Reset Indifferentiability and its Consequences

Paul Baecher[1] Christina Brzuska[2] Arno Mittelbach[1]

[1]Darmstadt University of Technology, Germany
www.cryptoplexity.de
[2]Tel-Aviv University, Israel
www.christinabrzuska.de
baecher@cs.tu-darmstadt.de brzuska@post.tau.ac.il arno.mittelbach@cased.de

Abstract. The equivalence of the random-oracle model and the ideal-cipher model has been studied in a long series of results. Holenstein, Künzler, and Tessaro (STOC, 2011) have recently completed the picture positively, assuming that, roughly speaking, equivalence is indifferentiability from each other. However, under the stronger notion of reset indifferentiability this picture changes significantly, as Demay et al. (EUROCRYPT, 2013) and Luykx et al. (ePrint, 2012) demonstrate.

We complement these latter works in several ways. First, we show that any simulator satisfying the reset indifferentiability notion must be stateless and pseudo deterministic. Using this characterization we show that, with respect to reset indifferentiability, two ideal models are either equivalent or incomparable, that is, a model cannot be strictly stronger than the other model. In the case of the random-oracle model and the ideal-cipher model, this implies that the two are incomparable. Finally, we examine weaker notions of reset indifferentiability that, while not being able to allow composition in general, allow composition for a large class of multi-stage games. Here we show that the seemingly much weaker notion of 1-reset indifferentiability proposed by Luykx et al. is equivalent to reset indifferentiability. Hence, the impossibility of coming up with a reset-indifferentiable construction transfers to the setting where only one reset is permitted, thereby re-opening the quest for an achievable and meaningful notion in between the two variants.

Contents

1 Introduction **3**

2 Preliminaries **7**
2.1 Indifferentiability 8

3 Pseudo-deterministic Stateless Simulators for Indifferentiability **9**
3.1 Multi-stage Indifferentiability 10
3.2 Pseudo-Deterministic Algorithms 11

4 The Random Oracle and Ideal Cipher Model are Incomparable **13**
4.1 The Duality Lemma for Multi-Stage Indifferentiability 14

5 Single versus Multi-Reset **15**

6 Acknowledgements **18**

A Missing Proofs for Section 3 **21**
A.1 Proof of Lemma 3.5 21

1 Introduction

Idealized Models. The standard approach to cryptographic security is to reduce the security of a scheme to a (hopefully) well-studied algebraic or combinatorial complexity assumption. Unfortunately, a large number of cryptographic schemes does not admit a security reduction in the standard model. In these cases, the community often resorts to an idealized model, where we can sometimes obtain a proof of security. It is, of course, highly controversial whether or not proofs in idealized models are acceptable, but there is a tendency to prefer an analysis in an idealized model over the utter absence of any proof at all—in particular, when one is concerned with schemes that are widely deployed in practice [BR94, BR96, BRSS10].

Arguably the most popular model of this kind is the random-oracle model (ROM) where all parties have oracle access to a public, randomly chosen function [BR93]. Somewhat related is the ideal-cipher model (ICM) which gives all parties oracle access to a public, randomly chosen (keyed) blockcipher [Sha49]. Knowing that there is a close relation between pseudorandom functions and pseudorandom permutations—namely existential equivalence—one could suspect that the random-oracle model and the ideal-cipher model are equivalent, too. However, formalizing the notion of equivalence is delicate and so are the proofs.

Equivalence of the ROM and ICM under Indifferentiability. Maurer, Renner and Holenstein [MRH04] introduced the concept of indifferentiability, which since then has been regarded as the prevalent and actually only notion of equivalence between ideal primitives. A construction G^π with access to some primitive π is called indifferentiable from another ideal primitive Π, if there is a simulator $\mathcal{S}$ such that the construction G^π implements an oracle that is indistinguishable from Π, even if the distinguisher $\mathcal{D}$ additionally gets access to π. Now, demanding the distinguisher $\mathcal{D}$ to distinguish (G^π, π) from Π is of little sense. Additionally to the oracle Π, the distinguisher gets access to the simulator $\mathcal{S}$ which tries to emulate π's behavior consistently with Π. Thus, the distinguisher tries to distinguish the pair of oracles (G^π, π) from the pair of oracles $(\Pi, \mathcal{S}^\Pi)$.

In the case of the ideal-cipher model and the random-oracle model, considerable effort has led to a proof of equivalence [CDMP05, CPS08, HKT11] under indifferentiability. The reason why indifferentiability was considered a suitable notion of equivalence is the appealing composition theorem established by Maurer et al. [MRH04]. Namely, they transform any reductionist argument in the presence of the ideal primitive Π into a proof that relies on the existence of π only. Their theorem, thus, transforms a reduction $\mathcal{R}$ into a reduction $\mathcal{R}'$, where the latter locally implements a single copy of the simulator $\mathcal{S}$. Jumping ahead, it will turn out that in this step, they rely on an implicit assumption.

Multi-Stage Adversaries. Ristenpart et al. [RSS11] were the first to point out scenarios where indifferentiability of G^π from Π was not sufficient to replace Π by G^π. Their counterexamples involve adversaries that run in multiple stages, i.e., an adversary $\mathcal{A}$ consists

of two or more sub-adversaries, say $\mathcal{A} = (\mathcal{A}_1, \mathcal{A}_2)$, that do not share state (or at least not arbitrary state). Now, a reduction $\mathcal{R}$ that reduces to such a *multi-stage game* also needs to be split into two parts $(\mathcal{R}_1, \mathcal{R}_2)$ where the same restriction upon the sharing of state applies. Hence, for the composition theorem by Maurer et al., each part of the reduction $\mathcal{R}_1$ and $\mathcal{R}_2$ needs to implement its own, independent copy of the simulator $\mathcal{S}$. However, in this case, the two copies of the simulator will not necessarily behave in the same way as opposed to the "real" primitive π which is, roughly, what makes the composition theorem collapse in the setting of multi-stage games.

Curiously, their composition holds in the presence of *strong*, colluding adversaries, while it does not in the setting of weaker, non-colluding ones. Usually in cryptography, a conservative approach corresponds to considering the strongest possible adversary, as a primitive that is secure against a strong adversary is also secure against a weaker adversary. However, the indifferentiability composition theorem is not, by itself, a security model or a proof of security. Instead, it is a tool to transform any proof in a security model in the presence of one ideal primitive into a security proof in the same security model in the presence of another ideal primitive. Hence, one tries to cover any type of security model, which, in particular, includes security models where stage-sharing adversaries can mount trivial attacks. And thus, a conservative approach in the setting of indifferentiability demands including also weaker, namely non-colluding state-sharing adversaries. Technically, the composition theorem is *harder* to prove for *weaker* adversaries, because it transforms an adversary of one type into another adversary of the same type. Considering a stronger adversary corresponds to a stronger assumption in the theorem, but also to a harder statement to prove, and vice versa for weaker adversaries.

One might hope that the distinction is of technical interest only. Unfortunately, as we argue, in basically all real-life scenarios, we need to consider multi-stage adversaries. Ristenpart et al. give several examples of multi-stage games for notions such as deterministic encryption [BBO07, BBN$^+$09], key-dependent message security [BRS02], related-key attacks [BK03], and non-malleable hash functions [BCFW09]. On the other hand, many classical notions of security seem inherently single stage: IND-CPA or IND-CCA security for encryption, or signature schemes which are existentially unforgeable under (adaptive) chosen-message attacks. However, any classical definition of security becomes multi staged if it is augmented with a leakage oracle. The reason is that, in the random-oracle model, every party should have access to the random oracle. In particular, this includes the leakage oracle and the adversarially specified leakage function, resulting in an implicit second stage [DKW11]. Hence, whenever side-channel attacks are reflected in a model, adversaries act at least in two stages—and for real-life applications, we cannot discard side-channel attacks.

In order to cope with the new challenge of multi-stage adversaries, Ristenpart et al. put forward a strengthened notion called *reset indifferentiability*. Roughly speaking, in this game, the distinguisher may reset the simulator's internal state between any two queries. Returning to ROM/ICM equivalence, an inspection of the simulators defined in [CDMP05] and [HKT11] (as well as [CPS08], for that matter) reveals that their behavior varies sub-

stantially with their state and, thus, they are not reset indifferentiable.

Equivalence of the ROM and ICM under Reset Indifferentiability. As plain indifferentiability is not sufficient to argue that two primitives are equivalent, the question regarding the ideal-cipher model and the random-oracle model is, thus, again open. Building on first negative results from [RSS11], the authors of [DGHM13, LAMP12] have recently shown that reset-indifferentiable constructions cannot be built via domain extension, thereby ruling out constructions from ideal ciphers that are reset indifferentiable from a random oracle; note that random oracles are usually perceived as having an infinite domain while ideal ciphers have a finite domain. With this result at hand, we thus know that ideal ciphers cannot be used to obtain random oracles via a reset-indifferentiable construction, but it might still be possible to construct an ideal cipher from a random oracle, i.e., either the two models are entirely incomparable, or the random-oracle model is strictly stronger.

