

Naor-Reingold Goes Public: The Complexity of Known-key Security^{*}

Pratik Soni and Stefano Tessaro

University of California, Santa Barbara
{pratik.soni,tessaro}@cs.ucsb.edu

Abstract. We study the complexity of building secure block ciphers in the setting where the key is known to the attacker. In particular, we consider two security notions with useful implications, namely public-seed pseudorandom permutations (or psPRPs, for short) (Soni and Tessaro, EUROCRYPT '17) and correlation-intractable ciphers (Knudsen and Rijmen, ASIACRYPT '07; Mandal, Seurin, and Patarin, TCC '12).

For both these notions, we exhibit constructions which make only *two* calls to an underlying non-invertible primitive, matching the complexity of building a pseudorandom permutation in the secret-key setting. Our psPRP result instantiates the round functions in the Naor-Reingold (NR) construction with a secure UCE hash function. For correlation intractability, we instead instantiate them from a (public) random function, and replace the pairwise-independent permutations in the NR construction with (almost) $O(k^2)$ -wise independent permutations, where k is the arity of the relations for which we want correlation intractability.

Our constructions improve upon the current state of the art, requiring five- and six-round Feistel networks, respectively, to achieve psPRP security and correlation intractability. To do so, we rely on techniques borrowed from Impagliazzo-Rudich-style black-box impossibility proofs for our psPRP result, for which we give what we believe to be the first *constructive* application, and on techniques for studying randomness with limited independence for correlation intractability.

Keywords: Foundations, known-key security, pseudorandomness, psPRPs, correlation-intractability, limited independence.

^{*} © IACR 2018. This is the full version of a paper that appeared in the proceedings of EUROCRYPT 2018.

1 Introduction

1.1 Overview and Motivation

Block ciphers are traditionally used within modes of operation where they are instantiated under a *secret* key. Provable security results typically assume them to be good *pseudorandom permutations* (PRPs). This has motivated a large body of theoretical works on building PRPs from weaker or less structured components, e.g., through the Feistel construction and its variants [LR88,NR97,HR10].

Block ciphers are however also frequently used in settings where the key is fixed, or at least *known*. We refer to this as the *known-key setting*. For instance, it is common to rely on *permutations*¹ or (equivalently) *fixed-key* ciphers to build hash functions [BDPV08,RS08], authenticated encryption [AJN15], PRNGs [BDPV10,GT16], and even more involved objects, such as garbling schemes [BHKR13].

As there is no secret key to rely upon, it is less clear what kind of security properties block ciphers should satisfy in this setting. Hence, security proofs typically assume the cipher to behave like an ideal random permutation on each key. A number of design paradigms for block ciphers (cf. e.g. [CPS08,HKT11,ABD⁺13,DSSL16,DKT16,DS16] to mention a few results) are therefore analyzed in terms of *indifferentiability* [MRH04], an ideal-model property which implies that for single-stage security games, the cipher inherits all properties of an ideal cipher. Still, the resulting proofs are notoriously involved, and the constructions more complex than seemingly necessary for the applications in which they are used. This is in sharp contrast with hash functions, where indifferentiability has helped shaping real-world designs.

OUR CONTRIBUTIONS. The only two exceptions to the above indifferentiability-based approach we are aware of are the notions of a *public-seed pseudorandom permutation* (psPRP) [ST17] and of *correlation-intractable block ciphers* [KR07,MPS12]. Block ciphers satisfying variants of both have been shown to be sufficient to instantiate several schemes that otherwise only enjoyed security proofs assuming the cipher is ideal. Yet, the complexity of actually building these primitives from simpler objects is not understood.

In this work, we present constructions for each of the notions which only make *two* calls to an underlying non-invertible round function. All of our constructions are instantiations of the *Naor-Reingold construction* [NR97], which is the most efficient known approach to build a secure PRP, and we thus show that it retains meaningful properties when the seed is made *public* under appropriate assumptions on the round functions. The previously known best constructions require Feistel networks with five rounds (for psPRPs) [ST17] and six rounds (for correlation intractability) [MPS12], and in both cases the security proofs relied indirectly on (weakened) forms of indifferentiability, inheriting seemingly unnecessary complexity. Here, we introduce substantially different techniques to bypass limitations of existing proofs, borrowing from areas such as black-box separations and applications of limited-independence.

We stress that our focus here is on foundations, and more specifically, breaking complexity barriers. While we follow the good practice of giving concrete bounds, we make no claims that these are suitable for practical applications. We hope however to spur quantitative research in this direction.

¹ Permutations, as in the sponge construction, correspond to the extreme case where there is only one possible key to choose from.

1.2 Public-Seed Pseudorandomness via the NR Construction

We start with our results on *public-seed pseudorandom permutations* (or psPRPs, for short), a notion recently introduced in [ST17], which considers a family of permutations E on n -bit strings, indexed by a seed s . (This could be obtained from a block cipher.) Ideally, we would like $E_s(\cdot)$ and $E_s^{-1}(\cdot)$ to be indistinguishable from ρ and ρ^{-1} (for a random permutation ρ), *even* if the seed s is known to the distinguisher. This is obviously impossible, yet an approach to get around this borrowed from the UCE framework [BHK13] is to *split* the distinguisher into two stages. A first stage, called the *source* S , gets access to either (E_s, E_s^{-1}) or (ρ, ρ^{-1}) , but does *not* know s , and then passes on some leakage L to a second-stage PPT D , the *distinguisher*, which learns s , but has no access to the oracle any more. If E is indeed secure, D will not be able to guess which one of the two oracles S had access to. This is very similar to the security definition of a UCE H , the only difference is that there the source accesses either H_s or a random *function* f .

Clearly, nothing is gained if L is unrestricted, and thus restrictions on S are necessary. Two classes of sources were in particular considered in [ST17], *unpredictable* and *reset-secure* sources, inspired by analogous notions for UCEs. The former demands that when the source S accesses ρ and ρ^{-1} , an (unbounded)² predictor P given the leakage L cannot then guess any of S 's queries (and their inverses). In contrast, the latter notion demands that a computationally unbounded distinguisher R , given L cannot tell apart whether it is given oracle access to the *same* permutation ρ , or an independent one, within a polynomial number of queries. Being a psPRP for all unpredictable sources is a potentially weaker assumption than being a psPRP for reset-secure sources, since every unpredictable source is reset-secure, but not vice versa.

APPLICATIONS. PsPRPs for such restricted source classes are a versatile notion. For example, a psPRP for all reset-secure sources can be used to instantiate the permutation within permutation-based hash functions admitting indistinguishability-based security proofs, such as the sponge construction [BDPV08] (which underlies SHA-3), turning them into a UCE-secure hash function sufficient for a number of applications, studied in multiple works [BHK13, DGG⁺15, MH14, BH15, BS16]. Also, [ST17] shows that psPRPs for unpredictable sources are sufficient to instantiate garbling schemes obtained from fixed-key block ciphers [BHKR13].

CONSTRUCTING PSPRPs: PREVIOUS WORK. But do psPRPs exist at all? Soni and Tessaro [ST17] show that they are implied by sufficiently strong UCEs:

Theorem (Informal) [ST17]. *The five-round Feistel construction, with round functions instantiated from a UCE H for reset-secure sources, is a psPRP for reset-secure sources.*

This left two obvious questions open, however: **(1)** Whether the number of rounds can be reduced, and **(2)** whether the same holds for unpredictable sources, too. The techniques of [ST17], based on proving a weaker notion of indistinguishability for five-round Feistel, fail to help answering both questions.

OUR CONTRIBUTIONS. We solve both questions, and even more in fact, showing that the Naor-Reingold (NR) construction [NR97] solves *both* (1) and (2). In particular, let H be a family of functions from $n + 1$ bits to n , and let P be a family of permutations on $2n$ bit strings. Then, the

² Computational versions of these notions can be defined, but the resulting notions can easily be shown impossible under the assumption that IO exists [BFM14], and are ignored in this paper.

NR construction on seed $\mathbf{s} = (s, s^{\text{in}}, s^{\text{out}})$ and input $u \in \{0, 1\}^{2n}$, outputs $v \in \{0, 1\}^{2n}$, where

$$\begin{aligned} x_0 \parallel x_1 &\leftarrow \mathsf{P}_{s^{\text{in}}}(u), x_2 \leftarrow \mathsf{H}_s(0 \parallel x_1) \oplus x_0, \\ x_3 &\leftarrow \mathsf{H}_s(1 \parallel x_2) \oplus x_1, v \leftarrow \mathsf{P}_{s^{\text{out}}}^{-1}(x_3 \parallel x_2). \end{aligned}$$

The key point here is that P only needs to satisfy a weak non-cryptographic property, namely that for a random s and for any distinct $u \neq u'$, the right halves of $\mathsf{P}_s(u)$ and $\mathsf{P}_s(u')$ only collide with negligible probability. Therefore, only two calls to a “cryptographically hard” round function H are made. Naor and Reingold [NR97] showed that NR is a (strong) PRP whenever H is a pseudorandom function. Here, we show the following public-seed counterparts:

Theorems 1 and 2. (Informal) *The NR construction, with round functions instantiated with a UCE H for X -sources, is a psPRP for X -sources, where $X \in \{\text{reset-secure, unpredictable}\}$.*

A detailed overview of our techniques is given below in Section 1.4. We remark here that such UCEs are of course strong, and the question of basing these on simpler assumptions is wide open. Still, we believe such results to be very important: First off, they show relations among notions, and getting a UCE (without any injectivity structure) is possibly simpler in practice than in theory (i.e., using the compression function of SHA-256). Second, even if we instantiate H from a random oracle (which gives a good UCE [BHK13]), the result *is* useful, as this would give us a simple instantiation of a (seeded) permutation in applications which are not even known to follow from full-fledged indistinguishability, as discussed by Mittelbach [Mit14].

1.3 Correlation Intractability

The notion of *correlation intractability* (CI) of *hash functions* was introduced by Canetti, Goldreich, and Halevi [CGH98] as a weakening of a random oracle. CI naturally extends to permutations and block ciphers [KR07, MPS12]: Given the seed s , an adversary should not be able to find an input-output pair (u, v) such that $\mathsf{E}_s(u) = v$ and such that $(u, v) \in R$, where R is a hard-to-satisfy relation for a truly random permutation, a so-called *evasive* relation. This, in turn, can be generalized to k -ary relations, where k input-output pairs are to be provided. CI is well-known not to hold in the standard model for arbitrary evasive relations.³ Therefore, here, we target constructions in ideal models.

APPLICATIONS. CI has important applications – for example, let E be a permutation family on $2n$ -bit strings. Then, for $n < m < 2n$, consider the function family H from m bits to n bits such that

$$\mathsf{H}_s(x) = \mathsf{E}_s(x \parallel 0^{2n-m})[1 \dots n],$$

i.e., this outputs the first n bits of $\mathsf{E}_s(x \parallel 0^{2n-m})$. Then, it is not too hard to show that if E satisfies CI for evasive binary relations, then H is collision resistant – indeed, a collision yields two distinct pairs $(u_1, v_1), (u_2, v_2)$ where $u_1[m + 1 \dots 2n] = u_2[m + 1 \dots 2n] = 0^{2n-m}$, whereas $v_1[1 \dots n] = v_2[1 \dots n]$. Along similar lines, one can prove that H can be used to instantiate the Fiat-Shamir transform [FS87] whenever E satisfies CI for *unary* relations. And so on.

³ Though, of course, it could be true for specific interesting relations.

CORRELATION INTRACTABILITY FOR FEISTEL NETWORKS. Indifferentiability is easily seen to imply CI, and therefore, by [DS16], the Feistel construction with 8 rounds is correlation intractable. In fact, Mandal, Patarin, and Seurin [MPS12] observed that a weaker notion of indifferentiability, called sequential indifferentiability, is sufficient for CI, and could show that 6 rounds are enough. It is known that the 5-round Feistel construction is *not* correlation intractable for evasive 4-ary relations (an attack was given in [MPS12]). Other weaker notions of indifferentiability are known to imply CI, but do not appear to lead to any complexity improvements for constructions from non-invertible primitives [ABM14,CS16].

OUR RESULTS. We study the correlation intractability of the NR construction described above where the two calls to H are replaced by two calls to (seedless) public random function f from $n+1$ to n bits, and the seed of the construction only consists of the seeds for P . (This is similar to the model of Ramzan and Reyzin [RR00], although they consider PRP security, and secret seeds.)

In general, this basic form of the NR construction cannot be correlation intractable – indeed, P can be instantiated by one-round Feistel with a pairwise-independent round function, and generic attacks against the correlation intractability of four-round Feistel would still apply. We show however the following result:

Theorem 3. (Informal) *For any constant $k = O(1)$, if P^{-1} is an almost $O(k^2)$ -wise independent permutation, then the NR construction is correlation intractable for every k -ary evasive relation.*

For unary relations (i.e., $k = 1$), we can in fact show that instantiating P with one-round Feistel using a 10-wise independent round function suffices. As this is effectively a four-round Feistel network, this confirms that no generic attacks exist for unary relations. Our result extends to non-constant k , however under some restrictions on the class of evasive relations for which we can prove correlation intractability. We believe an important part of our result is the technique we use, which gives a surprising paradigm to amplify CI unconditionally which we discuss below in Section 1.5.

LIMITATIONS. In contrast to existing 6-round results, our result is weaker in that it only covers evasive relations fully if $k = O(1)$. We are not aware of counter examples showing attacks for larger k 's, but our proof inherently fails. We note however that most applications of correlation intractability only require constant arity, and we leave the question of assessing whether six calls to a random function are necessary for arbitrary arity for future work.

1.4 Technical overview – psPRPs

Let us briefly recall the setting: For some PPT source S , which queries a permutation oracle on $2n$ -bit strings to produce a leakage L , we need to show that any PPT distinguisher D which learns L and $\mathbf{s} = (s, s^{\text{in}}, s^{\text{out}})$ cannot tell apart whether S was accessing NR using a UCE H (with seed \mathbf{s}) or a truly random permutation. We assume S is either (statistically) unpredictable (in Theorem 1) or reset-secure (in Theorem 2).

THE SOURCE \bar{S} . The natural approach we follow is to build another source, \bar{S} from S , for which H should be a secure UCE. This source thus accesses an oracle \mathcal{O} implementing a function from $n+1$ to n bits. It first samples seeds $s^{\text{in}}, s^{\text{out}}$ for P , and then simulates an execution of S . The oracle calls by the latter are processed by evaluating the NR construction using $s^{\text{in}}, s^{\text{out}}$, and the oracle $\mathcal{O}(\cdot)$ in lieu of $H(s, \cdot)$. Finally, when S produces its output L , \bar{S} outputs $(L, s^{\text{in}}, s^{\text{out}})$. We will show the following two facts:

- FACT 1. If S is unpredictable (w.r.t. the psPRP notion), then \bar{S} is unpredictable (w.r.t. the UCE notion).

- FACT 2. If S is reset-secure (w.r.t. the psPRP notion), then \bar{S} is reset-secure (w.r.t. the UCE notion).

Theorems 1 and 2 follow from Facts 1 and 2, respectively, by a fairly straightforward application of the (classical) indistinguishability of the NR construction with *random* round functions.⁴

THE UNPREDICTABLE CASE. Our approach to establish Fact 1 is inspired by an elegant proof of secure domain extension for UCEs via Wegman-Carter MACs [BHK14]. (The case of reset-secure sources will be more involved and use new techniques.)

Assume, towards a contradiction, that \bar{S} is *not* unpredictable; then there exists a strategy (not necessarily efficient) that given L and s^{in} and s^{out} , guesses one of the inner oracle queries of \bar{S} with non-negligible probability ε , when \bar{S} 's oracle is a random function from $n + 1$ to n bits. Imagine now that given $(L, s^{\text{in}}, s^{\text{out}})$ from \bar{S} , we *resample* an execution of \bar{S} (which in particular means re-sampling the oracle used by it) consistent with outputting $(L, s^{\text{in}}, s^{\text{out}})$, and look at the inner oracle queries in this virtual, re-sampled execution. Then, one can show that the real and the virtual executions are likely to share an oracle query, with probability roughly at least ε^2 , for our strategy to guess a query must be equally successful on the virtual execution.

We exploit this idea to build a predictor for the original source S , contradicting our hypothesis it is unpredictable. Note that S runs with a random permutation as its oracle, and produces leakage L . Imagine now we sample fresh seeds $s^{\text{in}}, s^{\text{out}}$ for P , and for each permutation query by S defining an input-output pair (u, v) , we define “fake” inputs x_0, x_1 from $x_0 \parallel x_1 = P_{s^{\text{in}}}(u)$ and $x_3 \parallel x_2 = P_{s^{\text{out}}}(v)$. Then, the indistinguishability of the NR construction from a random permutation, and the construction of \bar{S} , implies that if we resample a virtual execution of S consistent with leakage L , and compute the resulting fake inputs using s^{in} and s^{out} , then the real and the re-sampled execution will share a fake input with probability approx. ε^2 . The properties imposed on P then imply that with probability roughly ε^2 the real and the re-sampled execution must share the input (or output) of a permutation query. This leads naturally to a predictor that just re-samples an execution consistent with the leakage, and picks them as its prediction.

RESET-SECURITY. The case of reset-security is somewhat harder. Here we start from the premise that \bar{S} is not reset-secure: Hence, there exists an adversary \bar{R} which receives $L, s^{\text{in}}, s^{\text{out}}$ from \bar{S} , and can distinguish (with non-negligible advantage ε) being given access to the same random $f : \{0, 1\}^{n+1} \rightarrow \{0, 1\}^n$ used by \bar{S} from being given access to an independent f' . From this, we would like to build an adversary R which receives L from S , and can distinguish the setting where R and S are given access to the same random permutation ρ , from a setting where they access independent permutations ρ, ρ' .

The challenge here is that we want to simulate \bar{R} correctly, by using a permutation oracle ρ/ρ' rather than f/f' . To better see why this is tricky, say S is the source that queries its permutation oracle on a random $2n$ -bit string u , obtaining output v , and leaks $L = (u, v)$. (This defines the corresponding \bar{S} .)⁵ A clever \bar{R} on input $(L = (u, v), s^{\text{in}}, s^{\text{out}})$ could do the following: It computes $x_0 \parallel x_1 \leftarrow P(s^{\text{in}}, u)$ and $x_3 \parallel x_2 \leftarrow P(s^{\text{out}}, v)$. Then, it queries x_1 to its oracle, and outputs 1 iff the output equals $x_0 \oplus x_2$. This should always be true when \bar{R} accesses f , and almost never when it accesses f' .

⁴ A minor caveat is that we need indistinguishability even when s^{in} and s^{out} are revealed at the end of the interaction. We will show this to be true.

⁵ The reader should not be confused: \bar{S} is clearly not reset-secure, but remember we are in the setting of a proof by contradiction, so the reduction must work here, too.

The natural proof approach would now attempt to build R which runs \bar{R} accessing a simulated oracle consistent with the NR construction on the permutation queries made by S . However, the problem is that generically R does not know which queries S has made. Previous work [ST17] handled this by requiring the construction to satisfy a weaker notion of indifferenciability, called CP-sequential indifferenciability, which essentially implies that there exists a simulator that can simulate f consistently by accessing ρ and ρ^{-1} only, and only needs to know the queries \bar{R} makes to f . This would not work with NR and our \bar{R} , as the query x_1 is actually uniformly random, and the simulator would likely fail to set $x_0 \oplus x_2$ as the right output. This is why the approach of [ST17] ends up using the 5-round Feistel construction, as here \bar{R} 's attempt to evaluate the construction are readily detected, and answered consistently.

OUR PROOF STRATEGY VIA HEAVY-QUERY SAMPLING. Our main observation is that indifferenciability is an overkill in this setting. There is no reason \bar{R} should act adversarially to the simulator. Even more so, we can use everything \bar{R} knows, namely L , to our advantage! To do this, we use techniques borrowed from impossibility proofs in the random oracle model [IR89,BMG09]. Namely, R , on input L from S , first performs a number of permutation queries which are meant to include all of S 's likely queries to its oracle, at least when R and S are run with the same permutation oracle ρ . To do this, R samples executions of S consistent with L , and the partial knowledge of the oracle ρ acquired so far. Each time such a partial execution is sampled, all queries contained in it are made to ρ , and the process is repeated a number of times polynomial in $1/\varepsilon$. Then, R samples $s^{\text{in}}, s^{\text{out}}$, and internally defines an oracle $f : \{0, 1\}^{n+1} \rightarrow \{0, 1\}^n$ that will be used to simulate an execution of $\bar{R}^f(L, s^{\text{in}}, s^{\text{out}})$. To do this, R goes through all input-output pairs (u, v) for queries to ρ it has done while simulating executions of S ,⁶ and defines

$$f(0 \parallel x_1) \leftarrow x_0 \oplus x_2, f(1 \parallel x_2) \leftarrow x_1 \oplus x_3,$$

where $x_0 \parallel x_1 \leftarrow P_{s^{\text{in}}}(u)$ and $x_3 \parallel x_2 \leftarrow P_{s^{\text{out}}}(v)$. Then, f is defined to be random on every other input (this can be simulated via lazy sampling). The final output of the simulated \bar{R} is then R 's final output.

The core of our proof will show that when S and R share access to ρ , then the probability that R 's output is one is similar to that of \bar{R} outputting one when it accesses the same oracle as \bar{S} . This will combine properties of the NR construction (allowing us to switch between f and ρ), and similar arguments as those used in [IR89] to prove that R ensures consistency on all queries that matter.⁷

1.5 Technical overview – Correlation Intractability

Our approach towards achieving CI is based on the following blueprint. Let R be a relation which is evasive for permutations on $2n$ -bit strings, and let π, σ be permutations sampled from some given distribution (this will be meant to be instantiated unconditionally below). Then, we create a new relation $R_{\pi, \sigma}$ such that $(u, v) \in R$ iff $(\pi(u), \sigma(v)) \in R_{\pi, \sigma}$. The hope is to show that if R is an evasive relation, then $R_{\pi, \sigma}$ is hard to satisfy for a given construction \mathbf{E} which is only correlation-intractable for a subset of all evasive relations. Then, a new composed construction \mathbf{E}' which outputs $\sigma^{-1}(E_s(\pi(u)))$ on input u would be correlation intractable for all evasive relations, since satisfying R for \mathbf{E}' implies satisfying $R_{\pi, \sigma}$ for \mathbf{E} .

⁶ The actual simulation will be slightly more involved, for the benefit of simplifying the analysis.

⁷ We believe we could adapt our proof to use the better strategy of [BMG09] to get slightly better concrete parameters, yet we found adapting it to our setting not immediate.

TWO-ROUND FEISTEL. In our context, we instantiate E from a two-round Feistel network. That is, on input $x = x_0 \parallel x_1$, the two-round Feistel construction outputs $x_2 \parallel x_3$, where $x_2 \leftarrow f(0 \parallel x_1) \oplus x_0$ and $x_3 \leftarrow f(1 \parallel x_2) \oplus x_1$. In a model (as the one where we consider) where f is a random function to which the adversary is given access, this construction is *not* correlation intractable. For instance, take the (unary) relation which is satisfied by all input-output pairs $(x_0 \parallel x_1, x_2 \parallel x_3)$ where $x_1 = x_2$. This is clearly evasive, but trivial to satisfy for two-round Feistel. Worse is possible with k -ary relations.

However, many relations *are* hard, even for two-round Feistel. Take for instance any relation R with the property that for all x_0, x_1, x_2, x_3 , the number of x^* 's such that $(x^* \parallel x_1, x_2 \parallel x_3) \in R$ or $(x_0 \parallel x_1, x_2 \parallel x^*) \in R$ is at most $\delta \cdot 2^n$, for some negligible function δ . No adversary A making a polynomial number of queries to f will satisfy R , except with negligible probability. Indeed, when A queries (say) $y_2 \leftarrow f(1 \parallel x_2)$ for some x_2 , the only chance to produce a pair that satisfies R is $y_1 \leftarrow f(0 \parallel x_1)$ was previously queried for some x_1 , and additionally, $(x_2 \oplus y_1 \parallel x_1, x_2 \parallel x_1 \oplus y_2) \in R$. But because y_2 is being set randomly, $x_1 \oplus y_2$ is also random, this can only hold with probability at most δ by our assumption on δ . Thus, the probability that this pair satisfies R is negligible, and the union bound over all pairs of queries shows A is unlikely to *ever* satisfy R .

