Digital Signatures with Minimal Overhead

Eike Kiltz*[1], Krzysztof Pietrzak†[2], and Mario Szegedy[3]

[1]Horst-Görtz Institute for IT Security, Ruhr-Universität Bochum, Germany,
eike.kiltz@rub.de
[2]Institute of Science and Technology, Austria, pietrzak@ist.ac.at
[3]Rutgers University, USA, szegedy@dragon.rutgers.edu

Abstract

In a digital signature scheme with message recovery, rather than transmitting the message m and its signature σ, a single enhanced signature τ is transmitted. The verifier is able to recover m from τ and at the same time verify its authenticity. The two most important parameters of such a scheme are its security and the overhead $|\tau| - |m|$. A simple argument shows that for any scheme with "n bits security" $|\tau| - |m| \geq n$, i.e., the overhead is at least the security. The best previous constructions required an overhead of $2n$. In this paper we show that the n bit lower bound can basically be matched. Concretely, we propose a new simple RSA-based digital signature scheme that, for $n = 80$ bits security in the random oracle model, has an overhead of ≈ 90 bits.

At the core of our security analysis is an almost tight upper bound for the expected number of edges of the densest "small" subgraph of a random Cayley graph, which may be of independent interest.

Keywords: digital signatures, Feistel, combinatorics, Cayley graph.

1 Introduction

When transmitting a message m over an unauthenticated public channel, one usually appends a string σ to the message that can be used to verify (relative to a public key) the authenticity of the message. This string σ is called the digital signature of m. More generally, one transforms the message m into an *enhanced signature* τ such that (i) the original message m can be recovered from τ; (ii) the authenticity of m can be verified. This is called a digital signature scheme with message recovery and is used to save on bandwidth, i.e., to minimize the *signature overhead* informally defined as $oh = |\tau| - |m|$ (signature length minus message length). A natural question to ask is what is the minimal signature overhead for a fixed level of security?

Lower Bounds on the Overhead. Following [2], we say that a signature scheme has "n-bit security" if for all adversaries A attacking the scheme have success ration $\mathsf{SR}(\mathsf{A})$ at most 2^{-n}, where $\mathsf{SR}(\mathsf{A}) := \mathsf{success}(\mathsf{A})/\mathsf{time}(\mathsf{A})$. A natural lower bound for the overhead of a signature scheme (with or without message recovery) for n-bit security is $oh \geq n$ bits. This is since for a signature

*Funded by a Sofja Kovalevskaja Award of the Alexander von Humboldt Foundation and the German Federal Ministry for Education and Research.

†Supported by the European Research Council under the European Unions Seventh Framework Programme (FP7/2007-2013) / ERC Starting Grant (259668-PSPC)

Scheme	Underlying padding	Overhead for n bits security asymptotic	$n = 80$	$q_h + q_s \leq 2^{60}$	Security reduction
PSS-MR	2-round Feistel	$2n + 2(\log q_h)$	320	280	Bellare-Rogaway [4]
PSS-MR	2-round Feistel	$2n + \log(q_h) + \log(q_s)$	320	240	Coron [7]
PSS-MR	2-round Feistel	$n + \log(q_h)$	160	140	Kakvi-Kiltz [9]
SIG-MR$_{4F}$	4-round Feistel	$n + o(\log(q_h + q_s))$	93	88	this work

Table 1: Overhead of signature schemes with message recovery for n bits security. The table shows the overhead required for $n = 80$ (and only the trivial upper bound $q_h + q_s \leq 2^{80}$) and when we additionally assume that the number of random-oracle and signature queries is upper bounded by $q_h + q_s \leq 2^{60}$, as proposed in [4].

scheme with oh bits of overhead any random bit string τ constitutes a valid enhanced signature with probability 2^{-oh}. Hence an adversary A guessing a random authenticated message has success ratio $\mathsf{SR}(\mathsf{A}) = 2^{-oh}$.

Upper Bounds on the Overhead for existing Schemes. In standard digital signature schemes (without message recover) such as RSA full domain hash [4], the probabilistic signature scheme PSS [4], or BLS signatures [5] the overhead equals the size of a signature. Since a signature contains (at least one) group element (e.g., $\mathbb{Z}_N^*$ or an elliptic curve group) whose representation requires at least $2n$ bits (for n bits security, due to generic attacks) we cannot hope to obtain an overhead any smaller than $2n$ bits.

Computing the overhead turns out to be a bit subtle and depends on the security reduction. We exemplify such a calculation for the RSA-based probabilistic signature scheme with message recovery PSS-MR$[n_0, n_1]$ [4]. PSS-MR$[n_0, n_1]$ has an overhead of $n_0 + n_1$ bits, where parameter n_0 controls the randomness and n_1 the amount of added redundancy used during signing. The minimal size of n_0 and n_1 providing a given security level can be computed from the security reduction. The security reduction from [4] transforms an adversary against PSS-MR$[n_0, n_1]$ making q_s (online) signing and q_h (offline) hash (random oracle [3]) queries with success probability $\varepsilon_{\mathsf{PSS\text{-}MR}}$ into an adversary against RSA with success probability $\varepsilon_{\mathsf{RSA}}$ such that $\varepsilon_{\mathsf{PSS\text{-}MR}} = \varepsilon_{\mathsf{RSA}} + \varepsilon_{sim}$, where $\varepsilon_{sim} = (q_s + q_h)^2(2^{-n_0} + 2^{-n_1})$. An easy computation shows that this implies $oh_{\mathsf{PSS\text{-}MR}} = n_0 + n_1 \geq 2n + \log_2(q_h) + \log_2(q_s)$ bits of overhead for n bits security.[1] An improved security reduction by Coron gives $oh_{\mathsf{PSS\text{-}MR}} \geq 2n + \log_2(q_h) + \log_2(q_s)$. Recently, an alternative security reduction for PSS-MR was proposed in [9] demonstrating a tight security reduction for PSS-MR$[n_0 = 0, n_1]$ with zero-padding from the (stronger) phi-hiding assumption. However, the required overhead is still $oh_{\mathsf{PSS\text{-}MR}} = n + \log_2(q_h)$ bits, stemming from an additive term $\varepsilon_{sim} = q_h^2/2^{n_1}$ in the security reduction. Table 1 summarizes the signature overhead for PSS-MR and gives concrete parameters for a typical security parameter of $n = 80$ bits.

1.1 Our contribution

Our main contribution is to revisit the question if there exists a digital signature scheme with message recovery that has minimal ($\approx n$ bits) overhead. Concretely, we propose SIG-MR$_{4F}$, a scheme based on a four-round Feistel padding. When instantiated with the RSA trapdoor permutation and assuming the phi-hiding assumption, then SIG-MR$_{4F}$ has almost minimal overhead.

