Security Definitions for Hash Functions: Combining UCE and Indifferentiability

Daniel Jost(✉) and Ueli Maurer

Department of Computer Science, ETH Zurich, 8092 Zurich, Switzerland
{daniel.jost,maurer}@inf.ethz.ch

Abstract. Hash functions are one of the most important cryptographic primitives, but their desired security properties have proven to be remarkably hard to formalize. To prove the security of a protocol using a hash function, nowadays often the random oracle model (ROM) is used due to its simplicity and its strong security guarantees. Moreover, hash function constructions are commonly proven to be secure by showing them to be indifferentiable from a random oracle when using an ideal compression function. However, it is well known that no hash function realizes a random oracle and no real compression function realizes an ideal one.

As an alternative to the ROM, Bellare et al. recently proposed the notion of universal computational extractors (UCE). This notion formalizes that a family of functions "behaves like a random oracle" for "real-world" protocols while avoiding the general impossibility results. However, in contrast to the indifferentiability framework, UCE is formalized as a multi-stage game without clear composition guarantees.

As a first contribution, we introduce context-restricted indifferentiability (CRI), a generalization of indifferentiability that allows us to model that the random oracle does not compose generally but can only be used within a well-specified set of protocols run by the honest parties, thereby making the provided composition guarantees explicit. We then show that UCE and its variants can be phrased as a special case of CRI. Moreover, we show how our notion of CRI leads to generalizations of UCE. As a second contribution, we prove that the hash function constructed by Merkle-Damgård satisfies one of the well-known UCE variants, if we assume that the compression function satisfies one of our generalizations of UCE, basing the overall security on a plausible assumption. This result further validates the Merkle-Damgård construction and shows that UCE-like assumptions can serve both as a valid reference point for modular protocol analyses, as well as for the design of hash functions, linking those two aspects in a framework with explicit composition guarantees.

1 Introduction

1.1 Motivation and Background

The random oracle model (ROM) [3] is an important tool towards establishing confidence in the security of real-world cryptographic constructions. The

D. Catalano and R. De Prisco (Eds.): SCN 2018, LNCS 11035, pp. 83–101, 2018.
https://doi.org/10.1007/978-3-319-98113-0_5

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 [10] that no hash function realizes a random oracle; hence, once the random oracle is instantiated the security proof degenerates to a heuristic security argument.

The ROM is not only used as a model to prove protocols in, but it also serves as a reference point for the designers of hash functions. The indifferentiability framework [16], while being a general framework, is most famously used to phrase the security obligation of a hash function construction: the hash function is proven indifferentiable from a random oracle when using an ideal compression function (e.g. a fixed input-length random oracle), thereby excluding attacks exploiting the construction. Since indifferentiability is equipped with a composition theorem, this guarantee holds moreover irrespective of the context the hash function is used in. However, just as no hash function can instantiate a random oracle, no real compression function can instantiate the idealized version assumed in the proof.

More recently, Bellare et al. [2] 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 hash functions to "behave like a random oracle" and formalizes families of security notions aimed at bridging the gap between the general impossibility result and the apparent security of concrete protocols. So far, the research on the UCE framework has mainly been focused on two aspects: first, studying in which applications the ROM can be safely replaced by one of the UCE assumptions and, second, studying which ones of the UCE assumptions are generally uninstantiable and which one might actually be. Little attention, however, has been paid analyzing common hash function constructions within the UCE framework. Moreover, UCE is formalized as a multi-stage game without clear composition guarantees, which makes it therefore hard to directly apply as a modular step in an analysis of a complex protocol.

1.2 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 cannot be instantiated in every context but only within a well-specified set of contexts. We then mainly apply the general context-restricted indifferentiability framework to the random oracle, called *random-oracle context-restricted indifferentiability (RO-CRI)* security.

Secondly, we show that every UCE-class, i.e., every variant of the original UCE framework introduced by Bellare et al., can be expressed as a set of non-interactive contexts in which the random oracle can be instantiated. Hence, we prove that the UCE framework can be translated to RO-CRI and, thus, is

essentially a special case of it. Thereby we propose an alternative interpretation of the UCE framework in a traditional single-stage adversary model with well-defined composition guarantees and provide a direct relation between the UCE and the indifferentiability frameworks. In the full version [14] we furthermore show how two of the generalizations of UCE can be expressed within RO-CRI as well. Viewing UCE as a special case of CRI then allows us to generalize the split-source UCE-class to non-interactive contexts and we propose in particular a generalization that we call strong-split security.

Finally, we propose to consider CRI to analyze the security of common hash-function constructions. In contrast to indifferentiability, CRI allows us to consider more fine-grained versions of both the assumption on the compression function as well as the guarantee of the constructed hash function. As an example, we investigate the split-security of the Merkle-Damgård scheme using RO-CRI and we prove that the constructed hash function 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) if the hashed message has a sufficient min-entropy density from the distinguisher's point of view. We thereby generalize a lemma on min-entropy splitting by Damgård et al., which we believe might be of independent interest.

