Context-Restricted Indifferentiability: Generalizing UCE and Implications on the Soundness of Hash-Function Constructions

Daniel Jost Ueli Maurer

Department of Computer Science, ETH Zurich, Switzerland
{daniel.jost, maurer}@inf.ethz.ch

Abstract

Understanding how hash functions can be used in a sound manner within cryptographic protocols, as well as how they can be constructed in a sound manner from compression functions, are two important problems in cryptography with a long history. Two approaches towards solving the first problem are the random oracle model (ROM) methodology and the UCE framework, and an approach to solving the second problem is the indifferentiability framework.

This paper revisits the two problems and the above approaches and makes three contributions. First, indifferentiability, which comes with a composition theorem, is generalized to context-restricted indifferentiability (CRI) to capture settings that compose only in a restricted context. Second, we introduce a new composable notion based on CRI, called RO-CRI, to capture the security of hash functions. We then prove that a non-interactive version of RO-CRI is equivalent to the UCE framework, and therefore RO-CRI leads to natural interactive generalizations of existing UCE families. Two generalizations of split UCE-security, called strong-split CRI-security and repeated-split CRI-security, are introduced. Third, new, more fine-grained soundness properties for hash function constructions are proposed which go beyond collision-resistance and indifferentiability guarantees. As a concrete result, a new soundness property of the Merkle–Damgård construction is shown: If the compression function is strong-split CRI-secure, then the overall hash function is split secure. The proof makes use of a new lemma on min-entropy splitting which may be of independent interest.

Contents

1 Introduction

The random oracle model (ROM) [BR93] is an important tool towards establishing confidence in the security of real world cryptographic constructions. The paradigm can be described in two steps: first, to design a protocol and prove it secure in the ROM, thus using a random oracle instead of a hash function; second, to instantiate the random oracle with a cryptographic hash function. However, it is well known [CGH04] that no hash function realizes a random oracle; hence, once the random oracle is instantiated the security proof degenerates to a heuristic argument of the security. As a consequence, several different approaches have evolved to reinforce the confidence in such a security argument.

A commonly accepted technique is to use a hash function construction H^f that was proven indifferentiable from a random oracle when using an ideal compression function f. This reduces the security assumption about the hash function to the security assumption about the compression function, thereby excluding attacks exploiting the construction of H^f from f. The indifferentiability framework [MRH04] is a widely used framework that formalizes the proof obligation.

More recently, Bellare et al. [BHK14] proposed the notion of *universal computational extractors (UCE)*. This notion is based on the observation that for most "real world" protocols proven secure in the random oracle model, instantiating the random oracle with a concrete hash function is not known to be insecure. The UCE framework revisits the question of what it means for a family of functions to "behave like a random oracle" and formalizes a new security notion for hash functions aimed at bridging in a systematic and broad manner the gap between the general impossibility result and the apparent security of concrete protocols. The UCE framework has been applied to a set of applications, including security under key-dependent attacks, security under related-key attacks, garbling schemes, proofs of storage, and deterministic encryption. The security of many of these applications is defined as a multi-stage game, since the two-stage nature of UCE makes it hard to apply it directly to scenarios whose security definition is formalized using a more traditional monolithic central adversary.

Contributions. Our contributions are three-fold. First, we introduce a generalization of indifferentiability called context-restricted indifferentiability (CRI). This generalization allows us to model that a resource does not compose generally but can only be used within a well-specified set of contexts.

Second, we apply the general context-restricted indifferentiability framework to the random oracle and whether it can be instantiated with a hash function within certain contexts, called *random-oracle context-restricted indifferentiability (RO-CRI)* security. Moreover, we show that the original UCE definition is a special case of RO-CRI security[1]. Thereby we propose an alternative interpretation of the UCE framework in a traditional single-stage adversary model and provide well-defined composition guarantees. To this end we define the set of non-interactive contexts and then prove that every UCE class can be mapped to a subset of non-interactive contexts such that the UCE problem and the context-restricted indifferentiability statement become equivalent. This treatment directly leads to a generalization of the UCE framework that does not restrict itself to non-interactive contexts and allows for some limited interaction between the source and the distinguisher. More specifically, we propose two generalizations of the

[1] UCE is often understood as a more general paradigm than the original definition introduced by Bellare et al. and many variants thereof exists. Some of them are not directly captured by the RO-CRI definition.

split-source UCE class that we call strong-split security and repeated-split security, respectively. Moreover, we prove that interactive computational extractor (ICE) security [Mit14] implies strong-split security, establishing it as an intermediate notion between the (weak) original UCE notion and the stronger ICE notion.

Finally, we propose that context-restricted indifferentiability, and thereby UCE, could be viewed as a more fine-grained version of indifferentiability. In particular, we propose to use RO-CRI security as a tool for analyzing the soundness of hash function constructions and recommend candidate constructions to come with such proofs. As an example, we prove that the Merkle–Damgård scheme is split-secure if the underlying compression function is strong-split secure (as opposed to the usual much stronger assumption of the compression function being a random function). We thereby generalize a lemma on min-entropy splitting by Damgård et al., which we believe might be of independent interest.

1.1 Related work

We now discuss the relation between context-restricted indifferentiability and some related notions, including variants of UCE as well as variants of indifferentiability.

Universal Computational Extractors. The UCE framework was introduced by Bellare et al. [BHK13] as a tool to provide a family of notions of security for keyed hash functions, refining the predominant random oracle methodology. Since then, the impossibility of various UCE-classes has been shown by Brzuska et al. [BFM14; BM15] and Bellare et al. [BST16], and the possibility of a specific UCE-class in the standard model has been shown by Brzuska and Mittelbach [BM14]. Bellare et al. [BHK14] have also suggested to use the UCE framework to study the domain extension of a finite input-length random oracle to a UCE secure variable input length random oracle. Their motivation is mainly based on finding more efficient constructions if they only require the UCE security of the variable input length random oracle.

Interactive Computational Extractors. In [FM16] Farshim and Mittelbach introduced a generalization of UCE called interactive computational extractors. Generalizing UCE to interactive scenarios is also one of our contributions. The generalization they propose and the one we propose, however, differ on a very fundamental level and pursue different directions. ICE makes the two stages of the original UCE definition symmetrical and thereby eliminates the need for an uniformly random but publicly known key. The definition essentially allows the two stages to jointly form queries using a buffer, even allowing interaction, as long as neither one of the stages can predict the overall query. Their definition is very much motivated by making UCE interactive and symmetrical without falling into obvious impossibility results. In contrast, we exactly use the asymmetry of UCE to embed it in the traditional indifferentiability setting with one dishonest and one honest party. Our context-restricted indifferentiability framework is inherently motivated by giving a definition that is as general as possible while capturing that certain ideal resources such as the random oracle might only compose in a restricted way. Nevertheless, we prove that ICE security implies strong-split context-restricted indifferentiability, relating the two frameworks. Whether the reverse direction is true for some natural context-sets and notion of ICE security remains an interesting open problem, which might lead to a better understanding of the various notions.

Public-Seed Pseudorandom Permutations. In [ST17] Soni and Tessaro introduce the notion of public-seed pseudorandom permutations (psPRP) that are inspired by UCE. In fact, they introduce a generalization of UCE, called public-seed pseudorandomness, of which both psPRP and UCE are instantiations. For their psPRP notion they introduce the unpredictability and reset-security notions analogous to UCE, and moreover they study the relations between psPRP and UCE. In contrast to CRI, their definition is still purely game-based. Moreover, we show in Section 5 that public-seed pseudorandomness can be interpreted as CRI with a fixed real-world resource and a certain type of ideal-world resources, and thus CRI is a strict generalization of their notion.

Reset Indifferentiability. The reset indifferentiability notion has originally been introduced by Ristenpart, Shacham, and Shrimpton in [RSS11] as a workaround to the composition problems in multi-stage settings they highlighted. This variant of indifferentiability has later been proven to be equivalent to indifferentiability with stateless simulators in [DGHM13]. There are two main differences to our notion of CRI: first, our notion uses the traditional single-stage setting of the original indifferentiability framework, whereas reset indifferentiability focuses on multi-stage settings. Second, and more important, our notion weakens the classical indifferentiability notion, whereas reset indifferentiability strengthens it.

Resource-restricted indifferentiability. In [DGHM13], Demay et al. gave an alternative interpretation of the shortcoming of the traditional composition theorem that was highlighted in [RSS11]. They introduced the notion of resource-restricted indifferentiability, which makes the memory used in the simulator explicit in contrast to the original definition which only requires this memory to be polynomially bounded. They show that for security definitions where memory inherently matters, such as the proof of retrievability example of Ristenpart et al., such a fine-grained treatment is vital in order to get meaningful security statements. Again, their notions differs from ours by strengthening indifferentiability while context-restricted indifferentiability is a weakening.

Unsplittable Multi-stage Games. In [Mit14] Mittelbach presents a condition called unspittability on multi-stage games, which allows them to show that the composition theorem of indifferentiability can be salvaged for iterative hash function constructions. This work is similar to ours in the sense that it formalizes a condition in which situations the random oracle can be safely instantiated by a hash function. However, we would like to highlight that context-restricted indifferentiability is a general paradigm that not only applies to iterative hash function constructions but generally allows to formalize ideal resources which can only be used in limited but explicit ways. In addition, the notion of context-restricted indifferentiability also provides a weaker definition for the hash function construction, allowing to circumvent the known impossibility results or finding more efficient constructions, whereas Mittelbach's result assumes that the hash function is indifferentiable from a random oracle in the traditional sense.

2 Preliminaries

2.1 The (traditional) UCE framework

To circumvent the trivial and well-known impossibility result that no hash function family is indifferentiable from a random oracle, Bellare, Hoang, and Keelveedhi [BHK14] introduced the UCE framework to formalize a weaker version of what it means for a family of keyed hash functions to behave like a random oracle. The UCE framework defines a two-stage adversary, where only the first stage—the *source* S—has access to the oracle (either the hash function or the random oracle) and only the second stage—the *distinguisher* D—has access to the hash key hk. The source provides some *leakage* L to the distinguisher that then decides with which system the source interacted. The definition of the security game is presented in Algorithm 1. Here, H.Kg denotes the key generation algorithm, H.Ev the deterministic evaluation algorithm, and l the output length associated with the family of hash functions H.

Algorithm 1 The UCE game

```
function MAIN $\mathrm{UCE}_H^{S,D}(\lambda)$
    $b \xleftarrow{\$} \{0,1\}$; $hk \xleftarrow{\$} \mathrm{H.Kg}(1^\lambda)$
    $L \xleftarrow{\$} S^{\textsc{Hash}}(1^\lambda)$
    $b' \xleftarrow{\$} D(1^\lambda, hk, L)$
    return $(b' = b)$

function HASH$(x, 1^l)$
    if $T[x,l] = \perp$ then
        if $b = 1$ then
            $T[x,l] \leftarrow \mathrm{H.Ev}(1^\lambda, hk, x, 1^l)$
        else
            $T[x,l] \xleftarrow{\$} \{0,1\}^l$
    return $T[x,l]$
```

Without any further restriction, this game is trivial to win: the source queries some point x, obtains the result y, and then provides the tuple (x, y) as leakage to the distinguisher which then decides whether y matches with the hash of x. Therefore, in order for this definition to be meaningful, the leakage has to be restricted in some sense which gives rise to various *UCE classes* depending on the kind of restriction. The basic restriction proposed was that the queries of the source S must be unpredictable given the leakage L. Both statistical unpredictability as well as computational unpredictability have been proposed; however, the latter has been shown to be impossible [BFM14].

2.2 Resources and Converters

The indifferentiability framework by Maurer, Renner, and Holenstein [MRH04] is a widely adopted framework to analyze and prove the security of hash function constructions. The indifferentiability framework is a simulation-based framework that uses the so-called "real world – ideal world" paradigm and formalizes security guarantees as resources (called functionality in the Universal Composability framework [Can01]). A resource S captures the idea of a module which provides some well-defined functionality to the different parties–both the honest and the dishonest ones–which can then be used in a higher level protocol. A resource can either be something physically available, such as an insecure communication network, or can be constructed from another resource R using a cryptographic protocol π. In fact, the goal of the protocol π can be seen as constructing the ideal resource S from the real one R assumed to be available. The protocol is modeled as a converter that connects to the system R.

The indifferentiability framework formalizes this concept in a setting with a single honest and a single dishonest party. In the following we give a brief description of the system algebra used in this work. We basically follow the contemporary notation of indifferentiability presented in [MR16], while sticking to the original reducibility notion.

Formal definitions. A resource is a system with two interfaces via which the resource interacts with its environment. The (private) interface A and the (public) interface E can be thought as being assigned to an honest and a dishonest party, respectively. Let Φ denote the set of resources. All resources in Φ are *outbound* (as in the original version of indifferentiability) meaning that interaction at one interface does not influence the other interface. If two resources V and W are used in parallel, this is again a resource, denoted $[\mathsf{V}, \mathsf{W}]$, where each of the interfaces allows to access the corresponding interfaces of both subsystems. Moreover, we assume the existence of a dummy resource $\square \in \Phi$ such that $[\mathsf{R}, \square] = \mathsf{R}$ for any resource R.

Converters are systems that can be connected to an interface of a resource to translate the inputs and outputs. A converter has two interfaces: the outer interface out that becomes the new interface of the resource, and the inner interface in that is connected to the interface of the existing resource. Attaching a converter π to a specific interface of a resource R yields another resource. We understand the left and the right side of the symbol R as the interface A and E, respectively; thus, attaching π at interface A is denoted $\pi\mathsf{R}$ and attaching it at interface E is denoted $\mathsf{R}\pi$. Let Σ denote the set of converters. Two converters ϕ and ψ can be composed sequentially and in parallel: sequential composition is denoted as $\phi \circ \psi$ such that $(\phi \circ \psi)\mathsf{R} = \phi(\psi\mathsf{R})$ and parallel composition as $[\phi, \psi]$, where $[\phi, \psi][\mathsf{R}, \mathsf{S}] = [\phi\mathsf{R}, \psi\mathsf{S}]$. Moreover, we assume the existence of an identity converter id such that $\mathsf{id}\mathsf{R} = \mathsf{R}\,\mathsf{id} = \mathsf{R}$, for which in particular we have $[\phi, \mathsf{id}][\mathsf{R}, \mathsf{S}] = [\phi\mathsf{R}, \mathsf{S}]$.

Conventions for Describing Systems and Algorithms. We describe our systems using pseudocode. The following conventions are used: We write $x \leftarrow y$ for assigning the value y to the variable x. For a finite set $\mathcal{X}$, $x \overset{\$}{\leftarrow} \mathcal{X}$ denotes assigning x uniformly at random a value in $\mathcal{X}$. Furthermore, $x \overset{\mathsf{P}_X}{\leftarrow} \mathcal{X}$ denotes sampling x according to the indicated probability distribution P_X over $\mathcal{X}$.

Queries (also called inputs) to systems consist of a list of arguments, of which the first one is a suggestive keyword. If the input consists only of the keyword we omit the parenthesis, i.e., we write retrieve or (hash, x). When specifying the domain of the inputs, we ignore the keyword and write (hash, x) $\in \mathcal{X}$ to indicate $x \in \mathcal{X}$. If a system outputs a value x at the interface named int, we denote this "**output** x at int". We generally assume that all resources reply at the same interfaces they have been queried before processing any additional queries. Therefore, if a converter outputs a query at its inside interface, we write "let *var* denote the result" meaning that we wait for the value returned from the connected system and then store it in the variable *var*.

We depict resources using rectangular boxes and converters (which include simulators) using rounded boxes.

2.3 Indifferentiability

In contrast to game-based security definitions, indifferentiability gives composable security guarantees, i.e., the security guarantees obtained are not only with respect to specific attack scenarios but with respect to all possible attacks.

The fundamental idea of composition is then to prove the construction of S from R in isolation and be assured that in any higher level protocol ϕ making use of S, the resource S can be replaced with R with the protocol applied, without degrading the security of ϕ. The system S, while not existing in the real world, therefore serves as an abstraction boundary for the design of cryptographic schemes.

The security definition. In our security statements, we are interested in the advantage of a distinguisher D in distinguishing two resources. We define the advantage of a distinguisher D in distinguishing two resources, say R and S, as

$$\Delta^{\mathsf{D}}(\mathsf{R},\mathsf{S}) := \Pr[\mathsf{DS} = 1] - \Pr[\mathsf{DR} = 1],$$

where DS denotes the output of the distinguisher D when connected to the resource S. The distinguisher thereby gets access to both the interfaces of the resource S and the randomness is taken over both D and S. Moreover, let $\mathsf{R} \approx \mathsf{S}$ denote that $\Delta^{\mathsf{D}}(\mathsf{R},\mathsf{S})$ is negligible for every efficient distinguisher D.[2]

This now allows us to define indifferentiability.