[1] For n-bit security of PSS-MR$[n_0, n_1]$ we require $\mathsf{SR}(\mathsf{A}) \leq 2^{-n+1}$ which is implied by $\varepsilon_{\mathsf{RSA}}/\mathsf{time}(\mathsf{A}) \leq 2^{-n}$ and $\varepsilon_{sim}/\mathsf{time}(\mathsf{A}) \leq 2^{-n}$. With $\mathsf{time}(\mathsf{A}) \geq q_s + q_h$ we obtain $n_0 \geq n + \log_2(q_h)$ and $n_1 \geq n + \log_2(q_h)$ and consequently the overhead is $oh = n_0 + n_1 \geq 2n + 2\log_2(q_h)$.

Padding-based Signature schemes with message recovery. To prove our main result, we first introduce an abstract framework to analyze a class of signature schemes with message recovery based on a padding scheme PAD. A padding scheme $\mathsf{PAD} = (\pi, \pi^{-1})$ consists of an evaluation function π that pads a message m into a larger string $\pi(m)$, and an inversion function π^{-1} that either recovers the message m or returns $\perp$ if its input is not a correctly padded message. (In our setting, both π and π^{-1} have access to a random oracle $\mathcal{H}$.) Given a padding scheme and a trapdoor permutation $\mathsf{TDP} = (\mathsf{f}, \mathsf{f}^{-1})$ with matching domains, we can define a signature scheme with message recovery $\mathsf{SIG\text{-}MR}[\mathsf{PAD}, \mathsf{TDP}]$ as follows. The enhanced signature τ on a message m is defined as $\tau = \mathsf{f}^{-1}(\pi(m))$, where f^{-1} is the secret inversion algorithm of TDP. (Hence, signing uses the trapdoor td.) Signature recovery first evaluates the trapdoor permutation on τ and checks if the result is a correctly padded message or not, i.e., $\{m, \perp\} = \pi^{-1}(\mathsf{f}(\tau))$. If the result is not $\perp$, it returns message m.

To argue about the security of the resulting signature scheme $\mathsf{SIG\text{-}MR}[\mathsf{PAD}, \mathsf{TDP}]$ (in the sense of unforgeability against chosen-message attacks) we introduce the notion of ε_{sim}-*simulatability* for a padding scheme PAD. Essentially, PAD is simulatable if a properly distributed signature on *any message* can be computed if one is in control of the random oracle $\mathcal{H}$ but not of the trapdoor of the trapdoor permutation. The quality parameter ε_{sim} is used to measure the quality of the simulation of the signatures and the random oracles.

Generalizing [9], we show that each ε_{sim}-simulatable padding scheme yields a secure signature scheme with message recovery if the underlying trapdoor permutation is lossy [13]. The security reduction essentially only loses the additive factor ε_{sim} from the simulation quality of PAD (and is tight otherwise). Consequently, ε_{sim} will play a crucial role in determining the overhead of a given signature scheme.

Simulation Quality of padding schemes. As a simple warmup example we consider the simulation quality of a padding scheme $\mathsf{PAD}_{2f}[n_1]$ derived from a two-round Feistel network, where the parameter n_1 controls the redundancy of the padding (and hence the overhead of the signature scheme). Such a padding scheme was already implicitly used in PSS and $\mathsf{PSS\text{-}MR}$ [4]. We show that $\mathsf{PAD}_{2f}[n_1]$ has simulation quality $\varepsilon_{sim} = q_h^2/2^{n_1}$. As explained before, the resulting signature scheme $\mathsf{SIG\text{-}MR}_{2f}[n_1] = \mathsf{PSS\text{-}MR}[n_0 = 0, n_1]$ has an overhead of $n_1 = n + \log_2(q_h)$ bits. This reproves the overhead of $\mathsf{PSS\text{-}MR}$ obtained in [9].

The PAD_{4F} padding scheme. For an optimal overhead we would need a padding scheme satisfying $\varepsilon_{sim} \approx q_h/2^{n_1}$. We consider the simulation quality ε_{sim} of a new padding scheme $\mathsf{PAD}_{4F}[n_1]$ derived from a four-round Feistel network, where parameter n_1 controls the redundancy of the padding. The main result of this paper states that (for sufficiently large domain) $\mathsf{PAD}_{4F}[n_1]$ is ε_{sim}-simulatable with

$$\varepsilon_{sim} \leq q_h^{1+o(1)}/2^{n_1}. \tag{1}$$

Hence the resulting signature scheme $\mathsf{SIG\text{-}MR}_{4F}$ has an overhead of $oh_{\mathsf{SIG\text{-}MR}_{4F}} = n + o(\log(q_h))$ bits, cf. Table 1. The $o(1)$ term can be computed explicitly and leads to 93 bits overhead for $n = 80$ bits security if the domain of the TDP is at least 1024 bits. More concretely, the $o(1)$ term goes to 0 as the ratio of the security we want to achieve, divided by the domain size of the TDP, decreases.

In the proof of (1) a variable $Q(\mu, q_h)$ will play a central role, which can be cast in graph theoretical terms as follows. $Q(\mu, q_h)$ takes as value the number of edges of the densest $q_h \times q_h$ subgraph of a random bipartite $\mathbb{Z}_\mu \times \mathbb{Z}_\mu$ Cayley graph with degree q_h, i.e., $Q(\mu, q_h) = \max_{\mathcal{X},\mathcal{Z}} |\{(x, z) \mid x \in \mathcal{X}, z \in \mathcal{Z}, z - x \in \mathcal{B}\}|$, where $\mathcal{B}, \mathcal{X}, \mathcal{Z}$ are q_h element subsets of $\mathbb{Z}_\mu$, and $\mathcal{B}$ is sampled uniformly at random.

Concretely, we show (Theorem 4.4) that $\mathsf{PAD}_{4F}[n_1]$ is $\varepsilon_{sim} = O(\mathsf{E}[Q(\mu, q_h)]/2^{n_1})$ simulatable. Next, we prove by a compressing argument our main technical result (Theorem 6.1) which can be

stated as

$$\text{for each } 0 < a < 1/2 : Q(\mu, \mu^a) \leq \mu^{a+a^2/(1-2a)} \quad \text{(with probability extremely close to 1).} \tag{2}$$

We believe that this result may be of independent interest. It complements a result of Alon et al. (Theorem 4 in [1]) which states that $Q(\mu, \mu^a) \approx \mu^{3a-1}$ for $2/3 < a \leq 1$, i.e. their bound applies to large subgraphs of size $\geq \mu^{2/3}$. Bound (2) together with the above reduction (setting $q_h = \mu^a$) implies bound (1) on the simulatability of $\mathsf{PAD}_{4F}[n_1]$.