1.3 Related Work

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

Variants of Indifferentiability. Several variants of indifferentiability have been proposed in the past. The reset indifferentiability notion has been introduced by Ristenpart, Shacham, and Shrimpton in [20] as a workaround to the composition problems in multi-stage settings they highlighted. In [12], Demay et al. gave an alternative interpretation of those shortcomings. They prove that reset indifferentiability is equivalent to indifferentiability with stateless simulators. Moreover, they introduce 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. In contrast to our CRI notion that weakens indifferentiability, those two variants are a strengthening, i.e., any statement in those frameworks implies the traditional indifferentiability statement, but not vice-versa.

In [19], Mittelbach presents a condition called unsplittability on multi-stage games, that allows to show that the composition theorem of indifferentiability can be salvaged for iterative hash function constructions. They formalize a condition that specifies certain multi-stage games, in which the random oracle can be safely instantiated by an iterated hash function based on an idealized compression function. In contrast, CRI formalizes in which single-stage settings a hash function might be instantiable by an actual hash function, without having to assume an unrealistically ideal compression function. Moreover, CRI is a general paradigm that not only applies to iterative hash function constructions.

Universal Computation Extractors and Variants Thereof. The UCE framework was introduced by Bellare et al. [1] 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. [6,8] and Bellare et al. [4], and the possibility of a specific UCE-class in the standard model has been shown by Brzuska and Mittelbach [7]. Bellare et al. [2] 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 based on finding more efficient constructions if they only require the UCE-security of the variable input-length random oracle. In contrast, we consider the domain extension in a setting where we also assume the compression function to be only UCE secure.

In [13], Farshim and Mittelbach introduced a generalization of UCE called interactive computational extractors (ICE). 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 where the two stages jointly form the queries, requiring that neither one of them can predict the query. In contrast, we exactly use the asymmetry of UCE to embed it in the traditional indifferentiability setting with one dishonest and one honest party, where naturally the honest party knows the position where it queries the hash function.

In [21], 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. In the full version [14], we show that CRI is a strict generalization of their notion as well.

2 Preliminaries

2.1 The (Traditional) UCE Framework

To circumvent the well-known impossibility result that no hash function family is indifferentiable from a random oracle, Bellare, Hoang, and Keelveedhi [2] 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 $\mathrm{UCE}_H^{S,D}(\lambda)$
  $b \xleftarrow{\$} \{0,1\}$; $hk \xleftarrow{\$} \mathrm{H.Kg}(1^\lambda)$
  $L \xleftarrow{\$} S^{\mathrm{HASH}}(1^\lambda)$
  $b' \xleftarrow{\$} D(1^\lambda, hk, L)$
  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 assuming iO exists [6].

2.2 Resources and Converters

The indifferentiability framework by Maurer, Renner, and Holenstein [16] 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 (analogous to functionalities in the Universal Composability framework [9]). 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 [18], 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 $\mathtt{A}$ 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 resource $\square \in \Phi$ such that $[\mathsf{R}, \square] = \mathsf{R}$ for any 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 $\mathsf{id}\mathsf{R} = \mathsf{R}\,\mathsf{id} = \mathsf{R}$.

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 $\mathcal{X}$, $x \xleftarrow{\$} \mathcal{X}$ denotes assigning x uniformly at random a value in $\mathcal{X}$. Furthermore, $x \xleftarrow{\mathsf{P}_X} \mathcal{X}$ 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*.

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 (Fig. 1).

Fig. 1. The real (left) and the ideal (right) setting considered in indifferentiability. We depict resources using rectangular boxes and converters using rounded boxes. The honest party's interface is depicted on the left, and the dishonest's on the right side.

Indifferentiability formalizes this by demanding that there exists an efficient simulator σ, such that the real setting $\pi\mathsf{R}$ and the ideal setting $\mathsf{S}\sigma$ are indistinguishable according to the following definitions.

Definition 1. *The advantage of D in distinguishing R and S is defined as*

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

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

Definition 2 (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*

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

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

The formalism of indifferentiability composes in the natural way under some standard closure assumptions[1] 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:

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

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

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

[1] The set of distinguishers $\mathcal{D}$ needs to be closed under emulation of a resource and converter. The set of converters needs to be closed under sequential composition and parallel composition with the identity converter.

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 UC framework). 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 is then phrased as constructing the resource S from an assumed resource R and the fundamental idea of composition is 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 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 [10] implies, that there exists no deterministic and stateless protocol h, implementing a hash function, such that $\mathsf{HK} \overset{\mathsf{h}}{\Longrightarrow} \mathsf{PO}$, i.e., such that the hash function reduces the random oracle to the public hash key.

Traditionally, such an impossibility result is circumvented by weakening the guarantees S, and instead consider a restricted variant S'. However, for the random oracle, and many other examples, no such natural weakened version exists. As a second approach, one can restrict the class of distinguishers allowed. 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. In the following, we turn this motivation into a third approach: We restrict composition in a well-defined way. If there is a resource S that cannot be reduced to a resource R 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 P and some converter f applied by the honest party. 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. *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 Fig. 2 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*.