We rule out such a possibility. Our so-called duality lemma establishes that if there is no construction G_1^{π} that is reset indifferentiable from primitive Π, then also vice versa, there is no construction G_2^{Π} that is reset indifferentiable from primitive π. Hence, our theorem complements the results by Demay et al. and Luykx et al. [DGHM13, LAMP12] showing that there can also not be a domain-shrinking construction.

Proving that, according to plain indifferentiability, the ICM and ROM are equivalent had been a serious challenge and finally involved a Feistel network with many rounds. A Feistel network is a domain-doubling construction, and is thus ruled out by the previous impossibility results. The few leverages that remain to bypass the current impossibility results possibly require quite new techniques. Firstly, it might still be possible to build a construction that is neither domain shrinking, nor domain extending. However, as we will see later, that means settling either direction (RO from IC and vice versa) simultaneously, and this might be quite challenging. The second leverage is a distinction that has been irrelevant in most works in the area of indifferentiability so far and that we would like to point out. Namely, strong indifferentiability requires the simulator $\mathcal{S}$ to work for any distinguisher $\mathcal{D}$, while weak indifferentiability only demands that for every $\mathcal{D}$, there exists a good simulator $\mathcal{S}$. Known constructions are usually strongly indifferentiable, while most existing impossibility results rule out even weakly-indifferentiable constructions. In contrast, we do not rule out weakly-indifferentiable constructions. It would be interesting to see techniques that make non-black-box use of the distinguisher $\mathcal{D}$ and establish a reset-indifferentiable construction that is domain shrinking.

Notions between indifferentiability and reset indifferentiability. From the current state-of-the-art, there are two ways to proceed: firstly, we can develop new techniques to exploit the few remaining leverages left to bypass the existing impossibility results. Secondly, we might weaken the notion of reset indifferentiability as introduced by Ristenpart et al., to a notion that is achievable by constructions and which is sufficient for a subclass of multi-stage games.

Demay et al. [DGHM13] introduce resource-restricted indifferentiability where adver-

saries may share a limited amount of state. If a certain amount s of shared state is allowed, then their impossibility result shows that a reset-indifferentiable construction cannot extend the domain by more than $s + \lceil \log(s) \rceil$ bits. Maybe the additional bits allow to bypass the impossibility results more easily, as proving domain extension by a few bits might be easier than requiring equality of the domain sizes—however, in this setting, the composition result accounts for a certain class of games only.

Another approach that has been put forward by Luykx et al. [LAMP12] is to reduce the number of resets. Indeed, allowing for a polynomial number of resets/stages seems to be an overkill, as some games such as the security model for deterministic encryption [BBO07, BBN$^+$09] and also certain forms of leakage require a constant number of adversarial stages only. To this end, Luykx et al. propose the notion of *single-reset indifferentiability* where a distinguisher can make a single reset call only; naturally, a construction that is single reset indifferentiable would be sufficient in any security game consisting of exactly two distinct adversarial stages such as deterministic encryption. Analogously, one can define n-reset indifferentiability for $n + 1$ adversarial stages.

However, as we prove, single-reset indifferentiability is already equivalent to full-reset indifferentiability and so are all notions of n-reset indifferentiability. Hence, reducing the number of allowed reset queries does not help us to establish composition results for a restricted class of games. Thus, if a general indifferentiability result is indeed impossible, then it is a curious open question how to cope with the uncomfortable situation. It might be possible to establish indifferentiability results and composition theorems for a class of games that is restricted in another way than by the number of queries. Indeed, it would be interesting to see how such a class could look like and whether there are games for which, in general, finding a suitable, indifferentiable construction is impossible.

Summary of our Contributions. We first introduce the notion of *pseudo-deterministic* algorithms, which captures, that a probabilistic algorithm almost always returns the same answer on the same queries and thus shares many properties with deterministic algorithms. Essentially, a probabilistic (and possibly stateful) algorithm $\mathcal{A}$ is called pseudo deterministic, if no efficient distinguisher with black-box access to $\mathcal{A}$ can make $\mathcal{A}$ return two different answers on the same input. This notion of pseudo determinism can be seen as an average-case version of the pseudo-deterministic algorithms that were recently introduced by Goldreich, Goldwasser, and Ron [GGR12]. While they require probabilism to be hard to detect on any input, we only require indistinguishability for efficiently-generatable inputs, on the average. As stressed by Goldreich et al. [GGR12], pseudo-deterministic algorithms are practically as useful as deterministic algorithms, but they are also easier to construct—which we indeed exploit in our paper.

We will show in Section 3 that simulators for reset indifferentiability need to be stateless and pseudo deterministic. Simplifying pseudo determinism to determinism for the moment, this allows us to establish what we call the duality lemma. Perhaps surprisingly, it states that, with respect to reset indifferentiability, two idealized models are either *equivalent* or *incomparable*. The reason is that a deterministic and stateless simulator can act as a

construction and vice versa. Consequently, in order to prove equivalence in terms of reset indifferentiability, this lemma makes it sufficient to prove the "easier" direction, whichever this might be. In turn, for impossibility results, one might use this as a tool to prove impossibility more easily. In fact, we use the duality lemma to establish that not only domain-extending constructions are impossible, but also domain-shrinking constructions (Section 4) thereby complementing the results of [DGHM13]. Note that the duality lemma covers strong indifferentiability, leaving non-black-box use of the distinguisher as a potential leverage to bypass this impossibility.

The recently proposed [LAMP12] notion of single-reset indifferentiability intends to define a notion of indifferentiability that is easier to achieve and simultaneously covers an interesting class of multi-stage games that has two adversary stages only. Interestingly, as we establish, restricting the number of resets does not yield a weaker notion of equivalence. We prove that single- (and n-) reset indifferentiability is equivalent to reset indifferentiability (Section 5). Maybe surprisingly, our proof does not rely on a hybrid argument; instead, we establish a tight reduction that merely reduces the distinguisher's advantage by a factor of 2.

2 Preliminaries

For a natural number $n \in \mathbb{N}$ we denote by $\{0,1\}^n$ the set of all bit strings of length n. By $\{0,1\}^*$ we denote the set of all bit strings of finite length. As usual $|\mathcal{M}|$ denotes the cardinality of a set $\mathcal{M}$ and logarithms are to base 2. For some probabilistic algorithm $\mathcal{A}$ and input x we denote by $\mathcal{A}(x;R)$ the output of $\mathcal{A}$ on x using randomness R. Throughout this paper we assume that λ is a security parameter (if not explicitly given then implicitly assumed) and that algorithms (resp., Turing machines) run in polynomial time with respect to λ.

In this paper we consider random oracles and ideal ciphers (defined below) which we will collectively refer to as *ideal primitives*. Although we present most of the results directly for ideal ciphers and random oracles, the following more general notion of ideal primitives allows us to generalize some of our results:

Definition 2.1. *An* ideal primitive Π_λ *is a distribution on functions indexed by the security parameter λ. For some algorithm $\mathcal{A}$, security parameter λ and ideal primitive Π_λ we say that $\mathcal{A}$* has access to Π *if $\mathcal{A}$ has oracle access to a function f chosen from the distribution Π_λ.*

We simply write Π, i.e., omit the security parameter, if it is clear from the context.

Remark. We will usually encounter only single instances of an ideal primitive Π at a time. Unless stated otherwise, if multiple parties have access to Π, then we implicitly assume that the corresponding function f was chosen from the distribution Π using the same randomness for all parties, i.e., all parties have oracle access to the same function f.

RANDOM ORACLES AND IDEAL CIPHERS. A random oracle $(\mathcal{R}_{\ell,m})_\lambda$ is the uniform distribution on all functions mapping $\{0,1\}^\ell$ to $\{0,1\}^m$ with $\ell := \ell(\lambda)$ and $m := m(\lambda)$. An ideal cipher $(\mathcal{E}_{k,n})_\lambda$ is the uniform distribution on all keyed permutations of the form $\{0,1\}^k \times \{0,1\}^n \to \{0,1\}^n$ with $k := k(\lambda)$ and $n := n(\lambda)$. That is, for a cipher in the support of $(\mathcal{E}_{k,n})_\lambda$ each key $\kappa \in \{0,1\}^k$ describes a random (independent) permutation $\mathcal{E}_{k,n}(\kappa,\cdot) : \{0,1\}^n \to \{0,1\}^n$. By abuse of notation, the term random oracle (resp., ideal cipher) also refers to a specific instance chosen from the respective distribution.

KEYED VS. UNKEYED CIPHERS. The ideal-cipher model has either been considered as a public *unkeyed* permutation or as a public *keyed* permutation. We present our results in the keyed setting since we feel that the ideal cipher-model is usually perceived in this way. However, we want to point out that the results are equally valid for the unkeyed setting because our proofs do not rely on the presence of a key.

Independently of this, one might be tempted to argue that the settings are interchangeable since we know, for example, constructions of a keyed permutation from an ideal public permutation (Even and Mansour, [EM97]). Note though, that in order to make this argument work, one needs to show that these constructions are reset indifferentiable. However, the construction by Even and Mansour is a domain extender where the key size is twice the message size and we rule out reset indifferentiability for such extending constructions in Section 4. We note that it is an interesting open problem whether or not such (reset-) indifferentiable non-extending transformations exist.

