Domain Extension of Public Random Functions: Beyond the Birthday Barrier*

Ueli Maurer Stefano Tessaro

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

Abstract

A *public* random function is a random function that is accessible by all parties, including the adversary. For example, a (public) random oracle is a public random function $\{0,1\}^* \to \{0,1\}^n$. The natural problem of constructing a public random oracle from a public random function $\{0,1\}^m \to \{0,1\}^n$ (for some $m > n$) was first considered at Crypto 2005 by Coron et al. who proved the security of variants of the Merkle-Damgård construction against adversaries issuing up to $\mathcal{O}(2^{n/2})$ queries to the construction and to the underlying compression function. This bound is less than the square root of $n2^m$, the number of random bits contained in the underlying random function.

In this paper, we investigate domain extenders for public random functions approaching optimal security. In particular, for all $\epsilon \in (0,1)$ and all functions m and ℓ (polynomial in n), we provide a construction $\mathbf{C}_{\epsilon,m,\ell}(\cdot)$ which extends a public random function $\mathbf{R} : \{0,1\}^n \to \{0,1\}^n$ to a function $\mathbf{C}_{\epsilon,m,\ell}(\mathbf{R}) : \{0,1\}^{m(n)} \to \{0,1\}^{\ell(n)}$ with time-complexity polynomial in n and $1/\epsilon$ and which is secure against adversaries which make up to $\Theta(2^{n(1-\epsilon)})$ queries. A central tool for achieving high security are special classes of unbalanced bipartite expander graphs with small degree. The achievability of practical (as opposed to complexity-theoretic) efficiency is proved by a non-constructive existence proof.

Combined with the iterated constructions of Coron et al., our result leads to the first iterated construction of a hash function $\{0,1\}^* \to \{0,1\}^n$ from a component function $\{0,1\}^n \to \{0,1\}^n$ that withstands all recently proposed generic attacks against iterated hash functions, like Joux's multi-collision attack, Kelsey and Schneier's second-preimage attack, and Kelsey and Kohno's herding attacks.

1 Introduction

1.1 Secret vs. Public Random Functions

Primitives that provide some form of randomness are of central importance in cryptography, both as a primitive assumed to be given (e.g. a secret key), and as a primitive constructed from a weaker one to "behave like" a certain ideal random primitive (e.g. a random function), according to some security notion.

*An extended abstract of this paper appears in the proceedings of CRYPTO 2007. This is the full version.

An adversary may have different types of access to a random primitive. The two extreme cases are that the adversary has *no access* and that he has *complete access*[1] to it. For example, the adversary is assumed to have no access to a secret key, and a pseudo-random function (PRF) is a (computationally-secure) realization from such a secret key of a secret random function to which the adversary has no access. In contrast, a (public) random oracle, as used in the so-called random-oracle model [7], is a function $\{0,1\}^* \to \{0,1\}^n$ to which the adversary has *complete* access, like the legitimate parties. Similarly, a *public parameter* (e.g. the parameter selecting a hash function from a class) is a finite random string to which the adversary has complete access. It is natural to also consider finite-domain public random functions.

In this paper we are interested in such *public* random primitives and reductions among them. The question whether (and how) a certain primitive can be securely realized from another primitive is substantially more complex in the public setting, compared to the secret setting, and even the security notion is more involved. For example, while the CBC-construction can be seen as the secure realization of a secret random function $\{0,1\}^* \to \{0,1\}^n$ from a secret random function $\{0,1\}^n \to \{0,1\}^n$ [5, 21], the same statement is false if public functions (accessible to the adversary) are considered. Another famous example of a reduction problem for public primitives is the realization of a (public) random oracle from a public parameter. This was shown to be impossible [9, 23].

1.2 Domain Extension and the Birthday Barrier

A random primitive (both secret or public) can be characterized by the number of random bits it contains. An ℓ-bit key is a string (or table) containing ℓ random bits, a random function $\{0,1\}^m \to \{0,1\}^n$ corresponds to a table of $n2^m$ random bits which can be accessed efficiently, and a random oracle $\{0,1\}^* \to \{0,1\}^n$ corresponds to a countably infinite table of random bits.[2] Of course, a random table of N bits can be interpreted as a random function $\{0,1\}^m \to \{0,1\}^n$ for any m and n with $n2^m \leq N$. For example, n can be doubled at the apparently minor expense of reducing m by 1.

An important topic in cryptography is the secure expansion of such a table, considered as an ideal system. This is referred to as *domain extension*, say from $\{0,1\}^m$ to $\{0,1\}^{2m}$ (or to $\{0,1\}^*$), which corresponds to an exponential (or even infinite) blow-up of the table size. (In contrast, increasing the range, say from $\{0,1\}^n$ to $\{0,1\}^{2n}$, corresponds to merely a doubling of the table size.)

In [23] a generalization of indistinguishability to systems with public access, called *indifferentiability*, was proposed. Like for indistinguishability, there is a computational and a stronger, information-theoretic, version of indifferentiability. This general notion allows to discuss the secure realization of a *public* random primitive from another public random primitive. In [23] also a simple general framework was proposed, based on entropy arguments, for proving impossibility results like that of [9]. One can easily show that not even a single-bit extension of a public parameter, from ℓ to $\ell+1$ bits, is possible, let alone to an exponentially large table (corresponding to a public random function $\{0,1\}^m \to \{0,1\}^n$) or even to an infinite table (corresponding to the impossibility of realizing a random oracle [9, 23]).

[1] Side-channel attack analyses, where part of the secret key is assumed to leak, are examples of intermediate scenarios.

[2] Each bit can be accessed in time logarithmic in its position in the table, which is optimal since the specification of the position requires logarithmically many bits. In this paper we only consider such random primitives where the bits can be accessed efficiently, but there are also more complicated primitives, like an ideal cipher, which on one hand has a special permutation structure and also allows on the other hand a special additional type of access, namely inverse queries.

However, the situation is different if one starts from a public random function (as opposed to just a public random string). Coron et al. [13] considered the problem of constructing a random oracle $\{0,1\}^* \to \{0,1\}^n$ from a public random function $\{0,1\}^m \to \{0,1\}^n$ (where $m > n$) and showed that a modified Merkle-Damgård construction [25, 14] works, with information-theoretic security (i.e., indifferentiability) up to about $\mathcal{O}(2^{n/2})$ queries. This bound, only the square root of $\mathcal{O}(2^n)$, is usually called the "birthday barrier". The term "birthday" is used because the birthday paradox applies (as soon as two different inputs to the function occur which produce the same output, security is lost) and the term "barrier" is used because breaking it is non-trivial if at all possible.

For *secret* random functions, many constructions in the literature, also those based on universal hashing [11, 30] and the CBC-construction [5, 21], suffer from the birthday problem, and hence several researchers [1, 4, 21] considered the problem of achieving security beyond the birthday barrier. The goal of this paper is to solve the corresponding problem for public random functions. Namely, we want to achieve essentially maximal security, i.e., up to $\Theta(2^{n(1-\epsilon)})$ queries for any $\epsilon > 0$ (where the construction may depend on ϵ). Like for other problems (see e.g. [15]), going from the "secret case" to the "public case" appears to involve substantial new construction elements and analysis techniques.

1.3 Significance of Domain Extension for Public Random Functions

The domain extension problem for public random functions has important implications for the design of cryptographic functions, in addition to being of general theoretical interest. We also refer to [13] for a discussion of the significance of this problem.

Cryptographic functions with arbitrary input-length are of crucial importance in cryptography. Desirable properties for such functions are collision-resistance, second-preimage resistance, multi-collision resistance, being pseudo-random, or being a secure MAC, etc. A general paradigm for constructing a cryptographic function $\{0,1\}^* \to \{0,1\}^n$, both in the secret and the public case, is to make use of a component function $\mathbf{F} : \{0,1\}^m \to \{0,1\}^n$ and to embed it into an iterated construction $\mathbf{C}(\cdot)$ (e.g. the CBC or the Merkle-Damgård construction), resulting in the overall function $\mathbf{C}(\mathbf{F}) : \{0,1\}^* \to \{0,1\}^n$.

It is important to be able to separate the reasoning about the component function $\mathbf{F}$ and about the construction $\mathbf{C}(\cdot)$. Typically, $\mathbf{F}$ is simply assumed to have some property, like being collision-resistant, second-preimage resistant, a secure MAC, etc. In contrast, the construction $\mathbf{C}(\cdot)$ is (or should be!) designed in a way that one can *prove* certain properties.

There are two types of such proofs for $\mathbf{C}(\cdot)$. The first type is a complexity-theoretic reduction proof showing that if there exists an adversary breaking a certain property of $\mathbf{C}(\mathbf{F})$, then there exists a comparably efficient adversary breaking a property (the same or a different one) of $\mathbf{F}$. For example, using such an argument one can prove that the Merkle-Damgård [25, 14] construction is collision-resistant if the component function is. Similarly, one can prove that the CBC construction is a PRF if the component function is [5], or that certain constructions [2, 24] are secure MACs if the component function is.

A second type of proof, which is the subject of [13] and of this paper, is the proof that if $\mathbf{F}$ is a public random function, then so is $\mathbf{C}(\mathbf{F})$, up to a certain number B of queries. Such a proof implies the absence of a generic (black-box) attack against $\mathbf{C}(\mathbf{F})$, i.e., an attack which does not exploit specific properties of $\mathbf{F}$, but uses it merely as a black-box.[3] Such a generic proof is not an ultimate security proof for $\mathbf{C}(\mathbf{F})$, but it proves that the construction $\mathbf{C}(\cdot)$ itself has no

[3] This is analogous to security proofs in the generic group model [31, 22] which show that no attack exists that does not exploit the particular representation of group elements.

weakness. A main advantage of such a proof is that it applies to *every* cryptographic property of interest (which a random function has), not just to specific properties like collision-resistance.

The number B of queries up to which security is guaranteed is a crucial parameter of such a proof, especially in view of several surprises of the past years regarding weaknesses of iterated constructions. Joux [17] showed that the security of the Merkle-Damgård construction (with compression function with n-bit output) against finding multi-collisions is not much higher than the security against normal collision attacks, namely the birthday barrier $\mathcal{O}(2^{n/2})$, which is surprising because for a random function, finding an r-multi-collision requires $\Theta(2^{\frac{r-1}{r}n})$ queries. Joux's attack has been generalized to a wider class of constructions [16]. Other attacks in a similar spirit against iterated constructions are the second-preimage attack by Kelsey and Schneier [19], and herding attacks [18]. One possibility to overcome these issues is to rely on a compression function with input domain much larger than the size of the output of the construction (cf. for example the constructions in [20] and the double block-length construction of [12]), but this does not seem to be the best possible approach, both from a theoretical and from a practical viewpoint, as explained below.

A proof, like that of [13], for a construction $\mathbf{C}(\cdot)$ of a public random function, implies that $\mathbf{C}(\cdot)$ is secure against all possible attacks, up to the bound B on the number of queries stated in the proof. Since the bound in [13] is the birthday barrier, this implies nothing (beyond the birthday barrier) for attacks that require more queries, like the attacks of [17, 19, 12] mentioned above, and indeed the constructions of [13] also suffer from the same attacks.

The bound B is also of importance since it determines the input and output sizes of $\mathbf{F}$. For example, because collision-resistance is a property that can hold only up to $2^{n/2}$ queries (due to the birthday paradox), n must be chosen twice as large as one might expect to be feasible in a naïve security analysis. Moreover, since the function must be compressing to be useful in a construction $\mathbf{C}(\cdot)$, the input size m must be larger than the output size n. However, if collision-resistance is not required, but instead for example second-preimage resistance, then the input size m of $\mathbf{F}$ can potentially be smaller or, turning the argument around, security for a given m can be much higher.

The input size m of $\mathbf{F}$ is relevant for two more reasons. First, if one considers the (perhaps not very realistic) possibility of finding a random function in Nature (say, by scanning the surface of the moon or by appropriately accessing the WWW), then m is a crucial parameter since the table size $n2^m$ is exponential in m. Second, for a given maximal computing time for $\mathbf{F}$, the difficulty of designing a concrete cryptographic function $\mathbf{F} : \{0,1\}^m \to \{0,1\}^n$ that is supposed to "look random" increases significantly if m is large. This can be seen as follows. Such a function $\mathbf{F}$ for large m could be modified in many different ways to reduce m to $m' < m$ (e.g. set $m - m'$ input bits to 0 or to any fixed value, or repeat an input of size m' until a block of length m is filled, etc.), and for each of these modifications it would still have to be secure.[4] Hence simply designing a new function with doubled m is not a very reasonable solution for the birthday barrier problem. Rather, one should find a construction that doubles (or multiplies) the input size but at the same time preserves the security almost optimally.

1.4 Contributions and Outline of This Paper

The main contribution of this paper is a construction paradigm for breaking the birthday barrier for domain extension of public random functions. More precisely, in Section 3 we prove that for every $\epsilon \in (0,1)$, m and ℓ, there exists an efficient construction $\mathbf{C}_{\epsilon,m,\ell}(\cdot)$ which extends a public

[4] This argument applies even though we know that a public random function is not securely realizable from a public random parameter.

random function $\{0,1\}^n \to \{0,1\}^n$ to a public random function $\{0,1\}^m \to \{0,1\}^\ell$, and which guarantees security for up to $\Theta(2^{n(1-\epsilon)})$ queries.

A central tool in our approach is a new combinatorial object, which we call an *input-restricting function family*. Section 4 discusses constructions of such families from highly-unbalanced bipartite expander graphs. While current expander constructions only allow our paradigm to be efficient in a complexity-theoretic sense (i.e. polynomial-time), an existence proof shows that very efficient constructions exist which would be of real practical interest. We hope this to provide additional motivation to investigate explicit constructions of unbalanced bipartite expanders for a range of parameters which have not received much attention so far.

Finally, our techniques allow to use a public random function $\{0,1\}^n \to \{0,1\}^n$ to construct a compression function with sufficiently large domain and range and to plug it into the construction of [13] to achieve the first iterated construction of a public random oracle $\{0,1\}^* \to \{0,1\}^n$ from a public random function $\{0,1\}^n \to \{0,1\}^n$ with security above the birthday barrier. We discuss this in Section 5.

2 Preliminaries

2.1 Notation and Probabilities

Throughout this paper, calligraphic letters (e.g. $\mathcal{U}$) denote sets. Furthermore, the set $\mathcal{U}^k$ contains all k-tuples of elements from $\mathcal{U}$, and a k-tuple is denoted as $u^k = [u_1, \ldots, u_k]$. We use capital letters (e.g. U) to name random variables, whereas their concrete values are denoted by the corresponding lower-case letters (e.g. u). Also, we write P_U for the probability distribution of U, and we use the shorthand $\mathsf{P}_U(u)$ for $\mathsf{P}(U = u)$ and for some event $\mathcal{A}$ we write $\mathsf{P}_{\mathcal{A}U}(u)$ instead of $\mathsf{P}(\mathcal{A} \wedge U = u)$. Given events $\mathcal{A}$ and $\mathcal{B}$ and random variables U and V, then $\mathsf{P}_{\mathcal{A}U|\mathcal{B}V}$ denotes the corresponding conditional probability distribution, which is interpreted as a function $\mathcal{U} \times \mathcal{V} \to \mathbb{R}_{\geq 0}$, where the value $\mathsf{P}_{\mathcal{A}U|\mathcal{B}V}(u,v)$ is well-defined for all $u \in \mathcal{U}$ and $v \in \mathcal{V}$ such that $\mathsf{P}_{\mathcal{B}V}(v) > 0$ (and undefined otherwise). Two probability distributions P_U and $\mathsf{P}_{U'}$ on the same set $\mathcal{U}$ are equal, denoted $\mathsf{P}_U = \mathsf{P}_{U'}$, if $\mathsf{P}_U(u) = \mathsf{P}_{U'}(u)$ for all $u \in \mathcal{U}$. Also, for conditional probability distributions, equality holds if it holds for all inputs for which *both* are defined. We often need to deal with distinct random experiments where equally-named random variables and/or events appear. To avoid confusion, we add superscripts to probability distributions (e.g. $\mathsf{P}^{\mathcal{E}}_{\mathcal{A}U|\mathcal{B}V}(u,v)$) to make the random experiment explicit. Also, note that sometimes we simply write $\mathsf{P}^{\mathcal{E}}_{\mathcal{A}U|\mathcal{B}V}$ whenever the arguments u, v are clear from the context (or when the statement holds for any argument).

For binary strings $s, s' \in \{0,1\}^*$, we denote by $s\|s'$ their concatenation. Furthermore, we often use strings $s \in \{0,1\}^{tn}$ whose length $|s|$ is a multiple of n. In this case, the string $s^{(i)}$ the i'th n-bit block of the string s. Also, for a binary string $s \in \{0,1\}^m$ and $n \leq m$, the string $s|_n$ consists of the first n bits of s.

2.2 Indistinguishability of Random Systems

In this section, we review basic definitions and facts from the framework of *random systems* of [21]. A random system is the abstraction of the input-output behavior of a discrete system.

