical until event $\mathsf{Bad}(\lambda)$ occurs. We now prove that $\mathsf{Bad}(\lambda)$ only happens with negligible probability. For a uniformly random pair of values (x,y) we have that the probability that there exists a function f such that $\mathsf{PRG}(\mathsf{uT}(f,x)) = y$ is upper bounded by the probability that y is in the image of the PRG. As the PRG has a stretch of 2, the probability that a random y is in the image of the PRG is upper bounded by $2^{-n(\lambda)}$.

Finally, we bound the difference between Game_3 and Game_4 by the security of the indistinguishability obfuscator iO by noting that if $(x,y) \notin \mathsf{Bad}(\lambda)$ then circuit $C_{x,y}$ encodes the constant zero circuit. We now explain this formally.

Firstly, let us externalize some of the variables that the games use and introduce a unified notation for Game_3 and Game_4. For $i \in \{3,4\}$, let $\mathsf{Game}_i[x,y](\lambda)$ be equal to the game $\mathsf{Game}_i(\lambda)$ where the game chooses values x and y. We define $\mathcal{A}[x,y](C)$ to be an adversary against the indistinguishability obfuscator iO that gets a circuit C as input, where C is either an obfuscation of circuit $C_{x,y}$ or an obfuscation of the constant zero circuit $\mathbf{0}$. Adversary $\mathcal{A}[x,y](C)$ runs distinguisher D on input $(1^\lambda, C)$ and outputs whatever D outputs.

If $C = C_{x,y}$ then adversary $\mathcal{A}[x,y](C)$ perfectly simulates game $\mathsf{Game}_3[x,y](\lambda)$ and if $C = \mathbf{0}$ then the adversary simulates $\mathsf{Game}_4[x,y](\lambda)$. Thus, we can rewrite the difference between the game's distributions

$$\Pr[\mathsf{Game}_3(\lambda)] - \Pr[\mathsf{Game}_4(\lambda)]$$

as

$$
\begin{aligned}
&= \mathbb{E}_{x,y}\Big[\Pr[\mathsf{Game}_3[x,y](\lambda)]\Big] - \mathbb{E}_{x,y}\Big[\Pr[\mathsf{Game}_4[x,y](\lambda)]\Big] \\
&= \mathbb{E}_{x,y}\Big[\Pr[\mathsf{Game}_3[x,y](\lambda)] - \Pr[\mathsf{Game}_4[x,y](\lambda)]\Big] \\
&= \mathbb{E}_{x,y}\Big[\Pr\Big[\mathcal{A}[x,y](1^\lambda, \mathsf{iO}(C_{x,y})) = 1\Big] - \Pr\Big[\mathcal{A}[x,y](1^\lambda, \mathsf{iO}(\mathbf{0})) = 1\Big]\Big] \\
&= \mathbb{E}_{x,y}\Big[\mathsf{Adv}^{\mathsf{io}}_{\mathsf{iO},\mathcal{A}[x,y],C_{x,y},\mathbf{0}}(\lambda)\Big] \\
&\leq \max_{x,y} \mathsf{Adv}^{\mathsf{io}}_{\mathsf{iO},\mathcal{A}[x,y],C_{x,y},\mathbf{0}}(\lambda)
\end{aligned}
$$

By the security of the indistinguishability obfuscator, the advantage of any efficient adversary is negligible and, hence, also $\max_{x,y} \mathsf{Adv}^{\mathsf{io}}_{\mathsf{iO},\mathcal{A}[x,y],C_{x,y},\mathbf{0}}(\lambda)$ is negligible. □

3.2 Implications

Average case MB-AIPO is a relaxed notion of virtual-black-box point obfuscation in the presence of auxiliary input and in particular implied by it [MH14]. Consequently our impossibility result also shows that VBB obfuscation of multi-bit point functions secure in the presence of auxiliary input cannot exist if indistinguishability obfuscation exist:

Corollary 3.3. *If indistinguishability obfuscation exists, then VBB multi-bit point obfuscation secure with auxiliary input does not exist.*

We note that VBB multi-bit point obfuscation is also often referred to as *Digital Lockers*.

Canetti and Dakdouk [CD08] study the composition of point function obfuscation and show that composable AIPO implies the existence of composable MB-AIPO. And hence, applying our result we get the following corollary.

Corollary 3.4. *If indistinguishability obfuscation exists, then composable AIPO does not exist.*

Several results have been based on the existence of MB-AIPO (or composable AIPO). Matsuda and Hanaoka give a CCA secure public-key encryption scheme based on MB-AIPO [MH14] and Bitansky and Paneth give a three-round three-round weak zero-knowledge protocol for $\mathcal{NP}$ based on composable AIPO [BP12]. In Section 4 we present a weakened notion of MB-AIPO that we deem to fall outside our impossibility result. At a first glance, however, this weaker notion seems not sufficient for the applications in [BP12, MH14], so it remains to study whether other weak variants of MB-AIPO could be used.

3.3 On circumventing our impossibility result

Matsuda and Hanaoka [MH14] present a CCA-secure PKE scheme that is based on MB-AIPO and, thus, ruled out by our impossibility result, if indistinguishability obfuscation exists. However, they also present a version of a CCA-secure PKE scheme based on the weaker assumption of MB-AIPO that is secure only with respect to statistically unpredictable distributions. Indeed, our techniques do not carry over to ruling our MB-AIPO for statistically unpredictable distributions, because the way in which we use indistinguishability obfuscation, inherently relies on computational security.

Switching to a statistical notion of security was also proposed for UCEs in order to salvage a large number of applications [BFM14, BHK13].

Moreover, Canetti et al. [CKVW10] present notions of MB-AIPO in the setting of computational unpredictability that are not affected by our impossibility result. In the following section, we present a variant of MB-AIPO that is stronger than theirs but weaker than the definition that we show to be impossible. Namely, we strengthen the assumption on unpredictable distributions to remain unpredictable even in case the point value m is given. We call this notion strong unpredictability and, indeed, give a construction based on AIPO and indistinguishability obfuscation. An analogours notion of unpredictability has recently been introduced by Brzuska and Mittelbach in the context of UCE security [BM14].

4 Weak MB-AIPO from iO and AIPO

In this section we show that, despite the negative results from the previous section, point obfuscation (AIPO) and indistinguishability obfuscation (iO) together make a powerful team. We begin showing that iO can be used to construct a point obfuscation scheme which can securely obfuscate point functions given as input the point function, rather than the point address. This is inherently different from all proposed point obfuscation schemes so far [Can97, CMR98, Fis99, Wee05, CD08, CKVW10, BC10, BP12] which always take the point address as input. We call this sort of obfuscator point-independent as the obfuscator works independent of the actual point (in fact it may be computationally infeasible for the obfuscator to recover the point from its input). Using a similar technique we will then construct a mild form of a multi-bit point function obfuscation scheme secure in the presence of auxiliary input. We call notion *weak MB-AIPO*. In Section 5 we will then use our construction of weak MB-AIPOs to construct a public-key encryption scheme that is leakage resilient with respect to any computationally hard-to-invert leakage of the secret-key.