2.1 Indifferentiability

Let us now recall the indifferentiability notion of Maurer et al. [MRH04] in the version by Coron et al. [CDMP05] who replace *random systems* by oracle Turing machines (resp., ideal primitives). Since we are concerned with different types of indifferentiability, we will sometimes use the term *plain* indifferentiability when referring to this original notion of indifferentiability.

Definition 2.2. *A Turing machine G with black-box access to an ideal primitive π is* strongly indifferentiable *from an ideal primitive Π if there exists a simulator $\mathcal{S}^\Pi$, such that for any distinguisher $\mathcal{D}$ there exists negligible function* negl*, such that:*

$$\left|\text{Prob}\left[\mathcal{D}^{G^\pi,\pi}(1^\lambda)=1\right] - \text{Prob}\left[\mathcal{D}^{\Pi,\mathcal{S}^\Pi}(1^\lambda)=1\right]\right| \leq \mathsf{negl}(\lambda) \tag{1}$$

We say that the construction is weakly indifferentiable *if for any $\mathcal{D}$ there exists a simulator $\mathcal{S}$ such that* (1) *holds.*

We will use the term *real world* to denote that the distinguisher $\mathcal{D}$ talks to the construction G^π and the primitive π, whereas in the *ideal world*, the distinguisher $\mathcal{D}$ talks to the "target" primitive Π and simulator $\mathcal{S}^\Pi$. The goal of the distinguisher is to determine which of the two *pairs* of oracles he is talking to. Towards this goal, the distinguisher $\mathcal{D}$ queries its two oracles, of which one is called the honest interface h which is either G^π (in

the real world) or Π (in the ideal world). The other oracle is called the adversarial interface a and corresponds to either π (real world) or $\mathcal{S}^{\Pi}$ (ideal world). Thus, $(\mathsf{h}, \mathsf{a}) := (G^{\pi}, \pi)$ if distinguisher $\mathcal{D}$ is in the real world and $(\mathsf{h}, \mathsf{a}) := (\Pi, \mathcal{S}^{\Pi})$ if it is in the ideal world. The names h (honest) and a (adversarial) are in the style of [RSS11] and suggestive: an honest party uses a construction as the designer intended; an adversary could, however, use the underlying building blocks to gain an advantage.

Reset Indifferentiability. Ristenpart et al. show [RSS11] that, in general, we cannot securely replace a primitive Π by a construction G^{π} from primitive π, if the construction is indifferentiable only. Instead, G^{π} needs to be (weakly) *reset* indifferentiable from Π which extends the original indifferentiability definition by giving the distinguisher the power to reset the simulator at arbitrary times:

Definition 2.3. *Let the setup be as in Definition 2.2. An oracle Turing machine G^{π} is called* strongly *(resp.* weakly*)* reset indifferentiable *from ideal primitive Π if the distinguisher $\mathcal{D}$ can reset the simulator $\mathcal{S}$ to its initial state arbitrarily many times during the respective experiment.*

For reset indifferentiability the adversarial interface a in the real world simply ignores reset queries. Reset indifferentiability now allows composition in arbitrary games and not only in single-stage games, as does the original indifferentiability notion [RSS11, MRH04].

3 Pseudo-deterministic Stateless Simulators for Indifferentiability

Recall that the composition theorem by Maurer et al. [MRH04] for plain indifferentiability holds for single-stage adversaries only. Their theorem says that if (i) the construction G^{π} is indifferentiable from the ideal primitive Π and if (ii) there is a reduction $\mathcal{R}$ that transforms a successful adversary $\mathcal{A}$ against some notion of security into an adversary $\mathcal{R}^{\mathcal{A}}$ against a single-stage game in the presence of the ideal primitive Π, then also in the presence of the construction G^{π} there is a reduction $\mathcal{R}'$ that transforms a successful adversary $\mathcal{A}$ into an adversary $\mathcal{R}'^{\mathcal{A}}$ against the single-stage game.

In order to prove a general composition theorem, Ristenpart et al. [RSS11] strengthen the notion of indifferentiability to account for the different stages of the adversary. They introduce the notion of (weak) reset indifferentiability and prove that the aforementioned theorem works for arbitrary games, if the construction G^{π} is reset indifferentiable from the ideal primitive Π. In contrast to plain indifferentiability, here, the distinguisher gets extra powers, namely to reset the simulator at arbitrary times. Ristenpart et al. [RSS11] and Demay et al. [DGHM13] remark that reset indifferentiability is equivalent to plain indifferentiability with stateless simulators. Intuitively, this follows from the observation that the distinguisher in the reset indifferentiability game can simply reset the simulator after each query it asks. We believe that, albeit equivalent, stateless simulators are often easier to

handle than reset-resistant simulators and thus explicitly introduce indifferentiability with stateless simulators as *multi-stage indifferentiability* and then prove that it is equivalent to reset indifferentiability.

In Subsection 3.2, we prove that strong multi-stage indifferentiability implies that the simulators are also *pseudo deterministic*, a notion that we put forward in this section. Relative to a random oracle or an ideal cipher, we show how to derandomize pseudo-deterministic simulators, if the simulators are allowed to depend on the number of queries made by the distinguisher.

3.1 Multi-stage Indifferentiability

A stateless interactive algorithm is an algorithm whose behavior is statistically independent from the call/answer history of the algorithm. We now prove that indifferentiability with stateless simulators is equivalent to reset indifferentiability.

Definition 3.1. *A construction G with black-box access to primitive π is* strongly multi stage indifferentiable *from primitive Π if there exists a stateless probabilistic polynomial-time simulator $\mathcal{S}$ (with access to Π), such that for any probabilistic polynomial-time distinguisher $\mathcal{D}$ there exists negligible function* negl *such that:*

$$\left|\text{Prob}\left[\mathcal{D}^{G^{\pi},\pi}(1^{\lambda})=1\right]-\text{Prob}\left[\mathcal{D}^{\Pi,\mathcal{S}^{\Pi}}(1^{\lambda})=1\right]\right|\leq \mathsf{negl}(\lambda) \tag{2}$$

We say that a construction G^{π} is weakly multi stage indifferentiable *from Π if for any probabilistic polynomial-time distinguisher $\mathcal{D}$ there exists a stateless probabilistic polynomial-time simulator $\mathcal{S}$ such that* (2) *holds.*

Lemma 3.2. *A construction G with black-box access to primitive π is weakly (resp., strongly) multi stage indifferentiable from primitive Π if and only if G is weakly (resp., strongly) reset indifferentiable from primitive Π.*

Proof. First note that any stateless simulator is, naturally, indifferent to resets and thus multi-stage indifferentiability implies reset indifferentiability. Moreover, strong reset indifferentiability implies strong multi-stage indifferentiability since the simulator for reset indifferentiability must work for any distinguisher, in particular for those which reset after each query. Hence this stateful simulator can be simply initialized and run by a stateless simulator (the stateless simulator does this for each query it receives).

We now prove the remaining relation, i.e., that weak reset indifferentiability implies weak multi-stage indifferentiability. Assume that reset indifferentiability holds and consider an arbitrary distinguisher $\mathcal{D}$ in the multi-stage indifferentiability game. From this we construct a distinguisher $\mathcal{D}'$ for the reset indifferentiability game which runs $\mathcal{D}$ and sends a reset query to its adversarial a-interface after every a-query issued by $\mathcal{D}$. Let $\mathcal{S}'$ be the simulator for $\mathcal{D}'$ guaranteed to exist by reset indifferentiability. We construct a stateless simulator $\mathcal{S}$ for multi-stage indifferentiability which simply runs (the stateful) $\mathcal{S}'$ and resets its own state

after each query. Now the following equations hold for $b \in \{0,1\}$:

$$\text{Prob}\left[\mathcal{D}'^{\Pi,\mathcal{S}'}(1^\lambda) = b\right] = \text{Prob}\left[\mathcal{D}'^{\Pi,\mathcal{S}}(1^\lambda) = b\right] = \text{Prob}\left[\mathcal{D}^{\Pi,\mathcal{S}}(1^\lambda) = b\right].$$

Thus, if equation (2) holds for $(\mathcal{D}', \mathcal{S}')$, then it holds equally for $(\mathcal{D}, \mathcal{S})$. □

3.2 Pseudo-Deterministic Algorithms

Our notion of *pseudo-deterministic* algorithms intuitively captures that no distinguisher can query the algorithm on an input such that it returns something different from the most likely output. That is, the adversary wins if in its set of input/output pairs to the algorithm there is a query for which the algorithm did not return the most likely response. We also introduce a weak notion of this property, where we call $\mathcal{A}$ pseudo deterministic for a specific distinguisher if the probability of the distinguisher winning in the above experiment is negligible.

Our notion of pseudo determinism can be seen as an average-case version of the pseudo-deterministic algorithms as recently introduced by Goldreich et al. [GGR12]. While they require probabilism to be hard to detect on any input, we only require indistinguishability for efficiently generatable inputs, on average.