Fig. 2. The real (left) and the ideal (right) setting considered in context-restricted indifferentiability for a specific context (f, P) consisting of the filter f and the auxiliary parallel resource P.

Definition 4. *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 $\pi \in \Sigma$ in the sense of indifferentiability (denoted $\mathsf{R} \overset{\pi,\mathcal{C}}{\underset{\mathsf{cr}}{\Longrightarrow}} \mathsf{S}$), as*

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

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 [15,17], 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.

Before stating the composition theorem, we first observe that when a resource S is reduced to R in a context (f, P), the overall environment of S actually consists

of both (f, P) and the distinguisher. Especially, if S can be reduced to R within (f, P), so can it within $(\mathsf{f}' \circ \mathsf{f}, [\mathsf{P}, \mathsf{P}'])$, as f' and P' can be absorbed into the distinguisher. This leads to the following closure operation on context sets.

Definition 5. *Let $\mathcal{C} \subseteq \Sigma \times \Phi$ be a given set of contexts. We denote by $\overline{\mathcal{C}} \subseteq \Sigma \times \Phi$ the following set of contexts:*

$$\overline{\mathcal{C}} := \{(\mathsf{f}, \mathsf{P}) \in \Sigma \times \Phi \mid \exists (\mathsf{g}, \mathsf{Q}) \in \mathcal{C}\ \exists \mathsf{h} \in \Sigma\ \exists \mathsf{U} \in \Phi :\ \mathsf{h} \circ \mathsf{g} = \mathsf{f}\ \wedge\ [\mathsf{Q}, \mathsf{U}] = \mathsf{P}\}.$$

The following proposition is proven in the full version of this work [14].

Proposition 1. *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* $\mathsf{R} \overset{\pi, \mathcal{C}}{\underset{\mathsf{cr}}{\Longrightarrow}} \mathsf{S} \iff \mathsf{R} \overset{\pi, \overline{\mathcal{C}}}{\underset{\mathsf{cr}}{\Longrightarrow}} \mathsf{S}$.

Finally, the composition theorem of CRI can be stated.

Theorem 1. *Let $\mathsf{R}, \mathsf{S}, \mathsf{T}$, and U denote resources, let π_1 and π_2 denote protocols, and $\mathcal{C}_1$ and $\mathcal{C}_2$ contexts sets. We have*

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

iff for all $(\mathsf{f}, \mathsf{P}) \in \mathcal{C}_2$ it holds that $(\mathsf{f} \circ [\pi_2, \mathsf{id}], \mathsf{P}) \in \overline{\mathcal{C}_1}$. Moreover, we have

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

iff for all $(\mathsf{f}, \mathsf{P}) \in \mathcal{C}_2$ it holds that $(\mathsf{f}, [\mathsf{U}, \mathsf{P}]) \in \overline{\mathcal{C}_1}$.

The proof can be found in the full version [14]. Note that the additional conditions compared to the composition theorem of classical indifferentiability (cf. Sect. 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.

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 $\mathcal{C}_{\mathsf{id}} := \{(\mathsf{id}, \square)\}$, since $\overline{\mathcal{C}_{\mathsf{id}}} = \Sigma \times \Phi$, 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 considered resources and protocols, respectively. This can be summarized in the following proposition.

Proposition 2. *Let* $\mathcal{C}_{\mathrm{id}} \coloneqq \{(\mathsf{id}, \square)\}$. *For all resources* R, S, *protocol* π, *and context sets* $\mathcal{C} \subseteq \Sigma \times \Phi$, *we have*

$$\mathsf{R} \overset{\pi}{\Longrightarrow} \mathsf{S} \iff \mathsf{R} \underset{\mathsf{cr}}{\overset{\pi,\mathcal{C}_{\mathrm{id}}}{\Longrightarrow}} \mathsf{S},$$

$$\mathsf{R} \underset{\mathsf{cr}}{\overset{\pi,\mathcal{C}}{\Longrightarrow}} \mathsf{S} \iff \forall (\mathsf{f}, \mathsf{P}) \in \mathcal{C}\colon\ [\mathsf{R}, \mathsf{P}] \overset{\mathsf{f} \circ [\pi, \mathsf{id}]}{\Longrightarrow} \mathsf{f}[\mathsf{S}, \mathsf{P}].$$

Using $\overline{\mathcal{C}_{\mathrm{id}}} = \Sigma \times \Phi$, it is also easy to see that the composition theorem of regular indifferentiability is just a special case of Theorem 1.

4 Generalizing UCE Using CRI

In the following section we consider the ROM in context-restricted indifferentiability, i.e., consider the special case of CRI where the ideal-world resource S that we reduce is a random oracle. In the first subsection we prove that the UCE framework is actually a special case of CRI with a random oracle, and in the second subsection we propose a generalization of the split-security UCE-class based on CRI.