4.1 Point-independent point function obfuscation

Our first construction is based on an observation. Goldwasser and Rothblum [GR07] introduce the notion of best-possible obfuscation. Intuitively an obfuscator is a best-possible obfuscator if an obfuscation leaks as little information about the original program as any functionally equivalent circuit. In the context of point functions this means that a best-possible obfuscator for point functions (or for more general function classes) on input a point function I_x would need to output an obfuscated point function which is as least as good an obfuscation as, for example, an AIPO obfuscator produces on input the point address x. Goldwasser and Rothblum show that if we consider PPT obfuscators, then the notions of best-possible obfuscation and indistinguishability obfuscation are equivalent [GR07]. Let us recall the definition of best-possible obfuscation

Definition 4.1 ([GR07]: Best-possible obfuscation)**.** *A* PPT $\mathcal{O}$ *is a best-possible obfuscator for an ensemble* $\mathcal{C} = \{\mathcal{C}_\lambda\}_{\lambda\in\mathbb{N}}$ *of families of poly-size circuits if it satisfies the preserving functionality and polynomial slowdown properties as in Definition 2.1, and also has the following property (instead of the virtual black-box property).*

Computational Best-Possible Obfuscation. *For any polynomial size learner* $\mathcal{L}$*, there exists a polynomial size simulator* Sim *such that for every large enough input length* λ*, for any two circuit* $C_1 \in \mathcal{C}_\lambda$ *and for any circuit* $C_2 \in \mathcal{C}_\lambda$ *that computes the same function as* C_1 *and such that* $|C_1| = |C_2|$ *it holds for any* PPT *adversary* $\mathcal{A}$ *that*

$$\left|\Pr\Big[\mathcal{A}(1^\lambda, \mathcal{L}(\mathcal{O}(C_1))) = 1\Big] - \Pr\Big[\mathcal{A}(1^\lambda, \mathsf{Sim}(C_1)) = 1\Big]\right| \leq \mathrm{negl}(\lambda)$$

Thus, if iO is an indistinguishability obfuscator for all circuits in $\mathcal{P}/\text{poly}$, then the mapping $C \mapsto \mathsf{iO}(C)$ yields a point-independent point function obfuscator since the indistinguishability obfuscation of point function I_x must not leak any more about x than what can be extracted from the "best possible" point-obfuscation scheme on input x. Furthermore, if AIPO exists, then our construction yields an point-independent obfuscation scheme that is secure in the presence of auxiliary inputs.

4.2 Weak MB-AIPO from AIPO and iO

In the following we give a relaxed definition of MB-AIPO and subsequently give a construction based on plain AIPO and indistinguishability obfuscation. We weaken the original MB-AIPO definition (Definition 2.8) by requiring that an obfuscation must only be secure for unpredictable distributions that hide the point address even given the the point value m. We call this notion of unpredictability *strong unpredictability* and note that an analogous notion has recently been introduced in the context of UCE security [BM14]. If we restrict adversaries to strong unpredictability we yield an MB-AIPO notion that we call *weak MB-AIPO*.

Definition 4.2 (Strongly Unpredictable Distribution)**.** *We say that a distribution ensemble $\mathcal{D} = \{D_\lambda = (Z_\lambda, X_\lambda, M_\lambda)\}_{\lambda\in\mathbb{N}}$, on triples of strings is strongly unpredictable if no poly-size circuit family can predict X_λ from (Z_λ, M_λ). That is, for every poly-size circuit family $\{C_\lambda\}_{\lambda\in\mathbb{N}}$ and for all large enough λ:*

$$\Pr_{(z,x,m)\leftarrow\$\ D_n}[C_\lambda(z,m) = x] \leq \text{negl}(\lambda)$$

Definition 4.3 (Weak MB-AIPO)**.** *A* PPT *algorithm* AIPO *is a* weak *multi-bit point obfuscator if it is a MB-AIPO for strongly unpredictable distributions.*

Next, we present our construction of a weak MB-AIPO scheme. The idea will be to use a plain AIPO (for the point address x). We then construct a circuit which evaluates the AIPO and outputs m if, and only if, this evaluation returns 1. The construction will be the indistinguishability obfuscation of that circuit.

Construction 4.4. *Let* AIPO *be a secure AIPO and* iO *be a secure indistinguishability obfuscator for all circuits in* $\mathcal{P}/\text{poly}$*. We construct a weak MB-AIPO obfuscator* $\mathsf{MB\text{-}AIPO}$ *as follows. On input a point address x and value m* $\mathsf{MB\text{-}AIPO}$ *constructs a point obfuscation $p_x \leftarrow\$\ \mathsf{AIPO}(x)$. It then constructs the following circuit*

$$C[p_x, m](x^*) := \textbf{\textit{if}}\ (p_x(x^*) = 1)\ \textbf{\textit{then return}}\ m\ \textbf{\textit{else return}}\ \perp$$

and outputs an indistinguishability obfuscation of $C[p_x, m]$.

Proposition 4.5. *If AIPO exists and if* iO *is a secure indistinguishability obfuscator for all circuits in* $\mathcal{P}/\text{poly}$ *then the above construction is a weak MB-AIPO.*

Proof. Assume Proposition 4.5 does not hold. Then there exists adversary $(\mathcal{B}_1, \mathcal{B}_2)$ against the weak MB-AIPO property of $\mathsf{MB\text{-}AIPO}$ where $\mathcal{B}_1$ implements an unpredictable distribution, that is, on input the security parameter it outputs a point function description (x, m) together with some auxiliary information z. We will prove that both circuits $\mathsf{MB\text{-}AIPO}(x, m)$ as well as $\mathsf{MB\text{-}AIPO}(x, m')$ (for some uniformly random message m') are differing-inptus (given z) from the constant zero circuit $\mathbf{0}$ (that is, the circuit that outputs 0 on all inputs). As circuits $\mathsf{MB\text{-}AIPO}(x, m)$ and $\mathbf{0}$ differ on exactly one point (i.e., x) we can use the result by Boyle et al. [BCP14] (here recalled as Theorem 2.5) that already under indistinguishability obfuscation the two circuits are indistinguishable. We can, thus, proceed in three hybrid steps:

Game_1 Is the original MB-AIPO game, where $\mathcal{B}_2$ always receives an honest obfuscation of point function $I_{x,m}$.

Game_2 Is identical to before but now $\mathcal{B}_2$ receives an indistinguishability obfuscation of the constant zero circuit $\mathbf{0}$.

Game_3 Is the original MB-AIPO game, where $\mathcal{B}_2$ always receives an obfuscation of point function $I_{x,m'}$ where m' is a uniformly random point value.

We, thus, need to show that the difference between the above games is negligible. For the difference between games Game_1 and Game_2 we consider the following claim.

Claim 4.6. *Let* Sam *be the following sample algorithm. It runs adversary* $\mathcal{B}_1$ *to receive point function description* (x, m) *and auxiliary information* z*. It constructs an AIPO* $p_x \leftarrow\!\!\$\ \mathsf{AIPO}(x)$ *to then construct circuit* $C[p_x, m]$ *as in the construction. Additionally it constructs the constant zero circuit* $\mathbf{0}$ *padded to the same length as* $C[p_x, m]$*. It outputs* $(C[p_x, m], \mathbf{0}, z)$*.*

If AIPO *is secure then the such defined family of circuits is differing-inputs.*

Assume this is not the case. Then there exists an extractor Ext that on input $(C[p_x, m], \mathbf{0}, z)$ outputs a target value τ (with noticeable probability) such that $C[p_x, m](\tau) \neq \mathbf{0}(\tau)$ and, hence, $\tau = x$. We will use this extractor to break the security of the AIPO scheme. Let $(\mathcal{A}_1, \mathcal{A}_2)$ be an adversary against the security of AIPO. Adversary $\mathcal{A}_1$ runs $\mathcal{B}_1$ to receive point function description (x, m) and auxiliary information z. It chooses a random value r and computes $b \leftarrow \langle r, x\rangle$. It outputs $(x, (z, m, r, b))$. Adversary $\mathcal{A}_2$ gets as input a (z, m, r, b) and a point function obfuscation p which is either an obfuscation for I_x or for I_u for a uniformly random u. It constructs circuits $C[p, m]$ and the constant zero circuit $\mathbf{0}$ and runs extractor $\tau \leftarrow\!\!\$\ \mathsf{Ext}(C[p, m], \mathbf{0}, z)$. If $\tau = \perp$ it flips a bit and outputs it. Else, if $\tau \neq \perp$, then $p(\tau) = 1$ and, hence, value τ is either equal to x or to u. In this case adversary $\mathcal{A}_2$ outputs $\langle \tau, r\rangle = b$.