Definition 3.3. *Let λ be a security parameter and $\mathcal{A}^\mathcal{O}$ a stateless probabilistic polynomial-time oracle Turing machine with access to some oracle $\mathcal{O}$. Let $L[\mathcal{D}, \mathcal{A}, \mathcal{O}]$ denote the induced set of input/output pairs (x, y) of $\mathcal{A}^\mathcal{O}$ when queried arbitrarily many times by the distinguisher $\mathcal{D}$, where $\mathcal{A}$ uses fresh coins in each run. We say that $\mathcal{A}^\mathcal{O}$ is* pseudo deterministic *if for all probabilistic polynomial-time distinguishers $\mathcal{D}$ there exists a negligible function* negl*, such that*

$$\text{Prob}_{\mathcal{D},\mathcal{A},\mathcal{O}}\left[\forall (x,y) \in L[\mathcal{D},\mathcal{A},\mathcal{O}] \quad y = y_{x,\mathcal{A}^\mathcal{O}}\right] \geq 1 - \mathsf{negl}(\lambda). \tag{3}$$

The notation $y_{x,\mathcal{A}^\mathcal{O}}$ denotes the most likely output of $\mathcal{A}$ on input x over the randomness of *$\mathcal{A}$, i.e., conditioned on a fixed oracle $\mathcal{O}$. If there are two equally likely answers on input x, we choose $y_{x,\mathcal{A}^\mathcal{O}}$ to be the lexicographically smaller one.*

We say algorithm $\mathcal{A}^\mathcal{O}$ is pseudo deterministic for distinguisher $\mathcal{D}^{\mathcal{A}^\mathcal{O}(1^\lambda,\cdot)}(1^\lambda)$, if there exists negligible function negl*, such that equation* (3) *holds for $\mathcal{D}$.*

Note that the definition of $\mathcal{A}$ being pseudo deterministic for distinguisher $\mathcal{D}$ does not imply that it is hard to distinguish whether $\mathcal{A}$ is probabilistic or deterministic—it is only hard for a particular algorithm $\mathcal{D}$. Although this might sound like a weak and somewhat useless property, it will be sufficient to show that if a simulator is pseudo deterministic for a distinguisher, then the simulator can be entirely derandomized via random oracles/ideal ciphers.

We now show that strong multi-stage indifferentiability implies that the simulators are not only stateless but also pseudo deterministic. This is captured by the following lemma.

Lemma 3.4. *Let G^π be a construction with black-box access to primitive π which is strongly multi stage indifferentiable from primitive* Π*. Then there is a stateless* pseudo-deterministic

probabilistic polynomial-time simulator $\mathcal{S}$ such that for all probabilistic polynomial-time distinguishers $\mathcal{D}$ equation (2) *holds in the strong case.*

Proof. Let us assume there exists a stateless simulator $\mathcal{S}$ such that for all distinguishers $\mathcal{D}$ equation (2) holds and such that $\mathcal{S}$ is not pseudo deterministic. The latter implies that there exists distinguisher $\mathcal{D}_{pd}$ against the pseudo determinism of simulator $\mathcal{S}$, i.e., there is a non-negligible probability that $\mathcal{D}_{pd}$ asks a query to $\mathcal{S}$, where $\mathcal{S}$ has a non-negligible probability of returning a different value than the most likely one. We now construct distinguisher $\mathcal{D}'$ against strong multi-stage indifferentiability. Distinguisher $\mathcal{D}'$ runs $\mathcal{D}_{pd}$ on the adversarial a-interface. Let $q_1, \ldots, q_t$ be the queries asked by $\mathcal{D}_{pd}$. Distinguisher $\mathcal{D}'$ then sends the same queries once more to its a-interface and returns 1 if at least one response does not match and 0 otherwise. If $\mathcal{D}'$ is in the real world, talking to G^{π} and π algorithm $\mathcal{D}'$ will always output 0 as π is a function. If on the other hand, $\mathcal{D}'$ is in the ideal world, then $\mathcal{D}_{pd}$ will succeed with noticeable probability and hence $\mathcal{D}'$ will distinguish both worlds with noticeable probability, a contradiction. □

Deterministic Simulators. Bennett and Gill prove in [BG81] that relative to a random oracle the complexity classes $\mathcal{BPP}$ and $\mathcal{P}$ are equivalent. Let us quickly sketch their idea. Given a probabilistic polynomial time oracle Turing machine $\mathcal{M}^{\mathcal{R}}$ which has access to random oracle $\mathcal{R}$ and which decides a language $\mathcal{L}$ in $\mathcal{BPP}$ we can prove the existence of a *deterministic* polynomial time Turing machine $\mathcal{D}^{\mathcal{R}}$ which also decides $\mathcal{L}$. Let us by $p(|x|)$ denote the runtime of machine $\mathcal{M}^{\mathcal{R}}$ for inputs of length $|x|$. As $\mathcal{M}^{\mathcal{R}}$ runs in polynomial time there exists a polynomial upper bound $p(|x|)$ on the length of queries $\mathcal{M}^{\mathcal{R}}$ can pose to the random oracle. To derandomize $\mathcal{M}^{\mathcal{R}}$ we construct a deterministic machine $\mathcal{D}^{\mathcal{R}}$ which works analogously to $\mathcal{M}^{\mathcal{R}}$ with the single exception that when $\mathcal{M}^{\mathcal{R}}$ requests a random coin then $\mathcal{D}^{\mathcal{R}}$ generates this coin deterministically by querying the random oracle on the next smallest input that cannot have been queried by $\mathcal{M}^{\mathcal{R}}$ due to its runtime restriction. As the random oracle produces perfect randomness, the machines decide the same language with probability 1 over the choice of random oracle.

Using the techniques developed by Bennet and Gill [BG81] we now show that in the multi-stage indifferentiability setting, if a simulator is pseudo deterministic for a distinguisher $\mathcal{D}$, then it can be derandomized, in case the constructed primitive Π is a random oracle or an ideal cipher. When applied to a simulator $\mathcal{S}$ that is universal for all distinguishers (strong indifferentiability), these derandomization techniques yield a family of simulators that depends only on the number of queries made by the distinguisher (weak indifferentiability). We give the proof in Appendix A.

Lemma 3.5. *Let $\mathcal{A}^{\Pi}$ be a stateless probabilistic polynomial-time algorithm with oracle access to a random oracle $\mathcal{R}_{\ell,m}$ or an ideal cipher $\mathcal{E}_{k,n}$ for $\ell \in \omega(\log \lambda)$ (resp., $(k+n) \in \omega(\log \lambda)$). Let s be polynomial in λ. From $\mathcal{A}^{\Pi}$, we construct a deterministic algorithm $\mathcal{B}^{\Pi}$ such that the following holds: for all efficient distinguishers $\mathcal{D}$ that make less than s queries to their oracle, it holds that if $\mathcal{A}^{\Pi}$ is pseudo deterministic for $\mathcal{D}$, then*

$$\left|\mathrm{Prob}_{R,\Pi}\left[\mathcal{D}^{\Pi,\mathcal{A}^{\Pi}(R,\cdot)}(1^{\lambda})=1\right]-\mathrm{Prob}_{\Pi}\left[\mathcal{D}^{\Pi,\mathcal{B}^{\Pi}(\cdot)}(1^{\lambda})=1\right]\right|$$

is negligible, where the probability is over the choice of oracle Π *and algorithm* $\mathcal{A}$*'s and distinguisher* $\mathcal{D}$*'s internal coin tosses for the first case and over the choice of oracle* Π *and distinguisher* $\mathcal{D}$*'s internal coin tosses in the second.*

4 The Random Oracle and Ideal Cipher Model are Incomparable

In this section we prove that the random oracle-model and the ideal cipher-model are incomparable with respect to strong multi-stage indifferentiability. We start by giving an alternative, simpler proof of the fact that multi-stage indifferentiable constructions cannot be built via domain extension [DGHM13, LAMP12] (Lemma 4.1). [DGHM13] rule out domain extension even for a single bit of extension. In turn, we obtain an easier proof in the setting where the extension factor is super logarithmic. In Section 4.1 we then present our duality lemma for multi-stage indifferentiability which allows us to conclude that the ROM and the ICM are incomparable with respect to strong multi-stage indifferentiability.

Lemma 4.1. *Let* $\mathcal{R}$ *be a random oracle with domain* $\{0,1\}^\ell$ *(resp.,* $\mathcal{E}$ *be an ideal cipher with domain* $\{0,1\}^k \times \{0,1\}^n$*) and* π *be any ideal primitive with domain size* 2^v*. For* $\ell - v \in \omega(\log(\lambda))$ *(resp.,* $k + n - v \in \omega(\log(\lambda))$*) there exists no construction* G^π *that is weakly multi-stage indifferentiable from* $\mathcal{R}$ *(resp.,* $\mathcal{E}$*).*

We prove Lemma 4.1 for the random oracle case; the proof for ideal ciphers works analogously. Note that we prove the statement for weak multi-stage indifferentiability, thereby essentially ruling out any (possibly non-black-box) construction.

In the following proof we consider a particular distinguisher that tests for the ideal world by forcing the simulator to query its oracle on a particular value M. We show that no simulator is able to do this with more than negligible probability since M is drawn from a very large set while the simulator, being stateless, is only able to make queries from a negligible fraction of this large set; it thus fails to pass the test.