Definition 1. An $(\mathcal{X}, \mathcal{Y})$-*random system* $\mathbf{F}$ is a (generally infinite) sequence of conditional probability distributions $\mathsf{p}^{\mathbf{F}}_{Y_i|X^iY^{i-1}}$ for all $i \geq 1$. Two random systems $\mathbf{F}$ and $\mathbf{G}$ are *equivalent*, denoted $\mathbf{F} \equiv \mathbf{G}$, if $\mathsf{p}^{\mathbf{F}}_{Y_i|X^iY^{i-1}} = \mathsf{p}^{\mathbf{G}}_{Y_i|X^iY^{i-1}}$ for all $i \geq 1$.

That is, the system is described by the conditional probabilities $\mathsf{p}^{\mathbf{F}}_{Y_i|X^iY^{i-1}}(y_i, x^i, y^{i-1})$ (for $i \geq 1$) of obtaining the output $y_i \in \mathcal{Y}$ on query $x_i \in \mathcal{X}$ given the previous $i-1$ queries $x^{i-1} = [x_1, \ldots, x_{i-1}] \in \mathcal{X}^{i-1}$ and their corresponding outputs $y^{i-1} = [y_1, \ldots, y_{i-1}] \in \mathcal{Y}^{i-1}$. We use a lower-case p to stress the fact that these conditional distributions by themselves do not define a random experiment. Equivalently, one can describe the system by the conditional distributions $\mathsf{p}^{\mathbf{F}}_{Y^i|X^i}$ (for all $i \geq 1$) of the first i outputs, given the first i inputs. Both views are related by the equality $\mathsf{p}^{\mathbf{F}}_{Y^i|X^i} = \prod_{j=1}^{i} \mathsf{p}^{\mathbf{F}}_{Y_j|X^jY^{j-1}}$, and it is easy to see that $\mathbf{F}$ and $\mathbf{G}$ are equivalent if and only if $\mathsf{p}^{\mathbf{F}}_{Y^i|X^i} = \mathsf{p}^{\mathbf{G}}_{Y^i|X^i}$ for all $i \geq 1$. An example of a random system that we consider in the following is a *random function* $\mathbf{R} : \{0,1\}^m \to \{0,1\}^n$, which returns for every distinct input value $x \in \{0,1\}^m$ an independent and uniformly-distributed n-bit value. Moreover, a *random oracle* $\mathbf{O} : \{0,1\}^* \to \{0,1\}^n$ is a random function taking inputs of arbitrary length.

A *distinguisher* $\mathbf{D}$ for an $(\mathcal{X}, \mathcal{Y})$-random system is a $(\mathcal{Y}, \mathcal{X})$-random system which is one query ahead, i.e. it is defined by the conditional probability distributions $\mathsf{p}^{\mathbf{D}}_{X_i|X^{i-1}Y^{i-1}}$ for all $i \geq 1$. In particular, $\mathsf{p}^{\mathbf{D}}_{X_1}$ is the probability distribution of the first value queried by $\mathbf{D}$. Finally, the distinguisher outputs a bit after a certain number (say k) of queries depending on the transcript (X^k, Y^k). For an $(\mathcal{X}, \mathcal{Y})$-random system $\mathbf{F}$ and a distinguisher $\mathbf{D}$, we denote by $\mathbf{D} \circ \mathbf{F}$ the random experiment[5] where $\mathbf{D}$ interacts with $\mathbf{F}$. Furthermore, given an additional $(\mathcal{X}, \mathcal{Y})$-random system $\mathbf{G}$, the *distinguishing advantage* of $\mathbf{D}$ in distinguishing systems $\mathbf{F}$ and $\mathbf{G}$ is defined as $\Delta^{\mathbf{D}}(\mathbf{F}, \mathbf{G}) := \left|\mathsf{P}^{\mathbf{D}\circ\mathbf{F}}(1) - \mathsf{P}^{\mathbf{D}\circ\mathbf{G}}(1)\right|$, where $\mathsf{P}^{\mathbf{D}\circ\mathbf{F}}(1)$ and $\mathsf{P}^{\mathbf{D}\circ\mathbf{G}}(1)$ denote the probabilities that $\mathbf{D}$ outputs 1 after its k queries when interacting with $\mathbf{F}$ and $\mathbf{G}$, respectively.

We are interested in considering an internal *monotone condition* defined on a random system $\mathbf{F}$. Such a condition is initially true, and once it fails, it cannot become true any more. In particular, a *system $\mathbf{F}^{\mathcal{A}}$ with a monotone condition* $\mathcal{A}$ is an $(\mathcal{X}, \mathcal{Y}\times\{0,1\})$-random system, where the additional output bit indicates whether the condition $\mathcal{A}$ holds after the i'th query has been answered. In general, we characterize such a condition by a sequence of events $\mathcal{A} = A_0, A_1, \ldots$, where A_0 always holds, and A_i holds if the condition holds after query i. The condition *fails* at query i if $A_{i-1} \wedge \overline{A_i}$ occurs. For a system with a monotone condition $\mathbf{F}^{\mathcal{A}}$, we write $\mathbf{F}$ for the system where the additional output bit is ignored. Generally, we are interested in considering the behavior of systems only as long as a certain monotone condition holds: Given two systems $\mathbf{F}^{\mathcal{A}}$ and $\mathbf{G}^{\mathcal{B}}$ with monotone conditions $\mathcal{A}$ and $\mathcal{B}$, respectively, they are *equivalent*, denoted $\mathbf{F}^{\mathcal{A}} \equiv \mathbf{G}^{\mathcal{B}}$, if $\mathsf{p}^{\mathbf{F}}_{A_iY_i|X^iY^{i-1}A_{i-1}} = \mathsf{p}^{\mathbf{G}}_{B_iY_i|X^iY^{i-1}B_{i-1}}$ holds for all $i \geq 1$, or equivalently, if $\mathsf{p}^{\mathbf{F}}_{Y^iA_i|X^i} = \mathsf{p}^{\mathbf{G}}_{Y^iA_i|X^i}$ holds for all $i \geq 1$.

The probability that a distinguisher $\mathbf{D}$ issuing k queries makes a monotone condition $\mathcal{A}$ fail in the random experiment $\mathbf{D} \circ \mathbf{F}$ is defined as $\nu^{\mathbf{D}}(\mathbf{F}^{\mathcal{A}}) := \mathsf{P}^{\mathbf{D}\circ\mathbf{F}}_{\overline{A_k}}$. The following lemma from [21] relates this probability with the distinguishing advantage.

Lemma 1. *If $\mathbf{F}^{\mathcal{A}} \equiv \mathbf{G}^{\mathcal{B}}$ holds, then $\Delta^{\mathbf{D}}(\mathbf{F}, \mathbf{G}) \leq \nu^{\mathbf{D}}(\mathbf{F}^{\mathcal{A}}) = \nu^{\mathbf{D}}(\mathbf{G}^{\mathcal{B}})$ for all distinguishers $\mathbf{D}$.*

One can use a random system $\mathbf{F}$ as a component of a larger system: In particular, we are interested in *constructions* $\mathbf{C}(\cdot)$ such that the resulting random system $\mathbf{C}(\mathbf{F})$ invokes $\mathbf{F}$ as a subsystem. (Note that $\mathbf{C}(\cdot)$ itself is not a random system, while $\mathbf{C}(\mathbf{F})$ is a random system.)

Finally, we remark that in general when we mention that a construction (or a distinguisher) is *efficient* we mean that there exists a probabilistic interactive Turing machine implementing the same input-output behavior and with polynomial running time (in the understood security parameter).

[5] In particular, in this random experiment, the joint distribution $\mathsf{P}^{\mathbf{D}\circ\mathbf{F}}_{X^kY^k}$ is well-defined as $\prod_{i=1}^{k} \mathsf{p}^{\mathbf{D}}_{X_i|X^{i-1}Y^{i-1}} \cdot \mathsf{p}^{\mathbf{F}}_{Y_i|X^iY^{i-1}}$.

2.3 Indifferentiability, Reductions, and Public Random Primitives

The notion of *indifferentiability* [23] naturally extends the concept of indistinguishability to systems with a *public* and a *private* interface[6] adopting a simulation-based approach, in the same spirit as the security frameworks of [8, 29]. The public interface can be used by all parties, including the adversary, whereas the legitimate parties have exclusive access to the private interface. Generally, we denote such a system as an ordered pair $\mathbf{F} = [\mathbf{F}_{\text{pub}}, \mathbf{F}_{\text{priv}}]$. Furthermore, given constructions $\mathbf{S}(\cdot)$ and $\mathbf{C}(\cdot)$ leaving, respectively, private and public queries unmodified, we simply write $\mathbf{S}(\mathbf{F}) = [\mathbf{S}(\mathbf{F}_{\text{pub}}), \mathbf{F}_{\text{priv}}]$ and $\mathbf{C}(\mathbf{F}) = [\mathbf{F}_{\text{pub}}, \mathbf{C}(\mathbf{F}_{\text{priv}})]$.

Public random primitives are a special case of such systems. A *public random function (puRF)* $\mathbf{R} : \{0,1\}^m \to \{0,1\}^n$ is a system with a public and a private interface which behaves as the *same* random function at *both* interfaces.[7] In particular, both interfaces answer consistently. Furthermore, a *public random oracle (puRO)* $\mathbf{O} : \{0,1\}^* \to \{0,1\}^n$ is a public random function which takes inputs of arbitrary bit-length.

In the following definition, we refine the notion of (information-theoretic) indifferentiability from [23] to deal with concrete parameters.

Definition 2. Let $\alpha : \mathbb{N} \to \mathbb{R}_{\geq 0}$ and $\sigma : \mathbb{N} \to \mathbb{N}$ be functions. We say that a system $\mathbf{F}$ is (α, σ)-*indifferentiable* from $\mathbf{G}$, denoted $\mathbf{F} \overset{\alpha,\sigma}{\sqsubset} \mathbf{G}$, if there exists a simulator $\mathbf{S}$ such that $\Delta^{\mathbf{D}}([\mathbf{F}_{\text{pub}}, \mathbf{F}_{\text{priv}}], [\mathbf{S}(\mathbf{G}_{\text{pub}}), \mathbf{G}_{\text{priv}}]) \leq \alpha(k)$ for all distinguishers $\mathbf{D}$ making at most k queries, and $\mathbf{S}$ makes at most $\sigma(k)$ queries to $\mathbf{G}_{\text{pub}}$ when interacting with $\mathbf{D}$,.

The purpose of the simulator is to mimic $\mathbf{F}_{\text{pub}}$ by querying $\mathbf{G}_{\text{pub}}$, but without seeing the queries made to $\mathbf{G}_{\text{priv}}$. Indifferentiability directly implies a notion of reducibility.

Definition 3. A system $\mathbf{G}$ is (α, σ)-*reducible* to a system $\mathbf{F}$ if there exists an *efficient*, *deterministic*, and *stateless* construction $\mathbf{C}(\cdot)$ such that $[\mathbf{F}_{\text{pub}}, \mathbf{C}(\mathbf{F}_{\text{priv}})] \overset{\alpha,\sigma}{\sqsubset} \mathbf{G}$. The construction $\mathbf{C}(\cdot)$ is called an (α, σ)-*reduction*.

In Appendix A, we shortly discuss the achievable parameters for reducibility of public random primitives. The following lemma states that reducibility is transitive. We omit its simple proof.

Lemma 2. *Let* $\mathbf{E}, \mathbf{F}$, *and* $\mathbf{G}$ *be systems. If* $\mathbf{C}(\cdot)$ *is a* (α, σ)*-reduction of* $\mathbf{F}$ *to* $\mathbf{E}$, *and* $\mathbf{C}'(\cdot)$ *is an* (α', σ') *reduction of* $\mathbf{G}$ *to* $\mathbf{F}$ *that makes at most* $k_{\mathbf{C}'}(k)$ *queries to* $\mathbf{F}_{priv}$ *when queried* k *times, then* $\mathbf{C}'(\mathbf{C}(\cdot))$ *is an* $(\overline{\alpha}, \overline{\sigma})$*-reduction of* $\mathbf{G}$ *to* $\mathbf{E}$, *where* $\overline{\alpha}(k) = \alpha(k + k_{\mathbf{C}'}(k)) + \alpha'(k + \sigma(k))$ *and* $\overline{\sigma}(k) = \sigma'(\sigma(k))$.

The computational variant of indifferentiability is obtained by requiring $\mathbf{S}$ to be efficient and the advantage $\Delta^{\mathbf{D}}([\mathbf{F}_{\text{pub}}, \mathbf{F}_{\text{priv}}], [\mathbf{S}(\mathbf{G}_{\text{pub}}), \mathbf{G}_{\text{priv}}])$ to be negligible for all efficient $\mathbf{D}$. *Computational reducibility* is defined accordingly. In the information theoretic case, it is sometimes desirable to prove that the simulator is efficient when queried by an efficient distinguisher, as this then implies the corresponding complexity-theoretic statement. We refer the reader to [23, 13] for the implications of computational indifferentiability.

In contrast, as long as we are only interested in excluding generic attacks against security properties of a random function, the running time of the simulator is irrelevant. If $\mathbf{C}(\cdot)$ is an (α, σ)-reduction of a puRO $\mathbf{O} : \{0,1\}^* \to \{0,1\}^n$ (or of a puRF $\mathbf{R}' : \{0,1\}^m \to \{0,1\}^\ell$) to a puRF $\mathbf{R} : \{0,1\}^n \to \{0,1\}^n$, then $\mathbf{C}(\mathbf{R})$ inherits *all* the security properties of the truly-random

[6] Formally, this can be seen as a random system with a single interface and two types of queries.

[7] For this reason, we generally write both $\mathbf{R}_{\text{pub}}$ and $\mathbf{R}_{\text{priv}}$ as $\mathbf{R}$.

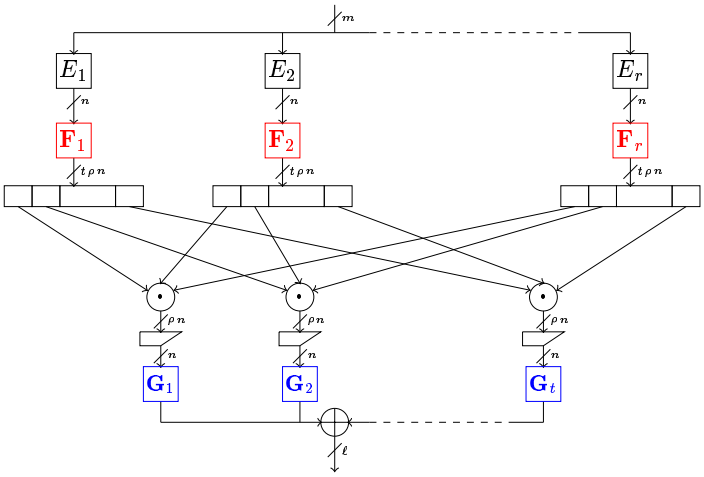

Figure 1: Main construction, where $\mathbf{F}_1, \ldots, \mathbf{F}_r$ and $\mathbf{G}_1, \ldots, \mathbf{G}_t$ are independent puRF's and $E_1, \ldots, E_r : \{0,1\}^m \to \{0,1\}^n$ are efficiently-computable functions.

oracle $\mathbf{O}$ (or of $\mathbf{R}'$), as long as the number of queries keeps $\alpha(k)$ small: Any adversary A making k queries (to both $\mathbf{R}$ and $\mathbf{C}(\mathbf{R})$) and breaking some property of $\mathbf{C}(\mathbf{R})$ with probability $\pi(k)$ can be transformed (combining it with the simulator) into an adversary A' making at most $k+\sigma(k)$ queries to $\mathbf{O}$ and breaking the same property for $\mathbf{O}$ with probability at least $\pi(k) - \alpha(k)$, and if no such A' can exist, then also no adversary A exists. The actual running time of A' is irrelevant, as the security of a random function (or oracle) with respect to a certain property is determined by the number of queries of the adversary, and not by its running time.

For example, if $\sigma(k) = \Theta(k)$, then, given a random element $s \in \{0,1\}^m$, no adversary can find a second preimage $s' \in \{0,1\}^m$ with $s' \neq s$ and $\mathbf{C}(\mathbf{R})(s) = \mathbf{C}(\mathbf{R})(s')$ with probability higher than $\Theta(k \cdot 2^{-n}) + \alpha(k)$.