2 Preliminaries

For $n \in \mathbb{N}$, we write 1^n for the string of n ones, and $[n]$ for $\{1, \ldots, n\}$. Moreover, $|x|$ denotes the length of a bitstring x, while $|S|$ denotes the size of a set S. Further, $s \leftarrow S$ denotes sampling an element from s uniformly at random from the set S. For an algorithm A, we write $z \leftarrow \mathsf{A}(x, y, \ldots)$ to indicate that A is a (probabilistic) algorithm that outputs z on input $(x, y, \ldots)$.

2.1 Digital signatures with message recovery

A digital signature scheme with message recovery $\mathsf{SIG\text{-}MR} = (\mathsf{G}_{\mathsf{SIG\text{-}MR}}, \mathsf{Sign}, \mathsf{Recover})$ consists of three algorithms and two families $\mathcal{M}(n), \mathcal{S}(n)$ of message and signature spaces. Key generation $\mathsf{G}_{\mathsf{SIG\text{-}MR}}$ generates a keypair $(pk, sk) \leftarrow \mathsf{G}(1^k)$ for a secret signing key sk and a public verification key pk. The signing algorithm Sign inputs a message $m \in \mathcal{M}(n)$ and the secret signing key, and returns a signature $\tau \leftarrow \mathsf{Sign}_{sk}(m) \in \mathcal{S}(n)$ of the message. The recovery algorithm $\mathsf{Recover}$ that takes a verification key pk and an enhanced signature τ as input, and returns $m \leftarrow \mathsf{Vrfy}_{pk}(m, \tau)$ where $m \in \mathcal{M} \cup \{\perp\}$. We require that $\Pr[\mathsf{Recover}_{pk}(\mathsf{Sign}_{sk}(m)) = m] = 1$.

Random Oracle Model. When the signature scheme contains a hash function $\mathcal{H} : \{0,1\}^* \to \{0,1\}^n$, it can be analyzed in the random oracle model [3]. In this model the hash function is treated as an idealized function whose outputs are independent random values and can only be access through oracle calls. (To instantiate such a scheme in the real world, one would instantiate the random oracle $\mathcal{H}$ with a hash function like SHA.)

Security. Let us recall the *existential unforgeability against chosen message attacks* (EUF-CMA) security experiment [8] in the random oracle model, played between a challenger and a forger F.

1. The challenger runs $\mathsf{G}_{\mathsf{SIG\text{-}MR}}(1^n)$ to generate a keypair (pk, sk). Forger $\mathcal{F}$ receives pk as input.
2. Forger $\mathcal{F}$ may ask the challenger to sign a number of messages and evaluate a number of hash queries. To query the i-th signature, F submits a message $m_i \in \mathcal{M}(n)$ to the challenger. The challenger returns an enhanced signature τ_i under sk for this message. For the j-th hash query, F submits a query x_i to the challenger who returns the values $\mathcal{H}(x_i)$.
3. Forger $\mathcal{F}$ outputs an enhanced signature τ^*.

Let $m^* \leftarrow \mathsf{Recover}(pk, \tau^*)$ be the recovered message of F's forgery. Forger F wins the game if $m^* \neq \perp$ (that is, τ^* is a valid enhanced signature) and $m^* \neq m_i$ for all i.

Definition 2.1 (Security and Overhead of $\mathsf{SIG\text{-}MR}$) *Let $\mathsf{SIG\text{-}MR}$ be a signature scheme with message recovery, where $\mathcal{M}(n)$ is the message and $\mathcal{S}(n)$ is the signature spaces. Let $t_{sig}, q_s, q_h, \varepsilon_{sig}$ be functions of a security parameter n.*

Security: SIG-MR *is* $(t_{sig}, q_s, q_h, \varepsilon_{sig})$*-secure, if all adversaries* F *running in time at most* t_{sig} *making at most* q_s *signing queries and* q_h *hash queries, have success probability at most* ε_{sig}*.*

n**-bit security:** *We say* SIG-MR *has* n *bits of security against* q_s, q_h *queries if it is* $(t_{sig}, q_s, q_h, \varepsilon_{sig})$*-secure for all* $t_{sig}, \varepsilon_{sig}$ *satisfying* $\varepsilon_{sig}/t_{sig} \leq 2^{-n}$*. We simply say it has* n *bits security if it has* n *bits security for any* q_s, q_h *(we can always assume the trivial upper bound* $q_s + q_h \leq t_{sig} \leq 2^n$*.*[2]*)*

Overhead: *The overhead of* SIG-MR *is* $\log_2 |\mathcal{S}(n)| - \log_2 |\mathcal{M}(n)|$*.*

Minimal overhead: *With* $oh_{\mathsf{SIG\text{-}MR}}(n, q)$ *denote the overhead required in the construction* SIG-MR *to reach* n *bits security against* $q_s + q_h \leq q$ *queries.* $oh_{\mathsf{SIG\text{-}MR}}(n)$ *denotes* $oh_{\mathsf{SIG\text{-}MR}}(n, 2^n)$*.*

2.2 Trapdoor Permutations

A trapdoor permutation $\mathsf{TDP} = (\mathsf{G_{TDP}}, \mathsf{f}, \mathsf{f}^{-1})$ over domain $\mathcal{D}(n)$ consists of three ppt algorithms. Key generation $\mathsf{G_{TDP}}$ generates a keypair $(ek, td) \leftarrow \mathsf{G_{TDP}}(1^n)$ of evaluation key and trapdoor. For every (ek, td) in the domain of $\mathsf{G_{TDP}}(1^n)$, $\mathsf{f}(ek, .)$ and $\mathsf{f}^{-1}(td, .)$ compute permutations $\mathsf{f}_{ek}(\cdot), \mathsf{f}_{td}^{-1}(\cdot)$ on $\mathcal{D}(n)$ s.t. for all $x \in \mathcal{D}(n)$: $\mathsf{f}_{td}^{-1}(\mathsf{f}_{ek}(x)) = x$. We say TDP is homomorphic if $(\mathcal{D}(n), \circ)$ is a group and for all $x_1, x_2 \in \mathcal{D}(n)$, $\mathsf{f}_{ek}(x_1) \circ \mathsf{f}_{ek}(x_2) = \mathsf{f}_{ek}(x_1 \circ x_2)$.

We now recall the security properties of one-wayness and regular lossiness [9, 13].