WHERE DOES AMPLIFICATION COME FROM? Let R be a unary evasive relation. Now, imagine, again for the sake of an oversimplified illustration, that π and σ are random permutations. Then, we want to show that with high probability over the choice of π and σ , the relation $R_{\pi, \sigma}$ is hard for two-round Feistel, even if the adversary learns the entire description of π and σ . Indeed, find any x_0, x_1, x_2 , fix π , and fix $u = \pi^{-1}(x_0 \parallel x_1)$. Because R is evasive, there exists at most $\delta \cdot 2^{2n}$ v 's (for some negligible δ) such that $(u, v) \in R$ – call the set of such v 's R_u . Because σ is random, the probability that $\sigma^{-1}(x_2 \parallel z) \in R_u$ is at most δ , and thus the *expected* number of z such that $(x_0 \parallel x_1, x_2 \parallel z) \in R_{\pi, \sigma}$ is at most $\delta \cdot 2^n$, by linearity of expectation. A concentration bound will show that the probability that we are far from this expectation is indeed small, say at most 2^{-4n} . Taking a union bound over all x_0, x_1, x_2 shows that the probability this is true for any x_0, x_1, x_2 is at most 2^{-n} . (The symmetric argument when fixing x_1, x_2, x_3 can be handled analogously.) Thus, we have just argued that with high probability over the choice of π and σ , $R_{\pi, \sigma}$ is hard for two-round Feistel!

CHALLENGES. But obviously, this is not very useful– random permutations π, σ are inefficient to sample and describe. Also, the above result only holds for unary relations, and it is interesting to extend this to k -ary relations.

Our first insight is that the above argument only requires a bounded degree of randomness, and that (almost) t -wise independent permutations for a sufficiently small t are sufficient. We prove this using techniques for bounding sums of random variables with bounded independence [SSS93, BR94], though this will require significant adaptation because almost t -wise independent permutations do not quite produce outputs which are t -wise independent, as they are required to be distinct, and also, only approximate a random permutation. We will instantiate these by using Feistel networks with sufficiently many rounds, and t -wise independent round functions, using bounds from [HR10]. In fact, for the case of unary relations, we will show that we can instantiate these permutations from one single Feistel round with a 10-wise independent round function.

Moving on to k -ary relations presents even more challenges. Our approach is inherently combinatorial, whereas evasiveness is defined indirectly through the inability of an adversary to win a security game. For this reason, our result will only deal with relations R that satisfy a more structured notion of evasiveness, which we refer to as *strongly* evasive. Most relations of interest that we are aware of are strongly evasive, but evasiveness does not always imply strong evasiveness.

<pre> MAIN UCE^{S,D}_{m,n,H}, psPRP^{S,D}_{n,E}: (1^r, t) ←_{\$} S(ε), b ←_{\$} {0, 1} s₁, ..., s_r ←_{\$} H.Kg E.Kg f₁, ..., f_r ←_{\$} Funcs(m, n) ρ₁, ..., ρ_r ←_{\$} Perms(n) L ←_{\$} S^O(t) b' ←_{\$} D(s₁, ..., s_r, L) return b' = b </pre>	<pre> ORACLE $\mathcal{O}(i, x)$: // UCE^{S,D}_{m,n,H} if b = 1 then return H(s_i, x) else return f_i(x) ORACLE $\mathcal{O}(i, (\sigma, x))$: // psPRP^{S,D}_{n,E} if b = 1 then if σ = + then return E(s_i, x) else return E⁻¹(s_i, x) else if σ = + then return ρ_i(x) else return ρ_i⁻¹(x) </pre>
--	--

Fig. 1: Games to define UCE and psPRP security. Here, S is the source and D is the distinguisher. Boxed statements are only executed in the corresponding game.

However, as a result of independent interest, we show that strong evasiveness and evasiveness are related, and asymptotically equivalent when k is a constant.

2 Preliminaries

NOTATIONAL PRELIMINARIES. Throughout this paper, we denote by $\text{Funcs}(X, Y)$ the set of functions $X \rightarrow Y$, and in particular use the shorthand $\text{Funcs}(m, n)$ whenever $X = \{0, 1\}^m$ and $Y = \{0, 1\}^n$. We also denote by $\text{Perms}(X)$ the set of permutations on the set X , and analogously, $\text{Perms}(n)$ denotes the special case where $X = \{0, 1\}^n$. For $n \in \mathbb{N}$, we let $[n]$ denote the set $\{1, \dots, n\}$.

Our security definitions and proofs will often use games, as formalized by Bellare and Rogaway [BR06]. Typically, our games will have boolean outputs – that is, either `true` or `false` – and we use the shorthand $\Pr[\mathbf{G}]$ to denote the probability that a certain game outputs the value `true`, or occasionally 1 (when the output is binary, rather than boolean). Most results in this paper will be concrete, but natural asymptotic statements can be made by allowing all parameters to be functions of the security parameter.

A *function family* with input set X and output set Y is a pair of algorithms $\mathbf{F} = (\mathbf{F.Kg}, \mathbf{F.Eval})$, where the randomized *key (or seed) generation algorithm* $\mathbf{F.Kg}$ outputs a seed s , and the deterministic *evaluation algorithm* $\mathbf{F.Eval}$ takes as inputs a valid seed s and an input $x \in X$, and returns $\mathbf{F.Eval}(s, x) \in Y$. If $X = \{0, 1\}^m$ and $Y = \{0, 1\}^n$, we say that \mathbf{F} is a family of functions from m -bits to n -bits. We usually write $\mathbf{F}(s, \cdot) = \mathbf{F.Eval}(s, \cdot)$. A *permutation family* $\mathbf{P} = (\mathbf{P.Kg}, \mathbf{P.Eval})$ on n bits is the special case where $X = \{+, -\} \times \{0, 1\}^n$ and $Y = \{0, 1\}^n$, and for every s , there exists a permutation π_s such that $\mathbf{P.Eval}(s, (+, x)) = \pi_s(x)$ and $\mathbf{P.Eval}(s, (-, y)) = \pi_s^{-1}(y)$. We usually write $\mathbf{P}(s, \cdot) = \mathbf{P}(s, (+, \cdot))$ and $\mathbf{P}^{-1}(s, \cdot) = \mathbf{P}(s, (-, \cdot))$.

2.1 UCes and psPRPs

We review the UCE notion introduced in [BHK13], and the psPRP notion [ST17]. As explained in the latter work, they can be seen as instantiations of a general paradigm. Yet, we consider separate security games for better readability.

Concretely, let \mathbf{H} be function family from m -bits to n -bits. Let S be an adversary called the *source* and D an adversary called the *distinguisher*. We associate with them the game $\text{UCE}_{m,n,H}^{S,D}$

<p>MAIN f-Pred$_{m,n,S}^P$, p-Pred$_{n,S}^P$:</p> <p>$Q \leftarrow \emptyset$ $(1^r, t) \stackrel{\\$}{\leftarrow} S(\varepsilon)$ $f_1, \dots, f_r \stackrel{\\$}{\leftarrow} \text{Funcs}(m, n)$ $\rho_1, \dots, \rho_r \stackrel{\\$}{\leftarrow} \text{Perms}(n)$ $L \stackrel{\\$}{\leftarrow} S^\mathcal{O}(t)$ $Q' \stackrel{\\$}{\leftarrow} P(1^r, L)$ return $(Q \cap Q' \neq \emptyset)$</p> <p>ORACLE $\mathcal{O}(i, x)$: // f-Pred$_{m,n,S}^P$ $Q \leftarrow Q \cup \{(i, x)\}$ return $f_i(x)$</p> <p>ORACLE $\mathcal{O}(i, (\sigma, x))$: // p-Pred$_{n,S}^P$ if $\sigma = +$ then $y \leftarrow \rho_i(x)$ else $y \leftarrow \rho_i^{-1}(x)$ $Q \leftarrow Q \cup \{(i, x), (i, y)\}$ return y</p>	<p>MAIN f-Reset$_{m,n,S}^R$, p-Reset$_{n,S}^R$:</p> <p>done \leftarrow false; $(1^r, t) \stackrel{\\$}{\leftarrow} S(\varepsilon)$ $f_1^0, f_1^1, \dots, f_r^0, f_r^1 \stackrel{\\$}{\leftarrow} \text{Funcs}(m, n)$ $\rho_1^0, \rho_1^1, \dots, \rho_r^0, \rho_r^1 \stackrel{\\$}{\leftarrow} \text{Perms}(n)$ $L \stackrel{\\$}{\leftarrow} S^\mathcal{O}(t)$; done \leftarrow true $b \stackrel{\\$}{\leftarrow} \{0, 1\}$; $b' \stackrel{\\$}{\leftarrow} R^\mathcal{O}(1^r, L)$ return $b' = b$</p> <p>ORACLE $\mathcal{O}(i, x)$: // f-Reset$_{m,n,S}^R$ if \negdone then return $f_i^0(x)$ else return $f_i^b(x)$</p> <p>ORACLE $\mathcal{O}(i, (\sigma, x))$: // p-Reset$_{n,S}^R$ if \negdone then if $\sigma = +$ then return $\rho_i^0(x)$ else return $\rho_i^{0^{-1}}(x)$ else if $\sigma = +$ then return $\rho_i^b(x)$ else return $\rho_i^{b^{-1}}(x)$</p>
---	---

Fig. 2: Games to define unpredictability (left) and reset-security (right) of sources. Here, S is the source, P is the predictor and R is the reset-adversary. Boxed statements are only executed in the corresponding game.

depicted in Fig. 1. For a family \mathbf{E} of permutations on n -bits, the psPRP-security game $\text{psPRP}_{n,\mathbf{E}}^{S,D}$ differs in that \mathcal{O} allows for inverse queries, and the ideal object is a random permutation. The corresponding advantage metrics for an (S, D) are defined as

$$\begin{aligned} \text{Adv}_{m,n,\mathbf{H}}^{\text{uce}}(S, D) &= 2 \Pr \left[\text{UCE}_{m,n,\mathbf{H}}^{S,D} \right] - 1 \\ \text{Adv}_{n,\mathbf{E}}^{\text{psprp}}(S, D) &= 2 \Pr \left[\text{psPRP}_{n,\mathbf{E}}^{S,D} \right] - 1 . \end{aligned} \tag{1}$$

Note that we adopt the multi-key versions of UCE and psPRP security, as they are the most general, and they are not known to follow from the single-key case. Our treatment scales down to the single-key version by forcing the source to always choose $r = 1$.

We say that \mathbf{H} is UCE-secure for a class of sources \mathcal{S} if $\text{Adv}_{m,n,\mathbf{H}}^{\text{uce}}(S, D)$ is negligible for all PPT D and all sources $S \in \mathcal{S}$. Similarly, \mathbf{E} is psPRP secure for \mathcal{S} if $\text{Adv}_{n,\mathbf{E}}^{\text{psprp}}(S, D)$ is negligible for all PPT D and all sources $S \in \mathcal{S}$. It is known that \mathcal{S} cannot contain all PPT algorithms for security to be attainable. Next, we discuss two important classes of restrictions – unpredictable and reset-secure sources – considered in the literature [BHK13,BHK14,ST17].

UNPREDICTABLE SOURCES. Let S be a source and P be an adversary called the *predictor*. We associate with them games $\text{f-Pred}_{m,n,S}^P$ and $\text{p-Pred}_{n,S}^P$ of Fig. 2 which capture the fact that P cannot predict any of the queries of S (or their inverses), when the latter interacts with a random function from m bits to n bits, or respectively a random permutation on n -bit strings. The corresponding

advantage metrics are

$$\text{Adv}_{m,n,S}^{\text{f-pred}}(P) = \Pr[\text{f-Pred}_{m,n,S}^P] , \text{Adv}_{n,S}^{\text{p-pred}}(P) = \Pr[\text{p-Pred}_{n,S}^P] . \quad (2)$$

We say S is *statistically unpredictable* if $\text{Adv}_{m,n,S}^{\text{f-pred}}(P)$ (respectively, $\text{Adv}_{n,S}^{\text{p-pred}}(P)$) is negligible for all predictors P outputting a set Q' of polynomial size. An analogous notion of computational unpredictability can be defined, but it is unachievable if IO exists [BFM14], and is usually not needed for applications. We also note that what we formalize here is the notion of *simple* unpredictability – P is not permitted to query the underlying primitive. The notion was proved equivalent (asymptotically) for UCEs [BHK13] to a version where we give P access to the primitive. A similar proof follows for psPRPs.

RESET-SECURE SOURCES. Let S be a source and R be an adversary called the reset-adversary. We associate to them the games $\text{f-Reset}_{m,n,S}^R$ and $\text{p-Reset}_{n,S}^R$ of Fig. 2 which formalize the reset-security of S against a random function and a random permutation, respectively. The idea here is that R should not be able to tell apart whether S is accessing the same set of oracles it accesses, or not. This is captured via the advantage metrics

$$\text{Adv}_{m,n,S}^{\text{f-reset}}(R) = 2 \Pr[\text{f-Reset}_{m,n,S}^R] - 1 , \text{Adv}_{n,S}^{\text{p-reset}}(R) = 2 \Pr[\text{p-Reset}_{n,S}^R] - 1 .$$

We say S is *statistically reset-secure* if the corresponding advantage is negligible for all reset-adversaries R making a polynomial number of *queries* to their oracle, but which are otherwise computationally unrestricted. It is known that a (statistically) unpredictable source is (statistically) reset-secure, for both UCEs [BHK13] and psPRPs [ST17]. The converse is not true – S may query a fixed known input, and let L be the empty string. S is reset-secure in the strongest sense, while being easily predictable.

2.2 Evasive Relations, Correlation Intractability

In the following, a k -ary relation R over $X \times Y$ is a set of subsets $S \subseteq X \times Y$, where $1 \leq |S| \leq k$.⁸ We are going to consider relations which are *evasive* with respect to a random permutation.

EVASIVE RELATIONS. Given a relation R over $\{0, 1\}^m \times \{0, 1\}^m$, we consider the following advantage metric, involving an adversary A :

$$\text{Adv}_{R,m}^{\text{evp}}(A) = \Pr_{\pi} \left[S \stackrel{\$}{\leftarrow} A^{\pi, \pi^{-1}} : S \in R \wedge \forall (u, v) \in S : \pi(u) = v \right] ,$$

where $\pi \stackrel{\$}{\leftarrow} \text{Perms}(m)$. We say that a relation R is (q, δ) -evasive for a random permutation if $\text{Adv}_{R,m}^{\text{evp}}(A) \leq \delta$ for all adversaries making q queries.

CORRELATION INTRACTABILITY. Let \mathbf{M}^f be a permutation family on m -bits which makes oracle calls to a function f from n bits to ℓ bits, to be modeled as a random function. Let R be a k -ary relation. Let A be any (possibly unbounded) adversary. We associate to A , \mathbf{M} and R the following cri-advantage metric:

$$\text{Adv}_{R,\mathbf{M}}^{\text{cri}}(A) = \Pr_{s,f} \left[S \stackrel{\$}{\leftarrow} A^f(s) : S \in R \wedge \forall (u, v) \in S : \mathbf{M}^f(s, u) = v \right] ,$$

where $f \stackrel{\$}{\leftarrow} \text{Funcs}(n, \ell)$ and $s \stackrel{\$}{\leftarrow} \text{M.Kg}$.

⁸ We think of the elements as *sets*, rather than tuples – this is because looking ahead, it only makes sense in the context of correlation intractability to consider symmetric relation, as an adversary can always re-order its outputs.

3 Public-seed Pseudorandomness of Naor-Reingold

This section revisits the Naor-Reingold construction [NR97] in the public-seed setting. We prove that it transforms a UCE into a psPRP, for both unpredictable (Section 3.2) and reset-secure sources (Section 3.3). Before turning to these results, however, Section 3.1 reviews the construction and proves a strong statement about its indistinguishability.

3.1 The NR Construction and its Indistinguishability

Let P be a permutation family on the $2n$ -bit strings. We say that P is α -right-universal if $\Pr_{s \leftarrow \mathsf{P.Kg}} [\mathsf{P}_1(s, u) = \mathsf{P}_1(s, u')] \leq \alpha$ for all distinct $u, u' \in \{0, 1\}^{2n}$, where P_1 denote the second n -bit half of the output of P . Note that a pairwise-independent permutation is a good candidate of P , but a simpler approach is to employ one-round of Feistel with a pairwise independent hash function H as the round function, i.e., $\mathsf{P}(s, (u_0, u_1)) = (u_1, \mathsf{H}(s, u_1) \oplus u_0)$.

THE NAOR-REINGOLD (NR) CONSTRUCTION. Let H be a function family from $n + 1$ bits to n bits. We define the permutation family $\mathsf{NR} = \mathsf{NR}[\mathsf{P}, \mathsf{H}]$ on the $2n$ -bit strings, where $\mathsf{NR.Kg}$ outputs $(s, s^{\text{in}}, s^{\text{out}})$ such that $s \leftarrow \mathsf{H.Kg}$ and $s^{\text{in}}, s^{\text{out}} \leftarrow \mathsf{P.Kg}$. Further, forward evaluation proceeds as follows (the inverse is obvious):

Proc. $\mathsf{NR}((s, s^{\text{in}}, s^{\text{out}}), U)$:
 $x_0 \parallel x_1 \leftarrow \mathsf{P}(s^{\text{in}}, U)$, $x_2 \leftarrow \mathsf{H}(s, 0 \parallel x_1) \oplus x_0$,
 $x_3 \leftarrow \mathsf{H}(s, 1 \parallel x_2) \oplus x_1$, $V \leftarrow \mathsf{P}^{-1}(s^{\text{out}}, x_3 \parallel x_2)$, **return** V

Naor and Reingold [NR97] proved that the NR construction with random round functions is indistinguishable from a random permutation under chosen ciphertext attacks. We will need a stronger result, which we prove here, that this is true even when the seed of P is given to the adversary after it stops making queries, and when the adversary can make queries to multiple instances of the construction. It will be convenient to re-use the notation already in place for the psPRP framework, and we denote by $\text{Adv}_{2n, \mathsf{NR}[\mathsf{P}, \mathsf{F}]}^{\text{psprp}^+}(S, D)$ the advantage obtained by (S, D) in the $\text{psPRP}_{2n, \mathsf{NR}[\mathsf{P}, \mathsf{F}]}^{S, D}$ game, with the modification that D is *not* given the seed for F , only the seeds used by the permutation P .

Proposition 1 (Indistinguishability of the NR construction). *Let $\mathsf{F} = \mathsf{F}[n + 1, n]$ be the family of all functions from $n + 1$ to n bits, equipped with the uniform distribution. Further, let P be α -right-universal. For all S, D , where S makes q queries, $\text{Adv}_{2n, \mathsf{NR}[\mathsf{P}, \mathsf{F}]}^{\text{psprp}^+}(S, D) \leq q^2 \cdot (2\alpha + \frac{1}{2^{2n}})$.*

The proof of Proposition 1 can be found in Appendix A.1.

3.2 The Case of Unpredictable Sources

We first prove that the NR construction transforms a UCE function family for statistically unpredictable sources into a psPRP for statistically unpredictable sources. Our proof uses a technique inspired from that of Bellare, Hoang, and Keelveedhi [BHK14], given originally in the setting of UCE domain extension. Concretely, we prove the following.

Theorem 1 (NR security for unpredictable sources). *Let P be a α -right universal family of permutations on $2n$ -bit strings. Let H be a family of functions from $n + 1$ bits to n bits. Then, for*

all distinguishers D and sources S making overall q queries to their oracle, there exists \bar{D} and \bar{S} such that

$$\text{Adv}_{2n, \text{NR}[\text{P}, \text{H}]}^{\text{psPRP}}(S, D) \leq \text{Adv}_{n+1, n, \text{H}}^{\text{uce}}(\bar{S}, \bar{D}) + q^2 \left(2\alpha + \frac{1}{2^{2n}} \right). \quad (3)$$

Here, \bar{D} and D are roughly as efficient, and \bar{S} and S are similarly as efficient. In particular, \bar{S} makes $2q$ queries. Moreover, for every predictor \bar{P} , there exists a predictor P such that

$$\text{Adv}_{n+1, n, \bar{S}}^{\text{f-pred}}(\bar{P}) \leq q^2 \cdot \left(2\alpha + \frac{1}{2^{2n}} \right) + p \cdot \sqrt{2q^2\alpha + \text{Adv}_{2n, S}^{\text{p-pred}}(P)}, \quad (4)$$

where p is a bound on the size of the set output by \bar{P} .

The asymptotic interpretation is that if $n = \omega(\log(\lambda))$ and α is negligible, if S is (statistically) unpredictable, then so is \bar{S} . Further, if H is a UCE for all unpredictable sources, then NR is a psPRP for all statistically unpredictable sources.

We stress that the predictor P built in the proof does not preserve the efficiency of \bar{P} , which is not a problem, as we only consider statistical notions. While we do not elaborate in the proof, it turns out that the running time of P is *exponential* in the length of S 's leakage, thus the statement carries over to computational unpredictability if $L = O(\log \lambda)$.

Proof. We first consider three games, G_0 , G_1 , and G_2 . Game G_0 is the game $\text{psPRP}_{2n, \text{NR}[\text{P}, \text{F}]}^{S, D}$ in the case $b = 1$, and modified to return true if $b' = 1$. Game G_2 is the game $\text{psPRP}_{2n, \text{NR}[\text{P}, \text{F}]}^{S, D}$ in the case $b = 0$, and modified to return true if $b' = 1$. The intermediate game G_1 is obtained by modifying G_0 as follows: Initially, r random functions $f_1, \dots, f_r \stackrel{\$}{\leftarrow} \text{Funcs}(n+1, n)$ are sampled, and when evaluating the NR construction within \mathcal{O} queries, the evaluation of $\text{H}(s_i, b \| x)$ is replaced by an evaluation of the random function $f_i(b \| x)$. Then,

$$\text{Adv}_{2n, \text{NR}[\text{P}, \text{H}]}^{\text{psPRP}}(S, D) = (\Pr[\text{G}_0] - \Pr[\text{G}_1]) + (\Pr[\text{G}_1] - \Pr[\text{G}_2]).$$

We can directly get $\Pr[\text{G}_1] - \Pr[\text{G}_2] \leq q^2 (2\alpha + \frac{1}{2^{2n}})$ as a corollary of Proposition 1, since neither of G_1 and G_2 uses the seeds generated by H.Kg .

Going on, let us consider the new source \bar{S} which simulates an execution of S , and uses access to an oracle $\mathcal{O}(i, X)$, implementing for each i a function from $n+1$ bits to n bits, to internally simulate the round functions NR construction used to answer S 's queries. A formal description is in Figure 3. Also consider the distinguisher \bar{D} such that

$$\bar{D}(L' = (L, \mathbf{s}^{\text{in}}, \mathbf{s}^{\text{out}}), \mathbf{s}) = D(L, (\mathbf{s}, \mathbf{s}^{\text{in}}, \mathbf{s}^{\text{out}})),$$

where $\mathbf{s} = (s_1, \dots, s_r)$, $\mathbf{s}^{\text{in}} = (s_1^{\text{in}}, \dots, s_r^{\text{in}})$, and $\mathbf{s}^{\text{out}} = (s_1^{\text{out}}, \dots, s_r^{\text{out}})$. Therefore, G_0 and G_1 behave exactly as $\text{UCE}_{n+1, n, \text{H}}^{\bar{S}, \bar{D}}$ with challenge bits $b = 1$ and $b = 0$, respectively, with the only difference of outputting true whenever the distinguisher's output is $b' = 1$. Consequently, $\text{Adv}_{n+1, n, \text{H}}^{\text{uce}}(\bar{S}, \bar{D}) = \Pr[\text{G}_0] - \Pr[\text{G}_1]$.

