A Coin-Free Oracle-Based Augmented Black Box Framework

Kyosuke Yamashita[1], Mehdi Tibouchi[1,2], and Masayuki Abe[1,2]

[1] Graduate School of Informatics, Kyoto University, Japan
yamashita.kyousuke.75w@st.kyoto-u.ac.jp, abe.masayuki.7a@kyoto-u.ac.jp
[2] Secure Platform Laboratories, NTT Corporation, Japan
tibouchi.mehdi@lab.ntt.co.jp

Abstract. After the work of Impagliazzo and Rudich (STOC, 1989), the black box framework has become one of the main research domain of cryptography. However black box techniques say nothing about non-black box techniques such as making use of zero-knowledge proofs. Brakerski *et al.* introduced a new black box framework named *augmented* black box framework, in which they gave a zero-knowledge proof oracle in addition to a base primitive oracle (TCC, 2011). They showed a construction of a non-interactive zero knowledge proof system based on a witness indistinguishable proof system oracle. They presented augmented black box construction of chosen ciphertext secure public key encryption scheme based on chosen plaintext secure public key encryption scheme and augmented black box separation between one-way function and key agreement.

In this paper we simplify the work of Brakerski *et al.* by introducing a proof system oracle without witness indistinguishability, named *coin-free* proof system oracle, that aims to give the same construction and separation results of previous work. As a result, the augmented black box framework becomes easier to handle. Since our oracle is not witness indistinguishable, our result encompasses the result of previous work.

Keywords: Black Box Construction · Zero-Knowledge Proof · NIZK · Witness Indistinguishability.

1 Introduction

Investigating the relationships between cryptographic primitives is one of the most important task in theoretical cryptography. After the work of Impagliazzo and Rudich [7], the black box framework has become one of the main research domain of cryptography. We can deeply understand the condition for the existence of primitives through the black box framework. Non-black box techniques are also extensively studied, whereas black box techniques say nothing about them. A widely known non-black box construction result is the work of Naor and Yung [10], which make use of a zero-knowledge (ZK) proof [5] to construct a chosen ciphertext secure public key encryption scheme (CCA-PKE) based on a chosen plaintext secure public key encryption scheme (CPA-PKE).

Although black box and non-black box techniques are developed independently each other, a new framework that combines them came into existence. Brakerski *et al.* [2] introduced the *augmented* black box framework, which makes use of a ZK oracle in addition to a cryptographic primitive oracle. They presented an oracle that instantiates a witness indistinguishable (WI) proof system [4] and showed that they could construct a non-interactive zero-knowledge proof (NIZK) based on the oracle in a black box manner. They demonstrated the power of the framework by showing construction and separation results; they showed the augmented black box construction of CCA-PKE based on CPA-PKE following the Naor-Yung construction [10], and the augmented black box separation between one-way function (OWF) and key agreement (KA) [3].

Here we explain the motivation of our work. In the black box research, making an oracle that implements a base primitive simpler is an important direction. Introducing a simplified oracle helps to handle the oracle. Moreover it may make security proofs simpler. One of the major black box technique is relativizing reduction [11], which assures that a black box construction/reduction result holds relative to any oracle that implements a base primitive. In the beginning of the line of the black box task, researchers treated simple oracles such as implementing OWF [10]. However as more sophisticated primitives appeared, researchers had to deal with oracles that implement these primitives in the black box framework. For instance they began to handle oracles implementing trapdoor permutation [1,13], which led more advanced security proof. Moreover in [2], the augmented black box framework was accompanied by further complicated oracle that implements a NIZK. Although the augmented black box framework is an elegant framework, security proofs in this framework might become cumbersome task due to the high complexity of the oracle. Thus it it fruitful to simplify the oracle in the augmented black box framework.

In this paper we simplify the work of [2] by introducing a simpler proof system oracle without witness indistinguishability that aims to give the same construction and separation results of previous work. More concrete we simplify the proof system oracle by removing randomness from the interface of the prover oracle, and show that we can construct a WI proof system based on the simplified oracle. As a result the augmented black box framework becomes easier to treat. Moreover our result encompasses the result of [2], since the new oracle implements a proof system without witness indistinguishability.

2 Preliminaries

We follow the terminologies in [2] . Throughout this paper $n \in \mathbb{N}$ denotes the security parameter. We denote polynomial functions and negligible functions by poly and negl respectively. A PPT machine represents a probabilistic polynomial time Turing machine for which there exists a poly s.t. for any input x the running time is bounded by poly($|x|$). An oracle machine is a Turing machine which is allowed to make queries to an oracle. We write M^O an oracle machine M with oracle access to an oracle O.

We write $\mathcal{L} \in \mathtt{NP}^O$ for a language $\mathcal{L}$ and an oracle O if there exists the following PPT M whose input is a pair (x, w) s.t.