3 Beyond-Birthday Domain Extension for Public Random Functions

3.1 The Construction

We first discuss at an abstract level the main construction of this paper (represented in Figure 1), which implements a function mapping m-bit strings to ℓ-bit strings from $r+t$ independent puRF's $\mathbf{F}_1, \ldots, \mathbf{F}_r : \{0,1\}^n \to \{0,1\}^{t\rho n}$ and $\mathbf{G}_1, \ldots, \mathbf{G}_t : \{0,1\}^n \to \{0,1\}^\ell$ (for given parameters r, t, and ρ). Let $E_1, \ldots, E_r : \{0,1\}^m \to \{0,1\}^n$ be efficiently-computable functions (to be instantiated below). On input $s \in \{0,1\}^m$, the construction operates in three stages:

1. The values $\mathbf{F}_p(E_p(s)) = \mathbf{F}_p^{(1)}(E_p(s)) \| \cdots \| \mathbf{F}_p^{(t)}(E_p(s)) \in \{0,1\}^{t\rho n}$ are computed for all $p = 1, \ldots, r$, where $\mathbf{F}_p^{(q)}(E_p(s)) \in \{0,1\}^{\rho n}$ for all $q = 1, \ldots, t$;

2. The value $w(s) = w^{(1)}(s) \| \cdots \| w^{(t)}(s)$ is computed, where $w^{(q)}(s)$ equals (for all $q = 1, \ldots, t$) the first n bits of the product $\bigodot_{p=1}^{r} \mathbf{F}_p^{(q)}(E_p(s))$, and $\odot$ denotes multiplication in $GF(2^{\rho n})$ with ρn-bit strings interpreted as elements of the finite field $GF(2^{\rho n})$;

3. Finally, the value $\bigoplus_{q=1}^{t} \mathbf{G}_q(w^{(q)}(s))$ is output.

Our approach relies on the observation that if for each new query to the construction with input $s \in \{0,1\}^m$ there exists an index $q \in \{1,\ldots,t\}$ for which $\mathbf{G}_q$ has not been queried yet at the value $w^{(q)}(s)$, either directly at its public interface or by the construction at the private interface, the resulting output value is uniformly distributed and independent from all previously-returned values. This resembles the approach taken to extend the domain of (secret) random functions [1, 4, 21]. However, we stress that the role of the first two stages (including the functions $E_1,\ldots,E_r$) is crucial here: Not only they have to guarantee that such an index q always exists, but they must also permit simulation of the puRF's $\mathbf{F}_1,\ldots,\mathbf{F}_r$ and $\mathbf{G}_1,\ldots,\mathbf{G}_t$ given only access to the public interface of an (ideal) puRF $\mathbf{R} : \{0,1\}^m \to \{0,1\}^\ell$, without seeing the queries made to the private interface of $\mathbf{R}$. Also, the probability that the simulation fails must be small enough to allow security beyond the birthday barrier.

3.2 Input-Restricting Functions

For every $s \in \{0,1\}^m$ one can always learn the value $w(s)$ by querying the public interfaces of $\mathbf{F}_1,\ldots,\mathbf{F}_r$ with appropriate inputs $E_1(s),\ldots,E_p(s)$, respectively. For every such s, the sum $\bigoplus_{q=1}^{t} \mathbf{G}_q(w^{(q)}(s))$ equals the output of the construction on input s. The simulator must ensure that its answers for queries to the functions $\mathbf{G}_1,\ldots,\mathbf{G}_t$ are consistent with these constraints. However, if $E_1,\ldots,E_r$ allow a relatively small number of queries to the functions $\mathbf{F}_1,\ldots,\mathbf{F}_t$ to reveal a too large number of values $w(s)$, then the simulator possibly fails to satisfy all constraints. For example, the *Benes* construction [1] adopts an approach similar to the one of our construction, but suffers from this problem and its security in the setting of puRF's is inherently bounded by the birthday barrier (cf. Appendix B for further details).

To overcome this problem, we introduce the following combinatorial notion.

Definition 4. Let $\epsilon \in (0,1)$, and let $m > n$. A family $\mathcal{E}$ of functions $E_1,\ldots,E_r : \{0,1\}^m \to \{0,1\}^n$ is called (m,δ,ϵ)-*input restricting* if it satisfies the following two properties:

Injective. For all $s \neq s' \in \{0,1\}^m$, there exists $p \in \{1,\ldots,r\}$ such that $E_p(s) \neq E_p(s')$.

Input-Restricting. For all subsets $\mathcal{U}_1,\ldots,\mathcal{U}_r \subseteq \{0,1\}^n$ such that $|\mathcal{U}_1| + \cdots + |\mathcal{U}_r| \leq 2^{n(1-\epsilon)}$, we have

$$\left|\{s \in \{0,1\}^m \mid E_p(s) \in \mathcal{U}_p \text{ for all } p = 1,\ldots,r\}\right| \leq \delta \cdot (|\mathcal{U}_1| + \cdots + |\mathcal{U}_r|)\,.$$

It is easy to see that $\delta \geq 1/r$ must hold. Furthermore, we need $r \cdot n \geq m$ for the family to be injective. When talking about efficiency, we can naturally extend the notion to asymptotic families $\mathcal{E} = \{\mathcal{E}_n\}_{n\in\mathbb{N}}$ of function families by letting m, δ, ϵ, and r be functions of n, and $\mathcal{E}_n = \{E_1^n,\ldots,E_{r(n)}^n\}$, with $E_p^n : \{0,1\}^{m(n)} \to \{0,1\}^n$. In particular, note that we allow the size of the family to grow with the security parameter. The family $\mathcal{E}_n$ is called *explicit* if $r = r(n)$ is polynomial in n and if there exists a (uniform) polynomial-time (in n) algorithm E that outputs $E_p^n(s) \in \{0,1\}^n$ on input $n \in \mathbb{N}$, $s \in \{0,1\}^{m(n)}$, and $p \in \{1,\ldots,r(n)\}$. The family is additionally called *invertible* if there exists an algorithm which on input the sets $\mathcal{U}_1,\ldots,\mathcal{U}_r \subseteq \{0,1\}^n$ and n returns the set of all $s \in \{0,1\}^m$ for which $E_p(s) \in \mathcal{U}_p$ for all $p = 1,\ldots,r$ in time polynomial in $|\mathcal{U}_1| + \cdots + |\mathcal{U}_r|$ and in n. We will not, however, stress the asymptotic point of view in the following, as long as it is clear from the context that the statements can be also formalized in this sense.

We postpone the discussion of the existence of explicit function families to Section 4, where we construct (for all constants ϵ) explicit families of (m,δ,ϵ)-input-restricting functions

for all polynomials m and sufficiently-small δ using highly unbalanced expander graphs with polynomial-degree.

3.3 Main Result

Let $\epsilon \in (0,1)$. The concrete construction $\mathbf{C}^{\mathcal{E}}_{\epsilon,m,\ell}(\cdot)$ is obtained from the description in Section 3.1 by instantiating the functions $E_1,\ldots,E_r$ with an explicit family $\mathcal{E} = \{E_1,\ldots,E_r\}$ of (m,δ,ϵ)-input restricting functions with n-bit output. Also, we let $\rho := \left\lceil \frac{m}{n} + 2 - \epsilon \right\rceil$ and $t := \lceil 2/\epsilon - 1 \rceil$. Note that underlying $r+t$ puRF's can be seen as a single puRF $\mathbf{R}' : \{0,1\}^{n+\phi(n)} \to \{0,1\}^n$, where $\phi(n) = \lceil \log(r \cdot t\rho + t\ell/n) \rceil$. If m, ℓ, and $1/\epsilon$ are polynomial in n, then in particular $\phi(n) = \mathcal{O}(\log n)$. Also, it is easy to see that $\mathbf{C}^{\mathcal{E}}_{\epsilon,m,\ell}(\cdot)$ is efficient, as long as the function family $\mathcal{E}$ is explicit. The following is the main theorem of this paper and it is proved in the next section.

Theorem 3. *The construction $\mathbf{C}^{\mathcal{E}}_{\epsilon,m,\ell}(\cdot)$ is an (α,σ)-reduction of the puRF $\mathbf{R} : \{0,1\}^m \to \{0,1\}^\ell$ to the puRF's $\mathbf{F}_1,\ldots,\mathbf{F}_r : \{0,1\}^n \to \{0,1\}^{t\cdot\rho n}$ and $\mathbf{G}_1,\ldots,\mathbf{G}_t : \{0,1\}^n \to \{0,1\}^\ell$, where for all $k \le 2^{n(1-\epsilon)} - r$,*

$$\alpha(k) \le 2r^t(\delta+1)^{t+1} \cdot k^{t+2} \cdot 2^{-nt} + \frac{1}{2}t(\delta+1) \cdot k \cdot (k+2r+1) \cdot 2^{m-\rho n}$$

and $\sigma(k) \le \delta(n) \cdot k$. If the family $\mathcal{E}$ is invertible, the simulator runs in time polynomial in k and n, and in particular $\mathbf{C}^{\mathcal{E}}_{\epsilon,m,\ell}(\cdot)$ is also a computational reduction.

We remark the following two important consequences of Theorem 3.

- First, if ϵ is constant and r,δ polynomial in n, the above advantage $\alpha(k)$ is negligible for all parameters k up to $k = 2^{n(1-\epsilon)} - r$. In particular, choosing $\epsilon < \frac{1}{2}$ leads to security beyond the birthday barrier,[8] and we are going to provide input-restricting families of functions with appropriate parameters in Section 4.
- Second, the result can be used to extend the domain of a puRF $\mathbf{R}' : \{0,1\}^n \to \{0,1\}^n$ with security up to $2^{n(1-\mu)}$ queries: One chooses any $\epsilon < \mu$ and n' maximal such that $n' + \phi(n') \le n$, and interprets the function $\mathbf{R}'$ as a puRF $\{0,1\}^{n'+\phi(n')} \to \{0,1\}^{n'}$ by dropping approximately $\phi(n')$ bits of the output. The above advantage is still negligible for all $k \le 2^{n'(1-\epsilon)} - r$, and hence for all $k \le 2^{n(1-\mu)}$ for n large enough, since $n - n' = o(n)$.

3.4 Proof of Theorem 3

We prove that there exists a simulator $\mathbf{S}$ such that $\Delta^{\mathbf{D}}(\mathbf{H}_1,\mathbf{H}_2)$ is bounded by the above expression for all distinguishers $\mathbf{D}$ making at most $k \le 2^{n(1-\epsilon)} - r$ queries, where for notational convenience $\mathbf{H}_1$ and $\mathbf{H}_2$ are defined as

$$\mathbf{H}_1 := [\mathbf{F}_1,\ldots,\mathbf{F}_r,\mathbf{G}_1,\ldots,\mathbf{G}_t,\mathbf{C}^{\mathcal{E}}_{\epsilon,m,\ell}(\mathbf{F}_1,\ldots,\mathbf{F}_r,\mathbf{G}_1,\ldots,\mathbf{G}_t)]$$
$$\mathbf{H}_2 := [\mathbf{S}(\mathbf{R}),\mathbf{R}].$$

There are three types of queries to the systems $\mathbf{H}_1$ and $\mathbf{H}_2$: The first two types are $\mathbf{F}$-*queries*, denoted (F,p,u) for $p \in \{1,\ldots,r\}$ and $u \in \{0,1\}^n$, and $\mathbf{G}$-*queries*, denoted (G,q,v), for $v \in \{0,1\}^n$ and $q \in \{1,\ldots,t\}$. In $\mathbf{H}_1$, a query (F,p,u) returns the value $\mathbf{F}_p(u)$ and a query (G,q,v) returns the value $\mathbf{G}_q(v)$, while in $\mathbf{H}_2$ both query-types are answered by the simulator $\mathbf{S}$. The

[8] Note that ϵ could even be some function going (slowly) towards zero, even though this may require setting t differently.

upon receiving an **F**-query $x_i = (\mathrm{F}, p, u)$ for the first time:
  if $\mathbf{F}_p(u)$ is undefined **then**
    set $\mathbf{F}_p(u)$ to a uniform random value
  compute $\Delta\mathcal{S}_i := \{s_1, \ldots, s_{|\Delta\mathcal{S}_i|}\}$
  for $j := 1$ to $|\Delta\mathcal{S}_i|$ **do**
    let $q_j \in \{1, \ldots, t\}$ be such that $w(s_j)^{(q_j)} \notin w^{(q_j)}(\mathcal{S}_{i-1} \cup \{s_1, \ldots, s_{j-1}\}) \cup \mathcal{G}_{q_j,i-1}$
    if no such q_j exists **then abort**
    for all $q \neq q_j$ **do**
      if $\mathbf{G}_q(w^{(q)}(s_j))$ is undefined **then** set $\mathbf{G}_q(w^{(q)}(s_j))$ to a uniform random value
    $\mathbf{G}_{q_j}(w^{(q_j)}(s_j)) := \mathbf{R}(s_j) \oplus \bigoplus_{q \neq q_j} \mathbf{G}_q(w^{(q)}(s_j))$
  return $\mathbf{F}_p(y)$

upon receiving a **G**-query $x_i = (\mathrm{G}, q, v)$ for the first time:
  if $\mathbf{G}_q(v)$ is undefined **then**
    set $\mathbf{G}_q(v)$ to a uniform random value
  return $\mathbf{G}_q(v)$

Figure 2: Simulator **S** in the proof of Theorem 3. The simulator also constantly keeps track of the sets $\mathcal{F}_{p,i}$ and $\mathcal{G}_{q,i}$ for all $p = 1, \ldots, r$, $q = 1, \ldots, t$, and $i = 1, 2, \ldots$.

third type of queries, called **R**-*queries*, are denoted (R, s) for $s \in \{0,1\}^m$ and are answered by the construction $\mathbf{C}^{\mathcal{E}}_{\epsilon,m,\ell}(\cdot)$ in $\mathbf{H}_1$, and by the private interface of the random function **R** in $\mathbf{H}_2$. Given the first i queries $x^i = [x_1, \ldots, x_i]$, where $x_j \in \{(\mathrm{F}, p, u), (\mathrm{G}, q, v), (\mathrm{R}, s)\}$ for all $j = 1, \ldots, i$, we define for all indices p and q the sets $\mathcal{F}_{p,i}$ and $\mathcal{G}_{q,i}$ that contain, respectively, all values $u \in \{0,1\}^n$ for which a query (F, p, u) and all $v \in \{0,1\}^n$ for which a query (G, q, v) appears in x^i. Also, we let $\mathcal{R}_i$ be the set of values $s \in \{0,1\}^m$ for which a query (R, s) appears in x^i, and we let $\mathcal{S}_i$ consist of all the values $s \in \{0,1\}^m$ such that $E_p(s) \in \mathcal{F}_{p,i}$ for all $p = 1, \ldots, r$. Furthermore, let $\Delta\mathcal{S}_i := \mathcal{S}_i \setminus \mathcal{S}_{i-1}$. Notice that the set $\mathcal{S}_i$ contains all inputs for which the values returned by the first i queries allow to compute the value $w(s)$. Clearly, $|\mathcal{S}_i| = \sum_{j=1}^{i} |\Delta\mathcal{S}_j| \leq \delta \cdot i$ for all $i \leq 2^{n(1-\epsilon)}$, since the family $\mathcal{E}$ is input-restricting. For $s \in \mathcal{S}_i$, we define $w(s) = w^{(1)}(s) \| \cdots \| w^{(t)}(s)$ as in the description of $\mathbf{C}^{\mathcal{E}}_{\epsilon,m,\ell}(\cdot)$ according to the answers of the first queries, and for a set $\mathcal{S} \subseteq \mathcal{S}_i$ we use the shorthand $w^{(q)}(\mathcal{S}) := \{w^{(q)}(s) \,|\, s \in \mathcal{S}\}$.