The remainder of the proof relates the unpredictability of S and that of \bar{S} , establishing (4) in the theorem statement. The detailed argument is deferred to Appendix A.2. \square

<p>Proc. $\overline{S}(\varepsilon)$:</p> <p>$(1^r, t) \stackrel{\\$}{\leftarrow} S(\varepsilon)$</p> <p>return $(1^r, (1^r, t))$</p> <p>Proc. $\overline{S}^{\mathcal{O}}(1^r, t)$:</p> <p>$s_1^{\text{in}}, s_1^{\text{out}}, \dots, s_r^{\text{in}}, s_r^{\text{out}} \stackrel{\\$}{\leftarrow} \text{P.Kg}$</p> <p>$L \stackrel{\\$}{\leftarrow} S^{\mathcal{O}}(t)$</p> <p>return $(L, s_1^{\text{in}}, s_1^{\text{out}}, \dots, s_r^{\text{in}}, s_r^{\text{out}})$</p>	<p>Proc. $\overline{\mathcal{O}}(i, (\sigma, U))$:</p> <p>if $\sigma = +$ then</p> <p style="padding-left: 20px;">$x_0 \parallel x_1 \leftarrow \text{P}(s_i^{\text{in}}, U)$</p> <p style="padding-left: 20px;">$x_2 \leftarrow \mathcal{O}(i, 0 \parallel x_1) \oplus x_0, x_3 \leftarrow \mathcal{O}(i, 1 \parallel x_2) \oplus x_1$</p> <p style="padding-left: 20px;">$V \leftarrow \text{P}^{-1}(s_i^{\text{out}}, x_3 \parallel x_2)$</p> <p>else</p> <p style="padding-left: 20px;">$x_3 \parallel x_2 \leftarrow \text{P}(s_i^{\text{out}}, U)$</p> <p style="padding-left: 20px;">$x_1 \leftarrow \mathcal{O}(i, 1 \parallel x_2) \oplus x_3, x_0 \leftarrow \mathcal{O}(i, 0 \parallel x_1) \oplus x_2$</p> <p style="padding-left: 20px;">$V \leftarrow \text{P}^{-1}(s_i^{\text{in}}, x_0 \parallel x_1)$</p> <p>return V</p>
--	---

Fig. 3: The source \overline{S} in the proof of Theorems 1 and 2.

3.3 The Case of Reset-Secure Sources

Theorem 1’s importance stems mostly from the fact that it establishes the equivalence of psPRPs and UCEs for the case of (statistically) unpredictable sources. The question was left open in [ST17]. Many applications (e.g., instantiating the permutation within sponges, or any other indifferentiable hash construction) however require the stronger notion of reset-security. For this, [ST17] show that the five-round Feistel construction suffices, using a weaker variant of indifferentiability, and left open the question of whether four-rounds suffice.

We do better here: we prove that the NR construction transforms a UCE for statistically reset-secure sources into a psPRP for the same class of sources. The proof starts as the one of Theorem 1, but then shows that the source \overline{S} built therein is in fact statistically reset-secure whenever S is. This step will resort to a variant of the heavy-query sampling method of Impagliazzo and Rudich [IR89] to simulate a random oracle from the leakage which captures “relevant correlations” with what is learnt by the source.

Theorem 2 (NR security for reset-secure sources). *Let P be a α -right universal family of permutations on $2n$ -bit strings, and let H be a function family from $n + 1$ bits to n bits. Then, for all distinguishers D and sources S making overall q queries to their oracle, there exists \overline{D} and \overline{S} such that*

$$\text{Adv}_{2n, \text{NR}[\text{P}, \text{H}]}^{\text{psprp}}(S, D) \leq \text{Adv}_{n+1, n, \text{H}}^{\text{uce}}(\overline{S}, \overline{D}) + q^2 \left(2\alpha + \frac{1}{2^{2n}} \right). \quad (5)$$

Here, \overline{D} and D are roughly as efficient, and \overline{S} and S are similarly as efficient. In particular, \overline{S} makes $2q$ queries. Moreover, for every reset-adversary \overline{R} making p queries, there exists a reset-adversary R such that

$$\text{Adv}_{n+1, n, \overline{S}}^{\text{f-reset}}(\overline{R}) \leq 2\text{Adv}_{2n, S}^{\text{p-reset}}(R) + 4 \left(q + \frac{8qp^2}{\varepsilon} \ln(4p/\varepsilon) \right)^2 \left(2\alpha + \frac{1}{2^{2n}} \right), \quad (6)$$

where $\varepsilon := \text{Adv}_{n+1, n, \overline{S}}^{\text{f-reset}}(\overline{R})$. In particular, R makes $4qp^2/\varepsilon \cdot \ln(4p/\varepsilon)$ queries to its oracle.

Asymptotically, (6) implies that if \overline{R} exists making $p = \text{poly}(\lambda)$ queries, and achieving non-negligible advantage ε , then R makes also a polynomial number of queries, and achieves non-negligible advantage, as long as α is negligible, and $n = \omega(\log \lambda)$. Thus, reset-security of R yields reset-security of \overline{R} .

We also believe that the technique of Barak and Mahmoody [BMG09] can be used to reduce the $8qp^2/\varepsilon$ term to $O(qp)/\varepsilon$. We did not explore this avenue here, as the proof approach of [IR89] is somewhat easier to adapt to our setting.

Proof. The setup of the proof is identical to that in Theorem 1, in particular the construction of \bar{S} from S (and of \bar{D} from D .) The difference is in relating the reset-security of S and \bar{S} . In particular, let

$$\varepsilon := \text{Adv}_{n+1,n,\bar{S}}^{\text{f-reset}}(\bar{R}) = \Pr \left[\text{f-Reset}_{n+1,n,\bar{S}}^{\bar{R}} \mid b = 0 \right] - \Pr \left[\text{-f-Reset}_{n+1,n,\bar{S}}^{\bar{R}} \mid b = 1 \right] .$$

The RHS is the difference of the probabilities of \bar{R} outputting 0 in the cases $b = 0$ and $b = 1$ respectively. We are going to build a new adversary R against S which satisfies (6). We assume without loss of generality that \bar{R} is deterministic, and makes *exactly* p *distinct* queries to its oracle.

We start the proof with some game transitions that will lead naturally to the definition of the adversary R . Formal descriptions are found in Figures 7 and 8 in the Appendix – our description here is self-contained.

The initial game G_1 is simply $\text{f-Reset}_{\bar{S}}^{\bar{R}}$ with the bit $b = 0$, i.e., \bar{S} and \bar{R} access the *same* functions f_1, \dots, f_r here. Further, G_1 returns true iff \bar{R} returns 0. Thus, $\Pr[G_1] = \Pr \left[\text{f-Reset}_{\bar{S}}^{\bar{R}} \mid b = 0 \right]$. Game G_2 slightly changes G_1 : It keeps track (in a set Q_P) of the triples (i, U, V) describing \bar{O} queries made by the simulated S within \bar{S} ; i.e., either S queried $(i, (+, U))$, and obtained V , or queried $(i, (-, V))$, and obtained U . After \bar{S} terminates with leakage (L, \mathbf{s}) , where $\mathbf{s} = (s_1^{\text{in}}, s_1^{\text{out}}, \dots, s_r^{\text{in}}, s_r^{\text{out}})$, for every $(i, U, V) \in Q_P$ we compute $x_0 \parallel x_1 \leftarrow P(s_i^{\text{in}}, U)$ and $x_3 \parallel x_2 \leftarrow P(s_i^{\text{out}}, V)$, and define table entries

$$T[i, 0 \parallel x_1] \leftarrow x_0 \oplus x_2, \quad T[i, 1 \parallel x_2] \leftarrow x_1 \oplus x_3 .$$

For later reference, we denote by X the set of pairs (i, x) for which we set $T[i, x]$ using Q_P and \mathbf{s} . We then run $\bar{R}(L, \mathbf{s})$, and answer its oracle queries (i, x) using $T[i, x]$. If the entry is undefined, then we return a random value. (As we assumed all of \bar{R} 's queries are distinct, we do not need to remember the output.) As before, G_2 outputs true iff \bar{R} outputs 0.

Note that we always have $T[i, x] = f_i(x)$ for very (i, x) such that $f_i(x)$ was queried by \bar{S} , and re-sampling values un-queried by \bar{S} upon \bar{R} 's queries does not change the distribution of \bar{R} 's output, and hence $\Pr[G_1] = \Pr[G_2]$.

THE INTERSECTION SAMPLER. The game G_3 generates a surrogate for Q_P . This is the output of an algorithm **Sam** which, after \bar{S} terminates with output (L, \mathbf{s}) , takes as input the leakage L (crucially, not \mathbf{s} !) and an iteration parameter $\eta = 4p/\varepsilon \ln(4p/\varepsilon)$ (we let also $\tau = p \cdot \eta$). **Sam** queries the very same \bar{O} implemented by \bar{S} to answer S 's queries (which internally simulates the NR construction using \bar{S} 's own oracle), and returns a set \tilde{Q}_P of 4-tuples (i, U, V, j) such that $j \in [p]$, and (i, U, V) is such that $\bar{O}(i, (+, U))$ would return V (or equivalently $\bar{O}(i, (-, V))$ would return U). Internally, **Sam** will make calls to another (randomized) sub-procedure \mathcal{Q} which takes as input the leakage L , as well as a set Q of tuples (i, U, V, j) consistent with \bar{O} , and returns a set Δ of at most q tuples (i, U, V) , which are not necessarily consistent with \bar{O} . We will specify in detail later below what \mathcal{Q} exactly does, as some further game transitions will come handy to set up proper notation. For now, a generic understanding will suffice. In particular, given such \mathcal{Q} , **Sam** operates as in Figure 4. As we can see, for each $(i, U, V, j) \in \tilde{Q}_P$, j indicates the outer iteration in which this query was added to \tilde{Q}_P . Using this information, for every $j \in [p]$, and every 4-tuple (i, U, V, j) we compute

```

ALGORITHM  $\text{Sam}^{\overline{\mathcal{O}}}(L, \tau)$  :
 $\tilde{Q}_P \leftarrow \emptyset$ 
for  $j = 1$  to  $p$  do
  for  $k = 1$  to  $\eta$  do
     $\Delta_{j,k} \leftarrow \mathcal{Q}(L, \tilde{Q}_P)$ 
    for all  $(i, U, V) \in \Delta_{j,k}$  do
       $V' \leftarrow \overline{\mathcal{O}}(i, (+, U)), U' \leftarrow \overline{\mathcal{O}}(i, (-, V)), \tilde{Q}_P \leftarrow \cup \{(i, U, V', j), (i, U', V, j)\}$ 
return  $\tilde{Q}_P$ 

```

Fig. 4: Description of algorithm Sam.

$x_0 \parallel x_1 \leftarrow P(s_i^{\text{in}}, U)$ and $x_3 \parallel x_2 \leftarrow P(s_i^{\text{out}}, V)$, define

$$\tilde{T}[i, 0 \parallel x_1] \leftarrow x_0 \oplus x_2, \quad \tilde{T}[i, 1 \parallel x_2] \leftarrow x_1 \oplus x_3,$$

and add $(i, 0 \parallel x_1), (i, 1 \parallel x_2)$ to the set \tilde{X}^j . A for now irrelevant caveat is that if one of the entries in \tilde{T} is already set, then we do not overwrite it.⁹

Then, after all of this, G_3 resumes by executing $\overline{R}(L, \mathbf{s})$. For \overline{R} 's j -th query (i, x) we do the following:

1. If $(i, x) \in \tilde{X}^{j'}$ for some $j' \leq j$, then we respond with $\tilde{T}[i, x]$.
2. Otherwise, if $(i, x) \in X$, but the first condition was not met, we respond with $T[i, x]$.
3. Finally, if neither of the above is true, we respond randomly.

As before, G_3 outputs true iff \overline{R} outputs 0. For now, all modifications are syntactical. Indeed, up to the point we start \overline{R} , we satisfy the invariant that $T[i, x] = f_i(x)$ or $\tilde{T}[i, x] = f_i(x)$ whenever these are defined, because $\overline{\mathcal{O}}$ behaves according to the NR construction using \mathbf{s} . On the other hand, if during the execution $\overline{R}(L, \mathbf{s})$ we respond randomly, we know for sure $f_i(x)$ was not queried by \overline{S} , and thus we can re-sample it. Thus, $\Pr[G_3] = \Pr[G_2] = \Pr[G_1]$.

Moving to G_4 , we now answer $\overline{\mathcal{O}}$ queries by S (within \overline{S}) and by **Sam** using random permutations π_1, \dots, π_r , instead of simulating the NR construction using f_1, \dots, f_r , i.e., $\overline{\mathcal{O}}(i, (+, U)) = \pi_i(U)$ and $\overline{\mathcal{O}}(i, (-, V)) = \pi_i^{-1}(V)$. The seeds \mathbf{s} are now independent of $\overline{\mathcal{O}}$. We do not change anything else. We note that the indistinguishability of G_3 and G_4 directly reduces to a suitable distinguisher for Proposition 1, as only **Sam** and S (within \overline{S}) make queries to $\overline{\mathcal{O}}$, but they do not get the keys \mathbf{s} , which are used only after all queries to $\overline{\mathcal{O}}$ have been made to define X and \tilde{X} . Therefore,

$$\Pr[G_1] = \Pr[G_3] \leq \Pr[G_4] + (q + 2q\tau)^2 \left(2\alpha + \frac{1}{2^{2n}} \right), \quad (7)$$

where we have used the fact that **Sam** makes $2q\tau = 2pq\eta$ queries.

The final game is G_5 is identical to G_4 , *except* that in the process of answering \overline{R} 's queries, if case 2 happens, we also set answer randomly. However, should such situation occur, a bad flag is set in G_5 , and since up to the point this flag is set, the behavior of G_4 and G_5 is identical,

$$\Pr[G_4] - \Pr[G_5] \leq \Pr[G_5 \text{ sets bad}] .$$

⁹ This does not matter here, as an entry can only be overwritten with the same value; below, we will change the experiment in a way that overwrites may be inconsistent, and we want to ensure we agree to keep the first value.

<p><u>ADVERSARY $R^{\mathcal{O}}(1^r, L)$:</u></p> <p>$c \leftarrow 0, \tilde{X} \leftarrow \emptyset$</p> <p>$\mathbf{s} = (s_1^{\text{in}}, s_1^{\text{out}}, \dots, s_r^{\text{in}}, s_r^{\text{out}}) \xleftarrow{\\$} \text{P.Kg}$</p> <p>$\tilde{Q}_{\mathbf{P}} \xleftarrow{\\$} \text{Sam}^{\mathcal{O}}(L)$</p> <p>for $j = 1$ to p do</p> <p> for all $(i, U, V, j) \in \tilde{Q}$ do</p> <p> $x_0 \parallel x_1 \leftarrow \mathbf{P}(s_i^{\text{in}}, U), x_3 \parallel x_2 \leftarrow \mathbf{P}(s_i^{\text{out}}, V)$</p> <p> $\tilde{X}^j \leftarrow \tilde{X}^j \cup \{(i, 0 \parallel x_1), (i, 1 \parallel x_2)\}$</p> <p> if $\tilde{T}[i, 0 \parallel x_1] = \perp$ then $\tilde{T}[i, 0 \parallel x_1] \leftarrow x_0 \oplus x_2$</p> <p> if $\tilde{T}[i, 1 \parallel x_2] = \perp$ then $\tilde{T}[i, 1 \parallel x_2] \leftarrow x_1 \oplus x_3$</p> <p>$b' \leftarrow \overline{R}^{\mathcal{O}'}(L, \mathbf{s})$</p> <p>return b'</p>	<p><u>Proc. $\mathcal{O}'(i, x)$:</u></p> <p>$c \leftarrow c + 1, \tilde{X} \leftarrow \tilde{X} \cup \tilde{X}^c$</p> <p>if $T[i, x] = \perp$ then</p> <p> if $(i, x) \in \tilde{X}$ then</p> <p> $T[i, x] \leftarrow \tilde{T}[i, x]$</p> <p> else $T[i, x] \xleftarrow{\\$} \{0, 1\}^n$</p> <p>return $T[i, x]$</p>
--	--

Fig. 5: Adversary R in the proof of Theorem 2.

To analyze the probability on the RHS, we need to specify $\mathcal{Q}(L, \tilde{Q})$ used by **Sam** here. (Note all statements so far were independent of it.) For a given L which appears with positive probability in \mathbf{G}_5 , consider the distribution of the input-output pairs $Q_{\mathbf{P}}$ defined by the interaction of S with $\overline{\mathcal{O}}$, conditioned on the leakage being L , and π_1, \dots, π_r being consistent with the triples defined by \tilde{Q} . Then, $\mathcal{Q}(L, \tilde{Q})$ outputs a sample of $Q_{\mathbf{P}}$ according to this distribution. Using this, we prove the following lemma in Appendix A.4, which uses ideas similar to those from [IR89], with some modifications due to the setting (and the fact that \overline{R} makes p queries).

Lemma 1. $\Pr[\mathbf{G}_5 \text{ sets bad}] \leq \varepsilon/2$

Given this, we are now ready to give our adversary R , which we build from \overline{R} and **Sam** as described in Figure 5. By a purely syntactical argument,

$$\Pr[\mathbf{G}_5] = \Pr[\text{p-Reset}_S^R \mid b = 0] , \quad (8)$$

recalling that the case $b = 0$ is the one where both S and R access the same permutations π_1, \dots, π_r . Therefore, we have established, combining (8), (7), Lemma 1,

$$\Pr[\text{p-Reset}_S^R \mid b = 0] \geq \Pr[\text{f-Reset}_{\overline{S}}^{\overline{R}} \mid b = 0] - \frac{\varepsilon}{2} - (q + 2q\tau)^2 \left(2\alpha + \frac{1}{2^{2n}}\right) . \quad (9)$$

In Appendix A.5 we also prove formally that in the case $b = 1$, R in the game p-Reset_S^R almost perfectly simulates an execution of $\text{f-Reset}_{\overline{S}}^{\overline{R}}$, or more formally,

$$\Pr[\neg \text{p-Reset}_S^R \mid b = 1] \leq \Pr[\neg \text{f-Reset}_{\overline{S}}^{\overline{R}} \mid b = 1] + (q + 2q\tau)^2 \left(2\alpha + \frac{1}{2^{2n}}\right) . \quad (10)$$

We can combine (10) and (9) to obtain, with $\Delta = 2(q + 2q\tau)^2 (2\alpha + \frac{1}{2^{2n}})$,

$$\begin{aligned} \text{Adv}_{2n, S}^{\text{p-reset}}(R) &= \Pr[\text{p-Reset}_S^R \mid b = 0] - \Pr[\neg \text{p-Reset}_S^R \mid b = 1] \\ &\geq \Pr[\text{f-Reset}_{\overline{S}}^{\overline{R}} \mid b = 0] - \Pr[\neg \text{f-Reset}_{\overline{S}}^{\overline{R}} \mid b = 1] - \frac{\varepsilon}{2} - \Delta \\ &\geq \varepsilon/2 - \Delta . \end{aligned}$$

This concludes the proof. □

4 Correlation Intractability of Public-seed Permutations

In this section, we study the correlation intractability (CI) of the NR construction against k -ary evasive relations. Firstly, in Section 4.1, we define a stronger notion of evasiveness – *strong evasiveness* – and show that evasiveness and strong evasiveness are asymptotically equivalent when $k = O(1)$. In Section 4.2 we study the relations that are hard for two-round Feistel. In Section 4.3 we show that for $k = O(1)$ the NR construction where P^{-1} is a family of almost $O(k^2)$ -wise independent permutations is correlation intractable against k -ary evasive relations. In the special case of unary evasive relations ($k = 1$), we show that (see Appendix B.6) P instead can be instantiated from one-round Feistel with a 10-wise independent round function.

4.1 Strong evasiveness

Evasiveness is defined through the hardness of winning a security game. For our results, we need instead a combinatorial understanding of evasive relations. To this end, we will rely on the following notion of evasiveness, which, as we show below, is generally implied by evasiveness if $k = O(1)$.

Definition 1 (Strongly evasive relations). *Let R be a k -ary relation over $X \times X$ and $\delta \in [0, 1]$. We say that R is δ -strongly evasive if the following are true for all $0 \leq j < k' \leq k$:*

- For all distinct $\mathbf{u}_1, \dots, \mathbf{u}_{k'} \in X$, all $\mathbf{v}_1, \dots, \mathbf{v}_j \in X$, we have

$$|\{(\mathbf{v}_{j+1}, \dots, \mathbf{v}_{k'}) : \{(\mathbf{u}_1, \mathbf{v}_1), \dots, (\mathbf{u}_{k'}, \mathbf{v}_{k'})\} \in R\}| \leq \delta \cdot \prod_{i=j}^{k'-1} (|X| - i).$$

- For all distinct $\mathbf{v}_1, \dots, \mathbf{v}_{k'} \in X$, all $\mathbf{u}_1, \dots, \mathbf{u}_j \in X$, we have

$$|\{(\mathbf{u}_{j+1}, \dots, \mathbf{u}_{k'}) : \{(\mathbf{u}_1, \mathbf{v}_1), \dots, (\mathbf{u}_{k'}, \mathbf{v}_{k'})\} \in R\}| \leq \delta \cdot \prod_{i=j}^{k'-1} (|X| - i).$$

It is not hard to see that if a relation is δ -strongly evasive, then it is also evasive, in the sense that it is $(q, q^k \delta)$ -evasive. In particular, $q^k \delta$ is negligible whenever δ is negligible, q polynomial, and $k = O(1)$.

We remark that there are relations R which are evasive, yet not strongly evasive. Consider for example the relation which contains $\{(0^{2n}, 0^{2n}), (\mathbf{u}, \mathbf{v})\}$ for all $\mathbf{u}, \mathbf{v} \neq 0^{2n}$. This relation is obviously evasive to start with – satisfying it requires $\pi(0^{2n}) = 0^{2n}$, which will happen with probability 2^{-2n} only, yet for $\mathbf{u}_1 = \mathbf{v}_1 = 0^{2n}$, and $\mathbf{u}_2 \neq 0^{2n}$, all strings \mathbf{v}_2 make $\{(\mathbf{u}_1, \mathbf{v}_1), (\mathbf{u}_2, \mathbf{v}_2)\}$ valid. Still, somehow, the intuition is that the core of R is the relation $R^* = \{(0^{2n}, 0^{2n})\}$, which is strongly evasive, with $\delta = 2^{-2n}$. Indeed, an attacker that satisfies the original relation R , can directly satisfy R^* , thus the fact that R^* is evasive (and in particular, strongly evasive) implies that R is evasive.

The following lemma generalizes this, and implies e.g. that for $\delta = \text{negl}(\lambda)$ and $k = O(1)$, evasiveness and strong evasiveness are (qualitatively) equivalent. The proof is found in Appendix B.1.

Lemma 2 (Normalization of evasive relations). *Let $\delta > 0$, and let R be a k -ary (k^2, δ^k) -evasive relation on $X \times X$ for random permutations. Then, there exists a relation R^* which is δ -strongly evasive for random permutations, and moreover, for every $S \in R$, there exists $\emptyset \neq S^* \subseteq S$ such that $S^* \in R^*$.*

Lemma 2 now is all we need. Say E is correlation intractable for all k -ary strongly evasive relations (for some negligible δ), where $k = O(1)$. Then, E must be correlation intractable for any (k^2, δ) -evasive relation R , too. Were it not, we could take an adversary A breaking the CI for R with non-negligible advantage, and use it to break CI of R^* . To this end, we simply run A , and when it outputs $S \in R$, we output the corresponding $S^* \in R^*$ guaranteed by Lemma 2. (As $k = O(1)$, a random subset of S will do with constant loss in the advantage.) But since R^* is $\sqrt[k]{\delta}$ -strongly evasive, this contradicts our assumption on E .

Clearly, the equivalence is merely asymptotic. If one is interested in concrete security, the best approach to use our results below is to directly assess the δ for which a specific relation R is δ -strongly evasive.

4.2 Partial Correlation Intractability of Two-round Feistel

The two-round Feistel construction Fei_2^f , is a permutation on $2n$ -bit strings that makes calls to an oracle $f : \{0, 1\}^{n+1} \rightarrow \{0, 1\}^n$. In particular, on input $\mathbf{x} = x_0 \| x_1$, where $x_0, x_1 \in \{0, 1\}^n$, running $\text{Fei}_2^f(\mathbf{x})$ outputs $\mathbf{y} = x_2 \| x_3$, where

$$x_2 \leftarrow x_0 \oplus f(0 \| x_1) ; x_3 \leftarrow x_1 \oplus f(1 \| x_2).$$

Symmetrically, upon an inverse query, $\text{Fei}_2^{f^{-1}}(\mathbf{y} = x_2 \| x_3)$ simply computes the values backwards, and outputs $\mathbf{x} = x_0 \| x_1$.

In this section, we discuss relations R on $2n$ -bit strings that are hard for two-round Feistel when instantiated with a random function. In particular, we will give a combinatorial characterization which is sufficient to achieve this.