- the running time of M is bounded by $\mathrm{poly}(|x|)$; and
- $x \in \mathcal{L}$ iff there exists an witness w s.t. $M^O(x, w)$ accepts.

For any $\mathcal{L} \in \mathtt{NP}^O$, we let $R_{\mathcal{L}}$ denote an $\mathtt{NP}$-relationship associated with $\mathcal{L}$.

We define (cryptographic) primitives formally.

Definition 1 (primitive). *A primitive P is a pair (F_P, R_P) of a set of functions $f : \{0,1\}^* \to \{0,1\}^*$ and a relation over pairs (f, M) where $f \in F_P$ and M is a Turing machine.*

We say f *implements* P or f is an *implementation* of P if $f \in F_P$. A Turing machine M *breaks* the security of P if there exists an implementation $f \in F_P$ s.t. $(f, M) \in R_P$. Thus, we say $f \in F_P$ is a *secure implementation* of P if there exists no PPT M s.t. $(f, M) \in R_P$.

Definition 2 (OWF). *A function $f : \{0,1\}^* \to \{0,1\}^*$ is a* one-way function *if the following conditions hold;*

- *there exists a PPT M s.t. $M(x) = f(x)$ for all x, and*
- *for any PPT $\mathcal{A}$, it holds that* $\Pr[\mathcal{A}(f(x)) \in f^{-1}(f(x))] \leq \mathrm{negl}$.

When the second condition holds for a negligible function ϵ, we say the OWF is ϵ-OWF and it has ϵ-*security*.

Definition 3. *For an oracle O, we say that*

- O implements *a primitive P if there exists an implementation $f \in F_P$ that can be computed by an oracle PPT machine with oracle access to O;*
- *an implementation $f \in F_P$ is* secure *relative to O if there is no oracle PPT machine M with oracle access to O s.t. M^O breaks the security of f; and*
- *a primitive P* exists *relative to O if there exists a secure implementation $f \in F_P$ relative to O.*

We present syntactical and security definitions of a NIZK.

Definition 4 (NIZK). *A tuple of Turing machines* $(\mathsf{Crs}, \mathsf{Prv}, \mathsf{Vrf}, \mathsf{CrsSim}, \mathsf{PrvSim})$ *that work as follows is a* non-interactive zero-knowledge proof system *for a language $\mathcal{L}$ in the common random string model where* Vrf *is deterministic and others are probabilistic:*

Crs: *$crs \leftarrow \mathsf{Crs}(1^n)$ takes a security parameter, and outputs crs.*
Prv: *$\pi \leftarrow \mathsf{Prv}(crs, x, w)$ takes crs, an instance x and a witness w, and outputs a proof π or $\perp$.*
Vrf: *$b \leftarrow \mathsf{Vrf}(crs, x, \pi)$ takes crs, an instance x and a proof π, and outputs a bit $b \in \{0,1\}$.*

CrsSim: $(crs, \tau) \leftarrow \mathsf{CrsSim}(1^n)$ *takes a security parameter, and outputs* τ *and* crs.

PrvSim: $\pi \leftarrow \mathsf{PrvSim}(crs, x, \tau)$ *takes* crs*, an instance* x *and* τ*, and outputs* π.

Definition 5 (security properties of a NIZK). *A NIZK* $(\mathsf{Crs}, \mathsf{Prv}, \mathsf{Vrf}, \mathsf{CrsSim}, \mathsf{PrvSim})$ *for a language* $\mathcal{L}$ *is a NIZK with* perfect complete, statistical sound and adaptive black box zero-knowledge *properties if it has the following properties;*

perfect completeness: *for any* $n \in \mathbb{N}$*, for any* $(x, w) \in R_{\mathcal{L}}$ *and any* $crs \in \{0,1\}^{\mathrm{poly}(n)}$*,* $\mathsf{Vrf}(crs, x, \mathsf{Prv}(crs, x, w)) = 1$*;*

statistical soundness: *for any* $n \in \mathbb{N}$*, for any* $x \notin \mathcal{L}$ *and any* $\pi \in \{0,1\}^{\mathrm{poly}(n)}$*,* $\Pr crs \leftarrow \mathsf{Crs}(1^n) \mathsf{Vrf}(crs, x, \pi) = 1 \leq \mathrm{negl}$*; and*

adaptive black box zero-knowledge: *for any adversary* $\mathcal{A}$*, the following is negligible;*

$$\left| \Pr\begin{bmatrix} \text{crs} \leftarrow \mathsf{Crs}(1^n); & \\ (x,w) \leftarrow \mathcal{A}(\text{crs}); & : \mathcal{A}(\pi) = 1 \\ \pi \leftarrow \mathsf{Prv}(crs, x, w) & \wedge (x,w) \in R_{\mathcal{L}} \end{bmatrix} - \Pr\begin{bmatrix} (crs, \tau) \leftarrow \mathsf{CrsSim}(1^n); & \\ (x,w) \leftarrow \mathcal{A}(crs); & : \mathcal{A}(\pi) = 1 \\ \pi \leftarrow \mathsf{PrvSim}(crs, x, \tau) & \wedge (x,w) \in R_{\mathcal{L}} \end{bmatrix} \right|.$$