Proof of Lemma 4.1. Assume towards contradiction that there exists construction G^π that is weakly multi stage indifferentiable from random oracle $\mathcal{R}$ and, hence, for every distinguisher $\mathcal{D}$ there exists a stateless simulator $\mathcal{S}$ such that $\mathcal{D}$ cannot distinguish between the real and ideal world.

We consider a distinguisher $\mathcal{D}^{\mathsf{h},\mathsf{a}}$ with access to honest and adversarial interfaces (h,a) which implement the random oracle $\mathcal{R}$ and simulator $\mathcal{S}$ in the ideal world and construction G^π and ideal primitive π in the real world. The distinguisher $\mathcal{D}$ chooses a message $M \in \{0,1\}^\ell$ uniformly at random and executes construction G via an internal simulation using its adversarial interface a, i.e., it computes $G^{\mathsf{a}}(M)$. Then, the distinguisher asks its honest interface on message M to compute $\mathsf{h}(M)$ and returns 1 if the two results agree and 0 otherwise. Note that in the real world distinguisher $\mathcal{D}$ will always output 1. Thus, the simulator $\mathcal{S}$ has to ensure that $G^{\mathcal{S}^{\mathcal{R}}}(M)$ is equal to $\mathcal{R}(M)$ with overwhelming probability over the choice of the random oracle $\mathcal{R}$. We now prove that, in the ideal world, the two

values match only with negligible probability over the choice of the message M and the two settings can thus be distinguished by $\mathcal{D}$.

Let us assume the ideal world and denote the query/response pairs to the a-interface with $(q_i, r_i)_{1 \le i \le t}$. We analyze the simulator's behavior when it is asked these queries q_1,...,q_t. If for none of the q_i the simulator $\mathcal{S}$ asks the random oracle on M, then the answer of $G^{\mathcal{S}^{\mathcal{R}}}(M)$ is independent of $\mathcal{R}(M)$ and thus different with overwhelming probability. By a simple counting argument, we now prove that, with high probability over the choice of M, on *no* query (not even one outside of the set $(q_i, r_i)_{1 \le i \le t}$), the simulator $\mathcal{S}$ asks $\mathcal{R}$ on M. For this, note that the queries which simulator $\mathcal{S}$ receives are of length v. Hence there are at most 2^v distinct possible queries to $\mathcal{S}$. Denote by c the upper bound on the number of queries that $\mathcal{S}$ asks to its random oracle over all possible queries that $\mathcal{S}$ itself receives. As the simulator $\mathcal{S}$ runs in polynomial time c exists and is polynomial. Noting that $\mathcal{S}$ is stateless, we conclude that $\mathcal{S}$ asks at most $c2^v \ll 2^\ell$ queries. Hence the probability that the distinguisher's M is in the set

$$\left\{M : \exists q\, \mathcal{S}^{\mathcal{R}} \text{ asks } M \text{ on input } q\right\}$$

is negligible. The probability that the distinguisher $\mathcal{D}$ returns 1 in the ideal world where it is given access to simulator $\mathcal{S}$ and a random oracle $\mathcal{R}$ is therefore also negligible. Thus, the distinguisher $\mathcal{D}$ has a distinguishing advantage of almost 1 which concludes the proof. □

4.1 The Duality Lemma for Multi-Stage Indifferentiability

We now prove the inverse direction, that is an ideal cipher cannot be build from a random oracle with larger domain. In contrast to the previous section we here give an impossibility result for strong multi-stage indifferentiability. Our result is, however, more general and of independent interest. Strong multi-stage indifferentiability guarantees the existence of a simulator that is stateless and deterministic. Constructions of ideal primitives often need to be stateless and deterministic as well. If for example, the construction, implements a publicly accessible function such as a hash function, it has to be stateless. Note that this is the case both for random oracles and ideal ciphers.

Now, if we assume that constructions are deterministic and stateless, then we show that, in the case of multi-stage indifferentiability, we can exchange the role of the construction and the role of the simulator, if the simulator is also deterministic and stateless. Our *Duality Lemma* establishes that in this case, an impossibility result (resp. feasibility result) in one direction translates into an impossibility result (resp. feasibility result) in the other direction. However, if the simulator is not deterministic, but only pseudo deterministic, then we need to slightly adapt our notion of *constructions* to also allow pseudo-deterministic constructions. For this note that pseudo deterministic constructions are as useful as deterministic ones since inconsistencies due to the pseudo determinism can only be detected with negligible probability. Formally, however, they are not known to be equivalent, in particular, because $\mathcal{P} \neq \mathcal{BPP}$ implies that pseudo-deterministic polynomial-time algorithms are more powerful than deterministic polynomial-time algorithms.

We prove the Duality Lemma in the case of strong multi-stage indifferentiability.

Lemma 4.2 (Duality Lemma for Multi-Stage Indifferentiability)**.** *Let π and π' by two ideal primitives. Assuming constructions are stateless and pseudo deterministic, then one of two following statements holds:*

1. *The two primitives are computationally equivalent, i.e., there exist constructions G_1, G_2 such that G_1^{π} is strongly multi stage indifferentiable from π' and $G_2^{\pi'}$ is strongly multi stage indifferentiable from π, or*

2. *π and π' are incomparable with respect to strong multi-stage indifferentiability.*

In essence this means that a positive or negative result in either direction gives us a result for the other direction. As we have already seen a negative result for domain extenders this gives us the result for the other directions, i.e., going from a large random oracle $\mathcal{R}$ to a small ideal cipher $\mathcal{E}$, or from a large ideal cipher $\mathcal{E}$ to a small random oracle $\mathcal{R}$.

Proof of Lemma 4.2. Assume construction G^{π} with black-box access to ideal primitive π is strongly multi stage indifferentiable from π'. Then by definition there exists a (pseudo-)deterministic, stateless simulator $\mathcal{S}$ such that no distinguisher $\mathcal{D}$ can tell apart the ideal world $(\pi', \mathcal{S}^{\pi'})$ from the real world (G^{π}, π). Likewise, by definition, G is stateless and (pseudo-)deterministic. We now exchange the roles of construction G and simulator $\mathcal{S}$, thereby getting a new "construction" $\mathcal{S}^{\pi'}$ implementing primitive π. It remains to show that $\mathcal{S}^{\pi'}$ is strongly multi-stage indifferentiable from π.

Let us assume the contrary. Then there exists distinguisher $\mathcal{D}$ that can distinguish between the settings $(\pi', \mathcal{S}^{\pi'})$ and the setting (G^{π}, π). This, however, contradicts the assumption that G^{π} is strongly multi stage indifferentiable from π'. □

An immediate consequence of the duality lemma and Lemma 4.1 is captured by the following corollary:

Corollary 4.3. *The ideal cipher model and the random oracle model are incomparable with respect to strong multi-stage indifferentiability.*

Remark. One interesting consequence of the duality lemma is best seen by an example: Can a random oracle with smaller domain be constructed from a random oracle with a larger domain? Intuitively, it feels natural to assume that this works. However, Lemma 4.1 tells us, that the inverse is not possible and, thus, by the duality lemma we can directly conclude that any construction using a large random oracle cannot be strongly multi stage indifferentiable from a small random oracle. So far, we have failed to either prove impossibility for weak multi-stage indifferentiability or to come up with a construction. We leave this for future work.

5 Single versus Multi-Reset

Luykx et al. [LAMP12] introduce the presumably weaker notion of n-reset indifferentiability, where the distinguisher is allowed to reset the simulator only n times. Naturally, for a

construction that is n-reset indifferentiable the composition theorem holds for games that have $n+1$ or less stages. In the following we show that, however, already the extreme single-reset notion implies full reset indifferentiability for simulators that do not depend on the distinguisher (i.e., the *strong* case). This yields that also for n-reset indifferentiability all our separations hold in a black-box fashion.

What we prove is that the advantage of an n-reset distinguisher is bounded by the advantage of an $(n-1)$-reset distinguisher and that of a single-reset distinguisher where the advantage of a distinguisher $\mathcal{D}$ in the n-reset indifferentiability game is defined as

$$\mathsf{Adv}^{n\text{-reset}}_{\mathcal{S},\mathcal{D}} := \left|\text{Prob}\left[\mathcal{D}^{\mathcal{R},\mathcal{S}^{\mathcal{R}}}(1^\lambda)=1\right] - \text{Prob}\left[\mathcal{D}^{G^\pi,\pi}(1^\lambda)=1\right]\right| .$$

Assuming that a construction is strongly single reset indifferentiable (and thus the advantage for any single-reset distinguisher is negligible) yields the above claim. We use

Lemma 5.1. *Let G^π be a construction with black-box access to primitive π. Then there exists simulator $\mathcal{S}$ such that for all $n>1$ and all distinguishers $\mathcal{D}_n$ that make at most n reset queries there exists a distinguisher $\mathcal{D}_{n-1}$ that makes at most $n-1$ reset queries and a distinguisher $\mathcal{D}_1$ that makes a single reset query and*

$$\mathsf{Adv}^{n\text{-}reset}_{\mathcal{S},\mathcal{D}_n}(1^\lambda) \leq \mathsf{Adv}^{(n-1)\text{-}reset}_{\mathcal{S},\mathcal{D}_{n-1}}(1^\lambda) + \mathsf{Adv}^{1\text{-}reset}_{\mathcal{S},\mathcal{D}_1}(1^\lambda)$$

is negligible in λ.