FEISTEL EVASIVENESS. We first note that in a relation R , certain sets $S \in R$ can never be satisfied by the two-round Feistel construction out of structural constraints. In particular, if we have two input-output pairs $(\mathbf{x}_1[0] \| \mathbf{x}_1[1], \mathbf{x}_1[2] \| \mathbf{x}_1[3])$ and $(\mathbf{x}_2[0] \| \mathbf{x}_2[1], \mathbf{x}_2[2] \| \mathbf{x}_2[3])$ with $\mathbf{x}_1[2] = \mathbf{x}_2[2]$ in the same set $S \in R$, then we *must* have $\mathbf{x}_1[3] \oplus \mathbf{x}_2[3] = \mathbf{x}_1[1] \oplus \mathbf{x}_2[1]$. Symmetrically, if $\mathbf{x}_1[1] = \mathbf{x}_2[1]$, then we must have $\mathbf{x}_1[0] \oplus \mathbf{x}_2[0] = \mathbf{x}_1[2] \oplus \mathbf{x}_2[2]$. It will thus be convenient to define the following.

Definition 2. For every k -ary relation R on $2n$ -bit strings, we define the relation $\bar{R} \subseteq R$ that only contains $S \in R$ if for every $(\mathbf{x}_1[0] \| \mathbf{x}_1[1], \mathbf{x}_1[2] \| \mathbf{x}_1[3]), (\mathbf{x}_2[0] \| \mathbf{x}_2[1], \mathbf{x}_2[2] \| \mathbf{x}_2[3]) \in S$, the following is true:

- If $\mathbf{x}_1[2] = \mathbf{x}_2[2]$, then $\mathbf{x}_1[3] \oplus \mathbf{x}_2[3] = \mathbf{x}_1[1] \oplus \mathbf{x}_2[1]$.
- If $\mathbf{x}_1[1] = \mathbf{x}_2[1]$, then $\mathbf{x}_1[0] \oplus \mathbf{x}_2[0] = \mathbf{x}_1[2] \oplus \mathbf{x}_2[2]$.

Clearly, the significance of this is that when assessing whether R is correlation intractable for two-round Feistel, it suffices to prove that \bar{R} is correlation intractable, as $S \in R \setminus \bar{R}$ can never be satisfied. We are now ready to state the following combinatorial requirement on relations, which we will prove to be evasive for two-round Feistel below.

Definition 3 (δ -2-Feistel evasive relations). Let R be a k -ary relation over $\{0, 1\}^{2n}$, and $\delta \in [0, 1]$. We say that R is δ -2-Feistel evasive if the following are true for all $0 \leq j < k' \leq k$:

- For all distinct $\mathbf{x}_1, \dots, \mathbf{x}_{k'} \in \{0, 1\}^{2n}$, distinct $\mathbf{y}_1, \dots, \mathbf{y}_j \in \{0, 1\}^{2n}$, and $y^* \in \{0, 1\}^n$ s.t. $\mathbf{x}_{j+1}[1], \dots, \mathbf{x}_{k'}[1]$ are distinct and $y^* \notin \{\mathbf{y}_1[0], \dots, \mathbf{y}_j[0]\}$,

$$\left| \{(y_{j+1}, \dots, y_{k'})\} : \{(\mathbf{x}_1, \mathbf{y}_1), \dots, (\mathbf{x}_j, \mathbf{y}_j), (\mathbf{x}_{j+1}, y^* \parallel y_{j+1}), \dots, (\mathbf{x}_{k'}, y^* \parallel y_{k'})\} \in \overline{R'} \right| \leq \delta' \cdot 2^n. \quad (11)$$

- For all distinct $\mathbf{y}_1, \dots, \mathbf{y}_{k'} \in \{0, 1\}^{2n}$, distinct $\mathbf{x}_1, \dots, \mathbf{x}_j \in \{0, 1\}^{2n}$, and $x^* \in \{0, 1\}^n$ s.t. $\mathbf{y}_{j+1}[0], \dots, \mathbf{y}_{k'}[0]$ are distinct and $x^* \notin \{\mathbf{x}_1[1], \dots, \mathbf{x}_j[1]\}$,

$$\left| \{(x_{j+1}, \dots, x_{k'})\} : \{(\mathbf{x}_1, \mathbf{y}_1), \dots, (\mathbf{x}_j, \mathbf{y}_j), (x_{j+1} \parallel x^*, \mathbf{y}_{j+1}), \dots, (x_{k'} \parallel x^*, \mathbf{y}_{k'})\} \in \overline{R'} \right| \leq \delta' \cdot 2^n. \quad (12)$$

Also, we let $\text{FEv}(k, \delta)$ denote the set of all k -ary δ -2-Feistel evasive relations.

FEISTEL CORRELATION INTRACTABILITY. We now prove that for all relations satisfying Definition 3, two-round Feistel is indeed correlation intractable in the model where both round functions are independent random functions, to which the adversary is given oracle access.

Proposition 2 (CI of Two-round Feistel). For $\delta \in [0, 1]$ and $k \geq 1$ be an integer, let $R \in \text{FEv}(k, \delta)$. For any (unbounded) adversary A making at most q queries to f , $\text{Adv}_{R, \text{Fei}_2^f}^{\text{cri}}(A) \leq 2k\delta \cdot q^{2k+1}$.

The proof of Proposition 2 can be found in Appendix B.2.

Remark 1. For the special case of $k = 1$, that is, unary relations, one can adapt the above proof and show that $\text{Adv}_{R, \text{Fei}_2^f}^{\text{cri}}(A) \leq \delta \cdot q^2$ where A makes q queries to f and $R \in \text{FEv}(1, \delta)$.

4.3 Correlation Intractability of the NR construction

In this section we view the NR construction as a family $\text{NR}^f[\mathbf{P}]$ that makes oracle calls to $f : \{0, 1\}^{n+1} \rightarrow \{0, 1\}^n$ and \mathbf{P} is a family of permutations on $2n$ -bits. The key generation algorithm NR.Kg now just outputs a tuple $(s^{\text{in}}, s^{\text{out}})$ where $s^{\text{in}}, s^{\text{out}} \stackrel{\$}{\leftarrow} \mathbf{P.Kg}$ and the evaluation algorithm NR.Eval proceeds as before but instead makes calls to f for evaluating the round function.

We show that $\text{NR}^f[\mathbf{P}]$, where \mathbf{P}^{-1} is a family of almost $O(k^2)$ -wise independent permutations, is correlation intractable against strongly evasive k -ary relations when the adversary is given the seed $(s^{\text{in}}, s^{\text{out}})$ of the NR construction and only oracle access to f . The proof of CI proceeds by showing that \mathbf{P} transforms a strongly evasive relation R into a 2-Feistel evasive relation $R_{\pi, \sigma}$ and hence for the adversary to break the CI of $\text{NR}^f[\mathbf{P}]$ it needs to break the CI of two-round Feistel against $R_{\pi, \sigma}$ which was studied in Section 4.2.

p -WISE INDEPENDENT PERMUTATIONS. For any $\varepsilon \in [0, 1]$ and $p \geq 1$, we say that a family of permutations \mathbf{P} on m -bit strings is (ε, p) -wise independent if for all distinct $u_1, \dots, u_p \in \{0, 1\}^m$, the distributions of $\mathbf{P}(s, u_1), \dots, \mathbf{P}(s, u_p)$ (for $s \stackrel{\$}{\leftarrow} \mathbf{P.Kg}$) and of $\rho(u_1), \dots, \rho(u_p)$ (for $\rho \stackrel{\$}{\leftarrow} \text{Perms}(m)$) are at most ε -apart in statistical distance.

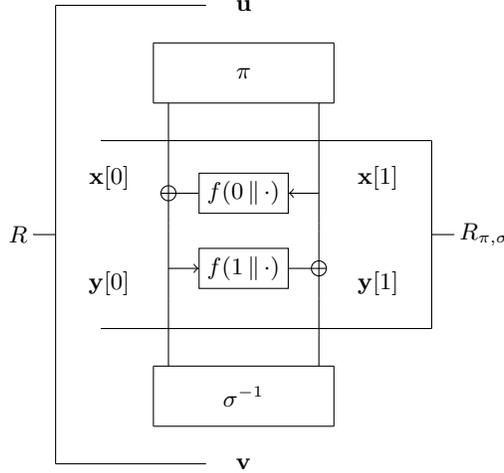


Fig. 6: The NR construction instantiated with a permutation family \mathbf{P} on $2n$ -bits such that \mathbf{P}^{-1} is $(\varepsilon, k \cdot t)$ -wise independent where $\pi = \mathbf{P}(s^{\text{in}}, \cdot)$ and $\sigma = \theta(\mathbf{P}(s^{\text{out}}, \cdot))$ for $s^{\text{in}}, s^{\text{out}} \stackrel{\$}{\leftarrow} \mathbf{P}.\text{Kg}$. Here, θ is a permutation on $2n$ -bits such that for all $\mathbf{x} \in \{0, 1\}^{2n}$ we have $\theta(\mathbf{x} = x_0 \| x_1) = x_1 \| x_0$. For some k -ary strongly-evasive relation R , we construct a 2-Feistel evasive relation $R_{\pi, \sigma}$ by transforming every $(\mathbf{u}, \mathbf{v}) \in S$ where $S \in R$, by applying π to \mathbf{u} and σ to \mathbf{v} .

FROM STRONGLY EVASIVE RELATION TO 2-FEISTEL-EVASIVE RELATION. Let R be a k -ary relation over $\{0, 1\}^{2n} \times \{0, 1\}^{2n}$. For $s^{\text{in}}, s^{\text{out}} \stackrel{\$}{\leftarrow} \mathbf{P}.\text{Kg}$, let $\pi = \mathbf{P}(s^{\text{in}}, \cdot)$ and $\sigma = \theta(\mathbf{P}(s^{\text{out}}, \cdot))$, where θ is a permutation on $2n$ -bits such that for $\mathbf{x} \in \{0, 1\}^{2n}$ we have $\theta(\mathbf{x} = x_0 \| x_1) = x_1 \| x_0$ ¹⁰. We define a relation $R_{\pi, \sigma}$ which is a result of transforming $\{(\mathbf{u}_1, \mathbf{v}_1), \dots, (\mathbf{u}_{k'}, \mathbf{v}_{k'})\} \in R$ via π and σ in the following way,

$$R_{\pi, \sigma} = \{ \{(\pi(\mathbf{u}_1), \sigma(\mathbf{v}_1)), \dots, (\pi(\mathbf{u}_{k'}), \sigma(\mathbf{v}_{k'}))\} \mid \{(\mathbf{u}_1, \mathbf{v}_1), \dots, (\mathbf{u}_{k'}, \mathbf{v}_{k'})\} \in R \} .$$

Then, for every δ -strongly evasive k -ary relation R we show that $R_{\pi, \sigma} \in \text{FEv}(k, \delta')$ for some δ' larger than δ , except with small probability, where the probability is taken over the random choice of (π, σ) . This is more formally captured in the following:

Proposition 3 (CI Amplification). *For $\delta \in [0, 1)$ and an integer $k \geq 1$, let R be a k -ary δ -strongly evasive relation over $\{0, 1\}^{2n}$. For an even integer $t \geq 2$, let \mathbf{P} be a family of permutations such that \mathbf{P}^{-1} is $(\varepsilon, k \cdot t)$ -wise independent. Then, for $\delta' \in [0, 1]$ such that $\delta' > \delta$,*

$$\Pr_{\pi, \sigma}[R_{\pi, \sigma} \notin \text{FEv}(k, \delta')] \leq 12k^2 \left(\frac{1}{\delta' - \delta} \right)^t 2^{(4k-1)n} \left[C_t \cdot \left(\frac{4\delta^* t}{2^n} \right)^{t/2} + \varepsilon \cdot (1 + \delta)^t \right] ,$$

where $C_t = 2e^{1/6t} \sqrt{\pi t} \left(\frac{5}{2e} \right)^{t/2}$, $\delta^* = \max\left(\delta, \frac{t \cdot 2^k}{2^n}\right)$, $\pi = \mathbf{P}(s^{\text{in}}, \cdot)$ and $\sigma = \theta(\mathbf{P}(s^{\text{out}}, \cdot))$ for $s^{\text{in}}, s^{\text{out}} \stackrel{\$}{\leftarrow} \mathbf{P}.\text{Kg}$.

¹⁰ It is easy to see that $\theta = \theta^{-1}$ hence $\sigma^{-1} = \mathbf{P}^{-1}(s^{\text{out}}, \theta(\cdot))$. We note that θ is introduced to ensure consistency with the definition of the NR construction as $\mathbf{P}^{-1}(s^{\text{out}})$ operates on $\mathbf{y}[1] \| \mathbf{y}[0]$ where \mathbf{y} is the output of the underlying two-round feistel.

Now, Proposition 3 can be combined with Proposition 2 to establish the correlation intractability of NR against strongly evasive relations (Theorem 3).

Theorem 3 (CI of NR). *For $\delta \in [0, 1)$ and an integer $k \geq 1$, let R be a k -ary δ -strongly evasive relation over $\{0, 1\}^{2n}$. Further, let \mathbf{P} be a family of permutations on $2n$ -bits such that \mathbf{P}^{-1} is $(\varepsilon, 10k^2)$ -wise independent where $\varepsilon \leq 1/2^{5kn}$. Then for any (potentially unbounded) adversary A making q queries,*

$$\text{Adv}_{R, \text{NR}^f[\mathbf{P}]}^{\text{cri}}(A) \leq \frac{24k^2 \cdot (40k\delta^*)^{5k} + 12k^2 \cdot (1 + \delta)^{10k}}{2^{4kn/9}} + 2k\delta' \cdot q^{2k+1}, \quad (13)$$

where $\delta' = \delta + 2^{-n/18}$ and $\delta^* = \max\left(\delta, \frac{10k \cdot 2^k}{2^n}\right)$.

The proof of Theorem 3 can be found in Appendix B.3. The asymptotic interpretation of Eq. (13) is that when $n = \omega(\log \lambda)$, $k = O(1)$, $\delta = \text{negl}(\lambda)$ and $q = \text{poly}(\lambda)$, $\text{NR}^f[\mathbf{P}]$ is correlation intractable for k -ary strongly evasive relations. Combining this with Lemma 2, the CI then extends to any k -ary evasive relation. We also remark that Theorem 3 extends to the setting of multi-key correlation intractability introduced in [CS16], but to avoid notational overhead we limit ourselves to the single-key setting for this version.

ON INSTANTIATING \mathbf{P}^{-1} FROM THEOREM 3. We detail the construction of (ε, p) -wise permutations in Appendix B.5 and show that an $O(k)$ -round Feistel construction with $10k^2$ -wise independent round functions can instantiate the permutation family \mathbf{P}^{-1} . We refer the reader to Appendix B.5 for more details.

4.4 Proof of Proposition 3

We will show that for every $0 \leq j < k' \leq k$ the following hold,

1. For all distinct $\mathbf{x}_1, \dots, \mathbf{x}_{k'}$, distinct $\mathbf{y}_1, \dots, \mathbf{y}_j$ and $y^* \in \{0, 1\}^n$ such that $\mathbf{x}_{j+1}[1], \dots, \mathbf{x}_{k'}[1]$ are distinct and $y^* \notin \{\mathbf{y}_1[0], \dots, \mathbf{y}_j[0]\}$,

$$\Pr[\text{Eq. (11) does not hold for } R_{\pi, \sigma}] \leq e_1(k', j) + \varepsilon \cdot e_2(k', j). \quad (14)$$

2. For all distinct $\mathbf{y}_1, \dots, \mathbf{y}_{k'}$, all $\mathbf{x}_1, \dots, \mathbf{x}_j$ and all $x^* \in \{0, 1\}^n$ such that $\mathbf{y}_{j+1}[0], \dots, \mathbf{y}_{k'}[0]$ are distinct and $x^* \notin \{\mathbf{x}_1[1], \dots, \mathbf{x}_j[1]\}$,

$$\Pr[\text{Eq. (12) does not hold for } R_{\pi, \sigma}] \leq e_1(k', j) + \varepsilon \cdot e_2(k', j), \quad (15)$$

where the probability is taken over the random choice of (π, σ) and

$$e_1(k', j) = 3 \cdot C_t \left(\frac{1}{\delta' - \delta} \right)^t \left(\frac{2\delta^* t}{2^n} \right)^{t/2} 2^{(k'-j)(t/2+1)},$$

$$e_2(k', j) = 2 \left(\frac{1 + \delta}{\delta' - \delta} \right)^t 2^{k'-j}.$$

Given that the above hold, we then take appropriate union bounds (for Eq. (14)) over all $\mathbf{y}_1, \dots, \mathbf{y}_{k'}$, $\mathbf{x}_1, \dots, \mathbf{x}_j$ and x^* and then over all j, k' . Symmetrically, we take union bounds (for Eq. (15)). Then

the following holds and this concludes the proof of Theorem 3.

$$\begin{aligned} \Pr[R_{\pi,\sigma} \notin \text{FEv}(k, \delta')] &\leq 2 \sum_{k'=1}^k \sum_{j=0}^{k'-1} 2^{n(2k'+2j+1)} \cdot (e_1(j, k') + \varepsilon \cdot e_2(j, k')) \\ &\leq 12k^2 \left(\frac{1}{\delta' - \delta} \right)^t 2^{(4k-1)n} \left[C_t \cdot \left(\frac{4\delta^* t}{2^n} \right)^{t/2} + \varepsilon \cdot (1 + \delta)^t \right]. \end{aligned}$$

From now on we focus on showing Eq. (15) and the analysis for Eq. (14) is symmetrical.

ESTABLISHING EQUATION (15). Let us fix some arbitrary k' and j such that $0 \leq j < k' \leq k$. Let us also fix some distinct $\mathbf{y}_1, \dots, \mathbf{y}_{k'} \in \{0, 1\}^{2n}$, distinct $\mathbf{x}_1, \dots, \mathbf{x}_j$ and $x^* \in \{0, 1\}^n$ such that $\mathbf{y}_{j+1}[0], \dots, \mathbf{y}_{k'}[0]$ are distinct and $x^* \notin \{\mathbf{x}_1[0], \dots, \mathbf{x}_j[0]\}$. Then, we are interested in counting the number of tuples $((\mathbf{u}_1, \mathbf{v}_1), \dots, (\mathbf{u}_{k'}, \mathbf{v}_{k'}))$ in R that on applying π and σ transform to $((\mathbf{x}_1, \mathbf{y}_1), \dots, (\mathbf{x}_j, \mathbf{y}_j), (\cdot \| x^*, \mathbf{y}_{j+1}), \dots, (\cdot \| x^*, \mathbf{y}_{k'}))$. Let us fix σ and this defines $\mathbf{v}_i = \sigma^{-1}(\mathbf{y}_i)$ for every $i \in [k']$ allowing us to focus only on the following set \mathcal{U} of tuples.

$$\mathcal{U} = \{(\mathbf{u}_1, \dots, \mathbf{u}_{k'}) \mid \{(\mathbf{u}_1, \mathbf{v}_1), \dots, (\mathbf{u}_{k'}, \mathbf{v}_{k'})\} \in R\}.$$

Then, we are interested in counting the number of tuples $\mathbf{U} = (\mathbf{u}_1, \dots, \mathbf{u}_{k'})$ in \mathcal{U} that satisfy,

1. $\pi(\mathbf{u}_1) = \mathbf{x}_1, \pi(\mathbf{u}_2) = \mathbf{x}_2, \dots, \pi(\mathbf{u}_j) = \mathbf{x}_j$.
2. $\pi_1(\mathbf{u}_{j+1}) = \pi_1(\mathbf{u}_{j+2}) \dots = \pi_1(\mathbf{u}_{k'}) = x^*$.
3. For every $i \in \{j+1, \dots, k'\}$, $\pi_0(\mathbf{u}_i) \oplus \pi_0(\mathbf{u}_{j+1}) = \Delta_i$, where $\Delta_i = \mathbf{y}_{j+1}[0] \oplus \mathbf{y}_i[0]$ ¹¹,

where $\pi_0(\mathbf{u})$ and $\pi_1(\mathbf{u})$ denote the first n -bits and last n -bits of $\pi(\mathbf{u})$. Or equivalently, count the number of \mathbf{U} 's such that $\pi(\mathbf{U})$ ¹² falls in \mathcal{X} where,

$$\mathcal{X} = \{(\mathbf{x}_1, \dots, \mathbf{x}_j, x \oplus \Delta_{j+1} \| x^*, \dots, x \oplus \Delta_{k'} \| x^*) \mid x \in \{0, 1\}^n\}.$$

Note that every element \mathbf{X} of \mathcal{X} is completely described by an n -bit string x . Now for $\mathbf{U} = (\mathbf{u}_1, \dots, \mathbf{u}_{k'}) \in \mathcal{U}$, let $I_{\mathbf{U}}$ be an indicator random variable,

$$I_{\mathbf{U}} = \begin{cases} 1 & \text{if } (\pi(\mathbf{u}_1), \dots, \pi(\mathbf{u}_{k'})) \in \mathcal{X}, \\ 0 & \text{otherwise.} \end{cases}$$

Then it suffices to prove that,

$$\Pr_{\pi} \left[\sum_{\mathbf{U} \in \mathcal{U}} I_{\mathbf{U}} > \delta' \cdot 2^n \right] \leq e_1(k', j) + \varepsilon \cdot e_2(k', j).$$

Instead of looking at the sum $\sum I_{\mathbf{U}}$, we look at an equivalent sum $\sum I_x$ of, albeit, different indicator random variables I_x 's, which will be convenient to analyse. For $x \in \{0, 1\}^n$ we define an indicator random variable I_x which is 1 if π^{-1} transforms $\mathbf{X} \in \mathcal{X}$ (that corresponds to x) into some $\mathbf{U} \in \mathcal{U}$. More formally,

$$I_x = \begin{cases} 1 & \text{if } (\pi^{-1}(\mathbf{x}_1), \dots, \pi^{-1}(x \oplus \Delta_{j+1} \| x^*), \dots, \pi^{-1}(x \oplus \Delta_{k'} \| x^*)) \in \mathcal{U}, \\ 0 & \text{otherwise.} \end{cases} \quad (16)$$

¹¹ As the definition of δ' -2-Feistel evasiveness concerns itself with $\overline{R_{\pi,\sigma}}$.

¹² By $\pi(\mathbf{U})$ we mean the tuple $(\pi(\mathbf{u}_1), \dots, \pi(\mathbf{u}_{k'}))$.

Then, it is easy to see that counting the number of $\mathbf{U} \in \mathcal{U}$ such that $I_{\mathbf{U}} = 1$ (or $\pi(\mathbf{U}) \in \mathcal{X}$) is the same as counting the number of $x \in \{0, 1\}^n$ such that $I_x = 1$ (or $\pi^{-1}(\mathbf{X}) \in \mathcal{U}$). Therefore, $\sum_{\mathbf{U} \in \mathcal{U}} I_{\mathbf{U}} = \sum_{x \in \{0, 1\}^n} I_x$ and we aim to show that,

$$\Pr_{\pi} \left[\sum_{x \in \{0, 1\}^n} I_x > \delta' \cdot 2^n \right] \leq e_1(k', j) + \varepsilon \cdot e_2(k', j). \quad (17)$$

PARTITIONING $\{0, 1\}^n$. We would like to use concentration bounds for the sum of random variables I_x 's. But note that they are not independent as they may depend on the output of π^{-1} on the same input. Therefore, as a first step towards constructing independent random variables, we partition $\{0, 1\}^n$ into subsets which will allow us to break the sum $\sum I_x$ into sums over these subsets.

Let us consider the following relation on $\{0, 1\}^n \times \{0, 1\}^n$. For any $x, x' \in \{0, 1\}^n$, we say that x is related to x' (denoted as $x \sim x'$) if there exists an index set $\mathcal{B} \subseteq \{j+1, \dots, k'\}$ where such that,

$$x = x' \oplus \bigoplus_{i \in \mathcal{B}} \Delta_i.$$

It is easy to see that the relation \sim is an equivalence relation. Then, for any $x \in \{0, 1\}^n$, let EQ_x denote its equivalence class, that is, $\text{EQ}_x = \{x' \in \{0, 1\}^n \mid x \sim x'\}$. Let $|\text{EQ}_x| = l$ and it is easy to see that $l \leq 2^{k'-j}$. Let $\{\text{EQ}_i\}_{i=1}^M$ be the M equivalence classes of \sim where $|\text{EQ}_i| = l$ and $M \cdot l = 2^n$. Furthermore, let $\text{EQ}_i = \{x_1^i, x_2^i, \dots, x_l^i\}$ be an enumeration of EQ_i where x_q^i is the q th member of the i th equivalence class EQ_i . Then, we can break the sum of I_x 's into,

$$\sum_{x \in \{0, 1\}^n} I_x = \sum_{i=1}^M \sum_{q=1}^l I_{x_q^i} = \sum_{q=1}^l \sum_{i=1}^M I_{x_q^i}.$$

For $q \in [l]$, let $X_q = \sum_{i=1}^M I_{x_q^i}$. In other words, X_q is the sum of q th member of each equivalence class EQ_i . We are going to show that for every $q \in [l]$,

$$\Pr[X_q > \delta' \cdot M] \leq 3C_t \cdot \frac{1}{(\delta' - \delta)^t} \left(\frac{2t\delta^*}{2^n} \right)^{t/2} l^{t/2} + 2\varepsilon \cdot \left(\frac{1 + \delta}{\delta' - \delta} \right)^t \quad (18)$$

Taking union bound over all $q \in [l]$ and using $l \leq 2^{k'-j}$, we have that Eq. (17) holds.