We simply denote a NIZK $(\mathsf{Crs}, \mathsf{Prv}, \mathsf{Vrf}, \mathsf{CrsSim}, \mathsf{PrvSim})$ with perfect complete, statistical sound and adaptive black box zero-knowledge properties by a NIZK.

3 WI Proof System Oracle

In this section we review the work of Brakerski *et al.*[2]. They introduced an instantiation of a WI proof system oracle and presented a construction of a NIZK based on the oracle. Moreover they defined the augmented black box framework and demonstrated the power of the framework; they showed the construction of a CCA-PKE based on a CPA-PKE oracle and the WI proof system oracle, and showed the separation between OWF and KA in the augmented black box framework.

Definition 6 (proof system). *A pair* (P, V) *of machines that works as follows is a* proof system *for a language* $\mathcal{L}$*;*

P: $\pi \leftarrow \mathsf{P}(x, w, r)$ *takes an instance* x*, a witness* w *and a random coin* r*, and outputs a proof* π*, and*

V: $b \leftarrow \mathsf{V}(x, \pi)$ *takes an instance* x *and a proof* π*, and outputs a bit* b*, where* V *accepts* π *if* $b = 1$ *and* V *rejects otherwise.*

Definition 7. *A proof system* (P, V) *for a language* $\mathcal{L}$ *is a proof system with* perfect complete and statistical sound *properties if it has the following properties;*

perfect completeness: *for any* $n \in \mathbb{N}$*, for any* $(x, w) \in R_{\mathcal{L}}$*, and any random coin* $r \in \{0,1\}^n$, $\mathsf{V}(x, \mathsf{P}(x, w, r)) = 1$*;*

perfect soundness: *for any* $n \in \mathbb{N}$*, any* $x \notin \mathcal{L}$*, and any* $\pi \in \{0,1\}^{\mathrm{poly}(n)}$*,* $\mathsf{V}(x, \mathsf{P}(x, w, r)) = 0$.

We simply say a proof system with perfect complete and perfect sound properties a proof system.

Definition 8 (WI proof system). *A proof system* $\mathsf{WI} = (\mathsf{P}, \mathsf{V})$ *for a language* $\mathcal{L}$ *is a* witness indistinguishable proof system*, if for any adversary* $\mathcal{A}$ *the advantage* $|\Pr[\mathtt{ExptWI}_{\mathcal{A}}(n) = 1] - \frac{1}{2}\ |$ *of the following experiment* $\mathtt{ExptWI}_{\mathcal{A}}(n)$ *is negligible;*

$$\begin{array}{rcl} (x, w_0, w_1) \leftarrow \mathcal{A}^{\mathsf{WI}}(1^n); & & \\ b \leftarrow \{0,1\};\ r \leftarrow \{0,1\}^n; & & \textit{if } (x, w_0), (x, w_1) \in R_{\mathcal{L}} \\ \pi \leftarrow \mathsf{P}(x, w_b, r); & : & \textit{output 1 iff } b' = b \\ b' = \mathcal{A}^{\mathsf{WI}}(1^n, \pi) & & \textit{else output a random bit.} \end{array}$$

Instantiation of a WI Proof System Oracle

Fix an oracle O that implements a primitive. For the reminder of this paper, we set $\mathcal{L} = \mathsf{CIRCUIT\text{-}SAT}^O$. The WI proof system oracle is defined as following prover and verifier oracles;

prover oracle: The prover oracle P is a random function s.t. $\mathsf{P} : \{0,1\}^{3n} \to \{0,1\}^{7n}$. The input is parsed as tuples $(x, w, r) \in \{0,1\}^n \times \{0,1\}^n \times \{0,1\}^n$. Note that P does not check if $(x, w) \in R_{\mathcal{L}}$.

verifier oracle: The verifier oracle V is a function s.t. $\mathsf{V} : \{0,1\}^{8n} \to \{0,1\}$. The input is parsed as pairs $(x, \pi) \in \{0,1\}^n \times \{0,1\}^{7n}$. The verifier oracle V is defines as

$$\mathsf{V}(x, \pi) = \begin{cases} 1 & \text{if } \exists w, r \text{ s.t. } \pi = \mathsf{P}(x, w, r)\ \wedge (x, w) \in R_{\mathcal{L}} \\ 0 & \text{otherwise.} \end{cases}$$

We denote the above oracle by $\mathsf{WI} = (\mathsf{P}, \mathsf{V})$.