Analysis. Let us denote by ϵ the probability that Ext outputs a value $\tau \neq \perp$ in the differing-inputs game. Then, if $\tau = x$ adversary $\mathcal{A}_2$ will always output 1 as, by construction, $\langle x, r\rangle = b$. If, on the other hand, $\tau = u$, then adversary $\mathcal{A}_2$ will output 1 only with probability $\frac{1}{2}$ as u and r are randomly chosen values and r remains hidden from Ext. Thus, our adversary has a distinguishing advantage of $\frac{1}{2}\epsilon$. In the following we make this intuition formal. We note that our simulation technique is inspired by Brzuska and Mittelbach [BM14] who build variants of UCE security based on puncturable PRFs, iO and AIPO and that the formal analysis is almost taken verbatim.

Let us denote by $d = 0$ the event that in the AIPO-game, the honest point function I_x gets obfuscated, and let $d = 1$ describe the event that in the AIPO-game, I_u gets obfuscated for a random u. Let further ϵ be the probability that Ext returns a value $\tau \neq \perp$ in the differing-inputs game, that is, $\epsilon := \Pr[\perp \neq \mathsf{Ext} \mid d = 0]$. For readability we will drop the inputs given to adversaries Ext and $\mathcal{A}_2$ in the following formal treatment. We can now consider the distinguishing probability of our

adversary $\mathcal{A}_2$

$$
\begin{aligned}
&\Pr[\mathcal{A}_2 = 1 \mid d = 0] - \Pr[\mathcal{A}_2 = 1 \mid d = 1] \\
=&\Pr[\mathcal{A}_2 = 1 \mid d = 0, \mathsf{Ext} \neq \perp] \cdot \Pr[\mathsf{Ext} \neq \perp \mid d = 0] + \Pr[\mathcal{A}_2 = 1 \mid d = 0, \mathsf{Ext} = \perp] \cdot \Pr[\mathsf{Ext} = \perp \mid d = 0] - \\
&\Pr[\mathcal{A}_2 = 1 \mid d = 1] \\
=&\Pr[\mathsf{Ext} \neq \perp \mid d = 0] + \frac{1}{2} \cdot \Pr[\mathsf{Ext} = \perp \mid d = 0] - \Pr[\mathcal{A}_2 = 1 \mid d = 1] \\
=&\Pr[\mathsf{Ext} \neq \perp \mid d = 0] + \frac{1}{2} \cdot \Big(1 - \Pr[\mathsf{Ext} \neq \perp \mid d = 0]\Big) - \Pr[\mathcal{A}_2 = 1 \mid d = 1] \\
=&\frac{1}{2} \cdot \Pr[\mathsf{Ext} \neq \perp \mid d = 0] + \frac{1}{2} - \Pr[\mathcal{A}_2 = 1 \mid d = 1] = \frac{1}{2}\epsilon + \frac{1}{2} - \Pr[\mathcal{A}_2 = 1 \mid d = 1]
\end{aligned}
$$

Let U denote a random variable describing the choice of point function I_u (in case $d = 1$).

$$
\begin{aligned}
=&\frac{1}{2}\epsilon + \frac{1}{2} - \Pr[\mathcal{A}_2 = 1 \mid d = 1, \mathsf{Ext} \neq \perp] \cdot \Pr[\mathsf{Ext} \neq \perp \mid d = 1] + \\
&\qquad \Pr[\mathcal{A}_2 = 1 \mid d = 1, \mathsf{Ext} = \perp] \cdot \Pr[\mathsf{Ext} = \perp \mid d = 1] \\
=&\frac{1}{2}\epsilon + \frac{1}{2} - \\
&\frac{1}{2^{\mathsf{H.il}(\lambda)}} \sum_{u \in \{0,1\}^{\mathsf{H.il}(\lambda)}} \Big(\Pr[\mathcal{A}_2 = 1 \mid d = 1, U = u, \mathsf{Ext} \neq \perp] \cdot \Pr[\mathsf{Ext} \neq \perp \mid U = u, d = 1] + \\
&\qquad \Pr[\mathcal{A}_2 = 1 \mid d = 1, U = u, \mathsf{Ext} = \perp] \cdot \Pr[\mathsf{Ext} = \perp \mid U = u, d = 1] \Big)
\end{aligned}
$$

If adversary Ext outputs a value u (given that $d = 1$), then the probability that $\mathcal{A}_2$ outputs 1 ($\Pr[\mathcal{A}_2 = 1 \mid d = 1, U = u, \mathsf{Ext} \neq \perp]$) is equivalent to $\Pr_{R,b}[\langle R, u\rangle = b]$ where random variable R denotes the choice of value r by $\mathcal{A}_1$ to compute $b = \langle r, x^*\rangle$. Note that extractor Ext is independent of R and b and, thus, $\Pr_{R,b}[\langle R, u\rangle = b] = \frac{1}{2}$. It follows

$$
\begin{aligned}
=&\frac{1}{2}\epsilon + \frac{1}{2} - \\
&\frac{1}{2^{\mathsf{H.il}(\lambda)}} \sum_{u \in \{0,1\}^{\mathsf{H.il}(\lambda)}} \Big(\Pr_{R,b}[\langle R, u\rangle = b] \cdot \Pr[\mathsf{Ext} \neq \perp \mid U = u, d = 1] + \frac{1}{2} \cdot \Pr[\mathsf{Ext} = \perp \mid U = u, d = 1] \Big) \\
=&\frac{1}{2}\epsilon + \frac{1}{2} - \frac{1}{2^{\mathsf{H.il}(\lambda)}} \sum_{u \in \{0,1\}^{\mathsf{H.il}(\lambda)}} \left(\frac{1}{2} \cdot \Big(\Pr[\mathsf{Ext} \neq \perp \mid U = u, d = 1] + \Pr[\mathsf{Ext} = \perp \mid U = u, d = 1] \Big) \right) \\
=&\frac{1}{2}\epsilon + \frac{1}{2} - \frac{1}{2^{\mathsf{H.il}(\lambda)}} \sum_{u \in \{0,1\}^{\mathsf{H.il}(\lambda)}} \frac{1}{2} \cdot 1 = \frac{1}{2}\epsilon
\end{aligned}
$$

This establishes that adversary $\mathcal{A}_2$ is able to distinguish with noticeable probability since, by assumption, the success probability ϵ of extractor Ext is noticeable.

It remains to show that $\mathcal{A}_1$ implements an unpredictable distribution. By Definition 4.3 point x remains hidden given values z and m. As r is a uniformly random value chosen independently of x and b is a single bit which can be guessed it follows that, indeed, $\mathcal{A}_1$ implements an unpredictable distribution.

This concludes the proof of Claim 4.6. ◊

We next show that with Claim 4.6 it follows that games Game_1 and Game_2 are negligibly close. Boyle et al. [BCP14] who show that every indistinguishability obfuscator is also a differing-inputs obfuscator for circuit families that differ on at most polynomially many points (we give their result as Theorem 2.5 on page 7). As the circuits considered in Claim 4.6 differ on only a single point it follows that their obfuscation under an indistinguishability obfuscator are computationally indistinguishable and hence games Game_1 and Game_2 are negligibly close.

