Towards Defeating Backdoored Random Oracles: Indifferentiability with Bounded Adaptivity

Yevgeniy Dodis[1], Pooya Farshim[2], Sogol Mazaheri[3], and Stefano Tessaro[4]

[1] New York University
dodis@cs.nyu.edu
[2] University of York
pooya.farshim@gmail.com
[3] Technische Universität Darmstadt
sogol.mazaheri@cryptoplexity.de
[4] University of Washington
tessaro@cs.washington.edu

Abstract. In the backdoored random-oracle (BRO) model, besides access to a random function H, adversaries are provided with a backdoor oracle that can compute arbitrary leakage functions f of the function table of H. Thus, an adversary would be able to invert points, find collisions, test for membership in certain sets, and more. This model was introduced in the work of Bauer, Farshim, and Mazaheri (Crypto 2018) and extends the auxiliary-input idealized models of Unruh (Crypto 2007), Dodis, Guo, and Katz (Eurocrypt 2017), Coretti et al. (Eurocrypt 2018), and Coretti, Dodis, and Guo (Crypto 2018). It was shown that certain security properties, such as one-wayness, pseudorandomness, and collision resistance can be re-established by combining two independent BROs, even if the adversary has access to both backdoor oracles.

In this work we further develop the technique of combining two or more independent BROs to render their backdoors useless in a more general sense. More precisely, we study the question of building an *indifferentiable* and backdoor-free random function by combining multiple BROs. Achieving full indifferentiability in this model seems very challenging at the moment. We however make progress by showing that the xor combiner goes well beyond security against preprocessing attacks and offers indifferentiability as long as the adaptivity of queries to different backdoor oracles remains logarithmic in the input size of the BROs. We even show that an extractor-based combiner of three BROs can achieve indifferentiability with respect to a linear adaptivity of backdoor queries. Furthermore, a natural restriction of our definition gives rise to a notion of *indifferentiability with auxiliary input*, for which we give two positive feasibility results.

To prove these results we build on and refine techniques by Göös et al. (STOC 2015) and Kothari et al. (STOC 2017) for decomposing distributions with high entropy into distributions with more structure and show how they can be applied in the more involved adaptive settings.

Keywords: Hash functions · Indifferentiability · Backdoors · Auxiliary input · Communication complexity

1 Introduction

Hash functions are one of the most fundamental building blocks in protocol design. For this reason, both the cryptanalysis and provable security of hash functions have been active areas of research in recent years. The first known instances of collisions and chosen-prefix collisions in SHA-1 were recently demonstrated by Stevens et al. [26] and Leurent and Peyrin [20], respectively. Furthermore, feasibility of built-in adversarial weaknesses (aka. backdoors) in efficient hash functions have been demonstrated by Fischlin, Janson, and Mazaheri [13]. A practical way to provide safeguards against similar failures of hash functions is to *combine* a number of independent hash functions so that the resulting function is at least as secure as their strongest. Most works in this area have focused attention on a setting where at least one of the hash functions is secure, although positive results when *all* underlying hash functions have weaknesses have also been demonstrated [22,15].

In this work we are interested in protecting hash functions against a variety of attacks that may arise due to backdoors, cryptanalytic advances, or preprocessing attacks. We carry out our study in the recent backdoored random-oracle (BRO) model, which uniformly treats these settings and also permits strong adversarial settings where all hash functions may be weak.

1.1 The BRO model

Bauer, Farshim, and Mazaheri (BFM) [3] at Crypto 2018 formulated a new model for the analysis of hash functions that substantially weakens the traditional random-oracle (RO) model. Here an adversary, on top of direct access to the random oracle, is able to obtain arbitrary functions of the function table of the random oracle.[1] The implications of this weakening are manifold. To start with, positive results in this model imply positive results in the traditional setting where all but one of the hash functions is weak. Second, this model captures arbitrary preprocessing attacks on hash functions, another highly active area of research [27,10,7,6]. Finally, it allows to model unrestricted adversarial capabilities, which can adaptively depend on input instances, and thus captures built-in as well as inadvertent weaknesses that may or may not be discovered in course of time.

BFM studied three natural combiners in this setting: those of concatenation, cascade, and xor combiners:

$$\mathsf{C}_{|}^{\mathsf{H}_1,\mathsf{H}_2}(x) := \mathsf{H}_1(x)|\mathsf{H}_2(x) \qquad \mathsf{C}_{\circ}^{\mathsf{H}_1,\mathsf{H}_2}(x) := \mathsf{H}_2(\mathsf{H}_1(x))$$

$$\mathsf{C}_{\oplus}^{\mathsf{H}_1,\mathsf{H}_2}(x) := \mathsf{H}_1(x) \oplus \mathsf{H}_2(x) \ .$$

They showed, using new types of reductions to problems with high communication complexity, that central cryptographic security properties, such as one-way

[1] The model allows for a parameterization of the class of functions that can be computed. Both BFM and we here work with respect to the full set of functions.

security, pseudorandomness, and collision resistance are indeed achievable by these combiners.

The reductions to communication complexity problems are at times tedious and very specific to the combiner. Moreover, the hardness of the communication complexity problem underlying collision resistance is conjectural and still remains to be proven. Furthermore, a number of deployed protocols have only been shown to be secure in the random-oracle model, and thus may rely on properties beyond one-wayness, pseudorandomness, or collision resistance.

This raises the question whether or not other cryptographic properties expected from a good hash function are also met by these combiners. In other words:

Can combining two or more backdoored random oracles render access to independent but adaptive auxiliary information useless?

We formalize and study this question in the indifferentiability framework, which has been immensely successful in justifying the soundness of hash-function designs.

1.2 Indifferentiability

A common paradigm in the design of hash functions is to start with some underlying primitive, and through some construction build a more complex one. The provable security of such constructions have been analyzed through two main approaches. One formulates specific goals (such as collision resistance) and goes on to show that the construction satisfies them if its underlying primitives satisfy their own specific security properties. Another is a general approach, whose goal is to show that a (wide) class of security goals are simultaneously met.

The latter has been formalized in a number of frameworks, notably in the UC framework of Canetti [5], the reactive systems framework of Pfitzmann and Waidner [24], and the indifferentiability framework of Maurer, Renner, and Holenstein [23]. The latter is by now a standard methodology to study the soundness of cryptographic constructions, particularly symmetric ones such as hash functions [8,4] and block-ciphers [9,16,1,12] in idealized models of computation.

In the MRH framework, a public primitive H is available and the goal is to build another primitive, say a random oracle RO, from H through a construction C^{H}. Indifferentiability formalizes a set of necessary and sufficient conditions for the construction C^{H} to securely replace its ideal counterpart RO in a wide range of environments: for a *simulator* Sim, the systems $(\mathsf{C}^{\mathsf{H}}, \mathsf{H})$ and $(\mathsf{RO}, \mathsf{Sim}^{\mathsf{RO}})$ should be indistinguishable. The composition theorem proved by MRH states that, if C^{H} is indifferentiable from RO, then C^{H} can securely replace RO in arbitrary single-stage contexts. A central corollary of this composition theorem is that indifferentiability implies any single-stage security goal, which includes among others, one-wayness, collision resistance, PRG/PRF security, and more.

1.3 Contributions

With the above terminology in hand, the central question tackled in this work is whether or not combiners that are *indifferentiable* from a conventional (backdoor-free) random oracle exist, when the underlying primitives are two (or more) backdoored random oracles.

Let us consider the concatenation combiner $\mathsf{H}_1(x)|\mathsf{H}_2(x)$, where H_1 and H_2 are both backdoored. This construction was shown to be one-way, collision resistant, and PRG secure if both underlying functions are highly compressing. Despite this, the concatenation combiner cannot be indifferentiable from a random oracle: using the backdoor oracle for H_1 an attacker can compute two inputs x and x' such that $\mathsf{H}_1(x) = \mathsf{H}_1(x')$, query them to the construction and return 1 iff the left sides of the outputs match. However, any simulator attempting to find such a pair with respect to a backdoor-free random oracle must place an exponentially large number of queries. Attacks on the cascade combiner $\mathsf{H}_2(\mathsf{H}_1(x))$ were also given in [3, Section D.2] for a wider range of parameter regimes, leaving only the expand-then-compress case as potentially indifferentiable. Finally, the xor combiner $\mathsf{H}_1(x) \oplus \mathsf{H}_2(x)$, which is simpler, more efficient, and one of the most common ways to combine hash functions, resists these.[2]

Decomposition of distributions. When proving results in the presence of auxiliary input, Uhruh [27] observed that pre-computation (or leakage) on a random oracle can reveal a significant amount of information only on restricted parts of its support. The problem of dealing with auxiliary input was later revised in a number of works [10,7,6]. In particular Coretti et al. [7], building on work in communication complexity, employed a *pre-sampling technique* to prove a number of positive results in the RO model with auxiliary input with tighter bounds. At a high level, this method permits writing a high min-entropy distribution (here, over a set of functions) as the convex combination of a (large) number of distributions which are fixed on a certain number (p) of points and highly unpredictable on the rest, the so-called $(p, 1-\delta)$-dense distributions. This technique was originally introduced in the work of Göös et al. [14].

The simulator. Our simulator for the xor combiner builds on this technique to decompose distributions into a convex combination of $(p, 1-\delta)$-dense distributions. Simulation of backdoor oracles is arguably quite natural and proceeds as follows. Starting with uniform random oracles H_1 and H_2, on each backdoor query f for H_1 the simulator computes $z = f(\mathsf{H}_1)$ and updates the distribution of the random oracle H_1 to be uniform conditioned on the output of f being z. This distribution is then decomposed into a convex combination of $(p, 1-\delta)$-dense distributions, from which one function is sampled. For all of the p fixed points, the simulator sets the value of H_2 consistently with the random oracle and the distribution of H_2 is updated accordingly. An analogous procedure is implemented as the simulator for the second backdoored random oracle.

[2] Further, an indifferentiability proof of the expand-then-compress cascade combiner would closely follow that of the xor combiner and thus we focus on the latter here.

Technical analysis. The first technical contribution of our work is a refinement of the decomposition technique which can be used to *adaptively* decompose distributions after backdoor queries. We show that this refinement is sufficiently powerful to allow proving indifferentiability up to a logarithmic (in the input size of the BROs) number of switches between the backdoor queries. We prove this via a sequence of games which are carefully designed so as to be compatible with the decomposition technique. A key observation is that in contrast to previous works in the AI-RO model, we do not replace the dense (intuitively, unpredictable) part of the distribution of random oracles with uniform: backdoor functions "see" the entire table of the random oracle and this replacement would result in a noticeable change. Second, we modify the number of fixed points in the (partially) dense distributions so that progressively smaller sets of points are fixed. Even though each leakage corresponds to fixing a large number of points, it is proportionally smaller than the previous number of fixed points. Thus the overall bound remains small.

Simulator efficiency. Our simulator runs in doubly exponential time in the bit-length of the random oracle and thus is of use in information-theoretic settings. These include the vast majority of symmetric constructions. Protocols based on computational assumptions (such as public-key encryption) escape this treatment: the overall adversary obtained via the composition would run the decomposition algorithm and hence will not be poly-time. This observation, however, also applies to the BRO model as the backdoor oracles also allow for non-polynomial time computation, trivially breaking any computational assumption if unrestricted. Despite this, in a setting where the computational assumption holds relative to the backdoor oracles, positive results may hold. We can for example restrict the backdoor capability to achieve this. Another promising avenue is to rely on an independent idealized model such as the generic-group model (GGM) and for instance, prove IND-CCA security of Hashed ElGamal in the BRO and (backdoor-free) GGM models. We leave exploring these solutions to future work.

An extractor-based combiner with improved security. We apply the above proof technique to the analysis of an alternative combiner for three independent backdoored random oracles, which relies on 2-out-of-3-source extractors that output good randomness as long as two out of the three of the inputs have sufficient min-entropy. Given such an extractor Ext, our combiner is

$$\mathsf{C}_{3\mathsf{ext}}^{\mathsf{H}_1,\mathsf{H}_2,\mathsf{H}_3}(x) := \mathsf{Ext}\big(\mathsf{H}_1(x), \mathsf{H}_2(x), \mathsf{H}_3(x)\big) \ .$$

As mentioned above, our simulator for the xor combiner programs H_2 on the fixed points for H_1 (and vice versa) using the random oracle. This results in a loss since dense values are replaced with uniform values. In contrast, here the extractor ensures that image values are closer to uniform and thus the overall loss is lower. We show that a 2-out-of-3-source extractor can tolerate even a number of switches between the backdoor oracles which is slightly sub-linear in the size of the BRO inputs. This gives us more hope for unbounded adaptivity, in case improved decomposition techniques are found.

Composition. Let c denote the number of times the adversary switches between one backdoor oracle to the other. Regarding the query complexities of our simulators, each query to the backdoor oracle translates to roughly $N^{1-2^{-c}}$ queries to the random oracle for the xor combiner and roughly $N^{1-3/(c+3)}$ queries to the random oracle for the extractor combiner. This in particular means that, for a wide range of parameters, composition is only meaningful with respect to security notions whereby the random oracle can tolerate a large number of queries. This, for example, would be the case for one-way, PRG, and PRF security notions where the security bounds are of the form $\mathcal{O}(q/N)$. However, with respect to a smaller number of switches (as well as in the auxiliary-input setting with no adaptivity), collision resistance can still be achieved.

Indifferentiability with auxiliary input. When our definition of indifferentiability is restricted so that only a single backdoor query to each hash function at the onset is allowed, we obtain a notion that formalizes *indifferentiability with auxiliary input*. This definition is interesting as it is sufficiently strong to allow for the generic replacement of random oracles with iterative constructions even in the presence of preprocessing attacks. Accordingly, our positive results in the BRO model when considered with no adaptivity translate to indifferentiability with independent preprocessing attacks. To complement this picture, we also discuss the case of auxiliary-input indifferentiability with a single BRO and show, as expected, that a salted indifferentiable construction is also indifferentiable with auxiliary input.

Open problems. In order to overcome the bounded adaptivity restriction and prove full indifferentiability, one would require an improved decomposition technique which fixes considerably less points after each leakage. This, at the moment, seems (very) challenging and is left as an open question. In particular, such a result would simultaneously give new proofs of known communication complexity lower bounds for a host of problems, such as set-disjointness and intersection, potentially a proof of the conjecturally hard problem stated in [3], and many others. (We note that improved decomposition techniques can potentially also translate to improved bounds.) Indeed the xor combiner may achieve security well beyond what we establish here (and indeed the original work of BFM does so for specific games). Finally, as the extractor combiner suggests, the form of the combiner and the number of underlying BROs can also affect the overall bounds.

2 Preliminaries

Throughout the paper, when we write $[N]$ for any uppercase letter N, we use the convention that N is an integer and a power of two, i.e., $N = 2^n$ for some $n \in \mathbb{N}$. Let $[N] := \{0, \ldots, N-1\}$ denote the set of all n-bit strings. We use $[M]^N$ to denote the set of all bit-strings of length $N \cdot \log M$, which corresponds to the set of all functions $\mathsf{F} : [N] \to [M]$. We denote the uniform distribution over an arbitrary finite set S by $\mathcal{U}_S$.

For $\mathsf{F} \in [M]^N$ and $I \subseteq [N]$ we denote by F_I the projection of F onto the points in I. Let μ be a probability density function over $[M]^N$. We define $\mu(D) := \Pr_{\mathsf{F}\sim\mu}[\mathsf{F} \in D]$ as the probability that a sample randomly drawn from μ falls into the domain $D \subseteq [M]^N$. By $\mu|_D$ we denote the density μ conditioned on the domain D. For a function $f : [M]^N \to \{0,1\}^\ell$ and $z \in \{0,1\}^\ell$, by $\mu|_{f(\cdot)=z}$ we denote μ conditioned on $f(\mathsf{F}) = z$ for all $\mathsf{F} \sim \mu|_{f(\cdot)=z}$.

For a set of assignments $A \subseteq \{(a,b) : (a,b) \in [N] \times [M]\}$, by $\mu|_A$ we denote μ conditioned on $\mathsf{F}_{\{a\}} = b$ for all $(a,b) \in A$ and all $\mathsf{F} \sim \mu|_A$. We further let $A_{.1} \subseteq [N]$ (resp. $A_{.2} \subseteq [M]$) denote the set containing the first (resp. second) coordinates of all elements in A.

For an algorithm Alg we denote by $\mathsf{Alg}[param](input)$ a call of the algorithm with (constant) parameters *param* and variable inputs *input*. This is to increase clarity among multiple calls to the algorithm about the main input, while the parameters remain unchanged.

2.1 Backdoored random oracles

We recall the definition of the backdoored random-oracle model from [3]. The $\mathsf{BRO}(N_1, M_1, \ldots, N_k, M_k)$ model (for some $k \in \mathbb{N}$) defines a setting where all parties have access to k functions $\mathsf{H}_1, \ldots, \mathsf{H}_k$, where H_i's are chosen uniformly and independently at random from $[M_i]^{N_i}$, while the adversarial parties also have access to the corresponding backdoor oracles $\textsc{Bd}_i$'s. A backdoor oracle $\textsc{Bd}_i$ can be queried on functions f and return $f(\mathsf{H}_i)$. If for all $i \in [k]$ we have $N_i = N$ and $M_i = M$, we simply refer to this model as k-$\mathsf{BRO}(N, M)$ and when N and M are clear from the context, we simply use k-BRO.

These models may be weakened by restricting the adversary to query $\textsc{Bd}_i$ only on functions f in some capability class $\mathcal{F}_i$. However our results as well as those in [3] hold for arbitrary backdoor capabilities. In other words an adversary can (adaptively) query arbitrary functions f to any of the backdoor oracles.

2.2 Indifferentiability in the BRO model

We follow the indifferentiability framework of Maurer, Renner, and Holenstein (MRH) [23]. Here the underlying honest interfaces are k random oracles H_i and respective adversarial interfaces $\textsc{Bd}_i$. We define the advantage of a differentiator $\mathcal{D}$ with respect to a construction $\mathsf{C}^{\mathsf{H}_i}$ and a simulator $\mathsf{Sim}^{\mathsf{RO}} := (\mathsf{SimH}_i^{\mathsf{RO}}, \mathsf{SimBD}_i^{\mathsf{RO}})$ as

$$\mathsf{Adv}^{\text{indiff}}_{\mathsf{C}^{\mathsf{H}_i},\mathsf{Sim}}(\mathcal{D}) := \left| \Pr\left[\mathcal{D}^{\mathsf{C}^{\mathsf{H}_i},\mathsf{H}_i,\textsc{Bd}_i}\right] - \Pr\left[\mathcal{D}^{\mathsf{RO},\mathsf{SimH}_i^{\mathsf{RO}},\mathsf{SimBD}_i^{\mathsf{RO}}}\right] \right| ,$$

where RO is a random oracle whose domain and co-domain match those of C.

We emphasize that the simulators do not get access to any backdoor oracles. This ensures that any attack against a construction with backdoors translates to one against the underlying random oracles *without* any backdoors.

2.3 Randomness extractors

Let X be a random variable. The min-entropy of X is defined as $\mathbf{H}_\infty(X) := -\log \max_x \Pr[X = x]$. The random variable X is called a (weak) k-source if $\mathbf{H}_\infty(X) \geq k$, i.e., $\Pr[X = x] \leq 2^{-k}$. The min-entropy of a distribution typically determines how many bits can be extracted from it which are close to uniform. The notion of closeness is formalized by the statistical distance. For two random variables X and Y over a common support D, their statistical distance is defined as $\mathrm{SD}(X, Y) := \frac{1}{2} \sum_{z \in D} |\Pr[X = z] - \Pr[Y = z]|$.