Theorem 1. *[2]* WI *is a WI proof system oracle.*

Theorem 2. *[2] Let* O *be an oracle s.t. there exists a OWF* f^O *relative to* O*, and let* WI *be a WI proof system oracle. Then* f^O *is one-way relative to* O *and* WI.

The Construction of a NIZK

Fix an oracle O and a WI proof system oracle WI s.t. a OWF f^O exists relative to O and WI. If such a OWF exists, then a pseudorandom generator (PRG) $G^O : \{0,1\}^n \to \{0,1\}^{2n}$ can be constructed[6]. Let

$$\mathcal{L}' = \{(x, crs) \mid \exists\ w \text{ s.t. } (x, w) \in R_{\mathcal{L}}\ \vee\ crs = G^O(w)\}.$$

The construction of a NIZK $(\mathsf{Crs}, \mathsf{Prv}, \mathsf{Vrf},\ \mathsf{CrsSim}, \mathsf{PrvSim})$ is as follows:

Crs: $crs \leftarrow \mathsf{Crs}(1^n)$
Given 1^n, output uniformly chosen $crs \in \{0,1\}^{2n}$;

Prv: $\pi \leftarrow \mathsf{Prv}(crs, x, w)$
Given $crs \in \{0,1\}^{2n}$ and $x, w \in \{0,1\}^n$. Let $x' := (crs, x)$. Note that if $(x, w) \in R_{\mathcal{L}}$ then $(x', w) \in R_{\mathcal{L}'}$. Apply Levin reduction[9] to $(x', w) \in R_{\mathcal{L}'}$ to obtain $(\hat{x}, \hat{w}) \in R_{\mathcal{L}}$. Choose $r \leftarrow \{0,1\}^{|\hat{x}|}$ and output $\pi \leftarrow \mathsf{P}(\hat{x}, \hat{w}, r)$;

Vrf: $b \leftarrow \mathsf{Vrf}(crs, x, \pi)$
Given $crs \in \{0,1\}^{2n}$, $x \in \{0,1\}^n$ and $\pi \in \{0,1\}^{7n}$. Let $x' := (crs, x)$. Apply Levin reduction to $x' \in \mathcal{L}'$ to obtain $\hat{x} \in \mathcal{L}$. Output $b = \mathsf{V}(\hat{x}, \pi)$;

CrsSim: $(crs, \tau) \leftarrow \mathsf{CrsSim}(1^n)$
Given 1^n, choose $\tau \leftarrow \{0,1\}^n$, and output τ and $crs = G^O(\tau)$; and

PrvSim: $\pi \leftarrow \mathsf{PrvSim}(crs', x, \tau)$
Given $x, \tau \in \{0,1\}^n$ and $crs \in \{0,1\}^{2n}$. Let $x' := (crs, x)$. Note that $(x', \tau) \in R_{\mathcal{L}'}$. Apply Levin reduction to (x', τ) to obtain $(\hat{x}, \hat{w}) \in R_{\mathcal{L}}$. Choose $r \leftarrow \{0,1\}^{|\hat{x}|}$ and output $\pi \leftarrow \mathsf{P}(\hat{x}, \hat{w}, r)$.

Theorem 3. *[2] The above* $(\mathsf{Crs}, \mathsf{Prv}, \mathsf{Vrf}, \mathsf{CrsSim}, \mathsf{PrvSim})$ *is a NIZK with perfect complete, statistical sound and adaptive black box zero-knowledge properties.*

Definition 9 (augmented black box construction). *[2] There exists an* (fully) augmented black box construction *of a primitive Q based on a primitive P if there are PPTs G and S s.t.*

- *for any oracle O and WI proof system oracle WI for NP^O where O implements P, the oracle machine $G^{O,\mathsf{WI}}$ implements Q; and*
- *for any oracle O, WI proof system oracle WI for NP^O and adversary $\mathcal{A}$ that Q-breaks $G^{O,\mathsf{WI}}$, the adversary $S^{\mathcal{A},O,\mathsf{WI}}$ P-breaks O or breaks witness indistinguishability of WI.*

In [2] they showed construction and separation results to demonstrate the power of the augmented black box framework (namely, to ensure that the augmented black box framework encompasses the power of ZK). They showed an augmented black box construction of CCA-PKE based on CPA-PKE such as [10,12], and an augmented black box separation between OWF and KA such as [7].