Definition 2.1 (Indifferentiability)**.** Let R and S be 2-interface resources. S is reducible to R by $\pi \in \Sigma$ in the sense of indifferentiability (denoted $\mathsf{R} \overset{\pi}{\Longmapsto} \mathsf{S}$), if

$$\mathsf{R} \overset{\pi}{\Longmapsto} \mathsf{S} \quad :\Longleftrightarrow \quad \exists \sigma \in \Sigma : \pi\mathsf{R} \approx \mathsf{S}\sigma$$

and we refer to the converters π and σ as the protocol and the simulator, respectively.

Figure 1: The real (left) and the ideal (right) setting considered in indifferentiability

In contrast to the original version we do not existentially quantify over the protocol instead explicitly specify it to make statements about concrete protocols. Of course the original definition can be obtained by simply adding the quantifier in a given statement. The two settings, called the real setting and the ideal setting, from the security condition of Definition 2.1 are depicted in Figure 1.

Composability. The formalism of indifferentiability composes in the natural way under some natural closure assumptions[3] on the sets Σ and $\mathcal{D}$ of converters and distinguishers considered. First, if T is reducible to S and S is reducible to R, then T is reducible to R by the composed protocol. Secondly, if S is reducible to R, then for any resource U, $[\mathsf{S},\mathsf{U}]$ is reducible to $[\mathsf{R},\mathsf{U}]$. More formally, for any resources R, S, T, and U we have the following two conditions:

[2] The systems, including the distinguishers, we consider in this work are, in fact, families of systems indexed by a security parameter. The distinguishing advantage is then a function of this parameter. To simplify the presentation, the security parameters are omitted.

[3] The set of distinguishers $\mathcal{D}$ needs to be closed under emulation of a converter. The set of converters needs to be closed under sequential composition.

$$\mathsf{R} \overset{\pi_1}{\Longmapsto} \mathsf{S} \wedge \mathsf{S} \overset{\pi_2}{\Longmapsto} \mathsf{T} \quad \Longrightarrow \quad \mathsf{R} \overset{\pi_2 \circ \pi_1}{\Longmapsto} \mathsf{T}$$

$$\mathsf{R} \overset{\pi}{\Longmapsto} \mathsf{S} \quad \Longrightarrow \quad [\mathsf{R}, \mathsf{U}] \overset{[\pi, \mathsf{id}]}{\Longmapsto} [\mathsf{S}, \mathsf{U}].$$

3 Context Restricted Indifferentiability

In this section we first revisit the motivation behind composable frameworks such as the indifferentiability framework. To handle cases where fully composable security is unachievable, we then introduce the notion of context-restricted indifferentiability, a single-stage security definition inspired by the original motivation behind the UCE-framework. In fact, in the next section we then prove that UCE can be seen as a special case of context-restricted indifferentiability.

3.1 The Limitations of General Composability

At the heart of every composable cryptographic framework, such as indifferentiability, lies the concept of a resource (called functionality in the Universal Composability framework [Can01]). A resource S captures the idea of a module which provides some well-defined functionality to the different parties–both the honest and the dishonest ones–which can then be used in a higher level protocol. The goal of a protocol π in a composable framework is phrased as constructing the resource S from an assumed resource R. The fundamental idea of composition is then to prove the construction of S from R in isolation and be assured that in any environment, the resource S can be replaced with $\pi\mathsf{R}$, without degrading the security. This not only leads to strong security guarantees, but also allows for a modular approach, since the construction of the resource S can be considered entirely independent of its use.

The modular approach of indifferentiability, however, fails if we use a resource S which cannot be reduced to any R available in the physical world, such as the random oracle. Let PO denote a public random oracle resource, and HK a public hash key resource. Then, the famous impossibility result [CGH04] states, that there exists no deterministic and stateless protocol h, implementing a hash function, such that $\mathsf{HK} \overset{\mathsf{h}}{\Longmapsto} \mathsf{PO}$, i.e., such that the hash function reduces the random oracle to the public hash key. We now discuss two well-known approaches to dealing with such resources and then propose a novel third one.

1. The most natural approach should be to weaken the guarantees provided by S, and instead consider a restricted variant S'.

For the random oracle, and many other examples, no such natural weakened version exists. In those cases, we need a different approach.

2. One can restrict the class of distinguishers allowed, e.g., split the distinguisher into multiple stages. The UCE framework is such an approach. Unless there is an application scenario where one can justify such a restricted attacker, this approach leads however to security definitions without evident semantics.

The original motivation of the UCE framework, though, has not been to consider restricted adversaries but to phrase that, in contrast to the impossibility results, real world protocols use the random oracle in “sensible” ways. We turn this motivation into a third approach:

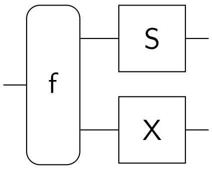


Figure 2: The resource S embedded in a context (f, X). We depict resources using rectangular boxes and converters using rounded boxes. The interface of the honest party of a resource is depicted on the left side, and the one of the dishonest party on the right side.

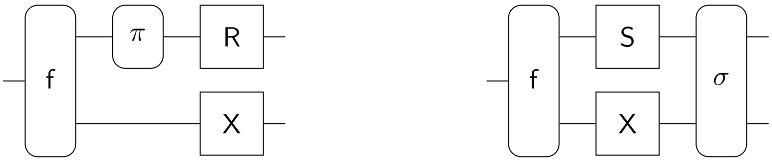


Figure 3: The real (left) and the ideal (right) setting considered in context-restricted indifferentiability.

3. One can restrict composition in a well-defined way. If there is a resource S that cannot be reduced to a resource R, i.e., we cannot replace S by $\pi\mathsf{R}$, for some converter π, in all contexts, we propose to make explicit in which contexts one *can* do it.

3.2 Context-Restriction

In this section we formally define the idea of restricting composition. In order to do so, we define a context in which we allow the resource S to be used. A context consists of an auxiliary parallel resource X and some converter f applied by the honest party, as depicted in Figure 2. We usually call this converter f a *filter* to indicate that its goal is to restrict the access to the resource S. To obtain general statements, we consider a *set* of contexts instead of a single one. This set should be general enough to capture many application scenarios but avoid those for which the impossibility is known.

Definition 3.1. A context set $\mathcal{C}$ is a subset of $\Sigma \times \Phi$, where Σ denotes the set of all converters and Φ denotes the set of all resources.

Recall that our goal is to make a modular statement: reducing S to another resource R in each of these contexts in $\mathcal{C}$, i.e., finding a single resource R and protocol π such that $\pi\mathsf{R}$ can instantiate S in each of these contexts in $\mathcal{C}$. Therefore, the same context appears in both the real and the ideal setting. See Figure 3 for an illustration of the distinction problem when fixing a specific context. Quantifying over all contexts of a set leads to the following definition of *context-restricted indifferentiability.*

Definition 3.2. Let $\mathcal{C} \subseteq \Sigma \times \Phi$ be a given set of contexts, and let R and S be 2-interface resources. We define S to be $\mathcal{C}$-restricted reducible to R by $\pi \in \Sigma$ and $\sigma \in \Sigma$ in the sense of indifferentiability (denoted $\mathsf{R} \overset{\pi,\mathcal{C}}{\underset{\mathsf{cr}}{\Longmapsto}} \mathsf{S}$), as

$$\mathsf{R} \overset{\pi,\mathcal{C}}{\underset{\mathsf{cr}}{\Longmapsto}} \mathsf{S} \quad :\Longleftrightarrow \quad \forall (\mathsf{f}, \mathsf{X}) \in \mathcal{C}\ \exists \sigma \in \Sigma\colon\ \mathsf{f}[\pi\mathsf{R}, \mathsf{X}] \approx \mathsf{f}[\mathsf{S}, \mathsf{X}]\sigma$$

and refer to the converters π and σ as the protocol and the simulator, respectively.

3.3 Composition

Composability generally refers to the property of a framework that from one, or multiple, given statements, new ones can be automatically deduced in a sound way without having to reprove them. More concretely, in CRI we are interested in deducing new reducibility statements from given ones. Using the abstract algebraic approach of constructive cryptography [MR11], such composition properties are usually consequences of composition-order invariance, a natural associativity property stating that the order in which we connect systems is irrelevant. As a consequence, composition properties can be easily illustrated by figures, and we thus omit fully formal proofs.

In order to formally state the composition theorem, we first define the closure of a context set as follows.

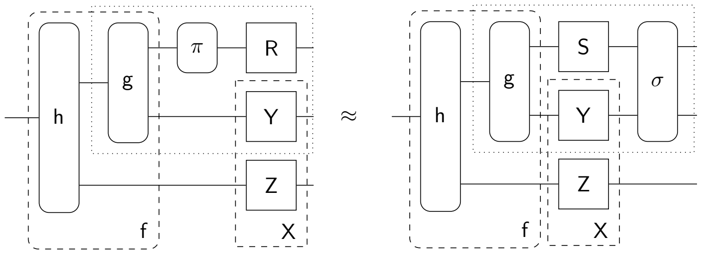
Definition 3.3. Let $\mathcal{C} \subseteq \Sigma \times \Phi$ be a given set of contexts. We denote by $\bar{\mathcal{C}} \subseteq \Sigma \times \Phi$ the following set of contexts:

$$\bar{\mathcal{C}} := \{(\mathsf{f}, \mathsf{X}) \in \Sigma \times \Phi \mid \exists (\mathsf{g}, \mathsf{Y}) \in \mathcal{C}\ \exists \mathsf{h} \in \Sigma\ \exists \mathsf{Z} \in \Phi :\ \mathsf{h} \circ \mathsf{g} = \mathsf{f} \wedge [\mathsf{Y}, \mathsf{Z}] = \mathsf{X}\}.$$

This closure operator follows the intuition that if a resource can be reduced to another one in a certain context, then this also holds true if we further restrict the context by requiring an additional parallel resource and filter. Thus, context-restricted indifferentiability is irrespective to the closure operator.

Proposition 3.4. *Let* $\mathsf{R}, \mathsf{S} \in \Phi$ *denote resources,* $\pi \in \Sigma$ *denote a converter, and let* $\mathcal{C}$ *denote a set of contexts. We then have* $\mathsf{R} \overset{\pi,\mathcal{C}}{\underset{\mathsf{cr}}{\Longmapsto}} \mathsf{S} \iff \mathsf{R} \overset{\pi,\bar{\mathcal{C}}}{\underset{\mathsf{cr}}{\Longmapsto}} \mathsf{S}$.

Proof. The implication $\Longleftarrow$ is trivial, since $\mathcal{C} \subseteq \bar{\mathcal{C}}$. We now prove the other direction. Let $(\mathsf{f}, \mathsf{X}) \in \bar{\mathcal{C}}$ and notice that by Definition 3.3 this implies that there exists $(\mathsf{g}, \mathsf{Y}) \in \mathcal{C}$, $\mathsf{h} \in \Sigma$, and $\mathsf{Z} \in \Phi$ such that $\mathsf{h} \circ [\mathsf{g}, \mathsf{id}] = \mathsf{f}$ and $[\mathsf{Y}, \mathsf{Z}] = \mathsf{X}$.



By our assumption, we know that $\mathsf{g}[\pi, \mathsf{id}][\mathsf{R}, \mathsf{Y}]$ is indistinguishable from $\mathsf{g}[\mathsf{S}, \mathsf{Y}]\sigma$, as indicated by the dotted box. Thus, if we add the additional filter h and resource Z, they remain indistinguishable. This concludes the proof. □

Finally, we can state the actual composition theorem. Note that the additional conditions compared to the composition theorem of classical indifferentiability (cf. Section 2.3) are a direct consequence of the context restrictions. For instance, if in the sequential case we reduce T to S in one of the given contexts, we have to ensure that now we are again in a valid context for reducing

S to R. This highlights that in order for context-restricted indifferentiability to be useful, the context-sets have to be defined in a form that containment can be easily verified. We discuss an example of the composition theorem in Section 3.5.

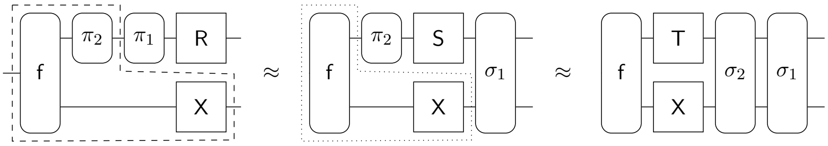
Theorem 3.5. *Let* R, S, T, *and* U *denote resources, let* π_1 *and* π_2 *denote protocols, and* $\mathcal{C}_1$ *and* $\mathcal{C}_2$ *contexts sets. If* $\forall(\mathsf{f},\mathsf{X}) \in \mathcal{C}_2 : \; (\mathsf{f} \circ [\pi_2, \mathsf{id}], \mathsf{X}) \in \overline{\mathcal{C}_1}$, *then we have*

$$\mathsf{R} \underset{\mathsf{cr}}{\overset{\pi_1,\mathcal{C}_1}{\Longmapsto}} \mathsf{S} \wedge \mathsf{S} \underset{\mathsf{cr}}{\overset{\pi_2,\mathcal{C}_2}{\Longmapsto}} \mathsf{T} \quad \Longrightarrow \quad \mathsf{R} \underset{\mathsf{cr}}{\overset{\pi_2\circ\pi_1,\mathcal{C}_2}{\Longmapsto}} \mathsf{T}.$$

Moreover, if $\forall(\mathsf{f},\mathsf{X}) \in \mathcal{C}_2 : \; (\mathsf{f}, [\mathsf{U},\mathsf{X}]) \in \overline{\mathcal{C}_1}$, *then we have*

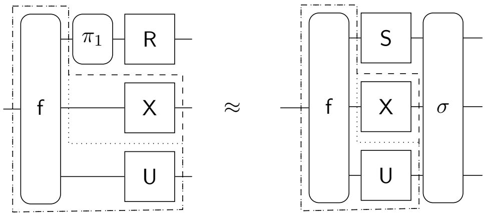
$$\mathsf{R} \underset{\mathsf{cr}}{\overset{\pi_1,\mathcal{C}_1}{\Longmapsto}} \mathsf{S} \quad \Longrightarrow \quad [\mathsf{R},\mathsf{U}] \underset{\mathsf{cr}}{\overset{\pi_1,\mathcal{C}_2}{\Longmapsto}} [\mathsf{S},\mathsf{U}].$$

Proof. We first show the sequential case. Assume that the prerequisite regarding the two context sets is satisfied. Moreover, consider an arbitrary context $(\mathsf{f},\mathsf{X}) \in \mathcal{C}_2$ and the three system configurations, depicted in the following figure.



Using the assumed property on $\mathcal{C}_1$ and Proposition 3.4, we know that the context indicated with the dashed line is a valid one for reducing S to R, yielding the first equality. The second equality follows directly from the premise.

In order to show the parallel composition property, assume again that the corresponding condition on the context sets is satisfied. Moreover, consider an arbitrary context $(\mathsf{f},\mathsf{X}) \in \mathcal{C}_1$ and the two system configurations, depicted in the following figure.



Using the assumed property on $\mathcal{C}_1$ and Proposition 3.4, we know that the context indicated with the dashed line is a valid one for reducing S to R. In short, parallel composition in CRI is just associativity: The resource U can be seen as both part of the context, indicated by the dashed line, or part of the real and ideal resources, indicated by the dotted line. This concludes the overall proof. □

3.4 Relation to Indifferentiability

Let id denote the identity converter, such that $\mathsf{id}\mathsf{R} = \mathsf{R}$ and $\square$ the neutral resource, such that $[\mathsf{R}, \square] = \mathsf{R}$, for any resource R. It is then easy to see that regular indifferentiability, which guarantees full composition, is a special case of context-restricted indifferentiability with the context set $\mathcal{C}_{id} \coloneqq \{(\mathsf{id}, \Box)\}$, since $\overline{\mathcal{C}_{id}} = \Sigma \times \Phi$, i.e., the closure equals to the set of all resources and converters. One can, however, also take the opposite point of view and consider context-restricted indifferentiability to be a special case of plain indifferentiability. From this perspective, CRI reducibility is just a set of normal reducibility statements where the context is part of the resources and protocols, respectively. This can be summarized in the following proposition.

Proposition 3.6. *Let* $\mathcal{C}_{id} \coloneqq \{(\mathsf{id}, \Box)\}$*. For all resources* R*,* S*, protocol* π*, and context sets* $\mathcal{C} \subseteq \Sigma \times \Phi$*, we have*

$$\mathsf{R} \overset{\pi}{\Longmapsto} \mathsf{S} \quad \Longleftrightarrow \quad \mathsf{R} \underset{\mathsf{cr}}{\overset{\pi,\mathcal{C}_{id}}{\Longmapsto}} \mathsf{S},$$

$$\mathsf{R} \underset{\mathsf{cr}}{\overset{\pi,\mathcal{C}}{\Longmapsto}} \mathsf{S} \quad \Longleftrightarrow \quad \forall (\mathsf{f}, \mathsf{X}) \in \mathcal{C}\colon\ [\mathsf{R}, \mathsf{X}] \overset{\mathsf{f} \circ [\pi, \mathsf{id}]}{\Longmapsto} \mathsf{f}[\mathsf{S}, \mathsf{X}].$$

Using $\overline{\mathcal{C}_{id}} = \Sigma \times \Phi$, it is also easy to see that the composition theorem of regular indifferentiability is just a special case of Theorem 3.5.