4.1 Modeling UCE in CRI

In the following, let $H\colon H.\mathcal{K} \times H.\mathcal{X} \to H.\mathcal{Y}$ denote a keyed hash function, let HK_{H} denote the public hash-key resource that chooses a key for H and outputs it at both interfaces, let $\mathsf{hash}_{\mathsf{H}}$ denote the converter that implements an oracle for H at the outside interface when connected to HK_{H} at the inside interface, and let $\mathsf{H} \coloneqq \mathsf{hash}_{\mathsf{H}}\mathsf{HK}_{\mathsf{H}}$ as a shorthand. Finally, let RO_{H} denote the private random oracle resource with the same input and output domains as H, where by private we mean that this resource only accepts queries at interface $\mathtt{A}$.[2]

We now present an alternative formalization of UCE based on context-restricted indifferentiability, more concretely that every possible UCE-class $\mathcal{S}^x$, where $x \in \{\mathrm{sup}, \mathrm{cup}, \mathrm{srs}, \mathrm{crs}, \mathrm{splt}, \ldots\}$, can be mapped to a set of contexts $\mathcal{C}^x$ for which the UCE statement implies the context-restricted indifferentiability statement $\mathsf{HK}_{\mathsf{H}} \underset{\mathsf{cr}}{\overset{\mathsf{hash}_{\mathsf{H}},\mathcal{C}}{\Longrightarrow}} \mathsf{RO}$, and moreover, if the CRI statement were restricted to a specific simulator, the reverse direction would hold as well.

In order to map every UCE-class 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. p. 5), 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.

[2] The choice to consider a private random oracle stems from the fact that in the UCE framework the hash key is just chosen uniformly at random instead of allowing an arbitrary efficient simulator with access to the random oracle to generate this key.

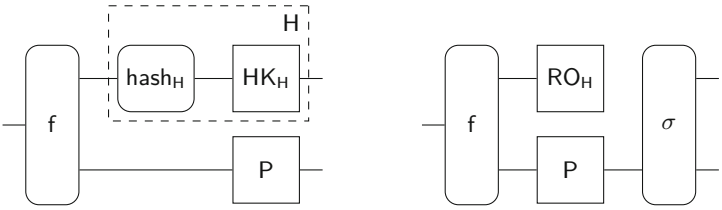

Fig. 3. The real (left) and the ideal (right) setting of context-restricted indifferentiability when applied to UCE.

Definition 6. *A* non-interactive resource P *is a resource that at the interface* $\mathtt{E}$ *accepts at most a single trigger query (usually called* `retrieve`*), and a* non-interactive filter *is a converter that at the outer interface just accepts a single trigger query (usually called* `retrieve`*). Let* Φ^{ni} *denote the set of all non-interactive resources, and* Σ^{ni} *denote the set of all non-interactive filters, respectively.*

Each UCE-source naturally corresponds to a set of non-interactive contexts. This is formally stated in the following lemma by providing a surjective mapping from the set of non-interactive contexts to the set of UCE sources $\mathcal{S}$.

Lemma 1. *The function* $\phi\colon \Sigma^{\mathsf{ni}} \times \Phi^{\mathsf{ni}} \to \mathcal{S}$ *that maps every context* (f, P) *to the following UCE source* S*, that internally emulates* f *and* P*, is surjective.*

1. S *queries the interface* $\mathtt{E}$ *of* P *to obtain* z*.*
2. S *queries the outside interface of the filter* f *to obtain* y*. The queries at the inside interface of* f *are forwarded to the resource* P *or output as queries to the hash oracle, respectively.*
3. S *outputs* $L = (y, z)$*.*

We now show, that for the specific simulator $\sigma_{\mathtt{H}}$ that chooses the hash key uniformly at random, the distinguishing problem of context-restricted indifferentiability for a fixed context (f, P) is as hard as the UCE game with the source $\phi(\mathsf{f}, \mathsf{P})$. In order to relate more directly to the traditional UCE definition, we first introduce the RO-CRI advantage, which is depicted in Fig. 3 for a specific context $(\mathsf{f}, \mathsf{P}) \in \mathcal{C}$.

Definition 7. *We define the* random-oracle context-restricted indifferentiability (RO-CRI) *advantage of a distinguisher* D *on a hash function* H *in a context* (f, P) *as*

$$\mathbf{Adv}^{\text{RO-CRI}}_{\mathsf{H},\mathsf{f},\mathsf{P},\sigma}(\mathsf{D}) \coloneqq \Delta^{\mathsf{D}}(\mathsf{f}[\mathsf{H}, \mathsf{P}], \mathsf{f}[\mathsf{RO}_{\mathsf{H}}, \mathsf{P}]\sigma),$$

for a simulator σ*. If there exists a simulator* σ *such that for all efficient distinguishers and all contexts* $(\mathsf{f}, \mathsf{P}) \in \mathcal{C}$*, the RO-CRI advantage is negligible, we say that* H *is* $\mathcal{C}$ *random-oracle context-restricted 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 $\sigma_{\mathtt{H}}$. The proof can be found in the full version of this work [14].