In this paper we are interested in extractors that do not require seeds but rather rely on multiple weak sources.

Definition 1 (Multi-source extractors). *An efficient function* $\mathsf{Ext} : [N_1] \times \ldots \times [N_t] \to [M]$ *is a* $(k_1, \ldots, k_t, \varepsilon)$*-extractor if for all weak* k_i*-sources* X_i *over domains* $[N_i]$*, we have:*

$$\mathrm{SD}\big(\mathsf{Ext}(X_1, \ldots, X_t), \mathcal{U}_{[M]}\big) \leq \varepsilon \ ,$$

where ε *is usually defined as a function of* $k_1, \ldots, k_t$*. We call* Ext *an* s*-out-of-*t $(k_1, \ldots, k_t, \varepsilon)$*-extractor if* $\mathsf{Ext}(X_1, \ldots, X_t)$ *is* ε*-close to uniform even if only* s *sources fulfill the min-entropy condition.*

Below we define useful classes of distributions, the so-called (partially) dense distributions, resp. dense probability density functions. Intuitively, bit strings from a dense distribution are unpredictable not only as a whole but also in any of their substrings and any combination of those substrings.

Definition 2 (Dense distributions). *Let* μ *be a probability density function over* $[M]^N$*. Then*

- μ *is called* $(1 - \delta)$-dense *if for* $\mathsf{F} \sim \mu$*, it holds that for every subset* $I \subseteq [N]$ *we have* $\mathbf{H}_\infty(\mathsf{F}_I) \geq (1 - \delta) \cdot |I| \cdot \log M$*.*
- μ *is called* $(p, 1 - \delta)$-dense *if for* $\mathsf{F} \sim \mu$ *there exists a set* $I \subseteq [N]$ *of size* $|I| \leq p$ *such that* $\mathbf{H}_\infty(\mathsf{F}_I) = 0$*, while for every subset* $J \subseteq [N] \setminus I$ *we have* $\mathbf{H}_\infty(\mathsf{F}_J) \geq (1 - \delta) \cdot |J| \cdot \log M$*. That is,* μ *is fixed on at most* p *coordinates and* $(1 - \delta)$*-dense on the rest.*

We call a distribution dense, if the corresponding density function is dense.

3 Decomposition of High Min-Entropy Distributions

Any high min-entropy distribution can be written as a convex combination of distributions that are fixed on a number of points and dense on the rest (i.e., $(p, 1 - \delta)$-dense distributions for some p and $\delta > 0$).[3] The decomposition technique introduced by Göös et al. [14] has its origins in communication complexity

[3] A convex combination of distributions $\mu_1, \ldots, \mu_n$ is a distribution that can be written as $\alpha_1 \cdot \mu_1 + \ldots + \alpha_n \cdot \mu_n$, where $\alpha_1, \ldots, \alpha_n$ are non-negative real numbers that sum up to 1.

theory. We generalize this technique, with a terminology closer to that of Kothari et al. [18], in order to allow for adaptive leakage. The original lemma, also used by Coretti et al. [7], can be easily derived as a special case of our lemma. For this, one assumes that the starting distribution before the leakage was uniform, in other words $(0, 1)$-dense.

When proving results in the auxiliary-input random-oracle (AI-RO) model, Uhruh [27] observed that pre-computation (or leakage) on a random oracle can cause a significant decrease of its min-entropy only on restricted parts of its support (i.e., on p points), causing that part to become practically fixed, while the rest remains indistinguishable from random to a bounded-query distinguisher. This means that after fixing p coordinates of the random oracle, the rest can be lazily sampled from a uniform distribution. Coretti et al. [7] recently gave a different and tighter proof consisting of two main steps. First, the decomposition technique is used to show that the distribution of a random oracle given some leakage is *statistically* close to a $(p, 1-\delta)$-dense distribution. Second, they prove that no *bounded-query* algorithm can distinguish a $(p, 1-\delta)$-dense distribution from one that is fixed on the same p points and is otherwise uniform (a so-called p-bit-fixing distribution), as suggested by Unruh [27].

Since in the BRO model adaptive queries are allowed, a function queried to the backdoor oracle is able to "see" the entire random oracle, rather than a restricted part of it. Hence, when analyzing the distribution of a random oracle after adaptive leakage, it is crucial that we keep the distributions *statistically close*. In other words we use $(p, 1-\delta)$-dense distributions instead of p-bit-fixing.

In the k-BRO model, we are concerned with multiple queries to the backdoor oracles, i.e., continuous and adaptive leakage that can depend on previously leaked information about both hash functions. Intuitively, since the leakage function can be arbitrary, it can in particular depend on the previously leaked values. We still need to argue that the distribution obtained after leakage about a $(p_{\mathrm{prv}}, 1-\delta_{\mathrm{prv}})$ distribution, which is not necessarily uniform, is also close to a convex combination of $(p, 1-\delta)$ distributions. Naturally, we have $\delta \geq \delta_{\mathrm{prv}}$, since min-entropy decreases after new leakage, and $p \geq p_{\mathrm{prv}}$, since additional points are fixed. Looking ahead, in the indifferentiability proofs, this refined decomposition lemma allows us to simply fix a new portion p_{frsh} of the simulated hash function after each leakage (i.e., backdoor query) and not to worry about the rest, which still has high entropy and can be lazily sampled (from a dense distribution) upon receiving the next query.

Lemma 1 (Refined decomposition after leakage). *Let μ be a $(p_{\mathrm{prv}}, 1-\delta_{\mathrm{prv}})$-dense density function over $[M]^N$ for some $p_{\mathrm{prv}}, \delta_{\mathrm{prv}} \geq 0$. Let $f : [M]^N \to \{0,1\}^\ell$ be an arbitrary function and $z \in \{0,1\}^\ell$ be a bit string. Then for any $p_{\mathrm{frsh}}, \gamma > 0$, the density function conditioned on the leakage $\mu|_{f(\cdot)=z}$ is γ-close to a convex combination of finitely many $(p, 1-\delta)$-dense density functions for some p and δ such that*

$$p_{\mathrm{prv}} \leq p \leq p_{\mathrm{prv}} + p_{\mathrm{frsh}} \quad \textit{and} \quad \delta_{\mathrm{prv}} \leq \delta \leq \frac{\delta_{\mathrm{prv}} \cdot \log M \cdot (N - p_{\mathrm{prv}}) + \ell_z + \log \gamma^{-1}}{p_{\mathrm{frsh}} \cdot \log M},$$

where $\ell_z := \mathbf{H}_\infty(\mathsf{G}) - \mathbf{H}_\infty(\mathsf{F})$ *is the min-entropy deficiency of* $\mathsf{F} \sim \mu|_{f(\cdot)=z}$ *compared to* $\mathsf{G} \sim \mu$.

Proof. This refined decomposition lemma differs from the original lemma in that the starting density function μ is $(p_{\mathrm{prv}}, 1-\delta_{\mathrm{prv}})$-dense. As a first step, we modify the original decomposition algorithm from [14,18] so that it additionally gets the set of p_{prv} indices $I_{\mathrm{prv}} \subseteq [N]$ that are already fixed in μ from the start.

Our refined decomposition algorithm RefinedDecomp, given below, recursively decomposes the domain $[M]^N$, according to the density function after leakage $\mu_z := \mu|_{f(\cdot)=z}$, into $d+1$ partitions $D_1, \ldots, D_d, D_{\mathsf{err}} \subseteq [M]^N$ such that $\left(\bigcup_{i=1}^{d} D_i\right) \cup D_{\mathsf{err}} = [M]^N$, where err stands for erroneous. For all i with $1 \leq i \leq d$ the partition D_i defines a $(p, 1-\delta)$-dense density function $\mu_z|_{D_i}$.

Each recursive call on a domain D to RefinedDecomp (other than the call leading to D_{err}, which we will discuss shortly) returns a pair (D_i, I_i), where D_i represents a subset of $[M]^N$, where the images of all points in the set $I_i \subset [N]$ are fixed to the same values under all functions $\mathsf{H} \in D_i$. In other words, we have $\mathsf{H}_{I_i} = \alpha_i$ for some $\alpha_i \in [M]^{|I_i|}$. The algorithm finds such a pair (D_i, I_i) by considering the biggest set I_i (excluding those points fixed from the start, i.e., I_{prv}) such that the min-entropy of F_{I_i} (for $\mathsf{F} \sim \mu_z|_D$) is too small (as determined by the rate δ) and then finding some α_i which is a very likely value of F_{I_i}. Then I_i is returned with some D_i as the partition that contains all H with $\mathsf{H}_{I_i} = \alpha_i$. The next recursive call will exclude D_i from the considered domain.

Decomposition halts either if the probability of a sample falling into the current domain is smaller than γ (i.e., $\mu_z(D) \leq \gamma$) or the current distribution is already $(p_{\mathrm{prv}}, 1-\delta)$-dense. In both cases the algorithm returns the current domain D together with an empty set. In the former case the returned domain is marked as an erroneous domain $D_{\mathsf{err}} := D$, since it may not define a $(p, 1-\delta)$-dense distribution. Let us without loss of generality assume that μ_z is not $(p_{\mathrm{prv}} + p_{\mathrm{frsh}}, 1-\delta)$-dense, as otherwise the claim holds trivially.

The formal definition of the algorithm RefinedDecomp is given below. We initialize the desired density rate as $\delta := \frac{\delta_{\mathrm{prv}} \cdot \log M \cdot (N - p_{\mathrm{prv}}) + \ell_z + \log \gamma^{-1}}{p_{\mathrm{frsh}} \cdot \log M}$ before calling RefinedDecomp.

RefinedDecomp$[\mu_z, \delta, \gamma, I_{\mathrm{prv}}](D)$

if $\mu_z(D) \leq \gamma$ **then return** $(D_{\mathsf{err}} \leftarrow D, \emptyset)$
if $\mu_z|_D$ is $(|I_{\mathrm{prv}}|, 1-\delta)$-dense **then return** $(D, \emptyset)$
for $\mathsf{F} \sim \mu_z|_D$ let $I \subseteq [N]$ be a maximal set such that
$$\mathbf{H}_\infty(\mathsf{F}_I) < (1-\delta) \cdot |I| \cdot \log M \quad \text{and} \quad I \cap I_{\mathrm{prv}} = \emptyset.$$
let $\alpha \in [M]^{|I|}$ be such that $\Pr[\mathsf{F}_I = \alpha] > 2^{-(1-\delta)\cdot|I|\cdot\log M}$.
$D_\alpha \leftarrow D \cap \{\mathsf{F} \in [M]^N \mid \mathsf{F}_I = \alpha\}$
$D_{\neq\alpha} \leftarrow D \cap \{\mathsf{F} \in [M]^N \mid \mathsf{F}_I \neq \alpha\}$
return $((D_\alpha, I), \mathsf{RefinedDecomp}[\mu_z, \delta, \gamma, I_{\mathrm{prv}}](D_{\neq\alpha}))$

Now we turn our attention to proving that every partition D_i (other than D_{err}) returned by the above decomposition algorithm defines a density function $\mu_z|_{D_i}$ which is $(p, 1-\delta)$-dense.

Claim 1. *For all values of i with $1 \leq i \leq d$ it holds that the density function $\mu_z|_{D_i}$ is $(p, 1-\frac{\delta_{\mathrm{prv}} \cdot \log M \cdot (N-p_{\mathrm{prv}}) + \ell_z + \log \gamma^{-1}}{p_{\mathrm{frsh}} \cdot \log M})$-dense, where $p_{\mathrm{prv}} \leq p \leq p_{\mathrm{prv}} + p_{\mathrm{frsh}}$.*

Proof. Let $\delta := \frac{\delta_{\mathrm{prv}} \cdot \log M \cdot (N-p_{\mathrm{prv}}) + \ell_z + \log \gamma^{-1}}{p_{\mathrm{frsh}} \cdot \log M}$. Let I be the set of freshly fixed points in $\mu_z|_{D_i}$ and $\overline{I \cup I_{\mathrm{prv}}} := [N] \setminus (I \cup I_{\mathrm{prv}})$. Let $\alpha_\cup \in [N]^{|I \cup I_{\mathrm{prv}}|}$ be such that $\mathsf{F}_{I \cup I_{\mathrm{prv}}} = \alpha_\cup$ for $\mathsf{F} \sim \mu_z|_{D_i}$. We first argue for the $(1-\delta)$-density of $\mu_z|_{D_i}$ on values projected to $\overline{I \cup I_{\mathrm{prv}}}$ and afterwards bound the size of I.

1. Suppose $\mu_z|_{D_i}$ is not $(1-\delta)$-dense on $\overline{I \cup I_{\mathrm{prv}}}$. Then there exists a non-empty set which violates the density property. That is, there exists a non-empty set $J \subseteq \overline{I \cup I_{\mathrm{prv}}}$ and some $\beta \in [N]^{|J|}$ such that, with the probability taken over $\mathsf{F} \sim \mu_z|_{D_i}$, we have:

$$\Pr[\mathsf{F}_J = \beta] > 2^{-(1-\delta)\cdot|J|\cdot \log M} .$$

   Now the union of the three sets $I^* := I \cup I_{\mathrm{prv}} \cup J$ forms a new set such that for some $\beta^* \in [N]^{|I \cup I_{\mathrm{prv}} \cup J|}$ we have

$$\begin{aligned}
\Pr[\mathsf{F}_{I^*} = \beta^*] &= \Pr[\mathsf{F}_{I \cup I_{\mathrm{prv}}} = \alpha_\cup \wedge \mathsf{F}_J = \beta] \\
&= \Pr[\mathsf{F}_{I \cup I_{\mathrm{prv}}} = \alpha_\cup] \cdot \Pr[\mathsf{F}_J = \beta | \mathsf{F}_{I \cup I_{\mathrm{prv}}} = \alpha_\cup] \\
&> 2^{-(1-\delta)\cdot|I \cup I_{\mathrm{prv}}|\cdot \log M} \cdot 2^{-(1-\delta)\cdot|J|\cdot \log M} \\
&= 2^{-(1-\delta)\cdot|I \cup I_{\mathrm{prv}} \cup J|\cdot \log M} .
\end{aligned}$$

   Since J was assumed to be non-empty and disjoint from $I \cup I_{\mathrm{prv}}$ (and in particular with I), its existence violates the maximality of I. Therefore, $\mathsf{F}_{\overline{I \cup I_{\mathrm{prv}}}}$ is $(1-\delta)$ dense.

2. We now bound the size of I, given that $\delta = \frac{\delta_{\mathrm{prv}} \cdot \log M \cdot (N-p_{\mathrm{prv}}) + \ell_z + \log \gamma^{-1}}{p_{\mathrm{frsh}} \cdot \log M}$. Let $\mathsf{F} \sim \mu_z$ and $\mathsf{G} \sim \mu$. We have $\mathbf{H}_\infty(\mathsf{F}) = \mathbf{H}_\infty(\mathsf{G}) - \ell_z \geq (1-\delta_{\mathrm{prv}}) \cdot (N - p_{\mathrm{prv}}) . \log M - \ell_z$, where the inequality holds, since μ is $(1-\delta_{\mathrm{prv}})$-dense in $N - p_{\mathrm{prv}}$ rows. Let $\beta \in [M]^{|I|}$. Then we have:

$$\begin{aligned}
\Pr_{\mu_z|_{D_i}}[\mathsf{F}_I = \beta] &\leq \Pr_{\mu_z}[\mathsf{F}_I = \beta] / \mu_z(D_i) \\
&\leq \Pr_{\mu_z}[\mathsf{F}_I = \beta] / \gamma \\
&= \sum_{\beta' \in [M]^{N-|I|-|I_{\mathrm{prv}}|}} \Pr_{\mu_z}[\mathsf{F}_I = \beta \wedge \mathsf{F}_{[N]\setminus(I \cup I_{\mathrm{prv}})} = \beta'] / \gamma \\
&\leq 2^{(N-|I|-p_{\mathrm{prv}})\cdot \log M} \cdot 2^{-\mathbf{H}_\infty(\mathsf{F})} / \gamma \\
&\leq 2^{(N-|I|-p_{\mathrm{prv}})\cdot \log M} \cdot 2^{-((1-\delta_{\mathrm{prv}})\cdot(N-p_{\mathrm{prv}})\cdot \log M - \ell_z)} / \gamma \\
&= 2^{\delta_{\mathrm{prv}} \cdot N \cdot \log M - \delta_{\mathrm{prv}} \cdot p_{\mathrm{prv}} \cdot \log M - |I| \cdot \log M + \ell_z} / \gamma
\end{aligned}$$

$$= 2^{\delta_{\mathrm{prv}} \cdot \log M \cdot (N - p_{\mathrm{prv}}) - |I| \cdot \log M + \ell_z + \log \gamma^{-1}} \ .$$

Since by definition of the decomposition algorithm, there exists an $\alpha \in [M]^I$ such that $\Pr_{\mu_z|_{D_i}}[\mathsf{F}_I = \alpha] > 2^{-(1-\delta)\cdot|I|\cdot\log M}$, we obtain

$$|I| \leq \frac{\delta_{\mathrm{prv}} \cdot \log M \cdot (N - p_{\mathrm{prv}}) + \ell_z + \log \gamma^{-1}}{\delta \cdot \log M} \ .$$

Substituting δ by $\frac{\delta_{\mathrm{prv}} \cdot \log M \cdot (N - p_{\mathrm{prv}}) + \ell_z + \log \gamma^{-1}}{p_{\mathrm{frsh}} \cdot \log M}$, we obtain $|I| \leq p_{\mathrm{frsh}}$ and therefore, for the total number of fixed points $p := |I \cup I_{\mathrm{prv}}|$ we get $p_{\mathrm{prv}} \leq p \leq p_{\mathrm{prv}} + p_{\mathrm{frsh}}$, as stated in the claim.

□

Therefore, μ_z can be written as a convex combination of $\mu_z|_{D_1}, \ldots, \mu_z|_{D_d}$ and $\mu_z|_{D_{\mathsf{err}}}$, i.e., $\mu_z = \sum_{i=1}^{d} \mu_z(D_i) \cdot \mu_z|_{D_i} + \mu_z(D_{\mathsf{err}}) \cdot \mu_z|_{D_{\mathsf{err}}}$. Since $\mu_z(D_{\mathsf{err}}) \leq \gamma$ when the algorithm RefinedDecomp terminates, the distribution μ_z is γ-close to a convex combination of $(p, 1-\delta)$ distributions. □

A special case of the above lemma for a uniform (i.e., $(0,1)$-dense) starting distribution μ, where $p_{\mathrm{prv}} = 0$ and $\delta_{\mathrm{prv}} = 0$, implies the bound $\delta \leq (\ell_z + \log \gamma^{-1})/(p_{\mathrm{frsh}} \cdot \log M)$ used by Coretti et al. [7].

Remark. Note that the coefficient of δ_{prv} in the right hand side of the inequality established in the lemma is of the order $\mathcal{O}(N/p_{\mathrm{frsh}})$. Looking ahead (see discussions on parameter estimation) this results in an increase in the number of points that the simulator needs to set. Thus any improvement in the bound established in this lemma would translate to tolerating a higher level of adaptivity and/or obtaining an improved bound.

Below we show that the expected min-entropy deficiency after leaking ℓ bits of information can be upper-bounded by ℓ bits.