3.5 An Example of CRI: Diffie-Hellman Key Exchange

The general setting. Consider the following simple example: two honest parties, e.g., Alice and Bob, perform a Diffie-Hellman key exchange using authenticated communication and then extract an actual key by hashing the group element g^{ab}, while an eavesdropper is present.

Since both the honest parties hash the exactly same element, there is no necessity to treat them as different parties and we can work in the indifferentiability setting with one honest party and the adversary. Consider the following resources: let DH be a Diffie-Hellman resource (modeling the authenticated key exchange) that outputs g^{ab} at interface $\mathtt{A}$ and (g^a, g^b) at interface $\mathtt{E}$, let PO denote a random oracle accessible by both parties, let HK denote a public hash key resource that outputs the key at both interfaces, and let KEY be a resource that outputs a uniformly random key at interface $\mathtt{A}$ and nothing at interface $\mathtt{E}$. The Diffie-Hellman converter π takes the group element g^{ab} at the inside interface, inputs it to the random oracle, and outputs the obtained result at the outside interface. It is easy to see that under the CDH assumption we have $[\mathsf{PO}, \mathsf{DH}] \overset{\pi}{\Longmapsto} \mathsf{KEY}$, using the simulator σ that chooses (g^a, g^b) uniformly at random and simulates the interface $\mathtt{E}$ of the public random oracle. Note that the fact that the random oracle "vanishes" and is simulated in the ideal world corresponds to the notion of a programmable random oracle...

Limitation of Indifferentiability. The explicit appearance of the resource PO in the above statement corresponds to a proof in the so called random oracle model. The corresponding simulator σ chooses (g^a, g^b) uniformly at random and simulates the interface $\mathtt{E}$ of the public random oracle[4]. If we want to obtain a proof in the standard model, i.e., getting rid of the assumed random oracle resource, we would need to find a (potentially) keyed hash function that instantiates the random oracle, which is of course impossible. Such a hash function is in our terminology just a converter h that reduces the random oracle to the public hash key resource HK, i.e., $\mathsf{HK} \overset{\mathsf{h}}{\Longmapsto} \mathsf{PO}$. If we had such a hash function, we could use parallel composition to obtain $[\mathsf{HK}, \mathsf{DH}] \overset{[\mathsf{h},\mathsf{id}]}{\Longmapsto} [\mathsf{PO}, \mathsf{DH}]$ and then sequential composition to obtain $[\mathsf{HK}, \mathsf{DH}] \overset{\pi \circ [\mathsf{h},\mathsf{id}]}{\Longmapsto} \mathsf{KEY}$.

[4] The fact that the random oracle "vanishes" and is simulated in the ideal world corresponds to the notion of a programmable random oracle.

Applying CRI. The main obstacle in the way of the modular approach is that there exists no hash function that reduces the random oracle to a public hash key. However, using the formalism of context-restricted indifferentiability there might be a context set $\mathcal{C}$ such that the random oracle is reducible for a given hash function h. Composing this with the second step should then be possible as long as the protocol which we want to actually apply is in the context set, i.e., $(\pi, \mathsf{DH}) \in \mathcal{C}$. We now show, that this is exactly what the composition theorem of CRI yields:

Assume that $\mathsf{HK} \overset{\mathsf{h},\mathcal{C}}{\underset{\mathsf{cr}}{\Longmapsto}} \mathsf{PO}$ for some context set $\mathcal{C}$ with $(\pi, \mathsf{DH}) \in \mathcal{C}$. Let $\mathcal{C}' := \{([\pi, \mathsf{id}], \square)\}$. According to the parallel composition rule of Theorem 3.5, we have that $[\mathsf{HK}, \mathsf{DH}] \overset{\mathsf{h},\mathcal{C}'}{\underset{\mathsf{cr}}{\Longmapsto}} [\mathsf{PO}, \mathsf{DH}]$, since by definition of the identity converter and the neutral resource, $([\pi, \mathsf{id}], [\mathsf{DH}, \square])$ is equivalent to (π, DH) and thus contained in $\mathcal{C}$. Using Proposition 3.4, we moreover have $[\mathsf{PO}, \mathsf{DH}] \overset{\pi,\mathcal{C}_{id}}{\underset{\mathsf{cr}}{\Longmapsto}} \mathsf{KEY}$ and since by definition $(\mathsf{id} \circ [\pi, \mathsf{id}], \square) = ([\pi, \mathsf{id}], \square) \in \mathcal{C}'$, we can apply sequential composition and obtain $[\mathsf{HK}, \mathsf{DH}] \overset{\pi\circ\mathsf{h},\mathcal{C}_{id}}{\underset{\mathsf{cr}}{\Longmapsto}} \mathsf{KEY}$, which is equivalent to $[\mathsf{HK}, \mathsf{DH}] \overset{\pi\circ\mathsf{h}}{\Longmapsto} \mathsf{KEY}$.

In summary, this shows that the composition theorem of context-restricted indifferentiability yields exactly what one expects: composition works if and only if the considered application is in the set of allowed contexts. This of course implies that the context set must be defined in such a way that verifying this fact becomes as easy as possible. For the above example, for instance, it is easy to see that this works if $\mathcal{C}$ is the context-set of split-security combined with computational unpredictability, or statistical unpredictability if we are willing to assume DDH instead of CDH. Split security is discussed in more detail in Section 6.1.

4 Random-oracle Context-Restricted Indifferentiability: generalizing UCE

In the following section we present an alternative formalization of UCE security based on context-restricted indifferentiability, namely that a random oracle is indeed reducible to a public hash key, provided the context is adequately restricted. To this end, we show that every possible UCE family $\mathcal{S}^x$, where $x \in \{\mathrm{sup}, \mathrm{cup}, \mathrm{srs}, \mathrm{crs}, \mathrm{splt}, \ldots\}$, can be mapped to a set of contexts $\mathcal{C}$ for which the UCE statement implies the context-restricted indifferentiability statement $\mathsf{HK}_\mathsf{H} \overset{\mathsf{hash}_\mathsf{H},\mathcal{C}}{\underset{\mathsf{cr}}{\Longmapsto}} \mathsf{RO}$, and moreover, if the context-restricted indifferentiability statement were restricted to a specific simulator, the reverse direction would hold as well.

Here $H\colon H.\mathcal{K} \times H.\mathcal{X} \to H.\mathcal{Y}$ denotes a keyed hash function, HK_H denotes the public hash key resource that outputs the key at both interfaces, hash_H denotes the protocol that implements the hash oracle using the hash key, and RO_H denotes the private random oracle resource with the same input and output domains as the hash function H. See Section 2.2 for a formal description of these resources and converters.

Remark. By private we mean that this resource only accepts queries at interface `A`. This is a deliberate design choice–of course one could also look at a public random oracle–which originates in the UCE framework just choosing the hash key uniformly at random instead of allowing an arbitrary efficient simulator with access to the random oracle to generate this key.

```
Resource $\mathsf{HK}_{\mathsf{H}}$

Initialization
    $k \xleftarrow{\$} H.\mathcal{K}$

Interface $\mathtt{i} \in \{\mathtt{A}, \mathtt{E}\}$
Input: getkey
    output $k$ at i
```

```
Converter $\mathsf{hash}_{\mathsf{H}}$

Initialization
    output getkey at in
    let $k$ denote the result

Interface out
Input: $(\mathtt{hash}, x) \in H.\mathcal{X}$
    output $H(k, x)$ at out
```

```
Converter $\sigma_{\mathtt{H}}$

Initialization
    $k \xleftarrow{\$} H.\mathcal{K}$

Interface out
Input: getkey
    output $k$ at out
```

```
Resource H

Initialization
    $k \xleftarrow{\$} H.\mathcal{K}$

Interface A
Input: $(\mathtt{hash}, x) \in H.\mathcal{X}$
    output $H(k, x)$ at A

Interface E
Input: getkey
    output $k$ at E
```

```
Resource $\mathsf{RO}_{\mathsf{H}}$

Initialization
    $k \xleftarrow{\$} H.\mathcal{K}$
    for all $x \in H.\mathcal{X}$ do
        $T[x] \leftarrow \perp$

Interface A
Input: $(\mathtt{hash}, x) \in H.\mathcal{X}$
    if $T[x] = \perp$ then
        $T[x] \xleftarrow{\$} H.\mathcal{Y}$
    output $T[x]$ at A
```

Figure 4: Formal definitions of the resources and converters.

4.1 Mapping Contexts to UCE Sources

In order to map every UCE family to an equivalent set of contexts, we first introduce the set of non-interactive contexts, i.e., the communication between the source and the distinguisher being unidirectional. This restricted set of contexts faithfully encodes the structural restrictions of the traditional UCE game (cf. page 6), where the communication between the source and the distinguisher is unidirectional. Recall that we are in the same general setting as the classical indifferentiability framework, where one only considers out-bound resources for which communication at one interface does not affect the other interface.

Definition 4.1. A *non-interactive filter* is a converter that at the outer interface just accepts a single trigger query (usually called `retrieve`). Let Σ^{ni} denote the set of all non-interactive filters.

Definition 4.2. A *non-interactive resource* X is a resource that at the interface $\mathtt{E}$ accepts at most a single trigger query (usually called `retrieve`). Let Φ^{ni} denote the set of all non-interactive resources.

So far we have defined the general set of contexts corresponding to the UCE setting. In order to relate specific sources, we now specify a surjective mapping from non-interactive contexts, which consists of a non-interactive filter and a non-interactive resource, to the set $\mathcal{S}$ of all UCE sources.

Lemma 4.3. *Consider the following function* $\phi\colon \Sigma^{\mathsf{ni}} \times \Phi^{\mathsf{ni}} \to \mathcal{S}$ *that maps any fixed context* (f, X) *to the following UCE source* S*, which internally emulates* f *and* X*:*

1. *It queries the interface* $\mathtt{E}$ *of* X *to obtain* z *(if* X *accepts any).*
2. *It queries the outside interface of the filter* f *to obtain* y*. The queries at the inside interface of* f *are forwarded to the resource* X *or output as queries to the hash oracle, respectively.*
3. *It outputs* $L = (y, z)$ *if* X *accepted the query at the interface* $\mathtt{E}$*, and* $L = y$ *otherwise.*

This function ϕ *is surjective.*

Proof. First, it is easy to see that ϕ is indeed a function from $\Sigma^{\mathsf{ni}} \times \Phi^{\mathsf{ni}}$ to $\mathcal{S}$, i.e., $\phi(\mathsf{f}, \mathsf{X})$ is a valid UCE source for every context $(\mathsf{f}, \mathsf{X}) \in \Sigma^{\mathsf{ni}} \times \Phi^{\mathsf{ni}}$. To see that this function is surjective, fix an arbitrary source S. Now, let f_{S} denote the filter that upon receiving the query `retrieve` at the outer interface internally runs S and answers this query with the leakage L. Each hash query of S is output at the inner sub-interface $\mathtt{in.H}$ and the corresponding answer is forwarded to S. Clearly $\phi(\mathsf{f}_{\mathsf{S}}, \square) = S$, where $\square \in \Phi^{\mathsf{ni}}$ denotes the dummy resource. □

4.2 The Equivalence

We now show, that for the specific simulator $\sigma_{\mathtt{H}}$ that chooses the hash key uniformly at random, if for every specific context (f, X) the distinguishing problem of context-restricted indifferentiability is hard, then the UCE game with the fixed source $\phi(\mathsf{f}, \mathsf{X})$ is hard as well, and vice versa. In order to relate more directly to the traditional UCE definition, we introduce the RO-CRI advantage, which is depicted in Figure 5 for a specific context $(\mathsf{f}, \mathsf{X}) \in \mathcal{C}$.

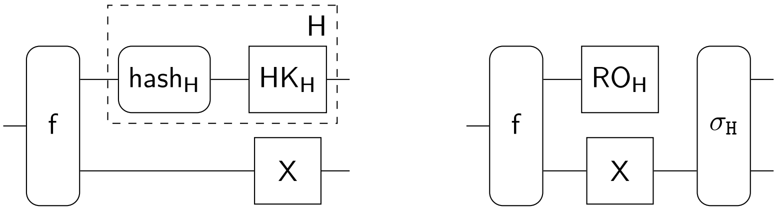


Figure 5: The real (left) and the ideal (right) setting of context restricted indifferentiability when applied to UCE.

Definition 4.4. We define the *random-oracle context-restricted indifferentiability (RO-CRI)* advantage of a distinguisher D on a hash function H in a context (f,X) as

$$\mathbf{Adv}^{\mathrm{RO-CRI}}_{\mathsf{H},\mathsf{f},\mathsf{X},\sigma}(\mathsf{D}) := \Delta^{\mathsf{D}}(\mathsf{f}[\mathsf{H},\mathsf{X}], \mathsf{f}[\mathsf{RO}_\mathsf{H},\mathsf{X}]\sigma),$$

for a simulator σ. If there exists a simulator σ such that for all efficient distinguishers and all contexts $(\mathsf{f},\mathsf{X}) \in \mathcal{C}$, the RO-CRI advantage is negligible, we say that H is $\mathcal{C}$ random-oracle context-restricted indifferentiable or, for short, $\mathcal{C}$ indifferentiable.

The following lemma implies that for non-interactive contexts this definition is equivalent to the game-based definition of UCE security, if we fix the simulator to $\sigma_{\mathtt{H}}$.

Lemma 4.5. *Let $\mathcal{S}$ denote the set of all UCE-sources and $\phi\colon \Sigma^{\mathsf{ni}} \times \Phi^{\mathsf{ni}} \to \mathcal{S}$ the surjective function from Lemma 4.3. For every distinguisher D, there is a distinguisher D' (with essentially the same efficiency) with*

$$\forall (\mathsf{f},\mathsf{X}) \in \Sigma^{\mathsf{ni}} \times \Phi^{\mathsf{ni}}\colon \mathbf{Adv}^{\mathrm{RO-CRI}}_{\mathsf{H},\mathsf{f},\mathsf{X},\sigma_{\mathtt{H}}}(\mathsf{D}) = \mathbf{Adv}^{\mathrm{uce}}_{H,\phi(\mathsf{f},\mathsf{X}),\mathsf{D}'}$$

Conversely, for every distinguisher D' there is a distinguisher D (with essentially the same efficiency) such that for all $(\mathsf{f},\mathsf{X}) \in \Sigma^{\mathsf{ni}} \times \Phi^{\mathsf{ni}}$ we have $\mathbf{Adv}^{\mathrm{uce}}_{H,\phi(\mathsf{f},\mathsf{X}),\mathsf{D}'} = \mathbf{Adv}^{\mathrm{RO-CRI}}_{\mathsf{H},\mathsf{f},\mathsf{X},\sigma_{\mathtt{H}}}(\mathsf{D})$.

Proof. For every distinguisher D for $\mathbf{Adv}^{\mathrm{RO-CRI}}_{\mathsf{H},\mathsf{f},\mathsf{X},\sigma_{\mathtt{H}}}(\mathsf{D})$ we can construct a distinguisher D' using a wrapper around D as follows: if D queries the interface $\mathtt{E}$ of the hash resource (for the key) or X we return hk or z, respectively; if D queries the outer interface of f, then y is returned. The bit b' is then set to the output bit of D. The key observation is that the resources $\mathsf{f}[\mathsf{H},\mathsf{X}]$ and $\mathsf{f}[\mathsf{RO},\mathsf{X}]\sigma_{\mathtt{H}}$ are independent to the order in which D does those queries. It is now easy to see that $\mathbf{Adv}^{\mathrm{RO-CRI}}_{\mathsf{H},\mathsf{f},\mathsf{X},\sigma_{\mathtt{H}}}(\mathsf{D}) = \mathbf{Adv}^{\mathrm{uce}}_{H,\phi(\mathsf{f},\mathsf{X}),\mathsf{D}'}$.

The reverse direction works with an analogous wrapper that first queries the system to obtain hk, z, and y. It then invokes D' with hk and $L = (y, z)$ as inputs and outputs the bit b'. □

The following theorem relates this particular context-restricted indifferentiability statement to the UCE security of the hash function H. More concretely, we show that every possible UCE family $\mathcal{S}^x$, where $x \in \{\mathrm{sup}, \mathrm{cup}, \mathrm{srs}, \mathrm{crs}, \mathrm{splt}, \ldots\}$, there exists a set of context for which the RO-CRI statement and the UCE statement are equivalent.