Definition 2.2 (Security of TDP**)** *Let* $t = t(n)$ *and* $\varepsilon = \varepsilon(n)$ *be functions of a security parameter* n*.* TDP *is* (ε, t)*-one-way if for all adversaries* A *running in time at most* t*,* $\Pr[\mathsf{A}(ek, \mathsf{f}_{ek}(x)) = x] \leq \varepsilon$*, where* $(ek, td) \leftarrow \mathsf{G_{TDP}}(1^n)$*,* $x \leftarrow \mathcal{D}(n)$*.*

Definition 2.3 (Lossy TDP**)** *Let* $t = t(n)$*,* $\ell = \ell(n)$ *and* $\varepsilon = \varepsilon(n)$ *be functions of a security parameter* n*. A trapdoor permutation* TDP *over domain* $\mathcal{D}(n)$ *is regular* (ε, t, ℓ)*-lossy if there exists a ppt algorithm* G_{lossy} *(the lossy key generator) that on input* 1^n *outputs* (ek, td) *such that*

1. *(indistinguishability of real and lossy keys) for all adversaries* A *running in time at most* t*,* $\Pr[\mathsf{A}(ek) = 1] - \Pr[\mathsf{A}(ek') = 1] \leq \varepsilon$*, where* $(ek, td) \leftarrow \mathsf{G_{TDP}}(1^n)$ *and* $ek' \leftarrow \mathsf{G}_{lossy}(1^n)$*;*
2. *(lossiness)* $\mathsf{f}_{ek'}(\cdot)$ *is* ℓ*-to-*1*, i.e.* $\forall x \in \mathcal{D}(n) : |\{z \ : \ f_{ek'}(z) = f_{ek'}(x)\}| = \ell$*.*

It is well known [13] that a lossy trapdoor permutation is collision-resistant when instantiated in lossy mode. The most important example of a trapdoor permutation is RSA with $\mathcal{D}(n) = \mathbb{Z}_N^*$, defined as $\mathsf{f}_{N,e}(x) = x^e \bmod N$. It is homomorphic with respect to modular multiplication. It is one-way under the RSA assumption; for all $e < N^{1/4}$ it is furthermore regular e-lossy under the phi-hiding assumption [9], where e is the public RSA exponent. Another example of a (homomorphic and regular lossy) trapdoor function is Paillier [12].

3 Padding-Based Signatures

3.1 Padding Schemes

A padding scheme PAD consists of three ppt algorithms $\mathsf{PAD} = (\mathsf{G_{PAD}}, \pi, \pi^{-1})$ and families $\mathcal{M}(n), \mathcal{R}(n)$ of message and range space with $|\mathcal{R}(n)| \geq |\mathcal{M}(n)|$. $\mathsf{G_{PAD}}(1^n)$ is a probabilistic algorithm that outputs a key k. For each k, π and π^{-1} implement functions

$$\pi_k : \mathcal{M}(n) \to \mathcal{R}(n), \qquad \pi_k^{-1} : \mathcal{R}(n) \to \mathcal{M}(n) \cup \{\perp\}.$$

[2] As $\varepsilon \leq 1$, $\varepsilon_{sig}/t_{sig} \leq 2^{-n}$ for every $t_{sig} \geq 2^n$, so we only have to look at the case $t_{sig} \leq 2^n$.

That is, π pads a message $m \in \mathcal{M}(n)$ into a larger space $\mathcal{R}(n)$. The redundancy of PAD is $\log_2 |\mathcal{R}(n)| - \log_2 |\mathcal{M}(n)|$.

For (perfect) correctness we require that for all $n \in \mathbb{N}$, $k \in \mathsf{G}_{\mathsf{PAD}}(1^n)$ and $m \in \mathcal{M}(n)$, $\pi_k^{-1}(\pi_k(m)) = m$. Hence π must be injective. For $k \in \mathsf{G}_{\mathsf{PAD}}(1^n)$ we define $I_k := \{\pi_k(m) \mid m \in \mathcal{M}(n)\} \subseteq \mathcal{R}(n)$ as the image of $\mathcal{M}$ under π_k. For (perfect) soundness we require that for all $y \in \{0,1\}^n \setminus I_k$, $\pi_k^{-1}(y) = \perp$. We will analyze the security of padding schemes PAD in the random oracle model in which case we assume for simplicity that there is no key-generation algorithm. Instead of getting access to a key k, the algorithms π, π^{-1} (and the adversary) have access to a random oracle $\mathcal{H} : \{0,1\}^* \to \{0,1\}^n$.

Simulability. We will now define what it means for a padding scheme $\mathsf{PAD} = (\pi, \pi^{-1})$ to be "simulatable" in the random oracle model. (This notion is closely related to indifferentiability as we will discuss below.)

To a padding scheme $\mathsf{PAD} = (\pi, \pi^{-1})$ defined in the random oracle model with hash function $\mathcal{H}$, a simulator S (which we specify in more detail below), and an adversary A we associate the following advantage function

$$\mathbf{Adv}^{\text{sim}}_{\mathsf{A},\mathsf{PAD},\mathsf{S}}(n) = |\Pr[\mathsf{A}^{\pi^{\mathcal{H}},\mathcal{H}}(1^n) = 1] - \Pr[\mathsf{A}^{\mathcal{F},\mathsf{S}^{\mathcal{F}}}(1^n) = 1]|,$$

where $\mathcal{F} : \mathcal{M}(n) \to \mathcal{R}(n)$ is a random function, $\mathcal{H} : \{0,1\}^* \to \mathcal{R}(n)$ is a random oracle and the (stateful, probabilistic) simulator $\mathsf{S}^{\mathcal{F}}$ gets to see all queries made to oracle $\mathcal{F}$.

Definition 3.1 (Simulability of a Padding Scheme) *We say* PAD *is* $(q_{\mathcal{F}}, q_{\mathcal{H}}, q_{\mathsf{S}}, t_{sim}, \varepsilon_{sim})$ *simulatable if there exists a simulator* S *making* q_{S} *oracle queries to* $\mathcal{F}$ *and running in time* t_{sim} *such that* $\mathbf{Adv}^{\text{sim}}_{\mathsf{A},\mathsf{PAD},\mathsf{S}} \leq \varepsilon_{sim}$*, for all* A *making* $q_{\mathcal{F}}$ *and* $q_{\mathcal{H}}$ *queries to the oracles* $\mathcal{F}$ *and* $\mathcal{H}$*, respectively.*

In informal discussions we will just say (q, ε) to denote $(q, q, \Theta(q), q \cdot \mathit{polylog}(q), \varepsilon)$ simulatable.