BOUNDING THE SUBSUM X_q . From now on, we will focus on analysing one of the subsums X_q and the other subsums can be analogously handled. Fix some q and let $X = X_q$. Let the corresponding set of x 's be $\{x^1, \dots, x^M\}$ where each x^i comes from a different equivalence class EQ_i . For every $i_1 \neq i_2 \in [M]$,

- Firstly, $\Delta_{j+1}, \dots, \Delta_{k'}$ are distinct as $\mathbf{y}_{j+1}[0], \dots, \mathbf{y}_{k'}[0]$ are distinct. Therefore, $x^{i_1} \oplus \Delta_{j+1}, \dots, x^{i_1} \oplus \Delta_{k'}$ are distinct.
- Secondly for any index set $\mathcal{B} \subseteq \{j+1, \dots, k'\}$,

$$x^{i_1} \neq x^{i_2} \oplus \bigoplus_{i \in \mathcal{B}} \Delta_i. \quad (19)$$

This implies that for any $I_{x^{i_1}}$ and $I_{x^{i_2}}$, $\{x_{j+1}^{i_1} \oplus \Delta_{j+1}, \dots, x_{k'}^{i_1} \oplus \Delta_{k'}\}$ and $\{x_{j+1}^{i_2} \oplus \Delta_{j+1}, \dots, x_{k'}^{i_2} \oplus \Delta_{k'}\}$ are disjoint. Therefore, except the first j (values that correspond to $\pi^{-1}(\mathbf{x}_i)$ for $i \in [j]$), the remaining set of values in the output of π^{-1} that each $I_{x^{i_1}}$ and $I_{x^{i_2}}$ depend on are disjoint.

We will crucially exploit these two properties of I_{x^i} 's to show that the following:

Lemma 3. *For X (as defined above), there exists a random variable Z with expectation $\mu = \mathbb{E}[Z] \leq \delta \cdot M$ where Z is a sum of M independent indicator random variables, such that for any integer $a > 0$,*

$$\Pr[|X - \mu| > a] \leq \frac{3 \cdot \mathbb{E}[(Z - \mu)^t]}{a^t} + 2\varepsilon \cdot \frac{(M + \mu)^t}{a^t}.$$

For each indicator random variable I_{x^i} , we first define another indicator random variable $I_{x^i}^\rho$ where the only difference is that we replace the $k \cdot t$ -wise independent permutation π^{-1} with a random permutation ρ . Note that the resulting $I_{x^i}^\rho$ are still not independent as they depend on the output of ρ . So, we then define a sequence of random variable $I_{x^i}^*$ that have the same marginal distribution as that of $I_{x^i}^\rho$ but are independent. Then, we show a domination argument that relates the t -th moment of $(Y - \mu)$ with the t -th moment of $(Z - \mu)$ where Y and Z are the sum of $I_{x^i}^\rho$ and $I_{x^i}^*$ respectively. The proof of Lemma 3 can be found in Appendix B.4. Next, we apply the following concentration bound due to [BR94] to the random variable Z .

Lemma 4 (A.4. from [BR94]). *Let $t \geq 2$ be an even integer. Suppose Z_1, \dots, Z_n are independent random variables taking values in $[0, 1]$. Let $Z = Z_1 + \dots + Z_n$ and $\mu = \mathbb{E}[Z]$. Then,*

$$\mathbb{E}[(Z - \mu)^t] \leq C_t \cdot (t\mu + t^2)^{t/2}.$$

Then as $\mu \leq \delta \cdot M$ we have,

$$\Pr[X > \delta \cdot M + a] \leq \Pr[X > \mu + a] \leq 3C_t \cdot \left(\frac{t\mu + t^2}{a^2}\right)^{t/2} + 2\varepsilon \cdot \left(\frac{M + \mu}{a}\right)^t,$$

Now let $a = (\delta' - \delta) \cdot M$ and using $M \cdot l = 2^n$ and $\delta^* = \max(\delta, t \cdot 2^k/2^n)$, we have

$$\Pr[X > \delta' \cdot M] \leq 3C_t \cdot \frac{1}{(\delta' - \delta)^t} \left(\frac{2t\delta^*}{2^n}\right)^{t/2} l^{t/2} + 2\varepsilon \cdot \left(\frac{1 + \delta}{\delta' - \delta}\right)^t$$

which establishes that Eq. (18) holds (which establishes that Eq. (15) holds) and thereby concludes the proof of Proposition 3. \square

Acknowledgments The authors were supported by NSF grants CNS-1553758 (CAREER), CNS-1423566, CNS-1719146, CNS-1528178, and IIS-1528041, and by a Sloan Research Fellowship.

References

- ABD⁺13. Elena Andreeva, Andrey Bogdanov, Yevgeniy Dodis, Bart Mennink, and John P. Steinberger. On the indistinguishability of key-alternating ciphers. In Ran Canetti and Juan A. Garay, editors, *CRYPTO 2013, Part I*, volume 8042 of *LNCS*, pages 531–550. Springer, August 2013.
- ABM14. Elena Andreeva, Andrey Bogdanov, and Bart Mennink. Towards understanding the known-key security of block ciphers. In Shiho Moriai, editor, *FSE 2013*, volume 8424 of *LNCS*, pages 348–366. Springer, March 2014.
- AJN15. Jean-Philippe Aumasson, Philipp Jovanovic, and Samuel Neves. NORX8 and NORX16: Authenticated encryption for low-end systems. Cryptology ePrint Archive, Report 2015/1154, 2015. <http://eprint.iacr.org/2015/1154>.

- BDPV08. Guido Bertoni, Joan Daemen, Michael Peeters, and Gilles Van Assche. On the indifferentiability of the sponge construction. In Nigel P. Smart, editor, *EUROCRYPT 2008*, volume 4965 of *LNCS*, pages 181–197. Springer, April 2008.
- BDPV10. Guido Bertoni, Joan Daemen, Michael Peeters, and Gilles Van Assche. Sponge-based pseudo-random number generators. In Stefan Mangard and François-Xavier Standaert, editors, *CHES 2010*, volume 6225 of *LNCS*, pages 33–47. Springer, August 2010.
- BFM14. Christina Brzuska, Pooya Farshim, and Arno Mittelbach. Indistinguishability obfuscation and UCEs: The case of computationally unpredictable sources. In Juan A. Garay and Rosario Gennaro, editors, *CRYPTO 2014, Part I*, volume 8616 of *LNCS*, pages 188–205. Springer, August 2014.
- BH15. Mihir Bellare and Viet Tung Hoang. Resisting randomness subversion: Fast deterministic and hedged public-key encryption in the standard model. In Elisabeth Oswald and Marc Fischlin, editors, *EUROCRYPT 2015, Part II*, volume 9057 of *LNCS*, pages 627–656. Springer, April 2015.
- BHK13. Mihir Bellare, Viet Tung Hoang, and Sriram Keelveedhi. Instantiating random oracles via UCEs. In Ran Canetti and Juan A. Garay, editors, *CRYPTO 2013, Part II*, volume 8043 of *LNCS*, pages 398–415. Springer, August 2013.
- BHK14. Mihir Bellare, Viet Tung Hoang, and Sriram Keelveedhi. Cryptography from compression functions: The UCE bridge to the ROM. In Juan A. Garay and Rosario Gennaro, editors, *CRYPTO 2014, Part I*, volume 8616 of *LNCS*, pages 169–187. Springer, August 2014.
- BHKR13. Mihir Bellare, Viet Tung Hoang, Sriram Keelveedhi, and Phillip Rogaway. Efficient garbling from a fixed-key blockcipher. In *2013 IEEE Symposium on Security and Privacy*, pages 478–492. IEEE Computer Society Press, May 2013.
- BMG09. Boaz Barak and Mohammad Mahmoody-Ghidary. Merkle puzzles are optimal - an $O(n^2)$ -query attack on any key exchange from a random oracle. In Shai Halevi, editor, *CRYPTO 2009*, volume 5677 of *LNCS*, pages 374–390. Springer, August 2009.
- BR94. Mihir Bellare and John Rompel. Randomness-efficient oblivious sampling. In *35th FOCS*, pages 276–287. IEEE Computer Society Press, November 1994.
- BR06. Mihir Bellare and Phillip Rogaway. The security of triple encryption and a framework for code-based game-playing proofs. In Serge Vaudenay, editor, *EUROCRYPT 2006*, volume 4004 of *LNCS*, pages 409–426. Springer, May / June 2006.
- BS16. Mihir Bellare and Igors Stepanovs. Point-function obfuscation: A framework and generic constructions. In Eyal Kushilevitz and Tal Malkin, editors, *TCC 2016-A, Part II*, volume 9563 of *LNCS*, pages 565–594. Springer, January 2016.
- CGH98. Ran Canetti, Oded Goldreich, and Shai Halevi. The random oracle methodology, revisited (preliminary version). In *30th ACM STOC*, pages 209–218. ACM Press, May 1998.
- CPS08. Jean-Sébastien Coron, Jacques Patarin, and Yannick Seurin. The random oracle model and the ideal cipher model are equivalent. In David Wagner, editor, *CRYPTO 2008*, volume 5157 of *LNCS*, pages 1–20. Springer, August 2008.
- CS14. Shan Chen and John P. Steinberger. Tight security bounds for key-alternating ciphers. In Phong Q. Nguyen and Elisabeth Oswald, editors, *EUROCRYPT 2014*, volume 8441 of *LNCS*, pages 327–350. Springer, May 2014.
- CS16. Benoît Cogliati and Yannick Seurin. Strengthening the known-key security notion for block ciphers. In Thomas Peyrin, editor, *FSE 2016*, volume 9783 of *LNCS*, pages 494–513. Springer, March 2016.
- DGG⁺15. Yevgeniy Dodis, Chaya Ganesh, Alexander Golovnev, Ari Juels, and Thomas Ristenpart. A formal treatment of backdoored pseudorandom generators. In Elisabeth Oswald and Marc Fischlin, editors, *EUROCRYPT 2015, Part I*, volume 9056 of *LNCS*, pages 101–126. Springer, April 2015.
- DKT16. Dana Dachman-Soled, Jonathan Katz, and Aishwarya Thiruvengadam. 10-round feistel is indifferentiable from an ideal cipher. In Marc Fischlin and Jean-Sébastien Coron, editors, *EUROCRYPT 2016, Part II*, volume 9666 of *LNCS*, pages 649–678. Springer, May 2016.
- DS16. Yuanxi Dai and John P. Steinberger. Indifferentiability of 8-round feistel networks. In Matthew Robshaw and Jonathan Katz, editors, *CRYPTO 2016, Part I*, volume 9814 of *LNCS*, pages 95–120. Springer, August 2016.
- DSSL16. Yevgeniy Dodis, Martijn Stam, John P. Steinberger, and Tianren Liu. Indifferentiability of confusion-diffusion networks. In Marc Fischlin and Jean-Sébastien Coron, editors, *EUROCRYPT 2016, Part II*, volume 9666 of *LNCS*, pages 679–704. Springer, May 2016.
- FS87. Amos Fiat and Adi Shamir. How to prove yourself: Practical solutions to identification and signature problems. In Andrew M. Odlyzko, editor, *CRYPTO’86*, volume 263 of *LNCS*, pages 186–194. Springer, August 1987.

- GT16. Peter Gazi and Stefano Tessaro. Provably robust sponge-based PRNGs and KDFs. In Marc Fischlin and Jean-Sébastien Coron, editors, *EUROCRYPT 2016, Part I*, volume 9665 of *LNCS*, pages 87–116. Springer, May 2016.
- HKT11. Thomas Holenstein, Robin Künzler, and Stefano Tessaro. The equivalence of the random oracle model and the ideal cipher model, revisited. In Lance Fortnow and Salil P. Vadhan, editors, *43rd ACM STOC*, pages 89–98. ACM Press, June 2011.
- HR10. Viet Tung Hoang and Phillip Rogaway. On generalized Feistel networks. In Tal Rabin, editor, *CRYPTO 2010*, volume 6223 of *LNCS*, pages 613–630. Springer, August 2010.
- IR89. Russell Impagliazzo and Steven Rudich. Limits on the provable consequences of one-way permutations. In *21st ACM STOC*, pages 44–61. ACM Press, May 1989.
- KR07. Lars R. Knudsen and Vincent Rijmen. Known-key distinguishers for some block ciphers. In Kaoru Kurosawa, editor, *ASIACRYPT 2007*, volume 4833 of *LNCS*, pages 315–324. Springer, December 2007.
- LR88. Michael Luby and Charles Rackoff. How to construct pseudorandom permutations from pseudorandom functions. *SIAM Journal on Computing*, 17(2), 1988.
- MH14. Takahiro Matsuda and Goichiro Hanaoka. Chosen ciphertext security via UCE. In Hugo Krawczyk, editor, *PKC 2014*, volume 8383 of *LNCS*, pages 56–76. Springer, March 2014.
- Mit14. Arno Mittelbach. Salvaging indistinguishability in a multi-stage setting. In Phong Q. Nguyen and Elisabeth Oswald, editors, *EUROCRYPT 2014*, volume 8441 of *LNCS*, pages 603–621. Springer, May 2014.
- MPR07. Ueli M. Maurer, Krzysztof Pietrzak, and Renato Renner. Indistinguishability amplification. In Alfred Menezes, editor, *CRYPTO 2007*, volume 4622 of *LNCS*, pages 130–149. Springer, August 2007.
- MPS12. Avradip Mandal, Jacques Patarin, and Yannick Seurin. On the public indistinguishability and correlation intractability of the 6-round Feistel construction. In Ronald Cramer, editor, *TCC 2012*, volume 7194 of *LNCS*, pages 285–302. Springer, March 2012.
- MRH04. Ueli M. Maurer, Renato Renner, and Clemens Holenstein. Indistinguishability, impossibility results on reductions, and applications to the random oracle methodology. In Moni Naor, editor, *TCC 2004*, volume 2951 of *LNCS*, pages 21–39. Springer, February 2004.
- NR97. Moni Naor and Omer Reingold. On the construction of pseudo-random permutations: Luby-Rackoff revisited (extended abstract). In *29th ACM STOC*, pages 189–199. ACM Press, May 1997.
- Pat92. Jacques Patarin. New results on pseudorandom permutation generators based on the DES scheme. In Joan Feigenbaum, editor, *CRYPTO'91*, volume 576 of *LNCS*, pages 301–312. Springer, August 1992.
- RR00. Zulfikar Ramzan and Leonid Reyzin. On the round security of symmetric-key cryptographic primitives. In Mihir Bellare, editor, *CRYPTO 2000*, volume 1880 of *LNCS*, pages 376–393. Springer, August 2000.
- RS08. Phillip Rogaway and John P. Steinberger. Security/efficiency tradeoffs for permutation-based hashing. In Nigel P. Smart, editor, *EUROCRYPT 2008*, volume 4965 of *LNCS*, pages 220–236. Springer, April 2008.
- SSS93. Jeanette P. Schmidt, Alan Siegel, and Aravind Srinivasan. Chernoff-hoeffding bounds for applications with limited independence. In Vijaya Ramachandran, editor, *4th SODA*, pages 331–340. ACM-SIAM, January 1993.
- ST17. Pratik Soni and Stefano Tessaro. Public-seed pseudorandom permutations. In Jean-Sébastien Coron and Jesper Buus Nielsen, editors, *EUROCRYPT 2017, Part II*, volume 10211 of *LNCS*, pages 412–441. Springer, May 2017.
- WC81. Mark N. Wegman and Larry Carter. New hash functions and their use in authentication and set equality. *Journal of Computer and System Sciences*, 22:265–279, 1981.

A Proofs for Section 3

A.1 Proof of Proposition 1

Proof. The proof follows from a simple application of the H -coefficient technique [Pat92, CS14]. In particular, we describe the interaction of S, D in both cases $b = 0$ and $b = 1$ using a *transcript* τ which contains:

- The sequence of queries made by S to the permutation, in either direction, which consists of tuples (i, U, V) , indicating that either $\mathcal{O}(i, (+, U))$ was queried, and returned V , or $\mathcal{O}(i, (-, V))$ was queried, returning U .

- The keys $s_1^{\text{in}}, s_1^{\text{out}}, \dots, s_r^{\text{in}}, s_r^{\text{out}}$ given to D . Recall that in the “real-world”, where S interacts with the NR construction, these are the actual keys. In the “ideal world”, these keys are generated by running P.Kg $2r$ times, independently of the permutation.

We also say that the transcript τ is *bad* if there exist two entries (i, U, V) and (i, U', V') for $U' \neq U, V \neq V'$ such that $P_1(s_i^{\text{in}}, U) = P_1(s_i^{\text{in}}, U')$ or $P_1(s_i^{\text{out}}, V) = P_1(s_i^{\text{out}}, V')$.

Fix a good transcript τ . Let $\mathbf{p}_1(\tau)$ be the interpolation probability of the transcript τ when interacting with NR construction (using random round functions sampled from \mathbf{F}), i.e., this is the probability that (S, D) obtains the answer in the transcript when asking the corresponding queries. Let $\mathbf{p}_0(\tau)$ be the probability in the ideal world, where the oracle queries are answered by independent permutations. Also, let q_i be the number distinct entries of form (i, U, V) , and assume wlog $i \in [r]$. Then,

$$\mathbf{p}_0(\tau) = \mathbf{p}(s_1^{\text{in}}, s_1^{\text{out}}, \dots, s_r^{\text{in}}, s_r^{\text{out}}) \cdot \prod_{i=1}^r \prod_{j=0}^{q_i-1} \frac{1}{2^{2n-j}},$$

where $\mathbf{p}(s_1^{\text{in}}, s_1^{\text{out}}, \dots, s_r^{\text{in}}, s_r^{\text{out}})$ is the probability that these $2r$ keys are sampled by P.Kg.

Let us compare this with $\mathbf{p}_1(\tau)$. To this end, fix $i \in [r]$, then for every entry (i, U^j, V^j) in τ , $j \in \{1, \dots, q_i\}$, define the corresponding values $x_0^j \parallel x_1^j = P(s_i^{\text{in}}, U^j)$ and $x_2^j \parallel x_3^j = P(s_i^{\text{out}}, V^j)$. If τ is good, then the values x_1^j are distinct, and so are x_2^j . This in particular means that we are interested in the probability, over the choice of a random f_i , that $f_i(0 \parallel x_1^j) = x_0^j \oplus x_2^j$ and $f_i(1 \parallel x_2^j) = x_3^j \oplus x_1^j$ for all j , which is exactly 2^{-2nq_i} . Overall, thus, taking a product over all $i = 1, \dots, r$,

$$\mathbf{p}_1(\tau) = \mathbf{p}(s_1^{\text{in}}, s_1^{\text{out}}, \dots, s_r^{\text{in}}, s_r^{\text{out}}) \cdot \prod_{i=1}^r 2^{-2nq_i} = \mathbf{p}(s_1^{\text{in}}, s_1^{\text{out}}, \dots, s_r^{\text{in}}, s_r^{\text{out}}) \cdot 2^{-2nq},$$

which in particular means that

$$\frac{\mathbf{p}_1(\tau)}{\mathbf{p}_0(\tau)} \geq \prod_{i=1}^r \prod_{j=0}^{q_i-1} \left(1 - \frac{j}{2^{2n}}\right) \geq 1 - \sum_{i=1}^r q_i^2 / 2^{2n} \geq 1 - \frac{q^2}{2^{2n}} = 1 - \varepsilon,$$

where we have used that $(1-a)(1-b) \geq 1-a-b$ for all $a, b \geq 0$, and we let $\varepsilon = \frac{q^2}{2^{2n}}$. Also, using α -right universality, and the union bound, it is not hard to see that the probability that an ideal transcript is *bad* is at most

$$\delta := \sum_{i=1}^r 2q_i^2 \alpha \leq 2q^2 \alpha.$$

Then, using the H coefficient method, we can conclude that the bound is at most $\varepsilon + \delta$. This concludes the proof. \square

A.2 Unpredictability Argument in the Proof of Theorem 1

Assume wlog in the following that \bar{P} is deterministic. We consider three games \mathbf{G}_{11} , \mathbf{G}_{12} , and \mathbf{G}_{13} .

Game \mathbf{G}_{11} is a small syntactical alteration of Game f-Pred $_{n+1, n, \bar{S}}^{\bar{P}}$. It keeps track in a set Q_P of the queries issued to \bar{O} by the simulated S within \bar{S} , i.e., an entry (i, U, V) is added to Q_P whenever

$\overline{\mathcal{O}}(i, (+, U))$ is queried, returning V , or (symmetrically) $\overline{\mathcal{O}}(i, (-, V))$ is queried, returning U . Then, at the end of S 's execution, we define

$$Q \leftarrow \{(i, 0 \parallel P_1(s_i^{\text{in}}, U)), (i, 1 \parallel P_1(s_i^{\text{out}}, V)) : (i, U, V) \in Q_P\},$$

where P_1 denotes the second n -bit half of the output of P . We also let $Q' \leftarrow \overline{P}(L, \mathbf{s})$ be the output of \overline{P} given the leakage made by L and the seeds $\mathbf{s} = (s_1^{\text{in}}, s_1^{\text{out}}, \dots, s_r^{\text{in}}, s_r^{\text{out}})$. Finally, G_{11} returns true iff $Q' \cap Q \neq \emptyset$. Clearly, as Q exactly captures the set of queries \overline{S} makes to its oracle, the game returns true iff \overline{P} predicts correctly, and we thus conclude that $\text{Adv}_{n+1, n, \overline{S}}^{\text{f-pred}}(\overline{P}) = \Pr[G_{11}]$.

The definition of G_{11} however allows to easily introduce G_{12} , which answers S 's $\overline{\mathcal{O}}$ -queries via r randomly sampled permutations π_1, \dots, π_r , i.e., $\overline{\mathcal{O}}(i, (+, U)) = \pi_i(U)$ and $\overline{\mathcal{O}}(i, (-, V)) = \pi_i^{-1}(V)$. This is the *only* difference. In particular, we still keep track of these queries in Q_P , compute Q , and set the winning condition as in G_{11} . It is not hard to see that Proposition 1 directly bounds $\Pr[G_{11}] - \Pr[G_{12}]$, and thus

$$\text{Adv}_{n+1, n, \overline{S}}^{\text{f-pred}}(\overline{P}) \leq q^2 \cdot \left(2\alpha + \frac{1}{2^{2n}}\right) + \Pr[G_{12}].$$

We are now going to construct a predictor P for S , whose advantage is related to $\Pr[G_{12}]$. This P will generally be inefficient and (interestingly enough) will in fact not make any usage of the predictor \overline{P} . To this end, we first introduce a new predictor \overline{P}' which on input L, \mathbf{s} outputs a singleton set consisting of the most likely element from $\overline{P}(L, \mathbf{s})$ to be in Q . We let G_{13} be G_{12} with \overline{P}' instead of \overline{P} . Clearly, $\Pr[G_{12}] \leq p \cdot \Pr[G_{13}]$.