Theorem 4.6. *Let $\mathcal{D}$ denote the set of all efficient distinguishers. For every family $\mathcal{S}^x$ of UCE sources, there exists a set of contexts $\mathcal{C}^x$ such that $\mathbf{Adv}^{\mathrm{RO-CRI}}_{\mathsf{H},\mathsf{f},\mathsf{X},\sigma_{\mathtt{H}}}(\mathsf{D})$ is negligible for every $D \in \mathcal{D}$ and every context $(\mathsf{f},\mathsf{X}) \in \mathcal{C}^x$ if and only if $\mathbf{Adv}^{\mathrm{uce}}_{H,S,D}(\cdot)$ is negligible for all $(S, D) \in \mathcal{S}^x \times \mathcal{D}$.*

Proof. Using the surjectivity of ϕ (Lemma 4.3), we have that for any family of UCE sources $\mathcal{S}^x$ we can define $\mathcal{C}^x := \phi^{-1}(\mathcal{S}^x)$ such that $\phi(\mathcal{C}^x) = \mathcal{S}^x$. Hence, by Lemma 4.5 we have that $\mathbf{Adv}^{\mathrm{RO-CRI}}_{\mathsf{H},\mathsf{f},\mathsf{X},\sigma_{\mathtt{H}}}(\mathsf{D})$ is negligible for all efficient distinguishers $\mathsf{D} \in \mathcal{D}$ and all contexts $(\mathsf{f}, \mathsf{X}) \in \mathcal{C}^x$ iff $\mathbf{Adv}^{\mathrm{uce}}_{H,S,D}(\cdot)$ is negligible for all $(S, D) \in \mathcal{S}^x \times \mathcal{D}$. □

This theorem implies that instead of viewing the source as the first stage of an adversary, one can view it as the set of contexts in which the hash function can safely be used. The following corollary establishes the unidirectional implication from UCE-security to context-restricted indifferentiability. The reverse direction does not necessarily hold, since the context-restricted indifferentiability notion allows for different simulators than the natural one $\sigma_{\mathtt{H}}$.

Corollary 4.7. *Let $\mathcal{D}$ denote the set of all efficient distinguishers. For every family $\mathcal{S}^x$ of UCE sources, there exists a set of contexts $\mathcal{C}^x$ such that $\mathbf{Adv}^{\mathrm{uce}}_{H,S,D}(\cdot)$ is negligible for all $(S, D) \in \mathcal{S}^x \times \mathcal{D}$, then $\mathsf{HK}_\mathsf{H} \overset{\mathsf{hash}_\mathsf{H},\mathcal{C}^x}{\underset{\mathsf{cr}}{\Longmapsto}} \mathsf{RO}_\mathsf{H}$.*

Proof. This follows directly from Definitions 3.2 and 4.4 and Theorem 4.6. □

5 The relation between CRI and public-seed pseudo-random permutations

In [ST17] Soni and Tessaro introduce the notion of public-seed pseudorandom permutations (psPRP) that are inspired by UCE. In fact, they introduce a generalization of UCE, called public-seed pseudorandomness, of which both psPRP and UCE are instantiations. In the following, we give an analogous equivalence result to the one of Section 4 between context-restricted indifferentiability and the general public-seed pseudorandomness notion. The equivalence for the psPRP notion then just follows as a trivial corollary.

5.1 Public-seed pseudorandomness

We first briefly recap the main definitions of public-seed pseudorandomness as introduced in [ST17]. The authors first introduce the notion of an ideal primitive, of which both random oracles and ideal random permutations are instantiations of.

Definition 5.1. An ideal primitive is a pair $I = (\Sigma, D)$, where $\Sigma = \{\Sigma_\lambda\}_{\lambda \in \mathbb{N}}$ is a family of sets of functions (such that all functions have the same domain and range), and $D = \{D_\lambda\}_{\lambda \in \mathbb{N}}$is a family of probability distributions, where the range of D_λ is a subset of Σ_λ for all $\lambda \in \mathbb{N}$. The ideal primitive I, once the security parameter λ is fixed, should be thought of as an oracle that initially samples a function I as its initial state according to D_λ from Σ_λ. Then, I provides access to I via queries i.e. on input x it returns $I(x)$.

Moreover, the authors of [ST17] then define the following notion of Σ-compatible function families. A function family corresponds to an algorithm that generalizes hash functions and pseudo-random permutations.

Definition 5.2. A function family $\mathrm{F} = (\mathrm{Kg}, \mathrm{Eval})$ consists of a key (or seed) generation algorithm F.Kg and an evaluation algorithm F.Eval.

- F.Kg is a randomized algorithm that on input the unary representation of the security parameter λ returns a key k, and we let $[\mathrm{F.Kg}(1^\lambda)]$ denote the set of all possible outputs of $\mathrm{F.Kg}(1^\lambda)$.
- F.Eval is a deterministic algorithm that takes three inputs; the security parameter in unary form 1^λ, a key $k \in [\mathrm{F.Kg}(1^\lambda)]$ and a query x such that $\mathrm{F.Eval}(1^\lambda, k, \cdot)$ implements a function that maps queries x to $\mathrm{F.Eval}(1^\lambda, k, x)$.

We say that F is efficient if both Kg and Eval are polynomial-time algorithms.

The goal of such a function family F is then to implement an ideal primitive I with respect to the UCE-like security game depicted in Algorithm 13, considering an adversary that is split into a source S and a distinguisher D. In contrast to the original definition, we only consider the game for a single session, which can easily be related to the multi-session one using a standard hybrid argument.

Algorithm 13 The public-seed pseudorandomness game (single-session)

```
function MAIN $\mathrm{psPR}_{F,I}^{S,D}(\lambda)$
    $b \xleftarrow{\$} \{0,1\}$
    $k \xleftarrow{\$} \mathrm{F.Kg}(1^\lambda)$
    $f \xleftarrow{\$} \mathrm{I}_\lambda$
    $L \xleftarrow{\$} S^{\mathcal{O}}(1^\lambda)$
    $b' \xleftarrow{\$} D(1^\lambda, k, L)$
    return $(b' = b)$

function $\mathcal{O}(x)$
    if $b = 1$ then
        return $\mathrm{F.Eval}(1^\lambda, k, x)$
    else
        return $f(x)$
```

Finally, Soni and Tessaro define the pspr-advantage as follows:

$$\mathbf{Adv}_{\mathrm{F},S,D}^{\mathsf{pspr}[I]}(\lambda) = 2\Pr\left[\mathsf{psPRS}_{\mathrm{F},I}^{S,D}(\lambda)\right] - 1.$$

5.2 Ideal primitives and function families in CRI

In the following section, we argue that every ideal primitive I can be understood as an ideal resource of an CRI statement, and every function family F as an pair of real resource and protocol, respectively. For simplicity, we ignore the security parameter λ in the following.

For every ideal primitive I and for every function family $\mathrm{F} = (\mathrm{Kg}, \mathrm{Eval})$, denote the corresponding resource and converters depicted in Figure 6. Moreover, we also define the simulator σ_{F}, which simply chooses a key according to Kg as well.

```
Resource KG_F

Initialization
    $k \xleftarrow{\$} \mathrm{F.Kg}(1^\lambda)$

Interface i ∈ {A, E}
Input: getkey
    output $k$ at i
```

```
Resource I

Initialization
    $f \xleftarrow{\$} I_\lambda$

Interface A
Input: (eval, $x$)
    output $f(x)$ at A
```

```
Converter eval_F

Initialization
    output getkey at in
    let $k$ denote the result

Interface out
Input: (eval, $x$)
    output $\mathrm{F.Eval}(1^\lambda, k, x)$ at out
```

```
Converter σ_F

Initialization
    $k \xleftarrow{\$} \mathrm{F.Kg}(1^\lambda)$

Interface out
Input: getkey
    output $k$ at out
```

Figure 6: The corresponding resources and converters.

5.3 The equivalence

We now show, that for the specific simulator $\sigma_{\mathtt{KG}}$, if for every specific context (f, X) the distinguishing problem of context-restricted indifferentiability is hard, then the UCE game with the fixed source $\phi(\mathsf{f}, \mathsf{X})$ is hard as well, and vice versa. In order to relate more directly, we introduce the psRP-CRI advantage.

Definition 5.3. We define the *public-seed pseudorandomness context-restricted indifferentiability (psRP-CRI)* advantage of a distinguisher D on a hash function H in a context (f, X) as

$$\mathbf{Adv}^{\mathrm{psPR-CRI}}_{\mathrm{F},I,\mathsf{f},\mathsf{X},\sigma}(\mathsf{D}) := \Delta^{\mathsf{D}}(\mathsf{f}[\mathsf{eval}_{\mathsf{F}}\mathsf{KG}_{\mathsf{F}}, \mathsf{X}], \mathsf{f}[\mathsf{I}, \mathsf{X}]\sigma),$$

for a simulator σ.

The following lemma implies that for non-interactive contexts this definition is equivalent to the game-based definition of UCE security, if we fix the simulator to σ_{F}.

Lemma 5.4. *Let $\mathcal{S}$ denote the set of all psPR-sources and $\phi\colon \Sigma^{\mathsf{ni}} \times \Phi^{\mathsf{ni}} \to \mathcal{S}$ the surjective function from Lemma 4.3. For every distinguisher D, there is a distinguisher D' (with essentially the same efficiency) with*

$$\forall (\mathsf{f}, \mathsf{X}) \in \Sigma^{\mathsf{ni}} \times \Phi^{\mathsf{ni}} \colon \mathbf{Adv}^{\mathrm{psPR-CRI}}_{\mathrm{F},I,\mathsf{f},\mathsf{X},\sigma_{\mathrm{F}}}(\mathsf{D}) = \mathbf{Adv}^{\mathsf{pspr}[I]}_{\mathrm{F},\phi(\mathsf{f},\mathsf{X}),\mathsf{D}'}$$

Conversely, for every distinguisher D' there is a distinguisher D (with essentially the same efficiency) such that for all $(\mathsf{f}, \mathsf{X}) \in \Sigma^{\mathsf{ni}} \times \Phi^{\mathsf{ni}}$ we have $\mathbf{Adv}^{\mathsf{pspr}[I]}_{\mathrm{F},\phi(\mathsf{f},\mathsf{X}),\mathsf{D}'} = \cdot \mathbf{Adv}^{\mathrm{psPR-CRI}}_{\mathrm{F},I,\mathsf{f},\mathsf{X},\sigma_{\mathrm{F}}}(\mathsf{D})$.

Proof. The proof is analogous to the one of Lemma 4.5. □

We can now state the main result of this section, relating the public-seed pseudorandomness game to context-restricted indifferentiability.

Theorem 5.5. *Let $\mathcal{D}$ denote the set of all efficient distinguishers. For every family $\mathcal{S}^x$ of UCE sources, there exists a set of contexts $\mathcal{C}^x$ such that $\mathbf{Adv}^{\mathrm{RO-CRI}}_{\mathsf{H},\mathsf{f},\mathsf{X},\sigma_{\mathtt{H}}}(\mathsf{D})$ is negligible for every $D \in \mathcal{D}$ and every context $(\mathsf{f}, \mathsf{X}) \in \mathcal{C}^x$ if and only if $\mathbf{Adv}^{\mathrm{uce}}_{H,S,D}(\cdot)$ is negligible for all $(S, D) \in \mathcal{S}^x \times \mathcal{D}$.*

Proof. The proof is analogous to the one of Theorem 4.6. □

This demonstrates that not only UCE is a special case of CRI but also the more general notion of psPR is still a special case of CRI, where each ideal primitive and function family correspond to the ideal and real world, respectively. Similarly to UCE, the psPR notion is still non-interactive and essentially hard-codes a specific simulator in the security game.

6 Generalization of split security

6.1 Split security

The split-source UCE class was initially proposed by Bellare et al. to prevent the indistinguishability obfuscation attack in the computational setting. The general idea is that the source needs to be further restricted in the way it operates so that $\mathrm{Obfs}(H(\,\cdot\,, x) = y)$ can no longer be computed by the source. Therefore, the split source class demands that the source can be divided into two parts (S_0, S_1). The first part S_0 chooses a vector x of query points (without having access to the oracle). The second part of the source S_1 then just gets the evaluations $y_i := Hash(x_i)$ (without having access to the oracle). Thus, no part of the source knows both x_i and its evaluation y_i, preventing the iO attack. A formal description of the split source $S := Splt[S_0, S_1]$ is found in Figure 7.

In the previous section we have seen that for every UCE class $\mathcal{S}^{\mathsf{x}}$, one can define a set of contexts as[5] $\mathcal{C}^{\mathsf{x}} := \phi^{-1}(\mathcal{S}^{\mathsf{x}})$ such that $\phi(\mathcal{C}^{\mathsf{x}}) = \mathcal{S}^{\mathsf{x}}$. Since the function ϕ is not bijective, however, there is no need to include all pre-images. Especially, if a UCE class has a structural property that translates to a separation between the filter f and the resource X of a context, then this can be reflected in the choice of the pre-image. Split-security is a case where this comes in handy, as we can model the structural requirement of split-security as following: S_0 can be mapped to the resource X that outputs the queries at the interface `A` and the leakage at the interface `E`, as it does not need access to the hash oracle. More generally, S_0 can be translated to a resource of the following type.

Definition 6.1. A *randomness seed* X with n outputs is a resource that initially draws random values $Y_1, \ldots, Y_n$ and Z according to some joint distribution. Then, it accepts at the interface `E` a single trigger query (usually called `retrieve`) that is answered with Z, and at the interface `A` n trigger queries answered with Y_1 to Y_n. Let Φ^{seed}_n denote the set of all randomness seeds with at most n outputs at the interface `A`.

The structural property that each of the queries is evaluated and then passed to the source S_1 can be expressed using the fixed filter $\mathsf{f}^{\mathsf{splt}}$ that is described in Figure 7.

Remark. It is worth observing that considering a fixed filter makes verifying whether a protocol can be implemented through this filter relatively easy. This aligns well with the general idea of making explicit for which protocols a resource can be used. Moreover, this factorization implies that any unpredictability requirement can be expressed as a property of the resource X only.

Algorithm 22 Splt SOURCE

```
function SPLT SOURCE$^{\text{HASH}}(1^\lambda)$
    $(L_0, x) \xleftarrow{\$} S_0(1^\lambda)$
    for $i = 1, \dots, |x|$ do
        $y[i] \leftarrow \text{HASH}(x[i])$
    $L_1 \xleftarrow{\$} S_1(1^\lambda, y)$
    $L \leftarrow (L_0, L_1)$
    output $L$
```

Converter $\mathsf{f}^{\mathsf{splt}}$

```
Outer Interface
Input: retrieve
    output retrieve at in.X
    let $x$ denote the result
    output $x$ at in.H
    let $y$ denote the result
    output $y$ at out
```

Figure 7: The definition of split sources in UCE.

The following lemma formally states that this translation of split-security is faithful.[6]

Lemma 6.2. *Let $\mathcal{C}_n^{\mathsf{splt}} := \{\mathsf{f} \circ \mathsf{f}^{\mathsf{splt}} \mid \mathsf{f} \in \Sigma^{\mathsf{ni}}\} \times \Phi_n^{\mathsf{seed}}$ and let $\mathcal{S}^{\mathsf{n}}$ denote the class of all UCE sources making at most n oracle queries. For the surjective function ϕ from Lemma 4.3, we have $\phi(\mathcal{C}_n^{\mathsf{splt}}) = \mathcal{S}^{\mathsf{splt}} \cap \mathcal{S}^{\mathsf{n}}$.*

Proof. (Sketch) First, we prove that $\phi(\mathcal{C}_n^{\mathsf{splt}}) \subseteq \mathcal{S}^{\mathsf{splt}} \cap \mathcal{S}^{\mathsf{n}}$, i.e., every context $(\mathsf{f} \circ \mathsf{f}^{\mathsf{splt}}, \mathsf{X}) \in \mathcal{C}_n^{\mathsf{splt}}$ is mapped to a UCE source in $\mathcal{S}^{\mathsf{splt}} \cap \mathcal{S}^{\mathsf{n}}$ by ϕ. We define S_0 to be the source which initially queries z at the interface `E` and all values $x = x_1, \dots, x_n$ at the interface `A` of X and set $L_0 = z$. The source S_1 then internally repeatedly queries f to obtain L_1. Whenever f outputs a query `retrieve` towards $\mathsf{f}^{\mathsf{splt}}$, then S_1 answers by using the next value y_i. Observe that, by definition of Φ_n^{seed}, obtaining all queries $x_1, \dots, x_n$ from X at interface `A` on demand or at the beginning and storing the results $y_1 = H(k, x_1), \dots, y_n = H(k, x_n)$ is equivalent. Thus, it is easy to verify that $\phi(\mathsf{f} \circ \mathsf{f}^{\mathsf{splt}}, \mathsf{X}) = \mathit{Splt}[S_0, S_1]$. Moreover, this source does at most n queries to the hash oracle.