3.2 Padding-based Signatures with message recovery

Let TDP be a trapdoor permutation over domain $\mathcal{D}(n)$ and PAD be a padding scheme with message space $\mathcal{M}(n)$ and range $\mathcal{R}(n) = \mathcal{D}(n)$. We build a signature scheme with message recovery $\mathsf{SIG\text{-}MR}_{\mathsf{TDP},\mathsf{PAD}} = (\mathsf{G}_{\mathsf{SIG\text{-}MR}}, \mathsf{Sign}, \mathsf{Recover})$ with message space $\mathcal{M}(n)$ and signature space $\mathcal{S}(n) = \mathcal{D}(n)$. (To obtain a scheme with arbitrary message space, one can apply the domain extension given in Section 3.4.) $\mathsf{G}_{\mathsf{SIG\text{-}MR}}(1^n)$ runs $k \leftarrow \mathsf{G}_{\mathsf{PAD}}(1^n)$ and $(ek, td) \leftarrow \mathsf{G}_{\mathsf{TDP}}(1^n)$. It returns $pk = (k, ek)$ and $sk = td$.

Algorithm $\mathsf{Sign}_{sk}(m \in \mathcal{M}(n))$	Algorithm $\mathsf{Recover}_{pk}(\tau \in \mathcal{D}(n))$
$y := \pi_k(m) \in \mathcal{D}(n)$	$y = \mathsf{f}_{ek}(\tau)$
Return $\tau = \mathsf{f}_{td}^{-1}(y) \in \mathcal{D}(n)$	If $\pi_k^{-1}(y) = \perp$ then return $\perp$
	Else return $m = \pi_k^{-1}(y)$

Note that (perfect) correctness of SIG follows by (perfect) correctness of PAD and since TDP is a permutation. The following theorem proves security provided TDP is regular lossy. Its proof is similar to the one of $\mathsf{RSA\text{-}FDH}$ in [9].

Theorem 3.2 *Suppose* TDP *is regular* $(\ell, t_{lossy}, \varepsilon_{lossy})$*-lossy (i.e., lossy by* $\log_2(\ell)$ *bits) and* PAD *is a perfectly sound and* $(q_{\mathcal{F}}, q_{\mathcal{H}}, q_{\mathsf{S}}, t_{sim}, \varepsilon_{sim})$*-simulatable padding scheme. Then* SIG *is* $(t_{sig}, q_h, q_s, \varepsilon_{sig})$ *secure in the random oracle model with*

$$q_h = q_{\mathcal{H}}, \quad q_s = q_{\mathcal{F}}, \quad t_{sig} = t_{lossy} - t_{sim}, \quad \varepsilon_{sig} = \varepsilon_{sim} + (2\ell - 1)/\ell \cdot \varepsilon_{lossy}.$$

Proof. Let F be an adversary against the signature scheme that runs in time t_{sig}, makes at most q_s queries to the signing oracle, q_h queries to the random oracle, and outputs a forgery with probability ε_{sig}.

Game G_0. This is the UF-CMA game and hence $\Pr[\mathsf{G}_0 = 1] = \varepsilon_{sig}$.

Game G_1. Consider the function $\mathcal{F} : \mathcal{M}(n) \to \mathcal{D}(n)$ as defined below. Since f_{ek} defines a permutation over $\mathcal{D}(n)$, $\mathcal{F}$ defines a perfectly random function. In Game G_1, replace random oracle $\mathcal{H}$ with $\mathsf{S}^{\mathcal{F}}$ and use Sign_1 to sign messages.

Algorithm $\mathcal{F}(m)$	Algorithm $\mathsf{Sign}_1(m \in \mathcal{M}(n))$
If $\mathcal{F}(m) \neq \perp$ then return $\mathcal{F}(m)$	If $\mathcal{F}(m) = \perp$ then call $\mathcal{F}(m)$
Else $\tau(m) \in_R \mathcal{D}(n)$	Return $\tau(m)$
Return $\mathcal{F}(m) := \mathsf{f}_{ek}(\tau(m))$	

We claim that

$$\Pr[\mathsf{G}_0 = 1] - \Pr[\mathsf{G}_1 = 1] \leq \varepsilon_{sim}.$$

To prove the claim we define an adversary $\mathsf{A}^{\mathcal{O}_1,\mathcal{O}_2}$ as follows. It runs $\mathsf{G}_{\mathsf{TDP}}$ to obtain the ek and td. Next, it runs F on $pk = ek$, answering its signing queries on a message m by $\tau_m = \mathsf{f}_{td}^{-1}(\mathcal{O}_1(m))$ and its hash queries using oracle $\mathcal{O}_2$. Finally, it outputs 1 iff the forgery output by F is valid. A runs in time $t + t_{sim}$, makes $q_{\mathcal{F}} = q_s$ queries to oracle $\mathcal{F}$ and $q_{\mathcal{H}} = q_h$ queries to oracle $\mathcal{H}$. Clearly, $\Pr[\mathsf{G}_0 = 1] = \Pr[\mathsf{A}^{\pi^{\mathcal{H}},\mathcal{H}}(1^n) = 1]$. Furthermore, if $(\mathcal{O}_1, \mathcal{O}_2) = (\mathcal{F}, \mathsf{S}^{\mathcal{F}})$, then the simulated signatures are of the form $\tau_m = \mathsf{f}_{td}^{-1}(\mathcal{F}(m)) = \mathsf{f}_{td}^{-1}(\mathsf{f}_{ek}(\tau(m))) = \tau(m)$ and hence $\Pr[\mathsf{G}_1 = 1] = \Pr[\mathsf{A}^{\mathcal{F},\mathsf{S}^{\mathcal{F}}}(1^n) = 1]$. Overall, $\Pr[\mathsf{G}_0 = 1] - \Pr[\mathsf{G}_1 = 1] = \Pr[\mathsf{A}^{\pi^{\mathcal{H}},\mathcal{H}}(1^n) = 1] - \Pr[\mathsf{A}^{\mathcal{F},\mathsf{S}^{\mathcal{F}}}(1^n) = 1] = \mathbf{Adv}^{\mathrm{sim}}_{\mathsf{A},\mathsf{PAD},\mathsf{S}}(n) \leq \varepsilon_{sim}$.

Game G_2. Switch ek of TDP to lossy. More formally, game G_2 is like G_1 with the difference that ek from pk is now generated using the lossy trapdoor generation algorithm $\mathsf{G}_{lossy}(1^n)$. Clearly, since from G_1 on signing does not use trapdoor td anymore,

$$\Pr[\mathsf{G}_1 = 1] - \Pr[\mathsf{G}_2 = 1] \leq \varepsilon_{lossy}.$$

By the regular lossyness of f_{ek}, the value $\tau(m^*)$ is information-theoretically hidden amongst the ℓ possible preimages of $\mathsf{f}_{ek}(\tau(m^*))$ and with probability $\frac{\ell-1}{\ell}$ we have $\mathsf{f}_{ek}(\tau(m^*)) = \mathsf{f}_{ek}(\tau^*)$ with $\tau(m^*) \neq \tau^*$. In the latter case we have a collision which contradicts again lossyness. More formally we can show that

$$\Pr[\mathsf{G}_2 = 1] \leq \frac{\ell - 1}{\ell} \cdot \varepsilon_{lossy}.$$

Summing up, we get $\varepsilon_{sig} \leq \Pr[G_0 = 1] - \Pr[G_2 = 1] \leq \varepsilon_{sim} + \varepsilon_{lossy} + \frac{\ell-1}{\ell}\varepsilon_{lossy}$ as claimed.