In the following, in G_{13} , let us consider the conditional distribution \mathcal{Q}_L of Q_P given some fixed L . Also, let $p(L)$ be the probability that a particular L is output by S in G_{12} , and let $q(\mathbf{s})$ be the probability that the seed-vector \mathbf{s} is chosen, where recall that $\mathbf{s} = (s_1^{\text{in}}, s_1^{\text{out}}, \dots, s_r^{\text{in}}, s_r^{\text{out}})$ contains the seeds for P . Finally, we let $Q(\mathbf{s}, Q_P)$ is the set Q computed from Q_P using the keys \mathbf{s} . Then, we can re-arrange the execution of G_{13} so that: (1) We first sample L according to the probability distribution $p(\cdot)$, (2) Then sample Q_P according to \mathcal{Q}_L , (3) Sample \mathbf{s} according to $q(\cdot)$, and (4) Check whether $\overline{P}'(L, \mathbf{s}) \in Q(\mathbf{s}, Q_P)$. Then,

$$\Pr[G_{13}] = \sum_{L, \mathbf{s}} p(L) \cdot q(\mathbf{s}) \cdot \Pr_{Q_P \stackrel{\mathbf{s}}{\leftarrow} \mathcal{Q}_L} [\overline{P}'(L, \mathbf{s}) \in Q(\mathbf{s}, Q_P)].$$

Note now that for any fixed L and \mathbf{s} ,

$$\begin{aligned} \Pr_{Q_P \stackrel{\mathbf{s}}{\leftarrow} \mathcal{Q}_L} [\overline{P}'(L, \mathbf{s}) \in Q(\mathbf{s}, Q_P)]^2 &= \Pr_{Q_P, Q'_P \stackrel{\mathbf{s}}{\leftarrow} \mathcal{Q}_L} [\overline{P}'(L, \mathbf{s}) \in Q(\mathbf{s}, Q_P) \cap Q(\mathbf{s}, Q'_P)] \\ &\leq \Pr_{Q_P, Q'_P \stackrel{\mathbf{s}}{\leftarrow} \mathcal{Q}_L} [Q(\mathbf{s}, Q_P) \cap Q(\mathbf{s}, Q'_P) \neq \emptyset]. \end{aligned}$$

Therefore,

$$\begin{aligned} \Pr[G_{13}] &\leq \sum_{s, L} p(L) \cdot q(\mathbf{s}) \cdot \sqrt{\Pr_{Q_P, Q'_P \stackrel{\mathbf{s}}{\leftarrow} \mathcal{Q}_L} [Q(\mathbf{s}, Q_P) \cap Q(\mathbf{s}, Q'_P) \neq \emptyset]} \\ &\leq \sqrt{\sum_{s, L} p(L) \cdot q(\mathbf{s}) \cdot \Pr_{Q_P, Q'_P \stackrel{\mathbf{s}}{\leftarrow} \mathcal{Q}_L} [Q(\mathbf{s}, Q_P) \cap Q(\mathbf{s}, Q'_P) \neq \emptyset]} \end{aligned}$$

where the last inequality follows by Jensen's inequality.

Now, in the following, assume L is fixed such that $p(L) > 0$, and we sample \mathbf{s} from $q(\cdot)$, as well as $Q_P, Q'_P \stackrel{\mathbf{s}}{\leftarrow} \mathcal{Q}_L$. We also define U_P and V_P such that $(i, U) \in U_P$ and $(i, V) \in V_P$ if and only if $(i, U, V) \in Q_P$. Define U'_P and V'_P analogously with respect to Q'_P . Then, let $\text{Good} = \text{Good}(Q_P, Q'_P)$ be the event that $U_P \cap U'_P \neq \emptyset$ or $V_P \cap V'_P \neq \emptyset$. Accordingly,

$$\Pr [Q(\mathbf{s}, Q_P) \cap Q(\mathbf{s}, Q'_P) \neq \emptyset] \leq \Pr [\text{Good}] + \Pr [Q(\mathbf{s}, Q_P) \cap Q(\mathbf{s}, Q'_P) \neq \emptyset \mid \overline{\text{Good}}] .$$

The latter probability is the probability that there exist $(i, U) \in U_P, (i, U') \in U'_P$ such that $P_1(s_i^{\text{in}}, U) = P_1(s_i^{\text{in}}, U')$ and $U \neq U'$, or there exist $(i, V) \in V_P, (i, V') \in V'_P$ such that $P_1(s_i^{\text{out}}, V) = P_1(s_i^{\text{out}}, V')$ and $V \neq V'$. Note that for any such pairs the probability of such a collision is at most α by α -right universality, and thus the overall probability is $2q^2\alpha$ by the union bound. Therefore, combining this with the above, we get

$$\Pr [\text{G}_{13}] \leq \sqrt{2q^2\alpha + \sum_L p(L) \cdot \Pr_{Q_P, Q'_P \stackrel{\mathbf{s}}{\leftarrow} \mathcal{Q}_L} [\text{Good}(Q_P, Q'_P)]} \quad (20)$$

To conclude the proof, we consider the predictor P against S which simply, given L , samples Q'_P from \mathcal{Q}_L , and then compute U'_P, V'_P , and outputs them. Since the original queries Q_P are also distributed according to \mathcal{Q}_L , we can rewrite (20) as

$$\Pr [\text{G}_{13}] \leq \sqrt{2q^2\alpha + \text{Adv}_{2n, S}^{\text{P-pred}}(P)} , \quad (21)$$

which concludes the proof.

A.3 Game Descriptions in Proof of Theorem 2

The description of game $\text{G}_1, \dots, \text{G}_5$ used in the proof are given in Figures 7 and 8.

A.4 Proof of Lemma 1

Proof (Of Lemma 1). Recall that \bar{R} makes exactly p distinct queries, and is deterministic. Also, recall that Sam goes through p outer iterations, and $\eta = 4p/\varepsilon \ln(4p/\varepsilon)$ inner iterations, and consequently we denote an iteration of Sam with a pair (j, k) , where $j \in [p]$ and $k \in [\eta]$.

Also, in the following, let us *fix* the leakage L , the keys \mathbf{s} for P , and the values Y_1, \dots, Y_p , where Y_i is the uniformly sampled value used to answer \bar{R} 's i -th query in G_5 if the query is not in \tilde{X} . (We can think of the values Y_1, \dots, Y_p as being pre-sampled, and then used as needed.) We will prove an upper bound on bad being set, even given these are fixed. The final bound then follows by averaging.

SOME MORE NOTATION. For every $(j, k) \in [p] \times [\eta]$, we denote by $\tilde{Q}_{P, j, k}$ the contents of \tilde{Q}_P within Sam at the end of iteration (j, k) . Further, we let $\tilde{Q}_{P, 1, 0} = \emptyset$, and $\tilde{Q}_{P, j, 0} = \tilde{Q}_{P, j-1, \eta}$ for all $j \geq 2$. Note that the final output is $\tilde{Q}_P = \tilde{Q}_{P, p, \eta}$. Also, for \mathbf{s} defined as in the games, and any set Q of triples (i, U, V) , we define for convenience $X(\mathbf{s}, Q)$ as the set

$$X(\mathbf{s}, Q) = \{(i, 0 \parallel x_1) : \exists (i, U, V, j) \in Q : P_1(s_i^{\text{in}}, U) = x_1\} \cup \{(i, 1 \parallel x_2) : \exists (i, U, V, j) \in Q : P_1(s_i^{\text{out}}, V) = x_2\} .$$

<p><u>MAIN G_1:</u></p> <p>$Q_P \leftarrow \emptyset, (r, t) \xleftarrow{\\$} S(\varepsilon)$</p> <p>$f_1, \dots, f_r \xleftarrow{\\$} \text{Funcs}(n+1, n)$</p> <p>$\mathbf{s} = (s_1^{\text{in}}, s_1^{\text{out}}, \dots, s_r^{\text{in}}, s_r^{\text{out}}) \xleftarrow{\\$} \text{P.Kg}$</p> <p>$L \xleftarrow{\\$} S^{\overline{O}}(t)$</p> <p>$b' \leftarrow \overline{R}^{\mathcal{O}'}(L, \mathbf{s})$</p> <p>return ($b' = 0$)</p> <p><u>MAIN G_2:</u></p> <p>$Q_P \leftarrow \emptyset, (r, t) \xleftarrow{\\$} S(\varepsilon)$</p> <p>$f_1, \dots, f_r \xleftarrow{\\$} \text{Funcs}(n+1, n)$</p> <p>$\mathbf{s} = (s_1^{\text{in}}, s_1^{\text{out}}, \dots, s_r^{\text{in}}, s_r^{\text{out}}) \xleftarrow{\\$} \text{P.Kg}$</p> <p>$L \xleftarrow{\\$} S^{\overline{O}}(t)$</p> <p>for all $(i, U, V) \in Q_P$ do</p> <p style="padding-left: 20px;">$x_0 \parallel x_1 \leftarrow \text{P}(s_i^{\text{in}}, U)$</p> <p style="padding-left: 20px;">$x_2 \parallel x_3 \leftarrow \text{P}(s_i^{\text{out}}, V)$</p> <p style="padding-left: 20px;">$T[i, 0 \parallel x_1] \leftarrow x_0 \oplus x_2$</p> <p style="padding-left: 20px;">$T[i, 1 \parallel x_2] \leftarrow x_1 \oplus x_3$</p> <p style="padding-left: 20px;">$X \leftarrow \bigcup \{(i, 0 \parallel x_1), (i, 1 \parallel x_2)\}$</p> <p>$b' \leftarrow \overline{R}^{\mathcal{O}'}(L, \mathbf{s})$</p> <p>return ($b' = 0$)</p>	<p><u>Proc. $\overline{O}(i, (\sigma, U))$:</u> // G_1, G_2, G_3</p> <p>if $\sigma = +$ then</p> <p style="padding-left: 20px;">$x_0 \parallel x_1 \leftarrow \text{P}(s_i^{\text{in}}, U)$</p> <p style="padding-left: 20px;">$x_2 \leftarrow f_i(0 \parallel x_1) \oplus x_0$</p> <p style="padding-left: 20px;">$x_3 \leftarrow f_i(1 \parallel x_2) \oplus x_1$</p> <p style="padding-left: 20px;">$V \leftarrow \text{P}^{-1}(s_i^{\text{out}}, x_3 \parallel x_2)$</p> <p style="padding-left: 20px;">$Q_P \leftarrow Q_P \cup \{(i, U, V)\}$</p> <p>else</p> <p style="padding-left: 20px;">$x_3 \parallel x_2 \leftarrow \text{P}(s_i^{\text{out}}, U)$</p> <p style="padding-left: 20px;">$x_1 \leftarrow f_i(1 \parallel x_2) \oplus x_3$</p> <p style="padding-left: 20px;">$x_0 \leftarrow f_i(0 \parallel x_1) \oplus x_2$</p> <p style="padding-left: 20px;">$V \leftarrow \text{P}^{-1}(s_i^{\text{in}}, x_0 \parallel x_1)$</p> <p style="padding-left: 20px;">$Q_P \leftarrow Q_P \cup \{(i, V, U)\}$</p> <p>return V</p> <p><u>Proc. $\mathcal{O}'(i, x)$:</u> // G_2</p> <p>if $T[i, x] \neq \perp$ then $y \leftarrow T[i, x]$</p> <p>else $y \xleftarrow{\\$} \{0, 1\}^n$</p> <p>return y</p> <p><u>Proc. $\mathcal{O}'(i, x)$:</u> // G_1</p> <p>return $f_i(X)$.</p>
---	---

Fig. 7: **Games to prove reset-security of the source \overline{S} .** Here, \overline{R} is deterministic and assumed to make p distinct queries, and thus it is not necessary for the oracle \mathcal{O}' to keep state.

In particular, $X(\mathbf{s}, \tilde{Q}_{P,j,\eta}) = \tilde{X}^1 \cup \dots \cup \tilde{X}^j$. For this reason, we note in particular that for every $j \in [p]$, the j -th query \mathbf{q}_j of \overline{R} is uniquely determined by $L, \mathbf{s}, Y_1, \dots, Y_{j-1}$ and $\tilde{Q}_{P,j,0}$. This fact relies on \overline{R} being deterministic, and somewhat more subtly, the fact that the values \tilde{T} are never overwritten.

THE BAD EVENTS. For $j \in [p]$, we also define the event bad_j that for the j -th query of \overline{R} we have $\mathbf{q}_j \notin \tilde{X}^1 \cup \dots \cup \tilde{X}^j$, and $\mathbf{q}_j \in X_P$, which is exactly the event that bad is set when answering the j -th query of \overline{R} . We will show that $\Pr[\text{bad}_j] \leq \varepsilon/2p$, and consequently

$$\Pr[\text{bad}] \leq \sum_{j=1}^p \Pr[\text{bad}_j] \leq p \cdot \varepsilon/2p = \varepsilon/2. \quad (22)$$

To bound $\Pr[\text{bad}_j]$ for any fixed $j \in [p]$, assume that we further fix $\tilde{Q}_{P,j,0}$ to some possible value which can occur with positive probability. This also (by the above argument) uniquely fixed the j -th query \mathbf{q}_j .

Now, if $\mathbf{q}_j \in X(\mathbf{s}, \tilde{Q}_{P,j,0})$, then $\Pr[\text{bad}_j]$ is 0 conditioned on this. Therefore, let us assume that $\mathbf{q}_j \notin X(\mathbf{s}, \tilde{Q}_{P,j,0})$. Then, define for every $k \in [\eta]$,

$$\alpha_{j,k} = \Pr \left[\Delta_j \xleftarrow{\$} \mathcal{Q}(L, \tilde{Q}_{P,j,k-1}) : \mathbf{q}_j \in X(\mathbf{s}, \Delta_j) \right].$$

<p>MAIN G_3, G_4, G_5:</p> $c \leftarrow 0$ $Q_P \leftarrow \emptyset, (r, t) \xleftarrow{\$} S(\varepsilon)$ $\pi_1, \dots, \pi_r \xleftarrow{\$} \text{Perms}(2n)$ $f_1, \dots, f_r \xleftarrow{\$} \text{Funcs}(n+1, n)$ $\mathbf{s} = (s_1^{\text{in}}, s_1^{\text{out}}, \dots, s_r^{\text{in}}, s_r^{\text{out}}) \xleftarrow{\$} \text{P.Kg}$ $L \xleftarrow{\$} S^{\overline{\mathcal{O}}}(t)$ $\tilde{Q}_P \xleftarrow{\$} \text{Sam}^{\overline{\mathcal{O}}}(L)$ <p>for all $(i, U, V) \in Q_P$ do</p> $x_0 \parallel x_1 \leftarrow \text{P}(s_i^{\text{in}}, U)$ $x_2 \parallel x_2 \leftarrow \text{P}(s_i^{\text{out}}, V)$ $T[i, 0 \parallel x_1] \leftarrow x_0 \oplus x_2$ $T[i, 1 \parallel x_2] \leftarrow x_1 \oplus x_3$ $X \leftarrow \{(i, 0 \parallel x_1), (i, 1 \parallel x_2)\}$ <p>for $j = 1$ to p do</p> <p>for all $(i, U, V, j) \in \tilde{Q}_P$ do</p> $x_0 \parallel x_1 \leftarrow \text{P}(s_i^{\text{in}}, U)$ $x_3 \parallel x_2 \leftarrow \text{P}(s_i^{\text{out}}, V)$ $\tilde{X}^j \xleftarrow{\$} \tilde{X}^j \cup \{(i, 0 \parallel x_1), (i, 1 \parallel x_2)\}$ <p>if $\tilde{T}[i, 0 \parallel x_1] = \perp$ then</p> $\tilde{T}[i, 0 \parallel x_1] \leftarrow x_0 \oplus x_2$ <p>if $\tilde{T}[i, 1 \parallel x_2] = \perp$ then</p> $\tilde{T}[i, 1 \parallel x_2] \leftarrow x_1 \oplus x_3$ $b' \leftarrow \overline{R}^{\mathcal{O}'}(L, \mathbf{s})$ <p>return $(b' = 0)$</p>	<p>Proc. $\mathcal{O}'(i, x)$: // G_3, G_4</p> $c \leftarrow c + 1, \tilde{X} \xleftarrow{\$} \tilde{X}^c$ <p>if $(i, x) \in \tilde{X}$ then $y \leftarrow \tilde{T}[i, x]$</p> <p>elseif $(i, x) \in X$ then</p> $\text{bad} \leftarrow \text{true}, y \leftarrow T[i, X]$ <p>elseif $y \xleftarrow{\\$} \{0, 1\}^n$</p> <p>return y</p> <p>Proc. $\mathcal{O}'(i, x)$: // G_5</p> $c \leftarrow c + 1, \tilde{X} \xleftarrow{\$} \tilde{X}^c$ <p>if $(i, x) \in \tilde{X}$ then $y \leftarrow \tilde{T}[i, x]$</p> <p>elseif $(i, x) \in X$ then</p> $\text{bad} \leftarrow \text{true}, y \xleftarrow{\$} \{0, 1\}^n$ <p>elseif $y \xleftarrow{\\$} \{0, 1\}^n$</p> <p>return y</p> <p>Proc. $\overline{\mathcal{O}}(i, (\sigma, U))$: // G_4, G_5</p> <p>if $\sigma = +$ then</p> $V \xleftarrow{\$} \pi_i(U)$ $Q_P \leftarrow Q_P \cup \{(i, U, V)\}$ <p>else</p> $V \xleftarrow{\$} \pi_i^{-1}(U)$ $Q_P \leftarrow Q_P \cup \{(i, V, U)\}$ <p>return V</p>
--	--

Fig. 8: **Games to prove reset-security of the source \overline{S} . (Continued)** Here, \overline{R} is deterministic and assumed to make p distinct queries, and thus it is not necessary for the oracle \mathcal{O}' to keep state.

Now, either there exists $k \in [\eta]$ such that $\alpha_{j,k} \leq \varepsilon/4p$, in which case the probability that $\mathbf{q}_j \in X$ is at most $\varepsilon/4p$, and thus the probability of bad_j conditioned on this is at most $\varepsilon/4p$. This follows from the following fact (made explicit in [IR89]): $\alpha_{j,k}$ is exactly the conditional probability, given the acquired information so far, that $\mathbf{q}_j \in X$ in the actual execution, and thus, conditioned on $\alpha_{j,k} = p$, the probability that $\mathbf{q}_j \in X$ is indeed p .

The other case is that no such k exists, i.e., $\alpha_{j,k} > \frac{\varepsilon}{4p}$ for all $k \in [\eta]$. In this case, we are going to necessarily have $\mathbf{q}_j \in X(\tilde{Q}_{P,j,\eta})$, except with probability

$$(1 - \varepsilon/4p)^\eta \leq e^{-(\varepsilon/4p)\eta} = \varepsilon/4p$$

using the fact that $\eta = 4p/\varepsilon \ln(4p/\varepsilon)$. Therefore, the probability that bad_j is set is at most $2 \times \varepsilon/4p = \varepsilon/2p$.

This concludes the proof of Lemma 1. □

A.5 Analysis of the $b = 1$ case

To conclude the proof, we still need to analyze the $b = 1$ case, i.e., the case where the source and the reset adversary are given independent oracles. To this end, we introduce two games G_{11}, G_{12} which simulate an execution of R against S where in the former game, R and S access independent sets of permutations, whereas in G_{12} these permutations are instantiated using the NR construction, sharing however the underlying keys \mathbf{s} for P . Clearly,

$$\Pr [\neg\text{p-Reset}_S^R \mid b = 1] = \Pr [G_{11}] . \quad (23)$$

To transition from G_{11} to G_{12} , we can use Proposition 1 *twice*, first with respect to S 's q queries to replace π_1^0, π_2^0, \dots with their counterparts obtained from the NR construction, and then another time to do the same π_1^1, π_1^1, \dots (this time there are $2q\tau$ queries being made by Sam to take into account). So overall,

$$\begin{aligned} \Pr [G_{11}] &\leq \Pr [G_{12}] + (q^2 + (2q\tau)^2) \left(2\alpha + \frac{1}{2^{2n}} \right) \\ &\leq \Pr [G_{12}] + (q + 2q\tau)^2 \left(2\alpha + \frac{1}{2^{2n}} \right) . \end{aligned} \quad (24)$$

Now, looking at G_{12} , we observe that $\bar{T}[i, x] = f_i^1(x)$ always holds by construction. Therefore, when running \bar{R} , the values returned by $\mathcal{O}'(i, x)$ are always either equal $f_i^b(x)$ or freshly random, and thus in both case random and independent than the values of f_1^0, \dots, f_r^0 . In other words, this perfectly simulates an execution of \bar{R} against \bar{S} in the $b = 1$ case, i.e.

$$\Pr [G_{12}] = \Pr [\neg\text{f-Reset}_{\bar{S}}^{\bar{R}} \mid b = 1] , \quad (25)$$

and thus combining (23), (24), and (25), we get

$$\Pr [\neg\text{p-Reset}_S^R \mid b = 1] \leq \Pr [\neg\text{f-Reset}_{\bar{S}}^{\bar{R}} \mid b = 1] + (q + 2q\tau)^2 \left(2\alpha + \frac{1}{2^{2n}} \right) , \quad (26)$$

as we wanted to show.

B Proofs for Section 4

B.1 Proof of Lemma 2

Proof. Before we proceed with the proof, note that it is enough to assume that R only contains k -sized sets in it. If not, then we can use the partitioning $R = \bigcup_{i=1}^k R_i$, where R_i only contains sets from R with exactly i input-output pairs. We can then apply the argument independently to obtain R_i^* , and let $R^* = \bigcup_{i=1}^k R_i^*$. The above lemma implies that R^* is $(\delta \cdot k)$ -strongly evasive, and that for every $S \in R$, there exists $\emptyset \neq S^* \subseteq S$ such that $S^* \in R^*$.

In the following, let $N := |X|$. We first describe a procedure to obtain R^* from R , and below show that the resulting R^* always has the desired properties. We initially set $R^* = R$. There are three nested iterations in our procedure. The outer iteration goes over all $j = k - 1, k - 2, \dots, 0$. The middle iteration iterates over all $k' = j + 1, \dots, k$. The inner iteration does two things:

<p><u>MAIN G_{11}, G_{12}:</u> $c \leftarrow 0$ $Q_P \leftarrow \emptyset, (r, t) \xleftarrow{\\$} S(\varepsilon)$ $\pi_1^0, \pi_1^1, \dots, \pi_r^0, \pi_r^1 \xleftarrow{\\$} \text{Perms}(2n)$ $f_1^0, f_1^1, \dots, f_r^0, f_r^1 \xleftarrow{\\$} \text{Funcs}(n+1, n)$ $\mathbf{s} = (s_1^{\text{in}}, s_1^{\text{out}}, \dots, s_r^{\text{in}}, s_r^{\text{out}}) \xleftarrow{\\$} \text{P.Kg}$ $L \xleftarrow{\\$} S^{\overline{\mathcal{O}}}(t)$ $\tilde{Q}_P \xleftarrow{\\$} \text{Sam}^{\overline{\mathcal{O}}}(L)$ for $j = 1$ to p do for all $(i, U, V, j) \in \tilde{Q}_P$ do $x_0 \parallel x_1 \leftarrow \mathbf{P}(s_i^{\text{in}}, U)$ $x_3 \parallel x_2 \leftarrow \mathbf{P}(s_i^{\text{out}}, V)$ $\tilde{X}^j \xleftarrow{\cup} \tilde{X}^j \cup \{(i, 0 \parallel x_1), (i, 1 \parallel x_2)\}$ if $\tilde{T}[i, 0 \parallel x_1] = \perp$ then $\tilde{T}[i, 0 \parallel x_1] \leftarrow x_0 \oplus x_2$ if $\tilde{T}[i, 1 \parallel x_2] = \perp$ then $\tilde{T}[i, 1 \parallel x_2] \leftarrow x_1 \oplus x_3$ $b' \leftarrow \overline{R}^{\mathcal{O}'}(L, \mathbf{s})$ return $(b' = 0)$</p>	<p><u>Proc. $\overline{\mathcal{O}}^b(i, (\sigma, U))$:</u> // G_{11} if $\sigma = +$ then $V \xleftarrow{\\$} \pi_i(U)$ else $V \xleftarrow{\\$} \pi_i^{-1}(U)$ return V</p> <p><u>Proc. $\overline{\mathcal{O}}^b(i, (\sigma, U))$:</u> // G_{12} if $\sigma = +$ then $x_0 \parallel x_1 \leftarrow \mathbf{P}(s_i^{\text{in}}, U)$ $x_2 \leftarrow f_i^b(0 \parallel x_1) \oplus x_0$ $x_3 \leftarrow f_i^b(1 \parallel x_2) \oplus x_1$ $V \leftarrow \mathbf{P}^{-1}(s_i^{\text{out}}, x_3 \parallel x_2)$ else $x_3 \parallel x_2 \leftarrow \mathbf{P}(s_i^{\text{out}}, U)$ $x_1 \leftarrow f_i^b(1 \parallel x_2) \oplus x_3$ $x_0 \leftarrow f_i^b(0 \parallel x_1) \oplus x_2$ $V \leftarrow \mathbf{P}^{-1}(s_i^{\text{in}}, x_0 \parallel x_1)$ return V</p> <p><u>Proc. $\mathcal{O}'(i, x)$:</u> // G_{11}, G_{12} $c \leftarrow c + 1, \tilde{X} \xleftarrow{\cup} \tilde{X}^c$ if $(i, x) \in \tilde{X}$ then $y \leftarrow \tilde{T}[i, x]$ elseif $y \xleftarrow{\\$} \{0, 1\}^n$ return y</p>
---	---

Fig. 9: **Games to prove reset-security of the source \overline{S} . (Continued)** Case $b = 1$. Here, \overline{R} is deterministic and assumed to make p distinct queries, and thus it is not necessary for the oracle \mathcal{O}' to keep state.