It remains to show that $\phi(\mathcal{C}_n^{\mathsf{splt}}) \supseteq \mathcal{S}^{\mathsf{splt}} \cap \mathcal{S}^{\mathsf{n}}$, i.e., for every split source there exists at least one context that maps to this source. It is easy to see that S_0 can be embedded accordingly in a resource $\mathsf{X} \in \Phi_n^{\mathsf{seed}}$ and S_1 in a filter $\mathsf{f} \in \Sigma^{\mathsf{ni}}$ such that $\phi(\mathsf{f} \circ \mathsf{f}^{\mathsf{splt}}, \mathsf{X}) = \mathit{Splt}[S_0, S_1]$, concluding the proof. □

6.2 Strong-split security

Split sources have several limitations. First, the distinguisher cannot influence the queries at all and, thus, all queries must be solely determined by the honest parties. This prevents, for example, queries like $H(hk, x||a)$ where a is a value which can be chosen by the distinguisher (e.g. a is transmitted over an insecure channel) even if x is unpredictable. In the following section, we introduce a generalization of split-security, called *strong-split* security, to address this limitation.

Note that this limitation is not specific to split-security, but is inherent to the traditional UCE-game. In their work [FM16] on Interactive Computational Extractors (ICEs), Farshim and Mittelbach have proposed an alternative relaxation of this issue. In Section 6.5, we show that ICE security implies strong-split context-restricted indifferentiability for statistical unpredictability. In addition, in Section 6.4 we present a furhter generalization of strong-split security, which allows for nested queries like $H(hk, H(hk, x))$.

[5] We use the following shorthand for surjective functions: $f^{-1}(y) = \{x \mid f(x) = y\}$ and $f^{-1}(\mathcal{Y}) = \{x \mid f(x) \in \mathcal{Y}\}$.

[6] Here, we do not consider split sources with a polynomial number of queries, although those could easily be phrased by adapting the restriction on the seed accordingly.

```
Converter $\mathsf{f}_p^{\mathsf{s-splt}}$

Outer Interface
Input: $(\texttt{retrieve}, f_1, \dots f_p) \in \mathcal{I}^p_{\mathcal{X}\times\mathcal{A}\to H.\mathcal{X}}$
    output retrieve at in.X
    let $(x, a)$ denote the result
    for $i = 1, \dots, p$ do
        $y[i] \leftarrow f_i(x, a)$
    if $\forall i \neq j \colon y[i] \neq y[j]$ then
        for $i = 1, \dots, p$ do
            output $y[i]$ at in.H
            let $z[i]$ denote the result
        output $z$ at out
    else
        output $\perp^p$ at out
```

Figure 8: The filter $\mathsf{f}_p^{\mathsf{s-splt}}$.

We propose the strong-split source RO-CRI context set, which requires only the entropy seed to be non-interactive. The filter is allowed to take input from the distinguisher. This ensures that the queries can depend on the hash key, whereas the leakage of the entropy seed cannot. The filter $\mathsf{f}_p^{\mathsf{s-splt}}$ is described in Figure 8: the filter expects the entropy seed to output a pair (x, a), then accepts an efficiently computable function f that is injective in the first argument, from the distinguisher, and outputs $f(x, a)$ to the hash oracle. Here, f being injective guarantees that $f(x, a)$ is at least as unpredictable as x. We denote the set of all such functions by $\mathcal{I}_{\mathcal{X}\times\mathcal{A}\to H.\mathcal{X}}$. The result is then simply output at the outer interface. The filter $\mathsf{f}_p^{\mathsf{s-splt}}$ can then be combined with an arbitrary randomness seed to obtain a strong-split RO-CRI context. It is easy to see that strong-split sources are a strict generalization of split sources for any $p \geq 1$, i.e. $\mathcal{C}_n^{\mathsf{splt}} \subsetneq \mathcal{C}_{n,p}^{\mathsf{s-splt}}$.

Definition 6.3. The *strong-split* RO-CRI context set is the set of filters and non-interactive randomness seed pairs of which the filter can be factorized into $\mathsf{f}_p^{\mathsf{s-splt}}$ followed by an arbitrary filter. Formally, $\mathcal{C}_{n,p}^{\mathsf{s-splt}} := \{\mathsf{f} \circ \mathsf{f}_p^{\mathsf{s-splt}} \mid \mathsf{f} \in \Sigma\} \times \Phi_n^{\mathsf{seed}}$.

Analogous to split-security, strong-split security is not a sufficient restriction to avoid trivial impossibility results. Rather, these notions are meant to be combined with a notion of unpredictability or reset-security. Moreover, it was pointed out in [BM14], the queries of a split-source must be distinct; otherwise arbitrary information can be communicated to the second stage. However, for strong-split security, requiring the seed to output distinct unpredictable values is still insufficient to guarantee the security: for instance, if the seed outputs x and $x + 1$, then the distinguisher can easily choose f and g such that $f(x, a_1) = g(x + 1, a_2)$. As a consequence, we introduce a suitable notion of statistical unpredictability in the next section that disallows strongly correlated outputs. To nevertheless allow for protocols that do query the hash function at correlated positions, such as $H(x||1)$ and $H(x||2)$, we introduced the explicit parameter p of the number of correlated queries.

6.3 Strict min-entropy seeds

We now define an information-theoretic restriction on the seed, which we call *strict min-entropy seeds*. This restriction is analogous to statistical unpredictability for UCE. Similar to Farshim and

Mittelbach [FM16] we choose to focus on statistical rather than computational unpredictability to ensure that our notion excludes interactive version of the attack highlighted in [BFM14].[7] Instead of considering the usual notion of statistical unpredictability, we focus in this work on min-entropy.[8] More concretely, we consider seeds whose outputs at interface $\mathtt{A}$ consist of pairs (Y_i, A_i), with A_i being an auxiliary value, such that Y_i has a certain amount of *average conditional min-entropy* given the leakage Z and all previous queries.

Definition 6.4. A non-interactive resource is said to be a *strict min-entropy k-bit seed*, denoted $\mathsf{X} \in \Phi^{\mathsf{s-me}}_{n,k}$, if $\mathsf{X} \in \Phi^{\mathsf{seed}}_{n}$ with $\mathcal{Y} \times \mathcal{A}$ as the output domain of interface $\mathtt{A}$, and the following property holds:

$$\forall i \le n\colon\ \tilde{H}_\infty\big(Y_i \,\big|\, \{Y_j\}_{j<i}, \{A_j\}_{j\le i}, Z\big) \ \ge\ k.$$

Moreover, let $\mathcal{C}^{\mathsf{s-me}}_{n,k} := \Sigma \times \Phi^{\mathsf{s-me}}_{n,k}$ denote the set of all strict min-entropy k-bit contexts.

When combining stong-split security with strict min-entropy seeds, the security of strong-split sources does not depend on the number n of sequential queries output by the seed, as stated in the following lemma.

Lemma 6.5. *If H is a $\mathcal{C}^{\mathsf{s-splt}}_{1,p}$ indifferentiable hash function for strict k-bit min-entropy seeds with a single output, then H is also $\mathcal{C}^{\mathsf{s-splt}}_{n,p}$ indifferentiable for strict k-bit min-entropy sources with n outputs.*

Formally, let $\mathcal{D}$ denote the set of distinguishers. Then there exists a reduction $\rho\colon \mathcal{D} \times \left(\mathcal{C}^{\mathsf{s-splt}}_{n,p} \cap \mathcal{C}^{\mathsf{s-me}}_{n,k}\right) \to \mathcal{D}$ (translating the distinguisher) and a translation of the context $\psi\colon \mathcal{C}^{\mathsf{s-splt}}_{n,p} \cap \mathcal{C}^{\mathsf{s-me}}_{n,k} \to \mathcal{C}^{\mathsf{s-splt}}_{1,p} \cap \mathcal{C}^{\mathsf{s-me}}_{1,k}$, such that for every $(\mathsf{f}, \mathsf{X}) \in \mathcal{C}^{\mathsf{s-splt}}_{n,p} \cap \mathcal{C}^{\mathsf{s-me}}_{n,k}$ we have

$$\mathbf{Adv}^{\mathrm{RO-CRI}}_{\mathsf{H},\mathsf{f},\mathsf{X},\sigma}(\mathsf{D}) \le \binom{np}{2} 2^{-k} + n \cdot \mathbf{Adv}^{\mathrm{RO-CRI}}_{\mathsf{H},\mathsf{f}',\mathsf{X}',\sigma}(\mathsf{D}')$$

with $\mathsf{D}' := \rho(\mathsf{D}, \mathsf{f}, \mathsf{X})$ and $(\mathsf{f}', \mathsf{X}') := \psi(\mathsf{f}, \mathsf{X})$.

Sketch. The proof works very similarly to the one of Lemma 6.6 presented in Appendix A; therefore, we only provide a brief sketch. As a first hybrid, we introduce a variant that uses a beacon instead of a random oracle, where a beacon is a resource with the same interface as the random oracle but always answers using fresh randomness even for repeated queries. Distinguishing this hybrid system from the ideal system (that uses the random oracle) can be bounded with the collision probability for the inputs. Since every of the input has k bits of conditional min-entropy, given all previous inputs, the collision for any of them can be bounded with 2^{-k} (c.f. the proof below) and there are at most np queries in total. Hence, the total distinction advantage can be bounded by $\binom{np}{2}2^{-k}$.

It remains to bound the distinction advantage between the real system (using the hash function) and our hybrid system (using the beacon) using the strong-split security for a single message. This can be shown by a simple hybrid-argument with n additional hybrids where the i-th randomness seeds X_i outputs the i-th message Y_i at interface $\mathtt{A}$ and the messages $Y_1, \ldots, Y_{i-1}$

[7] We would like to stress that this is an additional restriction independent of strong-split security. Assuming iO, however, the same attack from [BFM14] would also apply if we combine strong-split security with a computational unpredictability notion only.

[8] Having k bits of min-entropy corresponds to no adversary being able to guess the value with probability greater than 2^{-k} in a single attempt.

```
Converter $\mathsf{f}_{p,r}^{\mathsf{r-splt}}$

Initialization
    $b \leftarrow \perp^p$, $c \leftarrow 0$

Outer Interface
Input: get
    output $b$ at out
    $b \leftarrow \perp^p$

Input: $(\mathtt{query}, f_1, \dots, f_p) \in \mathcal{I}^p_{\mathcal{X},\mathcal{A}\to H.\mathcal{X}}$
    $c \leftarrow 1$
    output retrieve at in.X
    let $(x, a)$ denote the result
    for $i = 1, \dots, p$ do
        $y[i] \leftarrow f_i(x, a)$
    if $\forall i \neq j \colon y[i] \neq y[j]$ then
        for $i = 1, \dots, n$ do
            output $y[i]$ at in.H
            let $b[i]$ denote the result
    else
        $b \leftarrow \perp^p$

Input: $(\mathtt{repeat}, f_1, \dots, f_p) \in \mathcal{I}^p_{H.\mathcal{Y}\to H.\mathcal{X}}$
    $c \leftarrow c + 1$
    if $c \leq r \ \wedge \ b \neq \perp^p$ then
        for $i = 1, \dots, p$ do
            output $f_i(b[i])$ at in.H
            let $b[i]$ denote the result
    else
        $b \leftarrow \perp^p$
```

Figure 9: The filter $\mathsf{f}_{p,r}^{\mathsf{r-splt}}$.

as additional leakage at interface $\mathtt{E}$. The hybrid then answers the first $i-1$ queries by computing the hash function itself, the i-th message by actually querying the attached system (that uses either the hash function or the beacon), and the remaining queries by uniform random values, simulating the beacon. Defining the resource X' to be the one that chooses uniformly at random among $X_1, \dots, X_n$ yields the desired bound: $n \cdot \mathbf{Adv}^{\mathrm{RO-CRI}}_{\mathsf{H},\mathsf{f}',\mathsf{X}',\sigma_{\mathsf{H}}}(\mathsf{D}')$. □

6.4 The repeated-split source context set

We now further generalize our strong-split source class, to allow for repeated queries, such as $H\big(hk, H(hk, x||1)||2\big)$. The key idea is to introduce a buffer which stores the results obtained from the hash function. The distinguisher can then choose whether it wants to see those values, or whether it wants to use them as a new query. The filter $\mathsf{f}_{p,r}^{\mathsf{r-splt}}$ is depicted in Figure 9. The parameter r determines the maximal allowed nesting depth. Analogously to the strong-split source, we can then define the $\mathcal{C}_{n,p,r}^{\mathsf{r-splt}}$ context set based on this filter as $\mathcal{C}_{n,p,r}^{\mathsf{r-splt}} := \{\mathsf{f} \circ \mathsf{f}_{p,r}^{\mathsf{r-splt}} \mid \mathsf{f} \in \Sigma\} \times \Phi_n^{\mathsf{seed}}$.

We now prove that strong-split context-restricted indifferentiability implies repeated-split context-restricted indifferentiability when furthermore restricted to strict min-entropy sources. This allows to analyze hash functions only for strong-split security, but use them in contexts where repeated-split security is needed to implement a certain protocol.

Lemma 6.6. *Let $k' := \min(k, \log|H.\mathcal{Y}|)$. If H is a $\mathcal{C}^{\mathsf{s-splt}}$ indifferentiable hash function for strict k'-bit min-entropy sources, then H is also $\mathcal{C}^{\mathsf{r-splt}}$ indifferentiable hash function for strict k-bit min-entropy sources.*

Formally, let $\mathcal{D}$ denote the set of distinguishers. Then there exists a reduction $\rho \colon \mathcal{D} \times \left(\mathcal{C}_{n,p,r}^{\mathsf{r-splt}} \cap \mathcal{C}_{n,k}^{\mathsf{s-me}}\right) \to \mathcal{D}$ and a translation of the context $\psi \colon \mathcal{C}_{n,p,r}^{\mathsf{r-splt}} \cap \mathcal{C}_{n,k}^{\mathsf{s-me}} \to \mathcal{C}_{n,p}^{\mathsf{s-splt}} \cap \mathcal{C}_{n,k'}^{\mathsf{s-me}}$, such *that for every* $(\mathsf{f}, \mathsf{X}) \in \mathcal{C}^{\mathsf{r-splt}}_{n,p,r} \cap \mathcal{C}^{\mathsf{s-me}}_{n,k}$ *we have*

$$\mathbf{Adv}^{\mathrm{RO-CRI}}_{\mathsf{H},\mathsf{f},\mathsf{X},\sigma}(\mathsf{D}) \leq \binom{npr}{2} 2^{-(k'-1)} + r \cdot \mathbf{Adv}^{\mathrm{RO-CRI}}_{\mathsf{H},\mathsf{f}',\mathsf{X}',\sigma}(\mathsf{D}')$$

with $\mathsf{D}' := \rho(\mathsf{D}, \mathsf{f}, \mathsf{X})$ *and* $(\mathsf{f}', \mathsf{X}') := \psi(\mathsf{f}, \mathsf{X})$.

Proof. The proof can be found in Appendix A.

6.5 The relation between ICE and strong-split context-restricted indifferentiability

In this section we discuss the relation between RO-CRI and the ICE framework introduced in [FM16]. More concretely, we show that ICE security implies strong-split context-restricted indifferentiability for statistical unpredictability, as phrased in Theorem 6.7. Using this relation between the two frameworks, we especially inherit the random oracle feasibility result from the ICE framework.

The reverse direction, whether strong-split RO-CRI implies some natural notion of ICE, remains an interesting open problem. In general, there seems to be no natural mapping from ICE to RO-CRI. This can be explained by the fundamentally different motivation behind introducing this two generalizations of UCE: ICE tried to allow interaction by making the two stages of UCE more symmetric, whereas RO-CRI exploits the asymmetry of UCE to separate them even further into the protocol of the honest party and the regular distinguisher from indifferentiability.

In terms of random-oracle feasibility, this places RO-CRI as an intermediate notion between the original UCE notion and the stronger ICE notion, while it is still open whether a true separation between those frameworks exists.

Theorem 6.7. *Let* H *denote a keyed hash function where the key-space is exponential in the security parameter. If* $\mathsf{H} \in \mathrm{ICE}[\mathcal{C}^{sup}]$*, then* H *is* $\mathcal{C}^{\mathsf{s-splt}}_{n,p} \cap \mathcal{C}^{\mathsf{s-me}}_{n,k}$ *context-restricted indifferentiable from a random oracle for any polynomial* n *and* p*, and* k *such that the guessing probability is negligible.*

Proof. We sketch a proof that for the fixed simulator $\sigma_{\mathtt{H}}$, every context $(\mathsf{f}, \mathsf{X}) \in \mathcal{C}^{\mathsf{s-splt}}_{n,p} \cap \mathcal{C}^{\mathsf{s-me}}_{n,k}$ and distinguisher D can be turned into a pair of equivalent ICE distinguishers D_1 and D_2. Let D_1 internally emulates the distinguisher D and works as follows:

- It initially chooses the hash key hk uniformly at random (as $\sigma_{\mathtt{H}}$) and writes it into the buffer using a WRITE query. This is the only WRITE query D_1 does.
- In every round, it uses obtains the answer from L_2 and passes this to the distinguisher D to obtain the next query. According whether D queries the interface $\mathtt{A}$ with the function f or obtains the leakage at interface $\mathtt{E}$, it produces an appropriate output L_1, either $(\mathtt{A}, f)$ or $(\mathtt{E})$.
- If the distinguisher D outputs the decision bit, D_1 outputs the same bit.