3.3 Tight security from one-wayness

In case TDP only satisfies the weaker security property of $(t, \varepsilon_{one-way})$-one-wayness, then we can obtain a non-tight security reduction with respect to $\varepsilon_{one-way}$.[3] As we will show now, a tight security reduction from one-wayness can be obtained by padding m with one random bit b, using a reduction technique by Katz and Wang [10].

Let TDP be a trapdoor permutation over $\mathcal{D}(n)$ and PAD be a padding scheme with message space $\mathcal{M}(n) \times \{0,1\}$ and range $\mathcal{R}(n) = \mathcal{D}(n)$. We now define an alternative signature scheme $\mathsf{SIG\text{-}MR'}$ with message space $\mathcal{M}$ which can be proved tightly secure from one-wayness of TDP.

[3] Concretely, the reduction uses a different random function $\mathcal{F}$ which partitions into solved instances $\mathcal{F}(m) := \mathsf{f}_{ek}(\tau(m))$ and unsolved instances $\mathcal{F}(m) \leftarrow \mathcal{D}(n)$ with a certain independent probability. This leads to $\varepsilon_{sig} = q_h\varepsilon_{one-way} + \varepsilon_{sim}$ or, similar to [6], $\varepsilon_{sig} = \exp(1)q_s\varepsilon_{one-way} + \varepsilon_{sim}$ if TDP is homomorphic.

Algorithm $\mathsf{Sign}'_{sk}(m \in \mathcal{M}(n))$
$b \leftarrow \{0,1\}$
$y := \pi_k(b\|m) \in \mathcal{D}(n)$
Return $\tau = \mathsf{f}_{td}^{-1}(y) \in \mathcal{D}(n)$

Algorithm $\mathsf{Recover}'_{pk}(\tau \in \mathcal{D}(n))$
$y = \mathsf{f}_{ek}(z)$
If $\pi_k^{-1}(y) = \perp$ then return $\perp$
Else compute $b\|m = \pi_k^{-1}(y)$
Return m

Theorem 3.3 *Suppose* TDP *is homomorphic and* $(t, \varepsilon_{one-way})$*-one-way and* PAD *is a perfectly sound and* $(q_{\mathcal{F}}, q_{\mathcal{H}}, q_{\mathsf{S}}, t_{sim}, \varepsilon_{sim})$*-simulatable padding scheme. Then* SIG-MR′ *is* $(t, q_h, q_s, 2\varepsilon_{one-way} + \varepsilon_{sim})$ *secure in the random oracle model.*

Proof. Let F be an adversary against the signature scheme that runs in time t_{sig}, makes at most q_s queries to the signing oracle, q_h queries to the random oracle, and outputs a forgery with probability ε_{sig}.

Game G_0**.** This is the UF-CMA game and hence $\Pr[\mathsf{G}_0 = 1] = \varepsilon_{sig}$.

Game G_1**.** Define this game as in the proof of Theorem 3.2 with a different definition of $\mathcal{F}$ and Sign_1.

Algorithm $\mathcal{F}(b\|m)$
If $\mathcal{F}(b\|m) \neq \perp$ then return $\mathcal{F}(b\|m)$
If $b(m) = \perp$ then $b(m) \leftarrow \{0,1\}$
if $b(m) = b$ then
  $\tau(m) \leftarrow \mathcal{D}(n)$
  Return $\mathcal{F}(b\|m) := \mathsf{f}_{ek}(\tau(m))$
Else return $\mathcal{F}(b\|m) \leftarrow \mathcal{D}(n)$ (*)

Algorithm $\mathsf{Sign}_1(m \in \mathcal{M}(n))$
If $b(m) = \perp$ then $b(m) \leftarrow \{0,1\}$
If $\mathcal{F}(b(m)\|m) = \perp$ then call $\mathcal{F}(b(m)\|m)$
Return $\tau(m)$

With the same argument as in the proof of Theorem 3.2 we get

$$\Pr[\mathsf{G}_0 = 1] - \Pr[\mathsf{G}_1 = 1] \leq \varepsilon_{sim}.$$

Let τ^* be the forgery and let $b^*||m^*$ be the bit and the recovered message. Note that the value $b(m^*)$ is information-theoretically hidden from the adversary's view and with probability 1/2 we have $b(m^*) \neq b^*$. In the latter case the adversary has managed to find a pre-image under f_{ek} on an uniformly distributed value coming from the outside of the experiment. More formally we can show that

$$\Pr[\mathsf{G}_1 = 1] \leq \frac{1}{2} \cdot \varepsilon_{one-way}.$$

For the simulation, we need TDP to be homomorphic to be able to embed one single challenge from the one-wayness experiment into all extended signatures τ which contain $b||m$ such that $b \neq b(m)$. Concretely, the adversary A against one-wayness inputs ek and $y = \mathsf{f}_{ek}(x)$. It simulates the oracles $\mathcal{F}$ and Sign as above where in the case $(*)$ $(b \neq b(m))$ of the $\mathcal{F}$-simulation he defines $F(b||m) := \mathsf{f}_{ek}(x(m)) \circ y$, for $x(m) \leftarrow \mathcal{D}(n)$. Finally, when F outputs his forgery τ^*, A recovers $b^*||m^*$ and aborts if $b(m^*) = b^*$. This happens with probability 1/2. Otherwise, we have $\tau^* = \mathsf{f}^{-1}(\mathcal{F}(b^*||m^*)) = x(m^*) \circ \mathsf{f}_{ek}^{-1}(y)$, from which the pre-image of y can be computed.

Summing up, we get $\varepsilon_{sig} \leq \Pr[G_0 = 1] - \Pr[G_1 = 1] \leq \varepsilon_{sim} + \frac{1}{2}\varepsilon_{one-way}$ as claimed.

3.4 Domain Extension

The construction of a padding based signature scheme SIG-MR with message recovery from Sections 3.2 and 3.3 requires that the range $\mathcal{R}$ of the padding scheme is equivalent to the domain $\mathcal{D}$ of the trapdoor permutation. In particular, the message space $\mathcal{M}$ of SIG-MR cannot be larger than $|\mathcal{D}|$. (Because of the required redundancy, it is strictly smaller.)