The proof idea is simple. Given a distinguisher which makes n resets we construct one that ignores the first reset. Now, either this changes the input/output behavior of the simulator noticeably, which yields a distinguisher that only needs a single reset, or it does not in which case the distinguisher with $n-1$ resets is as good as the n-reset distinguisher.

Proof. Let $\mathcal{D}_n$ be a distinguisher that makes at most n reset queries. We construct a distinguisher $\mathcal{D}_{n-1}$ as follows. The distinguisher $\mathcal{D}_{n-1}$ runs exactly as $\mathcal{D}_n$ but does not perform the first reset query of $\mathcal{D}_n$.

In the real world, where the distinguisher is connected to the construction G^π and π, reset queries have no effect and thus we immediately have that

$$\text{Prob}_{r_\mathcal{D}}\left[\mathcal{D}_n^{G^\pi,\pi}(1^\lambda;r_\mathcal{D})=1\right] = \text{Prob}_{r_\mathcal{D}}\left[\mathcal{D}_{n-1}^{G^\pi,\pi}(1^\lambda;r_\mathcal{D})=1\right] \tag{4}$$

where the probability is over the random coins $r_\mathcal{D}$ of the distinguisher.

Let in the ideal world $L_2[\mathcal{D}_n,\mathcal{S},\mathcal{R},r_\mathcal{D},r_\mathcal{S}]$ denote the ordered list of query-answer pairs of queries by distinguisher $\mathcal{D}_n$ to simulator $\mathcal{S}$ up to the second reset query by $\mathcal{D}_n$ when $\mathcal{D}_n$ runs with randomness $r_\mathcal{D}$ and simulator $\mathcal{S}$ runs with randomness $r_\mathcal{S}$ and $\mathcal{R}$ is the random oracle. Note that after each reset query simulator $\mathcal{S}$ takes a fresh set of random coins. Thus, technically we have that $r_\mathcal{S} := r^1_\mathcal{S}\|r^2_\mathcal{S}\|\ldots$ where $r^1_\mathcal{S}$ denotes the simulator's coins up to the first reset and $r^2_\mathcal{S}$ its coins after the first and up to the second reset. All further random coins are irrelevant for the definition of L_2 since we only consider queries up to the second reset query.

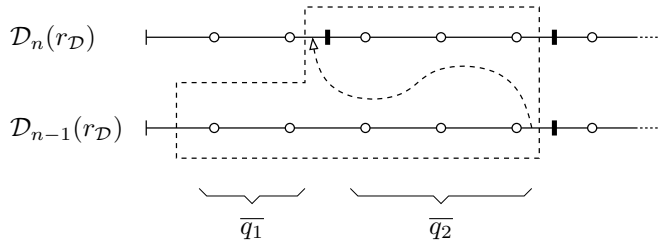

Figure 1: Illustration of $\mathcal{D}_n$ and $\mathcal{D}_{n-1}$'s operation; circles denote queries and rectangles denote resets. The dashed part resembles the resulting single-reset distinguisher $\mathcal{D}_1$ that asks the queries $\overline{q_2}$ twice (separated by a reset). Whether or not the answer to these two query sequences are identical is captured by the event E.

Similarly, we define $L_1[\mathcal{D}_{n-1}, \mathcal{S}, \mathcal{R}, r_\mathcal{D}, r_\mathcal{S}]$ to be the list of query-answer pairs by distinguisher $\mathcal{D}_{n-1}$ to simulator $\mathcal{S}$ up to the first reset query. Note that again $r_\mathcal{S} := r_\mathcal{S}^1 \| r_\mathcal{S}^2 \| \ldots$ but this time already the second part ($r_\mathcal{S}^2$) is irrelevant since we only consider queries up to the first reset query.

Define predicate $\mathsf{E}(\mathcal{R}, r_\mathcal{D}, r_\mathcal{S})$ to hold, iff

$$L_2[\mathcal{D}_n, \mathcal{S}, \mathcal{R}, r_\mathcal{D}, r_\mathcal{S}] = L_1[\mathcal{D}_{n-1}, \mathcal{S}, \mathcal{R}, r_\mathcal{D}, r_\mathcal{S}]$$

for a random oracle $\mathcal{R}$ and randomnesses $r_\mathcal{D}$ and $r_\mathcal{S}$. Note that in case of event $\mathsf{E}(\mathcal{R}, r_\mathcal{D}, r_\mathcal{S})$ it holds that

$$\mathrm{Prob}_{\mathcal{R}, r_\mathcal{D}, r_\mathcal{S}}\left[\mathcal{D}_n^{\mathcal{R}, \mathcal{S}^\mathcal{R}}(1^\lambda) = 1 \mid \mathsf{E}(\mathcal{R}, r_\mathcal{D}, r_\mathcal{S})\right] = \mathrm{Prob}_{\mathcal{R}, r_\mathcal{D}, r_\mathcal{S}}\left[\mathcal{D}_{n-1}^{\mathcal{R}, \mathcal{S}^\mathcal{R}}(1^\lambda) = 1 \mid \mathsf{E}(\mathcal{R}, r_\mathcal{D}, r_\mathcal{S})\right]. \tag{5}$$

In the following we simplify notation and do not make the probability space explicit. That is, the probabilities in the ideal world are always over the random oracle $\mathcal{R}$ the random coins of the distinguisher $r_\mathcal{D}$ and the various random coins of the simulator $r_\mathcal{S}$. Also, we simply write E instead of $\mathsf{E}(\mathcal{R}, r_\mathcal{D}, r_\mathcal{S})$.

Let $\mathcal{D}_1$ denote a distinguisher which makes only a single reset query and which works as follows: $\mathcal{D}_1$ runs $\mathcal{D}_n$ up to the second reset query, passing on queries to its own oracles but not passing on the two reset queries. Let $\overline{q_1}$ denote the queries to the simulator up to the first (ignored) reset query and $\overline{q_2}$ the queries to the simulator after the first (ignored) reset and up to the second (ignored) reset. Now, after the second ignored reset, distinguisher $\mathcal{D}_1$ makes its single reset query and once more sends the sequence $\overline{q_2}$ to the simulator. It outputs 0 in case the simulator's answers are consistent with the previous $\overline{q_2}$ sequence and else it outputs 1. See Figure 1 for a pictorial representation of this operation.

In the real world, distinguisher $\mathcal{D}_1$ will always output 0 since the answers will always

match. Thus, we observe that

$$\begin{aligned}
\mathsf{Adv}^{\text{1-reset}}_{\mathcal{S},\mathcal{D}_1}(1^\lambda) &= \text{Prob}\left[\mathcal{D}_1^{\mathcal{R},\mathcal{S}^\mathcal{R}}(1^\lambda)=1\right] - \text{Prob}\left[\mathcal{D}_1^{G^\pi,\pi}(1^\lambda)=1\right] \\
&= \text{Prob}\left[\mathcal{D}_1^{\mathcal{R},\mathcal{S}^\mathcal{R}}(1^\lambda)=1\right] \\
&\geq \text{Prob}\left[\overline{\mathsf{E}}\right] \cdot \text{Prob}\left[\mathcal{D}_1^{\mathcal{R},\mathcal{S}^\mathcal{R}}(1^\lambda)=1 \;\middle|\; \overline{\mathsf{E}}\right] \\
&= \text{Prob}\left[\overline{\mathsf{E}}\right]. \qquad (6)
\end{aligned}$$

For the last equality, note that if $\overline{\mathsf{E}}$ occurs then there is at least one query answer that differs in both runs. This difference must be during $\overline{q_2}$ since, up to $\mathcal{D}_n$'s first reset, both algorithms are identical and operate on the same coins with the same oracles. Hence $\mathcal{D}_1$ always detects this difference and outputs 1. Thus, we have

$$\begin{aligned}
\mathsf{Adv}^{n\text{-reset}}_{\mathcal{S},\mathcal{D}_n}(1^\lambda) &= \text{Prob}\left[\mathcal{D}_n^{\mathcal{R},\mathcal{S}^\mathcal{R}}(1^\lambda)=1\right] - \text{Prob}\left[\mathcal{D}_n^{G^\pi,\pi}(1^\lambda)=1\right] \\
&= \text{Prob}[\mathsf{E}] \cdot \text{Prob}\left[\mathcal{D}_n^{\mathcal{R},\mathcal{S}^\mathcal{R}}(1^\lambda)=1 \;\middle|\; \mathsf{E}\right] + \text{Prob}\left[\overline{\mathsf{E}}\right] \cdot \text{Prob}\left[\mathcal{D}_n^{\mathcal{R},\mathcal{S}^\mathcal{R}}(1^\lambda)=1 \;\middle|\; \overline{\mathsf{E}}\right] \\
&\quad - \text{Prob}\left[\mathcal{D}_n^{G^\pi,\pi}(1^\lambda)=1\right] \\
&\leq \text{Prob}\left[\mathcal{D}_n^{\mathcal{R},\mathcal{S}^\mathcal{R}}(1^\lambda)=1 \;\middle|\; \mathsf{E}\right] + \text{Prob}\left[\overline{\mathsf{E}}\right] - \text{Prob}\left[\mathcal{D}_n^{G^\pi,\pi}(1^\lambda)=1\right].
\end{aligned}$$