Theorem 4. *[2] There is an augmented black box construction of a CCA-PKE based on a CPA-PKE.*

Theorem 5. *[2] There is no augmented black box construction of KA based on OWF.*

4 Simplified Proof System Oracle

4.1 Coin-Free Proof System Oracle

In this section we introduce a more simplified proof system oracle, which leads a more general result than [2], by simplifying the WI proof system oracle defined in Section 3. As in Section 3, we treat the $\mathtt{NP}$-complete language $\mathcal{L} =$

$\mathsf{CIRCUIT\text{-}SAT}^O$. We first give intuitions. Let WI be a WI proof system oracle defined in Section 3. In [2], they constructed a NIZK by making use of witness indistinguishability of WI. Recall that the prover oracle is a random function and we can flip a random coin in the construction of a NIZK. Hence we observe that these randomness are enough to construct a NIZK, and we can omit the random coin r from its interface, resulting a simpler prover oracle. We first introduce such simplified proof system oracle. Then we show that we can construct a WI proof system based on the simplified oracle in the black box manner.

We introduce the following *coin-free* proof system oracle.

Definition 10 (coin-free proof system oracle). *A pair* (P, V) *of oracles is a* coin-free proof system oracle *for a language* $\mathcal{L}$ *if it works as following;*

prover oracle: *The prover oracle* P *is a random function* $\mathsf{P} : \{0,1\}^{2n} \to \{0,1\}^{6n}$. *The input is parsed as pairs of the form* $(x, w) \in \{0,1\}^n \times \{0,1\}^n$. *Note that* P *does not check if* $(x, w) \in R_{\mathcal{L}}$.

verifier oracle: *The verifier oracle* V *is* $\mathsf{V} : \{0,1\}^{7n} \to \{0,1\}$. *The input is parsed as pairs of the form* $(x, \pi) \in \{0,1\}^n \times \{0,1\}^{6n}$. *The verifier oracle* V *is defined as*

$$\mathsf{V}(x, \pi) = \begin{cases} 1 & \text{if } \exists w \text{ s.t. } \pi = \mathsf{P}_n(x, w) \ \wedge (x, w) \in R_{\mathcal{L}} \\ 0 & \text{otherwise.} \end{cases}$$

It is clear that the above pair of oracles constitutes a proof system. We denote a coin-free proof system oracle by $\mathsf{CF} = (\mathsf{P}, \mathsf{V})$. We remark that coin-free proof system oracle is no longer witness indistinguishable, since an adversary, given a proof π, can decide which of witness w_0 or w_1 was used to generate π by making queries $\mathsf{P}(x, w_0)$ and $\mathsf{P}(x, w_1)$.

Construction of WI Proof System

We show that we can construct a WI proof system based on a coin-free proof system oracle. Our construction is similar to the construction of the NIZK in Section 3. We flip a random coin and apply Levin reduction to the instance in the construction. The key difference is an “extended” language. We introduce a language that includes randomness, and this randomness yields the witness indistinguishability. However it does not work simply adding a randomness in the new language (if so, the WI prover have to send the randomness itself to prove her knowledge about it). Thus we include a OWF in the new language and let the WI prover to prove her knowledge about the output of OWF.

Now we are ready to present the construction of a WI proof system. Let O be an oracle and $\mathsf{CF} = (\mathsf{P}, \mathsf{V})$ be a coin-free proof system oracle for $\mathcal{L}$ s.t. there exists an ϵ-OWF $f^O : \{0,1\}^n \to \{0,1\}^{2n}$ relative to O and CF. We can argue this due to Theorem 2 and the fact that a WI proof system implies a proof system generally. We define $\mathcal{L}'$ to be

$$\mathcal{L}' = \{(x, c) \mid \exists\ w, r \text{ s.t. } c = f^O(r) \wedge (x, w) \in R_{\mathcal{L}}\}.$$

We construct a WI proof system $(\mathsf{Prv}, \mathsf{Vrf})$ as follows:

Prv: $\hat{\pi} \leftarrow \mathsf{Prv}(x, w)$

Given $x, w \in \{0,1\}^n$. Choose $r \leftarrow \{0,1\}^n$, and compute $c = f^O(r)$. Let $x' := (x, c)$ and $w' := (w, r)$. Note that if $(x, w) \in R_{\mathcal{L}}$ then $(x', w') \in R_{\mathcal{L}'}$. Apply Levin reduction to $(x', w') \in R_{\mathcal{L}'}$ to obtain $(\hat{x}, \hat{w}) \in R_{\mathcal{L}}$. Compute $\pi = \mathsf{P}(\hat{x}, \hat{w})$, and output $\hat{\pi} := (c, \pi)$.