The simulator **S** defines the function tables of $\mathbf{F}_1, \ldots, \mathbf{F}_r$ and of $\mathbf{G}_1, \ldots, \mathbf{G}_t$ *dynamically*. That is, all values $\mathbf{F}_p(u)$ and $\mathbf{G}_q(v)$ are initially *undefined* for all $u, v \in \{0,1\}^n$ and indices p and q. Upon processing a new **F**-query $x_i = (\mathrm{F}, p, u)$, the simulator sets the value $\mathbf{F}_p(u)$ to a fresh random value and computes the set $\Delta\mathcal{S}_i$: The simulator knows this set, as it processes all **F**-queries. For each $s \in \Delta S_i$, the equality $\bigoplus_{q=1}^{t} \mathbf{G}_q(w^{(q)}(s)) = \mathbf{R}(s)$ must be satisfied, and hence **S** tries to satisfy these constraints by appropriately setting the values of the functions $\mathbf{G}_1, \ldots, \mathbf{G}_t$. More precisely, it looks for an ordering of $\Delta\mathcal{S}_i = \{s_1, \ldots, s_{|\Delta\mathcal{S}_i|}\}$ with the property that for all $j = 1, \ldots, |\Delta\mathcal{S}_i|$ there exists $q_j \in \{1, \ldots, t\}$ such that $w^{(q_j)}(s_j) \notin \{w^{(q_j)}(s_1), \ldots, w^{(q_j)}(s_{j-1})\} \cup \mathcal{G}_{q,i-1}$, and sets $\mathbf{G}_{q_j}(w^{(q_j)}(s_j)) := \mathbf{R}(s_j) \oplus \bigoplus_{q \neq q_j} \mathbf{G}_q(w^{(q)}(s_j))$ for $j = 1, \ldots, |\Delta\mathcal{S}_i|$, where each undefined value in the sums is set to an independent random value. A query to the public interface of **R** is issued in order to learn $\mathbf{R}(s_j)$. If no such ordering exists, then the simulator aborts.[9] Finally, the value $\mathbf{F}_p(u)$ is returned. For a

[9]Note that there is no need to formalize the exact meaning of abortion, since whenever the simulator fails to find such an ordering, then the distinguisher is assumed to win.

query $x_i = (\mathrm{G}, q, v)$, the simulator returns $\mathbf{G}_q(v)$, defining it to a random value if undefined. In Figure 2, we provide a detailed pseudo-code description of the simulator $\mathbf{S}$. The number of $\mathbf{R}$-queries made by the simulator after $i \leq 2^{n(1-\epsilon)}$ queries is $|\mathcal{S}_i| \leq \delta \cdot i$. Also, as long as the family $\mathcal{E}$ is invertible and an appropriate ordering can be efficiently found, its running time is efficient in k and n. In fact, we show that with very high probability *any* ordering can be used. Without loss of generality, it is convenient to advance the generation of the random functions $\mathbf{F}_1, \ldots, \mathbf{F}_r$ to the initialization phase, that is, their *entire* function tables are generated once uniformly at random in both $\mathbf{H}_1$ and $\mathbf{H}_2$. Subsequently, all queries (F, p, u) are answered according to the initial choice. In particular, this means that in $\mathbf{H}_2$ the simulator $\mathbf{S}$ uses the value $\mathbf{F}_p(u)$ already defined instead of generating a new fresh random value. It is clear that the behavior of both systems is unchanged. This also allows us to define the value $w(s) = w^{(1)}(s) \| \cdots \| w^{(t)}(s)$ for *all* $s \in \{0,1\}^m$ and each such value induces a constraint, namely the answer of an $\mathbf{R}$-query (R, s) must equal $\bigoplus_{q=1}^{t} \mathbf{G}_q(w^{(q)}(s))$. Such a constraint remains hidden until $s \in \Delta\mathcal{S}_i$ from some i, and in this case the simulator attempts to fill the function tables of $\mathbf{G}_1, \ldots, \mathbf{G}_t$ consistently. To avoid possible problems, we have to account for two things captured by the two following monotone conditions which we define on both $\mathbf{H}_1$ and $\mathbf{H}_2$:

(a) The monotone condition $\mathcal{A} = A_0, A_1, \ldots$ fails at query i if there exists an $s \in \Delta\mathcal{S}_i$ such that $w^{(q)}(s) \in w^{(q)}(\mathcal{S}_i \setminus \{s\}) \cup \mathcal{G}_{q,i-1}$ for all $q = 1, \ldots, t$.

(b) The monotone condition $\mathcal{B} = B_0, B_1, \ldots$ fails at query i if there exists $s \in \mathcal{R}_i \setminus \mathcal{S}_i$ such that $w^{(q)}(s) \in w^{(q)}(\mathcal{S}_i \cup \mathcal{R}_i \setminus \{s\}) \cup \mathcal{G}_{q,i}$ for all $q = 1, \ldots, t$.

As long as $\mathcal{A}$ does not fail, the simulator never aborts. This in particular implies that $\mathbf{R}$-queries (R, s) for $s \in \mathcal{S}_i$ in $\mathbf{H}_2$ are consistent with $\mathbf{G}$-queries answered by the simulator. However, all $\mathbf{R}$-queries (R, s) for $s \notin \mathcal{S}_i$ are answered independently and uniformly at random in $\mathbf{H}_2$, and $\mathcal{B}$ ensures that this happens in $\mathbf{H}_1$ as well. In Section 3.5, we prove the following lemma, which formalizes this argument and states that as long as neither $\mathcal{A}$ nor $\mathcal{B}$ fail, then $\mathbf{H}_1$ and $\mathbf{H}_2$ behave identically.

Lemma 4. $\mathbf{H}_1^{\mathcal{A}\wedge\mathcal{B}} \equiv \mathbf{H}_2^{\mathcal{A}\wedge\mathcal{B}}$.

To provide some intuition as to why the probability that a distinguisher $\mathbf{D}$ makes $\mathcal{A} \wedge \mathcal{B}$ fail is small, let us assume first that for any two distinct $s, s' \in \{0,1\}^m$ (such that at least one of them is not in $\mathcal{S}_i$) and for all $q = 1, \ldots, t$, the probability (conditioned on the answers to the previous queries) that $w^{(q)}(s) = w^{(q)}(s')$ is bounded by some small value φ (say $\varphi \approx 2^{-n}$). In order to upper bound the probability of $\mathcal{A}$ failing after query i, combining the union bound with the above assumption we see that $\mathsf{P}(w^{(q)}(s) \in w^{(q)}(\mathcal{S}_i \setminus \{s\}) \cup \mathcal{G}_{q,i-1}) \leq |w^{(q)}(\mathcal{S}_i \setminus \{s\}) \cup \mathcal{G}_{q,i-1}| \cdot \varphi \leq (\delta+1) \cdot i \cdot \varphi$ for all $s \in \Delta\mathcal{S}_i$, since $\mathcal{E}$ is input-restricting. Furthermore, for all distinct $q, q' \in \{1, \ldots, t\}$ and $s, s' \in \{0,1\}^n$ (possibly $s = s'$), the structure of the first two stages of $\mathbf{C}^{\mathcal{E}}_{\epsilon,m,\ell}(\cdot)$ ensures that the values $w^{(q)}(s)$ and $w^{(q')}(s')$ are statistically independent, and hence

$$\mathsf{P}(\forall q : w^{(q)}(s) \in w^{(q)}(\mathcal{S}_i \setminus \{s\}) \cup \mathcal{G}_{q,i-1}) \leq (\delta+1)^t \cdot i^t \cdot \varphi^t.$$

Therefore, the probability $\mathsf{p}^{\mathbf{H}_1}_{\overline{A_i}|X^i Y^{i-1} A_{i-1}}(x^i, y^{i-1}) = \mathsf{p}^{\mathbf{H}_2}_{\overline{A_i}|X^i Y^{i-1} A_{i-1}}(x^i, y^{i-1})$ that there exists an $s \in \Delta\mathcal{S}_i$ making $\mathcal{A}$ fail after query i is bounded by $|\Delta\mathcal{S}_i| \cdot (\delta+1)^t \cdot i^t \cdot \varphi^t$, where $|\Delta\mathcal{S}_i|$ is small for all $i \leq 2^{n(1-\epsilon)}$.

Nevertheless, turning this intuition into a formal proof (and extending it to the monotone condition $\mathcal{B}$) requires additional care. The probability that $w^{(q)}(s)$ equals $w^{(q)}(s')$ happens to be small only with overwhelming probability (taken over the answers to the previous queries):

This fact follows from the use of multiplication in $GF(2^{\rho n})$ and the choice of a sufficiently large parameter ρ.

In particular, Section 3.6 provides a complete proof of the following lemma.

Lemma 5. *For all distinguishers* $\mathbf{D}$ *making at most* $k \leq 2^{n(1-\epsilon)} - r$ *queries we have*

$$\nu^{\mathbf{D}}(\mathbf{H}_1^{\mathcal{A}\wedge\mathcal{B}}) = \nu^{\mathbf{D}}(\mathbf{H}_2^{\mathcal{A}\wedge\mathcal{B}}) \leq 2r^t(\delta+1)^{t+1} \cdot k^{t+2} \cdot 2^{-nt} + \frac{1}{2}t(\delta+1) \cdot k \cdot (k+2r+1) \cdot 2^{m-\rho n}.$$

By combining Lemmas 4 and 5, Theorem 3 follows making use of Lemma 1.

3.5 Proof of Lemma 4

We want to prove that $\mathsf{p}^{\mathbf{H}_1}_{Y^i A_i B_i|X^i} = \mathsf{p}^{\mathbf{H}_2}_{Y^i A_i B_i|X^i}$ for all $i \geq 1$. We fix the first i queries $x^i = [x_1, \ldots, x_i]$, and assume without loss of generality that $w^{(q)}(\mathcal{S}_i) \subseteq \mathcal{G}_{q,i}$ for all $q = 1, \ldots, t$. If this does not hold, we can extend x^i to a j-tuple $x^j = [x_1, \ldots, x_i, x_{i+1}, \ldots, x_j]$, where the last $j - i$ queries are all $\mathbf{G}$-queries (G, q, v) for all $q = 1, \ldots, t$ and $v \in w^{(q)}(\mathcal{S}_i) \setminus \mathcal{G}_{q,i}$ (in any order). It is easy to verify that if A_i and B_i hold, then also A_j and B_j hold, and hence

$$\mathsf{p}^{\mathbf{H}_b}_{Y^i A_i B_i|X^i}(y^i, x^i) = \sum_{y_{i+1},\ldots,y_j} \mathsf{p}^{\mathbf{H}_b}_{Y^j A_j B_j|X^j}([y_1, \ldots, y_i, y_{i+1}, \ldots, y_j], x^j), \tag{1}$$

and hence it is sufficient to prove equality for input sequences with $w^{(q)}(\mathcal{S}_i) \subseteq \mathcal{G}_{q,i}$ for all $q = 1, \ldots, t$, as the general case follows by (1).

We denote by F the random variable representing the concatenation of the random tables of the puRF's $\mathbf{F}_1, \ldots, \mathbf{F}_r$. For $b \in \{1, 2\}$, summing over all possible values of F yields

$$\mathsf{p}^{\mathbf{H}_b}_{A_i B_i Y^i|X^i} = \sum_F \mathsf{p}^{\mathbf{H}_b}_{F|X^i} \cdot \mathsf{p}^{\mathbf{H}_b}_{A_i B_i|X^i F} \cdot \mathsf{p}^{\mathbf{H}_b}_{Y^i|X^i F A_i B_i}.$$

Clearly, we have $\mathsf{p}^{\mathbf{H}_1}_{F|X^i} = \mathsf{p}^{\mathbf{H}_2}_{F|X^i}$, since the function tables are chosen uniformly in both $\mathbf{H}_1$ and $\mathbf{H}_2$. Also, we have $\mathsf{p}^{\mathbf{H}_1}_{A_i B_i|X^i F} = \mathsf{p}^{\mathbf{H}_2}_{A_i B_i|X^i F} \in \{0, 1\}$, as A_i and B_i depend deterministically on X^i and F. Finally, we show that $\mathsf{p}^{\mathbf{H}_1}_{Y^i|X^i F A_i B_i} = \mathsf{p}^{\mathbf{H}_2}_{Y^i|X^i F A_i B_i}$. Note that since F is fixed, in both systems $\mathbf{F}$-queries are obviously answered in the same way.

In system $\mathbf{H}_1$, if we restrict ourselves to the outputs of the $\mathbf{G}$-queries, then the values returned are uniform and independent. Furthermore for every $\mathbf{R}$-query (R, s) such that $s \in \mathcal{S}_i$ the value returned is uniquely determined by the answers to the $\mathbf{G}$-queries as $\bigoplus_{q=1}^{t} \mathbf{G}_q(w^{(q)}(s))$, and all these $\mathbf{G}$-queries are asked, since $w^{(q)}(\mathcal{S}_i) \subseteq \mathcal{G}_{q,i}$ for all $q = 1, \ldots, t$ by assumption. Finally, for all $s \in \mathcal{R}_i \setminus \mathcal{S}_i$, since B_i holds, there exists a q such that $w^{(q)}(s) \notin w^{(q)}(\mathcal{S}_i \cup \mathcal{R}_i \setminus \{s\}) \cup \mathcal{G}_{q,i}$: the value $\mathbf{G}_q(w^{(q)}(s))$ is random and independent of all other returned values, and every such $\mathbf{R}$-query returns a random value which is independent of all other values.

For system $\mathbf{H}_2$, since A_i holds, it also easy to see (by the construction of the simulator) that the joint probability distribution of the outputs of all $\mathbf{G}$-queries is uniform. Furthermore, an $\mathbf{R}$-query (R, s) with $s \in \mathcal{S}_i \setminus \mathcal{R}_i$ is always answered by an independent and uniform random value, since these queries are answered by a random function. However, if $s \in \mathcal{S}_i$, then the answer is determined uniquely by the answers to $\mathbf{G}$-queries, again by the construction of the simulator.

3.6 Proof of Lemma 5

We first recall the following well-known result, of which we omit the proof.

Theorem 6 (Schwartz-Zippel)**.** *Let $\mathbb{F}$ be a finite field, and let $P \in \mathbb{F}[X_1, \ldots, X_n]$ be an n-variate polynomial over $\mathbb{F}$ with degree d. Then, the number of tuples $(x_1, \ldots, x_n) \in \mathbb{F}^n$ that satisfy $P(x_1, \ldots, x_n) = 0$ is at most $d \cdot |\mathbb{F}|^{n-1}$.*

For our setting $\mathbb{F} = GF(2^{\rho n})$, and we work with some representation of the elements as ρn-bit strings. We need the following simple corollary of Theorem 6.

Corollary 7. *Let $a, b \in GF(2^{\rho n})$, not both equal to 0, let $X_1, \ldots, X_N \in GF(2^{\rho n})$ be independent and uniformly-distributed random variables, and let $\mathcal{J}, \mathcal{J}' \subseteq \{1, \ldots, N\}$. Then:*

(i) If $\mathcal{J} \neq \mathcal{J}'$, then $\mathsf{P}((a \cdot \bigodot_{j \in \mathcal{J}} X_j)|_n = (b \cdot \bigodot_{j \in \mathcal{J}'} X_j)|_n) \leq \max\{|\mathcal{J}|, |\mathcal{J}'|\} \cdot 2^{-n}$.

(ii) If $\mathcal{J} = \mathcal{J}'$ and $a \neq b$, then $\mathsf{P}(a \cdot \bigodot_{j \in \mathcal{J}} X_j|_n = b \cdot \bigodot_{j \in \mathcal{J}'} X_j|_n) \leq |\mathcal{J}| \cdot 2^{-n}$.

Throughout the proof of Lemma 5, we work with system $\mathbf{H}_2$, as this makes some arguments easier. Notice that Lemmas 1 and 4 allow this, since $\nu^{\mathbf{D}}(\mathbf{H}_1^{\mathcal{A} \wedge \mathcal{B}}) = \nu^{\mathbf{D}}(\mathbf{H}_2^{\mathcal{A} \wedge \mathcal{B}})$ for all distinguishers $\mathbf{D}$. First, we introduce some additional notation. For $i \geq 1$, let $x^i = [x_1, \ldots, x_i]$ be the first i queries, where $x_j \in \{(\mathrm{F}, p, u), (\mathrm{G}, q, v), (\mathrm{R}, s)\}$ for all $j = 1, \ldots, i$. For any $s \in \{0,1\}^m$, define the set $\mathcal{P}_i(s)$ as the set of indices $p \in \{1, \ldots, r\}$ such that $x_j = (\mathrm{F}, p, E_p(s))$ appears among the first i queries. Furthermore, we let $\overline{w}_i(s) = \overline{w}_i^{(1)}(s) \| \cdots \| \overline{w}_i^{(t)}(s)$ be the component-wise product of the values $\mathbf{F}_p(E_p(s))$ for all $p \in \mathcal{P}_i(s)$, that is $\overline{w}_i^{(q)}(s) := \bigodot_{p \in \mathcal{P}_i(s)} \mathbf{F}_p^{(q)}(E_p(s))$ for all $q = 1, \ldots, t$.

We also need to introduce two additional monotone conditions for the remainder of the proof. The condition $\mathcal{C} = C_0, C_1, \ldots$ fails after i queries if there exists distinct $s, s' \in \{0,1\}^m$ such that $\mathcal{P}_i(s) = \mathcal{P}_i(s')$, $E_{\bar{p}}(s) = E_{\bar{p}}(s')$ for all $\bar{p} \notin \mathcal{P}_i(s)$, and $\overline{w}_i^{(q)}(s) = \overline{w}_i^{(q)}(s')$ for some $q \in \{1, \ldots, t\}$. Note that the fact that C_0 holds follows from the fact that the family $\mathcal{E}$ is injective. Also, a further monotone condition $\mathcal{D} = D_0, D_1, \ldots$ fails after i queries if there exists $s \in \{0,1\}^m$ such that $\overline{w}_i^{(q)}(s) = 0$ for some $q \in \{1, \ldots, t\}$. Clearly, $\mathsf{P}^{\mathbf{D} \circ \mathbf{H}_2}(\overline{A_k} \vee \overline{B_k}) \leq \mathsf{P}^{\mathbf{D} \circ \mathbf{H}_2}(\overline{A_k} \vee \overline{B_k} \vee \overline{C_k} \vee \overline{D_k})$.