For games Game_2 and Game_3, the analysis is analogous. The only difference consists in m' being chosen at random. Thus, we can make use of Claim 4.6 with the sample adapted to output circuit $C[p_x, m']$. This concludes the proof. □

A SECOND CONSTRUCTION. In the next section we will construct a leakage resilient public-key encryption scheme, for which we will need an extended construction of MB-AIPO that we present next. Namely, we prove that the construction is still secure if, additionally, we return the AIPO of x. Intuitively, that should not harm security, because, given an MB-AIPO for (x, m), it is easy to construct an AIPO for x. However, making this statement formal requires some care.

Construction 4.7. *Let* AIPO *be a secure AIPO and* iO *be a secure indistinguishability obfuscator for all circuits in* $\mathcal{P}/\mathrm{poly}$*. We construct a weak MB-AIPO obfuscator* $\mathsf{MB\text{-}AIPO}$ *as follows. On input a point address* x *and value* m $\mathsf{MB\text{-}AIPO}$ *constructs a point obfuscation* $p_x \leftarrow\!\!\$\ \mathsf{AIPO}(x)$*. It then constructs the following circuit*

$$C[p_x, m](x^*) := \textbf{\textit{if}}\ (p_x(x^*) = 1)\ \textbf{\textit{then return}}\ m\ \textbf{\textit{else return}}\ \perp$$

and outputs an indistinguishability obfuscation of $C[p_x, m]$ *together with* p_x*.*

We next show that also this adapted construction fulfills the security properties of a weak MB-AIPO scheme.

Proof. We proceed by the following game hops where the first is identical to the MB-AIPO setting where the adversary gets an honest obfuscation of point function $I_{x,m}$ and the last is identical to the dual setting where it gets as input an obfuscation of $I_{x,m'}$ for a uniformly random point value m'.

Game_1 Is the original MB-AIPO game with $b = 0$. The adversary $\mathcal{B}_2$ gets $((\mathsf{iO}(C[p_x, m]), p_x), z)$.

Game_2 Instead of returning p_x, we construct a fresh point obfuscation of x, that is, the adversary $\mathcal{B}_2$ gets $((\mathsf{iO}(C[p_x, m]), \mathsf{AIPO}(x)), z)$.

Game_3 Instead of returning $\mathsf{AIPO}(x)$, we return $\mathsf{AIPO}(u)$ for a random point u, that is, the adversary $\mathcal{B}_2$ gets $((\mathsf{iO}(C[p_x, m]), \mathsf{AIPO}(u)), z)$.

Game_4 Instead of returning $\mathsf{iO}(C[p_x, m])$, we return $\mathsf{iO}(C[p_x, m'])$ for a random m', that is, the adversary $\mathcal{B}_2$ gets $((\mathsf{iO}(C[p_x, m']), \mathsf{AIPO}(u)), z)$.

Game_5 Instead of returning $\mathsf{AIPO}(u)$ for a random point u, we return $\mathsf{AIPO}(x)$, that is, the adversary $\mathcal{B}_2$ gets $((\mathsf{iO}(C[p_x, m']), \mathsf{AIPO}(x)), z)$.

Game_6 Instead of returning a fresh point obfuscation of x, we return p_x, that is, the adversary $\mathcal{B}_2$ gets $((\mathsf{iO}(C[p_x, m']), p_x), z)$.

Note that the last game corresponds to the $\mathsf{MB\text{-}AIPO}$-game where the point value is chosen uniformly at random. Hence, it suffices to show that the six games are computationally indistinguishable.

Game_1 TO Game_2. We reduce to the security of the indistinguishability obfusactor iO. Note that the two circuits p_x (given to the adversary in Game_1) and $\mathsf{AIPO}(x)$ (given to the adversary in Game_2) are functionally equivalent as they are two independently generated obfusactions of point function I_x. Let $(\mathcal{B}_1, \mathcal{B}_2)$ be a distinguisher between Game_1 and Game_2. Then, we construct an adversary against the security property of obfusactor iO analogously to the final game hop of the proof of Lemma 3.2.

Game_2 TO Game_3. We reduce to the security of the AIPO. Let $(\mathcal{B}_1, \mathcal{B}_2)$ be a distinguisher between Game_2 and Game_3. Then, we construct an adversary $(\mathcal{C}_1, \mathcal{C}_2)$ against the AIPO as follows: $\mathcal{C}_1$ runs $\mathcal{B}_1$ to obtain (z, x, m). It runs $p_x \leftarrow\!\!\$\ \mathsf{AIPO}(x)$, computes $C \leftarrow\!\!\$\ \mathsf{iO}(C[p_x, m])$ and returns $((z, C), x)$. That is, the auxiliary input returned by $\mathcal{C}_1$ is (z, C). $\mathcal{C}_2$ receives $((z, C), p)$ and runs $\mathcal{B}_2$ on $(z, (C, p))$. $\mathcal{C}_2$ outputs whatever $\mathcal{B}_2$ outputs. If p is $\mathsf{AIPO}(x)$, then the input distribution to $(\mathcal{B}_1, \mathcal{B}_2)$ is as in Game_2. If p is $\mathsf{AIPO}(u)$, then the input distribution to $(\mathcal{B}_1, \mathcal{B}_2)$ is as in Game_3. Hence, if $(\mathcal{B}_1, \mathcal{B}_2)$ is successful, then so is $(\mathcal{C}_1, \mathcal{C}_2)$.

It remains to show that $(\mathcal{C}_1, \mathcal{C}_2)$ is a valid adversary against AIPO, that is, that $\mathcal{C}_1$ is unpredictable. This follows from the strong unpredictability of $\mathcal{B}_1$ and the security of the AIPO, that is, given a predicator $\mathcal{P}$ against the strong unpredictability of $\mathcal{C}_1$, we will either construct an adversary against the AIPO or a predictor $\mathcal{R}$ against the strong unpredictability of $\mathcal{B}_1$.

Let $\mathcal{P}$ be a predictor against the unpredictability of $\mathcal{C}_1$. Then, the $\mathcal{R}$ against the strong unpredictability of $\mathcal{B}_1$ behaves as follows: It gets as input (z, m). It draws a random point v and runs $p_v \leftarrow\!\!\$\ \mathsf{AIPO}(v)$, $C \leftarrow\!\!\$\ \mathsf{iO}(C[p_v, m])$ and outputs whatever $\mathcal{P}(C, z)$ returns. We now need to argue that $\mathcal{P}$ produces the right output, although v is used in the generation of C and not x. We reduce to the AIPO security.

Assume that $\mathcal{P}$ has non-negligible probability of returning x when getting $(\mathsf{iO}(C[p_x, m]), z)$, but not when getting $(\mathsf{iO}(C[p_v, m]), z)$. Then, we construct an adversary $(\mathcal{D}_1, \mathcal{D}_2)$ against the AIPO security as follows. $\mathcal{D}_1$ runs $\mathcal{B}_1$ to get (z, x, m). It draws a random string r and sets b to be the inner product of r and x. $\mathcal{D}_1$ outputs $((z, m, r, b), x)$, that is, its leakage is (z, m). Now, $\mathcal{D}_2$ gets (z, m, r, b) as well as a point function p. It runs the predictor $\mathcal{P}(\mathsf{iO}(C[p, m], z))$ to obtain some value x'. It tests whether $p(x') = 1$. If not, it returns a random bit. If yes, then it returns 1 if and only if the inner product of r and x' is equal to b.