Vrf: $b \leftarrow \mathsf{Vrf}(x, \hat{\pi})$

Given $x \in \{0,1\}^n$ and $\hat{\pi} = (c, \pi) \in \{0,1\}^n \times \{0,1\}^{6n}$. Let $x' := (x, c)$. Apply Levin reduction to $x' \in \mathcal{L}'$ to obtain $\hat{x} \in \mathcal{L}$. Output $b = \mathsf{V}(\hat{x}, \pi)$.

Lemma 1. *The above* $(\mathsf{Crs}, \mathsf{Prv})$ *is a WI proof system for* $\mathcal{L} \in \mathtt{NP}^O$*.*

Proof. The perfect completeness property is immediate. We show that $(\mathsf{Prv}, \mathsf{Vrf})$ is perfectly sound. Considering the definition of $\mathcal{L}'$, we can apply Karp reduction [8] to an instance of $\mathcal{L}$ to obtain an instance of $\mathcal{L}'$. Thus if there exists an instance $(x, c) \notin \mathcal{L}'$ but applying Levin reduction results in an instance $\hat{x} \in \mathcal{L}$, then we can break the perfect soundness of CF.

We show the witness indistinguishability of $(\mathsf{Prv}, \mathsf{Vrf})$ following the idea of the proof of Theorem 1 in [2]. Let $\mathcal{A}$ be an adversary and q be a polynomial upper bound on the number of queries that $\mathcal{A}$ can make. We note that an adversary in the experiment $\mathtt{ExptWI}$ has oracle access to O and CF. We abuse notation to write $\mathcal{A}$ to denote $\mathcal{A}^{O,\mathsf{CF}}$. Without loss of generality, we assume that $\mathcal{A}$ outputs values (x, w_0, w_1) with $(x, w_0), (x, w_1) \in R_{\mathcal{L}}$. Then $\mathcal{A}$ is given a proof $\hat{\pi} = (c, \pi)$ for the instance (x, w_b) where $b \in \{0,1\}$ and tries to decide whether w_0 or w_1 was used to generate $\hat{\pi}$. In the following we first define an bad event s.t. $\mathcal{A}$ breaks the witness indistinguishability by accident and prove that such an event occurs only with negligible probability. Then we show that, assuming such event never happens, if $\mathcal{A}$ breaks the witness indistinguishability of $(\mathsf{Prv}, \mathsf{Vrf})$, then there exists an adversary that breaks the ϵ-security of f^O.

Let $\mathtt{Spoof}$ be the event that $\mathcal{A}$ makes a query $\mathsf{V}(x^*, \pi^*)$ returning 1, yet no query $\mathsf{P}(x^*, w^*)$ with $(x^*, w^*) \in R_{\mathcal{L}}$ was made previously. We prove that the probability $\mathtt{Spoof}$ occurs is negligible. At most 2^{2n} elements are uniformly distributed in the domain of P, and the size of the range is 2^{6n}. Although making a P-query reveals one point in the range, it tells nothing about other points since P is a random function. Thus the probability that $\mathcal{A}$ makes a query $\mathsf{V}(x^*, \pi^*)$ returning 1 yet π^* was not output by P previously is at most 2^{-4n}. Taking a union bound, the probability that $\mathtt{Spoof}$ occurs is at most $q \cdot 2^{-4n}$.

We prove that, assuming Spoof never occurs, if $(\mathsf{Prv}, \mathsf{Vrf})$ is not witness indistinguishable then there exists an adversary $\mathcal{A}'$ that breaks the ϵ-security of f^O. Since P is a random function, the adversary $\mathcal{A}$ that breaks the witness indistinguishability of $(\mathsf{Prv}, \mathsf{Vrf})$ makes the P-query resulting in $\hat{\pi}$. In the course of such computation, $\mathcal{A}$ has to find the pre-image of c as c is independent of the witness w_b. Thus an adversary $\mathcal{A}'$, given c, simulates $\mathcal{A}$ and outputs the pre-image of c, which contradicts the ϵ-security of f^O. Summing the above discussion, the probability that an adversary breaks witness indistinguishability of $(\mathsf{Prv}, \mathsf{Vrf})$ is at most $q \cdot 2^{-4n} + \epsilon$, which is negligible.

Corollary 1. *Let O be an oracle that implements a primitive Q,* WI *be a WI proof system oracle defined in Section 3 and* CF *be a coin-free proof system oracle.*

If there exists an augmented black box construction of a primitive P based on O and CF*, then there exists an augmented black box construction of P based on O and* WI*.*

We say an augmented black box construction that making use of a coin-free proof system oracle a *simplified* augmented black box construction.

4.2 Construction

We show that we can construct a CCA-PKE based on a CPA-PKE in the simplified augmented black box model. If we can construct a NIZK, then we can construct a CCA-PKE by following the Naor-Yung construction [10]. Due to the construction of the NIZK in Section 3 and Lemma 1, we can construct a NIZK based on a coin-free proof system oracle. Thus we can construct a CCA-PKE based on a CPA-PKE in the simplified augmented black box model.