Lemma 2. *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 1. For every distinguisher D, there is a distinguisher D' (with essentially the same efficiency) with*

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

where $\mathbf{Adv}^{\text{uce}}_{H,S,D}$ denotes the uce-advantage of (S, D) on H. Conversely, for every distinguisher D' there is an equally efficient distinguisher D such that for all $(\mathsf{f},\mathsf{P}) \in \Sigma^{\mathsf{ni}} \times \Phi^{\mathsf{ni}}$ we have $\mathbf{Adv}^{\text{uce}}_{H,\phi(\mathsf{f},\mathsf{P}),\mathsf{D}'} = \mathbf{Adv}^{\text{RO-CRI}}_{\mathsf{H},\mathsf{f},\mathsf{P},\sigma_{\mathtt{H}}}(\mathsf{D})$.

We now state the main result of this section, relating the UCE game to context-restricted indifferentiability. It 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.

Theorem 2. *Let $\mathcal{D}$ denote the set of all efficient distinguishers. For every class $\mathcal{S}^x$ of UCE sources, there exists a set of contexts $\mathcal{C}^x$ such that $\mathbf{Adv}^{\text{RO-CRI}}_{\mathsf{H},\mathsf{f},\mathsf{P},\sigma_{\mathtt{H}}}(\mathsf{D})$ is negligible for every $D \in \mathcal{D}$ and every context $(\mathsf{f},\mathsf{P}) \in \mathcal{C}^x$ if and only if $\mathbf{Adv}^{\text{uce}}_{H,S,D}(\cdot)$ is negligible for all $(S, D) \in \mathcal{S}^x \times \mathcal{D}$.*

Proof. Using the surjectivity of ϕ (Lemma 1), we have that for any UCE-class $\mathcal{S}^x$ we can define $\mathcal{C}^x \coloneqq \phi^{-1}(\mathcal{S}^x)$ such that $\phi(\mathcal{C}^x) = \mathcal{S}^x$. Hence, by Lemma 2 we have that $\mathbf{Adv}^{\text{RO-CRI}}_{\mathsf{H},\mathsf{f},\mathsf{P},\sigma_{\mathtt{H}}}(\mathsf{D})$ is negligible for all efficient distinguishers $\mathsf{D} \in \mathcal{D}$ and all contexts $(\mathsf{f},\mathsf{P}) \in \mathcal{C}^x$ iff $\mathbf{Adv}^{\text{uce}}_{H,S,D}(\cdot)$ is negligible for all $(S, D) \in \mathcal{S}^x \times \mathcal{D}$.

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

Proof. This follows directly from Definitions 4 and 7 and Theorem 2.

4.2 Generalizing Split Security

In this section, we present a generalization of the split-source UCE-class, that cannot be formalized in plain UCE, based on CRI. The split-source UCE-class has been proposed by Bellare et al. after it has been shown that computational-unpredictable UCE-security and computational-reset-secure UCE-security is infeasible if indistinguishability obfuscation exists. Note that split-security is not a stand-alone UCE-class in the sense that it is designed to be combined with either computational unpredictability or reset-security, respectively.

The general idea of split-security is, that the source must not be able to compute $\text{Obfs}(H(\cdot, x) = y)$. To achieve this, the source must be dividable into two parts (S_0, S_1), where S_0 chooses a vector $(x_1, \ldots, x_n)$ of query points, without having access to the hash oracle, and S_1 then just gets the evaluations $y_i \coloneqq \text{Hash}(x_i)$, without having access to the hash oracle either.

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. Second, split-security does not allow nested queries like $H(hk, H(hk, x))$. In the full version [14] we present a further generalization to address this issue as well.

Remark 1. Note that the first limitation is not specific to split-security, but is inherent to the traditional UCE-game. In their work [13] on Interactive Computational Extractors (ICEs), Farshim and Mittelbach have proposed an alternative relaxation of this issue. In the full version [14], we show that ICE security implies strong-split context-restricted indifferentiability for statistical unpredictability.

In order to allow the distinguisher to influence the queries while ensuring that the overall query is still unpredictable from the viewpoint of the distinguisher, we allow him to apply any *injective* function on the preliminary inputs x specified by the first part of the source S_0, which will then be evaluated and passed on to S_1. That is, we use the simple fact that for any injective function f guessing $f(x_i)$ is at least as hard as guessing x_i. To formally model this as a context set for RO-CRI, we use a specific filter $\mathsf{f}_p^{\mathsf{s-splt}}$. This filter expects the resource P to output a sequence of pairs (x_i, a_i), where x_i is intended to be unpredictable. We will call such a resource P *seed* in the following. For each of them the distinguisher can then input p functions $f_i^1, \ldots, f_i^p$ that are injective in the first arguments, upon which the filter will output $(f_i^1(x_i, a_i), \ldots, f_i^p(x_i, a_i))$ to the hash oracle and forwards the results to the distinguisher. A formal definition of is depicted in Fig. 4. The filter $\mathsf{f}_p^{\mathsf{s-splt}}$ can then be combined with an arbitrary non-interactive resource to obtain a strong-split RO-CRI context.