The second distinguisher D_2 internally emulates the context $(\mathsf{f}^{\mathsf{s-splt}}, \mathsf{X})$. It works as follows:

- In every round it inspects the value L_1.

  - If L_1 is of the form $(\mathtt{A}, f)$, it passes f to the internal emulation of the context, to obtain the value x that would be queried to the hash function. It then writes x to the buffer and queries the hash function. The resulting value y is returned as L_2.
  - If L_2 is of the form $(\mathtt{E})$, then it queries the interface $\mathtt{E}$ of the internal resource X and returns the result as L_2.

- It always sets $b_2 = 0$.

It is easy to see that the ICE game now behaves exactly the same as the RO-CRI system. Moreover, the queries of D_2 are exactly as unpredictable given the state and randomness of D_1 as are the queries in the RO-CRI system given access to the interface $\mathtt{E}$. Finally, if the hash key hk is unpredictable, then none of the queries of D_1 can be predicted given the complete state and randomness of D_2. This concludes the proof. □

7 New Types of Soundness Statements for Hash Function Constructions

In the following subsection we propose to use RO-CRI as a new tool to analyze the soundness of hash function constructions. As an example, we look at the Merkle–Damgård construction, which is one of the most fundamental ones. We prove that for a single message source the constructed hash function is statistically $\mathcal{C}^{\mathsf{splt}}$ indifferentiable if the underlying compression function is statistically $\mathcal{C}^{\mathsf{s-splt}}$ indifferentiable. As a technical tool, we show a lemma on min-entropy splitting.

7.1 CRI of the Merkle–Damgård Construction

In this section we prove that the Merkle–Damgård construction is split context-restricted indifferentiable if the underlying compression function is strong-split context-restricted indifferentiable for blocks with sufficient min-entropy. Here "sufficient" means that the compression function must be strong-split secure for a single message with $\min(k, n)$ bits of min-entropy, where k denotes the maximal min-entropy of any block of the message, and n the output length of the compression function.

We first formalize the class of sources, which outputs a single binary message X that can be split into block of size n, i.e., $X = X_1||X_2||X_b$ for some b and $|X_i| = n$, out of which at least one has k-bits of min-entropy given all preceding blocks. Recall that the average conditional min-entropy is defined as $\tilde{H}_\infty(X \mid Z) := -\log \mathbb{E}_z\big[\max_x \mathsf{P}_{X|Z}(x|z)\big]$.

Definition 7.1. A non-interactive resource is said to be a *k out of n-bit min-entropy block seed*, denoted $\mathsf{X} \in \Phi^{\mathsf{me-blk}}_{k,n}$, if $\mathsf{X} \in \Phi^{\mathsf{seed}}_1$ with $\{0,1\}^* \times \mathcal{A}$ as the output domain of interface $\mathtt{A}$, and there exists a random variable $C \in \{1, \dots, B\}$ such that

$$\tilde{H}_\infty\big(Y_C \,\big|\, Y_{C+1}, \dots, Y_B, C, A, Z\big) \;\geq\; k,$$

where $B = \left\lceil \frac{Y}{n} \right\rceil$ denotes the total number of blocks and Y_i denotes the i-th block of the message Y padded with zeros the a multiple of the block-length n. Moreover, let $\mathcal{C}^{\mathsf{me-blk}}_{k,n} := \Sigma \times \Phi^{\mathsf{me-blk}}_{k,n}$ denote the set of all k-bit min-entropy block contexts.

We now present a sufficient condition for a seed to satisfy Definition 7.1 based on the length of the message and its overall min-entropy. More concretely, we prove that if a message is split into b blocks of size n, and has overall min-entropy of k bits, then there exists a block with $\frac{k}{b} - \log_2(b)$ bits of min-entropy, given all succeeding blocks.[9]

Lemma 7.2. *Let $X_1, \ldots, X_b$ and Z be random variables (over possibly different alphabets) with $\tilde{H}_\infty(X_1 \ldots X_b \,|\, Z) \geq k$. Then, there exists a random variable C over the set $\{1, \ldots, b\}$ such that $\tilde{H}_\infty(X_C \,|\, X_1 \ldots X_{C-1} C Z) \geq k/b - \log_2(b)$.*

Proof. Let $Y_C := (X_1, \ldots, X_{C-1})$, with Y_0 denoting the empty string λ. Second, let for every z in the support of Z,

$$p_z := \max_{x_1,\ldots,x_b} \mathsf{P}_{X_1 \ldots X_b | Z}(x_1, \ldots, x_b, z),$$

that is, $\tilde{H}_\infty(X_1 \ldots X_b \,|\, Z) = -\log \mathbb{E}_z[p_z]$. Moreover, once C is defined (see below), let

$$q_z := \mathop{\mathbb{E}}_{c,y}\left[\max_x \mathsf{P}_{X_C | C Y_C Z}(x, c, y, z) \,\middle|\, Z = z\right]$$

and note that $\tilde{H}_\infty(X_C \,|\, C Y_C Z) = -\log \mathbb{E}_z[q_z]$. We now proceed by showing that for all z, $q_z \leq b \cdot p_z^{1/b}$. To this end, we extend the probability distribution $\mathsf{P}_{X_1 \ldots X_b Z}$ by defining the random variable C as follows:

$$C = \begin{cases} 1 & \text{if } \mathsf{P}_{X_1|Z}(x_1, z) < p_z^{1/b} \\ 2 & \text{else if } \mathsf{P}_{X_1 X_2|Z}(x_1, x_2, z) < p_z^{2/b} \\ \vdots & \\ b-1 & \text{else if } \mathsf{P}_{X_1 \ldots X_{k-1}|Z}(x_1, \ldots, x_{b-1}, z) < p_z^{(b-1)/k} \\ b & \text{else.} \end{cases}$$

Observe that with

$$\mathcal{Y}_{c,z} := \{y \mid \mathsf{P}_{C Y_C | Z}(c, y, z) > 0\}$$
$$\mathcal{X}_{c,z,y} := \{x \mid \mathsf{P}_{X_C C Y_C | Z}(x, c, y, z) > 0\}$$

[9] Note that in order to obtain a result that is more closely resembles the chain rule of Shannon entropy, the proposition is stated with conditioning on all preceding message $X_1 \ldots X_{C-1}$ instead of all succeeding ones as required for Definition 7.1. The converse result can easily be obtained by simply relabeling the blocks.

we can bound q_z as follows:

$$\begin{aligned}
q_z &= \mathop{\mathbb{E}}_{c,y}\left[\max_x \mathsf{P}_{X_C|CY_CZ}(x,c,y,z) \,\middle|\, Z=z\right] \\
&= \sum_{c=1}^{b} \sum_{y\in\mathcal{Y}_{c,z}} \mathsf{P}_{CY_C|Z}(c,y,z)\cdot \max_{x\in\mathcal{X}_{c,z,y}} \mathsf{P}_{X_C|CY_CZ}(x,c,y,z) \\
&= \sum_{c=1}^{b} \sum_{y\in\mathcal{Y}_{c,z}} \mathsf{P}_{CY_C|Z}(c,y,z)\cdot \max_{x\in\mathcal{X}_{c,z,y}} \frac{\mathsf{P}_{X_CCY_C|Z}(x,c,y,z)}{\mathsf{P}_{CY_C|Z}(c,y,z)} \\
&= \sum_{c=1}^{b} \sum_{y\in\mathcal{Y}_{c,z}} \max_{x\in\mathcal{X}_{c,z,y}} \mathsf{P}_{X_CCY_C|Z}(x,c,y,z) \\
&\leq \sum_{c=1}^{b} \sum_{y\in\mathcal{Y}_{c,z}} \max_{x\in\mathcal{X}_{c,z,y}} \mathsf{P}_{X_CY_C|Z}(x,y,z).
\end{aligned}$$

We now further bound this term using a case distinction on c. First, consider the case $c = 1$. Since $Y_1 = \lambda$ is constant, we have $\mathcal{Y}_{1,z} \subseteq \{\lambda\}$ and $\mathsf{P}_{X_1Y_1|Z}(x,\lambda,z) = \mathsf{P}_{X_1|Z}(x,z)$. Moreover, $x \in \mathcal{X}_{1,z,\lambda}$ implies $\mathsf{P}_{X_1C|Z}(x,1,z) > 0$, which by the definition of C in turn implies $\mathsf{P}_{X_1|Z}(x,z) < p_z^{1/b}$. Hence

$$\sum_{y\in\mathcal{Y}_{1,z}} \max_{x\in\mathcal{X}_{1,z,y}} \mathsf{P}_{X_1Y_1|Z}(x,y,z) \leq \max_{x\in\mathcal{X}_{1,z,\lambda}} \mathsf{P}_{X_1|Z}(x,z) \leq p_z^{1/b}.$$

For all $i \in \{2,\ldots,b-1\}$ observe that by the definition of C we have that $\mathcal{X}_{i,z,y} \subseteq \{x \mid \mathsf{P}_{X_iY_i|Z}(x,y,z) < p_z^{i/b}\}$ and $\mathcal{Y}_{i,z} \subseteq \{y \mid \mathsf{P}_{Y_i|Z}(y,z) \geq p_z^{(i-1)/b}\}$. From the latter we can conclude that $|\mathcal{Y}_{i,z}| \leq \frac{1}{p_z^{(i-1)/b}}$ and, hence, we obtain

$$\sum_{y\in\mathcal{Y}_{i,z}} \max_{x\in\mathcal{X}_{i,z,y}} \mathsf{P}_{X_iY_i|Z}(x,y,z) \leq \sum_{y\in\mathcal{Y}_{i,z}} p_z^{i/b} \leq \frac{p_z^{i/b}}{p_z^{(i-1)/b}} = p_z^{1/b}.$$

Finally, for $c = b$, we have that $\mathcal{Y}_{b,z} \subseteq \{y \mid \mathsf{P}_{Y_b|Z}(y,z) \geq p_z^{(b-1)/b}\}$ and, thus, $|\mathcal{Y}_{b,z}| \leq \frac{1}{p_z^{(b-1)/b}}$. Using the definition of $Y_b = (X_1,\ldots,X_{b-1})$ and p_z, we get $\max_{x\in\mathcal{X}_{b,z,y}} \mathsf{P}_{X_bY_b|Z}(x,y,z) \leq p_z$ for every $y = (x_1,\ldots,x_{b-1})$. Therefore,

$$\sum_{y\in\mathcal{Y}_{b,z}} \max_{x\in\mathcal{X}_{b,z,y}} \mathsf{P}_{X_bY_b|Z}(x,y,z) \leq \sum_{y\in\mathcal{Y}_{b,z}} p_z \leq \frac{p_z}{p_z^{(b-1)/b}} = p_z^{1/b}$$

as well. In summary,

$$\begin{aligned}
q_z &= \mathop{\mathbb{E}}_{c,y}\left[\max_x \mathsf{P}_{X_C|CY_CZ}(x,c,y,z) \,\middle|\, Z=z\right] \\
&\leq \sum_{c=1}^{b} \sum_{y\in\mathcal{Y}_{c,z}} \max_{x\in\mathcal{X}_{c,z,y}} \mathsf{P}_{X_CY_C|Z}(x,y,z) \\
&\leq \sum_{c=1}^{b} p_z^{1/b} \\
&\leq b\cdot p_z^{1/b}
\end{aligned}$$

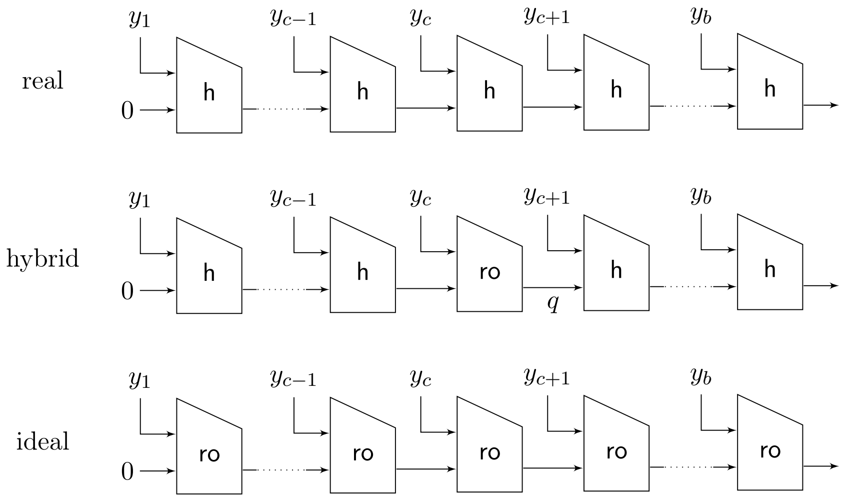


Figure 10: The real and the ideal setting for the Merkle–Damgård construction if block c has high min-entropy.

Using the monotonicity of the expected value, Jensen's inequality, and the assumption on the average conditional min-entropy, $\tilde{H}_\infty(X_1 \dots X_b \mid Z) \geq k$, yields

$$\begin{aligned} 2^{-\tilde{H}_\infty(X_C \mid C Y_C Z)} &= \mathop{\mathbb{E}}_z[q_z] \leq \mathop{\mathbb{E}}_z\left[b \cdot p_z^{1/b}\right] \leq b \cdot \mathop{\mathbb{E}}_z[p_z]^{1/b} \\ &= b \cdot \left(2^{-\tilde{H}_\infty(X_1 \dots X_b \mid Z)}\right)^{1/b} \leq 2^{\log b} \cdot 2^{-k/b} = 2^{-(k/b - \log b)} \end{aligned}$$

concluding the proof. □

This lemma is a generalization of the randomized chain rule proven by the authors of [DFR+07] (similar variants exists also in [BK12; Wul07; DKZZ15]) stating that there exists a binary random variable C such that $H_\infty(X_{1-C}C) \geq H_\infty(X_0 X_1)/2$. Note that the main difference of our result is, that it conditions on all previous blocks, i.e., it is basically the min-entropy equivalence of the strong chain rule $H(X_0) + H(X_1|X_0) = H(X_0 X_1)$ instead of $H(X_0) + H(X_1) \geq H(X_0 X_1)$.

We can now state the following theorem, stating that the Merkle–Damgård construction is split context-restricted indifferentiable if the underlying compression function is strong-split context-restricted indifferentiable for blocks with sufficient min-entropy. Let us first present an informal argument: Assume the message y is split into b blocks, out of which at least one has k bits of min-entropy. Let y_c denote this block. Hence, according to our assumption on the compression function, the output q of this block cannot be distinguished from a uniformly random value of length n and, by induction, neither can be the output of any subsequent block. Therefore, the final output cannot be distinguished from the uniform random value $RO(X)$. See Figure 10 for a graphical illustration.

Theorem 7.3. *Let $h\colon \{0,1\}^m \to \{0,1\}^n$ denote a fixed input-length compression function and for any $b \in \mathbb{N}$, let $H_b\colon \bigcup_{\ell \leq b(m-n)} \{0,1\}^\ell \to \{0,1\}^n$ denote the hash function obtained by first padding the message with zeros to a multiple of the block-length and then applying the Merkle–Damgård scheme using h. Then there exists a pair of reductions (translating the distinguisher) $\rho_1, \rho_2\colon \mathcal{D} \times (\mathcal{C}_1^{\mathsf{splt}} \cap \mathcal{C}_{k,m-n}^{\mathsf{me-blk}}) \to \mathcal{D}$ and a pair of context translations $\psi_1, \psi_2\colon \mathcal{C}_1^{\mathsf{splt}} \cap \mathcal{C}_{k,m-n}^{\mathsf{me-blk}} \to \mathcal{C}_{1,1}^{\mathsf{s-splt}} \cap \mathcal{C}_{1,k'}^{\mathsf{s-me}}$ such that for all distinguishers D and all contexts $(\mathsf{f}, \mathsf{X}) \in \mathcal{C}_1^{\mathsf{splt}} \cap \mathcal{C}_{k,m-n}^{\mathsf{me-blk}}$ we have*

$$\mathbf{Adv}^{\mathrm{RO-CRI}}_{\mathsf{H_b},\mathsf{f},\mathsf{X},\sigma}(\mathsf{D}) \leq \binom{b}{2} \cdot 2^{-(k'-1)} + b \cdot \mathbf{Adv}^{\mathrm{RO-CRI}}_{\mathsf{h},\mathsf{f}',\mathsf{X}',\sigma'}(\mathsf{D}') + \mathbf{Adv}^{\mathrm{RO-CRI}}_{\mathsf{h},\mathsf{f}'',\mathsf{X}'',\sigma''}(\mathsf{D}'')$$

with $k' := \min(k,n)$*,* $\mathsf{D}' := \rho_1(\mathsf{D},\mathsf{f},\mathsf{X})$*,* $\mathsf{D}'' := \rho_2(\mathsf{D},\mathsf{f},\mathsf{X})$*,* $(\mathsf{f}',\mathsf{X}') := \psi_1(\mathsf{f},\mathsf{X})$*,* $(\mathsf{f}'',\mathsf{X}'') := \psi_2(\mathsf{f},\mathsf{X})$*, and* σ' *and* σ'' *denoting slightly modified variants of* σ*.*

Proof. Given any k out of n-bit min-entropy block seed X, we first introduce two k'-bit min-entropy seeds X' and X''. Note that the function ψ_1 and ψ_2 are just mappings from one context to another one relating the two problems and, in contrast to the reduction translating the distinguisher, do not need to be efficiently computable. Therefore, it is sufficient to know that such a random variable C from Definition 7.1 exists for the seed X.