1. It goes over all $\mathbf{u}_1 < \dots < \mathbf{u}_{k'}$, and all $\mathbf{v}_1 < \dots < \mathbf{v}_j$, where $<$ denotes lexicographic ordering of elements of X . For each of them, it checks whether

$$|\{(\mathbf{v}_{j+1}, \dots, \mathbf{v}_{k'}) : \{(\mathbf{u}_1, \mathbf{v}_1), \dots, (\mathbf{u}_{k'}, \mathbf{v}_{k'})\} \in R^*\}| > \delta \cdot \prod_{i=j}^{k'-1} (N - i).$$

If so (and only if so), we remove *every* such set $\{(\mathbf{u}_1, \mathbf{v}_1), \dots, (\mathbf{u}_{k'}, \mathbf{v}_{k'})\}$ from R^* , and add instead $\{(\mathbf{u}_1, \mathbf{v}_1), \dots, (\mathbf{u}_j, \mathbf{v}_j)\}$ to R^* (unless it is already in R^*).

2. Still within the middle iteration (i.e., before changing j and k'), we also iterate over all $\mathbf{v}_1 < \dots < \mathbf{v}_{k'}, \mathbf{u}_1 < \dots < \mathbf{u}_j$, and perform the symmetric operation to what describe above.

Now, we first show that R^* is δ -strongly evasive. Assume to this end it is not, then there exists $j < k' \leq k, \mathbf{u}_1, \dots, \mathbf{u}_{k'} \in X$, all $\mathbf{v}_1, \dots, \mathbf{v}_j \in X$, such that

$$|\{(\mathbf{v}_{j+1}, \dots, \mathbf{v}_{k'}) : \{(\mathbf{u}_1, \mathbf{v}_1), \dots, (\mathbf{u}_{k'}, \mathbf{v}_{k'})\} \in R^*\}| > \delta \cdot \prod_{i=j}^{k'-1} (N - i).$$

Note that when we consider $\mathbf{u}_1, \dots, \mathbf{u}_{k'} \in X$ and $\mathbf{v}_1, \dots, \mathbf{v}_j \in X$ in our iteration, this must have been already true for R^* at that point, as after that, we have only added tuples of length at most

$j < k'$ to the relation R^* . (We have used crucially here that we iterate over j *backwards* here!) But then, all of these tuples would have been removed from R^* and never added back, and we have a contradiction. The symmetric argument holds when swapping inputs and outputs.

We also need to make sure that *for every* $S \in R$, there exists $S^* \subseteq S$ such that $S^* \neq \emptyset$ and $S^* \in R^*$. This will follow from a more general statement. Namely, denote by R_j^* the relation obtained at the end of the outer iteration for index j , with $R_k^* = R$, and $R_0^* = R^*$. Then, we show that R_1^* is (k, δ) -evasive.

Claim. R_j^* is $(k \cdot j, \delta^j)$ -evasive for $1 \leq j \leq k$.

Proof (Of claim). The proof goes by induction. We know R_k^* is (k^2, δ^k) -evasive. Assume towards a contradiction that R_j^* for $1 \leq j < k$ is not $(k \cdot j, \delta^j)$ -evasive, but R_{j+1}^* is $(k \cdot (j+1), \delta^{j+1})$ -evasive. Then, there exists a $k \cdot j$ -query adversary A_j satisfying R_j^* with probability larger than δ^j . We can get a $k(j+1)$ -query adversary A_{j+1} for R_{j+1}^* as follows. It first runs A_j , and if the output is a j' -sized set for $j' > j$, it outputs it, as a set of cardinality larger than j is in R_{j+1}^* iff it is in R_j^* , as the transition from $j+1$ to j only adds j -sized sets. In contrast, if the output is a j -set $\{(\mathbf{u}_1, \mathbf{v}_1), \dots, (\mathbf{u}_j, \mathbf{v}_j)\} \in R_j^*$, then it is not in R_{j+1}^* . However, one of the following applies, as this set was added in the process of going from R_{j+1}^* to R_j^* .

- There exist $\mathbf{u}_{j+1}, \dots, \mathbf{u}_{k'}$ such that the number of $(\mathbf{v}_{j+1}, \dots, \mathbf{v}_{k'})$ with $\{(\mathbf{u}_1, \mathbf{v}_1), \dots, (\mathbf{u}_{k'}, \mathbf{v}_{k'})\} \in R_{j+1}^*$ is larger than $\delta \cdot \prod_{i=j}^{k'-1} (N - i)$. In this case, A_{j+1} queries $\mathbf{u}_{j+1}, \dots, \mathbf{u}_{k'}$ to π , gets $\tilde{\mathbf{v}}_{j+1}, \dots, \tilde{\mathbf{v}}_{k'}$, and outputs $\{(\mathbf{u}_1, \mathbf{v}_1), \dots, (\mathbf{u}_j, \mathbf{v}_j), (\mathbf{u}_{j+1}, \tilde{\mathbf{v}}_{j+1}), \dots, (\mathbf{u}_{k'}, \tilde{\mathbf{v}}_{k'})\}$.
- There exist $\mathbf{v}_{j+1}, \dots, \mathbf{v}_{k'}$ such that the number of $(\mathbf{u}_{j+1}, \dots, \mathbf{u}_{k'})$ with $\{(\mathbf{u}_1, \mathbf{v}_1), \dots, (\mathbf{u}_{k'}, \mathbf{v}_{k'})\} \in R_{j+1}^*$ is larger than $\delta \cdot \prod_{i=j}^{k'-1} (N - i)$. In this case, A_{j+1} queries $\mathbf{v}_{j+1}, \dots, \mathbf{v}_{k'}$ to π^{-1} , gets $\tilde{\mathbf{u}}_{j+1}, \dots, \tilde{\mathbf{u}}_{k'}$, and outputs $\{(\mathbf{u}_1, \mathbf{v}_1), \dots, (\mathbf{u}_j, \mathbf{v}_j), (\tilde{\mathbf{u}}_{j+1}, \mathbf{v}_{j+1}), \dots, (\tilde{\mathbf{u}}_{k'}, \mathbf{v}_{k'})\}$.

It is not hard to verify that A_{j+1} 's success probability is lower bounded by $\delta^j \cdot \delta = \delta^{j+1}$. Moreover, A_{j+1} makes at most k extra queries compared with A_j . Thus, this contradicts our assumption on R_{j+1}^* . \square

By the above, we know that R_1^* is (k, δ) -evasive, and this in particular means that for all k' , all $\mathbf{u}_1, \dots, \mathbf{u}_{k'}$,

$$|\{(\mathbf{v}_1, \dots, \mathbf{v}_{k'}) : \{(\mathbf{u}_1, \mathbf{v}_1), \dots, (\mathbf{u}_{k'}, \mathbf{v}_{k'})\} \in R\}| \leq \delta \cdot \prod_{i=0}^{k'-1} (N - i), \quad (27)$$

for otherwise, the naïve k -query adversary which non-adaptively queries $\mathbf{u}_1, \dots, \mathbf{u}_k$ would be able to break evasiveness with probability larger than δ , thus contradicting our assumption. Symmetrically, for all $\mathbf{v}_1, \dots, \mathbf{v}_{k'}$,

$$|\{(\mathbf{u}_1, \dots, \mathbf{u}_{k'}) : \{(\mathbf{u}_1, \mathbf{v}_1), \dots, (\mathbf{u}_{k'}, \mathbf{v}_{k'})\} \in R\}| \leq \delta \cdot \prod_{i=0}^{k'-1} (N - i). \quad (28)$$

Now, assume that for some $S \in R$, there is no $S^* \neq \emptyset$ in $R^* = R_0^*$ which is a subset of S . Note that at every outer iteration j , the procedure above can at best replace a set S with a subset of S

<p>Game G: $F \leftarrow \perp, Q_1, Q_2 \leftarrow \perp$ $S \stackrel{\\$}{\leftarrow} A^\mathcal{O}$ for $(x_0 \parallel x_1, x_2 \parallel x_3) \in S$ do if $\neg \text{isValid}^\mathcal{O}(x_0, x_1, x_2, x_3)$ then return false if $S \notin R$ then return false return true</p> <p>PROCEDURE $\text{isValid}^\mathcal{O}(x_0, x_1, x_2, x_3)$: $y_1 \leftarrow \mathcal{O}(0 \parallel x_1); y_2 \leftarrow \mathcal{O}(1 \parallel x_2)$ if $(x_2 = x_0 \oplus y_1) \wedge (x_3 = x_1 \oplus y_2)$ then return true return false</p>	<p>ORACLE $\mathcal{O}(i \parallel x)$: if $F[i \parallel x] = \perp$ then $F[i \parallel x] \stackrel{\\$}{\leftarrow} \{0, 1\}^n$ $Q_i \leftarrow Q_i \cup \{x\}$ return $F[i \parallel x]$</p>
---	--

Fig. 10: The Game G used in the proof of Proposition 2.

of size j . Therefore, in R_1^* , there must be a non-empty subset $S' \subseteq S$ left. However, because R_1^* is (k, δ) -evasive, nothing will be removed in the iteration for $j = 0$, since this would require violating (27) or (28), and thus $R_0^* = R_1^*$. This concludes the proof. \square

B.2 Proof of Proposition 2

Proof. Consider the game G shown in the Fig. 10 where A is given access to an oracle \mathcal{O} that implements a function from $\text{Funcs}(n+1, n)$ using lazy sampling. \mathcal{O} maintains a table F (initially empty) to respond to A 's queries. For query $(i \parallel x)$ such that $F[i \parallel x] = \perp$, \mathcal{O} randomly samples an n -bit string and then assigns it to $F[i \parallel x]$ and responds with $F[i \parallel x]$ to A . For a query $(i \parallel x)$ ¹³ such that $F[i \parallel x] \neq \perp$, it just responds with $F[i \parallel x]$. Then, it is easy to see that,

$$\text{Adv}_{R, \text{Fei}_2^f}^{\text{cri}}(A) = \Pr[\text{G}] .$$

Now, let S be the set output by A such that it contains k' tuples ($0 < k' \leq k$). A wins if $S \in R$ and for every $(x_0 \parallel x_1, x_2 \parallel x_3) \in S$ the isValid procedure returns true. We call such a set as a *winning set*. We can then assume that for every $(x_0 \parallel x_1, x_2 \parallel x_3) \in S$, A issues queries $(0 \parallel x_1)$ and $(1 \parallel x_2)$ at some point in G. Note that this is w.l.o.g because the procedure isValid does make the above queries irrespective of whether A makes them. Note that any pair $(x_0 \parallel x_1, x_2 \parallel x_3)$ (in a winning set) is completely defined by (x_1, x_2) and F as $x_0 = x_2 \oplus F[0 \parallel x_1]$ and $x_3 = x_1 \oplus F[1 \parallel x_2]$. Then, to bound the probability that A wins, it is sufficient to bound the probability that a winning set can be constructed from the queries/responses issued/received by A during its interaction with \mathcal{O} . Finally, let us assume also that A only makes distinct queries to its oracle. Therefore, on every query (after its response) has been sampled, we will compute the probability that a winning set can be constructed.

¹³ $i \in \{0, 1\}$ and $x \in \{0, 1\}^n$.

SOME NOTATION. Let us assume that we are at a point in the execution of G where A has made i_1 queries to of the form $(0 \parallel \cdot)$ and i_2 queries of the form $(1 \parallel \cdot)$, for $i_1 + i_2 = t \leq q - 1$. Let $Q_0 = \{x_1, \dots, x_{i_1}\}$ be the set of $(0 \parallel \cdot)$ queries and $Q_1 = \{x_1, \dots, x_{i_2}\}$ be the set of $(1 \parallel \cdot)$ queries maintained by \mathcal{O} . At this point, A makes another query (i.e., $t + 1$ -th query) to its oracle and let the query be $(i \parallel x^*)$ for $i \in \{0, 1\}$. Once $F[i \parallel x^*]$ is set to a random string, we update $Q_i \leftarrow Q_i \cup \{x^*\}$ as shown in Fig. 10. Then we consider all the sets containing at most k pairs of the form $(y_1 \oplus x_2 \parallel x_1, x_2 \parallel y_2 \oplus x_1)$ where $x_1 \in Q_0$, $x_2 \in Q_1$, $y_1 = F[0 \parallel x_1]$ and $y_2 = F[1 \parallel x_2]$. At this point, we say an event **Bad** happens, if one of the above constructed sets is a winning set. More formally:

Definition 4. *The event **Bad** happens during the execution,*

1. *If immediately after the assignment $F[0 \parallel x] \stackrel{\$}{\leftarrow} \{0, 1\}^n$, a winning set can be constructed from Q_0 and Q_1 .*
2. *If immediately after the assignment $F[1 \parallel x] \stackrel{\$}{\leftarrow} \{0, 1\}^n$, a winning set can be constructed from Q_0 and Q_1 .*

Then, it is easy to see that the following holds,

$$\Pr[\mathsf{G}] \leq \Pr[\mathsf{Bad}] .$$

Let us assume that w.l.o.g the query made by A was $(0 \parallel x^*)$ and also assume that prior to the query $(0 \parallel x^*)$, event **Bad** did not happen. Then, we are interested in bounding the probability of event **Bad** happening when $F[0 \parallel x^*]$ is assigned. For this, we will first bound the number of sets of size at most k that can be constructed using Q_0 and Q_1 and then determine the probability that one of these sets is a winning set.

COMPUTING THE NUMBER OF $k' \leq k$ -SIZED SETS. Since, **Bad** did not happen prior to this query, then it is easy to see that any set we consider (as a candidate winning set) henceforth must have at least one tuple of the form $(\cdot \parallel x^*, \cdot)$. Let us fix some $0 < k' \leq k$ and some $0 \leq j < k'$ and count the number of (candidate winning) sets that can be constructed from Q_0 and Q_1 where exactly $k' - j$ of the pairs (in a set) are of the form $(\cdot \parallel x^*, \cdot)$. Let $C(k', j, t)$ be the number of such sets. It is easy to see that, if x^* is going to be a part of $k' - j$ tuples, there there are $\binom{i_2}{k' - j}$ choices of x_2 's. Furthermore, for the remaining j tuples, we can choose from a set of $i_1 \cdot i_2$ tuples.

$$C(k', j, t) \leq \binom{i_1 i_2}{j} \cdot \binom{i_2}{k' - j} \leq \binom{t^2}{j} \cdot \binom{t}{k' - j} \leq t^{k' + j} .$$

Therefore, number of sets containing at most k' tuples is,

$$C(k', t) = \sum_{j=0}^{k'-1} C(k', j, t) \leq t^{k'} \sum_{j=0}^{k'-1} t^j \leq (1 + t)^{2k'} ,$$

where we use $\sum_{j=1}^{k'-1} t^j \leq (1 + t)^{k'}$. Then, the number of sets of containing at most k tuples is,

$$C(t) = \sum_{k'=1}^k C(k', t) \leq \sum_{k'=1}^k (1 + t)^{2k'} \leq k \cdot (1 + t)^{2k} .$$

PROBABILITY OF THE EXISTENCE OF A WINNING SET. Now, as $R \in \text{FEv}(k, \delta)$, we know that for each of the $C(t)$ sets, there are at most $\delta \cdot 2^n$ values of $F[0 \parallel x^*]$ that will lead to **Bad**. Therefore,

$$\Pr[\text{Bad due to } (t+1)\text{-th query}] \leq 2k\delta \cdot (1+t)^{2k},$$

where the factor 2 is to account for a query of the form $(1 \parallel x^*)$. Therefore, the probability that **Bad** occurs at any point in the execution is upper-bounded by (union bound over the number of queries),

$$\Pr[\text{Bad}] \leq 2k\delta \cdot \sum_{t=0}^{q-1} (1+t)^{2k} \leq 2k\delta \cdot q^{2k+1},$$

which upper-bounds the advantage of A . □

B.3 Proof of Theorem 3

Proof. Let $f \xleftarrow{\$} \text{Funcs}(n+1, n)$ and $s = (s^{\text{in}}, s^{\text{out}})$ be the seed chosen for the $\text{NR}^f[\text{P}]$ construction. Let A be an adversary that makes q queries to its oracle f and is given access to s . Then, for A to break the CI of $\text{NR}^f[\text{P}]$ against R , it needs to break the CI of two-round Feistel (Fei_2^f) against $R_{\pi, \sigma}$ where $\pi = \text{P}(s^{\text{in}}, \cdot)$ and $\sigma = \text{P}(s^{\text{out}}, \cdot)$. Therefore,

$$\text{Adv}_{R, \text{NR}^f[\text{P}]}^{\text{cri}}(A) = \mathbb{E} \left[\text{Adv}_{R_{\pi, \sigma}, \text{Fei}_2^f}^{\text{cri}}(A(s^{\text{in}}, s^{\text{out}})) \right],$$

where the expectation is taken over the choice of (π, σ) . Then for $\delta' > \delta$ (will be set shortly),

$$\begin{aligned} & \mathbb{E} \left[\text{Adv}_{R_{\pi, \sigma}, \text{Fei}_2^f}^{\text{cri}}(A(s^{\text{in}}, s^{\text{out}})) \right] \\ & \leq \Pr[R_{\pi, \sigma} \notin \text{FEv}(k, \delta')] + \Pr[R_{\pi, \sigma} \in \text{FEv}(k, \delta')] \cdot \text{Adv}_{R_{\pi, \sigma}, \text{Fei}_2^f}^{\text{cri}}(A(s^{\text{in}}, s^{\text{out}})) \\ & \leq \Pr[R_{\pi, \sigma} \notin \text{FEv}(k, \delta')] + 2k\delta' \cdot q^{2k+1}. \end{aligned}$$

Above holds due to Proposition 2. Then letting $\delta' = \delta + \frac{1}{2^{n/18}}$ and $t = 10k$ in the bound¹⁴ due to Proposition 3, we get,

$$\text{Adv}_{R, \text{NR}^f[\text{P}]}^{\text{cri}}(A) \leq \frac{24k^2 \cdot (40k\delta^*)^{5k} + 12k^2 \cdot (1+\delta)^{10k}}{2^{4kn/9}} + 2k\delta' \cdot q^{2k+1}.$$

□

B.4 Proof of Lemma 3

Proof. To prove Lemma 3, we start off by using Markov's inequality, which gives us the following,

$$\Pr[|X - \mu| > a] = \Pr[|X - \mu|^t > a^t] \leq \frac{\mathbb{E}[|X - \mu|^t]}{a^t}. \quad (29)$$

Furthermore, when t is even, $\mathbb{E}[|X - \mu|^t] = \mathbb{E}[(X - \mu)^t]$. For the remainder of the proof, we will focus on bounding the t -th moment of $X - \mu$ as a function of the t -th moment of $(Z - \mu)$.

¹⁴ When $t > 4$, $C_t \leq 2$ which holds in our case because $t \geq 10$.

FROM $(\varepsilon, k \cdot t)$ -WISE INDEPENDENT PERMUTATION TO RANDOM PERMUTATION: For every I_{x^i} as defined in Eq. (16), consider another random variable $I_{x^i}^\rho$ where the only difference between their definitions is that π^{-1} is replaced with a random permutation ρ .

$$I_x^\rho = \begin{cases} 1 & \text{if } (\rho(\mathbf{x}_1), \dots, \rho(\mathbf{x}_j), \rho(x \oplus \Delta_{j+1} || x^*), \dots, \rho(x \oplus \Delta_{k'} || x^*)) \in \mathcal{U} \\ 0 & \text{otherwise.} \end{cases} \quad (30)$$

Let $Y = \sum_{i=1}^M I_{x^i}^\rho$. Then,

$$|\mathbb{E} [(X - \mu)^t] - \mathbb{E} [(Y - \mu)^t]| \leq \sum_{i=1}^t \binom{t}{i} \mu^{t-i} |\mathbb{E} [X^i] - \mathbb{E} [Y^i]| \quad (31)$$

Consider the multinomial expansion of $X^i = (I_{x^1} + \dots + I_{x^M})^i$. In the expansion there are M^i monomials and each of the form $\prod_{\beta=1}^M I_{x^\beta}^{t_\beta}$ where $\sum_{\beta=1}^M t_\beta = i$ and $t_\beta \geq 0$. It is easy to see that every such monomial depends on at most $i \leq t$ distinct I_x 's, where each depend on at most $k' \leq k$ outputs of π^{-1} . Therefore, every monomial depends on at most $k \cdot t$ outputs of π^{-1} . Hence, using the $(\varepsilon, k \cdot t)$ -wise independence of π^{-1} one can show that,

$$|\mathbb{E} [X^i] - \mathbb{E} [Y^i]| \leq 2\varepsilon \cdot M^i .$$

Substituting the above in Equation (31) we have,

$$\mathbb{E} [(X - \mu)^t] \leq \mathbb{E} [(Y - \mu)^t] + 2\varepsilon \cdot (M + \mu)^t . \quad (32)$$

FIXING OUTPUTS OF ρ . We now fix the values of ρ on $\mathbf{x}_1, \dots, \mathbf{x}_j$ which are all distinct. Let $\mathbf{u}_i = \rho(\mathbf{x}_i)$ and let $\mathcal{U}' \subseteq \mathcal{U}$,

$$\mathcal{U}' = \{(\mathbf{u}_{j+1}, \dots, \mathbf{u}_{k'}) \mid (\mathbf{u}_1, \dots, \mathbf{u}_{k'}) \in \mathcal{U}\} .$$

Due to the δ -strong evasiveness of R ,

$$|\mathcal{U}'| \leq \delta \prod_{i=0}^{k'-j-1} (D_j - i) , \quad (33)$$

where $D_j = 2^{2n} - j$. For x^i , let us consider another random variable I'_{x^i} ,

$$I'_{x^i} = \begin{cases} 1 & \text{if } (\rho(x^i \oplus \Delta_{j+1} || x^*), \dots, \rho(x^i \oplus \Delta_{k'} || x^*)) \in \mathcal{U}' \\ 0 & \text{otherwise.} \end{cases}$$

Let $Y' = \sum_{i=1}^M I'_{x^i}$, then

$$\mathbb{E} [(Y - \mu)^t] \leq \mathbb{E} [(Y' - \mu)^t] . \quad (34)$$

DOMINATION ARGUMENT. For any x^i , we know that $x^i \oplus \Delta_{j+1}, \dots, x^i \oplus \Delta_{k'}$ are all distinct and hence I'_{x^i} depends on a tuple of exactly $k' - j$ outputs of ρ . Furthermore, we know that each of I'_{x^i} depend on disjoint sets of outputs of ρ . The above two follow from the two properties of x^i 's previously described. Moreover, as $x^* \notin \{\mathbf{x}_1[1], \dots, \mathbf{x}_j[1]\}$, the output of ρ on which I'_{x^i} depend on are not yet fixed. Therefore, the sum Y' depends on exactly $M \cdot (k' - j)$ outputs of ρ (which are yet to be sampled). It is easy to see that $M(k' - j) \leq 2^n$ as we fixed half of the input, that is, x^* . So, $(Y' - \mu)^t$ can be alternatively viewed as some function g which takes a tuple U of $M \cdot (k' - j) \leq 2^n$ strings from $\{0, 1\}^{2n} \setminus \{\mathbf{u}_1, \dots, \mathbf{u}_j\}$, and associates the first $k' - j$ strings of U with I'_{x_1} , second block of $k' - j$ strings with I'_{x_2} and so on, where all strings in U must be distinct. So, $\mathbb{E}[(Y' - \mu)^t]$ can be expressed as the $\mathbb{E}[g]$ with the following distribution p on the domain of g .

$$p(U) = \begin{cases} \prod_{i=0}^{M \cdot (k' - j) - 1} \frac{1}{D_j - i} & \text{if all strings in tuple } U \text{ are distinct,} \\ 0 & \text{otherwise} \end{cases}$$

Then,

$$\mathbb{E}[(Y' - \mu)^t] = \sum_U p(U) \cdot g(U) .$$

Next, for x^i , let us define the random variable $I^*_{x^i}$ (which was promised in the statement of Lemma 3) which is 1 with probability that a tuple of $k' - j$ strings sampled uniformly at random and without replacement from $\{0, 1\}^{2n} \setminus \{\mathbf{u}_1, \dots, \mathbf{u}_j\}$ falls in the set \mathcal{U}' . Let $Z = \sum_{x^i} I^*_{x^i}$. It is easy to see that,

1. $I^*_{x^i}$ are independent and $\mathbb{E}[I^*_{x^i}] \leq \delta$ (due to Eq. 33).
2. $\mu = \mathbb{E}[Z] \leq \delta \cdot M$.