We will now describe a simple (almost) generic way to turn such a scheme SIG-MR into a signature scheme SIG-MR* with arbitrary message space $\mathcal{M} \times \{0,1\}^*$ without increasing the redundancy and where SIG-MR* comes with almost the same security guarantee (in the random oracle model) as SIG-MR.

Let $\mathcal{H}$ denote the hash function used in SIG-MR. The scheme SIG-MR* has the same key-space as SIG-MR, and signs a message $(m, m^*) \in \mathcal{M} \times \{0,1\}^*$ by computing $\tau \leftarrow \mathsf{SIG\text{-}MR}(m)$, but where each hash function call $\mathcal{H}(x)$ is replaced with $\mathcal{H}(m^*, x)$.[4] The signature is (m^*, τ). To verify (m^*, τ) one simply verifies τ as in SIG-MR, but using the hash function $\mathcal{H}(m^*, \cdot)$. If this verification accepts and outputs a message m, the verification for SIG-MR* accepts and outputs (m, m^*).[5]

We claim that SIG-MR* is secure if SIG-MR is. To see this, first assume the adversary against SIG-MR* only makes signature/hash queries for the same fixed m^* (i.e., signature queries (m, m^*) for any m and hash queries (m^*, x) for any x.) Then the security of SIG-MR* can be proven exactly as for SIG-MR, except that throughout the security experiment we use the random oracle $\mathcal{H}(m^*, \cdot)$ instead of $\mathcal{H}(\cdot)$.

Let us now consider the general case where the adversary makes signature/hash queries for different m^*'s. In the security proof for SIG-MR we run a simulator S to program the random oracle. In the proof for SIG-MR* we now simply start a new simulator S_{m^*} whenever the adversary makes a query with a new m^*. We can think of this proof as programming many independent random oracles $\mathcal{H}(m^*, \cdot)$ for different m^*. The simulation fails, if any of the simulators S_{m^*} fails. If our padding scheme is proven to be $(q, \varepsilon(q))$ simulatable where $\varepsilon(q)$ is convex, in particular, if for any $q_1, \ldots, q_t, \sum_{i=1}^t q_i = q$ it satisfies $\sum_{i=1}^t \varepsilon(q_i) \leq \varepsilon(q)$. Then we also get $(q, \varepsilon(q))$ simulability for the potentially many different simulations. The bounds on simulability we prove are of the form $\varepsilon(q) = q^{1+c}/d$ (for some constant $c > 0$ and a term d that does not depend on q), and thus satisfy this convexity condition.

4 Padding schemes from Feistel networks

4.1 The two round Feistel network

Consider the (random-oracle) padding scheme $\mathsf{PAD}_{2f}[\rho] = (\pi, \pi^{-1})$ from Figure 1 (left) which is derived from an unbalanced two-round Feistel network ϕ_{2f} instantiated with random oracles $\mathcal{H}_1 : \mathbb{Z}_\mu \to \mathbb{Z}_\rho, \mathcal{H}_2 : \mathbb{Z}_\rho \to \mathbb{Z}_\mu$

$$\phi_{2f}(x, v) = (x + \mathcal{H}_2(\mathcal{H}_1(x) + v), \mathcal{H}_1(x) + v) \qquad \phi_{2f}^{-1}(w, y) = (w - \mathcal{H}_2(y), y - \mathcal{H}_1(w - \mathcal{H}_2(y))$$

$$\text{as} \qquad \pi(x) = \phi_{2f}(x, 0) \qquad \pi^{-1}(w, y) = \begin{cases} x & \text{if } \phi_{2f}^{-1}(w, y) = (x, 0) \\ \perp & \text{otherwise} \end{cases}$$

[4] Here $\mathcal{H}(a, b)$ means we invoke $\mathcal{H}$ on some efficiently uniquely decodable encoding of the message pair (a, b). Such an encoding is, for example, given by $0^{l_a} \| L_a \| a \| b$ where L_a is the length of a in binary, and l_a is the length of L_a.

[5] A more efficient solution (whenever the padding queries $\mathcal{H}$ more than once) is to prepend $\mathcal{G}(m^*)$ instead of m^* for some collision resistant hash function $\mathcal{G}$ (e.g., a random oracle). Alternatively, if $\mathcal{H}$ is an iterated hash function, one must hash the prefix m^* only once, and can then evaluate $\mathcal{H}(m^*, x)$ at basically the cost of hashing only x. The complexity added by the two solutions outlined above is just the cost of hashing m^* once.

This will serve as an example of an easy to analyze padding scheme and to prepare for our four round Feistel network in the next Section. (For that reason the Feistel network uses modular addition and not binary xor, as usual.)

Theorem 4.1 (Simulability of PAD_{2f}, implicit in [4]) *The padding scheme $\mathsf{PAD}_{2f}[\rho]$ as illustrated in Figure 1 (left) is $(q_{\mathcal{F}}, q_{\mathcal{H}}, q_{\mathsf{S}}, t_{sim}, \varepsilon_{sim})$ simulatable for any $q_{\mathcal{F}}, q_{\mathcal{H}}$ and with $q_{\mathsf{A}} = q_{\mathcal{F}} + q_{\mathcal{H}}$*

$$q_{\mathsf{S}} = q_{\mathsf{A}} \qquad t_{sim} = q_{\mathsf{A}} \cdot polylog(\mu) \qquad \varepsilon_{sim} = q_{\mathsf{A}}^2/2\rho.$$

More precisely, we can set $t_{sim} = O(q_{\mathsf{A}} \log(q_{\mathsf{A}}) \log(\mu))$ using that the cost per (find or insert) operation on a sorted list with $\leq q_{\mathsf{A}}$ elements of size $\log(\mu)$ bits is $O(\log(q_{\mathsf{A}}) \log(\mu))$.

Proof. Let $\pi_{2f}^{\mathcal{H}}(\cdot) = \phi_{2f}(\cdot, 0)$ denote the evaluation function of PAD_{2f} instantiated with two randomly chosen functions $\mathcal{H}_1 : \mathbb{Z}_\mu \to \mathbb{Z}_\rho, \mathcal{H}_2 : \mathbb{Z}_\rho \to \mathbb{Z}_\mu$. We have to specify the simulator S such that for any $(q_{\mathcal{F}}, q_{\mathcal{H}})$-query adversary A and a random function $\mathcal{F} : \mathbb{Z}_\mu \to \mathbb{Z}_\mu \times \mathbb{Z}_\rho$

$$|\Pr[\mathsf{A}^{\pi_{2f}^{\mathcal{H}}, \mathcal{H}}(1^n) = 1] - \Pr[\mathsf{A}^{\mathcal{F}, \mathsf{S}^{\mathcal{F}}}(1^n) = 1]| \leq (q_{\mathcal{F}} + q_{\mathcal{H}})^2/\rho \tag{3}$$