We also define a (non-monotone!) sequence of events $U_0, U_1, U_2, \ldots$ such that U_i is false if there exists $s \in \Delta\mathcal{S}_i$ such that $w^{(q)}(s) \in w^{(q)}(\mathcal{S}_i \setminus \{s\}) \cup \mathcal{G}_{q,i}$ for all $q = 1, \ldots, t$. A further (non-monotone) sequence of events $V_0, V_1, \ldots$ is such that V_i is false if there exists $s \in \mathcal{R}_i \setminus \mathcal{S}_i$ such that $w^{(q)}(s) \in w^{(q)}(\mathcal{S}_i \cup \mathcal{R}_i \setminus \{s\}) \cup \mathcal{G}_{q,i}$. If $\overline{A_k} \vee \overline{B_k} \vee \overline{C_k} \vee \overline{D_k}$ holds, there must exist an $i \in \{1, \ldots, k\}$ such that (at least) one of the following events occurs: (i) $\overline{D_i} \wedge D_{i-1}$, (ii) $\overline{C_i} \wedge C_{i-1} \wedge D_{i-1}$, (iii) $\overline{U_i} \wedge C_{i-1} \wedge D_{i-1}$, or (iv) $\overline{V_i} \wedge C_i \wedge D_i$. Using the union bound and the fact that $\mathsf{P}(\mathcal{E} \wedge \mathcal{E}') \leq \mathsf{P}(\mathcal{E}'|\mathcal{E})$ for any two events $\mathcal{E}$ and $\mathcal{E}'$ such that $\mathsf{P}(\mathcal{E}) \geq 0$, we obtain

$$\begin{aligned}\mathsf{P}^{\mathbf{D} \circ \mathbf{H}_2}(\overline{A_k} \vee \overline{B_k}) \leq \sum_{i=1}^{k} \mathsf{P}^{\mathbf{D} \circ \mathbf{H}_2}(\overline{D_i}|D_{i-1}) + \sum_{i=1}^{k} \mathsf{P}^{\mathbf{D} \circ \mathbf{H}_2}(\overline{C_i}|C_{i-1}D_{i-1}) \\ + \sum_{i=1}^{k} \mathsf{P}^{\mathbf{D} \circ \mathbf{H}_2}(\overline{U_i}|C_{i-1}D_{i-1}) + \sum_{i=1}^{k} \mathsf{P}^{\mathbf{D} \circ \mathbf{H}_2}(\overline{V_i}|C_iD_i) \quad (2)\end{aligned}$$

The following lemma is the central step in the proof of Lemma 5.

Lemma 8. *For all $i \leq 2^{n(1-\epsilon)} - r$, all x^i, y^{i-1}, and y_i, we have*

(i) $\mathsf{p}^{\mathbf{H}_2}_{\overline{D_i}|X^iY^{i-1}D_{i-1}}(x^i, y^{i-1}) \le t \cdot 2^{m-\rho n}$;

(ii) $\mathsf{p}^{\mathbf{H}_2}_{\overline{C_i}|X^iY^{i-1}C_{i-1}D_{i-1}}(x^i, y^{i-1}) \le t \cdot \delta \cdot (i+r) \cdot 2^{m-\rho n}$;

(iii) $\mathsf{p}^{\mathbf{H}_2}_{\overline{U_i}|X^iY^{i-1}C_{i-1}D_{i-1}}(x^i, y^{i-1}) \le |\Delta\mathcal{S}_i| \cdot (\delta+1)^t \cdot i^t \cdot 2^{-nt}$;

(iv) $\mathsf{p}^{\mathbf{H}_2}_{\overline{V_i}|X^iY^iC_iD_i}(x^i, y^i) \le r^t \cdot (\delta+1)^t \cdot i^{t+1} \cdot 2^{-nt}$.

Before we turn to the proof of Lemma 8, we briefly show that it implies the upper bound in the proof of Lemma 5. Since the bounds hold for all x^i, y^{i-1}, and y_i that can appear, they also clearly hold without being conditioned on these values by a simple averaging argument. Therefore, we obtain for all $k \le 2^{n(1-\epsilon)} - r$,

$$\sum_{i=1}^{k} \mathsf{P}^{\mathbf{D}\circ\mathbf{H}_2}(\overline{D_i}|D_{i-1}) \;\le\; k \cdot t \cdot 2^{m-\rho n}, \tag{3}$$

$$\sum_{i=1}^{k} \mathsf{P}^{\mathbf{D}\circ\mathbf{H}_2}(\overline{C_i}|C_{i-1}D_{i-1}) \;\le\; t \cdot \delta \cdot 2^{m-\rho n} \cdot \sum_{i=1}^{k}(i+r) = t \cdot \delta \cdot \frac{k(k+2r+1)}{2} \cdot 2^{m-\rho n}. \tag{4}$$

Also, generously bounding $|\Delta\mathcal{S}_i| \le \delta \cdot i$, we have

$$\begin{aligned}\sum_{i=1}^{k} \mathsf{P}^{\mathbf{D}\circ\mathbf{H}_2}(\overline{U_i}|C_{i-1}D_{i-1}) + \sum_{i=1}^{k} \mathsf{P}^{\mathbf{D}\circ\mathbf{H}_2}(\overline{V_i}|C_iD_i) &\le 2 \cdot r^t \cdot (\delta+1)^{t+1} \cdot 2^{-nt} \cdot \sum_{i=1}^{k} i^{t+1} \\ &\le 2 \cdot r^t \cdot (\delta+1)^{t+1} \cdot k^{t+2} \cdot 2^{-nt}.\end{aligned} \tag{5}$$

Plugging (3), (4) and (5) into (2) yields the desired upper bound. We finally turn back to the proof of Lemma 8

Proof of Lemma 8. For (i), (ii), and (iii), assume that the i'th query is a new $\mathbf{F}$-query $x_i = (\mathrm{F}, p, u)$, all other types of queries cannot provoke the failure of the conditions. In particular, let $U = [U^{(1)}, \dots U^{(t)}] \in \{0,1\}^{t\cdot\rho n}$ be the random value returned by the query, which is independent from all other previously-returned values. In fact, this is the only randomness involved in computing the first three probabilities (and we use the notation P^U to stress this fact).

For (i), since D_{i-1} holds, we have $\overline{w}^{(q)}_{i-1}(s) \neq 0$ for all $s \in \{0,1\}^m$ and all $q = 1, \dots, t$. Hence, the union bound and Theorem 6 imply

$$\mathsf{p}^{\mathbf{D}\circ\mathbf{H}_2}_{\overline{D_i}|X^iY^{i-1}D_{i-1}} \le \sum_{s:E_p(s)=u} \mathsf{P}^U(\exists q \in \{1,\dots,t\} : \overline{w}^{(q)}_{i-1}(s) \odot U^{(q)} = 0) \le t \cdot 2^m \cdot 2^{-\rho n}.$$

To prove (ii) choose any $s \in \{0,1\}^m$ with the property that $E_p(s) = u$, and define the set $\mathcal{S}'$ of those $s' \in \{0,1\}^m$ such that $\mathcal{P}_{i-1}(s) = \mathcal{P}_{i-1}(s')$ and $E_{\bar{p}}(s) = E_{\bar{p}}(s')$ for all $\bar{p} \notin \mathcal{P}_{i-1}(s)$. (In particular, $E_p(s') = u$ for all $s' \in \mathcal{S}'$.) Also, let $\mathcal{S}''$ be the set of those $s'' \in \{0,1\}^m$ with $\mathcal{P}_{i-1}(s'') = \mathcal{P}_{i-1}(s) \cup \{p\}$, and $E_{\bar{p}}(s) = E_{\bar{p}}(s'')$ for all $\bar{p} \notin \mathcal{P}_{i-1}(s'')$. Let $\overline{C_{i,s}}$ denote the event that there exists $\bar{s} \in \mathcal{S}' \cup \mathcal{S}'' \setminus \{s\}$ such that $\overline{w}^{(q)}_i(s) = \overline{w}^{(q)}_i(\bar{s})$ for some $q \in \{1,\dots,t\}$. By repeatedly applying the union bound, we derive

$$\begin{aligned}\mathsf{p}^{\mathbf{D}\circ\mathbf{H}_2}_{\overline{C_{i,s}}|X^iY^{i-1}C_{i-1}D_{i-1}} \le \sum_{q=1}^{t} &\mathsf{P}^U(\exists s' \in \mathcal{S}' : \overline{w}^{(q)}_{i-1}(s) \odot U^{(q)} = \overline{w}^{(q)}_{i-1}(s') \odot U^{(q)}) \\ &+ \mathsf{P}^U(\exists s'' \in \mathcal{S}'' : \overline{w}^{(q)}_{i-1}(s) \odot U^{(q)} = \overline{w}^{(q)}_{i}(s'')) \le t \cdot (|\mathcal{S}'| + |\mathcal{S}''|) \cdot 2^{-\rho n},\end{aligned}$$

since $\overline{w}_{i-1}^{(q)}(s) \neq \overline{w}_{i-1}^{(q)}(s')$ for all $s' \in \mathcal{S}'$ by C_{i-1}, and since $\overline{w}_{i-1}^{(q)}(s) \neq 0$ by D_{i-1}, and hence we can use Theorem 6. Furthermore, $|\mathcal{S}'| + |\mathcal{S}''| \leq \delta \cdot (i+r)$, as $\mathcal{E}$ is input-restricting and at most additional $r - |\mathcal{P}_{i-1}(s)| - 1 \leq r$ queries reveal the values $w(s)$ for the inputs in $\mathcal{S}' \cup \mathcal{S}''$. Using once again the union bound, we conclude

$$\mathsf{p}^{\mathbf{D}\circ\mathbf{H}_2}_{\overline{C_i}|X^iY^{i-1}C_{i-1}D_{i-1}} \leq \sum_{s:E_p(s)=u} \mathsf{p}^{\mathbf{D}\circ\mathbf{H}_2}_{\overline{C_{i,s}}|X^iY^{i-1}C_{i-1}D_{i-1}} \leq t \cdot \delta \cdot (i+r) \cdot 2^{m-\rho n}.$$

To prove (iii), note that $s \in \Delta\mathcal{S}_i$ implies that $\mathcal{P}_{i-1}(s) = \{1,\ldots,r\} - \{p\}$. Also note that $w^{(q)}(s) = \overline{w}_{i-1}^{(q)} \odot U^{(q)}|_n$ for all $s \in \Delta\mathcal{S}_i$. Since the randomness of each ρn-bit block is independent, we upper bound

$$\begin{aligned}\mathsf{p}^{\mathbf{D}\circ\mathbf{H}_2}_{\overline{U_i}|X^iY^{i-1}C_{i-1}D_{i-1}} &= \mathsf{P}^U\Big(\bigvee_{s\in\Delta\mathcal{S}_i}\bigwedge_{1\leq q\leq t} w^{(q)}(s) \in w^{(q)}(\mathcal{S}_i \setminus \{s\}) \cup \mathcal{G}_{q,i-1}\Big) \\ &\leq \sum_{s\in\Delta\mathcal{S}_i}\prod_{q=1}^{t} \mathsf{P}^U\Big(w^{(q)}(s) \in w^{(q)}(\mathcal{S}_i \setminus \{s\}) \cup \mathcal{G}_{q,i-1}\Big). \quad (6)\end{aligned}$$

We fix some $s \in \Delta\mathcal{S}_i$ and some $q \in \{1,\ldots,t\}$ and see that

$$\begin{aligned}\mathsf{P}^U\Big(w^{(q)}(s) \in w^{(q)}(\mathcal{S}_i \setminus \{s\}) \cup \mathcal{G}_{q,i-1}\Big) &\leq \mathsf{P}^U\Big(w^{(q)}(s) \in w^{(q)}(\Delta\mathcal{S}_i \setminus \{s\})\Big) \\ &\quad + \mathsf{P}^U\Big(w^{(q)}(s) \in w^{(q)}(\mathcal{S}_{i-1})\Big) + \mathsf{P}^U(w^{(q)}(s) \in \mathcal{G}_{q,i-1})\end{aligned}$$

First, since D_{i-1} holds, $\overline{w}_{i-1}^{(q)}(s) \neq 0$ for all $s \in \Delta\mathcal{S}_i$, and hence $\mathsf{P}^U(\overline{w}_{i-1}^{(q)}(s) \odot U^{(q)}|_n \in \mathcal{G}_{q,i-1}) \leq |\mathcal{G}_{q,i-1}| \cdot 2^{-n} \leq i \cdot 2^{-n}$ by Corollary 7. For the same reason,

$$\mathsf{P}^U\Big(w^{(q)}(s) \in w^{(q)}(\mathcal{S}_i)\Big) \leq \sum_{s'\in\mathcal{S}_{i-1}} \mathsf{P}^U\Big(\overline{w}_{i-1}^{(q)}(s) \odot U^{(q)}|_n = w^{(q)}(s')\Big) \leq |\mathcal{S}_{i-1}| \cdot 2^{-n}.$$

Also, since C_{i-1} holds, we have $\overline{w}_{i-1}^{(q)}(s) \neq \overline{w}_{i-1}^{(q)}(s')$ for all $s' \in \Delta\mathcal{S}_i \setminus \{s\}$ and all $q = 1,\ldots,t$, and we obtain

$$\mathsf{P}\Big(w^{(q)}(s) \in w^{(q)}(\Delta\mathcal{S}_i \setminus \{s\})\Big) \leq \sum_{s'\in\Delta\mathcal{S}_i\setminus\{s\}} \mathsf{P}^U\Big(\overline{w}_{i-1}^{(q)}(s) \odot U^{(q)}|_n = \overline{w}_{i-1}^{(q)}(s') \odot U^{(q)}|_n\Big),$$

which is bounded by $|\Delta\mathcal{S}_i| \cdot 2^{-n}$, once again as a consequence of Corollary 7. Plugging these bounds into (6) leads to

$$\mathsf{p}^{\mathbf{D}\circ\mathbf{H}_2}_{\overline{U_i}|X^iY^{i-1}C_{i-1}D_{i-1}} \leq |\Delta\mathcal{S}_i| \cdot \prod_{q=1}^{t} \underbrace{\big(|\Delta\mathcal{S}_i| + |\mathcal{S}_{i-1}| + |\mathcal{G}_{q,i-1}|\big)}_{\leq(\delta+1)\cdot i} \cdot 2^{-n} \leq |\Delta\mathcal{S}_i| \cdot (\delta+1)^t \cdot i^t \cdot 2^{-nt}.$$

To prove (iv), note that the values $w^{(q)}(s)$ for all $s \in \mathcal{R}_i \setminus \mathcal{S}_i$ have all the form $\overline{w}_i^{(q)}(s) \odot \bigodot_{p\notin\mathcal{P}_i(s)} \mathbf{F}_p^{(q)}(E_p(s))$ for all $q = 1,\ldots,t$. Moreover, conditioned on the outcomes of X^i and Y^i as well as the events C_i and D_i, the values $\mathbf{F}_p(E_p(s))$ for all $s \in \mathcal{R}_i \setminus \mathcal{S}_i$ and $p \notin \mathcal{P}_i(s)$ are independent and uniformly distributed, and the probability for computing the upper bound

of (iv) is taken over these values. (We use the notation $\mathsf{P}^{\mathbf{F}}$ to stress this.) As we did in (iii), we can upper bound

$$\mathsf{p}^{\mathbf{D}\circ\mathbf{H}_2}_{\overline{V_i}|X^iY^iC_iD_i} \leq \sum_{s\in\mathcal{R}_i\setminus\mathcal{S}_i}\prod_{q=1}^{t}\Big[\mathsf{P}^{\mathbf{F}}\Big(w^{(q)}(s)\in w^{(q)}(\mathcal{S}_i)\Big) + \mathsf{P}^{\mathbf{F}}\Big(w^{(q)}(s)\in w^{(q)}(\mathcal{R}_i\setminus(\mathcal{S}_i\cup\{s\}))\Big) + \mathsf{P}^{\mathbf{F}}\Big(w^{(q)}(s)\in\mathcal{G}_{q,i}\Big)\Big]$$

Since D_i holds, we have $\mathsf{P}^{\mathbf{F}}\Big(w^{(q)}(s)\in\mathcal{G}_{q,i}\Big)\leq r\cdot|\mathcal{G}_{q,i}|\cdot 2^{-n}$ by Corollary 7, and for the same reason $\mathsf{P}^{\mathbf{F}}\Big(w^{(q)}(s)\in w^{(q)}(\mathcal{S}_i)\Big)\leq r\cdot\delta\cdot i\cdot 2^{-n}$. Furthermore, we note that again by applying Corollary 7,

$$\mathsf{P}^{\mathbf{F}}\Big(w^{(q)}(s)\in w^{(q)}(\mathcal{R}_i\setminus(\mathcal{S}_i\cup\{s\}))\Big)\leq r\cdot|\mathcal{R}_i\setminus\mathcal{S}_i|\cdot 2^{-n}$$

since for any $s'\in\mathcal{R}_i\setminus\mathcal{S}_i$ such that $s'\neq s$ we have

- either there exists p such that $E_p(s)\neq E_p(s')$ and $p\notin\mathcal{P}_i(s)\cap\mathcal{P}_i(s')$ holds, and Corollary 7 (i) applies;
- or $\mathcal{P}_i(s)=\mathcal{P}_i(s')\neq\emptyset$ and $E_p(s)=E_p(s')$ for all $p\notin\mathcal{P}_i(s)$, in which case $\overline{w}_i^{(q)}(s)\neq\overline{w}_i^{(q)}(s')$ by C_i, and thus Corollary 7 (ii) applies.