Definition of X':

Let X' denote the randomness seed which samples (y,z) using the same distribution as X, applies the padding, and splits it into the blocks $y_1,\ldots,y_b$. Then, it sample the random variable C to obtain the index c. Finally, it outputs the pair (a',y') with $a' = (y_0,y_1,\ldots,y_{c-1})$ and $y' = y_c$ at interface `A` and $z' = (z,c,y_{c+1},\ldots,y_b)$ at interface `E`.

Definition of X'':

Let X'' denote the randomness seed which samples (y,z) using the same distribution as X, applies the padding, and splits it into the blocks $y_1,\ldots,y_b$. Then, it sample the random variable C to obtain the index c and chooses $q \in \{0,1\}^n$ uniformly at random, outputs the pair $(a',y') := (\perp,q)$ at interface `A`, and the value $z' := (z,c,y_{c+1},\ldots,y_b)$ at interface `E`.

Observe that X' is a $k \geq k'$ bit (strict) min-entropy seed, since X is k out of n-bit min-entropy block seed. Similarly, since q is chosen independently of all other random variables, the seed X'' is a $n \geq k'$ bit strict min-entropy seed. Moreover, both of them output only a single value, i.e., $\mathsf{X}',\mathsf{X}'' \in \Phi^{\mathsf{seed}}_1$.

Next, we briefly sketch the two simulators σ' and σ'': they both internally run σ. Whenever σ request for the leakage z of the seed, they query the leakage z' at the corresponding inner interface and return the first component z to σ.

Now, we introduce two converter systems C' and C'' that at the inside interface connect to both the interface `A` and the interface `E` of the connected system, and at the outside interface emulates both the interfaces as well.

The system C' works as follows:

First it obtains hk and $z' = (z,c,y_{c+1},\ldots,y_b)$ at the interfaces `E.H` and `E.X` of the connected system. When receiving the input `retrieve` at the outside interface `A`, it outputs $(\texttt{retrieve}, f)$ at the inside interface `A`, where f is the function that on input (y_c,a') first splits $a' = (y_0,\ldots,y_{c-1})$, then computes the prefix $p = h_{hk}(\ldots h_{hk}(h_{hk}(0||y_0)||y_1)\ldots||y_{c-1})$, and finally returns $p||y_c$. Since both p and y_c are of fixed length, this function is injective in the first argument. When obtaining the returned value y', it then computes the suffix $s = h_{hk}(\ldots h_{hk}(h_{hk}(y'||y_{c+1})||y_{c+2})\ldots||y_b)$ and returns s at the outside interface `A`. When receiving the input `retrieve` at either the interface `E.H` or `E.X` it returns hk or z, respectively.

The system C'' works as follows:

First it obtains hk and $z' = (z,c,x_{c+1},\ldots,x_b)$ at the interfaces `E.H` and `E.X` of the connected system. When receiving the input `retrieve` at the outside interface `A`, it first outputs $(\mathtt{query}, f)$ at the inside interface `A`, where f is the function that on input $(q, \perp)$ returns $q||y_{c+1}$. This function is injective in the first argument. Then, for $i = c+2, \ldots, b$ it outputs $(\mathtt{repeat}, f)$ at the inside interface `A`, where f is the function that on input (x) returns $x||y_i$. Finally, it outputs `get` at the inside interface `A` and returns the obtained value at the outside interface `A`. When receiving the input `retrieve` at either the interface `E.H` or `E.R` it returns hk or z, respectively.

It is easy to verify, that the composed system $\mathsf{C}'\mathsf{f}_1^{\mathsf{s-splt}}[\mathsf{h}, \mathsf{X}']$ at the interface `A` outputs $H(y)$ and, thus, we have the equivalence $\mathsf{f}^{\mathsf{splt}}[\mathsf{H}, \mathsf{X}] \equiv \mathsf{C}'\mathsf{f}_1^{\mathsf{s-splt}}[\mathsf{h}, \mathsf{X}']$. Moreover, it is easy to verify that the final output of the composed system $\mathsf{C}''\mathsf{f}_{1,p}^{\mathsf{r-splt}}[\mathsf{ro}, \mathsf{X}'']\sigma''$ at the interface `A` is just a uniform random value independent of hk and z. Hence, this system behaves equivalently to $\mathsf{f}^{\mathsf{splt}}[\mathsf{RO}, \mathsf{X}]\sigma$ that outputs a single uniform random value as well. In short, we have $\mathsf{f}^{\mathsf{splt}}[\mathsf{RO}, \mathsf{X}]\sigma \equiv \mathsf{C}''\mathsf{f}_{1,p}^{\mathsf{r-splt}}[\mathsf{ro}, \mathsf{X}'']\sigma''$.

Using those two equivalences and by introducing the two hybrid systems $\mathsf{C}'\mathsf{f}_1^{\mathsf{s-splt}}[\mathsf{ro}, \mathsf{X}']\sigma'$ and $\mathsf{C}''\mathsf{f}_{1,b}^{\mathsf{r-splt}}[\mathsf{h}, \mathsf{X}'']\sigma''$, we can rewrite the distinction advantage as:

$$\begin{aligned}\Delta^{\mathsf{D}}\Big(\mathsf{f}^{\mathsf{splt}}[\mathsf{H}, \mathsf{X}], \mathsf{f}^{\mathsf{splt}}[\mathsf{RO}, \mathsf{X}]\sigma\Big) &= \Delta^{\mathsf{D}}\Big(\mathsf{C}'\mathsf{f}_1^{\mathsf{s-splt}}\big[\mathsf{h}, \mathsf{X}'\big], \mathsf{C}'\mathsf{f}_1^{\mathsf{s-splt}}\big[\mathsf{ro}, \mathsf{X}'\big]\sigma'\Big) \\ &\quad + \Delta^{\mathsf{D}}\Big(\mathsf{C}'\mathsf{f}_1^{\mathsf{s-splt}}\big[\mathsf{ro}, \mathsf{X}'\big]\sigma', \mathsf{C}''\mathsf{f}_{1,\mathsf{b}}^{\mathsf{r-splt}}\big[\mathsf{h}, \mathsf{X}''\big]\Big) \\ &\quad + \Delta^{\mathsf{D}}\Big(\mathsf{C}''\mathsf{f}_{1,\mathsf{b}}^{\mathsf{r-splt}}\big[\mathsf{h}, \mathsf{X}''\big], \mathsf{C}''\mathsf{f}_{1,\mathsf{b}}^{\mathsf{r-splt}}\big[\mathsf{ro}, \mathsf{X}''\big]\sigma''\Big).\end{aligned}$$

Finally, observe that the systems $\mathsf{C}'\mathsf{f}_1^{\mathsf{s-splt}}[\mathsf{ro}, \mathsf{X}']\sigma'$ and $\mathsf{C}''\mathsf{f}_{1,b}^{\mathsf{r-splt}}[\mathsf{h}, \mathsf{X}'']\sigma''$ both implement exactly the same hybrid system depicted in Figure 10: The system $\mathsf{C}'\mathsf{f}_1^{\mathsf{s-splt}}[\mathsf{ro}, \mathsf{X}']\sigma'$ actually computes this value by first using the compression function h on the blocks 1 to $c-1$, then uses the fixed input size random oracle on the block c, and finishes by using h on the remaining blocks. However, note that the value output by ro is just a uniform random value, as ro is private and not used beside this one query. The system $\mathsf{C}''\mathsf{f}_{1,b}^{\mathsf{r-splt}}[\mathsf{h}, \mathsf{X}'']$ skips the initial computes and chooses q uniformly at random (in X'').

As a result, we can simplify the distinction advantage to

$$\begin{aligned}\Delta^{\mathsf{D}}\Big(\mathsf{f}^{\mathsf{splt}}[\mathsf{H}_{\mathsf{b}}, \mathsf{X}], \mathsf{f}^{\mathsf{splt}}[\mathsf{RO}, \mathsf{X}]\sigma\Big) &= \Delta^{\mathsf{DC}'}\Big(\mathsf{f}_1^{\mathsf{s-splt}}\big[\mathsf{h}, \mathsf{X}'\big], \mathsf{f}_1^{\mathsf{s-splt}}\big[\mathsf{ro}, \mathsf{X}'\big]\sigma'\Big) \\ &\quad + \Delta^{\mathsf{DC}''}\Big(\mathsf{f}_{1,\mathsf{b}}^{\mathsf{r-splt}}\big[\mathsf{h}, \mathsf{X}''\big], \mathsf{f}_{1,\mathsf{b}}^{\mathsf{r-splt}}\big[\mathsf{ro}, \mathsf{X}''\big]\sigma''\Big).\end{aligned}$$

Applying the definition of $\mathbf{Adv}^{\mathrm{RO-CRI}}_{\mathsf{H_b},\mathsf{f},\mathsf{X},\sigma}(\mathsf{D})$ and applying Lemma 6.6 (with $p = 1$) concludes the proof. □

Remark. In the proof we used that we only compute the hash value of a single message. Using Lemma 6.5, this statement can easily be generalized to multiple messages with appropriate conditional min-entropy; however, we do not allow for any strongly correlated queries since they exhibit the well-known length-extension attacks on the Merkle–Damgård scheme. Whether a more advanced construction with a finalization function, e.g. HMAC, could be proven secure for the general statistically unpredictable split-security remains an interesting open problem.

7.2 Using RO-CRI to Analyze the Soundness

The indifferentiability framework is widely accepted as a tool to analyze the soundness of hash function constructions. In fact all candidates for the recent SHA-3 competition were strongly encouraged to include an indifferentiability proof. In such a proof the underlying primitive—typically a compression function—is replaced by an idealized version of it—a fixed input-length random oracle. Then, it is shown that the constructed hash function is indifferentiable from a random oracle. However, the impossibility results for the random oracle model directly apply to compression functions: no concrete one behaves like a fixed input-length random oracle. Therefore, such an analysis of the hash function construction closely resembles a proof in the random oracle model; while giving some indication of the security, it does not give any concrete security guarantees.

The UCE framework, and especially RO-CRI, provides a viable alternative to assess the soundness of hash function constructions by aiming at proving the $\mathcal{C}$ indifferentiability of the constructed hash function by assuming that the underlying compression function is $\mathcal{C}'$ indifferentiable, where $\mathcal{C}$ and $\mathcal{C}'$ are potentially different context sets. This gives rise to a family of such statements: the more general $\mathcal{C}$ is, or the more restricted $\mathcal{C}'$ is, the stronger the statement is. The classical indifferentiability statement is just one special case with the strongest possible assumption and conclusion. Nevertheless, most hash function constructions have never been analyzed in a more fine-grained way, expect for the one in the initial work of [BHK14].

A Proof of Lemma 6.6

Lemma 6.6. *Let $k' := \min(k, \log|H.\mathcal{Y}|)$. If H is a $\mathcal{C}^{\mathsf{s-splt}}$ indifferentiable hash function for strict k'-bit min-entropy sources, then H is also $\mathcal{C}^{\mathsf{r-splt}}$ indifferentiable hash function for strict k-bit min-entropy sources.*

Formally, let $\mathcal{D}$ denote the set of distinguishers. Then there exists a reduction $\rho\colon \mathcal{D} \times \left(\mathcal{C}^{\mathsf{r-splt}}_{n,p,r} \cap \mathcal{C}^{\mathsf{s-me}}_{n,k}\right) \to \mathcal{D}$ and a translation of the context $\psi\colon \mathcal{C}^{\mathsf{r-splt}}_{n,p,r} \cap \mathcal{C}^{\mathsf{s-me}}_{n,k} \to \mathcal{C}^{\mathsf{s-splt}}_{n,p} \cap \mathcal{C}^{\mathsf{s-me}}_{n,k'}$, such that for every $(\mathsf{f}, \mathsf{X}) \in \mathcal{C}^{\mathsf{r-splt}}_{n,p,r} \cap \mathcal{C}^{\mathsf{s-me}}_{n,k}$ we have

$$\mathbf{Adv}^{\mathrm{RO-CRI}}_{\mathsf{H},\mathsf{f},\mathsf{X},\sigma}(\mathsf{D}) \leq \binom{npr}{2} 2^{-(k'-1)} + r \cdot \mathbf{Adv}^{\mathrm{RO-CRI}}_{\mathsf{H},\mathsf{f}',\mathsf{X}',\sigma}(\mathsf{D}')$$

with $\mathsf{D}' := \rho(\mathsf{D}, \mathsf{f}, \mathsf{X})$ and $(\mathsf{f}', \mathsf{X}') := \psi(\mathsf{f}, \mathsf{X})$.

Proof. Let $(\mathsf{f}, \mathsf{X}) \in \mathcal{C}^{\mathsf{r-splt}}_{n,p,r} \cap \mathcal{C}^{\mathsf{s-me}}_{n,k}$. By definition, we then have $\mathsf{f} := \mathsf{g} \circ \mathsf{f}^{\mathsf{r-splt}}_{p,r}$ for some filter g. This filter can also be thought of as an reduction of the distinguisher (which follows from the composition-order independence [MR11]), and thus we can rewrite

$$\begin{aligned}\mathbf{Adv}^{\mathrm{RO-CRI}}_{\mathsf{H},\mathsf{f},\mathsf{X},\sigma}(\mathsf{D}) &:= \Delta^{\mathsf{D}}(\mathsf{f}[\mathsf{H},\mathsf{X}], \mathsf{f}[\mathsf{RO},\mathsf{X}]\sigma) \\ &= \Delta^{\mathsf{D}''}\left(\mathsf{f}^{\mathsf{r-splt}}_{\mathsf{p,r}}[\mathsf{H},\mathsf{X}], \mathsf{f}^{\mathsf{r-splt}}_{\mathsf{p,r}}[\mathsf{RO},\mathsf{X}]\sigma\right)\end{aligned}$$

with $\mathsf{D}'' = \rho_1(\mathsf{D}) := \mathsf{D}\mathsf{g}$.

Consider the beacon resource B that has the same interface as the random oracle interface, but response with a fresh random value for each query (i.e., it ignores the consistency condition for repeated queries). Moreover, we introduce the following shorthand notation: $\mathsf{S}^{\mathsf{H}} := \mathsf{f}^{\mathsf{r-splt}}_{p,r}[\mathsf{H},\mathsf{X}]$,

$\mathsf{S}^{\mathsf{RO}} := \mathsf{f}_{p,r}^{\mathsf{r-splt}}[\mathsf{RO},\mathsf{X}]\sigma$, and $\mathsf{S}^{\mathsf{B}} := \mathsf{f}_{p,r}^{\mathsf{r-splt}}[\mathsf{B},\mathsf{X}]\sigma$, which allows the advantage of the distinguisher D'' to be expressed as

$$\Delta^{\mathsf{D}''}(\mathsf{f}[\mathsf{H},\mathsf{X}],\mathsf{f}[\mathsf{RO},\mathsf{X}]\sigma) = \Delta^{\mathsf{D}''}\left(\mathsf{S}^{\mathsf{H}},\mathsf{S}^{\mathsf{B}}\right) + \Delta^{\mathsf{D}''}\left(\mathsf{S}^{\mathsf{B}},\mathsf{S}^{\mathsf{RO}}\right).$$

We now describe the reduction ρ_2 that bounds the first term of the sum with $\binom{npr}{2}2^{-k'} + r \cdot \mathbf{Adv}^{\mathrm{RO-CRI}}_{\mathsf{H},\mathsf{f}',\mathsf{X}',\sigma}(\mathsf{D}')$ using a simple hybrid argument.