Lemma 2. *Let F be a random variable over $[M]^N$ and $f : [M]^N \to \{0,1\}^\ell$ be an arbitrary function. Let $\ell_z := \mathbf{H}_\infty(\mathsf{F}) - \mathbf{H}_\infty(\mathsf{F}|f(\mathsf{F}) = z)$ be the min-entropy deficiency of $\mathsf{F}|f(\mathsf{F})=z$. Then, we have $\mathbb{E}_{z \in f(\mathsf{supp}(\mathsf{F}))}[\ell_z] \leq \ell$.*

Proof. Recall that $\widetilde{\mathbf{H}}_\infty(A|B) := -\log\left(\mathbb{E}_b\left[\max_a \Pr[A = a|B = b]\right]\right)$ defines the average min-entropy of A, given B.

$$\begin{aligned}
\mathbb{E}_{z \in f(\mathsf{supp}(\mathsf{F}))}[\ell_z] &= \mathbf{H}_\infty(\mathsf{F}) - \mathbb{E}_{z \in f(\mathsf{supp}(\mathsf{F}))}[\mathbf{H}_\infty(\mathsf{F}|f(\mathsf{F}) = z)] \\
&\leq \mathbf{H}_\infty(\mathsf{F}) - \widetilde{\mathbf{H}}_\infty(\mathsf{F}|f(\mathsf{F}) = z) \\
&\leq \mathbf{H}_\infty(\mathsf{F}) - \mathbf{H}_\infty(\mathsf{F}) + \log|f(\mathsf{supp}(\mathsf{F}))| \leq \ell \ ,
\end{aligned}$$

where for deriving the second line we have used Jensen's inequality and for the third line we have used [11, Lemma 2.2.b].[4] □

[4] The lemma is as follows. Let A, B be random variables. Then we have $\widetilde{\mathbf{H}}_\infty(A|B) \geq \mathbf{H}_\infty(A, B) - n \geq \mathbf{H}_\infty(A) - n$, where B has at most 2^n possible values.

4 The xor Combiner

In this section, we study the indifferentiability of the xor combiner $\mathsf{C}_{\oplus}^{\mathsf{H}_1,\mathsf{H}_2}(x) := \mathsf{H}_1(x) \oplus \mathsf{H}_2(x)$ in the 2-BRO model from a random oracle RO. We show indifferentiability against adversaries that switch between the two backdoor oracles BD_1 and BD_2 only a logarithmic number of times, while arbitrarily interleaving queries to the underlying BROs H_1 and H_2, as well as to the random oracle RO.

To prove indifferentiability we need to show that there exists a simulator $\mathsf{Sim} := (\mathsf{SimH}_1^{\mathsf{RO}}, \mathsf{SimH}_2^{\mathsf{RO}}, \mathsf{SimBD}_1^{\mathsf{RO}}, \mathsf{SimBD}_2^{\mathsf{RO}})$ such that no distinguisher placing a "reasonable" number of queries can distinguish

$$(\mathsf{C}_{\oplus}^{\mathsf{H}_1,\mathsf{H}_2}, \mathsf{H}_1, \mathsf{H}_2, \mathrm{BD}_1, \mathrm{BD}_2) \quad \text{and} \quad (\mathsf{RO}, \mathsf{SimH}_1^{\mathsf{RO}}, \mathsf{SimH}_2^{\mathsf{RO}}, \mathsf{SimBD}_1^{\mathsf{RO}}, \mathsf{SimBD}_2^{\mathsf{RO}}) .$$

Such a simulator is described in Figure 1. Simulating the evaluation queries to H_1 and H_2 is straightforward. In simulating the backdoor queries, we take advantage of the decomposition technique (discussed in Section 3) for transforming high min-entropy distributions into distributions that have a number of fixed points and are dense otherwise. The backdoor simulator SimBD_1 (resp. SimBD_2) computes the queried function f on the truth table of H_1 (resp. H_2), where H_1 and H_2 are initialized by picking two functions uniformly at random. For the sake of simplicity, we consider an adversary that makes Q consecutive queries, ignoring evaluation and RO-queries in between, to one backdoor oracle before moving to the other. After the i-th sequence of Q queries to one of the backdoor oracles, the leaked backdoor information is translated into fixing p_i rows of the hash function such that the rest is dense and the resulting distribution is statistically close to the true one. In other words, the distribution conditioned on the leakage is γ-close (for some $\gamma > 0$) to a convex combination of $(p, 1-\delta)$-dense distributions obtained after decomposition.

Regarding the density rates δ_i's, we use odd values of i for the distributions obtained after backdoor queries on H_1 and even values of i for distributions of H_2. Note that is crucial for the statistical distance of these two distributions on the entire table to remain small, since the distinguisher can adaptively query a backdoor oracle which sees and can depend on the *entire* hash function table (as opposed to a limited number of rows).

Finding a distribution, which is partly fixed and partly dense, is performed by the $\mathsf{FixRows}$ algorithm from 1. On input of a distribution μ_z, integer $p \in \mathbb{N}$, and a set $I_{\mathrm{prv}} \in [N]$, the algorithm $\mathsf{FixRows}$ returns a new distribution which is fixed on points in a set I of size at most $p + |I_{\mathrm{prv}}|$ and is for some δ, $(1-\delta)$-dense on the rest, together with a set of assignments A for elements in I according to the output distribution. The $\mathsf{FixRows}$ algorithm internally calls the refined decomposition algorithm, whose existence is guaranteed by Lemma 1 and its output distribution is one of the distributions in the convex combination returned by $\mathsf{RefinedDecomp}$.

Upon fixing rows of one simulated BRO, the same rows in the other simulated BRO have to be fixed in a way that consistency with RO is assured. More precisely, for any x if $\mathsf{H}_1(x)$ is fixed, the simulator SimBD_1 will immediately

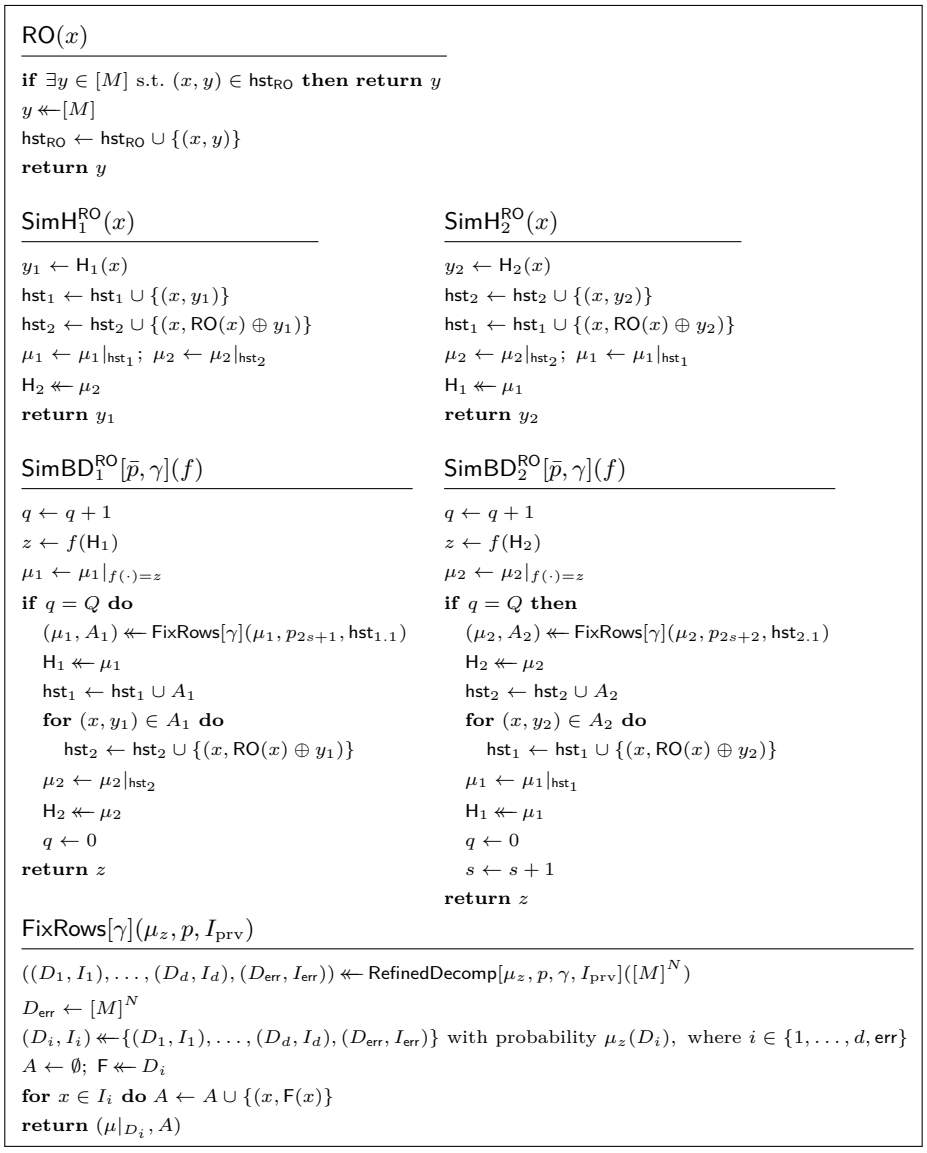


Fig. 1: Indifferentiability simulator for the xor combiner. We assume initial values $\mathsf{hst}_1 = \mathsf{hst}_2 = \mathsf{hst}_{\mathsf{RO}} := \emptyset$, $\mu_1 = \mu_2 := \mathcal{U}_{[M]^N}$, $\mathsf{H}_1, \mathsf{H}_2 \twoheadleftarrow \mathcal{U}_{[M]^N}$, $q := 0$, and $s := 0$.

set $\mathsf{H}_2(x) := \mathsf{RO}(x) \oplus \mathsf{H}_1(x)$ (and, analogously, so does SimBD_2). The simulator specifies the number of points that it can afford to fix (since every such query requires a call to RO) and the statistical distance that it wants. Such a strategy to assure consistency with RO is also followed by evaluation simulators SimH_1 and SimH_2, where only one coordinate of each BRO is fixed.

Note that the simulator SimBD_1 programs values of H_2, which were supposed to be dense (after a first SimBD_2 query), to values that are uniform instead. Hence, we need to argue later that the statistical distance between a uniform and a dense distribution is small for the number of points that are being treated this way. This is formalized in Claim 2, below. Looking ahead, the need to keep the advantage of the differentiator small is the reason why the simulator adapts the number of fixed points with a differentiator's switch to the other backdoor oracle. Finally, via a hybrid argument we can upper bound the total number of random oracle queries by the simulator and the advantage of the differentiator.

Claim 2. *Let $\mathcal{U}$ be the uniform distribution and $\mathcal{V}$ be a $(1-\delta)$-dense distribution, both over the domain $[M]^t$. Then we have $\mathrm{SD}(\mathcal{U},\mathcal{V}) \leq t \cdot \delta \cdot \log M$.*

Proof. This proof follows that of [7, Claim 3]. Let V_+ be the set of all values $z \in [M]^t$ for which $\Pr[\mathcal{V}=z] > 0$ holds. We can write the statistical distance between $\mathcal{U}$ and $\mathcal{V}$ as:

$$\begin{aligned}\mathrm{SD}(\mathcal{U},\mathcal{V}) &= \sum_{z\in[M]^t} \max\left\{0, \Pr[\mathcal{V}=z] - \Pr[\mathcal{U}=z]\right\} \\ &= \sum_{z\in V_+} \max\left\{0, \Pr[\mathcal{V}=z] - \Pr[\mathcal{U}=z]\right\} \\ &= \sum_{z\in V_+} \Pr[\mathcal{V}=z] \cdot \max\left\{0, 1 - \frac{\Pr[\mathcal{U}=z]}{\Pr[\mathcal{V}=z]}\right\} .\end{aligned}$$

Now, observe that for any value $z \in [M]^t$, we have $\Pr[\mathcal{V}=z] \leq M^{-(1-\delta)\cdot t}$ and $\Pr[\mathcal{U}=z] = M^{-t}$. Hence we have:

$$\mathrm{SD}(\mathcal{U},\mathcal{V}) \leq 1 - M^{-\delta\cdot t} \leq t \cdot \delta \cdot \log M \ ,$$

where the last inequality uses the fact that for all $x \geq 0$, it holds that $2^{-x} \geq 1-x$ (and hence, $x \geq 1-2^{-x}$). □

The following theorem states our indifferentiability result for xor.

Theorem 1 (Indifferentiability of xor in 2-BRO with bounded adaptivity). *Consider the xor combiner $\mathsf{C}^{\mathsf{H}_1,\mathsf{H}_2}_{\oplus}(x) := \mathsf{H}_1(x) \oplus \mathsf{H}_2(x)$ in the 2-BRO model with backdoored hash functions $\mathsf{H}_1, \mathsf{H}_2 \in [M]^N$. It holds that for any $\bar{p} := (p_1, \ldots, p_{c+1}) \in \mathbb{N}^{c+1}$, $0 < \gamma < 1$, and an integer $c \geq 0$, there exists a simulator $\mathsf{Sim}[\bar{p},\gamma] := (\mathsf{SimH}^{\mathsf{RO}}_1, \mathsf{SimH}^{\mathsf{RO}}_2, \mathsf{SimBD}^{\mathsf{RO}}_1[\bar{p},\gamma], \mathsf{SimBD}^{\mathsf{RO}}_2[\bar{p},\gamma])$ such that for any differentiator $\mathcal{D}$ that always makes Q queries to a backdoor oracle (starting from B_{D_1} and always receiving an ℓ-bit response) before switching to the other, with a total number of c switches, while being allowed to arbitrarily interleave up to q_{H} primitive queries as well as q_{C} construction queries, we have*

$$\begin{aligned}\mathsf{Adv}^{\mathrm{indiff}}_{\mathsf{C}^{\mathsf{H}_1,\mathsf{H}_2}_{\oplus},\mathsf{Sim}[\bar{p},\gamma]}(\mathcal{D}) &\leq (c+1)\cdot\gamma \\ &\quad + \log M \cdot \Big(\sum_{i=1}^{c} p_i \cdot \delta_{i-1} + q_{\mathsf{H}} \cdot \delta_{c+1} + q_{\mathsf{C}} \cdot (\delta_c + \delta_{c+1})\Big) \ ,\end{aligned}$$

where $\delta_{-1} := \delta_0 := 0$ *and the density rate after the* i*-th sequence of* Q*-many backdoor queries is* $\delta_i := \big(\delta_{i-2} \cdot (N - \sum_{j=1}^{i-2} p_j) \cdot \log M + Q \cdot \ell + \log \gamma^{-1}\big) / \big(p_i \cdot \log M\big)$*. The simulator places at most* $q_{\mathsf{Sim}} \leq q_{\mathsf{H}} + \sum_{i=1}^{c+1} p_i$ *queries to the random oracle* RO*.*

Proof. We prove indifferentiability by (1) defining a simulator, (2) upper bounding the advantage of any differentiator in distinguishing the real and the simulated worlds, and (3) upper bounding the number of queries that the simulator makes to the random oracle.

Simulator. All four sub-algorithms of the simulator are described in Figure 1. They share state, in particular, variables to keep track of the fixed history and the current distribution of the hash functions. Two sets $\mathsf{hst}_1, \mathsf{hst}_2$ are used to keep track of the fixed coordinates of the simulated hash functions H_1 and H_2, respectively. The density functions, from which the simulated backdoored hash functions will be sampled, are denoted by μ_1 and μ_2. Furthermore, the simulator uses a counter s to recognize switches from one backdoor oracle to the other in order to use the appropriate number of points to fix from the list $\bar{p}$. It also maintains a counter q for counting the number of consecutive queries to a backdoor oracle in order to decompose, i.e., substitute the current distribution with a partially fixed and partially dense distribution, only when necessary which is the case after each set of Q backdoor queries. We assume the initial values $\mu_1 = \mu_2 := \mathcal{U}_{[M]^N}$, $\mathsf{H}_1, \mathsf{H}_2 \twoheadleftarrow \mathcal{U}_{[M]^N}$, $\mathsf{hst}_1 = \mathsf{hst}_2 = \mathsf{hst}_{\mathsf{RO}} := \emptyset$, $q := 0$, and $s := 0$.

Security analysis. Here we analyze the indifferentiability of the xor combiner using a sequence of eight games $\mathsf{Game}_0, \ldots, \mathsf{Game}_7$, where Game_0 and Game_7 are the real and ideal indifferentiability games, respectively. In the following we use the shorthand notation $\Pr[\mathcal{D}^{\mathsf{Game}_i}] := \Pr[\mathcal{D}^{\mathsf{Game}_i} = 1]$, where $\mathcal{D}^{\mathsf{Game}_i}$ indicates the interaction of an adversary $\mathcal{D}$ with a game Game_i. We define the intermediate games Game_1 through Game_6 by gradually modifying the oracles and highlighting the changes in each step. Unchanged oracles are omitted in games and correspond to those from their direct predecessor. We bound the advantage of differentiators in distinguishing every two consecutive games.

$\underline{\mathsf{Game}_0 : \mathsf{C}^{\mathsf{H}_1,\mathsf{H}_2}_{\oplus}(x)}$

$y_1 \leftarrow \mathsf{H}_1(x);\ y_2 \leftarrow \mathsf{H}_2(x)$
$y \leftarrow y_1 \oplus y_2$
return y

$\underline{\mathsf{Game}_0 : \mathsf{H}_1(x)}$	$\underline{\mathsf{Game}_0 : \mathsf{H}_2(x)}$	$\underline{\mathsf{Game}_0 : \mathrm{BD}_1(f)}$	$\underline{\mathsf{Game}_0 : \mathrm{BD}_2(f)}$
$y_1 \leftarrow \mathsf{H}_1(x)$	$y_2 \leftarrow \mathsf{H}_2(x)$	$z \leftarrow f(\mathsf{H}_1)$	$z \leftarrow f(\mathsf{H}_2)$
return y_1	**return** y_2	**return** z	**return** z

Game_1. We next update the distributions of hash functions based on past evaluation queries, backdoor queries, and the history of coordinates that are fixed

through construction queries. The distributions μ_i are conditioned on these updates, but are never actually used (i.e., sampled from) in the game. Hence it is easy to see that these two games are identical, i.e., $\mathrm{SD}(\mathsf{Game}_0, \mathsf{Game}_1) = 0$.

$\mathsf{Game}_1 : \mathsf{C}_{\oplus}^{\mathsf{H}_1,\mathsf{H}_2}(x)$

$y_1 \leftarrow \mathsf{H}_1(x);\ y_2 \leftarrow \mathsf{H}_2(x)$
$\mathsf{hst}_1 \leftarrow \mathsf{hst}_1 \cup \{(x, y_1)\};\ \mathsf{hst}_2 \leftarrow \mathsf{hst}_2 \cup \{(x, y_2)\}$
$\mu_1 \leftarrow \mu_1|_{\mathsf{hst}_1};\ \mu_2 \leftarrow \mu_2|_{\mathsf{hst}_2}$
$y \leftarrow y_1 \oplus y_2$
return y

$\mathsf{Game}_1 : \mathsf{H}_1(x)$

$y_1 \leftarrow \mathsf{H}_1(x)$
$\mathsf{hst}_1 \leftarrow \mathsf{hst}_1 \cup \{(x, y_1)\}$
$\mu_1 \leftarrow \mu_1|_{\mathsf{hst}_1}$
return y_1

$\mathsf{Game}_1 : \mathsf{H}_2(x)$

$y_2 \leftarrow \mathsf{H}_2(x)$
$\mathsf{hst}_2 \leftarrow \mathsf{hst}_2 \cup \{(x, y_2)\}$
$\mu_2 \leftarrow \mu_2|_{\mathsf{hst}_2}$
return y_2

$\mathsf{Game}_1 : \mathrm{BD}_1(f)$

$z \leftarrow f(\mathsf{H}_1)$
$\mu_1 \leftarrow \mu_1|_{f(\cdot)=z}$
return z

$\mathsf{Game}_1 : \mathrm{BD}_2(f)$

$z \leftarrow f(\mathsf{H}_2)$
$\mu_2 \leftarrow \mu_2|_{f(\cdot)=z}$
return z

Game_2. Here, after each sequence of Q queries to a backdoor oracle, i.e., right before a switch, a $(p, 1-\delta)$-dense distribution μ_i' is obtained using the algorithm $\mathsf{FixRows}$ by decomposing the distribution of the corresponding hash function after responding to the last query (i.e., $\mu_i|_{f(\cdot)=z}$). However, since the new distributions μ_i' are never actually used elsewhere, Game_2 remains identical to Game_1, i.e., $\mathrm{SD}(\mathsf{Game}_1, \mathsf{Game}_2) = 0$.