Therefore, combining the different bounds we get $\mathsf{p}^{\mathbf{D}\circ\mathbf{H}_2}_{\overline{V_i}|X^iY^iC_iD_i}\leq r^t\cdot(\delta+1)^t\cdot i^{t+1}\cdot 2^{-nt}$. □

4 Existence of Input-Restricting Function Families

In this following, we prove the existence of input-restricting function families as in Definition 4, and we study their relationship to *highly unbalanced* bipartite expander graphs. First, we recall the following definition.

Definition 5. A bipartite graph $G=(V_1,V_2,E)$ is (K,γ)-*expanding* if $|\Gamma(X)|\geq\gamma\cdot|X|$ for all subsets $X\subset V_1$ such that $|X|\leq K$, where $\Gamma(X)\subseteq V_2$ is the set of neighbors of X. Furthermore, such a graph has *left-degree* D if the degree of all $v\in V_1$ is bounded by D.

In the asymptotic case, a *family* of graphs $G=(V_1,V_2,E)$ with $V_1:=\{0,1\}^{m(n)}$, $V_2:=\{0,1\}^n$ (parameterized by the security parameter n) with left-degree $D=D(n)$ is called *explicit* if there exists a (uniform) algorithm which, on input 1^n, $v\in\{0,1\}^{m(n)}$ and $i\in\{1,\ldots,D(n)\}$ outputs the i'th neighbor of v in time polynomial in n. (The ordering of the neighbors is arbitrary.) It turns out that explicit families with appropriate parameters imply the existence of input-restricting families of functions.

Lemma 9. *Let m be such that $m\geq n$. Assume that there exists an explicit family of bipartite (K,γ)-expander graphs $G=(V_1,V_2,E)$ with polynomially-bounded left-degree D where $V_1=\{0,1\}^m$ and $V_2=\{0,1\}^n$. Then, for all $\epsilon>0$ such that $\epsilon>1-\frac{\log(K\gamma)}{n}$ for n large enough, there exists an explicit (m,δ,ϵ)-input-restricting family of functions with $\delta=\gamma^{-1}$ and cardinality $r:=D+\lceil m/n\rceil$. Furthermore, if $\lceil m/n\rceil$ is constant, then the family is invertible.*

Proof. First, we define $E_1, \ldots, E_D : \{0,1\}^m \to \{0,1\}^n$ such that $E_p(s)$ is the p'th neighbor of s for all $p = 1, \ldots, D$. Furthermore, the functions $E_{D+1}, \ldots, E_{D+\lceil m/n \rceil}$ are defined as $E_{D+p}(s) = s^{(p)}$ for $p = 1, \ldots, \lceil m/n \rceil$, where extra zeros are appended to s to make its length a multiple of n. Let $\mathcal{E} = \{E_1, \ldots, E_r\}$, where $r := D + \lceil m/n \rceil$. Clearly, the family is injective. Furthermore, explicitness of the family is due to the the explicitness of G and the fact that r is polynomial.

To prove the input-restricting property, assume towards a contradiction that there exist r sets $\mathcal{U}_1, \ldots, \mathcal{U}_r \subseteq \{0,1\}^n$ with cardinality $|\mathcal{U}_1| + \cdots + |\mathcal{U}_r| \leq 2^{n(1-\epsilon)}$ such that $|\mathcal{S}| > \delta \cdot (|\mathcal{U}_1| + \cdots + |\mathcal{U}_r|)$, where $\mathcal{S} := \{s \in \{0,1\}^m \mid E_p(s) \in \mathcal{U}_p \text{ for all } p = 1, \ldots, r\}$. Also, define $\mathcal{U} := \bigcup_{p=1}^{r} \mathcal{U}_p$. Clearly, in G we have $\Gamma(\mathcal{S}) \subseteq \mathcal{U}$ by the definition of $\mathcal{E}$, and in particular $|\Gamma(\mathcal{S})| \leq |\mathcal{U}|$. If $|\mathcal{S}| \leq K$, then $|\mathcal{U}| \geq |\Gamma(\mathcal{S})| \geq \delta^{-1} \cdot |\mathcal{S}| > \delta^{-1} \cdot \delta \cdot (|\mathcal{U}_1| + \cdots + |\mathcal{U}_r|) \geq |\mathcal{U}|$, which leads to a contradiction. If $|\mathcal{S}| > K$, take $\mathcal{S}' \subseteq \mathcal{S}$ such that $|\mathcal{S}'| = K$. Clearly, $\Gamma(\mathcal{S}') \subseteq \Gamma(\mathcal{S})$. Additionally, $|\mathcal{U}| \geq |\Gamma(\mathcal{S})| \geq |\Gamma(\mathcal{S}')| \geq \gamma \cdot |\mathcal{S}'| = \gamma \cdot K > 2^{n(1-\epsilon)}$, for n large enough by the choice of ϵ, which is a contradiction.

Finally, the family is invertible if $\lceil m/n \rceil$ is constant: Given the sets $\mathcal{U}_1, \ldots, \mathcal{U}_{D+\lceil m/n \rceil}$, the algorithm simply enumerates all $s \in \{0,1\}^m$ such that $E_p(s) \in \mathcal{U}_p$ for all $p = D+1, \ldots, D + \lceil m/n \rceil$, and keeps only those satisfying $E_p(s) \in \mathcal{U}_p$ for all $p = 1, \ldots, D$. This inversion algorithm runs in time $\text{poly}(n) \cdot |\mathcal{U}_{D+1}| \cdots |\mathcal{U}_{D+\lceil m/n \rceil}|$. □

For example, if a family exists with $K = 2^{n(1-\eta)}$ and constant expansion factor $\gamma > 1$, then $1 - \frac{\log K\gamma}{n} = \eta - o(1)$, and hence the family is (m, γ^{-1}, η)-input restricting. It remains to show that an explicit family of unbalanced expander graphs with sufficiently small (i.e. polynomially-bounded) left-degree exists. Much work in this area has been devoted to *lossless* unbalanced expanders, i.e. with $\gamma \approx D$, but the best known constructions [32, 26] for this case for our choice of parameters lead to either super-polynomial degree or a much too small bound K. However, we are satisfied even if the expansion factor is much smaller than the left-degree, as long as the latter stays small, and it is possible to obtain such graphs by appropriately composing known constructions. In Appendix C.1 we prove the following theorem.

Theorem 10. *For all polynomials γ and constants $\eta \in (0,1)$, and all functions m (polynomially-bounded in n), there exists an explicit family of expander graphs $G = (V_1, V_2, E)$ with $V_1 = \{0,1\}^m$, $V_2 = \{0,1\}^n$ which is $(2^{n(1-\eta)}, \gamma)$-expanding and has left-degree polynomially-bounded in n.*

Note the techniques we discuss in Appendix C.1 even allow to obtain slightly stronger results, for instance allowing η to be a moderately vanishing function (cf. the discussion at the end of Appendix C.1). Combining this with Lemma 9 we see that for all constants $\epsilon \in (0,1)$ there exist explicit (m, δ, ϵ)-input-restricting families with δ^{-1} polynomial in n. We note, however, that by dropping the explicitness requirement, families with much better parameters exist. In particular, the following result is proved in Appendix C.2.

Lemma 11. *Let K and γ be arbitrary such that $K \cdot \gamma \leq 2^n$, and let m be such that $m \geq n$. There exists a graph $G = (V_1, V_2, E)$ where $V_1 = \{0,1\}^m$ and $V_2 = \{0,1\}^n$ which is (K, γ)-expanding and with left-degree $D = \left\lceil \frac{1+\gamma \log e + m}{n - \log(K\gamma)} + \gamma \right\rceil$.*

For example, setting $m = \ell = 2n$, $\gamma = 1$ and $K = 2^{n(1-\epsilon)}$, we obtain left-degree $D = 1 + \frac{2}{\epsilon} + (\log e + 1)/(\epsilon \cdot n)$. For $\epsilon = \frac{1}{4}$ and $n = 128$, this leads to a family of size 12 by Lemma 9. Furthermore in this case $t = 7$ and $\rho = 4$, and all these values do not grow with n. (And a similar reasoning applies to all constants $\epsilon > 0$.) With these parameters, the construction is of practical interest, as it only relies on the design of a secure component function $\{0,1\}^n \to \{0,1\}^n$ which

may be very efficient. We hope this to motivate further research in de-randomizing families of unbalanced expander graphs for a wider range of parameters.

5 Constructing Public Random Oracles

In this section, we first review (a slightly generalized version of) the *prefix-free Merkle-Damgård* construction [13]. Let n be the given output size, and let $\ell \geq n$. We are given both a compression function $f : \{0,1\}^{b+\ell} \to \{0,1\}^{\ell}$ and a *prefix-free padding scheme*, that is, a mapping $\mathsf{pad} : \{0,1\}^* \to \left(\{0,1\}^b\right)^+$ such that $\mathsf{pad}(s)$ is not a prefix of $\mathsf{pad}(s')$ for all distinct $s, s' \in \{0,1\}^*$. The *prefix-free Merkle-Damgård construction* $\mathbf{pfMD}_{b,\ell,n}(f)$ proceeds as follows. On input $s \in \{0,1\}^*$, it computes $s_1 \| \cdots \| s_l = \mathsf{pad}(s)$ (with $s_i \in \{0,1\}^b$) and the chaining values $v_i := f(s_i, v_{i-1})$ for all $1 \leq i \leq l$, where v_0 is set to some initialization vector $IV \in \{0,1\}^{\ell}$. Finally, the construction outputs the first n bits of v_l. The following theorem easily[10] follows from Theorem 2 in [13].

Theorem 12. *Let* $\mathbf{F} : \{0,1\}^{\ell+b} \to \{0,1\}^{\ell}$ *be a puRF and let* $\mathbf{O} : \{0,1\}^* \to \{0,1\}^n$ *be a puRO. Then* $\mathbf{pfMD}_{b,\ell,n}(\cdot)$ *is an* (α', σ')*-reduction of* $\mathbf{O}$ *to* $\mathbf{F}$ *with* $\alpha'(k) = \mathcal{O}((l_{\max} \cdot k)^2 \cdot 2^{-\ell})$ *and* $\sigma'(k) = k$*, where* $l_{\max}$ *is the maximal length (of the padding) of a message input to the construction.*

We note that there exists a trade-off between the number of queries and the length of the queries to the construction.[11] This issue is inevitable in all iterated constructions. We take now ℓ, $b > 0$ as in the above explanation, and some $\epsilon > 0$. We set $m := \ell + b$, and we let $\mathcal{E}$ be an explicit (m, δ, ϵ)-input restricting family of functions. If given only a compression function $\mathbf{R}' : \{0,1\}^{n+\phi(n)} \to \{0,1\}^n$ (for $\phi(n)$ defined as in Section 3.3), we obtain a construction $\mathbf{pfMD}_{b,\ell,n}(\mathbf{C}^{\mathcal{E}}_{\epsilon,m,\ell}(\cdot))$ which replaces calls to the compression functions by calls to the construction $\mathbf{C}^{\mathcal{E}}_{\epsilon,m,\ell}(\cdot)$. We obtain the following theorem using Lemma 2.

Theorem 13. *The construction* $\mathbf{pfMD}_{b,\ell,n}(\mathbf{C}^{\mathcal{E}}_{\epsilon,m,\ell}(\cdot))$ *is an* $(\overline{\alpha}, \overline{\sigma})$*-reduction of a puRO* $\mathbf{O} : \{0,1\}^* \to \{0,1\}^n$ *to* $\mathbf{R}'$*, where* $\overline{\alpha}(k) = \alpha((l_{\max}+1)k) + \alpha'((\delta+1)k)$ *and* $\overline{\sigma}(k) = \delta \cdot k$*, with* α *and* α' *as in Theorems 3 and 12, respectively.*

Setting $\ell > 2n(1-\epsilon)$ leads to security for all distinguishers such that $l_{\max} \cdot k \leq \Theta(2^{n(1-\epsilon)})$. We finally note that our approach also works with all other known constructions of a public random oracle from a public compression function, as for example the constructions of [6, 12], or other constructions discussed in [13].

Setting ϵ small enough provides high levels of security for properties like preimage resistance, second preimage resistance, multicollision resistance, or CTFP preimage resistance [18], and also excludes the existence of attacks for these properties (up to the obtained bound), that is, even with respect to adversaries which perform enough queries to find collisions for the component function $f : \{0,1\}^n \to \{0,1\}^n$.

[10] The only difference with respect to the original result is that we allow the chaining value to be larger than the output value, i.e. $\ell > n$. The validity of our theorem follows from the simple observation (which we do not formalize) that dropping some bits of the output is a *perfect* reduction of public random function to a perfect random function with longer output size.

[11] A possible distinguishing strategy would consist of doing few very long queries, instead of many queries, and security is guaranteed only as long as $l_{\max} \cdot k < 2^{\ell/2}$.

Acknowledgments

This research was partially supported by the Swiss National Science Foundation (SNF), project no. 200020-113700/1. It is also a pleasure to thank Thomas Holenstein, Krzysztof Pietrzak, and Vassilis Zikas for helpful discussions.

References

[1] W. Aiello and R. Venkatesan, "Foiling birthday attacks in length-doubling transformations," in *Advances in Cryptology — EUROCRYPT '96*, vol. 1070 of *Lecture Notes in Computer Science*, pp. 307–320, 1996.

[2] J. H. An and M. Bellare, "Constructing VIL-MACs from FIL-MACs: Message authentication under weakened assumptions," in *Advances in Cryptology — CRYPTO '99*, vol. 1666 of *Lecture Notes in Computer Science*, pp. 252–269, 1999.

[3] A. Baltz, G. Jäger, A. Srivastav, and A. Ta-Shma, "An explicit construction of sparse asymmetric connectors." Available at `http://www.cse.wustl.edu/~jaegerg/publ/explconn.pdf`., 2003.

[4] M. Bellare, O. Goldreich, and H. Krawczyk, "Stateless evaluation of pseudorandom functions: Security beyond the birthday barrier," in *Advances in Cryptology — CRYPTO '99*, vol. 1666 of *Lecture Notes in Computer Science*, pp. 270–287, 1999.

[5] M. Bellare, J. Kilian, and P. Rogaway, "The security of the cipher block chaining message authentication code," *Journal of Computer and System Sciences*, vol. 61, no. 3, pp. 362–399, 2000.

[6] M. Bellare and T. Ristenpart, "Multi-property-preserving hash domain extension and the EMD transform," in *Advances in Cryptology — ASIACRYPT '06*, vol. 4284 of *Lecture Notes in Computer Science*, pp. 299–314, 2006.

[7] M. Bellare and P. Rogaway, "Random oracles are practical: A paradigm for designing efficient protocols," in *CCS '93: Proceedings of the 1st ACM conference on Computer and Communications Security*, pp. 62–73, 1993.

[8] R. Canetti, "Universally composable security: A new paradigm for cryptographic protocols," in *FOCS '01: Proceedings of the 42nd IEEE Symposium on Foundations of Computer Science*, pp. 136–145, 2001.

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

[10] M. Capalbo, O. Reingold, S. Vadhan, and A. Wigderson, "Randomness conductors and constant-degree lossless expanders," in *STOC '02: Proceedings of the 34th Annual ACM Symposium on Theory of Computing*, pp. 659–668, 2002.

[11] J. L. Carter and M. N. Wegman, "Universal classes of hash functions," *Journal of Computer and System Sciences*, vol. 18, no. 2, pp. 143–154, 1979.

[12] D. Chang, S. Lee, M. Nandi, and M. Yung, "Indifferentiable security analysis of popular hash functions with prefix-free padding," in *Advances in Cryptology — ASIACRYPT '06*, vol. 4284 of *Lecture Notes in Computer Science*, pp. 283–298, 2006.

[13] J.-S. Coron, Y. Dodis, C. Malinaud, and P. Puniya, "Merkle–Damgård revisited: How to construct a hash function," in *Advances in Cryptology — CRYPTO '05*, vol. 3621 of *Lecture Notes in Computer Science*, pp. 430–448, 2005.

[14] I. B. Damgård, "A design principle for hash functions," in *Advances in Cryptology — CRYPTO '89*, vol. 435 of *Lecture Notes in Computer Science*, pp. 416–427, 1989.

[15] Y. Dodis and P. Puniya, "On the relation between the ideal cipher and the random oracle models," in *Theory of Cryptography — TCC 2006*, vol. 3876 of *Lecture Notes in Computer Science*, pp. 184–206, 2006.

[16] J. J. Hoch and A. Shamir, "Breaking the ICE — finding multicollisions in iterated concatenated and expanded (ICE) hash functions," in *Fast Software Encryption — FSE '06*, vol. 4047 of *Lecture Notes in Computer Science*, pp. 179–194, 2006.

[17] A. Joux, "Multicollisions in iterated hash functions. application to cascaded constructions," in *Advances in Cryptology — CRYPTO '04*, vol. 3152 of *Lecture Notes in Computer Science*, pp. 306–316, 2004.

[18] J. Kelsey and T. Kohno, "Herding hash functions and the Nostradamus attack," in *Advances in Cryptology — EUROCRYPT '06*, vol. 4004 of *Lecture Notes in Computer Science*, pp. 183–200, 2006.