Definition 8. *The* strong-split *RO-CRI context set is the set of filters and non-interactive resource pairs of which the filter can be factorized into* $\mathsf{f}_p^{\mathsf{s-splt}}$ *followed by an arbitrary filter. Formally,*

$$\mathcal{C}_p^{\mathsf{s-splt}} \coloneqq \{\mathsf{f} \circ \mathsf{f}_p^{\mathsf{s-splt}} \mid \mathsf{f} \in \Sigma\} \times \Phi^{\mathsf{ni}}.$$

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. However, for strong-split security, requiring the seed to output distinct unpredictable values is still insufficient to guarantee the security: for instance, if the resource P outputs (x, a_1) and $(x + 1, a_2)$, then the distinguisher can easily choose f and g such that $f(x, a_1) = g(x + 1, a_2)$. Therefore, we introduce suitable notion of unpredictability in the next subsection, which when combined with strong-split security presents a plausible assumption for a hash function family.

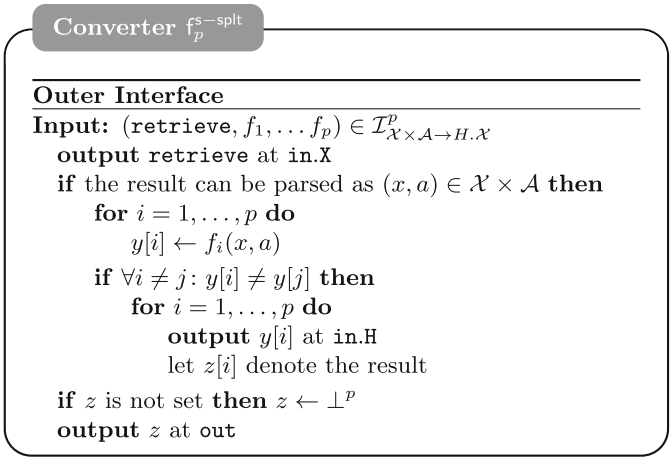

Fig. 4. The strong-split filter $\mathsf{f}_p^{\mathsf{s-splt}}$ for RO-CRI, where $\mathcal{I}_{\mathcal{X}\times\mathcal{A}\to H.\mathcal{X}}$ denotes the set of all efficiently computable functions from $\mathcal{X} \times \mathcal{A}$ to $H.\mathcal{X}$ that are injective in the first argument. Note that it was pointed out in [7] that the queries of a split-source must be distinct; otherwise arbitrary information can be communicated to the second stage.

Strict Min-Entropy Seeds. We now define an information-theoretic restriction on the seed called *strict min-entropy seeds*. Similar to Farshim and Mittelbach [13] we choose to focus on statistical rather than computational unpredictability to ensure that our notion excludes interactive version of the attack highlighted in [6].[3] More concretely, we consider seeds whose outputs at interface A consist of pairs (X_i, A_i), with A_i being an auxiliary value, such that X_i has high *average conditional min-entropy* given the leakage Z and all previous queries.

Definition 9. *A* strict min-entropy k-bit seed with n outputs *is a resource that initially draws random values* $X_1, \ldots, X_n$, $A_1, \ldots, A_n$, *and* Z *according to some joint distribution, such that*

$$\forall i \leq n\colon \ \tilde{H}_\infty\left(X_i \middle| \{X_j\}_{j<i}, \{A_j\}_{j\leq i}, Z\right) \ \geq \ k.$$

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* (X_1, A_1) *to* (X_n, A_n)*. Let* $\Phi^{\mathsf{s-me}}_{n,k} \subset \Phi^{\mathsf{ni}}$ *denote the set of all strict min-entropy* k*-bit seed with* n *outputs. 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 maximal number n of values

[3] We would like to stress that while split-security was originally introduced for the computational setting, it is still a natural class to consider even when combined with a statistical unpredictability notion.

produced by the seed. The following lemma is proven in the full version [14], using a simple hybrid-argument.

Lemma 3. *Let n be polynomially bounded. If H is a $\mathcal{C}_p^{\mathsf{s-splt}} \cap \mathcal{C}_{1,k}^{\mathsf{s-me}}$ indifferentiable hash function, then H is also $\mathcal{C}_p^{\mathsf{s-splt}} \cap \mathcal{C}_{n,k}^{\mathsf{s-me}}$ indifferentiable.*

5 Split Security of the Merkle-Damgård Construction

Indifferentiability is widely used to prove the security of hash function constructions. Since CRI is essentially a refined version of indifferentiability, it is hence natural to consider the RO-CRI security as well. It is easy to show that any indifferentiable hash function construction is reset-UCE secure if the underlying compression function is reset-UCE secure. On the other hand, for split security no corresponding result has been proven so far. In the following we investigate the split-security of the Merkle-Damgård construction using the RO-CRI framework. While ideally one could prove that the Merkle-Damgård construction is split secure if the compression function is so, or that the Merkle-Damgård construction is strong-split secure if the compression function is so, we will prove a slightly weaker result:

Consider the Merkle-Damgård construction that splits the message into blocks of length m. We show that the Merkle-Damgård construction is split-secure for inputs having at least one block with k bits of min-entropy, if the compression function is strong-split secure for inputs with $\min(k, m)$ *bits of min-entropy.*

In contrast to the definition of strict min-entropy seeds (c.f. Definition 9) we require that at least one of the blocks has high min-entropy and not just the overall message has. Moreover, in order for the proof to actually work, we require that this block has k bits of min-entropy given all subsequent blocks. In Lemma 4 we then show that having a high min-entropy density, i.e., the fraction between the min-entropy and the message length, is a sufficient criteria for this. First, however, let us formally introduce this CRI context set.