Now, as above, $(Z - \mu)^t$ can also be viewed as a function g which takes a tuple of U of $M \cdot (k' - j)$ strings from $\{0, 1\}^{2n} \setminus \{\mathbf{u}_1, \dots, \mathbf{u}_j\}$, where now U is sampled according to another distribution q ,

$$q(U) = \begin{cases} \left(\prod_{i=0}^{k' - j - 1} \frac{1}{D_j - i} \right)^M & \text{if every block of } (k' - j) \text{ strings are distinct} \\ 0 & \text{otherwise} \end{cases}$$

Then,

$$\mathbb{E}[(Z - \mu)^t] = \sum_U q(U) \cdot g(U) .$$

For any tuple U of $M \cdot (k' - j)$ strings the following holds,

$$p(U) \leq 3 \cdot q(U) . \tag{35}$$

Then Eq. (35) coupled with the fact that g is positive (as t is even) it follows,

$$\mathbb{E}[(Y - \mu)^t] \leq 3 \cdot \mathbb{E}[(Z - \mu)^t] . \tag{36}$$

Firstly, for any U the following is true.

$$q(U) = 0 \implies p(U) = 0 .$$

Therefore, we only focus on U for which $p(U) > 0$. Consider the following,

$$p(U) = \prod_{i=0}^{M \cdot (k' - j) - 1} \frac{1}{D_j - i} = \left(\frac{1}{D_j} \right)^{M \cdot (k' - j)} \prod_{i=1}^{M \cdot (k' - j) - 1} \frac{1}{1 - \frac{i}{D_j}} .$$

We use the fact that $(1 - a)(1 - b) \geq 1 - a - b$ for all $a, b > 0$ and get

$$p(U) \leq \left(\frac{1}{D_j} \right)^{M \cdot (k' - j)} \frac{1}{1 - \frac{(M \cdot (k' - j))(M \cdot (k' - j))}{2 \cdot D_j}} .$$

Since, $M(k' - j) \leq 2^n$ and $k < 2^n$,

$$p(U) \leq 3 \cdot \left(\frac{1}{D_j} \right)^{M \cdot (k' - j)} \leq 3 \cdot \left(\prod_{i=0}^{k' - j - 1} \frac{1}{D_j - i} \right)^M = 3 \cdot q(U) .$$

This establishes Eq. (35). Then combining Eqs. (29),(32),(34),(36) concludes the proof of Lemma 3. \square

B.5 Constructing (ε, p) -wise Independent Permutations

Before giving the construction of (ε, p) -wise independent permutations, let us first review the definition of p -wise independent functions and their construction.

P-WISE INDEPENDENT FUNCTIONS. For $p \geq 1$, we say that a family of functions \mathbf{H} from m to n bits is *p-wise independent* if for all distinct $u_1, \dots, u_p \in \{0, 1\}^m$, the distributions of $\mathbf{H}(s, u_1), \dots, \mathbf{H}(s, u_p)$ (for $s \xleftarrow{\$} \mathbf{H}.\text{Kg}$) and of $f(u_1), \dots, f(u_p)$ (for $f \xleftarrow{\$} \text{Funcs}(m, n)$) are identical.

CONSTRUCTING p -WISE INDEPENDENT FUNCTIONS. A p -wise independent hash function family can be constructed from the polynomial with random coefficients method due to [WC81]. We detail the construction below for completeness.

Let \mathbf{H}_p be a function family from n -bits to $m \leq n$ -bits such that $(s_0, \dots, s_{p-1}) \xleftarrow{\$} \mathbf{H}_p.\text{Kg}$ where $s_i \xleftarrow{\$} \{0, 1\}^n$ and evaluation algorithm on inputs s and x , that is $\mathbf{H}_p(s, x)$, returns the m -bit prefix of the binary representation of $\sum_{i=0}^{p-1} s_i \cdot x^i$ where the arithmetic is in $\text{GF}(2^n)$, the finite field of 2^n elements. Then,

Lemma 5. *For $p \geq 1$, \mathbf{H}_p is a p -wise independent function family.*

(ε, p) -WISE INDEPENDENT PERMUTATIONS FROM FEISTEL. The most natural way of constructing p -wise independent permutations is to use the Feistel construction with sufficient number of rounds where the underlying round functions are instantiated from a p -wise independent function family as constructed above. In particular, we consider the construction $\mathbf{P}[r, \mathbf{H}]$ implementing a permutation on $2n$ -bit strings where $r \in \mathbb{N}$ and \mathbf{H} is a family of functions from n -bits to n -bits. The key generation algorithm $\mathbf{P}[r, \mathbf{H}].\text{Kg}$ outputs (s_1, \dots, s_r) where $s_1, \dots, s_r \xleftarrow{\$} \mathbf{H}.\text{Kg}$ and for inputs the seed $s = (s_1, \dots, s_r)$ and $\mathbf{x} = x_0 \parallel x_1$, $\mathbf{P}(s, \mathbf{x})$ returns $\mathbf{y} = x_r \parallel x_{r+1}$ where,

$$x_{i+1} = x_{i-1} \oplus \mathbf{H}(s_i, x_i) \text{ for all } i = 1, \dots, r . \quad (37)$$

The inverse algorithm $P^{-1}(s, \mathbf{y} = (x_r \parallel x_{r+1}))$ simply computes the values backwards and outputs $\mathbf{x} = x_0 \parallel x_1$.

It follows from the analysis of Theorem 4 in [HR10] that for $r \geq 1$ the family $P[3r, H]$ is an (ε, p) -wise independent permutation family when H is a family of p -wise independent hash functions and¹⁵

$$\varepsilon = \frac{p}{r+1} \left(\frac{4p}{2^n} \right)^r. \quad (38)$$

Therefore to instantiate the permutation P^{-1} from Section 4.3 via $P[3r, H]$ one needs to find a sufficiently large r such that the above error ε becomes less than $1/2^{5kn}$ where $p = 10k^2$. When $k \leq 2^{n/3}$, the permutation family $P[21k, H]$ ($r = 7$) where H is a family of $10k^2$ -wise independent functions can instantiate the family P^{-1} .

B.6 The Case of Unary Relations

In this section, we consider the $NR^f[P]$ construction where $P = P[1, H]$, that is one-round of Feistel where the round function is implemented by a function family H . We direct the reader to Section B.5 for a precise description of the family $P[1, H]$.

We show that $NR^f[P[1, H]]$, where H is a family of $O(1)$ -wise independent functions, is correlation intractable against strongly evasive unary relations. The proof of CI proceeds by showing that $P[1, H]$ transforms any unary δ -strongly evasive relation R into a 2-Feistel evasive relation and hence for the adversary to break the CI of $NR^f[P[1, H]]$ it needs to break the CI of two-round Feistel against a 2-Feistel evasive relation.

FROM STRONGLY EVASIVE TO 2-FEISTEL EVASIVE RELATION. For $\delta \in [0, 1]$, let R be a unary δ -strongly evasive relation over $2n$ -bit strings. For $s^{\text{in}}, s^{\text{out}} \xleftarrow{\$} P[1, H].\text{Kg}$, we let $\pi(\cdot) = P[1, H](s^{\text{in}}, \cdot)$ and $\sigma = \theta \cdot P[1, H](s^{\text{out}}, \cdot)$, where θ is a permutation on $2n$ -bit strings such that $\theta(\mathbf{x} = x_0 \parallel x_1) = x_1 \parallel x_0$. Furthermore, we let $g = H(s^{\text{in}}, \cdot)$ and $h = H(s^{\text{out}}, \cdot)$ be the round functions of π and σ . We define a relation $R_{g,h}$ which is a result of transforming $(\mathbf{u}, \mathbf{v}) \in R$ via π and σ in the following way,

$$R_{g,h} = \{(\pi(\mathbf{u}), \sigma(\mathbf{v})) \mid (\mathbf{u}, \mathbf{v}) \in R\}.$$

Then, for every δ -strongly unary evasive relation R we show that $R_{g,h} \in \text{FEv}(1, \delta')$ for some δ' larger than δ , except with small probability, where the probability is taken over the random choice of (g, h) . This is more formally captured in the following proposition.

Proposition 4 (CI Amplification). *For $\delta \in [0, 1]$, let R be a unary δ -strongly evasive relation over $\{0, 1\}^{2n}$. For an even integer $p \geq 2$, let H be a family of $(0, p)$ -wise independent functions. Then, for $\delta' \in [0, 1]$ such that $\delta' > \delta$,*

$$\Pr_{g,h} [R_{g,h} \notin \text{FEv}(1, \delta')] \leq 2 \cdot C_p \cdot \frac{2^{3n}}{(\delta' - \delta)^p} \left(\frac{2\delta^* p}{2^n} \right)^{p/2},$$

where $C_p = 2e^{1/6p} \sqrt{\pi p} \left(\frac{5}{2e}\right)^{p/2}$, $\delta^* = \max(\delta, p/2^n)$, $g = H(s^{\text{in}}, \cdot)$ and $h = H(s^{\text{out}}, \cdot)$ for $s^{\text{in}}, s^{\text{out}} \xleftarrow{\$} H.\text{Kg}$.

¹⁵ We note that another bound for ε can be derived as a corollary of Theorem 1 in [MPR07] which however is qualitatively weaker than the one due to [HR10].

Proof. We will show that the following holds,

1. For every $\mathbf{x} \in \{0, 1\}^n$ and $y^* \in \{0, 1\}^n$,

$$\Pr[\text{Eq. (11) does not hold for } R_{g,h}] \leq C_p \cdot \frac{1}{(\delta' - \delta)^p} \cdot \left(\frac{2\delta^* p}{2^n}\right)^{p/2}. \quad (39)$$

2. For every $\mathbf{y} \in \{0, 1\}^n$ and $x^* \in \{0, 1\}^n$,

$$\Pr[\text{Eq. (12) does not hold for } R_{g,h}] \leq C_p \cdot \frac{1}{(\delta' - \delta)^p} \cdot \left(\frac{2\delta^* p}{2^n}\right)^{p/2}, \quad (40)$$

where the probability is taken over random choice of (g, h) .

Given that the above hold, we then take appropriate union bounds (for Eq. 39) over all \mathbf{x} and y^* . Symmetrically, we take union bounds (for Eq. 40) over all \mathbf{y} and x^* . Then it is easy to see the following holds and this concludes the proof of Proposition 4,

$$\Pr[R_{g,h} \notin \text{FEv}(1, \delta')] \leq 2 \cdot 2^{3n} \cdot C_p \cdot \frac{1}{(\delta' - \delta)^p} \cdot \left(\frac{2\delta^* p}{2^n}\right)^{p/2}.$$

From now on we will focus on showing that Eq. (40) holds and the analysis for Eq. (39) is symmetrical.

ESTABLISHING EQ. (40). First let us fix some $\mathbf{y} \in \{0, 1\}^{2n}$ and $x^* \in \{0, 1\}^n$. Let us furthermore fix h . This defines $\mathbf{v} = \sigma^{-1}(\mathbf{y})$ which allows us to only focus on tuples in R of the form (\cdot, \mathbf{v}) . Consider the set \mathcal{U} as follows,

$$\mathcal{U} = \{\mathbf{u} \mid (\mathbf{u}, \mathbf{v}) \in R\}, \quad (41)$$

By the δ -strong evasiveness of R , we know that

$$|\mathcal{U}| \leq \delta \cdot 2^{2n}. \quad (42)$$

Now, for every $\mathbf{u} = (\mathbf{u}[0] \parallel \mathbf{u}[1]) \in \mathcal{U}$, we define an indicator random variable $I_{\mathbf{u}}$ as follows,

$$I_{\mathbf{u}} = \begin{cases} 1 & \text{if } g(\mathbf{u}[1]) \oplus \mathbf{u}[0] = x^* \\ 0 & \text{otherwise.} \end{cases} \quad (43)$$

In other words, $I_{\mathbf{u}} = 1$ if \mathbf{u} transforms to a tuple of the form $(\cdot \parallel x^*)$ under π (implemented with g). Then to show that Eq. (40) we would like to show that the number of such $\mathbf{u} \in \mathcal{U}$ are not more than $\delta' \cdot 2^n$. That is,

$$\Pr \left[\sum_{\mathbf{u} \in \mathcal{U}} I_{\mathbf{u}} > \delta' \cdot 2^n \right] \leq C_p \cdot \frac{1}{(\delta' - \delta)^p} \cdot \left(\frac{2\delta^* p}{2^n}\right)^{p/2}. \quad (44)$$

Note that there can be $\mathbf{u}_1 \neq \mathbf{u}_2 \in \mathcal{U}$ that share the same right half, that is $u = \mathbf{u}_1[1] = \mathbf{u}_2[1]$ and hence $I_{\mathbf{u}_1}$ and $I_{\mathbf{u}_2}$ both depend on $g(u)$. However, it is easy to see that $I_{\mathbf{u}_1} + I_{\mathbf{u}_2} \leq 1$ as $\mathbf{u}_1[0] \neq \mathbf{u}_2[0]$. Then, we define a sequence of new indicator random variables I_u which ‘‘corresponds’’ to the sum

of all $I_{\mathbf{u}}$ such that $\mathbf{u}[1] = u$. Furthermore, we will show that I_u s are p -wise independent. Let's do this step precisely. Let us first define \mathcal{S} , the set of all distinct right-halves of $\mathbf{u} \in \mathcal{U}$,

$$\mathcal{S} = \{\mathbf{u}[1] \mid \mathbf{u} \in \mathcal{U}\},$$

Next, for every $u \in \mathcal{S}$, let us define another set \mathcal{S}_u ,

$$\mathcal{S}_u = \{\mathbf{u}[0] \oplus x^* \mid \mathbf{u} \in \mathcal{U} \wedge \mathbf{u}[1] = u\}.$$

Now, let us define the following random variable I_u for $u \in \mathcal{S}$,

$$I_u = \begin{cases} 1 & \text{if } g(u) \in \mathcal{S}_u \\ 0 & \text{otherwise.} \end{cases}$$

Next, it is easy to see that the following holds $\sum_{\mathbf{u} \in \mathcal{U}} I_{\mathbf{u}} = \sum_{u \in \mathcal{S}} I_u$.

Now, let us look at the sum $\sum_{u \in \mathcal{S}} I_u$. Here for every $u \in \mathcal{U}$, $\mathbb{E}[I_u] = |\mathcal{S}_u|/2^n$. Then,

$$\mu = \mathbb{E}\left[\sum_u I_u\right] = \sum_u |\mathcal{S}_u|/2^n \leq |\mathcal{U}|/2^n \leq \delta \cdot 2^n.$$

Furthermore, as g is p -wise independent and each of the I_u s depend on outputs of g on distinct inputs, we establish that I_u s are p -wise independent too. Therefore, for $X = \sum_{u \in \mathcal{S}} I_u$, the sum of p -wise independent random variables with mean μ , we invoke Lemma 2.3 from [BR94] which for any integer $a > 0$ gives us,

$$\Pr[|X - \mu| > a] \leq C_p \cdot \left(\frac{p\mu + p^2}{a^2}\right)^{p/2}.$$

Since $\mu \leq \delta \cdot 2^n$, we can substitute $a = (\delta' - \delta) \cdot 2^n$ and $p, \mu \leq \delta^* \cdot 2^n$ and

$$\Pr[X > \delta' \cdot 2^n] \leq C_p \cdot \frac{1}{(\delta' - \delta)^p} \left(\frac{2\delta^* p}{2^n}\right)^{p/2}.$$

The above then establishes Equation (44) which concludes the proof. \square

Now Proposition 4 can be combined with Proposition 2 to establish the correlation intractability of $\text{NR}^f[\mathbf{P}]$ against strongly evasive unary relations when $\mathbf{P} = \mathbf{P}[1, \mathbf{H}]$ and \mathbf{H} is a family of 10-wise independent functions.

Theorem 4 (CI of NR). *For $\delta \in [0, 1)$, let R be a δ -strongly evasive unary relation. Further, let \mathbf{H} be a 10-wise independent family of functions on n -bits and let $\mathbf{P} = \mathbf{P}[1, \mathbf{H}]$. Then for any (potentially unbounded) adversary A making q queries,*

$$\text{Adv}_{R, \text{NR}^f[\mathbf{P}]}^{\text{cri}}(A) \leq \frac{128 \cdot 10^5 \cdot (\delta^*)^5}{2^{3n/4}} + \delta' q^2, \quad (45)$$

where $\delta' = \delta + 1/2^{n/8}$ and $\delta^* = \max(\delta, 10/2^n)$.

Asymptotically when $n = \omega(\log \lambda)$, $\delta = \text{negl}(\lambda)$ and $q = \text{poly}(\lambda)$, the RHS of Eq. (45) is also negligible which establishes the correlation intractability of $\text{NR}^f[\text{P}[1, \text{H}]]$ against strongly evasive unary relations. Combining this with Lemma 2 then establishes the CI of against all evasive unary relations.

Proof. Let $f \xleftarrow{\$} \text{Funcs}(n+1, n)$ and $s = (s^{\text{in}}, s^{\text{out}})$ be the seed chosen for the permutation P ($s^{\text{in}}, s^{\text{out}}$ are actually the seeds of H). Let A be an adversary that makes q queries to its oracle f and is given access to s . Then, for A to break the CI of $\text{NR}^f[\text{P}]$ against R , it needs to break the CI of two-round Feistel against $R_{g,h}$. where $g = \text{H}(s^{\text{in}}, \cdot)$ and $h = \text{H}(s^{\text{out}}, \cdot)$. Therefore,

$$\text{Adv}_{R, \text{NR}^f[\text{P}]}^{\text{cri}}(A) = \mathbb{E} \left[\text{Adv}_{R_{g,h}, \text{Fei}_2^f}^{\text{cri}}(A) \right],$$

where the expectation is taken over the choice of (g, h) . Then for $\delta' > \delta$ (will be set shortly),

$$\begin{aligned} & \mathbb{E} \left[\text{Adv}_{R_{g,h}, \text{Fei}_2^f}^{\text{cri}}(A) \right] \\ & \leq \Pr[R_{g,h} \notin \text{FEv}(1, \delta')] + \Pr[R_{g,h} \in \text{FEv}(1, \delta')] \cdot \text{Adv}_{R_{g,h}, \text{Fei}_2^f}^{\text{cri}}(A) \\ & \leq \Pr[R_{g,h} \notin \text{FEv}(1, \delta')] + \delta' q^2. \end{aligned}$$

The above holds due to Remark 1. Then letting $\delta' = \delta + \frac{1}{2^{n/8}}$ and $p = 10$ in the bound¹⁶ due to Proposition 4, we get,

$$\text{Adv}_{R_{g,h}, \text{NR}^f[\text{P}]}^{\text{cri}}(A) \leq \frac{128 \cdot 10^5 \cdot (\delta^*)^5}{2^{3n/4}} + \delta' q^2.$$

□

In this section we have shown that $\text{NR}^f[\text{P}[1, \text{H}]]$, which essentially is a four-round Feistel network, is correlation intractable for unary evasive relations even when the adversary is given the entire function table of the first and the fourth rounds. In Appendix B.7 we show that four-round feistel is not correlation intractable against 4-ary evasive relations even in the weaker setting where A only has oracle access to the underlying round functions. We do not know whether four-round feistel is CI for binary or ternary relations even in the above weaker setting and leave it for future work.

B.7 Four-round Feistel is not correlation intractable

The four-round Feistel construction Fei_4^f is a permutation family on $2n$ -bit strings that makes calls to an oracle $f : \{0, 1\}^{n+2} \rightarrow \{0, 1\}^n$. In particular, on input $\mathbf{x} = x_0 \| x_1$, where $x_0, x_1 \in \{0, 1\}^n$, running $\text{Fei}_4^f(\mathbf{x})$ outputs $\mathbf{y} = x_4 \| x_5$, where

$$\begin{aligned} x_2 & \leftarrow x_0 \oplus f(00 \| x_1); x_3 \leftarrow x_1 \oplus f(01 \| x_2) \\ x_4 & \leftarrow x_2 \oplus f(10 \| x_3); x_5 \leftarrow x_3 \oplus f(11 \| x_4). \end{aligned}$$

Symmetrically, upon an inverse query, $\text{Fei}_4^{f^{-1}}(\mathbf{y} = x_4 \| x_5)$ simply computes the values backwards, and outputs $\mathbf{x} = x_0 \| x_1$.

We show that four-round Feistel is not correlation intractable against 4-ary evasive relations in a

¹⁶ When $p > 4$, $C_p \leq 2$ which holds in our case because $p \geq 10$.

model where adversary gets oracle access to f . In particular, consider the following relation R on $2n$ -bit strings which is $(q, 2q^4/2^n)$ - evasive (for $q \leq 2^n$),

$$R = \left\{ (\mathbf{x}_1, \mathbf{y}_1), \dots, (\mathbf{x}_4, \mathbf{y}_4) \mid \mathbf{x}_1[1] \oplus \mathbf{x}_2[1] \oplus \mathbf{x}_3[1] \oplus \mathbf{x}_4[1] = 0^n \right. \\ \left. \wedge \mathbf{y}_1[0] \oplus \mathbf{y}_2[0] \oplus \mathbf{y}_3[0] \oplus \mathbf{y}_4[0] = 0^n \right\} .$$

We now describe an adversary A that given access to f outputs a set S such that $S \in R$ and for all $(\mathbf{x}, \mathbf{y}) \in S$, $\text{Fei}_4^f(\mathbf{x}) = \mathbf{y}$. A chooses some arbitrary $x_2^1 \neq x_2^2$ and $x_3^1 \neq x_3^2$ and makes the corresponding queries to f to compute,

$$\begin{aligned} x_1^1 &\leftarrow f(01 \parallel x_2^1) \oplus x_3^1 ; x_4^1 \leftarrow f(10 \parallel x_3^1) \oplus x_2^1 , \\ x_1^2 &\leftarrow f(01 \parallel x_2^2) \oplus x_3^1 ; x_4^2 \leftarrow f(10 \parallel x_3^2) \oplus x_2^1 , \\ x_1^3 &\leftarrow f(01 \parallel x_2^1) \oplus x_3^2 ; x_4^3 \leftarrow f(10 \parallel x_3^1) \oplus x_2^2 , \\ x_1^4 &\leftarrow f(01 \parallel x_2^2) \oplus x_3^2 ; x_4^4 \leftarrow f(10 \parallel x_3^2) \oplus x_2^2 . \end{aligned}$$

It is easy to see that $x_1^1 \oplus x_1^2 \oplus x_1^3 \oplus x_1^4 = 0$ and $x_4^1 \oplus x_4^2 \oplus x_4^3 \oplus x_4^4 = 0$. Finally, it issues queries of the form $(00 \parallel \cdot)$ and $(11 \parallel \cdot)$,

$$\begin{aligned} x_0^1 &\leftarrow f(00 \parallel x_1^1) \oplus x_2^1 ; x_5^1 \leftarrow f(11 \parallel x_4^1) \oplus x_3^1 , \\ x_0^2 &\leftarrow f(00 \parallel x_1^2) \oplus x_2^1 ; x_5^2 \leftarrow f(11 \parallel x_4^2) \oplus x_3^1 , \\ x_0^3 &\leftarrow f(00 \parallel x_1^3) \oplus x_2^2 ; x_5^3 \leftarrow f(11 \parallel x_4^3) \oplus x_3^2 , \\ x_0^4 &\leftarrow f(00 \parallel x_1^4) \oplus x_2^2 ; x_5^4 \leftarrow f(11 \parallel x_4^4) \oplus x_3^2 . \end{aligned}$$

Then A outputs the following set S and achieves $\text{Adv}_{R, \text{Fei}_4^f}^{\text{cri}}(A) = 1$.

$$S = \left\{ (x_0^1 \parallel x_1^1, x_4^1 \parallel x_5^1), \quad (x_0^2 \parallel x_1^2, x_4^2 \parallel x_5^2), \quad (x_0^3 \parallel x_1^3, x_4^3 \parallel x_5^3), \quad (x_0^4 \parallel x_1^4, x_4^4 \parallel x_5^4) \right\} .$$