Using equations (4) and (5) we can exchange distinguisher $\mathcal{D}_n$ for distinguisher $\mathcal{D}_{n-1}$ and after reordering we get that

$$= \text{Prob}\left[\mathcal{D}_{n-1}^{\mathcal{R},\mathcal{S}^\mathcal{R}}(1^\lambda)=1 \;\middle|\; \mathsf{E}\right] - \text{Prob}\left[\mathcal{D}_{n-1}^{G^\pi,\pi}(1^\lambda)=1\right] + \text{Prob}\left[\overline{\mathsf{E}}\right].$$

Using equation (6)

$$\begin{aligned}
&\leq \text{Prob}\left[\mathcal{D}_{n-1}^{\mathcal{R},\mathcal{S}^\mathcal{R}}(1^\lambda)=1 \;\middle|\; \mathsf{E}\right] - \text{Prob}\left[\mathcal{D}_{n-1}^{G^\pi,\pi}(1^\lambda)=1\right] + \mathsf{Adv}^{\text{1-reset}}_{\mathcal{S},\mathcal{D}_1}(1^\lambda) \\
&\leq \mathsf{Adv}^{(n-1)\text{-reset}}_{\mathcal{S},\mathcal{D}_{n-1}}(1^\lambda) + \mathsf{Adv}^{\text{1-reset}}_{\mathcal{S},\mathcal{D}_1}(1^\lambda)
\end{aligned}$$

which yields the desired statement. □

6 Acknowledgements

We thank the anonymous reviewers, Pooya Farshim, and Giorgia Azzurra Marson for their valuable comments on preliminary versions of this work. Paul Baecher is supported by grant Fi 940/4-1 of the German Research Foundation (DFG). Christina Brzuska is supported by the Israel Science Foundation (grant 1076/11 and 1155/11), the Israel Ministry of Science and Technology grant 3-9094), and the German-Israeli Foundation for Scientific Research and Development (grant 1152/2011). Arno Mittelbach is supported by CASED (`www.cased.de`).

References

[BBN+09] Mihir Bellare, Zvika Brakerski, Moni Naor, Thomas Ristenpart, Gil Segev, Hovav Shacham, and Scott Yilek. Hedged public-key encryption: How to protect against bad randomness. In Matsui [Mat09], pages 232–249. (Cited on pages 4 and 6.)

[BBO07] Mihir Bellare, Alexandra Boldyreva, and Adam O'Neill. Deterministic and efficiently searchable encryption. In Alfred Menezes, editor, *Advances in Cryptology – CRYPTO 2007*, volume 4622 of *Lecture Notes in Computer Science*, pages 535–552, Santa Barbara, CA, USA, August 19–23, 2007. Springer, Berlin, Germany. (Cited on pages 4 and 6.)

[BCFW09] Alexandra Boldyreva, David Cash, Marc Fischlin, and Bogdan Warinschi. Foundations of non-malleable hash and one-way functions. In Matsui [Mat09], pages 524–541. (Cited on page 4.)

[BG81] C. H. Bennett and J. Gill. Relative to a random oracle A, $P^A \neq NP^A \neq coNP^A$ with probability 1. *SIAM Journal on Computing*, 10(1):96–113, 1981. (Cited on page 12.)

[BK03] Mihir Bellare and Tadayoshi Kohno. A theoretical treatment of related-key attacks: RKA-PRPs, RKA-PRFs, and applications. In Eli Biham, editor, *Advances in Cryptology – EUROCRYPT 2003*, volume 2656 of *Lecture Notes in Computer Science*, pages 491–506, Warsaw, Poland, May 4–8, 2003. Springer, Berlin, Germany. (Cited on page 4.)

[BR93] Mihir Bellare and Phillip Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. In V. Ashby, editor, *ACM CCS 93: 1st Conference on Computer and Communications Security*, pages 62–73, Fairfax, Virginia, USA, November 3–5, 1993. ACM Press. (Cited on page 3.)

[BR94] Mihir Bellare and Phillip Rogaway. Optimal asymmetric encryption. In Alfredo De Santis, editor, *Advances in Cryptology – EUROCRYPT'94*, volume 950 of *Lecture Notes in Computer Science*, pages 92–111, Perugia, Italy, May 9–12, 1994. Springer, Berlin, Germany. (Cited on page 3.)

[BR96] Mihir Bellare and Phillip Rogaway. The exact security of digital signatures: How to sign with RSA and Rabin. In Ueli M. Maurer, editor, *Advances in Cryptology – EUROCRYPT'96*, volume 1070 of *Lecture Notes in Computer Science*, pages 399–416, Saragossa, Spain, May 12–16, 1996. Springer, Berlin, Germany. (Cited on page 3.)

[BRS02] John Black, Phillip Rogaway, and Thomas Shrimpton. Encryption-scheme security in the presence of key-dependent messages. In Kaisa Nyberg and Howard M.

Heys, editors, *SAC 2002: 9th Annual International Workshop on Selected Areas in Cryptography*, volume 2595 of *Lecture Notes in Computer Science*, pages 62–75, St. John's, Newfoundland, Canada, August 15–16, 2002. Springer, Berlin, Germany. (Cited on page 4.)

[BRSS10] John Black, Phillip Rogaway, Thomas Shrimpton, and Martijn Stam. An analysis of the blockcipher-based hash functions from PGV. *Journal of Cryptology*, 23(4):519–545, October 2010. (Cited on page 3.)

[CDMP05] Jean-Sébastien Coron, Yevgeniy Dodis, Cécile Malinaud, and Prashant Puniya. Merkle-Damgård revisited: How to construct a hash function. In Victor Shoup, editor, *Advances in Cryptology – CRYPTO 2005*, volume 3621 of *Lecture Notes in Computer Science*, pages 430–448, Santa Barbara, CA, USA, August 14–18, 2005. Springer, Berlin, Germany. (Cited on pages 3, 4, and 8.)

[CPS08] Jean-Sébastien Coron, Jacques Patarin, and Yannick Seurin. The random oracle model and the ideal cipher model are equivalent. In David Wagner, editor, *Advances in Cryptology – CRYPTO 2008*, volume 5157 of *Lecture Notes in Computer Science*, pages 1–20, Santa Barbara, CA, USA, August 17–21, 2008. Springer, Berlin, Germany. (Cited on pages 3 and 4.)

[DGHM13] Gregory Demay, Peter Gazi, Martin Hirt, and Ueli Maurer. Resource-restricted indifferentiability. In Thomas Johansson and Phong Q. Nguyen, editors, *Advances in Cryptology – EUROCRYPT 2013*, volume 7881 of *Lecture Notes in Computer Science*, pages 664–683, Athens, Greece, May 26–30, 2013. Springer, Berlin, Germany. (Cited on pages 5, 7, 9, and 13.)

[DKW11] Stefan Dziembowski, Tomasz Kazana, and Daniel Wichs. Key-evolution schemes resilient to space-bounded leakage. In Phillip Rogaway, editor, *Advances in Cryptology – CRYPTO 2011*, volume 6841 of *Lecture Notes in Computer Science*, pages 335–353, Santa Barbara, CA, USA, August 14–18, 2011. Springer, Berlin, Germany. (Cited on page 4.)

[EM97] Shimon Even and Yishay Mansour. A construction of a cipher from a single pseudorandom permutation. *Journal of Cryptology*, 10(3):151–162, 1997. (Cited on page 8.)

[GGR12] Oded Goldreich, Shafi Goldwasser, and Dana Ron. On the possibilities and limitations of pseudodeterministic algorithms. *Electronic Colloquium on Computational Complexity (ECCC)*, 19:101, 2012. (Cited on pages 6 and 11.)

[HKT11] Thomas Holenstein, Robin Künzler, and Stefano Tessaro. The equivalence of the random oracle model and the ideal cipher model, revisited. In Lance Fortnow and Salil P. Vadhan, editors, *43rd Annual ACM Symposium on Theory of Computing*, pages 89–98, San Jose, California, USA, June 6–8, 2011. ACM Press. (Cited on pages 3 and 4.)

[LAMP12] Atul Luykx, Elena Andreeva, Bart Mennink, and Bart Preneel. Impossibility results for indifferentiability with resets. Cryptology ePrint Archive, Report 2012/644, 2012. `http://eprint.iacr.org/`. (Cited on pages 5, 6, 7, 13, and 15.)

[Mat09] Mitsuru Matsui, editor. *Advances in Cryptology – ASIACRYPT 2009*, volume 5912 of *Lecture Notes in Computer Science*, Tokyo, Japan, December 6–10, 2009. Springer, Berlin, Germany. (Cited on page 19.)