Let O be an oracle that implements a CPA-PKE $(\mathsf{G}, \mathsf{E}, \mathsf{D})$ and $\mathsf{CF} = (\mathsf{P}, \mathsf{V})$ be a coin-free proof system oracle. As shown in the previous discussion, we can construct a NIZK $(\mathsf{Crs}, \mathsf{Prv}, \mathsf{Vrf}, \ \mathsf{CrsSim}, \mathsf{PrvSim})$ in the simplified augmented black box model. Moreover we can translate $(\mathsf{Prv}, \mathsf{Vrf})$ into a *simulation sound* NIZK[12] $(\mathsf{Prv}_{ssZK}, \ \mathsf{Vrf}_{ssZK})$ for a language

$$\mathcal{L}' = \{(c_0, c_1, pk_0, pk_1) \mid \exists \ m, r_0, r_1 \text{ s.t.} c_0 = \mathsf{E}^{O}_{pk_0}(m, r_0) \ \wedge \ c_1 = \mathsf{E}^{O}_{pk_1}(m, r_1)\}.$$

Lemma 2. *Let O be an oracle that implements a CPA-PKE and* CF *be a coin-free proof system oracle. We can construct a CCA-PKE based on O and* CF *in the simplified augmented black box model.*

4.3 Separation Result

We show that there is no construction of KA with perfect completeness based on OWF in the simplified augmented black box model. As stated in Section 1, one of the motivation of our work is to simplify security proofs in the augmented black box framework. However, in the proof of [2], they did not make use of the witness indistinguishability (i.e., the random coin r) of the proof system oracle, resulting the same proof logic in the simplified augmented black box framework. Thus, we present the overview of the construction of the adversary and the separation proof in [2].

Let O be a uniformly chosen random oracle and $\mathsf{WI} = (\mathsf{P}, \mathsf{V})$ be a WI proof system oracle, where there exists a OWF relative to O and WI. Let (A, B) be an augmented black box construction of KA with perfect completeness based on O and WI. Note that it is sufficient to consider a 1-bit KA construction. Let q be the running time of (A, B) (i.e., the number of queries that A and B can make to O and WI are restricted by q in total). Given security parameter 1^n, A and B interact each other, resulting a transcript T and a shared key k. Let r_A be a random tape of A and $Q(A)$ be a set of query/answer pair that A makes to O and WI, in the execution of KA. A pair $(r_A, Q(A))$ is said a *view* of A. Similary the set of query/answer pairs that B makes in the execution is denoted by $Q(B)$. Note that $|Q(A) \cup Q(B)| \leq q$.

The Adversary

In [2] they showed an adversary E that breaks KA but cannot break OWF, where E is computationally unbounded but makes at most polynomially many queries to those oracles. Given 1^n and T, E simulates the view of A at first, then learns about O and WI based on the simulation. More formally, E works as follows. Let $Q(E)$ be a set of query/answer pair and K be a set of "key candidates." First E sets $Q(E) := \phi$ and $K := \phi$. Then E repeats the following $2q + 1$ times:

Simulation Phase: E simulates the view of A that is consistent with T and $Q(E)$. Let $\hat{Q}(A)$ denote the set of the simulated query/answer pairs. Note that $\hat{Q}(A)$ is not necessary consistent with the real oracles O and WI. Following the simulated view, E outputs a key $\hat{k}$ and sets $K := K \cup \{\hat{k}\}$.

Update Phase: E makes all queries in $\hat{Q}(A) \setminus Q(E)$ to O and WI, and adds the resulting query/answer pairs to $Q(E)$.

After $2q + 1$ iterations, E outputs the majority of K as a simulated shared key.

The Separation Proof

In [2] they showed that the above adversary breaks KA, but cannot break OWF. We give the overview of the proof. Although in [2], they dealt with some subtleties, we ignore them and focus only on essential part of the proof.

Since O is a random oracle, no adversary that makes at most polynomially many queries cannot break OWF based on O. The reason that E can break KA is as follows. They defined an event that $\hat{Q}(A)$ disagrees with $Q(A) \cup Q(B)$ on the answer to some O-, P- or V-query. (Note that they defined other two events, however they proved that these events are essentially the same as the above event.) They showed that if this event occurs in an iteration then E learns at least one query/answer pair in $Q(A) \cup Q(B)$, otherwise she adds a correct key to K. Since $|Q(A) \cup Q(B)| \leq q$, this event occurs at most q times. Thus at least $q+1$ keys in K are correct keys in the final step of the attack, resulting a correct shared key.