Now, to see that $(\mathcal{D}_1, \mathcal{D}_2)$ breaks the AIPO, we first show that it is a valid adversary, i.e., that $\mathcal{D}_1$ is unpredictable and then analyse the success probability. $\mathcal{D}_1$ is unpredictable because $\mathcal{B}_1$ is strongly unpredictable and because b is a single bit that can be guessed. Let us turn to the success probability of $(\mathcal{D}_1, \mathcal{D}_2)$. If p is a point obfuscation of p_x, then $\mathcal{P}$ returns x with non-negligible probability ν and in these cases, $\mathcal{D}_2$ returns 1 with probability 1, because the bit always matches. Thus, if p is a point function of p_x, then $\mathcal{D}_2$ returns 1 with probability $\nu + (1 - \nu) \cdot \frac{1}{2} = \frac{1}{2} + \frac{\nu}{2}$. If p is a point obfuscation of a random point p_v, then, independently of the behaviour of $\mathcal{P}$, $\mathcal{D}_2$ returns 1 with probability $\frac{1}{2}$. Hence, the success probability is non-negligible.

Game_3 TO Game_4. We reduce to the $\mathsf{MB\text{-}AIPO}$ security of Construction 4.4 that we established in Proposition 4.5. Let $(\mathcal{B}_1, \mathcal{B}_2)$ be an adversary that distinguishes between Game_3 and Game_4, where $\mathcal{B}_1$ is strongly unpredictable. We construct an adversary $(\mathcal{C}_1, \mathcal{C}_2)$ against the weak MB-AIPO property of Construction 4.4 as follows. $\mathcal{C}_1$ runs $\mathcal{B}_1$ and outputs whatever $\mathcal{B}_1$ outputs. $\mathcal{C}_2$ gets (C, z). It draws a random value u and computes $p \leftarrow\!\!\$\ \mathsf{AIPO}(u)$. It then runs $\mathcal{B}_2$ on $((C, p), z)$ and outputs whatever $\mathcal{B}_2$ outputs. $\mathcal{C}_1$ is strongly unpredictable, because $\mathcal{B}_1$ is. Moreover, the simulation is perfect. And hence, the advantage of $(\mathcal{C}_1, \mathcal{C}_2)$ is as big as the advantage of $(\mathcal{B}_1, \mathcal{B}_2)$.

Game_4 TO Game_5. Analogous to the game hop from Game_2 to Game_3.

Game_5 TO Game_6. Analogous to the game hop from Game_1 to Game_2. □

5 Leakage resilient public-key encryption

In this section, we will use the construction of weak MB-AIPO to build a leakage resilient public-key encryption scheme. Our result is inspired by Canetti et al. who show that multi-bit point obfuscation is tightly connected to symmetric encryption [CKVW10]. They give an intriguingly simple construction of a symmetric encryption scheme from an MB-AIPO as follows. Encryption under key k is defined as $\mathsf{Enc}_k(m) := \mathsf{MB\text{-}AIPO}(k, m)$. Correspondingly, decryption works as $\mathsf{Dec}_k(c) := c(k)$. Furthermore, they show how to build an MB-AIPO scheme from symmetric encryption. They classify the relationships between the two primitives depending on the strength of the MB-AIPO (resp., encryption scheme). In particular, they show that a version of MB-AIPO obfuscation implies the existence of a symmetric key encryption scheme secure in the presence of leakage (of the key) with the only requirement that the leakage computationally hides the secret key.

Let us recall a somewhat simplified version of their notion of semantic security of a symmetric encryption scheme with weak keys and auxiliary inputs:

Definition 5.1 ([CKVW10]: Symmetric Encryption with Weak Keys and Auxiliary Inputs)**.** *Let $\mathcal{D} = \{D_\lambda = (Z_\lambda, X_\lambda)\}_{\lambda\in\mathbb{N}}$ be an unpredictable distribution ensemble. We say that an encryption scheme has semantic security with keys chosen from $\{\mathcal{X}_\lambda\}_{\lambda\in\mathbb{N}}$ and auxiliary inputs from $\{\mathcal{Z}_\lambda\}_{\lambda\in\mathbb{N}}$ if there exists a* PPT *algorithm $\mathsf{Sim}(1^\lambda, \ell)$ such that, for all* PPT *adversaries $\mathcal{A}$ we have:*

$$\left|\Pr\left[\mathsf{SEM}_0^{\mathcal{X},\mathsf{Sim}}(\mathcal{A},\lambda)=1\right] - \Pr\left[\mathsf{SEM}_1^{\mathcal{X},\mathsf{Sim}}(\mathcal{A},\lambda)=1\right] - \right| \leq \mathrm{negl}(\lambda)$$

where the games $\mathsf{SEM}_0^{\mathcal{X},\mathsf{Sim}}$ for $b = 0, 1$ are defined via the following experiment:

1. *$(z, k) \leftarrow_\$ \mathcal{D}_\lambda$ and give z to $\mathcal{A}$*
2. *Adversary $\mathcal{A}$ submits a query m. Set $c_0 \leftarrow_\$ \mathsf{Enc}_k(m), c_1 \leftarrow_\$ \mathsf{Sim}(\lambda, |m|)$ and give c_b to $\mathcal{A}$.*
3. *The output of the game is the output of $\mathcal{A}$.*

Canetti et al. show that such a strong form of symmetric encryption exists if, and only if, certain types of MB-AIPOs exist. Their type of MB-AIPO requires that the message m is drawn independently from the point x. Their notion of MB-AIPO for *independent messages* is weaker than our notion of MB-AIPO against strongly computationally unpredictable distributions as presented in Definition 4.3 and, in particular, not affected by our impossibility result.

We improve the result by Canetti et al. by building a leakage-resilient encryption scheme, that is public key rather than symmetric key. We first give the variant of IND-CPA security with hard-to-invert key-leakage of a public-key encyption scheme $\mathcal{E}$ that we consider (we give the pseudocode in Figure 2). In the IND-CPA game with adversary $(\mathcal{A}_0, \mathcal{A}_1, \mathcal{A}_2)$ an initial adversary $\mathcal{A}_0$ takes as input the secret key and outputs some leakage z. Adversary $\mathcal{A}_1$ is run on input the public key pk and leakage z and outputs a single messages m together with some state st. Then, according to a secret bit b either message m or a uniformly random message m' of the same length is encrypted yielding ciphertext c which is given together with state st to the final adversary $\mathcal{A}_2$ which needs to guess bit b.[2]

[2]The described real-or-random notion of IND-CPA can be shown to be equivalent upto a factor of 2 in the reduction to the more frequently used left-or-right security notion [BDJR97].

$\mathsf{IND\text{-}CPA}_{\mathcal{E}}^{\mathcal{A}_0,\mathcal{A}_1,\mathcal{A}_2}(\lambda)$

$b \leftarrow\$ \{0,1\}$
$(pk, sk) \leftarrow\$ \mathcal{E}.\mathsf{KGen}(1^\lambda)$
$z \leftarrow\$ \mathcal{A}_0(1^\lambda, sk)$
$(m, st) \leftarrow\$ \mathcal{A}_1(1^\lambda, pk, z)$
if $b = 0$ **then**
  $c \leftarrow\$ \mathcal{E}.\mathsf{Enc}(m)$
else
  $r \leftarrow\$ \{0,1\}^{|m|}$
  $c \leftarrow\$ \mathcal{E}.\mathsf{Enc}(r)$
$b' \leftarrow\$ \mathcal{A}_2(1^\lambda, c, st)$
return $(1 = b')$

Figure 2: The IND-CPA game for public-key encryption schemes with hard-to-invert key leakage. An adversary is deemed admissible if it is PPT and if the output of $\mathcal{A}_0$ computationally hides the key, that is, z has super-logarithmic min-entropy.