$\mathsf{Game}_2 : \mathrm{BD}_1(f)$

$q \leftarrow q + 1$
$z \leftarrow f(\mathsf{H}_1);\ \mu_1 \leftarrow \mu_1|_{f(\cdot)=z}$
if $q = Q$ **then**
  $(\mu_1', A_1) \twoheadleftarrow \mathsf{FixRows}[\gamma](\mu_1, p_{2s+1}, \mathsf{hst}_{1.1})$
  $q \leftarrow 0$
return z

$\mathsf{Game}_2 : \mathrm{BD}_2(f)$

$q \leftarrow q + 1$
$z \leftarrow f(\mathsf{H}_2);\ \mu_2 \leftarrow \mu_2|_{f(\cdot)=z}$
if $q = Q$ **then**
  $(\mu_2', A_2) \twoheadleftarrow \mathsf{FixRows}[\gamma](\mu_2, p_{2s+2}, \mathsf{hst}_{2.1})$
  $q \leftarrow 0$
  $s \leftarrow s + 1$
return z

Game_3. In this game, evaluation queries on a value x, fix the image of both functions, i.e., to $\mathsf{H}_1(x)$ and $\mathsf{H}_2(x)$. Similarly, in backdoor simulation the rows in the assignments A_1 (resp. A_2) are fixed for the other hash function H_2 (resp. H_1) according to its current distribution. In both games, the oracles' responses are at all times consistent with their past responses (and the construction) and we still do not sample from the updated distributions. Hence, it does not matter, if more or less of the hash function tables are fixed in each query and therefore the two games are identical, i.e., $\mathrm{SD}(\mathsf{Game}_2, \mathsf{Game}_3) = 0$.

$\mathsf{Game}_3 : \mathsf{H}_1(x)$

$y_1 \leftarrow \mathsf{H}_1(x)$
$\mathsf{hst}_1 \leftarrow \mathsf{hst}_1 \cup \{(x, y_1)\}$
$\mathsf{hst}_2 \leftarrow \mathsf{hst}_2 \cup \{(x, \mathsf{H}_2(x)\}$
$\mu_1 \leftarrow \mu_1|_{\mathsf{hst}_1}$; $\mu_2 \leftarrow \mu_2|_{\mathsf{hst}_2}$
return y_1

$\mathsf{Game}_3 : \mathsf{H}_2(x)$

$y_2 \leftarrow \mathsf{H}_2(x)$
$\mathsf{hst}_2 \leftarrow \mathsf{hst}_2 \cup \{(x, y_2)\}$
$\mathsf{hst}_1 \leftarrow \mathsf{hst}_1 \cup \{(x, \mathsf{H}_1(x))\}$
$\mu_2 \leftarrow \mu_2|_{\mathsf{hst}_2}$; $\mu_1 \leftarrow \mu_1|_{\mathsf{hst}_1}$
return y_2

$\mathsf{Game}_3 : \mathrm{BD}_1(f)$

$q \leftarrow q + 1$
$z \leftarrow f(\mathsf{H}_1)$; $\mu_1 \leftarrow \mu_1|_{f(\cdot)=z}$
if $q = Q$ **then**
  $(\mu_1', A_1) \twoheadleftarrow \mathsf{FixRows}[\gamma](\mu_1, p_{2s+1}, \mathsf{hst}_{1.1})$
  for $x \in A_{1.1}$ **do**
    $\mathsf{hst}_2 \leftarrow \mathsf{hst}_2 \cup \{(x, \mathsf{H}_2(x))\}$
  $\mu_2 \leftarrow \mu_2|_{\mathsf{hst}_2}$
  $q \leftarrow 0$
return z

$\mathsf{Game}_3 : \mathrm{BD}_2(f)$

$q \leftarrow q + 1$
$z \leftarrow f(\mathsf{H}_2)$; $\mu_2 \leftarrow \mu_2|_{f(\cdot)=z}$
if $q = Q$ **then**
  $(\mu_2', A_2) \twoheadleftarrow \mathsf{FixRows}[\gamma](\mu_2, p_{2s+2}, \mathsf{hst}_{2.1})$
  for $x \in A_{2.1}$ **do**
    $\mathsf{hst}_1 \leftarrow \mathsf{hst}_1 \cup \{(x, \mathsf{H}_1(x))\}$
  $\mu_1 \leftarrow \mu_1|_{\mathsf{hst}_1}$
  $q \leftarrow 0$
  $s \leftarrow s + 1$
return z

Game_4. In this game the distributions obtained by decomposition actually replace the distributions conditioned on leakage. Hence, the histories are also updated and a new hash function H_i is later sampled for potential usage in the construction. According to Lemma 1, there is a convex combination of $(p, 1-\delta)$-dense distributions which is γ-close to the real distribution, one of such distributions being the one returned by $\mathsf{FixRows}$. Hence, the distinguishing advantage can increase by γ for every Q sequence of backdoor queries. I.e., $\left|\Pr[\mathcal{D}^{\mathsf{Game}_3}] - \Pr[\mathcal{D}^{\mathsf{Game}_4}]\right| \leq (c+1) \cdot \gamma$.

$\mathsf{Game}_4 : \mathrm{BD}_1(f)$

$q \leftarrow q + 1$
$z \leftarrow f(\mathsf{H}_1)$; $\mu_1 \leftarrow \mu_1|_{f(\cdot)=z}$
if $q = Q$ **then**
  $(\mu_1, A_1) \twoheadleftarrow \mathsf{FixRows}[\gamma](\mu_1, p_{2s+1}, \mathsf{hst}_{1.1})$
  $\mathsf{hst}_1 \leftarrow \mathsf{hst}_1 \cup A_1$
  $\mathsf{H}_1 \twoheadleftarrow \mu_1$
  for $x \in A_{1.1}$ **do**
    $\mathsf{hst}_2 \leftarrow \mathsf{hst}_2 \cup \{(x, \mathsf{H}_2(x))\}$
  $\mu_2 \leftarrow \mu_2|_{\mathsf{hst}_2}$
  $q \leftarrow 0$
return z

$\mathsf{Game}_4 : \mathrm{BD}_2(f)$

$q \leftarrow q + 1$
$z \leftarrow f(\mathsf{H}_2)$; $\mu_2 \leftarrow \mu_2|_{f(\cdot)=z}$
if $q = Q$ **then**
  $(\mu_2, A_2) \twoheadleftarrow \mathsf{FixRows}[\gamma](\mu_2, p_{2s+2}, \mathsf{hst}_{2.1})$
  $\mathsf{hst}_2 \leftarrow \mathsf{hst}_2 \cup A_2$
  $\mathsf{H}_2 \twoheadleftarrow \mu_2$
  for $x \in A_{2.1}$ **do**
    $\mathsf{hst}_1 \leftarrow \mathsf{hst}_1 \cup \{(x, \mathsf{H}_1(x))\}$
  $\mu_1 \leftarrow \mu_1|_{\mathsf{hst}_1}$
  $q \leftarrow 0$
  $s \leftarrow s + 1$
return z

Game_5. This game behaves exactly as Game_4 except when fixing the same rows for the distribution of the other BRO. It fixes those points by calling $\mathsf{C}_\oplus$ (rather than directly) and then redundantly updates the history with e.g.,

some $(x, \mathsf{H}_1(x) \oplus \mathsf{C}_\oplus(x))$ and samples a new BRO from the updated distribution. However, since the construction $\mathsf{C}_\oplus$ itself calls the BROs, Game_5 is only taking a detour and the two games are perfectly indistinguishable. Hence $\mathrm{SD}(\mathsf{Game}_4, \mathsf{Game}_5) = 0$.

$\mathsf{Game}_5 : \mathsf{H}_1(x)$

$y_1 \leftarrow \mathsf{H}_1(x)$
$\mathsf{hst}_1 \leftarrow \mathsf{hst}_1 \cup \{(x, y_1)\}$
$\mathsf{hst}_2 \leftarrow \mathsf{hst}_2 \cup \{(x, \mathsf{C}_\oplus(x) \oplus y_1\}$
$\mu_1 \leftarrow \mu_1|_{\mathsf{hst}_1};\ \mu_2 \leftarrow \mu_2|_{\mathsf{hst}_2}$
$\mathsf{H}_2 \twoheadleftarrow \mu_2$
return y_1

$\mathsf{Game}_5 : \mathsf{H}_2(x)$

$y_2 \leftarrow \mathsf{H}_2(x)$
$\mathsf{hst}_2 \leftarrow \mathsf{hst}_2 \cup \{(x, y_2)\}$
$\mathsf{hst}_1 \leftarrow \mathsf{hst}_1 \cup \{(x, \mathsf{C}_\oplus(x) \oplus y_2)\}$
$\mu_2 \leftarrow \mu_2|_{\mathsf{hst}_2};\ \mu_1 \leftarrow \mu_1|_{\mathsf{hst}_1}$
$\mathsf{H}_1 \twoheadleftarrow \mu_1$
return y_2

$\mathsf{Game}_5 : \mathrm{BD}_1(f)$

$q \leftarrow q + 1$
$z \leftarrow f(\mathsf{H}_1);\ \mu_1 \leftarrow \mu_1|_{f(\cdot)=z}$
if $q = Q$ **then**
  $(\mu_1, A_1) \twoheadleftarrow \mathsf{FixRows}[\gamma](\mu_1, p_{2s+1}, \mathsf{hst}_{1.1})$
  $\mathsf{hst}_1 \leftarrow \mathsf{hst}_1 \cup A_1$
  $\mathsf{H}_1 \twoheadleftarrow \mu_1$
  for $(x, y_1) \in A_1$ **do**
    $\mathsf{hst}_2 \leftarrow \mathsf{hst}_2 \cup \{(x, \mathsf{C}_\oplus(x) \oplus y_1)\}$
  $\mu_2 \leftarrow \mu_2|_{\mathsf{hst}_2}$
  $\mathsf{H}_2 \twoheadleftarrow \mu_2$
  $q \leftarrow 0$
return z

$\mathsf{Game}_5 : \mathrm{BD}_2(f)$

$q \leftarrow q + 1$
$z \leftarrow f(\mathsf{H}_2);\ \mu_2 \leftarrow \mu_2|_{f(\cdot)=z}$
if $q = Q$ **then**
  $(\mu_2, A_2) \twoheadleftarrow \mathsf{FixRows}[\gamma](\mu_2, p_{2s+2}, \mathsf{hst}_{2.1})$
  $\mathsf{hst}_2 \leftarrow \mathsf{hst}_2 \cup A_2$
  $\mathsf{H}_2 \twoheadleftarrow \mu_2$
  for $(x, y_2) \in A_2$ **do**
    $\mathsf{hst}_1 \leftarrow \mathsf{hst}_1 \cup \{(x, \mathsf{C}_\oplus(x) \oplus y_2)\}$
  $\mu_1 \leftarrow \mu_1|_{\mathsf{hst}_1}$
  $\mathsf{H}_1 \twoheadleftarrow \mu_1$
  $q \leftarrow 0$
  $s \leftarrow s + 1$
return z

Game_6. We now modify $\mathsf{C}_\oplus$ to start to resemble a lazily sampled random oracle. In the new construction oracle, a query is stored together with its image in the history $\mathsf{hst}_{\mathsf{RO}}$. In case a query is repeated, its stored image is simply returned. Otherwise, there are three cases to consider: the corresponding row to the current query x is fixed in both hash functions, in one of them, or in neither one. In the first case the output of the construction is computed by xoring the individual images stored in hst_1 and hst_2. In the second case, a uniformly random value is chosen (and later stored in $\mathsf{hst}_{\mathsf{RO}}$). In the final case, Game_6 behaves exactly as Game_5. So, the distinguishing advantage is bounded by distinguishing uniform points (set to uniform when xoring with the returned uniform value of $\mathsf{C}_\oplus$) from dense points. In fact, according to Claim 2, for each evaluation query it adds at most $\delta_{c+1} \cdot \log M$, since δ_i's are increasing. Further, for all points that are fixed upon a backdoor query this adds $p_i \cdot \delta_{i-1} \cdot \log M$, except for the last one, since there will be no backdoor query after that which can see the entire p_{c+1} points.

$$\left| \Pr[\mathcal{D}^{\mathsf{Game}_5}] - \Pr[\mathcal{D}^{\mathsf{Game}_6}] \right| \leq \log M \cdot \Big(\sum_{i=1}^{c} p_i \cdot \delta_{i-1} + q_{\mathsf{H}} \cdot \delta_{c+1} \Big)$$

$\mathsf{Game}_6 : \mathsf{C}_{\oplus}^{\mathsf{H}_1,\mathsf{H}_2}(x)$

if $\exists y \in [M]$ s.t. $(x, y) \in \mathsf{hst}_{\mathsf{RO}}$ **then return** y
if $\exists y_1, y_2 \in [M]$ s.t. $(x, y_1) \in \mathsf{hst}_1 \wedge (x, y_2) \in \mathsf{hst}_2$ **then return** $y_1 \oplus y_2$
if $\exists y' \in [M]$ s.t. $(x, y') \in \mathsf{hst}_1 \vee (x, y') \in \mathsf{hst}_2$ **then**
  $y \twoheadleftarrow [M]$
else
  $y_1 \leftarrow \mathsf{H}_1(x)$; $y_2 \leftarrow \mathsf{H}_2(x)$
  $\mathsf{hst}_1 \leftarrow \mathsf{hst}_1 \cup \{(x, y_1)\}$; $\mathsf{hst}_2 \leftarrow \mathsf{hst}_2 \cup \{(x, y_2)\}$
  $\mu_1 \leftarrow \mu_1|_{\mathsf{hst}_1}$; $\mu_2 \leftarrow \mu_2|_{\mathsf{hst}_2}$
  $y \leftarrow y_1 \oplus y_2$
$\mathsf{hst}_{\mathsf{RO}} \leftarrow \mathsf{hst}_{\mathsf{RO}} \cup \{(x, y)\}$
return y

Game_7. The construction oracle in this game differs from Game_6 in that it never evaluates the individual hash functions anymore. Here, we can safely remove the second case distinction, where x is in both hst_1 and hst_2, since this case is covered by the first case where x has been queried to the construction itself. It remains to bound the distinguisher's advantage in distinguishing the two games while making queries x to the construction that are prior to the query fixed for neither hash function.

Claim. Let X and Y be two independent $(1-\delta)$ and $(1-\delta')$-dense distributions over a domain $[M]^N$. Then the xor distribution $X \oplus Y$ is $(1-(\delta+\delta'))$-dense over the same domain $[M]^N$.

Proof. Let $I \subseteq [N]$ and $z \in [M]^{|I|}$ be arbitrary. Then we have:

$$\begin{aligned}\Pr[X_I \oplus Y_I = z] &= \sum_x \Pr[X_I = x \wedge Y_I = x \oplus z] = \sum_x \Pr[X_I = x] \cdot \Pr[Y_I = x \oplus z] \\ &\leq 2^{|I|\cdot\log M} \cdot 2^{-(1-\delta)\cdot|I|\cdot\log M} \cdot 2^{-(1-\delta')\cdot|I|\cdot\log M} \\ &= 2^{-(1-(\delta+\delta'))\cdot|I|\cdot\log M} \ .\end{aligned}$$

□

We can now bound the distinguisher's advantage by computing the distance between the sum of two dense distributions from uniform, given that only q_{C} queries to $\mathsf{C}_{\oplus}$ are allowed. Below, in the second line, we use the fact that according to Lemma 1, δ's should increase.

$$\left|\Pr[\mathcal{D}^{\mathsf{Game}_6}] - \Pr[\mathcal{D}^{\mathsf{Game}_7}]\right| \leq q_{\mathsf{C}} \cdot \log M \cdot \max_{0 \leq i \leq c} \{\delta_i + \delta_{i+1}\} = q_{\mathsf{C}} \cdot \log M \cdot (\delta_c + \delta_{c+1}) \ .$$

$\mathsf{Game}_7 : \mathsf{C}_{\oplus}^{\mathsf{H}_1,\mathsf{H}_2}(x)$

if $\exists y \in [M]$ s.t. $(x, y) \in \mathsf{hst}_{\mathsf{RO}}$ **then return** y
~~**if** $\exists y_1, y_2 \in [M]$ s.t. $(x, y_1) \in \mathsf{hst}_1 \wedge (x, y_2) \in \mathsf{hst}_2$ **then return** $y_1 \oplus y_2$~~
~~**if** $\exists y' \in [M]$ s.t. $(x, y') \in \mathsf{hst}_1 \vee (x, y') \in \mathsf{hst}_2$ **then**~~
  $y \twoheadleftarrow [M]$
~~**else**~~
  ~~$y_1 \leftarrow \mathsf{H}_1(x)$; $y_2 \leftarrow \mathsf{H}_2(x)$~~
  ~~$\mathsf{hst}_1 \leftarrow \mathsf{hst}_1 \cup \{(x, y_1)\}$; $\mathsf{hst}_2 \leftarrow \mathsf{hst}_2 \cup \{(x, y_2)\}$~~
  ~~$\mu_1 \leftarrow \mu_1|_{\mathsf{hst}_1}$; $\mu_2 \leftarrow \mu_2|_{\mathsf{hst}_2}$~~
  ~~$y \leftarrow y_1 \oplus y_2$~~
$\mathsf{hst}_{\mathsf{RO}} \leftarrow \mathsf{hst}_{\mathsf{RO}} \cup \{(x, y)\}$
return y

The last game Game_7 is identical to the simulated world. Therefore, the overall advantage of $\mathcal{D}$ is as stated in the theorem.

Query complexity. The queries made by the simulator to RO consist of those made when simulating evaluation queries and those made when simulating backdoor queries. Responding to each evaluation query requires exactly one query to RO, which makes a total of q_{H} queries. Right after the Q-th consecutive backdoor query (i.e., right before a switch), the simulator fixes some rows of the other BRO, where for each fixed row one query to the random oracle RO is made. The maximum number of rows that should be fixed after each sequence of Q queries to BD_1 (resp. BD_2) is predetermined by the simulator's parameter $\bar{p}$. Hence we obtain the claimed query complexity $q_{\mathsf{H}} + \sum_{i=1}^{c+1} p_i$. □

We now provide estimates for the involved parameters.