Definition 10. *For a block length $\ell \in \mathbb{N}_+$, let Pad_ℓ denote the usual padding scheme of the Merkle-Damgård scheme, that is $Pad_\ell\colon \{0,1\}^* \to (\{0,1\}^\ell)^+$ that pads a message x by first appending zeros up to a multiple of the block length ℓ, and then appending an encoding of the number of zeros appended as a last block. Moreover, for $X \in \{0,1\}^*$, we denote by X^i the i-th block of $Pad_\ell(X)$.*

Definition 11. *A non-interactive resource is said to be a k* out of ℓ-bit strict min-entropy block*, denoted $\mathsf{P} \in \Phi_{k,\ell,b,n}^{\mathsf{me-blk}}$, if $\mathsf{P} \in \Phi_{k,n}^{\mathsf{s-me}}$ with $\bigcup_{i \le (b-1)\ell} \{0,1\}^i \times \mathcal{A}$ as the output domain of interface $\mathtt{A}$, and there exist random variables $C_1, \ldots, C_n$ such that $C_i \in \{1, \ldots, \frac{|Pad_\ell(X_i)|}{\ell}\}$ and*

$$\forall i \le n\colon\ \tilde{H}_\infty(X_i^{C_i} \big| \{X_i^j\}_{j>C_i}, \{X_j\}_{j<i}, \{C_j\}_{j\le i}, \{A_j\}_{j\le i}, Z) \ \ge\ k.$$

Moreover, let $\mathcal{C}_{k,\ell,b,n}^{\mathsf{me-blk}} \coloneqq \Sigma \times \Phi_{k,\ell,b,n}^{\mathsf{me-blk}}$.

Note, that contrary to the classical indifferentiability of the Merkle-Damgård construction, we do not require Pad to be prefix-free: when combined with the strict min-entropy condition $H(X)$ cannot be extended to $H(\mathrm{Pad}(X)||Y)$, as for $\mathrm{Pad}(X)||Y$ having high min-entropy given X, Y must have so, and thereby the well-known length-extension attack is excluded. Whether a more advanced construction with a finalization function, e.g. HMAC, could be proven secure for a more relaxed context set remains an interesting open problem. We now phrase our main result of this section; the proof can be found in the full version [14].

Theorem 3. *Let* $h\colon \{0,1\}^{m+\ell} \to \{0,1\}^m$ *denote a fixed input-length compression function,* $H\colon \{0,1\}^* \to \{0,1\}^m$ *denote the hash function obtained by first padding the message using* Pad_ℓ *and then applying the Merkle-Damgård scheme using* h*, and let* $k' := \min(k, m)$*. Then, if* h *is* $\mathcal{C}_1^{\mathsf{s-splt}} \cap \mathcal{C}_{1,k'}^{\mathsf{s-me}}$ *RO-CRI secure, then* H *is* $\mathcal{C}^{\mathsf{splt}} \cap \mathcal{C}_{k,\ell,b,n}^{\mathsf{me-blk}}$ *RO-CRI secure for any polynomial* b *and* n*.*

To conclude this section, we now present a sufficient condition for a seed to satisfy Definition 11 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. In order to more closely resembles the chain rule of Shannon entropy, the proposition is stated with conditioning on all preceding message $X_1 \dots X_{C-1}$ instead of all succeeding ones. The converse result can easily be obtained by simply relabeling the blocks. The proof can be found in the full version of this work [14].

Lemma 4. *Let* $X_1, \dots, X_b$ *and* Z *be random variables (over possibly different alphabets) with* $\tilde{H}_\infty(X_1 \dots X_b \mid Z) \geq k$*. Then, there exists a random variable* C *over the set* $\{1, \dots, b\}$ *such that* $\tilde{H}_\infty(X_C \mid X_1 \dots X_{C-1} C Z) \geq k/b - \log_2(b)$*.*

This lemma is a generalization of the randomized chain rule proven by the authors of [11] (similar variants exists also in [5,22]) stating that there exists a binary random variable C such that $H_\infty(X_{1-C}C) \geq H_\infty(X_0X_1)/2$. Note that the main difference of our result is, that it conditions on all previous blocks, i.e., it essentially represents the min-entropy equivalence of the strong chain rule $H(X_0) + H(X_1 \mid X_0) = H(X_0X_1)$ instead of $H(X_0) + H(X_1) \geq H(X_0X_1)$.