[19] J. Kelsey and B. Schneier, "Second preimages on n-bit hash functions for much less than 2^n work," in *Advances in Cryptology — EUROCRYPT '05*, vol. 3494 of *Lecture Notes in Computer Science*, pp. 474–490, 2005.

[20] S. Lucks, "A failure-friendly design principle for hash functions," in *Advances in Cryptology — ASIACRYPT '05*, vol. 3788 of *Lecture Notes in Computer Science*, pp. 474–494, 2005.

[21] U. Maurer, "Indistinguishability of random systems," in *Advances in Cryptology — EUROCRYPT '02*, vol. 2332 of *Lecture Notes in Computer Science*, pp. 110–132, 2002.

[22] U. Maurer, "Abstract models of computation in cryptography," in *Cryptography and Coding 2005*, vol. 3796 of *Lecture Notes in Computer Science*, pp. 1–12, 2005.

[23] U. Maurer, R. Renner, and C. Holenstein, "Indifferentiability, impossibility results on reductions, and applications to the random oracle methodology," in *Theory of Cryptography — TCC 2004*, vol. 3378 of *Lecture Notes in Computer Science*, pp. 21–39, 2004.

[24] U. Maurer and J. Sjödin, "Single-key AIL-MACs from any FIL-MAC," in *ICALP 2005: Proceedings of the 32nd International Colloquium on Automata, Languages and Programming*, vol. 3580 of *Lecture Notes in Computer Science*, pp. 472–484, 2005.

[25] R. C. Merkle, "A certified digital signature," in *Advances in Cryptology — CRYPTO '89*, vol. 435 of *Lecture Notes in Computer Science*, pp. 218–238, 1989.

[26] T. Moran, R. Shaltiel, and A. Ta-Shma, "Non-interactive timestamping in the bounded storage model," in *Advances in Cryptology — CRYPTO '04*, vol. 3152 of *Lecture Notes in Computer Science*, pp. 460–476, 2004. Full version available at `http://cs.haifa.ac.il/~ronen/online_papers/timestamping-full.ps`.

[27] N. Nisan and A. Ta-Shma, "Extracting randomness: a survey and new constructions," *Journal of Computer and System Sciences*, vol. 58, no. 1, pp. 148–173, 1999.

[28] J. Patarin and A. Montreuil, "Benes and butterfly schemes revisited," in *8th International Conference on Information Security and Cryptology - ICISC 2005*, vol. 3935 of *Lecture Notes in Computer Science*, pp. 92–116, 2005.

[29] B. Pfitzmann and M. Waidner, "A model for asynchronous reactive systems and its application to secure message transmission," in *SP '01: Proceedings of the 2001 IEEE Symposium on Security and Privacy*, pp. 184–200, 2001.

[30] V. Shoup, "On fast and provably secure message authentication based on universal hashing," in *Advances in Cryptology — CRYPTO '96*, vol. 1109 of *Lecture Notes in Computer Science*, pp. 313–328, 1996.

[31] V. Shoup, "Lower bounds for discrete logarithms and related problems," in *Advances in Cryptology — EUROCRYPT '97*, vol. 1233 of *Lecture Notes in Computer Science*, pp. 256–266, 1997.

[32] A. Ta-Shma, C. Umans, and D. Zuckerman, "Lossless condensers, unbalanced expanders, and extractors," in *STOC '01: Proceedings of the 33rd Annual ACM Symposium on Theory of Computing*, pp. 143–152, 2001.

A Impossibility of Extending Random Primitives

We prove that if a public random primitive $\mathbf{R}$ with N-bit table is extended to a public random primitives $\mathbf{R}'$ with N'-bit table, where $N' > N$, then we cannot guarantee security against distinguishers retrieving at least $2N + 1$ bits. The result is an application of the techniques from [23] to public random primitives, and for completeness we provide a self-contained proof here. (Note that the results from [23] apply to a wider range of systems.)

Lemma 14. *Let $\mathbf{R}$ and $\mathbf{R}'$ be public random primitives with N and N'-bit function tables, respectively, where $N' > N$. Furthermore, let $\mathbf{C}(\cdot)$ be a deterministic and stateless construction. Then, for all $t > 0$ (with $N+t \leq N'$) and all (not necessarily efficient) simulators $\mathbf{S}$, there exists a distinguisher $\mathbf{D}$ which retrieves $2N+t$ bits, and such that $\Delta^{\mathbf{D}}([\mathbf{R}, \mathbf{C}(\mathbf{R})], [\mathbf{S}(\mathbf{R}'), \mathbf{R}]) \geq 1-2^{-t}$.*

Proof. Define $\mathbf{H}_1 := [\mathbf{R}, \mathbf{C}(\mathbf{R})]$ and $\mathbf{H}_2 := [\mathbf{S}(\mathbf{R}'), \mathbf{R}']$. Without loss of generality assume that the public and the private interfaces are accessed bit-wise as N- and N'-bit tables. We consider the following distinguisher $\mathbf{D}$ which, given the system $\mathbf{H}_b = [\mathbf{H}_{\text{pub}}, \mathbf{H}_{\text{priv}}]$ (for $b \in \{1, 2\}$), first retrieves all N bits from $\mathbf{H}_{\text{pub}}$. Denote the resulting string as $\overline{R} \in \{0,1\}^N$. Note that the construction $\mathbf{C}(\cdot)$ can be seen as a mapping $\{0,1\}^N \to \{0,1\}^{N'}$, and the distinguisher (locally) computes the first $N + t$ bits $\overline{R}' \in \{0,1\}^{N+t}$ of $\mathbf{C}(\overline{R})$. Finally, it retrieves the first $N + t$ bits $\widetilde{R}' \in \{0,1\}^{N+t}$ of $\mathbf{H}_{\text{priv}}$. If $\overline{R}' = \widetilde{R}'$, it outputs 1, and 0 otherwise. Clearly $\mathsf{P}^{\mathbf{D}\circ\mathbf{H}_1}(\overline{R}' = \widetilde{R}') = 1$. Note that, independently of the simulator $\mathbf{S}$, there are at most 2^N values the random variable $\overline{R}'$ can take on, and let $\overline{\mathcal{R}}'$ be the set of these values. Therefore, we have $\mathsf{P}^{\mathbf{D}\circ\mathbf{H}_2}(\overline{R}' = \widetilde{R}') \leq \mathsf{P}^{\mathbf{D}\circ\mathbf{H}_2}(\widetilde{R}' \in \overline{\mathcal{R}}') \leq 2^N \cdot 2^{-(N+t)} = 2^{-t}$, which implies the statement of the lemma. □

This result has two main interpretations:

(i) If N is small (say polynomial in some understood security parameter), then there exists no efficient construction which extends $\mathbf{R}$, not even by a single bit, and not even with computational security. (This is due to the fact that in this case the distinguisher in the proof of the lemma is efficient.)

(ii) If we want to extend the domain of a public random function $\mathbf{R} : \{0,1\}^n \to \{0,1\}^n$ to $m > n$ bits, then we cannot hope to get security for adversaries making more than $2^{n+1} + 1$ queries.[12] (And this paper addresses the question of how close to this bound we can get.)

B Insecurity of the Benes-Construction

Aiello and Venketasan [1] proposed a construction named *Benes* (or *Double Butterfly*) for extending the domain of a (private) random function with security beyond the birthday barrier.[13] The construction is an instantiation of our general paradigm of Section 3.1. In this section, we show that its security in the case of public random functions is inherently bounded by the birthday bound. This should help clarify the crucial role of the functions $E_1, \ldots, E_r$ in our approach. We also stress that this attack can be adapted to hold even with respect to the *honest-but-curious* variant of indifferentiability introduced by Dodis and Puniya [15].

Formally, we look at the following variant of the original construction: We are given four random functions $\mathbf{F}_1, \mathbf{F}_2 : \{0,1\}^{2n} \to \{0,1\}^{2n}$ and $\mathbf{G}_1, \mathbf{G}_2 : \{0,1\}^{2n} \to \{0,1\}^n$. The construction $\mathbf{BE} : \{0,1\}^{2n} \to \{0,1\}^n$ takes an input $s = s^{(1)} \| s^{(2)}$, and computes first $w(s) = w^{(1)}(s) \| w^{(2)}(s) = \mathbf{F}_1(s^{(1)}) \oplus \mathbf{F}_2(s^{(2)})$ and outputs $\mathbf{G}_1(w^{(1)}(s)) \oplus \mathbf{G}_2(w^{(2)}(s))$. (We note that the original construction has $2n$-bit output, our attack however works even for the case of n-bit output.) Furthermore, let $\mathbf{R} : \{0,1\}^{2n} \to \{0,1\}^n$ be a public random function. For notational consistency with the proof of Theorem 3, we define

$$\mathbf{H}_1 := [\mathbf{F}_1, \mathbf{F}_2, \mathbf{G}_1, \mathbf{G}_2, \mathbf{BE}(\mathbf{F}_1, \mathbf{F}_2, \mathbf{G}_1, \mathbf{G}_2)]$$
$$\mathbf{H}_2 := [\mathbf{S}(\mathbf{R}), \mathbf{R}],$$

for an arbitrary simulator $\mathbf{S}$. We consider three types of queries: The first two types are $\mathbf{F}$-queries, with form (F, p, u), for $p = 1, 2$ and $u \in \{0,1\}^n$, and $\mathbf{G}$-queries with form (G, q, v) for $q = 1, 2$ and $v \in \{0,1\}^n$, which are both answered by the corresponding puRF's in $\mathbf{H}_1$ and by the simulator in $\mathbf{H}_2$, as well as $\mathbf{R}$-queries of form (R, s), for $s \in \{0,1\}^{2n}$, which are answered by the construction $\mathbf{BE}$ in $\mathbf{H}_1$ and by $\mathbf{R}$ in $\mathbf{H}_2$.

We construct a distinguisher $\mathbf{D}$ which — regardless of the simulator $\mathbf{S}$ — distinguishes $\mathbf{H}_1$ and $\mathbf{H}_2$ which constant probability when making approximately $2^{n/2}$ queries. Let $s_1, \ldots, s_{\overline{k}} \in \{0,1\}^n$ be fixed values for some even integer $\overline{k}$. The distinguisher $\mathbf{D}$ proceeds as follows. It first makes $\mathbf{F}$-queries $(\mathrm{F}, 1, s_i)$ for all $i = 1, \ldots, \overline{k}$, obtaining values $U_1, \ldots, U_{\overline{k}} \in \{0,1\}^{2n}$, and $\mathbf{F}$-queries $(\mathrm{F}, 2, s_j)$ for $j = 1, \ldots, \overline{k}$; let $V_1, \ldots, V_{\overline{k}} \in \{0,1\}^{2n}$ denote the resulting values. We define for all $i, j \in \{1, \ldots, \overline{k}\}$ the random variable $W_{ij} := U_i \oplus V_j$. The distinguisher $\mathbf{D}$ looks

[12] Actually, for the information-theoretic setting, one can even prove the stronger statement that there exists a distinguisher retrieving $N + t$ bits from the private interface only and distinguishing with advantage $1 - 2^{-t}$. This is due to the fact that the statistical distance of the first $N + t$ bits of $\mathbf{C}(\mathbf{R})$ from the uniform distribution is at least $1 - 2^{-t}$. However, in this case, if N is polynomially-bounded, the distinguisher is not necessarily efficient.

[13] In [1] optimal security is claimed, but the result turns out to be partially incorrect. However, the construction achieves security beyond the birthday barrier. This can be seen using the techniques from [21]. Also, in [28] direct proofs of improved bounds are given.

for $i \neq i'$ and $j \neq j'$ such that $W_{ij} = U_i \oplus V_j = U_{i'} \oplus V_{j'} = W_{i'j'}$. Note that this also implies that $W_{ij'} = U_i \oplus V_{j'} = U_{i'} \oplus V_j = W_{i'j}$ by rearranging terms. Finally, $\mathbf{D}$ performs four $\mathbf{R}$-queries $(\mathrm{R}, s_i \| s_j)$, $(\mathrm{R}, s_{i'} \| s_{j'})$, $(\mathrm{R}, s_i \| s_{j'})$, and $(\mathrm{R}, s_{i'} \| s_j)$. Denote by Y_1, Y_2, Y_3, and Y_4 the respective answers. If $Y_1 = Y_2$ and $Y_3 = Y_4$,the distinguisher outputs 1. In any other case (in particular also if such i, j and i', j' do not exist), it outputs 0.

Let $\mathcal{E}$ be the event that $W_{ij} = W_{i'j'}$ holds for some $i \neq i'$ and $j \neq j'$. Furthermore, let $\mathcal{K} := \{1, \ldots, \overline{k}/2\}$, and $\overline{\mathcal{K}} := \{\overline{k}/2 + 1, \ldots, \overline{k}\}$. We have

$$\mathsf{P}^{\mathbf{D}\circ\mathbf{H}_1}(\mathcal{E}) \geq \mathsf{P}^{\mathbf{D}\circ\mathbf{H}_1}\Big(\bigvee_{i,j\in\mathcal{K},i',j'\in\overline{\mathcal{K}}} W_{ij} = W_{i'j'} \Big) \geq \sum_{i,j\in\mathcal{K},i',j'\in\overline{\mathcal{K}}} \mathsf{P}^{\mathbf{D}\circ\mathbf{H}_1}(W_{ij} = W_{i'j'}) - \sum_{\substack{i,j,\bar{i},\bar{j}\in\mathcal{K},i',j',\bar{i}',\bar{j}'\in\overline{\mathcal{K}} \\ \{(i,j),(i',j')\}\neq\{(\bar{i},\bar{j}),(\bar{i}',\bar{j}')\}}} \mathsf{P}^{\mathbf{D}\circ\mathbf{H}_1}(W_{ij} = W_{i'j'} \wedge W_{\bar{i}\bar{j}} = W_{\bar{i}'\bar{j}'})$$

where the last inequality follows from the fact that $\mathsf{P}(\bigvee_{i=1}^r \mathcal{A}_i) \geq \sum_{i=1}^r \mathsf{P}(\mathcal{A}_i) - \sum_{1\leq i<i'\leq r} \mathsf{P}(\mathcal{A}_i \wedge \mathcal{A}_{i'})$ for all events $\mathcal{A}_1, \ldots, \mathcal{A}_r$. It is easy to see that W_{ij} and $W_{i'j'}$ are independent if $i \neq i'$ and $j \neq j'$, and thus $\sum_{i,j\in\mathcal{K},i',j'\in\overline{\mathcal{K}}} \mathsf{P}(W_{ij} = W_{i'j'}) = \frac{\overline{k}^4}{16} 2^{-2n}$. For the second sum, we consider two cases. First, assume that $(i,j) \neq (\bar{i},\bar{j})$, and $(i',j') \neq (\bar{i}',\bar{j}')$. Then, the random variables $W_{ij}, W_{i'j'}, W_{\bar{i}\bar{j}}$, and $W_{\bar{i}'\bar{j}'}$ are independent. Note that there are $\frac{1}{2}\left(\frac{\overline{k}^2}{4}\left(\frac{\overline{k}^2}{4} - 1\right)\right)^2 \leq \frac{\overline{k}^8}{512}$ possibilities to choose four such random variables, and in this case $\mathsf{P}(W_{ij} = W_{i'j'} \wedge W_{\bar{i}\bar{j}} = W_{\bar{i}'\bar{j}'}) = 2^{-4n}$. The second case takes place whenever either $(i,j) = (\bar{i},\bar{j})$ or $(i',j') = (\bar{i}',\bar{j}')$ holds. We have $\frac{\overline{k}^2}{4}\frac{\overline{k}^2}{4}\left(\frac{\overline{k}^2}{4} - 1\right) \leq \frac{\overline{k}^6}{64}$ ways of choosing the indices, and in this case $\mathsf{P}(W_{ij} = W_{i'j'} \wedge W_{\bar{i}\bar{j}} = W_{\bar{i}'\bar{j}'}) = 2^{-4n}$. Therefore,

$$\mathsf{P}^{\mathbf{D}\circ\mathbf{H}_1}(Y_1 = Y_2 \wedge Y_3 = Y_4) \geq \mathsf{P}^{\mathbf{D}\circ\mathbf{H}_1}(\mathcal{E}) \geq \frac{\overline{k}^4}{16} 2^{-2n} - \frac{\overline{k}^8}{512} 2^{-4n} - \frac{\overline{k}^6}{64} 2^{-4n}.$$

To bound $\mathsf{P}^{\mathbf{D}\circ\mathbf{H}_2}(Y_1 = Y_2 \wedge Y_3 = Y_4)$, note that given any simulator $\mathbf{S}$ making $\sigma(k)$ queries when queried k times and which ensures that $Y_1 = Y_2 \wedge Y_3 = Y_4$ holds with probability ϵ, then we can combine $\mathbf{S}$ and $\mathbf{D}$ into an adversary $\mathbf{A}$ that makes at most $4 + \sigma(2\overline{k})$ queries to $\mathbf{R}$ and finds $x \neq x'$ and $y \neq y'$ such that $\mathbf{R}(x\|y) = \mathbf{R}(x'\|y')$ and $\mathbf{R}(x\|y') = \mathbf{R}(x'\|y)$ with probability ϵ. However, it is not hard to see that the probability that some adversary finds such values within k queries is at most $k^2 \cdot 2^{-2n}$. Therefore,

$$\begin{aligned}\Delta^{\mathbf{D}}(\mathbf{H}_1, \mathbf{H}_2) &\geq \left|\mathsf{P}^{\mathbf{D}\circ\mathbf{H}_1}(Y_1 = Y_2 \wedge Y_3 = Y_4) - \mathsf{P}^{\mathbf{D}\circ\mathbf{H}_1}(Y_1 = Y_2 \wedge Y_3 = Y_4)\right| \\ &\geq \frac{\overline{k}^4}{16} 2^{-2n} - \frac{\overline{k}^8}{256} 2^{-4n} - \frac{\overline{k}^6}{64} 2^{-4n} - (4 + \sigma(2\overline{k}))^2 \cdot 2^{-2n}.\end{aligned}$$

Setting $\sigma(k) = k \cdot \mathrm{poly}(n)$, and $\overline{k} = 2^{n/2}$ leads to constant distinguishing advantage.