Corollary 1. *Let the number of switches be $c \geq 1$. Then for any $\alpha_1 > 1 - 1/F_{c+1}$, where F_i are the Fibonacci numbers, there is an indifferentiability simulator Sim for the $\mathsf{C}_{\oplus}$ construction in the 2-BRO model which has query complexity $q_{\mathsf{H}} + (c+1) \cdot N^{\alpha_1}$ for any distinguisher with q_{H} queries to the underlying BROs. Furthermore, any such distinguisher which places q_{C} construction queries and Q consecutive queries to the same backdoor oracle before switching has advantage at most*

$$(c+1) \cdot \gamma + \log M \cdot (c^2 B + 2q_{\mathsf{H}} + 2q_{\mathsf{C}}) \cdot N^{(1-\alpha_1) \cdot F_{c+1}/F_{c+2} - 1/F_{c+2}} \ ,$$

against the simulator, where $B := (Q\ell + \log \gamma^{-1})/\log M$. Asymptotically the query complexity is $q_{\mathsf{H}} + \mathcal{O}(N^{1-1/F_{c+2}})$ and the advantage $\mathcal{O}((q_{\mathsf{H}} + q_{\mathsf{C}}) \cdot Q \cdot \ell / N^{0.38/F_{c+2}})$.

Proof. From Lemma 1 we have that

$$\delta_i \leq (\delta_{i-2} \cdot A + B)/p_i \ ,$$

where $A := N$ and $B := (Q\ell + \log \gamma^{-1})/\log M$. Recursively applying the equation we get for odd i

$$\delta_i \leq \frac{B}{p_i} + \frac{AB}{p_i p_{i-2}} + \cdots + \frac{A^{(i-1)/2}B}{p_i p_{i-2} \cdots p_1}$$

Using $p_i < A$, the terms progressively get larger. Thus, in general

$$\delta_i \leq \frac{c \cdot N^{(i-2+i \bmod 2)/2} B}{p_i p_{i-2} \cdots p_{1+(i+1) \bmod 2}} \ .$$

For the indifferentiability advantage to be small, we would need to minimize

$$\sum_{i=1}^{c} p_i \cdot \delta_{i-1} + (q_{\mathsf{H}} + q_{\mathsf{C}})(\delta_c + \delta_{c+1}).$$

Let's assume $p_i = N^{\alpha_i}$ for some $\alpha_i \in [0, 1)$. Then the i-th summand for $i > 1$ is

$$c \cdot B \cdot N^{\alpha_i - \alpha_{i-1} - \alpha_{i-3} - \cdots - \alpha_{1+i \bmod 2} + (i-3+(i-1) \bmod 2)/2} \ .$$

To minimize, we set all terms equal to a common value $c \cdot B \cdot N^{\theta}$. We obtain

$$\alpha_i - \alpha_{i-1} - \ldots - \alpha_{1+i \bmod 2} + (i - 3 + (i-1) \bmod 2)/2 = \theta \ ,$$

Solving this system of linear equations gives

$$\alpha_i = F_i \cdot \theta + F_{i-1} \cdot (\alpha_1 - 1) + 1 \ ,$$

where F_i are the Fibonacci numbers with $F_0 = 0$ and $F_1 = 1$.

We may arrange the terms so that $(\delta_c + \delta_{c+1}) = 2 \cdot N^{\theta}$ (not including the $(q_{\mathsf{H}} + q_{\mathsf{C}})$ factor). To this end, we set $\alpha_{c+2} = 0$ so that $\delta_{c+1} = N^{\theta}/p_{c+2} = N^{\theta}$ and $\delta_c = N^{\theta}/p_{c+1} \leq N^{\theta}/p_{c+2} = N^{\theta}$. Thus we set $\alpha_{c+2} = 0$. This gives $\theta = (1-\alpha_1) \cdot F_{c+1}/F_{c+2} - 1/F_{c+2}$. Now for $\theta < 0$ we would need that $\alpha_1 > 1 - 1/F_{c+1}$. This means that the query complexity of the simulator is $q_{\mathsf{H}} + (c+1) \cdot N^{\alpha_1}$ and its advantage is

$$(c+1) \cdot \gamma + \log M \cdot (c^2 B + 2q_{\mathsf{H}} + 2q_{\mathsf{C}}) \cdot N^{(1-\alpha_1) \cdot F_{c+1}/F_{c+2} - 1/F_{c+2}} \ .$$

We obtain the bound stated in the asymptotic part of the corollary by setting $\alpha_1 := 1 - 1/F_{c+2} > 1 - 1/F_{c+1}$. □

We note that in the special case where $c = 1$, we must have that $\alpha_1 > 1 - 1/F_2 = 0$. In particular we can set $\alpha_1 := 1/4$ to obtain a simulator that places $N^{\alpha_1} = N^{1/4} \leq \sqrt{N}$ queries. Thus in this case we obtain collision resistance. Note, however, that as soon as $c \geq 2$ we would need to have that $\alpha_1 > 1 - 1/F_3 = 1/2$, which means the simulator places at least $\sqrt{N}$ queries, and we do not get collision resistance.

The above corollary shows that the xor combiner can only tolerate a logarithmic number of switches in $\log N$, which we think of as the security parameter. This is due to the fact that the simulator complexity needs to be less than $N/2$ for it to be non-trivial. Although our bounds are arguably weak, they are still meaningful, and we conjecture that much better bounds in reality hold.

5 An Extractor-Based Combiner

In this section we study the indifferentiability of extractor-based combiners and show that they can give better security parameters compared to the xor combiner of Section 4. Recall that in the k-BRO model one considers adversaries that have access to all k backdoor oracles. A query to the backdoor oracle $\textsc{Bd}_i$ reveals some information about the underlying BRO H_i. The resulting distribution conditioned on the leakage can, using the decomposition technique, be translated into a distribution with a number of fixed coordinates, while the distribution of the rest remains dense. An indifferentiability simulator then fixes the same rows of the other BRO(s) in a way that consistency with the random oracle (which is to be indistinguishable from the construction) is ensured.

We demonstrated this idea for the xor combiner, where, before a switch to the other backdoor oracle, the simulator substituted p images of that BRO by uniformly random values, i.e., the result of the random oracle values xored with the ones just fixed. This causes a security loss of $p \cdot \delta \cdot \log M$ per switch, which corresponds to the advantage of an adversary distinguishing p uniform values from $(1-\delta)$-dense ones. Now consider a multi-source $(k_1, \ldots, k_t, \varepsilon)$-extractor as the combiner in t-BRO. The hope would be that as long as the images of the BROs have high min-entropy, the output of the extractor is ε-close to uniform. This makes it possible for us to express the loss described above in terms of a negligible ε and forgo the requirement on δ to be negligible.

In this section we focus on 2-out-of-3-source extractors as combiners, i.e., extractors that only require a minimal amount of min-entropy from two of the sources. More formally, let $\mathsf{Ext} : [M]^3 \to [2]$ be a 2-out-of-3-source $(k_1, k_2, k_3, \varepsilon)$-extractor. For three functions $\mathsf{H}_1, \mathsf{H}_2, \mathsf{H}_3 : [N] \to [M]$, the combiner $\mathsf{C}_{3\mathrm{ext}}^{\mathsf{H}_1,\mathsf{H}_2,\mathsf{H}_3} : [N] \to [2]$ is defined as $\mathsf{C}_{3\mathrm{ext}}^{\mathsf{H}_1,\mathsf{H}_2,\mathsf{H}_3}(x) := \mathsf{Ext}\big(\mathsf{H}_1(x), \mathsf{H}_2(x), \mathsf{H}_3(x)\big)$. Here we show that in the 3-BRO model the construction $\mathsf{C}_{3\mathrm{ext}}^{\mathsf{H}_1,\mathsf{H}_2,\mathsf{H}_3}$ is indifferentiable from a random oracle.

Why not a two-source extractor? Note that we cannot guarantee that images which are being fixed by the simulator in some H_i as a result of a $\textsc{Bd}_i$-query have *any* min-entropy whatsoever. To understand why, simply consider an adversary that makes a backdoor query to $\textsc{Bd}_1$ requesting a preimage of the zero-string $y^* := 0^{\log M}$ under H_1. Suppose $\textsc{Bd}_1$ responds to this query with $x^* \in [N]$. In this case $\mathsf{H}_1(x^*)$ has no min-entropy, since $y^* = \mathsf{H}_1(x^*)$ was chosen by the adversary and is, therefore, completely predictable. Hence, $\mathsf{H}_1(x^*)$ cannot be used in a (k_1, k_2, ε)-two-source extractor, i.e., $\mathsf{Ext}(\mathsf{H}_1(x^*), \mathsf{H}_2(x^*))$, which relies on min-entropy from both sources for its output to be ε-close to uniform. Overall, using a two-source extractor does not seem to have any advantage over the xor combiner in the 2-BRO model. On the contrary, when using a 2-out-of-3-source extractor, assuming that the rows under consideration are not already fixed in the function tables of all three BROs due to some previous query, there will be two images with high min-entropy, from which we can extract a value ε-close to uniform.

Theorem 2 (Indifferentiability of 2-out-of-3-source extractors in the 3-BRO model with bounded adaptivity). *Let* $\mathsf{Ext} : [M]^3 \to [2]$ *be a* $(k_1, k_2, k_3, \varepsilon)$*-2-out-of-3-source randomness extractor, where* ε *is a function of* k_1, k_2, k_3*. Consider the combiner* $\mathsf{C}_{3\mathrm{ext}}^{\mathsf{H}_1,\mathsf{H}_2,\mathsf{H}_3}(x) := \mathsf{Ext}(\mathsf{H}_1(x), \mathsf{H}_2(x), \mathsf{H}_3(x))$ *in the 3-BRO model with backdoored hash functions* $\mathsf{H}_1, \mathsf{H}_2, \mathsf{H}_3 \in [M]^N$*. It holds that for all values of* $\bar{p} := (p_1, \ldots, p_{c+1}) \in \mathbb{N}^{c+1}$*,* $0 < \gamma < 1$*, and an integer* $c \geq 0$*, there exists a simulator* $\mathsf{Sim}[\bar{p}, \gamma] := (\mathsf{SimH}_1^{\mathsf{RO}}, \mathsf{SimH}_2^{\mathsf{RO}}, \mathsf{SimH}_3^{\mathsf{RO}}, \mathsf{SimBD}_1^{\mathsf{RO}}[\bar{p}, \gamma], \mathsf{SimBD}_2^{\mathsf{RO}}[\bar{p}, \gamma], \mathsf{SimBD}_3^{\mathsf{RO}}[\bar{p}, \gamma])$ *such that for any differentiator* $\mathcal{D}$ *that always makes* Q *queries to one backdoor oracle (always receiving an* ℓ*-bit response) before switching to the next, with a total number of* c *switches, while arbitrarily interleaving up to* q_{H} *primitive queries and* q_{C} *construction queries, we have*

$$\begin{aligned}\mathsf{Adv}^{\mathrm{indiff}}_{\mathsf{C}_{3\mathrm{ext}}^{\mathsf{H}_1,\mathsf{H}_2,\mathsf{H}_3},\mathsf{Sim}[\bar{p},\gamma]}(\mathcal{D}) \leq\ & (c+1)\cdot\gamma \\ & + \sum_{i=1}^{c} \mathrm{SD}\big(E_1|\cdots|E_{p_i}, \mathcal{U}_{[2]^{p_i}}\big) + q_{\mathsf{H}} \cdot \mathrm{SD}\big(E_1, \mathcal{U}_{[2]}\big) \\ & + q_{\mathsf{C}} \cdot \varepsilon\big((1-\delta_{c-1})\cdot\log M, (1-\delta_c)\cdot\log M, (1-\delta_{c+1})\cdot\log M\big)\ ,\end{aligned}$$

where for all $n \in \mathbb{N}$*, we define* $E_n := \mathsf{Ext}(X, Y, Z)$ *for some random variables* X, Y, Z *over* $[M]$ *such that at least 2 of them have min-entropy* $(1-\delta_c)\cdot\log M$*. Furthermore, we let* $\delta_{-2} := \delta_{-1} := \delta_0 := 0$ *and for other values of* $i \leq c+1$ *let* $\delta_i := \big(\delta_{i-3}\cdot(N - \sum_{j=1}^{i-3} p_j)\cdot\log M + Q\cdot\ell + \log\gamma^{-1}\big)/\big(p_i\cdot\log M\big)$ *be the density rate after the* i*-th sequence of* Q*-many backdoor queries. The simulator places at most* $q_{\mathsf{Sim}} \leq q_{\mathsf{H}} + \sum_{i=1}^{c+1} p_i$ *queries to the random oracle* RO*.*

Proof. The proof structure closely follows the proof of Theorem 1. We show indifferentiability by (1) defining a simulator, (2) upper bounding the advantage of any differentiator in distinguishing the real world from the simulated world, and (3) upper-bounding the number RO-queries made by the given simulator.

Simulator. The simulator is described in Figure 2 by algorithms SimH_i and SimBD_i (for $i = 1, 2, 3$). The simulator sub-algorithms share state and keep track of the current distribution of the backdoored hash functions. The histories $\mathsf{hst}_1, \mathsf{hst}_2, \mathsf{hst}_3$, initialized as empty sets, are used to keep track of the fixed coordinates of the simulated BROs. The distributions, according to which the simulated backdoored hash functions are sampled, are denoted by μ_1, μ_2, and μ_3 and initialized as $\mathcal{U}_{[M]^N}$, since the hash functions without the backdoors are supposed to behave like random oracles. The corresponding hash functions are initialized as uniform random functions $\mathsf{H}_i \twoheadleftarrow \mathcal{U}_{[M]^N}$. Furthermore, the simulator uses a counter q to keep track of the number of consecutive queries to a backdoor oracle and use this information to substitute the current distribution with a partially fixed and partially dense distribution, only when necessary (i.e., when $q = Q$), as opposed to doing so upon every backdoor query. Each time images of a simulated H_i are fixed by the simulator BD_i, images of the same rows must be fixed for H_j and H_k (i.e., the other two functions) to provide consistency with the random oracle RO. For this, images of H_j are fixed truthfully according to

the currently sampled function, while H_k is tweaked in a way that the extracted values match images of RO. Note that the simulators need to re-sample H_i and H_k if their distribution is modified in a non-trivial way, i.e., not just fixing more values, but either through $\mathsf{FixRows}$ or to force consistency with RO.

$\mathsf{SimH}_i^{\mathsf{RO}}(x)$

$y_i \leftarrow \mathsf{H}_i(x)$; $\mathsf{hst}_i \leftarrow \mathsf{hst}_i \cup \{(x, y_i)\}$
$\mu_i \leftarrow \mu_i|_{\mathsf{hst}_i}$
$j \leftarrow (i \bmod 3) + 1$; $k \leftarrow (j \bmod 3) + 1$
$y \leftarrow \mathsf{RO}(x)$
if $i = 1$ **then** $\mathsf{H}_k \twoheadleftarrow \mu_k|_{\mathsf{Ext}(y_i, \mathsf{H}_j(x), \mathsf{H}_k(x))=y}$
elseif $i = 2$ **then** $\mathsf{H}_k \twoheadleftarrow \mu_k|_{\mathsf{Ext}(\mathsf{H}_k(x), y_i, \mathsf{H}_j(x))=y}$
else $\mathsf{H}_k \twoheadleftarrow \mu_k|_{\mathsf{Ext}(\mathsf{H}_j(x), \mathsf{H}_k(x), y_i)=y}$
$\mathsf{hst}_j \leftarrow \mathsf{hst}_j \cup \{(x, \mathsf{H}_j(x))\}$; $\mathsf{hst}_k \leftarrow \mathsf{hst}_k \cup \{(x, \mathsf{H}_k(x))\}$
$\mu_j \leftarrow \mu_j|_{\mathsf{hst}_j}$; $\mu_k \leftarrow \mu_k|_{\mathsf{hst}_k}$
$\mathsf{H}_k \twoheadleftarrow \mu_k$
return y_i

$\mathsf{RO}(x)$

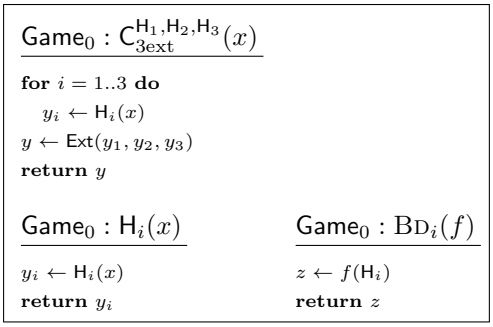
if $\exists y \in [2]$ s.t. $(x, y) \in \mathsf{hst}_{\mathsf{RO}}$ **then**
  return y
$y \twoheadleftarrow [2]$
$\mathsf{hst}_{\mathsf{RO}} \leftarrow \mathsf{hst}_{\mathsf{RO}} \cup \{(x, y)\}$
return y

$\mathsf{SimBD}_i^{\mathsf{RO}}[\bar{p}, \gamma](f)$

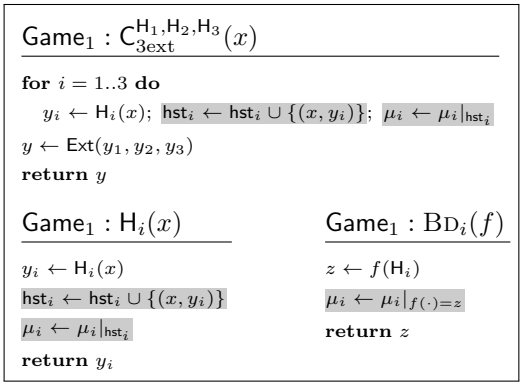
$q \leftarrow q + 1$
$z \leftarrow f(\mathsf{H}_i)$
$\mu_i \leftarrow \mu_i|_{f(\cdot)=z}$
$j \leftarrow (i \bmod 3) + 1$; $k \leftarrow (j \bmod 3) + 1$
if $q = Q$ **then**
  $(\mu_i, A_i) \twoheadleftarrow \mathsf{FixRows}[\gamma](\mu_i, p_{3s+i}, \mathsf{hst}_{i.1})$
  $\mathsf{H}_i \twoheadleftarrow \mu_i$
  $\mathsf{hst}_i \leftarrow \mathsf{hst}_i \cup A_i$
  for $x \in A_{i.1}$ **do** $r_x \leftarrow \mathsf{RO}(x)$
  if $i = 1$ **then** $\mathsf{H}_k \twoheadleftarrow \mu_k|_{\forall (x,y_i) \in A_i .\, \mathsf{Ext}(y_i, \mathsf{H}_j(x), \mathsf{H}_k(x))=r_x}$
  elseif $i = 2$ **then** $\mathsf{H}_k \twoheadleftarrow \mu_k|_{\forall (x,y_i) \in A_i .\, \mathsf{Ext}(\mathsf{H}_k(x), y_i, \mathsf{H}_j(x))=r_x}$
  else $\mathsf{H}_k \twoheadleftarrow \mu_k|_{\forall (x,y_i) \in A_i .\, \mathsf{Ext}(\mathsf{H}_j(x), \mathsf{H}_k(x), y_i)=r_x}$
  for $x \in A_{i.1}$ **do**
    $\mathsf{hst}_j \leftarrow \mathsf{hst}_j \cup \{(x, \mathsf{H}_j(x))\}$; $\mathsf{hst}_k \leftarrow \mathsf{hst}_k \cup \{(x, \mathsf{H}_k(x))\}$
  $\mu_j \leftarrow \mu_j|_{\mathsf{hst}_j}$; $\mu_k \leftarrow \mu_k|_{\mathsf{hst}_k}$
  $\mathsf{H}_k \twoheadleftarrow \mu_k$
  $q \leftarrow 0$
  if $i = 3$ **then** $s \leftarrow s + 1$
return z

Fig. 2: Indifferentiability simulator for the 2-out-of-3-source extractor. We assume for $i = 1..3$ initialization values $\mathsf{hst}_i = \mathsf{hst}_{\mathsf{RO}} := \emptyset$, $\mu_i := \mathcal{U}_{[M]^N}$, $\mathsf{H}_i \twoheadleftarrow \mathcal{U}_{[M]^N}$, $q := 0$, and $s := 0$. The $\mathsf{FixRows}$ algorithm is identical to that of Figure 1.