Construction 5.2. *Let λ be the security parameter, let* AIPO *denote a point obfuscater and* iO *an indistinguishability obfuscator. Key generation picks a secret key $sk \leftarrow\$ \{0,1\}^\lambda$ as a uniformly random bit string of length λ. As public key it outputs a point obfuscation of x, that is, it outputs $pk \leftarrow\$ \mathsf{AIPO}(x)$. To encrypt a message m one constructs the circuit*

$$C[pk, m](x^*) := \textbf{\textit{if}}\ (pk(x^*) = 1)\ \textbf{\textit{then return}}\ m\ \textbf{\textit{else return}}\ \perp$$

and computes an indistinguishability obfuscation $c \leftarrow\$ \mathsf{iO}(C[pk, m])$ which yields the ciphertext c. For decryption one computes $m \leftarrow c(x)$.

Correctness of the scheme follows from the correctness criteria of indistinguishability obfuscation and AIPO. We reduce IND-CPA security of the scheme to the security of weak MB-AIPO (note, that an encryption is nothing but an obfuscation following Construction 4.4).

Proposition 5.3. *If Construction 4.7 is a weak* MB-AIPO*, then Construction 5.2 is a IND-CPA secure in the presence of computationally uninvertible leakage on the secret-key.*

Proof. Assume that there exists a successful adversary $\mathcal{A}_0, \mathcal{A}_1, \mathcal{A}_2$. We are going to construct an adversary $\mathcal{B}_1, \mathcal{B}_2$ against MB-AIPO.

Adversary $\mathcal{B}_1$ selects a random point x and runs adversary $\mathcal{A}_0$ on input the security parameter and x to receive some leakage z. It constructs an AIPO obfuscation $pk \leftarrow\$ \mathsf{AIPO}(x)$ and runs adversary $\mathcal{A}_1$ on input $(1^\lambda, pk, z)$ to receive a message m and state st. It outputs $((x, m), st)$. Adversary $\mathcal{B}_2$ gets as input an obfuscation c that is either equal to $\mathsf{MB\text{-}AIPO}(x, m)$ or equal to $\mathsf{MB\text{-}AIPO}(x, m')$ as well as state st. It runs adversary $\mathcal{A}_2$ on input $(1^\lambda, c, st)$ and outputs whatever $\mathcal{A}_2$ outputs.

Analysis. Our analysis proceeds in two steps. First, we show that if $\mathcal{A}_0, \mathcal{A}_1, \mathcal{A}_2$ is successful, then so is $\mathcal{B}_1, \mathcal{B}_2$. Then, we prove that $\mathcal{B}_1$ implements a strongly unpredictable distribution as required for weak MB-AIPO.

To see that $\mathcal{B}_1, \mathcal{B}_2$ are successful, we observe that the simulation is perfect. It remains to show that $\mathcal{B}_1$ implements a strongly unpredictable distribution. We reduce to the unpredictability of $\mathcal{A}_0$. Let $\mathcal{P}$ be a predictor against the strong unpredictability of $\mathcal{B}_1$, then we construct a predictor $\mathcal{R}$ against the unpredictability of $\mathcal{A}_0$.

$\mathcal{R}$ receives the leakage z that was created by $\mathcal{A}_0$. It draws a random point u and sets $pk \leftarrow\!\!\$\ \mathsf{AIPO}(u)$. Then, the predictor $\mathcal{R}$ runs $\mathcal{A}_1$ on (z, pk) to obtain a message m and some state st. It runs $\mathcal{P}$ on (m, st) to get a value x' and returns x'. Now, $\mathcal{A}_1$ gets as input $\mathsf{AIPO}(u)$ instead of $\mathsf{AIPO}(x)$. We argue that, assuming the security of AIPO, $\mathcal{P}$ is also successful on this distribution. Assume that $\mathcal{P}$ has non-negligible probability ν in returning x when $\mathcal{A}_1$ is run on $(z, \mathsf{AIPO}(x))$, but negligible probability ν in returning x when $\mathcal{A}_1$ is run on $(z, \mathsf{AIPO}(u))$ for a random u. Then, we construct an adversary $(\mathcal{C}_1, \mathcal{C}_2)$ against the AIPO as follows. $\mathcal{C}_1$ runs $\mathcal{A}_0$ to create (z, x). Then, $\mathcal{C}_1$ draws a random string r and sets b to be the inner product of r and x. $\mathcal{C}_1$ returns $((z, r, b), x)$. The second stage $\mathcal{C}_2$ gets $((z, r, b), p)$ and runs $\mathcal{A}_1$ on (z, p) to obtain a message m and some state st. It runs $\mathcal{P}$ on (m, st) to get a value x'. It checks whether $p(x') = 1$. If no, it returns a random bit. If yes, then it returns 1 if and only if the inner product of x' and r is equal to b.

Firstly, the first stage $\mathcal{C}_1$ is unpredictable because $\mathcal{A}_0$ is and b is only a single bit of x. Now, let us see that $(\mathcal{C}_1, \mathcal{C}_2)$ are also successful. If p is a point obfuscation of x, then $\mathcal{P}$ returns x with probability ν and thus, $\mathcal{C}_2$ returns 1 with probability $\nu + (1 - \nu) \cdot \frac{1}{2}$. If p is a point obfuscation of x, then, indepdently of the behaviour of $\mathcal{P}$, $\mathcal{C}_2$ returns 1 with probability $\frac{1}{2}$ thus yielding an overall advantage of $\frac{\nu}{2}$ which concludes the proof. □

Acknowledgments

We would like to especially thank Paul Baecher, Victoria Fehr and Giorgia Azzurra Marson for many helpful comments and discussions throughout the various stages of this work. This work was supported by CASED (`www.cased.de`).

References

[ABG+13] Prabhanjan Ananth, Dan Boneh, Sanjam Garg, Amit Sahai, and Mark Zhandry. Differing-inputs obfuscation and applications. Cryptology ePrint Archive, Report 2013/689, 2013. `http://eprint.iacr.org/2013/689`. (Cited on pages 4 and 6.)

[BC10] Nir Bitansky and Ran Canetti. On strong simulation and composable point obfuscation. In Tal Rabin, editor, *Advances in Cryptology – CRYPTO 2010*, volume 6223 of *Lecture Notes in Computer Science*, pages 520–537, Santa Barbara, CA, USA, August 15–19, 2010. Springer, Berlin, Germany. (Cited on pages 3 and 13.)

[BCP14] Elette Boyle, Kai-Min Chung, and Rafael Pass. On extractability obfuscation. In Yehuda Lindell, editor, *TCC 2014: 11th Theory of Cryptography Conference*, volume 8349 of *Lecture Notes in Computer Science*, pages 52–73, San Diego, CA, USA, February 24–26, 2014. Springer, Berlin, Germany. (Cited on pages 4, 6, 7, 14, and 17.)

[BDJR97] Mihir Bellare, Anand Desai, Eric Jokipii, and Phillip Rogaway. A concrete security treatment of symmetric encryption. In *38th Annual Symposium on Foundations of Computer Science*, pages 394–403, Miami Beach, Florida, October 19–22, 1997. IEEE Computer Society Press. (Cited on page 19.)

[BFM14] Christina Brzuska, Pooya Farshim, and Arno Mittelbach. Indistinguishability obfuscation and UCEs: The case of computationally unpredictable sources. In Juan A. Garay and Rosario Gennaro, editors, *CRYPTO 2014*, Lecture Notes in Computer Science, pages ??–??, Santa Barbara, CA, USA, August 17–21, 2014. Springer, Berlin, Germany. (Cited on pages 4, 9, and 13.)

[BGI+01] Boaz Barak, Oded Goldreich, Russell Impagliazzo, Steven Rudich, Amit Sahai, Salil P. Vadhan, and Ke Yang. On the (im)possibility of obfuscating programs. In Joe Kilian, editor, *Advances in Cryptology – CRYPTO 2001*, volume 2139 of *Lecture Notes in Computer Science*, pages 1–18, Santa Barbara, CA, USA, August 19–23, 2001. Springer, Berlin, Germany. (Cited on pages 3, 4, and 6.)