References

1. Bellare, M., Hoang, V.T., Keelveedhi, S.: Instantiating random oracles via UCEs. In: Canetti, R., Garay, J.A. (eds.) CRYPTO 2013. LNCS, vol. 8043, pp. 398–415. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-40084-1_23
2. Bellare, M., Hoang, V.T., Keelveedhi, S.: Cryptography from compression functions: the UCE bridge to the ROM. In: Garay, J.A., Gennaro, R. (eds.) CRYPTO 2014. LNCS, vol. 8616, pp. 169–187. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-662-44371-2_10
3. Bellare, M., Rogaway, P.: Random oracles are practical: a paradigm for designing efficient protocols. In: 1st ACM Conference on Computer and Communications Security, CCS 1993, pp. 62–73. ACM Press, New York (1993)

4. Bellare, M., Stepanovs, I., Tessaro, S.: Contention in cryptoland: obfuscation, leakage and UCE. In: Kushilevitz, E., Malkin, T. (eds.) TCC 2016. LNCS, vol. 9563, pp. 542–564. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-49099-0_20
5. Brakerski, Z., Kalai, Y.T.: A parallel repetition theorem for leakage resilience. In: Cramer, R. (ed.) TCC 2012. LNCS, vol. 7194, pp. 248–265. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-28914-9_14
6. Brzuska, C., Farshim, P., Mittelbach, A.: Indistinguishability obfuscation and UCEs: the case of computationally unpredictable sources. In: Garay, J.A., Gennaro, R. (eds.) CRYPTO 2014. LNCS, vol. 8616, pp. 188–205. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-662-44371-2_11
7. Brzuska, C., Mittelbach, A.: Using indistinguishability obfuscation via UCEs. In: Sarkar, P., Iwata, T. (eds.) ASIACRYPT 2014. LNCS, vol. 8874, pp. 122–141. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-662-45608-8_7
8. Brzuska, C., Mittelbach, A.: Universal computational extractors and the superfluous padding assumption for indistinguishability obfuscation. Cryptology ePrint Archive, Report 2015/581 (2015). https://eprint.iacr.org/2015/581
9. Canetti, R.: Universally composable security: a new paradigm for cryptographic protocols. In: 42nd IEEE Symposium on Foundations of Computer Science, FOCS 2001, pp. 136–145. IEEE Computer Society (2001)
10. Canetti, R., Goldreich, O., Halevi, S.: The random oracle methodology, revisited. J. ACM **51**(4), 557–594 (2004)
11. Damgård, I.B., Fehr, S., Renner, R., Salvail, L., Schaffner, C.: A tight high-order entropic quantum uncertainty relation with applications. In: Menezes, A. (ed.) CRYPTO 2007. LNCS, vol. 4622, pp. 360–378. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-74143-5_20
12. Demay, G., Gaži, P., Hirt, M., Maurer, U.: Resource-restricted indifferentiability. In: Johansson, T., Nguyen, P.Q. (eds.) EUROCRYPT 2013. LNCS, vol. 7881, pp. 664–683. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-38348-9_39
13. Farshim, P., Mittelbach, A.: Modeling random oracles under unpredictable queries. In: Peyrin, T. (ed.) FSE 2016. LNCS, vol. 9783, pp. 453–473. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-52993-5_23
14. Jost, D., Maurer, U.: Security definitions for hash functions: combining UCE and indifferentiability. Cryptology ePrint Archive, Report 2017/461 (2017). https://eprint.iacr.org/2017/461. (Full version of this paper)
15. Maurer, U.: Constructive cryptography – a new paradigm for security definitions and proofs. In: Mödersheim, S., Palamidessi, C. (eds.) TOSCA 2011. LNCS, vol. 6993, pp. 33–56. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-27375-9_3
16. Maurer, U., Renner, R., Holenstein, C.: Indifferentiability, impossibility results on reductions, and applications to the random oracle methodology. In: Naor, M. (ed.) TCC 2004. LNCS, vol. 2951, pp. 21–39. Springer, Heidelberg (2004). https://doi.org/10.1007/978-3-540-24638-1_2
17. Maurer, U., Renner, R.: Abstract cryptography. In: Innovations in Computer Science, ICS 2011, pp. 1–21. Tsinghua University (2011)
18. Maurer, U., Renner, R.: From indifferentiability to constructive cryptography (and back). In: Hirt, M., Smith, A. (eds.) TCC 2016. LNCS, vol. 9985, pp. 3–24. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-53641-4_1

19. Mittelbach, A.: Salvaging indifferentiability in a multi-stage setting. In: Nguyen, P.Q., Oswald, E. (eds.) EUROCRYPT 2014. LNCS, vol. 8441, pp. 603–621. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-642-55220-5_33
20. Ristenpart, T., Shacham, H., Shrimpton, T.: Careful with composition: limitations of the indifferentiability framework. In: Paterson, K.G. (ed.) EUROCRYPT 2011. LNCS, vol. 6632, pp. 487–506. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-20465-4_27
21. Soni, P., Tessaro, S.: Public-seed pseudorandom permutations. In: Coron, J.-S., Nielsen, J.B. (eds.) EUROCRYPT 2017. LNCS, vol. 10211, pp. 412–441. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-56614-6_14
22. Wullschleger, J.: Oblivious-transfer amplification. In: Naor, M. (ed.) EUROCRYPT 2007. LNCS, vol. 4515, pp. 555–572. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-72540-4_32