Security Analysis. We analyze indifferentiability of 3ext-combiner using a sequence of eight games $\mathsf{Game}_0, \ldots, \mathsf{Game}_7$, where Game_0 and Game_7 are the real and the ideal indifferentiability games, respectively. The modified lines in each game are highlighted. Oracles are omitted in some games if they have not changed since the previous game.



We use the shorthand notation $\Pr[\mathcal{D}^{\mathsf{Game}}] := \Pr[\mathcal{D}^{\mathsf{Game}} = 1]$, where $\mathcal{D}^{\mathsf{Game}}$ indicates the interaction of an adversary $\mathcal{D}$ with a game Game. In each game hop, we bound the adversary's advantage in distinguishing any two consecutive games from one another. The first game Game_0 is the real game, where the adversary interacts with the 3ext-combiner and the backdoor oracles of the underlying BROs.



Game_1. Game Game_1 updates the distribution of hash functions based on evaluation queries, backdoor queries, and the history of coordinates that are fixed through construction queries. The distributions μ_i are conditioned on these values but are never actually sampled from in the game. Hence the two games are identical, i.e., $\mathrm{SD}(\mathsf{Game}_0, \mathsf{Game}_1) = 0$.

$\mathsf{Game}_2 : \mathrm{BD}_i(f)$

$q \leftarrow q + 1$
$z \leftarrow f(\mathsf{H}_i)$
$\mu_i \leftarrow \mu_i|_{f(\cdot)=z}$
if $q = Q$ **then**
  $(\mu'_i, A_i) \twoheadleftarrow \mathsf{FixRows}[\gamma](\mu_i, p_{3s+i}, \mathsf{hst}_{i.1})$
  $q \leftarrow 0$
  if $i = 3$ **then** $s \leftarrow s + 1$
return z

Game_2. In game Game_2, after each sequence of Q queries to a backdoor oracle, i.e., right before switching to a different one, a $(p, 1-\delta)$-dense distribution μ'_i is obtained from the real distribution using the algorithm $\mathsf{FixRows}$ by decomposing the distribution of the corresponding hash function after responding to the last query (i.e., $\mu_i|_{f(\cdot)=z}$). The number of fixed points p is a parameter determined by the simulator and the density rate δ can be obtained by applying Lemma 1. However, since the new distributions μ'_i are never used elsewhere, Game_2 remains identical to the previous Game_1, i.e., $\mathrm{SD}(\mathsf{Game}_1, \mathsf{Game}_2) = 0$.

$\mathsf{Game}_3 : \mathsf{H}_i(x)$

$y_i \leftarrow \mathsf{H}_i(x)$
$\mathsf{hst}_i \leftarrow \mathsf{hst}_i \cup \{(x, y_i)\}$
$\mu_i \leftarrow \mu_i|_{\mathsf{hst}_i}$
$j \leftarrow (i \bmod 3) + 1$; $k \leftarrow (j \bmod 3) + 1$
$\mathsf{hst}_j \leftarrow \mathsf{hst}_j \cup \{(x, \mathsf{H}_j(x))\}$
$\mathsf{hst}_k \leftarrow \mathsf{hst}_k \cup \{(x, \mathsf{H}_k(x))\}$
$\mu_j \leftarrow \mu_j|_{\mathsf{hst}_j}$; $\mu_k \leftarrow \mu_k|_{\mathsf{hst}_k}$
return y_i

$\mathsf{Game}_3 : \mathrm{BD}_i(f)$

$q \leftarrow q + 1$
$z \leftarrow f(\mathsf{H}_i)$
$\mu_i \leftarrow \mu_i|_{f(\cdot)=z}$
$j \leftarrow (i \bmod 3) + 1$; $k \leftarrow (j \bmod 3) + 1$
if $q = Q$ **then**
  $(\mu'_i, A_i) \twoheadleftarrow \mathsf{FixRows}[\gamma](\mu_i, p_{3s+i}, \mathsf{hst}_{i.1})$
  for $x \in A_{i.1}$ **do**
    $\mathsf{hst}_j \leftarrow \mathsf{hst}_j \cup \{(x, \mathsf{H}_j(x))\}$
    $\mathsf{hst}_k \leftarrow \mathsf{hst}_k \cup \{(x, \mathsf{H}_k(x))\}$
  $\mu_j \leftarrow \mu_j|_{\mathsf{hst}_j}$; $\mu_k \leftarrow \mu_k|_{\mathsf{hst}_k}$
  $q \leftarrow 0$
  if $i = 3$ **then** $s \leftarrow s + 1$
return z

Game_3. In this game, the fixed rows in one simulated BRO are also fixed for the other two BROs. E.g., in backdoor simulation, the rows in the assignment A_i are fixed for H_j and H_k. In both games, the oracles' behaviors are at all times consistent with their past responses as well as the construction. Hence, it does not matter, if more or less of the hash function tables are fixed in each query. The two games are again perfectly indistinguishable, i.e., we have $\mathrm{SD}(\mathsf{Game}_2, \mathsf{Game}_3) = 0$.

$\mathsf{Game}_4 : \mathrm{BD}_i(f)$

$q \leftarrow q + 1$
$z \leftarrow f(\mathsf{H}_i)$
$\mu_i \leftarrow \mu_i|_{f(\cdot)=z}$
$j \leftarrow (i \bmod 3) + 1;\ k \leftarrow (j \bmod 3) + 1$
if $q = Q$ **then**
  $(\mu_i, A_i) \twoheadleftarrow \mathsf{FixRows}[\gamma](\mu_i, p_{3s+i}, \mathsf{hst}_{i.1})$
  $\mathsf{H}_i \twoheadleftarrow \mu_i$
  $\mathsf{hst}_i \leftarrow \mathsf{hst}_i \cup A_i$
  for $x \in A_{i.1}$ **do**
    $\mathsf{hst}_j \leftarrow \mathsf{hst}_j \cup \{(x, \mathsf{H}_j(x))\}$
    $\mathsf{hst}_k \leftarrow \mathsf{hst}_k \cup \{(x, \mathsf{H}_k(x))\}$
  $\mu_j \leftarrow \mu_j|_{\mathsf{hst}_j};\ \mu_k \leftarrow \mu_k|_{\mathsf{hst}_k}$
  $q \leftarrow 0$
  if $i = 3$ **then** $s \leftarrow s + 1$
return z

$\mathsf{Game}_5 : \mathsf{H}_i(x)$

$y_i \leftarrow \mathsf{H}_i(x)$
$\mathsf{hst}_i \leftarrow \mathsf{hst}_i \cup \{(x, y_i)\}$
$\mu_i \leftarrow \mu_i|_{\mathsf{hst}_i}$
$j \leftarrow (i \bmod 3) + 1;\ k \leftarrow (j \bmod 3) + 1$
$y \leftarrow \mathsf{RO}(x)$
if $i = 1$ **then** $\mathsf{H}_k \twoheadleftarrow \mu_k|_{\mathsf{Ext}(y_i, \mathsf{H}_j(x), \mathsf{H}_k(x))=y}$
elseif $i = 2$ **then** $\mathsf{H}_k \twoheadleftarrow \mu_k|_{\mathsf{Ext}(\mathsf{H}_k(x), y_i, \mathsf{H}_j(x))=y}$
else $\mathsf{H}_k \twoheadleftarrow \mu_k|_{\mathsf{Ext}(\mathsf{H}_j(x), \mathsf{H}_k(x), y_i)=y}$
$\mathsf{hst}_j \leftarrow \mathsf{hst}_j \cup \{(x, \mathsf{H}_j(x))\}$
$\mathsf{hst}_k \leftarrow \mathsf{hst}_k \cup \{(x, \mathsf{H}_k(x))\}$
$\mu_j \leftarrow \mu_j|_{\mathsf{hst}_j};\ \mu_k \leftarrow \mu_k|_{\mathsf{hst}_k}$
$\mathsf{H}_k \twoheadleftarrow \mu_k$
return y_i

$\mathsf{Game}_5 : \mathrm{BD}_i(f)$

$q \leftarrow q + 1$
$z \leftarrow f(\mathsf{H}_i)$
$\mu_i \leftarrow \mu_i|_{f(\cdot)=z}$
$j \leftarrow (i \bmod 3) + 1;\ k \leftarrow (j \bmod 3) + 1$
if $q = Q$ **then**
  $(\mu_i, A_i) \twoheadleftarrow \mathsf{FixRows}[\gamma](\mu_i, p_{3s+i}, \mathsf{hst}_{i.1})$
  $\mathsf{H}_i \twoheadleftarrow \mu_i$
  $\mathsf{hst}_i \leftarrow \mathsf{hst}_i \cup A_i$
  for $x \in A_{i.1}$ **do** $r_x \leftarrow \mathsf{C}_{3\mathrm{ext}}(x)$
  if $i = 1$ **then**
    $\mathsf{H}_k \twoheadleftarrow \mu_k|_{\forall (x,y_i) \in A_i .\ \mathsf{Ext}(y_i, \mathsf{H}_j(x), \mathsf{H}_k(x))=r_x}$
  elseif $i = 2$ **then**
    $\mathsf{H}_k \twoheadleftarrow \mu_k|_{\forall (x,y_i) \in A_i .\ \mathsf{Ext}(\mathsf{H}_k(x), y_i, \mathsf{H}_j(x))=r_x}$
  else $\mathsf{H}_k \twoheadleftarrow \mu_k|_{\forall (x,y_i) \in A_i .\ \mathsf{Ext}(\mathsf{H}_j(x), \mathsf{H}_k(x), y_i)=r_x}$
  for $x \in A_{i.1}$ **do**
    $\mathsf{hst}_j \leftarrow \mathsf{hst}_j \cup \{(x, \mathsf{H}_j(x))\}$
    $\mathsf{hst}_k \leftarrow \mathsf{hst}_k \cup \{(x, \mathsf{H}_k(x))\}$
  $\mu_j \leftarrow \mu_j|_{\mathsf{hst}_j};\ \mu_k \leftarrow \mu_k|_{\mathsf{hst}_k}$
  $\mathsf{H}_k \twoheadleftarrow \mu_k$
  $q \leftarrow 0$
  if $i = 3$ **then** $s \leftarrow s + 1$
return z

Game_4. In Game_4 the distribution obtained by $\mathsf{FixRows}$ finally replaces the true distribution, i.e., the one conditioned on the recent backdoor responses. Hence, the history is updated. Notably, a new function H_i must be sampled for future references, since its distribution has changed in a non-trivial way. According to Lemma 1, there is a convex combination of $(p, 1-\delta)$-dense distributions

which is γ-close to the real distribution, one of such distributions being the one returned by FixRows. Thus, the distinguishing advantage increases by γ after each sequence of backdoor queries, i.e., $\left|\Pr[\mathcal{D}^{\mathsf{Game}_3}] - \Pr[\mathcal{D}^{\mathsf{Game}_4}]\right| \leq (c+1)\cdot\gamma$.

Game_5. Contrary to Game_4, the next game Game_5 somewhat indirectly fixes images of rows x in A_i (and x queries to SimH_i) for the other functions H_j and H_k. More precisely, the simulator calls the construction $\mathsf{C}_{3\mathrm{ext}}$ on freshly fixed rows according to A_i and samples a H_k in such a way that it is consistent with those construction images, and aligned H_i and H_k images. Notice that a query to the construction already fixes the images for the underlying BROs and therefore, sampling H_k in a consistent way and fixing coordinates of H_j and H_k in the simulator is simply redundant. Hence $\mathrm{SD}(\mathsf{Game}_4, \mathsf{Game}_5) = 0$.

$\mathsf{Game}_6 : \mathsf{C}_{3\mathrm{ext}}^{\mathsf{H}_1,\mathsf{H}_2,\mathsf{H}_3}(x)$

if $\exists y \in [M]$ s.t. $(x,y) \in \mathsf{hst}_{\mathsf{RO}}$ **then return** y
if $\exists y_1, y_2, y_3 \in [M]$ s.t. $(x,y_1) \in \mathsf{hst}_1 \wedge (x,y_2) \in \mathsf{hst}_2 \wedge (x,y_3) \in \mathsf{hst}_3$ **then return** $\mathsf{Ext}(y_1,y_2,y_3)$
if $\exists y' \in [M]$ s.t. $(x,y') \in \mathsf{hst}_1 \vee (x,y') \in \mathsf{hst}_2 \vee (x,y') \in \mathsf{hst}_3$ **then**
  $y \twoheadleftarrow [M]$
else
  for $i = 1..3$ **do**
    $y_i \leftarrow \mathsf{H}_i(x);\ \mathsf{hst}_i \leftarrow \mathsf{hst}_i \cup \{(x,y_i)\};\ \mu_i \leftarrow \mu_i|_{\mathsf{hst}_i}$
  $y \leftarrow \mathsf{Ext}(y_1,y_2,y_3)$
$\mathsf{hst}_{\mathsf{RO}} \leftarrow \mathsf{hst}_{\mathsf{RO}} \cup \{(x,y)\}$
return y

Game_6. In this game we modify $\mathsf{C}_{3\mathrm{ext}}$ so that it starts to resemble a lazily sampled random oracle. Query-response pairs of the construction are kept in a set $\mathsf{hst}_{\mathsf{RO}}$ and in case a query is repeated the stored image is simply returned. Otherwise, we distinguish three cases: (a) the corresponding row to the current query x is fixed in all hash functions, (b) in one of them, or (c) in none of them. In case (a), Game_6 computes the output of the construction by extracting from the individual images stored in histories of the BROs. Note, however, that this case is never reached, since if the current x is in all individual histories, then the construction must have already been called on x in some previous evaluation or backdoor query. Hence, x must also be in $\mathsf{hst}_{\mathsf{RO}}$. In case (b), a uniformly random value is chosen (and stored in $\mathsf{hst}_{\mathsf{RO}}$). In the final case (c), Game_6 behaves exactly as Game_5.

Overall, the distinguishing advantage is bounded by distinguishing p uniform (chosen by the construction) points each time a backdoor query fixes p points from values that were supposed to be extracted from three sources, from which one is not guaranteed to have any min-entropy, as well as q_{H} many times distinguishing a single extracted value from random. Let $E_n := \mathsf{Ext}\big(\mathsf{H}_1(x_n), \mathsf{H}_2(x_n),$

$\mathsf{H}_3(x_n)\big)$, where $x_n \in A_{i.1}$ is a row being fixed. Then we have:

$$\big|\Pr[\mathcal{D}^{\mathsf{Game}_5}] - \Pr[\mathcal{D}^{\mathsf{Game}_6}]\big| \le \sum_{i=1}^{c} \mathrm{SD}\big(E_1|\cdots|E_{p_i}, \mathcal{U}_{[2]^{p_i}}\big) + q_{\mathsf{H}} \cdot \mathrm{SD}\big(E_1, \mathcal{U}_{[2]}\big) \ ,$$

$\mathsf{Game}_7 : \mathsf{C}_{3\mathrm{ext}}^{\mathsf{H}_1,\mathsf{H}_2,\mathsf{H}_3}(x)$

if $\exists y \in [M]$ s.t. $(x,y) \in \mathsf{hst}_{\mathsf{RO}}$ **then return** y
~~**if** $\exists y_1, y_2, y_3 \in [M]$ s.t. $(x,y_1) \in \mathsf{hst}_1 \wedge (x,y_2) \in \mathsf{hst}_2 \wedge (x,y_3) \in \mathsf{hst}_3$ **then return** $\mathsf{Ext}(y_1,y_2,y_3)$~~
~~**if** $\exists y' \in [M]$ s.t. $(x,y') \in \mathsf{hst}_1 \vee (x,y') \in \mathsf{hst}_2 \vee (x,y') \in \mathsf{hst}_3$ **then**~~
  $y \twoheadleftarrow [M]$
~~**else**~~
  ~~**for** $i = 1..3$ **do**~~
    ~~$y_i \leftarrow \mathsf{H}_i(x);\ \mathsf{hst}_i \leftarrow \mathsf{hst}_i \cup \{(x,y_i)\};\ \mu_i \leftarrow \mu_i|_{\mathsf{hst}_i}$~~
  ~~$y \leftarrow \mathsf{Ext}(y_1,y_2,y_3)$~~
$\mathsf{hst}_{\mathsf{RO}} \leftarrow \mathsf{hst}_{\mathsf{RO}} \cup \{(x,y)\}$
return y

Game_7. The $\mathsf{C}_{3\mathrm{ext}}$ oracle in Game_7 differs from Game_6 in that it never evaluates the underlying BROs any more and rather acts as a lazily sampled random oracle. We can safely remove the case distinction (a), where x is included in all histories hst_1, hst_2, and hst_3, since this x would also be in $\mathsf{hst}_{\mathsf{RO}}$. It remains to bound the adversary's advantage in distinguishing the two games while making up to q_{C} fresh queries x to the construction $\mathsf{C}_{3\mathrm{ext}}$ that are not fixed for any of the BROs. While the outputs of the construction are uniformly random in Game_7, they are extracted from three dense images in Game_6. The distinguisher can only try to maximize the distance between q_{C} uniform values vs. values extracted from three dense images of BROs by querying the construction on values and at times which it can choose freely.

$$\begin{aligned}
\big|\Pr[\mathcal{D}^{\mathsf{Game}_6}] - \Pr[\mathcal{D}^{\mathsf{Game}_7}]\big| &\le \sum_{t=1}^{q_{\mathsf{C}}} \max_{x_t,\mathsf{H}_1,\mathsf{H}_2,\mathsf{H}_3} \Big(\mathrm{SD}\big(\mathsf{Ext}(\mathsf{H}_1(x_t),\mathsf{H}_2(x_t),\mathsf{H}_3(x_t)),\mathcal{U}_{[2]}\big)\Big) \\
&\le q_{\mathsf{C}} \cdot \max_{x,\mathsf{H}_1,\mathsf{H}_2,\mathsf{H}_3} \Big(\mathrm{SD}\big(\mathsf{Ext}(\mathsf{H}_1(x),\mathsf{H}_2(x),\mathsf{H}_3(x)),\mathcal{U}_{[2]}\big)\Big) \\
&\le q_{\mathsf{C}} \cdot \varepsilon\big((1-\delta_{c-1})\cdot\log M, (1-\delta_c)\cdot\log M, \\
&\qquad\qquad (1-\delta_{c+1})\cdot\log M\big) \ ,
\end{aligned}$$

where according to Lemma 1 we have δ_i as defined in the theorem statement with ℓ_i being the min-entropy deficiency after the i-th sequence of Q-many backdoor queries. Note that the maximum statistical distance corresponds to minimum entropy of the BRO-images, which is in turn given for the last three $(c-1, c, c+1)$ values of the density rate.