[BGI+12] Boaz Barak, Oded Goldreich, Russell Impagliazzo, Steven Rudich, Amit Sahai, Salil Vadhan, and Ke Yang. On the (im)possibility of obfuscating programs. *J. ACM*, 59(2):6:1–6:48, May 2012. (Cited on pages 3, 4, and 6.)

[BHK13] Mihir Bellare, Viet Tung Hoang, and Sriram Keelveedhi. Instantiating random oracles via UCEs. In Ran Canetti and Juan A. Garay, editors, *Advances in Cryptology – CRYPTO 2013, Part II*, volume 8043 of *Lecture Notes in Computer Science*, pages 398–415, Santa Barbara, CA, USA, August 18–22, 2013. Springer, Berlin, Germany. (Cited on pages 4, 9, and 13.)

[BM14] Christina Brzuska and Arno Mittelbach. Using indistinguishability obfuscation via uces. Cryptology ePrint Archive, Report 2014/381, 2014. `http://eprint.iacr.org/`. (Cited on pages 13, 14, and 15.)

[BP12] Nir Bitansky and Omer Paneth. Point obfuscation and 3-round zero-knowledge. In Ronald Cramer, editor, *TCC 2012: 9th Theory of Cryptography Conference*, volume 7194 of *Lecture Notes in Computer Science*, pages 190–208, Taormina, Sicily, Italy, March 19–21, 2012. Springer, Berlin, Germany. (Cited on pages 3, 4, 5, 7, 9, 12, 13, and 25.)

[BR06] Mihir Bellare and Phillip Rogaway. The security of triple encryption and a framework for code-based game-playing proofs. In Serge Vaudenay, editor, *Advances in Cryptology – EUROCRYPT 2006*, volume 4004 of *Lecture Notes in Computer Science*, pages 409–426, St. Petersburg, Russia, May 28 – June 1, 2006. Springer, Berlin, Germany. (Cited on page 11.)

[BST13] Mihir Bellare, Igors Stepanovs, and Stefano Tessaro. Poly-many hardcore bits for any one-way function. Cryptology ePrint Archive, Report 2013/873, 2013. `http://eprint.iacr.org/`. (Cited on page 4.)

[BZ13] Dan Boneh and Mark Zhandry. Multiparty key exchange, efficient traitor tracing, and more from indistinguishability obfuscation. Cryptology ePrint Archive, Report 2013/642, 2013. `http://eprint.iacr.org/2013/642`. (Cited on page 4.)

[Can97] Ran Canetti. Towards realizing random oracles: Hash functions that hide all partial information. In Burton S. Kaliski Jr., editor, *Advances in Cryptology – CRYPTO'97*, volume 1294 of *Lecture Notes in Computer Science*, pages 455–469, Santa Barbara, CA, USA, August 17–21, 1997. Springer, Berlin, Germany. (Cited on pages 3, 7, 13, and 25.)

[CD08] Ran Canetti and Ronny Ramzi Dakdouk. Obfuscating point functions with multibit output. In Nigel P. Smart, editor, *Advances in Cryptology – EUROCRYPT 2008*, volume 4965 of *Lecture Notes in Computer Science*, pages 489–508, Istanbul, Turkey, April 13–17, 2008. Springer, Berlin, Germany. (Cited on pages 3, 12, 13, and 25.)

[CKVW10] Ran Canetti, Yael Tauman Kalai, Mayank Varia, and Daniel Wichs. On symmetric encryption and point obfuscation. In Daniele Micciancio, editor, *TCC 2010: 7th Theory*

of Cryptography Conference, volume 5978 of *Lecture Notes in Computer Science*, pages 52–71, Zurich, Switzerland, February 9–11, 2010. Springer, Berlin, Germany. (Cited on pages 3, 5, 9, 13, and 19.)

[CMR98] Ran Canetti, Daniele Micciancio, and Omer Reingold. Perfectly one-way probabilistic hash functions (preliminary version). In *30th Annual ACM Symposium on Theory of Computing*, pages 131–140, Dallas, Texas, USA, May 23–26, 1998. ACM Press. (Cited on pages 3 and 13.)

[CTL97] Christian Collberg, Clark Thomborson, and Douglas Low. A taxonomy of obfuscating transformations. Technical Report 148, Department of Computer Science, University of Auckland, July 1997. (Cited on page 3.)

[DKL09] Yevgeniy Dodis, Yael Tauman Kalai, and Shachar Lovett. On cryptography with auxiliary input. In Michael Mitzenmacher, editor, *41st Annual ACM Symposium on Theory of Computing*, pages 621–630, Bethesda, Maryland, USA, May 31 – June 2, 2009. ACM Press. (Cited on page 3.)

[Fis99] Marc Fischlin. Pseudorandom function tribe ensembles based on one-way permutations: Improvements and applications. In Jacques Stern, editor, *Advances in Cryptology – EUROCRYPT'99*, volume 1592 of *Lecture Notes in Computer Science*, pages 432–445, Prague, Czech Republic, May 2–6, 1999. Springer, Berlin, Germany. (Cited on pages 3 and 13.)

[GGH+13] Sanjam Garg, Craig Gentry, Shai Halevi, Mariana Raykova, Amit Sahai, and Brent Waters. Candidate indistinguishability obfuscation and functional encryption for all circuits. In *54th Annual Symposium on Foundations of Computer Science*, pages 40–49, Berkeley, CA, USA, October 26–29, 2013. IEEE Computer Society Press. (Cited on pages 4 and 6.)

[GGHR14] Sanjam Garg, Craig Gentry, Shai Halevi, and Mariana Raykova. Two-round secure MPC from indistinguishability obfuscation. In Yehuda Lindell, editor, *TCC 2014: 11th Theory of Cryptography Conference*, volume 8349 of *Lecture Notes in Computer Science*, pages 74–94, San Diego, CA, USA, February 24–26, 2014. Springer, Berlin, Germany. (Cited on page 4.)

[GGHW13] Sanjam Garg, Craig Gentry, Shai Halevi, and Daniel Wichs. On the implausibility of differing-inputs obfuscation and extractable witness encryption with auxiliary input. Cryptology ePrint Archive, Report 2013/860, 2013. `http://eprint.iacr.org/2013/860`. (Cited on page 5.)

[GK05] Shafi Goldwasser and Yael Tauman Kalai. On the impossibility of obfuscation with auxiliary input. In *46th Annual Symposium on Foundations of Computer Science*, pages 553–562, Pittsburgh, PA, USA, October 23–25, 2005. IEEE Computer Society Press. (Cited on pages 3 and 6.)

[GLSW14] Craig Gentry, Allison Lewko, Amit Sahai, and Brent Waters. Indistinguishability obfuscation from the multilinear subgroup elimination assumption. Cryptology ePrint Archive, Report 2014/309, 2014. `http://eprint.iacr.org/`. (Cited on page 4.)

[GR07] Shafi Goldwasser and Guy N. Rothblum. On best-possible obfuscation. In Salil P. Vadhan, editor, *TCC 2007: 4th Theory of Cryptography Conference*, volume 4392 of *Lecture Notes in Computer Science*, pages 194–213, Amsterdam, The Netherlands, February 21–24, 2007. Springer, Berlin, Germany. (Cited on pages 4 and 13.)

[Had10] Satoshi Hada. Secure obfuscation for encrypted signatures. In Henri Gilbert, editor, *Advances in Cryptology – EUROCRYPT 2010*, volume 6110 of *Lecture Notes in Computer Science*, pages 92–112, French Riviera, May 30 – June 3, 2010. Springer, Berlin, Germany. (Cited on page 3.)