Let $\{\mathsf{X_i}\}_{i\in[q]}$ denote the sequence of hybrid resources that behave as follows: at the interface $\mathtt{E}$, the resource first outputs the index i and subsequently behaves exactly as X. At the interface $\mathtt{A}$, if $i = 1$ then it behaves exactly as X, and if $i > 1$ then it outputs n independent uniformly at random chosen values from the set $H.\mathcal{Y}$. It is easy to see, that if $\mathsf{X} \in \Phi_n^{\mathsf{seed}} \cap \Phi_{k'}^{\mathsf{s-me}}$, then $\mathsf{X_i} \in \Phi_n^{\mathsf{seed}} \cap \Phi_{k'}^{\mathsf{s-me}}$ for all i. In addition, let X' denote the resource which chooses $i \in [q]$ uniformly at random and then behaves like $\mathsf{X_i}$. Furthermore, let $\mathsf{f}' := \mathsf{f}_p^{\mathsf{s-splt}}$ and, hence $(\mathsf{f}',\mathsf{X}') \in \mathcal{C}_{n,p}^{\mathsf{s-splt}} \cap \mathcal{C}_{k'}^{\mathsf{s-me}}$. Analogously to above, let us define the following shorthand notation: $\mathsf{T}^{\mathsf{H}} := \mathsf{f}_p^{\mathsf{s-splt}}[\mathsf{R},\mathsf{X}']$, $\mathsf{T}_{\mathsf{i}}^{\mathsf{H}} := \mathsf{f}_p^{\mathsf{s-splt}}[\mathsf{R},\mathsf{X_i}]$, and $\mathsf{T}^{\mathsf{R}} := \mathsf{f}_p^{\mathsf{s-splt}}[\mathsf{R},\mathsf{X}']\sigma$ and $\mathsf{T}_{\mathsf{i}}^{\mathsf{R}} := \mathsf{f}_p^{\mathsf{s-splt}}[\mathsf{R},\mathsf{X_i}]\sigma$ for $\mathsf{R} \in \{\mathsf{RO},\mathsf{B}\}$.

Now, consider the reduction $\mathsf{D}' := \rho_2(\mathsf{D}'') = \rho_2(\rho_1(\mathsf{D}))$ where ρ_2 is implemented using a special type of system C that translates one setting into the other. Formally C is a converter that has an inside and an outside interface, where the inside interface connects to all the (merged) interfaces of the attached resource (here interface $\mathtt{A}$ and $\mathtt{E}$) and the outside interface becomes the interfaces of the composed resource. Now consider the following reduction system C, which on the inside expects to be connected either to the resource $\mathsf{T}_{\mathsf{i}}^{\mathsf{H}}$ or $\mathsf{T}_{\mathsf{i}}^{\mathsf{B}}$. At the outside interfaces, it simulates the according interfaces of $\mathsf{f}_{p,r}^{\mathsf{r-splt}}[\mathsf{H},\mathsf{X}]$ and $\mathsf{f}_{p,r}^{\mathsf{r-splt}}[\mathsf{B},\mathsf{X}]\sigma$. The system C first gets the index i and the hash key hk at the inside interface. In every sequence of queries of the form $(\mathtt{query}, f_1), (\mathtt{repeat}, f_2), (\mathtt{repeat}, f_3), \ldots$, the queries 1 to $i-1$ are simulated internally as queries to the beacon by sampling a value uniformly at random and storing it in a buffer b. The i-th query in each such sequence is then answered using the actual resource connected at the inside interface. For the remaining queries, the system C computes the hash function H itself. A formal description of the reduction system is provided in Figure 11.

The following system equivalences are easy to verify:

$$\mathsf{C}\mathsf{T}_1^{\mathsf{H}} \equiv \mathsf{S}^{\mathsf{H}} \tag{1}$$

$$\mathsf{C}\mathsf{W}_{\mathsf{r}}^{\mathsf{B}} \equiv \mathsf{S}^{\mathsf{B}} \tag{2}$$

$$\mathsf{C}\mathsf{W}_{\mathsf{i-1}}^{\mathsf{B}} \equiv \mathsf{C}\mathsf{W}_{\mathsf{i}}^{\mathsf{H}} \quad \forall i \in \{2,\ldots,q\}. \tag{3}$$

```
Resource C

Initialization
    $b \leftarrow \bot^p$
    $j \leftarrow 0$
    output retrieve at in.E.H
    let $hk$ denote the result
    output retrieve at in.E.X
    let $i$ denote the result

Outer Interface A
Input: get
    output $b$ at out.A
    $b \leftarrow \bot^p$

Input: $(\texttt{query}, f_1, \dots, f_p)$
    $j \leftarrow 1$
    if $i = 1$ then
        output $(\texttt{retrieve}, f_1, \dots, f_p)$ at in.A
        let $b$ denote the result
    else
        $b \overset{\$}{\leftarrow} H.\mathcal{Y}^p$

Input: $(\texttt{repeat}, f_1, \dots, f_p)$
    $j \leftarrow j + 1$
    if $b \neq \bot^p$ then
        for $\ell = 1, \dots, p$ do
            if $j < i$ then
                $b[\ell] \overset{\$}{\leftarrow} H.\mathcal{Y}$
            else if $j = i$ then
                output $(\texttt{retrieve}, (x, a) \mapsto f_\ell(x))$ at in.A
                let $b[\ell]$ denote the result
            else
                $b[\ell] \leftarrow H(hk, f_l(b[\ell]))$

Outer Interface E.i $i \in \{H, X\}$
Input: retrieve
    output retrieve at in.E.i
    let $y$ denote the result
    output $y$ at out.E.i
```

Figure 11: The reduction system C.

As a consequence, we can rewrite the second term as

$$\begin{aligned}\Delta^{\mathsf{D}''}\left(\mathsf{S}^{\mathsf{H}}, \mathsf{S}^{\mathsf{B}}\right) &= \Delta^{\mathsf{D}''}\left(\mathsf{CT}^{\mathsf{H}}_{1}, \mathsf{CT}^{\mathsf{B}}_{\mathsf{r}}\right)\\ &= \Delta^{\mathsf{D}''}\left(\mathsf{CT}^{\mathsf{H}}_{1}, \mathsf{CT}^{\mathsf{B}}_{1}\right) + \Delta^{\mathsf{D}''}\left(\mathsf{CT}^{\mathsf{B}}_{1}, \mathsf{CT}^{\mathsf{H}}_{2}\right)\\ &\quad + \Delta^{\mathsf{D}''}\left(\mathsf{CT}^{\mathsf{H}}_{2}, \mathsf{CT}^{\mathsf{B}}_{2}\right) + \Delta^{\mathsf{D}}\left(\mathsf{CT}^{\mathsf{B}}_{2}, \mathsf{CT}^{\mathsf{H}}_{3}\right)\\ &\quad + \dots\\ &\quad + \Delta^{\mathsf{D}}\left(\mathsf{CT}^{\mathsf{H}}_{\mathsf{r}}, \mathsf{CT}^{\mathsf{B}}_{\mathsf{r}}\right)\\ &= \sum_{i=1}^{r} \Delta^{\mathsf{D}''}\left(\mathsf{CT}^{\mathsf{H}}_{\mathsf{i}}, \mathsf{CT}^{\mathsf{B}}_{\mathsf{i}}\right)\\ &= r \cdot \Delta^{\mathsf{D}''}\left(\mathsf{CT}^{\mathsf{H}}, \mathsf{CT}^{\mathsf{B}}\right)\\ &= r \cdot \Delta^{\mathsf{D}'}\left(\mathsf{T}^{\mathsf{H}}, \mathsf{T}^{\mathsf{B}}\right)\\ &= r \cdot \mathbf{Adv}^{\text{RO}-\text{CRI}}_{\mathsf{H},\mathsf{f}',\mathsf{R}',\sigma}(\mathsf{D}') + r \cdot \Delta^{\mathsf{D}'}\left(\mathsf{T}^{\mathsf{RO}}, \mathsf{T}^{\mathsf{B}}\right)\end{aligned}$$

where in the third step we used equation (3). In the forth step we used that the distinguishing advantage of D'' on the problem with R' is the average of the distinguising advantage of D on resources with the the fixed i. Hence, the sum of these r terms is equal to r times the average.

The overall claim is then directly implied by the following two bounds, which remain to be shown:

$$\Delta^{\mathsf{D}'}\left(\mathsf{T}^{\mathsf{RO}}, \mathsf{T}^{\mathsf{B}}\right) \leq \binom{np}{2} 2^{-k'} \tag{4}$$

$$\Delta^{\mathsf{D}''}\left(\mathsf{S}^{\mathsf{B}}, \mathsf{S}^{\mathsf{RO}}\right) \leq \binom{npr}{2} 2^{-k'} \tag{5}$$

In both cases the two resources behave exactly identically until a repeated query to the oracle occurs. Hence, we can bound the distinction advantage by the probability of managing non-adaptively to query twice the same input [Mau13]. In the following, we only prove 5, as 4 follows by an analogous argument.

Let $Z_1, Z_2, \dots, Z_{npr}$ denote the queries, which are submitted to the beacon. The collision probability can then be bounded using the union bound

$$\Pr(\exists i \neq j \;\; Z_i = Z_j) \leq \sum_{i \neq j} \Pr(Z_i = Z_j).$$

Observe that all queries are either of the form $f(Y_s, A_s)$, where (Y_s, A_s) is the s-th pair output by the entropy source, or $f(Y)$, where Y is an output of the beacon. If either Z_i or Z_j is of the latter type, then the collision probability is trivially upper bounded by $\frac{1}{|H.\mathcal{Y}|} \leq 2^{-k'}$, using that f is injective. If both of them are of the former type, then that by definition of the filter $\mathsf{f}^{\mathsf{r-splt}}_{p,r}$ the two inputs Z_i, Z_j cannot collide if they depend on the same underlying value X_s from the entropy source. Hence, assume w.l.o.g. that $Y_i = f(Y_s, A_s)$ and $Y_j = f(Y_t, A_t)$ with $s > t$. For every pair of fixed auxiliary information (a_s, a_t), we obtain the following bound:

$$\begin{aligned}
&\Pr(f_i(Y_s, a_s) = f_j(Y_t, a_t)) \\
&= \sum_z \Pr(f_i(Y_s, a_s) = z \wedge f_j(Y_t, a_t) = z) \\
&= \sum_z \Pr(Y_t = f_j^{-1}(z, a_t)) \cdot \Pr(Y_s = f_i^{-1}(z, a_s) \mid Y_t = f_j^{-1}(z, a_t)) \\
&\leq \sum_z \Pr(Y_t = f_j^{-1}(z, a_t)) \cdot \max_{t_s} \Pr(Y_s = t_s \mid Y_t = f_j^{-1}(z, a_t)) \\
&= \sum_{y_t} \Pr(Y_t = y_t) \cdot \max_{y_s} \Pr(Y_s = y_s \mid Y_t = y_t) \\
&= 2^{-\tilde{H}_\infty(Y_s \,|\, Y_t)} = 2^{-k} \leq 2^{-k'}.
\end{aligned}$$

Averaging over the choice of (a_s, a_t) yields the desired result $\Pr(f_i(Y_s, A_s) = f_j(Y_t, A_t)) \leq 2^{-k'}$. In summary, the distinction advantage $\Delta^{\mathsf{D}}\big(\mathsf{S}^{\mathsf{B}}, \mathsf{S}^{\mathsf{RO}}\big)$ can be bounded as

$$\Pr(\exists i \neq j \;\; Z_i = Z_j) \leq \sum_{i \neq j} \Pr(Z_i = Z_j) \leq \binom{npr}{2} 2^{-k'},$$

concluding our proof. □

References

[BHK13] M. Bellare, V. T. Hoang, and S. Keelveedhi, "Instantiating Random Oracles via UCEs", in *Advances in Cryptology - CRYPTO 2013: 33rd Annual Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2013. Proceedings, Part II*, Springer Berlin Heidelberg, 2013, pp. 398–415.

[BHK14] ——, "Cryptography from compression functions: The UCE bridge to the ROM", in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, J. A. Garay and R. Gennaro, Eds., vol. 8616 LNCS, Springer Berlin Heidelberg, 2014, pp. 169–187.

[BR93] M. Bellare and P. Rogaway, "Random oracles are practical", in *Proceedings of the 1st ACM conference on Computer and communications security - CCS '93*, New York, New York, USA: ACM Press, Dec. 1993, pp. 62–73.

[BST16] M. Bellare, I. Stepanovs, and S. Tessaro, "Contention in Cryptoland: Obfuscation, Leakage and UCE", in *Theory of Cryptography: 13th International Conference, TCC 2016-A, Tel Aviv, Israel, January 10-13, 2016, Proceedings, Part II*, E. Kushilevitz and T. Malkin, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2016, pp. 542–564.

[BK12] Z. Brakerski and Y. Kalai, "A parallel repetition theorem for leakage resilience", *Theory of Cryptography*, 2012.

[BFM14] C. Brzuska, P. Farshim, and A. Mittelbach, "Indistinguishability Obfuscation and UCEs: The Case of Computationally Unpredictable Sources", in *Advances in Cryptology - CRYPTO 2014: 34th Annual Cryptology Conference, Santa Barbara, CA, USA, August 17-21, 2014, Proceedings, Part I*, Springer Berlin Heidelberg, 2014, pp. 188–205.

[BM14] C. Brzuska and A. Mittelbach, “Using Indistinguishability Obfuscation via UCEs”, in *Advances in Cryptology - ASIACRYPT 2014: 20th International Conference on the Theory and Application of Cryptology and Information Security, Kaoshiung, Taiwan, R.O.C., December 7-11, 2014, Proceedings, Part II*, Springer Berlin Heidelberg, 2014, pp. 122–141.

[BM15] ——, *Universal Computational Extractors and the Superfluous Padding Assumption for Indistinguishability Obfuscation*, Cryptology ePrint Archive, Report 2015/581, 2015.

[Can01] R. Canetti, “Universally Composable Security: A New Paradigm for Cryptographic Protocols”, in *Proceedings of the 42Nd IEEE Symposium on Foundations of Computer Science*, ser. FOCS ’01, Washington, DC, USA: IEEE Computer Society, 2001, pp. 136–145.

[CGH04] R. Canetti, O. Goldreich, and S. Halevi, “The random oracle methodology, revisited”, *Journal of the ACM*, vol. 51, no. 4, pp. 557–594, Jul. 2004.

[DFR+07] I. B. Damgård, S. Fehr, R. Renner, L. Salvail, and C. Schaffner, “A Tight High-Order Entropic Quantum Uncertainty Relation with Applications”, in *Advances in Cryptology - CRYPTO 2007*, Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 360–378.

[DGHM13] G. Demay, P. Gaži, M. Hirt, and U. Maurer, “Resource-Restricted Indifferentiability”, in *Advances in Cryptology – EUROCRYPT 2013: 32nd Annual International Conference on the Theory and Applications of Cryptographic Techniques, Athens, Greece, May 26-30, 2013. Proceedings*, Springer Berlin Heidelberg, 2013, pp. 664–683.

[DKZZ15] K. Durnoga, T. Kazana, M. Zając, and M. Zdanowicz, “Leakage-resilient Cryptography with key derived from sensitive data”, *CoRR*, Jan. 2015. arXiv: `1502.00172`.

[FM16] P. Farshim and A. Mittelbach, “Modeling Random Oracles Under Unpredictable Queries”, in *Fast Software Encryption: 23rd International Conference, FSE 2016, Bochum, Germany, March 20-23, 2016, Revised Selected Papers*, T. Peyrin, Ed., Springer Berlin Heidelberg, 2016, pp. 453–473.

[MRH04] U. Maurer, R. Renner, and C. Holenstein, “Indifferentiability, impossibility results on reductions, and applications to the random oracle methodology”, *Theory of cryptography*, 2004.

[Mau13] U. Maurer, “Conditional equivalence of random systems and indistinguishability proofs”, in *2013 IEEE International Symposium on Information Theory*, IEEE, Jul. 2013, pp. 3150–3154.

[MR11] U. Maurer and R. Renner, “Abstract cryptography”, *In Innovations in Computer Science*, 2011.

[MR16] ——, *From Indifferentiability to Constructive Cryptography (and Back)*, Cryptology ePrint Archive, Report 2016/903, 2016.

[Mit14] A. Mittelbach, “Salvaging Indifferentiability in a Multi-stage Setting”, in *Advances in Cryptology – EUROCRYPT 2014*, Springer Berlin Heidelberg, 2014, pp. 603–621.

[RSS11] T. Ristenpart, H. Shacham, and T. Shrimpton, *Careful with Composition: Limitations of Indifferentiability and Universal Composability*, Cryptology ePrint Archive, Report 2011/339, 2011.

[ST17] P. Soni and S. Tessaro, “Public-Seed Pseudorandom Permutations”, in *Advances in Cryptology – EUROCRYPT 2017: 36th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Paris, France, April 30 – May 4, 2017, Proceedings, Part II*, Springer International Publishing, 2017, pp. 412–441.

[Wul07] J. Wullschleger, “Oblivious-Transfer Amplification”, in *Advances in Cryptology - EUROCRYPT 2007: 26th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Barcelona, Spain, May 20-24, 2007. Proceedings*, Springer Berlin Heidelberg, 2007, pp. 555–572.