[MRH04] Ueli M. Maurer, Renato Renner, and Clemens Holenstein. Indifferentiability, impossibility results on reductions, and applications to the random oracle methodology. In Moni Naor, editor, *TCC 2004: 1st Theory of Cryptography Conference*, volume 2951 of *Lecture Notes in Computer Science*, pages 21–39, Cambridge, MA, USA, February 19–21, 2004. Springer, Berlin, Germany. (Cited on pages 3, 8, and 9.)

[RSS11] Thomas Ristenpart, Hovav Shacham, and Thomas Shrimpton. Careful with composition: Limitations of the indifferentiability framework. In Kenneth G. Paterson, editor, *Advances in Cryptology – EUROCRYPT 2011*, volume 6632 of *Lecture Notes in Computer Science*, pages 487–506, Tallinn, Estonia, May 15–19, 2011. Springer, Berlin, Germany. (Cited on pages 3, 5, and 9.)

[Sha49] Claude E. Shannon. Communication theory of secrecy systems. *Bell Systems Technical Journal*, 28(4):656–715, 1949. (Cited on page 3.)

A Missing Proofs for Section 3

A.1 Proof of Lemma 3.5

Lemma (Lemma 3.5, restated)**.** *Let $\mathcal{A}^{\Pi}$ be a stateless probabilistic polynomial-time algorithm with oracle access to a random oracle $\mathcal{R}_{\ell,m}$ or an ideal cipher $\mathcal{E}_{k,n}$ for $\ell \in \omega(\log \lambda)$ (resp., $(k+n) \in \omega(\log \lambda)$). Let s be polynomial in λ. From $\mathcal{A}^{\Pi}$, we construct a deterministic algorithm $\mathcal{B}^{\Pi}$ such that the following holds: for all efficient distinguisher $\mathcal{D}$ that make less than s queries to their oracle, it holds that if $\mathcal{A}^{\Pi}$ is pseudo deterministic for $\mathcal{D}$, then*

$$\left|\mathrm{Prob}_{R,\Pi}\left[\mathcal{D}^{\Pi,\mathcal{A}^{\Pi}(R,\cdot)}(1^{\lambda})=1\right]-\mathrm{Prob}_{\Pi}\left[\mathcal{D}^{\Pi,\mathcal{B}^{\Pi}(\cdot)}(1^{\lambda})=1\right]\right|$$

is negligible, where the probability is over the choice of oracle Π and algorithm $\mathcal{A}$'s and distinguisher $\mathcal{D}$'s internal coin tosses for the first case and over the choice of oracle Π and distinguisher $\mathcal{D}$'s internal coin tosses in the second.

Proof. Let $\mathcal{A}^{\Pi}$ be a stateless algorithm with access to ideal primitive Π where Π is either a random oracle $\mathcal{R}_{\ell,m}$ or an ideal cipher $\mathcal{E}_{k,n}$.

Let $\mathcal{D}$ be an efficient distinguisher for which $\mathcal{A}^{\Pi}$ is pseudo deterministic. As distinguisher $\mathcal{D}$ is efficient, there exists an upper bound $p(|\lambda|)$ on the number of queries to the Π-interface by $\mathcal{D}$. We construct a deterministic algorithm $\mathcal{B}$ which works as $\mathcal{A}$ with the only exception

that $\mathcal{B}$ deterministically generates "random" bits by querying its random oracle, whenever $\mathcal{A}$ makes use of a random bit. For the jth requested random bit, algorithm $\mathcal{B}$ calls the Π-oracle (either random oracle $\mathcal{R}$ or ideal cipher $\mathcal{E}$ where it uses the encryption interface of $\mathcal{E}$) on $p(|\lambda|)+j$ distinct values xor-ing the result and choosing a bit from this result. Note that as $\ell \in \omega(\log \lambda)$ (resp., $n+k \in \omega(\log \lambda)$) there exist sufficiently many distinct values.

Remember that we denote by $y_{q,\mathcal{A}^{\mathcal{O}}}$ the most likely output of algorithm $\mathcal{A}$ on input q conditioned on fixed oracle $\mathcal{O}$. We want to prove that

$$\left|\mathrm{Prob}_{\Pi,\mathcal{D},\mathcal{A}}\left[\mathcal{D}^{\Pi,\mathcal{A}^{\Pi}}(1^{\lambda})=1\right]-\mathrm{Prob}_{\Pi,\mathcal{D}}\left[\mathcal{D}^{\Pi,\mathcal{B}^{\Pi}}(1^{\lambda})=1\right]\right|$$

is negligible in λ. We prove a stronger statement, namely, that the outputs of $\mathcal{A}$ and $\mathcal{B}$ are likely to be identical. We define event C capturing that "the outputs of $\mathcal{A}$ and $\mathcal{B}$ agree on all inputs." Towards this goal we define event A as "algorithm $\mathcal{A}$ returns $y_{q_i,\mathcal{A}^{\Pi}}$ for all queries q_i" where $y_{q_i,\mathcal{A}^{\Pi}}$ is the most likely answer of $\mathcal{A}^{\Pi}$ on input q_i, i.e., we set $y_{q_i,\mathcal{A}^{\Pi}} := \arg\max_y \left\{\mathrm{Prob}_R\left[\mathcal{A}^{\Pi}(q_i;R)=y\right]\right\}$ (cf. Definition 3.3). Likewise, we define event B as "algorithm $\mathcal{B}$ returns $y_{q_i,\mathcal{A}^{\Pi}}$ for all queries q_i." We will show that

$$\mathrm{Prob}_{\Pi,\mathcal{D},\mathcal{A}}[\mathsf{A}] \geq 1-\mathsf{negl} \tag{7}$$

and

$$\mathrm{Prob}_{\Pi,\mathcal{D}}[\mathsf{B}] \geq 1-\mathsf{negl}. \tag{8}$$

Clearly, the probability that $\mathcal{A}$ and $\mathcal{B}$ produce the same answers for all q_i is lower bounded by the probability that $\mathcal{A}$ and $\mathcal{B}$ both output $y_{q_i,\mathcal{A}^{\Pi}}$ for all q_i. Thus,

$$\begin{aligned}\mathrm{Prob}_{\Pi,\mathcal{D},\mathcal{A}}[\mathsf{C}] &\geq \mathrm{Prob}_{\Pi,\mathcal{D},\mathcal{A}}[\mathsf{A}\wedge\mathsf{B}]\\ &= 1-\mathrm{Prob}_{\Pi,\mathcal{D},\mathcal{A}}[\neg\mathsf{A}\vee\neg\mathsf{B}]\\ &\geq 1-(\mathrm{Prob}_{\Pi,\mathcal{D},\mathcal{A}}[\neg\mathsf{A}]+\mathrm{Prob}_{\Pi,\mathcal{D}}[\neg\mathsf{B}])\\ &\geq 1-\mathsf{negl}-\mathsf{negl}.\end{aligned}$$

Let us now make these statements formal as well as prove inequalities (7) and (8). We denote with q_i the queries to $\mathcal{A}$ by $\mathcal{D}$ and by R_i the randomness used by $\mathcal{A}$ on query q_i. We say that event A occurs (over $\Pi,\mathcal{D},R_1,...,R_n$), if

$$\forall i\ \mathcal{A}^{\Pi}(q_i;R_i)=y_{q_i,\mathcal{A}^{\Pi}}.$$

Note that the pseudo-determinism of $\mathcal{A}$ for $\mathcal{D}$ directly implies that

$$\mathrm{Prob}_{\Pi,\mathcal{D},R_1,\ldots,R_n}\left[\forall i\ \mathcal{A}^{\Pi}(q_i;R_i)=y_{q_i,\mathcal{A}^{\Pi}}\right]\geq 1-\mathsf{negl}, \tag{9}$$

which establishes inequality (7). We say that event B occurs (over $\Pi,\mathcal{D}$), if

$$\forall i\ \mathcal{B}^{\Pi}(q_i)=y_{q_i,\mathcal{A}^{\Pi}},$$

where q_i now denotes the queries by $\mathcal{D}$ to algorithm $\mathcal{B}$. Inequality (8) we derive from inequality (7) via an averaging argument. Note that in inequality (9) we consider fresh randomness R_i for every query q_i. If for all queries q_i a random choice of randomness is good with overwhelming probability, then a random choice of randomness is good for all q_i with overwhelming probability:

$$\mathrm{Prob}_{\Pi,\mathcal{D},R}\left[\forall i\ \mathcal{A}^{\Pi}(q_i;R) = y_{q_i,\mathcal{A}^{\Pi}}\right] \geq 1 - \mathsf{negl}. \tag{10}$$

Moreover, when considering the random oracle via lazy sampling, one can observe that the randomness generated by $\mathcal{B}$ from Π is independent from the part of Π that is used in the experiment, which yields that

$$\begin{aligned}\mathrm{Prob}_{\Pi,\mathcal{D}}\left[\forall i; \mathcal{B}^{\Pi}(q_i) = y_{q_i,\mathcal{A}^{\Pi}}\right] &= \mathrm{Prob}_{\Pi,\mathcal{D},R}\left[\forall i; \mathcal{A}^{\Pi}(q_i;R) = y_{q_i,\mathcal{A}^{\Pi}}\right] \\ &\geq 1 - \mathsf{negl}\end{aligned}$$

as desired.

□