C Proofs for Section 4

C.1 Proof of Theorem 10

In this section, we provide a construction[14] of highly-unbalanced expander graphs with polynomial left-degree. We first review some basic notation needed throughout this section. Recall that

[14]To our knowledge, a very similar construction appears in an unpublished manuscript [3], hence the results of this section should not be considered an original contribution of this paper.

the *statistical distance* of two random variables X and Y with the same range $\mathcal{X}$ is $d(X,Y) = \frac{1}{2}\sum_{x\in\mathcal{X}} |\mathsf{P}_X(x) - \mathsf{P}_Y(x)|$ and furthermore this quantity equals $\max_{\mathcal{A}\subseteq\mathcal{X}} |\mathsf{P}(X \in \mathcal{A}) - \mathsf{P}(Y \in \mathcal{A})|$. The *min-entropy* of a random variable X is defined as $H_\infty(X) = -\log \max_{x\in\mathcal{X}} \mathsf{P}_X(x)$ and X is called a *k-source* if $H_\infty(X) \geq k$. Furthermore, it is a (k,ϵ)-source if there exists a k-source Y such that $d(X,Y) \leq \epsilon$. Of course, the same notions are defined for probability distributions instead of random variables. In the following, U_d will denote a uniformly-distributed d-bit random string which is independent from any other value. Also, the notation $\mathsf{P}_X \cdot \mathsf{P}_Y$ denotes the joint distribution of X and Y when they are chosen independently according to P_X and P_Y, respectively.

We make use of the notion of (simple) *randomness conductors* [10], which naturally generalizes *randomness extractors*. In particular, we also consider a slight modification of the original notion which generalizes strong extractors.

Definition 6. A function $C : \{0,1\}^m \times \{0,1\}^d \to \{0,1\}^n$ is a $(k_{\max}, a, \epsilon)$-*conductor* if for any $0 \leq k \leq k_{\max}$ and any k-source X over $\{0,1\}^m$ the output $C(X,U_d)$ is a $(k+a,\epsilon)$-source. The function C is a *strong* $(k_{\max}, a, \epsilon)$-*conductor* if $[C(X,U_d),U_d]$ is a $(k+a+d,\epsilon)$-source for all k-sources X with $0 \leq k \leq k_{\max}$. Finally, a conductor is *extracting* if $k_{\max} = n - a$.

One is generally interested in constructing *explicit* families of conductors, that is, (asymptotic) families of conductors which are computable in polynomial-time. To our knowledge, the best construction of an explicit strong conductor has the following parameters (cf. the full version of [26] for a proof.[15])

Theorem 15. *For every $m \geq n$ and every constant $\epsilon > 0$, there exists an explicit strong $(k_{\max}, -\Delta, \epsilon)$-conductor $C : \{0,1\}^m \times \{0,1\}^d \to \{0,1\}^n$, where $\Delta = \Delta(\epsilon) = \mathcal{O}(1)$ and $d = d(m,n,k_{\max},\epsilon) = \mathcal{O}(\log m + \log^3(k_{\max}))$, for all $k_{\max} \leq n + \Delta$.*

It is not difficult to see that a conductor can be interpreted as an unbalanced bipartite expander graph (this is indeed the starting point of [10]). However, we cannot use the result Theorem 15 directly, as we need $k_{\max} = \Theta(n)$, and this leads to super-polynomial degree. In order to overcome this problem, we introduce the following natural weakening of conductors.

Definition 7. A function $C : \{0,1\}^m \times \{0,1\}^d \to \{0,1\}^{nt}$ is a $(k_{\max}, a, \epsilon, \alpha)$-somewhere conductor if for all $0 \leq k \leq k_{\max}$ and all k-sources X over $\{0,1\}^m$ there exists a function $I : \{0,1\}^m \to \{1,\ldots,t\} \cup \{\perp\}$ such that $\mathsf{P}_{I(X)}(\perp) \leq \alpha$ and $\mathsf{P}_{C^{(i)}(X,U_d)|I(X)=i}$ is a $(k+a,\epsilon)$-source for all $i = 1,\ldots,t$ with $\mathsf{P}_{I(X)}(i) > 0$, where $C(X,U_d) = C^{(1)}(X,U_d)\| \cdots \|C^{(t)}(X,U_d)$, and $C^{(i)}(X,U_d) \in \{0,1\}^n$ for all $i = 1,\ldots,t$.

Given a function $C : \{0,1\}^m \times \{0,1\}^d \to \{0,1\}^{nt}$, we construct a graph $G_C = (V_1, V_2, E)$ where $V_1 := \{0,1\}^m$, $V_2 := \{0,1\}^n$, and $(x,z) \in E$ if and only if there exists $i \in \{1,\ldots,t\}$ and $y \in \{0,1\}^d$ such that $C^{(i)}(x,y) = z$. The following lemma generalizes a result from [32].

Lemma 16. *If $C : \{0,1\}^m \times \{0,1\}^d \to \{0,1\}^{nt}$ is a $(k_{\max}, \epsilon, \alpha, a)$-somewhere conductor with $\alpha < 1$, then G_C as above is a $(2^{k_{max}}, 2^a(1-\epsilon))$-expander graph with left degree 2^d.*

Proof. Let $\mathcal{X} \subseteq \{0,1\}^m$ with $|\mathcal{X}| \leq 2^{k_{\max}}$. Consider the source X which is uniformly distributed over $\mathcal{X}$, and let $k := H_\infty(X) = \log|\mathcal{X}| \leq k_{\max}$. Let $I : \{0,1\}^m \to \{1,\ldots,t\} \cup \{\perp\}$ be the function which is guaranteed to exist (for the source X), and fix an arbitrary i such that $\mathsf{P}_{I(X)}(i) > 0$.

[15] Actually, the proof in [26] considers a variant of strong extractors, called *strong universal extractors*, which give the additional guarantee that there exists a subset of the output bits which is almost uniformly-distributed.

Let $\mathcal{Z}$ be the support of $\mathsf{P}_{C^{(i)}(X,U_d)|I(X)=i}$. Clearly, $\mathcal{Z} \subseteq \Gamma(\mathcal{X})$. Moreover, there exists a $(k+a)$-source Z which satisfies $d(\mathsf{P}_{C^{(i)}(X,U_d)|I(X)=i}, Z) \leq \epsilon$, that is $\epsilon \geq \sum_{z\in\mathcal{Z}} \mathsf{P}_{C^{(i)}(X,U_d)|I(X)=i}(z) - \mathsf{P}_Z(z) = 1 - \sum_{z\in\mathcal{Z}} \mathsf{P}_Z(z) \geq 1 - |\mathcal{Z}| \cdot 2^{-k-a}$ by the definition of the statistical distance. By rearranging terms, we obtain $|\Gamma(\mathcal{X})| \geq (1-\epsilon)\cdot 2^{k+a} = (1-\epsilon)\cdot 2^a \cdot |\mathcal{X}|$. □

Let $C_1 : \{0,1\}^m \times \{0,1\}^{d_1} \to \{0,1\}^{d_2}$ and $C_2 : \{0,1\}^m \times \{0,1\}^{d_2} \to \{0,1\}^n$ be functions. Also, for a string $x \in \{0,1\}^m$, denote as $x_{(a,b)}$ the string consisting of the bits $x_a, x_{a+1}, \ldots, x_{b-1}, x_b$, with extra 0's to make its length equal to m. (If $b < a$, the string is the string 0^m.) We let $C : \{0,1\}^m \times \{0,1\}^{d_2} \to \{0,1\}^{(m+1)(d_1+d_2+n)}$ be such that $C(x,y) = C^{(1)}(x,y)\| \cdots \|C^{(m+1)}(x,y)$, where for all $1 \leq i \leq m+1$ we define

$$z_1^{(i)} := y \qquad z_2^{(i)} := C_1(x_{(i,m)}, y) \quad \text{and} \quad z_3^{(i)} := C_2(x_{(1,i-1)}, z_2^{(i)})$$

and we set $C^{(i)}(x,y) := z_1^{(i)}\|z_2^{(i)}\|z_3^{(i)} \in \{0,1\}^{d_1+d_2+n}$. The following lemma extends Theorem 3 from [27] to our setting. As the proof is very similar, we only provide a brief proof sketch.

Lemma 17. *Let $s > 0$ be given, and C be constructed as above. If C_1 is a strong (a_1, ϵ_1)-extracting conductor, and C_2 is a strong (k_2, a_2, ϵ_2)-conductor, then C is a $(d_2 - a_1 + k_2 + s, \min\{a_1, a_1 + a_2\} + d_1 - s, \epsilon_1 + \epsilon_2, 8m \cdot 2^{-s/3})$-somewhere conductor.*

Proof sketch. In the following, let $k_1 := d_2 - a_1$. Let X be a k-source with $k \leq k_1 + k_2 + s$. On the one hand, if X is such that $H_\infty(X) = k = k' + s$ with $k' \leq k_1$, then $[Z_1^{(1)}, Z_2^{(1)}] = [U_{d_1}, C^{(1)}(X, U_{d_1})]$, which is a $(k + a_1 + d_1 - s, \epsilon_1)$-source, and thus this also holds for the variable $[Z_1^{(1)}, Z_2^{(1)}, Z_3^{(1)}]$. On the other hand, if X is such that $H_\infty(X) = k = s + k_1 + k'$, where $k' \leq k_2$, then as in [27] there exists a selector function $I : \{0,1\}^m \to \{1, \ldots, m+1\} \cup \{\perp\}$ such that

1. $\mathsf{P}_{I(X)}(\perp) \leq 8m \cdot 2^{-s/3}$,
2. If $\mathsf{P}_{I(X)|X_{(1,i-1)}}(i, x_{(1,i-1)}) > 0$, then $H_\infty(X_{(i,m)}|I = i \wedge X_{(1,i-1)} = x_{(1,i-1)}) \geq k_1$, and
3. $H_\infty(X_{(1,i-1)}|I = i) \geq k'$.

In particular, this means that the distribution $\mathsf{P}_{Z_1^{(i)}Z_2^{(i)}|I(X)=iX_{(1,i-1)}=x_{(1,i-1)}}$ is ϵ_1-close to the distribution $\mathsf{P}_{U_{d_1}U_{d_2}|I(X)=i,X_{(1,i-1)}=x_{(1,i-1)}} = \mathsf{P}_{U_{d_1}} \cdot \mathsf{P}_{U_{d_2}}$ for all $x_{(1,i-1)}$ with the property that $\mathsf{P}_{I(X)|X_{(1,i-1)}}(i, x_{(1,i-1)}) > 0$. This implies in particular that $P_{Z_1^{(i)}Z_2^{(i)}X_{(1,i-1)}|I(X)=i}$ is ϵ_1-close to $\mathsf{P}_{U_{d_1}} \cdot \mathsf{P}_{U_{d_2}} \cdot \mathsf{P}_{X_{(1,i-1)}|I(X)=i}$. Furthermore, the distribution $\mathsf{P}_{U_{d_1}} \cdot \mathsf{P}_{U_{d_2}C(X_{(1,i-1)},U_{d_2})|I=i}$ is a $(k'+a_2+d_1+d_2, \epsilon_2)$-source. Also, by the triangle inequality, the distribution $\mathsf{P}_{Z_1^{(i)}Z_2^{(i)}Z_3^{(i)}|I(X)=i}$ is a $(k + a_1 + a_2 + d_1 - s, \epsilon_1 + \epsilon_2)$-source, and this concludes the proof. □

In the following, we instantiate both functions C_1 and C_2 using Theorem 15. First, set $s = 3 \cdot \log(9m) = 3\log m + \mathcal{O}(1)$ (note that with this $8m \cdot 2^{-s/3} < 1$). Also, fix some constant $\epsilon > 0$. Given $k_{\max} = (1-\eta)n$, we choose $\bar{n} := k_{\max}$, and

$$\begin{aligned} d_2 &= d(m, \bar{n}, k_{\max}, \epsilon) = \mathcal{O}(\log^3(n)) \\ d_1 &= \max\{d(m, d_2, d_2 + \Delta, \epsilon), 2\Delta + s + \log\gamma - \log(1-2\epsilon)\} = \mathcal{O}(\log n), \end{aligned}$$

and then take $C_1 : \{0,1\}^m \times \{0,1\}^{d_1} \to \{0,1\}^{d_2}$ and $C_2 : \{0,1\}^m \times \{0,1\}^{d_2} \to \{0,1\}^{\bar{n}}$ according to the theorem. (For C_1, we potentially need a longer seed, and the construction simply ignores the extra bits.) Then, C as above leads to a $(k_{\max}, d_1 - 2\Delta - s, 2\epsilon, \alpha)$-somewhere conductor

with $\alpha < 1$. Note that $(1-2\epsilon)2^{d_1-2\Delta-s} \geq \gamma$. Furthermore, for n sufficiently large, $\bar{n}+d_1+d_2 \leq n$, hence Lemma 16 gives the desired expander.

The proof works even if one takes any $\eta = \omega\left(\frac{\log^3(n)}{n}\right)$, since $\bar{n}+d_1+d_2 \leq n$ still holds for n large enough. Also note that we have made use of Lemma 17 in a very simple way, and (with some additional work in choosing parameters carefully) it allows to construct expanders for the case where η is even smaller. The results of [3] discuss this case.

C.2 Proof of Lemma 11

Let $V_1 := \{0,1\}^m$ and $V_2 := \{0,1\}^n$, where $m > n$, and let D be the left-degree of the graph (to be fixed later) and γ the desired expansion factor (which in particular satisfies $K \cdot \gamma \leq 2^n$ and $\gamma \leq D$). Also, for notational convenience let $M := 2^m$ and $N := 2^n$. The proof is an application of the probabilistic method. We sample a graph as follows: For every vertex $v \in V_1$, we pick D (not necessarily distinct) neighbors uniformly at random from V_2. Let $\mathcal{S}$ be some subset of V_1, with $i := |\mathcal{S}| \leq K$. Note that whenever $|\Gamma(\mathcal{S})| < \gamma \cdot |\mathcal{S}|$ holds, there exists a set $\mathcal{T}$ of size $\gamma \cdot |\mathcal{S}|$ (without loss of generality assume this value to be an integer) such that $\Gamma(\mathcal{S}) \subseteq \mathcal{T}$. When sampling a graph as explained, the probability that all neighbors of $\mathcal{S}$ are in $\mathcal{T}$ is $(\gamma \cdot i/N)^{D \cdot i}$. Therefore, by the union bound, the probability that there exists a set $\mathcal{S}$ with $|\mathcal{S}| \leq K$ and $|\Gamma(\mathcal{S})| < \gamma \cdot |\mathcal{S}|$ is at most

$$\sum_{i=1}^{K} \binom{M}{i} \cdot \binom{N}{\gamma \cdot i} \cdot \left(\frac{\gamma \cdot i}{N}\right)^{D \cdot i} \leq \sum_{i=1}^{K} \left[e^{\gamma} \cdot M \cdot \left(\frac{\gamma \cdot i}{N}\right)^{D-\gamma} \right]^{i},$$

where we have bounded $\binom{M}{i} \leq M^i$ and $\binom{N}{\gamma i} \leq \left(\frac{eN}{\gamma i}\right)^{\gamma i}$. We now set $D := \frac{1+\gamma \cdot \log e + m}{n - \log(K\gamma)} + \gamma$, and it is easy to verify that

$$e^{\gamma} \cdot M \cdot \left(\frac{\gamma \cdot i}{N}\right)^{D-\gamma} \leq e^{\gamma} \cdot M \cdot \left(\frac{\gamma \cdot K}{N}\right)^{D-\gamma} = \frac{1}{2}.$$

With this value of D, the above sum is upper bounded by $\sum_{i=1}^{K} \frac{1}{2^i} < \sum_{i=1}^{\infty} \frac{1}{2^i} = 1$, and hence a good graph exists with positive probability.