Query complexity. The simulator makes queries to the random oracle RO to set images of the other BROs each time one point of some BRO is fixed, either caused

by evaluation queries or by backdoor queries right after the Q-th consecutive backdoor query (i.e., before a switch). Hence we obtain the bound $q_{\mathsf{Sim}} \leq q_{\mathsf{H}} + \sum_{i=1}^{c+1} p_i$ on the number of queries that the simulator makes to the random oracle. □

5.1 Instantiation with the pairwise inner-product extractor

Next we investigate a concrete instantiation of such a 2-out-of-3-source extractor. General multi-source extractors such as those from [2,25,21] which require a minimal amount of min-entropy from *every* source are inapplicable in our setting. We can, however, use the pairwise inner-product extractor as introduced by Lee et al. [19], which roughly speaking needs the sum of min-entropies to be sufficient. Formally a pairwise inner-product extractor $\mathsf{Ext}_{\mathrm{pip}} : [M]^t \to [2]$ is defined as:

$$\mathsf{Ext}_{\mathrm{pip}}(x_1, \ldots, x_t) := \sum_{1 \leq i < j \leq t} x_i \cdot x_j \ .$$

This extractor is proven ([19], Corollary 1) to be a $(k_1, \ldots, k_t, \varepsilon)$-extractor with $\varepsilon = 2^{-(k+k'-\log M+1)/2}$, where k and k' are the two largest values among $k_1, \ldots, k_t$. Hence, $\mathsf{Ext}_{\mathrm{pip}}$ is also a 2-out-of-t extractor.

Corollary 2. *Let* $\mathsf{Ext}_{\mathrm{pip}} : [M]^t \to [2]$ *be a pairwise inner-product extractor. Then the construction* $\mathsf{C}_{\mathrm{pip}}^{\mathsf{H}_1,\mathsf{H}_2,\mathsf{H}_3}(x) := \mathsf{Ext}_{\mathrm{pip}}(\mathsf{H}_1(x), \mathsf{H}_2(x), \mathsf{H}_3(x))$ *in the* 3*-BRO model is indifferentiable from a random oracle, where*

$$\begin{aligned}\mathsf{Adv}^{\mathrm{indiff}}_{\mathsf{C}^{\mathsf{H}_1,\mathsf{H}_2,\mathsf{H}_3}_{3\mathrm{ext}},\mathsf{Sim}[p,\gamma]}(\mathcal{D}) \leq & (c+1)\cdot\gamma \\ & + c \cdot \sqrt{(e^{p \cdot M^{-(1-2\delta_c)}} - 1)/2} \\ & + \big(q_{\mathsf{H}} + q_{\mathsf{C}}\big) \cdot 2^{-((1-2\delta_{c+1})\cdot \log M + 1)/2} \ ,\end{aligned}$$

while the simulator makes up to $q_{\mathsf{Sim}} \leq q_{\mathsf{H}} + (c+1) \cdot p$ *queries to* RO.

Proof. The differentiator's advantage stated in the corollary is easily obtained by upper bounding the term $\mathrm{SD}\big(E_1, \mathcal{U}_{[2]}\big)$ by $2^{-((1-2\delta_{c+1})\cdot \log M+1)/2}$ and upper bounding the term$\mathrm{SD}\big(E_1 | \cdots | E_p, \mathcal{U}_{[2]^p}\big)$ from the advantage in Theorem 2, using the following claim.

Claim. Let $x_i \in [N]$ and $E_i := \mathsf{Ext}_{\mathrm{pip}}\big(\mathsf{H}_1(x_i), \mathsf{H}_2(x_i), \mathsf{H}_3(x_i)\big)$ for $i = 1..n$. Suppose that for all x_i, at least two of the (distributions of the) functions H_1, H_2, H_3 are $(1-\delta)$-dense. Then for all $n \in \mathbb{N}$ we have:

$$\mathrm{SD}(E_1 | \cdots | E_n, \mathcal{U}_{[2]^n}) \leq \sqrt{(e^{n \cdot M^{-(1-2\delta)}} - 1)/2} \ .$$

Proof. In the proof below we use the parity lemma [5] (1) and the fact that the pairwise inner product (in $[L]$) is linear, i.e., $\sum_{n \in I} E_n = \sum_{n \in I} \mathsf{Ext}_{\mathrm{pip}}(\mathsf{H}_1(x_n),$

[5] Let X be a random variable over $[2^\ell]$. Then we have $\mathrm{SD}(X, \mathcal{U}_{[2^\ell]}) \leq \sqrt{\sum_{0^\ell \neq a \in [2^\ell]} \big(\mathrm{SD}(X \cdot a, \mathcal{U}_{[2]})\big)^2}$.

$\mathsf{H}_2(x_n), \mathsf{H}_3(x_n)) = \mathsf{Ext}_{\mathrm{pip}}(\mathsf{H}_1(x_1)|\cdots|\mathsf{H}_1(x_{|I|}), \mathsf{H}_2(x_1)|\cdots|\mathsf{H}_2(x_{|I|}), \mathsf{H}_3(x_1)|\cdots|\mathsf{H}_3(x_{|I|})) = E_I$ (2).

$$\begin{aligned}
\mathrm{SD}(E_1|\cdots|E_n, \mathcal{U}_{[2]^n}) &\leq \sqrt{\sum_{0^{n\log 2} \neq a \in [2]^n} \big(\mathrm{SD}(E_1|\cdots|E_n \cdot a, \mathcal{U}_{[2]})\big)^2} && (1)\\
&= \sqrt{\sum_{\emptyset \neq I \subseteq \{1,\ldots,n\}} \Big(\mathrm{SD}\big(\sum_{i\in I} E_i, \mathcal{U}_{[2]}\big)\Big)^2}\\
&= \sqrt{\sum_{\emptyset \neq I \subseteq \{1,\ldots,n\}} \big(\mathrm{SD}(E_I, \mathcal{U}_{[2]})\big)^2} && (2)\\
&\leq \sqrt{\sum_{\emptyset \neq I \subseteq \{1,\ldots,n\}} 2^{-(|I|\cdot \log M \cdot (1-2\delta) + 2 - \log 2)}}\\
&= \sqrt{2^{-2+\log 2} \cdot \sum_{\emptyset \neq I \subseteq \{1,\ldots,n\}} 2^{-|I|\cdot \log M \cdot (1-2\delta)}}\\
&= \sqrt{2^{-1} \cdot \sum_{\emptyset \neq I \subseteq \{1,\ldots,n\}} \big(M^{-(1-2\delta)}\big)^{|I|}}\\
&= \sqrt{\Big(\big(1 + M^{-(1-2\delta)}\big)^n - 1\Big)/2}\\
&\leq \sqrt{(e^{n\cdot M^{-(1-2\delta)}} - 1)/2} && \square
\end{aligned}$$

Hence, the claim about the advantage holds. The query complexity of the simulator is bounded by the sum of q_{H} and $(c+1)\cdot p$. $\square$

We now provide estimates for the involved parameters.

Corollary 3. *Let the number of switches be $c \geq 1$ and assume the range size of the three random oracles are $M \geq N^9$. Then there is an indifferentiability simulator* Sim *for the* $\mathsf{C}_{\mathrm{pip}}$ *construction in the* 3*-BRO model that places at most*

$$q_{\mathsf{H}} + (c+1)\cdot\left(\frac{6Q\ell}{\log M}\right)^{1/\alpha(c)} \cdot N^{1-1/\alpha(c)}$$

queries to RO*, where $\alpha(c) := \lfloor \frac{c}{3} \rfloor + 1$, against any distinguisher with q_{H} queries to the underlying BROs. Further, any such distinguisher with q_{C} construction queries and Q consecutive queries to the same backdoor oracle before switching, has advantage at most $(c+1)\cdot\gamma + (c + q_{\mathsf{H}} + q_{\mathsf{C}})/N$ against this simulator.*

Proof. The recurrence relations for δ_i in the statement of Theorem 2 can be written as

$$\delta_i \leq A \cdot \delta_{i-3} + B\ ,$$

where $A := N/p$ and $B := (Q\ell + \log\gamma^{-1})/p\log M$. Solving this recurrence relation we get

$$\delta_i \leq \frac{A^{\lfloor \frac{i-1}{3} \rfloor + 1} - 1}{A-1} \cdot B \ .$$

We set $\delta_{c+1} \leq 1/3$ so that the term $1 - 2\delta_{c+1}$ is positive. To this end, it is sufficient to have that

$$\frac{A^{\lfloor \frac{c}{3} \rfloor + 1} - 1}{A-1} \cdot B \leq \frac{1}{3} \ .$$

Substituting A and B and removing the -1 in the numerator we need to have that

$$\left(\frac{N}{p}\right)^{\lfloor \frac{c}{3} \rfloor + 1} \leq \frac{A-1}{3B} = \frac{(N/p-1)p\log M}{3Q\ell} = \frac{N\log M - p\log M}{3Q\ell} \leq \frac{N\log M}{6Q\ell} \ ,$$

where for the last inequality we have assumed that $p \leq N/2$. Thus,

$$p \geq \left(\frac{6Q\ell}{\log M}\right)^{1/\alpha(c)} \cdot N^{1-1/\alpha(c)} \ ,$$

where $\alpha(c) := \lfloor \frac{c}{3} \rfloor + 1$. For sufficiently large c, the factor above is at most 2.

The advantage stated in Corollary 2 is

$$(c+1)\cdot\gamma + c\cdot\sqrt{p/M^{1-2\delta_c}} + (q_{\mathsf{H}} + q_{\mathsf{C}})\cdot\sqrt{1/M^{1-2\delta_{c+1}}} \ .$$

Since $1 - 2\delta_{c+1} \leq 1 - 2/3 = 1/3$, $\delta_c \leq \delta_{c+1}$, $p \leq N$ and $M \geq N^9$, the advantage is upper-bounded by $(c+1)\cdot\gamma + (c + q_{\mathsf{H}} + q_{\mathsf{C}})/N$. □

Note that for $c = 1, 2$ the query complexity of the simulator does not involve the $N^{1-1/\alpha(c)}$ factor, and hence we obtain collision resistance. For $c \geq 3$, however there is a factor of at least $N^{1/2}$.

The above corollary shows that the extractor combiner can tolerate a *linear* number of switches in $\log N$ (which can be thought of as the security parameter) for the simulator query complexity to be less than $N/2$. As for the xor combiner we conjecture that (much) better bounds for the extractor combiner are possible.

6 Indifferentiability with Auxiliary Input

In this section we discuss indifferentiability in a setting where there is no adaptivity and the backdoor oracles are called only once at the onset. Although this may seem overly restrictive, the resulting definition is sufficiently strong to model indifferentiability in the presence of auxiliary input, whereby we would like to securely replace random oracles in generic applications even in the presence of auxiliary input.

In this setting we can view an indifferentiability simulator as operating in two stages: An off-line stage which responds to the single backdoor queries for

each BRO, and an on-line stage which simulates direct evaluation calls to the underlying BROs. As defined, the off-line phase of the simulator can pass an arbitrary state to its on-line phase. Further, both stages have access to the reference object oracles (although the query complexities of both stages need to be small). More precisely, this definition in the 2-BRO requires that for any $(\mathcal{D}_{0,1}, \mathcal{D}_{0,2}, \mathcal{D}_1)$ in the real world with two BROs H_1 and H_2 with

$$z_1 \leftarrow\!\!\!\ll \mathcal{D}_{0,1}(\mathsf{H}_1);\ z_2 \leftarrow\!\!\!\ll \mathcal{D}_{0,2}(\mathsf{H}_2, z_1);\ b \leftarrow\!\!\!\ll \mathcal{D}_1^{\mathsf{C}^{\mathsf{H}_1,\mathsf{H}_2},\mathsf{H}_1,\mathsf{H}_2}(z_1, z_2)\ ,$$

there exists some $(\mathsf{Sim}_{0,1}^{\mathsf{RO}}, \mathsf{Sim}_{0,2}^{\mathsf{RO}}, \mathsf{Sim}_{1,1}^{\mathsf{RO}}, \mathsf{Sim}_{1,2}^{\mathsf{RO}})$ in the ideal (simulated) world

$$(z_1, st) \leftarrow\!\!\!\ll \mathsf{Sim}_{0,1}^{\mathsf{RO}}();\ (z_2, st) \leftarrow\!\!\!\ll \mathsf{Sim}_{0,2}^{\mathsf{RO}}(st);\ b \leftarrow\!\!\!\ll \mathcal{D}_1^{\mathsf{RO},\mathsf{Sim}_{1,1}^{\mathsf{RO}}[st],\mathsf{Sim}_{1,2}^{\mathsf{RO}}[st]}(z_1, z_2)\ ,$$

with indistinguishable outputs b. The on-line simulators can also share state.

Let us now take a step back and define indifferentiability with auxiliary input driven by a composition theorem: for any game $\mathcal{G}$ and any attacker $\mathcal{A}_1$ in this game against $\mathsf{C}^{\mathsf{H}_1,\mathsf{H}_2}$ which receives auxiliary input on H_1 and H_2, there is an attacker $\mathcal{B}_1$ on RO in the same game $\mathcal{G}$ but now *without* auxiliary input. More explicitly, the real world

$$z \leftarrow\!\!\!\ll \mathcal{A}_0(\mathsf{H}_1, \mathsf{H}_2);\ b \leftarrow\!\!\!\ll \mathcal{G}^{\mathsf{C}^{\mathsf{H}_1,\mathsf{H}_2},\mathcal{A}_1^{\mathsf{H}_1,\mathsf{H}_2}(z)}$$

and the ideal world

$$(z, st) \leftarrow\!\!\!\ll \mathcal{B}_0^{\mathsf{RO}}();\ b \leftarrow\!\!\!\ll \mathcal{G}^{\mathsf{RO},\mathcal{B}_1^{\mathsf{RO}}(z,st)}$$

are indistinguishable. Once again the query complexity of $\mathcal{B}_0$ should be small (or even zero) to obtain a definition which meaningfully formalizes indifferentiability from random oracles without auxiliary input. This definition, however, turns out to be unachievable: $\mathcal{A}_0$ can simply encode a pair of collisions for the construction, which $\mathcal{B}_0$ will not be able to match (with respect to RO) without an exponentially large number of queries to RO.[6]

There are two natural ways to overcome this: (1) restrict the interface of the construction; or (2) restrict the form of preprocessing. The former is motivated by use of salting as a means to defeat preprocessing, and the latter by independence of preprocessing for BROs.

A final question arises here: is it possible to simplify this definition further by removing the quantification over $\mathcal{A}_1$ (as done for standard indifferentiability)? This could be done in the standard way by absorbing $\mathcal{A}_1$ into $\mathcal{G}$ to form a differentiator $\mathcal{D}$. However, this means that $\mathcal{D}$ must receive the auxiliary information z.

[6] One can formulate an intermediate notion of indifferentiability from random oracle *with* auxiliary input. Without salting, this notion would not be of great help. Consider, for example, the case of domain extension via an iterative hashing mode. Due to Joux's multi-collision attack [17] one can encode exponentially many collisions for the construction in a small auxiliary input, whereas this would not be possible for the random oracle.

The resulting notion is stronger and models composition with respect to *games* that also depend on preprocessing. Thus, due to its simplicity, strength, and the fact that we can establish positive results for it, we focus on this definitional approach. We now make the two definitions arising from (1) and (2) explicit.

Salted AI-indifferentiability. We call a construction C^{H} *salted* if the construction takes a salt $hk \in \{0,1\}^k$ as input and prepends all calls to H with hk. We define salted AI-indifferentiability from a random oracle by requiring that for any $(\mathcal{D}_0, \mathcal{D}_1)$ in the real world

$$z \twoheadleftarrow \mathcal{D}_0(\mathsf{H});\ hk \twoheadleftarrow \{0,1\}^k;\ b \twoheadleftarrow \mathcal{D}_1^{\mathsf{C}^{\mathsf{H}(hk,\cdot)}(hk,\cdot),\mathsf{H}}(hk, z)$$

there is a simulator $(\mathsf{Sim}_0^{\mathsf{RO}}, \mathsf{Sim}_1^{\mathsf{RO}})$ in the ideal world

$$(z, st) \twoheadleftarrow \mathsf{Sim}_0^{\mathsf{RO}}();\ hk \twoheadleftarrow \{0,1\}^k;\ b \twoheadleftarrow \mathcal{D}_1^{\mathsf{RO}(hk,\cdot),\mathsf{Sim}_1^{\mathsf{RO}}[st]}(hk, z)$$

resulting in indistinguishable outputs b. We denote the advantage of $\mathcal{D}$ in the salted AI-indifferentiability game with simulator Sim for a construction C^{H} by $\mathsf{Adv}^{\text{s-ai-indiff}}_{\mathsf{C}^{\mathsf{H}},\mathsf{Sim}}(\mathcal{D})$. Notice that in the above definition, the distinguisher gets access to a *salted* RO. A different definition arises when the distinguisher gets access to an unsalted RO instead. However, since the simulated auxiliary information is computed given access to an unsalted RO (which can be interpreted as having implicit access to the salt), such a definition calls for the existence of a more powerful simulator. In particular, such Sim_0 and $\mathcal{D}_1$ can easily call RO on common points. The practical implications of such a definition are unclear to us, and moreover, it is strictly weaker than our definition.

AI-indifferentiability with independent preprocessing. We define AI-indifferentiability with independent preprocessing by requiring that for any adversary $(\mathcal{D}_{0,1}, \mathcal{D}_{0,2}, \mathcal{D}_1)$ in the real world

$$z_1 \twoheadleftarrow \mathcal{D}_{0,1}(\mathsf{H}_1);\ z_2 \twoheadleftarrow \mathcal{D}_{0,2}(\mathsf{H}_2);\ b \twoheadleftarrow \mathcal{D}_1^{\mathsf{C}^{\mathsf{H}_1,\mathsf{H}_2},\mathsf{H}_1,\mathsf{H}_2}(z_1, z_2)$$

there is a simulator $(\mathsf{Sim}_{0,1}^{\mathsf{RO}}, \mathsf{Sim}_{0,2}^{\mathsf{RO}}, \mathsf{Sim}_{1,1}^{\mathsf{RO}}, \mathsf{Sim}_{1,2}^{\mathsf{RO}})$ in the ideal world

$$(z_1, st) \twoheadleftarrow \mathsf{Sim}_{0,1}^{\mathsf{RO}}();\ (z_2, st) \twoheadleftarrow \mathsf{Sim}_{0,2}^{\mathsf{RO}}(st);\ b \twoheadleftarrow \mathcal{D}_1^{\mathsf{RO},\mathsf{Sim}_{1,1}^{\mathsf{RO}}[st],\mathsf{Sim}_{1,2}^{\mathsf{RO}}[st]}(z_1, z_2)$$

resulting in indistinguishable outputs b. Note that this is slightly weaker than the definition of indifferentiability in 2-BRO since z_2 is fully independent of z_1, whereas BRO indifferentiability allows for a limited amount of dependence. We denote by $\mathsf{Adv}^{\text{ai-indiff}}_{\mathsf{C}^{\mathsf{H}},\mathsf{Sim}}(\mathcal{D})$ the advantage of $\mathcal{D}$ in the AI-indifferentiability game with independent preprocessing with respect to a simulator Sim and a construction $\mathsf{C}^{\mathsf{H}_1,\mathsf{H}_2}$ in the 2-BRO model.

We are now ready to prove our feasibility results for AI-indifferentiability.