[HMLS07] Dennis Hofheinz, John Malone-Lee, and Martijn Stam. Obfuscation for cryptographic purposes. In Salil P. Vadhan, editor, *TCC 2007: 4th Theory of Cryptography Conference*, volume 4392 of *Lecture Notes in Computer Science*, pages 214–232, Amsterdam, The Netherlands, February 21–24, 2007. Springer, Berlin, Germany. (Cited on page 3.)

[HRsV07] Susan Hohenberger, Guy N. Rothblum, abhi shelat, and Vinod Vaikuntanathan. Securely obfuscating re-encryption. In Salil P. Vadhan, editor, *TCC 2007: 4th Theory of Cryptography Conference*, volume 4392 of *Lecture Notes in Computer Science*, pages 233–252, Amsterdam, The Netherlands, February 21–24, 2007. Springer, Berlin, Germany. (Cited on page 3.)

[HSW14] Susan Hohenberger, Amit Sahai, and Brent Waters. Replacing a random oracle: Full domain hash from indistinguishability obfuscation. In Phong Q. Nguyen and Elisabeth Oswald, editors, *Advances in Cryptology – EUROCRYPT 2014*, volume 8441 of *Lecture Notes in Computer Science*, pages 201–220, Copenhagen, Denmark, May 11–15, 2014. Springer, Berlin, Germany. (Cited on page 4.)

[MH14] Takahiro Matsuda and Goichiro Hanaoka. Chosen ciphertext security via point obfuscation. In Yehuda Lindell, editor, *TCC 2014: 11th Theory of Cryptography Conference*, volume 8349 of *Lecture Notes in Computer Science*, pages 95–120, San Diego, CA, USA, February 24–26, 2014. Springer, Berlin, Germany. (Cited on pages 4, 5, 8, 9, and 12.)

[PST13] Rafael Pass, Karn Seth, and Sidharth Telang. Indistinguishability obfuscation from semantically-secure multilinear encodings. Cryptology ePrint Archive, Report 2013/781, 2013. `http://eprint.iacr.org/`. (Cited on page 4.)

[SW13] Amit Sahai and Brent Waters. How to use indistinguishability obfuscation: Deniable encryption, and more. Cryptology ePrint Archive, Report 2013/454, 2013. `http://eprint.iacr.org/2013/454`. (Cited on page 4.)

[Wee05] Hoeteck Wee. On obfuscating point functions. In Harold N. Gabow and Ronald Fagin, editors, *37th Annual ACM Symposium on Theory of Computing*, pages 523–532, Baltimore, Maryland, USA, May 22–24, 2005. ACM Press. (Cited on pages 3, 13, and 25.)

A Constructions of point obfuscation schemes

In the following we present the two constructions (and their underlying assumptions) of AIPOs secure with respect to Definition 2.7. These constructions are, to the best of our knowledge, the only two candidates that achieve AIPO security. Other constructions, such as the construction

by Wee [Wee05] either do not consider auxiliary information or put additional restrictions on the auxiliary information.

The first construction is due to Canetti [Can97] who bases his construction on a strong variant of the DDH assumption. We here present the construction in the formulation of [BP12] and then present the assumption it is based on.

Construction A.1 (AIPO obfuscator due to [Can97])**.** *Let $\mathcal{G} := \{\mathbb{G}_\lambda\}_{\lambda\in\mathbb{N}}$ be a group ensemble, where each $\mathbb{G}_\lambda$ is a group of prime order $p_\lambda \in (2^{\lambda-1}, 2^\lambda)$. We define an obfuscator $\mathcal{O}$ for points in the domain $\mathbb{Z}_{p_\lambda}$ as follows: $I_x \overset{\mathcal{O}}{\mapsto} C(r, r^x)$, where $r \leftarrow\!\!\$\ \mathbb{G}_\lambda$ is a random generator of $\mathbb{G}_\lambda$, and $C(r, r^x)$ is a circuit which on input i, checks whether $r^x = r^i$.*

Assumption A.2 ([Can97],[BP12])**.** *There exists an ensemble of prime order groups $\mathcal{G} := \{\mathbb{G}_\lambda\}_{\lambda\in\mathbb{N}}$ such that for any unpredictable distribution $\mathcal{D} = \{D_\lambda = (Z_\lambda, Y_\lambda)\}_{\lambda\in\mathbb{N}}$ with support $\{0,1\}^{\mathrm{poly}(\lambda)} \times \mathbb{Z}_{p_\lambda}$, it holds that for all* PPT *algorithms $\mathcal{A}$ there exists a negligible function* negl *such that*

$$\left|\Pr_{r\leftarrow\!\!\$\ \mathbb{G}_\lambda, \leftarrow\!\!\$\,(z,x)\leftarrow\!\!\$\ D_\lambda}[\mathcal{A}(z,r,r^x)=1] - \Pr_{r\leftarrow\!\!\$\ \mathbb{G}_\lambda, z\leftarrow\!\!\$\ Z_\lambda, u\leftarrow\!\!\$\ \mathbb{Z}_{p_\lambda}}[\mathcal{A}(z,r,r^u)=1]\right| \leq \mathrm{negl}(\lambda)$$

The second candidate construction for AIPO is due to Bitansky and Paneth [BP12] who adapt the point obfuscation scheme of Wee [Wee05] to allow for auxiliary input. Their construction is based on an assumption on the existence of strong pseudorandom permutations. Let us recall the underlying assumption (which generalizes the original assumption due to Wee [Wee05]) before recalling the construction.

Assumption A.3 ([BP12])**.** *There exists an ensemble of permutation families $\mathcal{F} = \{\mathcal{F}_\lambda = \{f\}\}$ such that for any unpredictable distribution ensemble $\mathcal{D} = \{D_\lambda = (Z_\lambda, Y_\lambda)\}_{\lambda\in\mathbb{N}}$, the following two distribution ensembles are also unpredictable:*

- $((Z_\lambda, f(Y_\lambda), f); Y_\lambda)$
- $((Z_\lambda, f); f(Y_\lambda))$,

where in both $f \leftarrow\!\!\$\ \mathcal{F}_\lambda$ (independently of D_λ).

Based on Assumption A.3, Bitansky and Paneth show that the following construction yields an AIPO obfuscator satisfying Definition 2.7 [BP12].

Construction A.4 ([BP12])**.** *Let $\mathcal{F}$ be a family of permutations as given by Assumption A.3. AIPO obfuscator $\mathcal{O}$ works as follows: given a point $y \in \{0,1\}^\lambda$, $\mathcal{O}$ samples 3λ permutations $\{f_i\}_{i\in[3\lambda]}$ from $\mathcal{F}_\lambda$ and 3λ strings $\{r_i\}_{i\in[3\lambda]}$ from $\{0,1\}^\lambda$. For every $i \in [3\lambda]$, let $f^i := f_i \circ f_{i-1} \circ \ldots \circ f_1$ (where $\circ$ denotes composition). Obfuscator $\mathcal{O}$ outputs a circuit C_y that has hardcoded into it the randomness of $\mathcal{O}$, $\{f_i, r_i\}_{i\in[3\lambda]}$ and the bits $\{b_i := \langle r_i, f^i(y)\rangle\}_{i\in[3\lambda]}$, where $\langle .,.\rangle$ denotes the inner product over $\mathbb{GF}_2$. Circuit C_y outputs 1 on a point x if for all $i \in [3\lambda] : b_i = \langle r_i, f^i(x)\rangle$; and 0 otherwise.*

From AIPO to MB-AIPO. Constructions of point obfuscation schemes for point functions with multi-bit output have first been studied by Canetti and Dakdouk [CD08] who show that composability of plain AIPOs is a necessary condition for the existence of MB-AIPOs. To the best of our knowledge no direct constructions of MB-AIPOs have been proposed in the literature.