Separation

We can construct the same adversary in the simplified augmented black box framework by simply replacing a WI proof system oracle with a coin-free proof system oracle. The event defined above occurs because O and P are random oracles, i.e., the adversary cannot predict their output. Since a prover oracle of a coin-free proof system oracle is still random function, the above adversary breaks the constructed KA. To sum up the above, we obtain the following lemma:

Lemma 3. *Let O be a uniformly chosen random oracle and CF be a coin-free proof system oracle s.t. a OWF exists relative to O and CF. There is no simplified augmented black box construction of KA with perfect completeness based on the OWF.*

5 Conclusion

In this paper we introduced coin-free proof system oracle, a more simplified one, and showed the same construction and separation results as in [2]. Thus when we apply the augmented black box framework to some black box construction or separation proof, we become to be able to prove it in more simplified and general condition.

There are open questions still remain. One of such question is to show other construction or separation results in the simplified black box model (especially to known black box separation results). Focusing on specific topic, the construction of the NIZK is based on a proof system oracle for $\mathtt{NP}$-complete language, which seems too strong. It is still debatable whether we can construct a NIZK based on a proof system oracle for more restricted language.

References

1. Boneh, D., Papakonstantinou, P., Rackoff, C., Vahlis, Y., Waters, B.: On the impossibility of basing identity based encryption on trapdoor permutations. In: Proceedings of the 2008 49th Annual IEEE Symposium on Foundations of Computer Science. pp. 283–292. FOCS '08, IEEE Computer Society, Washington, DC, USA (2008). https://doi.org/10.1109/FOCS.2008.67
2. Brakerski, Z., Katz, J., Segev, G., Yerukhimovich, A.: Limits on the power of zero-knowledge proofs in cryptographic constructions. In: Theory of Cryptography - 8th Theory of Cryptography Conference, TCC 2011. Lecture Notes in Computer Science, vol. 6597, p. 559. Springer (2011). https://doi.org/10.1007/978-3-642-19571-6_34
3. Diffie, W., Hellman, M.: New directions in cryptography. Information Theory, IEEE Transactions on **22**, 644 – 654 (1976). https://doi.org/10.1109/TIT.1976.1055638
4. Feige, U., Shamir, A.: Witness indistinguishable and witness hiding protocols. In: Proceedings of the Twenty-second Annual ACM Symposium on Theory of Computing. pp. 416–426. STOC '90, ACM, New York, NY, USA (1990). https://doi.org/10.1145/100216.100272
5. Goldwasser, S., Micali, S., Rackoff, C.: The knowledge complexity of interactive proof-systems. In: Proceedings of the Seventeenth Annual ACM Symposium on Theory of Computing. pp. 291–304. STOC '85, ACM, New York, NY, USA (1985). https://doi.org/10.1145/22145.22178
6. Håstad, J., Impagliazzo, R., Levin, L.A., Luby, M.: A pseudorandom generator from any one-way function. SIAM J. Comput. **28**(4), 1364–1396 (1999). https://doi.org/10.1137/S0097539793244708
7. Impagliazzo, R., Rudich, S.: Limits on the provable consequences of one-way permutations. In: Proceedings of the Twenty-first Annual ACM Symposium on Theory of Computing. pp. 44–61. STOC '89, ACM, New York, NY, USA (1989). https://doi.org/10.1145/73007.73012
8. Karp, R.M.: Reducibility among Combinatorial Problems, pp. 85–103. Springer US, Boston, MA (1972). https://doi.org/10.1007/978-1-4684-2001-2_9
9. Levin, L.A.: Universal sequential search problems. Problems of Information Transmission **9**(3), 265–266 (1973)

10. Naor, M., Yung, M.: Public-key cryptosystems provably secure against chosen ciphertext attacks. In: Proceedings of the Twenty-second Annual ACM Symposium on Theory of Computing. pp. 427–437. STOC '90, ACM, New York, NY, USA (1990). https://doi.org/10.1145/100216.100273
11. Reingold, O., Trevisan, L., Vadhan, S.: Notions of reducibility between cryptographic primitives. In: Naor, M. (ed.) Theory of Cryptography. pp. 1–20. Springer Berlin Heidelberg, Berlin, Heidelberg (2004)
12. Sahai, A.: Non-malleable non-interactive zero knowledge and adaptive chosen-ciphertext security. In: Proceedings of the 40th Annual Symposium on Foundations of Computer Science. pp. 543–. FOCS '99, IEEE Computer Society, Washington, DC, USA (1999)
13. Yao, A.C.: Theory and application of trapdoor functions. In: Proceedings of the 23rd Annual Symposium on Foundations of Computer Science. pp. 80–91. SFCS '82, IEEE Computer Society, Washington, DC, USA (1982). https://doi.org/10.1109/SFCS.1982.95