Theorem 3 (AI-Indifferentiability). *Any construction $\mathsf{C}^{\mathsf{H}_1,\mathsf{H}_2}$ that is indifferentiable with backdoors from a random oracle with no adaptive backdoor queries*

is also AI-indifferentiable from a random oracle with respect to independent preprocessing attacks. More precisely, for any auxiliary-input differentiator $\mathcal{D} := (\mathcal{D}_{0,1}, \mathcal{D}_{0,2}, \mathcal{D}_1)$ *with independent preprocessing for two random oracles there is a 2-BRO differentiator* $\tilde{\mathcal{D}}$ *with one-time non-adaptive access to each backdoor oracle such that for any 2-BRO indifferentiability simulator* $\tilde{\mathsf{Sim}}$ *there is an auxiliary-input simulator* $\mathsf{Sim} := (\mathsf{Sim}_{0,1}, \mathsf{Sim}_{0,2}, \mathsf{Sim}_{1,1}, \mathsf{Sim}_{1,2})$ *such that*

$$\mathsf{Adv}^{\text{ai-indiff}}_{\mathsf{C}^{\mathsf{H}_1,\mathsf{H}_2},\mathsf{Sim}}(\mathcal{D}) = \mathsf{Adv}^{\text{indiff}}_{\mathsf{C}^{\mathsf{H}_1,\mathsf{H}_2},\tilde{\mathsf{Sim}}}(\tilde{\mathcal{D}}) \ .$$

Further, any salted construction C^{H} *that is indifferentiable (in the standard sense) from a random oracle is also salted AI-indifferentiable from a random oracle. More precisely, for any auxiliary-input differentiator* $\mathcal{D} := (\mathcal{D}_0, \mathcal{D}_1)$*, with an auxiliary input of size* ℓ*, there is a (standard) differentiator* $\tilde{\mathcal{D}}$ *such that for any indifferentiability simulator* $\tilde{\mathsf{Sim}}$ *there is an auxiliary-input simulator* $\mathsf{Sim} := (\mathsf{Sim}_0, \mathsf{Sim}_1)$ *such that for any* $p \in \mathbb{N}$ *and any* $\gamma > 0$

$$\mathsf{Adv}^{\text{s-ai-indiff}}_{\mathsf{C}^{\mathsf{H}},\mathsf{Sim}}(\mathcal{D}) \leq \mathsf{Adv}^{\text{indiff}}_{\mathsf{C}^{\mathsf{H}},\tilde{\mathsf{Sim}}}(\tilde{\mathcal{D}}) + \frac{\ell + \log \gamma^{-1}}{p} + \frac{p}{2^k} + \gamma \ .$$

Proof. The first part of the theorem follows directly from the discussion above that indifferentiability with backdoors and no adaptivity is stronger than indifferentiability with auxiliary input for independent preprocessing.

We now prove the second part of the theorem.

Game_0*:.* We start with the real game in the salted AI-indifferentiability game:

$$z \twoheadleftarrow \mathcal{D}_0(\mathsf{H});\ hk \twoheadleftarrow \{0,1\}^k;\ b \twoheadleftarrow \mathcal{D}_1^{\mathsf{C}^{\mathsf{H}(hk,\cdot)}(hk,\cdot),\mathsf{H}}(hk, z) \ .$$

Game_1*:.* We now move to the bit-fixing RO model

$$(z, A) \twoheadleftarrow \tilde{\mathcal{D}}_0();\ hk \twoheadleftarrow \{0,1\}^k;\ b \twoheadleftarrow \mathcal{D}_1^{\mathsf{C}^{\mathsf{H}[A](hk,\cdot)}(hk,\cdot),\mathsf{H}[A]}(hk, z) \ .$$

Here $\tilde{\mathcal{D}}_0$ runs $\mathcal{D}_0$ by simulating an H for it and then runs the decomposition algorithm to get a set of assignments A for p fixed points (for any $p \in \mathbb{N}$). We may now apply [7, Theorem 5] to deduce that for any $\gamma > 0$,

$$\Pr[\mathsf{Game}_1] - \Pr[\mathsf{Game}_0] \leq \frac{\ell + \log \gamma^{-1}}{p} + \gamma \ ,$$

where ℓ is the size of auxiliary information.

Game_2*:.* We now move to a setting where C uses H rather than $\mathsf{H}[A]$

$$(z, A) \twoheadleftarrow \tilde{\mathcal{D}}_0();\ hk \twoheadleftarrow \{0,1\}^k;\ b \twoheadleftarrow \mathcal{D}_1^{\mathsf{C}^{\mathsf{H}(hk,\cdot)}(hk,\cdot),\mathsf{H}[A]}(hk, z) \ .$$

This modification is justified by the fact that the probability that a uniform hk is (the prefix of the first component of some point) in A is at most $p/2^k$. We have that $\Pr[\mathsf{Game}_2] - \Pr[\mathsf{Game}_1] \leq p/2^k$.

Game_3∴. We now move to a world where $\mathcal{D}_1$ is replaced by a differentiator $\tilde{\mathcal{D}}_1$ that gets the list A and does not query H on points in A:

$$(z, A) \twoheadleftarrow \tilde{\mathcal{D}}_0(); \; hk \twoheadleftarrow \{0,1\}^k; \; b \twoheadleftarrow \tilde{\mathcal{D}}_1^{\mathsf{C}^{\mathsf{H}(hk,\cdot)}(hk,\cdot),\mathsf{H}}(hk, z, A) \; .$$

Here $\tilde{\mathcal{D}}_1(hk, z, A)$ runs $\mathcal{D}_1(hk, z)$ relaying its queries to the first oracle to its own first oracle and the second oracle queries to its own second oracle except when a queried point appears as a prefix of the first component of an entry in A in which case $\tilde{\mathcal{D}}_1$ uses A to answer the query. We have that $\Pr[\mathsf{Game}_3] - \Pr[\mathsf{Game}_2] = 0$.

Game_4∴. We now absorb $\tilde{\mathcal{D}}_0$ and $\tilde{\mathcal{D}}_1$ into a single differentiator $\tilde{\mathcal{D}}$:

$$b \twoheadleftarrow \tilde{\mathcal{D}}^{\mathsf{C}^{\mathsf{H}(hk,\cdot)}(hk,\cdot),\mathsf{H}} \; .$$

Here $\tilde{\mathcal{D}}$ simply runs $\tilde{\mathcal{D}}_0$, followed by picking $hk \twoheadleftarrow \{0,1\}^k$, and then running $\tilde{\mathcal{D}}_1$. We have that $\Pr[\mathsf{Game}_4] - \Pr[\mathsf{Game}_3] = 0$.

Game_5∴. We now use the standard indifferentiability of the construction to move to the world

$$b \twoheadleftarrow \tilde{\mathcal{D}}^{\mathsf{RO}(hk,\cdot),\tilde{\mathsf{Sim}}^{\mathsf{RO}}} \; ,$$

where $\tilde{\mathsf{Sim}}$ is an indifferentiability simulator. We have that $\Pr[\mathsf{Game}_3] - \Pr[\mathsf{Game}_2] \leq \mathsf{Adv}^{\mathrm{indiff}}_{\mathsf{C}^{\mathsf{H}},\tilde{\mathsf{Sim}}}(\tilde{\mathcal{D}})$.

Game_6∴. We now syntactically unroll $\tilde{\mathcal{D}}$ into $(\tilde{\mathcal{D}}_0, \tilde{\mathcal{D}}_1)$:

$$(z, A) \twoheadleftarrow \tilde{\mathcal{D}}_0(); \; hk \twoheadleftarrow \{0,1\}^k; \; b \twoheadleftarrow \tilde{\mathcal{D}}_1^{\mathsf{RO}(hk,\cdot),\tilde{\mathsf{Sim}}^{\mathsf{RO}}}(hk, z, A) \; .$$

We have that $\Pr[\mathsf{Game}_6] - \Pr[\mathsf{Game}_5] = 0$.

Game_7∴. We further unroll $\tilde{\mathcal{D}}_1$ into $\mathcal{D}_1$ and define $\mathsf{Sim}_1[A]$ to be $\tilde{\mathsf{Sim}}$ except that it uses A to answers queries in A:

$$(z, A) \twoheadleftarrow \tilde{\mathcal{D}}_0(); \; hk \twoheadleftarrow \{0,1\}^k; \; b \twoheadleftarrow \mathcal{D}_1^{\mathsf{RO}(hk,\cdot),\mathsf{Sim}_1^{\mathsf{RO}}[A]}(hk, z) \; .$$

We have that $\Pr[\mathsf{Game}_7] - \Pr[\mathsf{Game}_6] = 0$.

Game_8∴. Finally we define $\mathsf{Sim}_0 := \tilde{\mathcal{D}}_0$ and arrive at the simulated world

$$(z, A) \twoheadleftarrow \mathsf{Sim}_0(); \; hk \twoheadleftarrow \{0,1\}^k; \; b \twoheadleftarrow \mathcal{D}_1^{\mathsf{RO}(hk,\cdot),\mathsf{Sim}_1^{\mathsf{RO}}[A]}(hk, z) \; .$$

We have that $\Pr[\mathsf{Game}_8] - \Pr[\mathsf{Game}_7] = 0$.

The second part of theorem follows by summing the (in)equalities established above; that is for any $p \in \mathbb{N}$ and any $\gamma > 0$ we get that

$$\begin{aligned}\mathsf{Adv}^{\text{s-ai-indiff}}_{\mathsf{C}^{\mathsf{H}},(\mathsf{Sim}_0,\mathsf{Sim}_1)}(\mathcal{D}_0, \mathcal{D}_1) &= \Pr[\mathsf{Game}_0] - \Pr[\mathsf{Game}_8] \\ &\leq \mathsf{Adv}^{\mathrm{indiff}}_{\mathsf{C}^{\mathsf{H}},\tilde{\mathsf{Sim}}}(\tilde{\mathcal{D}}) + \frac{\ell + \log \gamma^{-1}}{p} + \frac{p}{2^k} + \gamma \; .\end{aligned}$$

□

We can instantiate the first part of the above theorem with the xor combiner, which gives us the following corollary.

Corollary 4. *The xor combiner* $\mathsf{C}_{\oplus}^{\mathsf{H}_1,\mathsf{H}_2}(x) := \mathsf{H}_1(x) \oplus \mathsf{H}_2(x)$ *is AI-indifferentiable from a random oracle with respect to independent preprocessing attacks for hash functions* $\mathsf{H}_1, \mathsf{H}_2 \in [M]^N$*. More precisely, for any* $p \in \mathbb{N}$ *and* $0 < \gamma < 1$*, there exists a simulator* $\mathsf{Sim}[p] := (\mathsf{Sim}_{0,1}[p], \mathsf{Sim}_{0,2}[p], \mathsf{Sim}_{1,1}, \mathsf{Sim}_{1,2})$ *with oracle access to* RO*, such that for any auxiliary-input differentiator* $\mathcal{D} := (\mathcal{D}_{0,1}, \mathcal{D}_{0,2}, \mathcal{D}_1)$ *with auxiliary input of size* ℓ *for each hash function, where* $\mathcal{D}_1$ *makes up to* q_{H} *evaluation queries to* H_1 *and* H_2 *as well as* q_{C} *construction queries, we have*

$$\mathsf{Adv}^{\text{ai-indiff}}_{\mathsf{C}_{\oplus}^{\mathsf{H}_1,\mathsf{H}_2},\mathsf{Sim}[p]}(\mathcal{D}) \leq 2\gamma + \frac{(q_{\mathsf{H}} + 2q_{\mathsf{C}}) \cdot (\ell + \log \gamma^{-1})}{p} \ ,$$

while the simulator places at most $q_{\mathsf{H}} + 2p$ *queries to the random oracle* RO*.*

Proof. The claim follows from the first part of Theorem 3 together with our indifferentiability result for xor (given in Theorem 1). However, deriving the concrete bounds using Corollary 1 results in somewhat suboptimal bounds with simulator query complexity $\mathcal{O}(p)$ and advantage $\mathcal{O}(1/\sqrt{p})$ with $p = N^{\alpha_1}$.

Here we directly use Theorem 1 for a simulator which fixes p points while simulating an ℓ-bit response of BD_1 and the same number of points while simulating an ℓ-bit response of BD_2. Note that in the auxiliary-input setting we only consider one query to each backdoor oracle and therefore we have $Q = 1$. Overall we will have a simulator $\mathsf{Sim}[p]$ for the above corollary, such that its off-line phase (i.e., $\mathsf{Sim}_{0,1}[p]$ and $\mathsf{Sim}_{0,2}[p]$) makes no queries to the RO and it simulates the auxiliary inputs by randomly choosing the hash functions and computing the output of the desired auxiliary-input functions (similar to a queried backdoor function) on them. This off-line phase then can use the refined decomposition algorithm of Lemma 1 for some small γ to come up with and (in addition to the auxiliary input) output two sets of pre-set points, each of size p, as its state. The state will be shared with the on-line phase of simulation, i.e., $\mathsf{Sim}_{1,1}$ and $\mathsf{Sim}_{1,2}$. Now this on-line simulator is a simple xor indifferentiability simulator which ensures consistency with the pre-set points. Note that our on-line simulator fixes p points for H_1 and again p points for H_2. This results in simulator query complexity of $q_{\mathsf{H}} + 2p$.

In this case, since $\delta_{-1} = \delta_0 = 0$ we obtain that

$$\delta_1 = \delta_2 = \frac{\ell + \log \gamma^{-1}}{p \log M} \ .$$

Plugging these back into the advantage bound in Theorem 1 we obtain the bound claimed above. □

Note that for $p = o(\sqrt{N})$ we get a bound that is meaningful for collision resistance. As a result, we get that the xor combiner is collision resistant in the presence of independent auxiliary input (with no-salting). We note that the xor construction comes with added advantage that its security goes beyond AI-indifferentiability, and is also more domain efficient. Strictly speaking, however, the two settings are incomparable as the form of auxiliary information changes.

Acknowledgments

Dodis was partially supported by gifts from VMware Labs, Facebook and Google, and NSF grants 1314568, 1619158, 1815546. Mazaheri was supported by the German Federal Ministry of Education and Research (BMBF) and by the Hessian State Ministry for Higher Education, Research and the Arts, within ATHENE. Tessaro was partially supported by NSF grants CNS-1930117 (CAREER), CNS-1926324, CNS-2026774, a Sloan Research Fellowship, and a JP Morgan Faculty Award.

References

1. E. Andreeva, A. Bogdanov, Y. Dodis, B. Mennink, and J. P. Steinberger. On the indifferentiability of key-alternating ciphers. In *CRYPTO 2013, Part I*, pages 531–550, 2013. (Cited on page 3.)
2. B. Barak, G. Kindler, R. Shaltiel, B. Sudakov, and A. Wigderson. Simulating independence: new constructions of condensers, ramsey graphs, dispersers, and extractors. In *37th ACM STOC*, pages 1–10, 2005. (Cited on page 31.)
3. B. Bauer, P. Farshim, and S. Mazaheri. Combiners for backdoored random oracles. In *CRYPTO 2018, Part II*, pages 272–302, 2018. (Cited on pages 2, 4, 6, and 7.)
4. G. Bertoni, J. Daemen, M. Peeters, and G. Van Assche. On the indifferentiability of the sponge construction. In *EUROCRYPT 2008*, pages 181–197, 2008. (Cited on page 3.)
5. R. Canetti. Universally composable security: A new paradigm for cryptographic protocols. In *42nd FOCS*, pages 136–145, 2001. (Cited on page 3.)
6. S. Coretti, Y. Dodis, and S. Guo. Non-uniform bounds in the random-permutation, ideal-cipher, and generic-group models. In *CRYPTO 2018, Part I*, pages 693–721, 2018. (Cited on pages 2 and 4.)
7. S. Coretti, Y. Dodis, S. Guo, and J. P. Steinberger. Random oracles and non-uniformity. In *EUROCRYPT 2018, Part I*, pages 227–258, 2018. (Cited on pages 2, 4, 9, 12, 15, and 36.)
8. J.-S. Coron, Y. Dodis, C. Malinaud, and P. Puniya. Merkle-Damgård revisited: How to construct a hash function. In *CRYPTO 2005*, pages 430–448, 2005. (Cited on page 3.)
9. J.-S. Coron, J. Patarin, and Y. Seurin. The random oracle model and the ideal cipher model are equivalent. In *CRYPTO 2008*, pages 1–20, 2008. (Cited on page 3.)
10. Y. Dodis, S. Guo, and J. Katz. Fixing cracks in the concrete: Random oracles with auxiliary input, revisited. In *EUROCRYPT 2017, Part II*, pages 473–495, 2017. (Cited on pages 2 and 4.)
11. Y. Dodis, L. Reyzin, and A. Smith. Fuzzy extractors: How to generate strong keys from biometrics and other noisy data. In *EUROCRYPT 2004*, pages 523–540, 2004. (Cited on page 12.)
12. Y. Dodis, M. Stam, J. P. Steinberger, and T. Liu. Indifferentiability of confusion-diffusion networks. In *EUROCRYPT 2016, Part II*, pages 679–704, 2016. (Cited on page 3.)
13. M. Fischlin, C. Janson, and S. Mazaheri. Backdoored hash functions: Immunizing HMAC and HKDF. In *CSF 2018*, pages 105–118, 2018. (Cited on page 2.)

14. M. Göös, S. Lovett, R. Meka, T. Watson, and D. Zuckerman. Rectangles are nonnegative juntas. In *47th ACM STOC*, pages 257–266, 2015. (Cited on pages 4, 8, and 10.)
15. J. J. Hoch and A. Shamir. On the strength of the concatenated hash combiner when all the hash functions are weak. In *ICALP 2008, Part II*, pages 616–630, 2008. (Cited on page 2.)
16. T. Holenstein, R. Künzler, and S. Tessaro. The equivalence of the random oracle model and the ideal cipher model, revisited. In *43rd ACM STOC*, pages 89–98, 2011. (Cited on page 3.)
17. A. Joux. Multicollisions in iterated hash functions. Application to cascaded constructions. In *CRYPTO 2004*, pages 306–316, 2004. (Cited on page 34.)
18. P. K. Kothari, R. Meka, and P. Raghavendra. Approximating rectangles by juntas and weakly-exponential lower bounds for LP relaxations of CSPs. In *49th ACM STOC*, pages 590–603, 2017. (Cited on pages 9 and 10.)
19. C. Lee, C. Lu, S. Tsai, and W. Tzeng. Extracting randomness from multiple independent sources. *IEEE Trans. Information Theory*, 51(6):2224–2227, 2005. (Cited on page 31.)
20. G. Leurent and T. Peyrin. From collisions to chosen-prefix collisions application to full SHA-1. In *EUROCRYPT 2019, Part III*, pages 527–555, 2019. (Cited on page 2.)
21. X. Li. Three-source extractors for polylogarithmic min-entropy. In *56th FOCS*, pages 863–882, 2015. (Cited on page 31.)
22. M. Liskov. Constructing an ideal hash function from weak ideal compression functions. In *SAC 2006*, pages 358–375, 2007. (Cited on page 2.)
23. U. M. Maurer, R. Renner, and C. Holenstein. Indifferentiability, impossibility results on reductions, and applications to the random oracle methodology. In *TCC 2004*, pages 21–39, 2004. (Cited on pages 3 and 7.)
24. B. Pfitzmann and M. Waidner. A model for asynchronous reactive systems and its application to secure message transmission. In *2001 IEEE Symposium on Security and Privacy*, pages 184–200, 2001. (Cited on page 3.)
25. R. Raz. Extractors with weak random seeds. In *37th ACM STOC*, pages 11–20, 2005. (Cited on page 31.)
26. M. Stevens, E. Bursztein, P. Karpman, A. Albertini, and Y. Markov. The first collision for full SHA-1. In *CRYPTO 2017, Part I*, pages 570–596, 2017. (Cited on page 2.)
27. D. Unruh. Random oracles and auxiliary input. In *CRYPTO 2007*, pages 205–223, 2007. (Cited on pages 2, 4, and 9.)