The probability is over the choice of $\mathcal{F}, \mathcal{H}$ and the randomness used by S and A. The simulator $\mathsf{S}^{\mathcal{F}}$ will internally define fake random oracles $\hat{\mathcal{H}}_1, \hat{\mathcal{H}}_2$ by lazy sampling which initially are undefined on all inputs. $\hat{\mathcal{H}}_1(x) = \perp$ denotes $\hat{\mathcal{H}}_1$ is undefined on input x. The set $\mathcal{X} \subset \mathbb{Z}_\mu$ will denote the inputs on which $\hat{\mathcal{H}}_1$ has already been defined, and $\mathcal{A} = \hat{\mathcal{H}}_1(\mathcal{X})$ are the corresponding outputs. Similarly $\mathcal{Y}, \mathcal{B} = \hat{\mathcal{H}}_2(\mathcal{Y})$ denote the inputs and the corresponding outputs on which $\hat{\mathcal{H}}_2$ has been defined. The simulator also initializes a variable $\mathsf{FAIL} := 0$ which will only be used in the proof below. Informally, if at the end of the experiment $\mathsf{FAIL} = 1$ then this indicates that $\mathsf{S}^{\mathcal{F}}$ failed to define the $\hat{\mathcal{H}}_1, \hat{\mathcal{H}}_2$ such that $\pi_{2f}(\cdot)$ looks consistent with $\mathcal{F}(\cdot)$ given all queries made so far. If this happens, the simulator aborts which means it refuses to answer any more queries. We now define how $\mathsf{S}^{\mathcal{F}}$ answers queries to $\hat{\mathcal{H}}_1$ and $\hat{\mathcal{H}}_2$ and how it updates its state if $\mathsf{A}^{\mathcal{F}, \mathsf{S}^{\mathcal{F}}}$ makes an $\mathcal{F}$ query.

$\hat{\mathcal{H}}_2$ **query** $y \in \mathbb{Z}_\rho$ **:** If $y \notin \mathcal{Y}$ (equivalently $\hat{\mathcal{H}}_2(y) = \perp$) sample $b \leftarrow \mathbb{Z}_\mu$ and set $\hat{\mathcal{H}}_2(y) := b$. Output $\hat{\mathcal{H}}_2(y)$.

$\hat{\mathcal{H}}_1$ **or** $\mathcal{F}$ **query** $x \in \mathbb{Z}_\mu$ **:** If $x \notin \mathcal{X}$ try to program the $\hat{\mathcal{H}}_i$ s.t. $\pi_{2f}^{\hat{\mathcal{H}}}(x, 0) = \mathcal{F}(x)$ as follows

1. query $(f_0, f_1) \leftarrow \mathcal{F}(x)$ and set $\hat{\mathcal{H}}_1(x) := f_1$.
2. if $f_1 \in \mathcal{Y}$ set $\mathsf{FAIL} := 1$ and abort.
3. set $\hat{\mathcal{H}}_2(f_1) := x - f_0$.

If this is an $\hat{\mathcal{H}}_1$ (and not a $\mathcal{F}$) query output $\hat{\mathcal{H}}_1(x)$.

Considering the efficiency of our simulator, note that $\mathsf{S}^{\mathcal{F}}$ does exactly one oracle query for every $\mathcal{F}$ and $\mathcal{H}_1$ query of $\mathsf{A}^{\mathcal{F}, \mathcal{H}}$ (and no query for an $\mathcal{H}_2$ query), so $q_{\mathsf{S}} \leq q_{\mathcal{H}} + q_{\mathcal{F}}$.

To prov eq.(3) we'll first bound the probability that $\mathsf{FAIL} = 1$ in the above experiment. Note that $|\mathcal{Y}|$ (i.e. the number of inputs on which $\hat{\mathcal{H}}_2$ is defined) increases by at most 1 on every $\hat{\mathcal{H}}_1, \hat{\mathcal{H}}_2$ and $\mathcal{F}$ query, thus

$$|\mathcal{Y}| \leq (q_{\mathcal{H}} + q_{\mathcal{F}})$$

Further, FAIL can only be set to 1 on a $\hat{\mathcal{H}}_1$ or $\mathcal{F}$ query, and this happens if the uniformly random f_1 is in $\mathcal{Y}$. For every query, this happens with probability $\leq |\mathcal{Y}|/\rho$. Taking the union bound over all $\hat{\mathcal{H}}_1, \mathcal{F}$ queries we get

$$\Pr[\mathsf{FAIL} = 1] \leq (q_{\mathcal{H}} + q_{\mathcal{F}})^2/\rho$$

Next, well argue that

$$\left|\Pr[\mathsf{A}^{\pi^{\mathcal{H}},\mathcal{H}}(1^n)=1]-\Pr[\mathsf{A}^{\mathcal{F},\mathsf{S}^{\mathcal{F}}}(1^n)=1]\right| \le \Pr[\mathsf{FAIL}=1] \tag{4}$$

Note that the two equation above imply eq.(3).

Let $\langle \mathsf{A}^{\mathcal{F},\mathsf{S}^{\mathcal{F}}}\rangle$ denote the transcript containing all queries and corresponding answers of the oracle queries made by A. For any possible transcript τ

$$\forall \tau \ : \ \Pr[\tau = \langle \mathsf{A}^{\mathcal{F},\mathsf{S}^{\mathcal{F}}}\rangle \wedge \mathsf{FAIL}=0] \le \Pr[\tau = \langle \mathsf{A}^{\pi^{\mathcal{H}},\mathcal{H}}\rangle] \tag{5}$$

To see this, note S assigns uniformly random values to the $\hat{\mathcal{H}}_i$ (the randomness is either sampled directly or comes from the random function $\mathcal{F}$) which are independent of anything that happend so far. Eq.(5) implies eq.(4) by standard arguments like the fundamental lemma of game-playing. ■

Corollary 4.2 (Security and minimal overhead for $\mathsf{SIG\text{-}MR}_{2f}$) *Let TDP be a $(t_{lossy}, \varepsilon_{lossy}, \ell)$-lossy trapdoor permutation with domain $\mathcal{R}(n)$. Then by Theorem 3.2 $\mathsf{SIG\text{-}MR}_{2f}[\rho] = \mathsf{SIG\text{-}MR}[\mathsf{PAD}_{2f}[\rho], \mathsf{TDP}]$ is a $(t_{sig}, q_s, q_h, \varepsilon_{sig})$-secure signature scheme with*

$$t_{sig} = t_{lossy} - (q_h+q_s)\cdot poly(n), \quad \varepsilon_{sig} = \frac{(q_s+q_h)^2}{2\rho} + \frac{2\ell-1}{\ell}\varepsilon_{lossy}. \tag{6}$$

Assuming $2\varepsilon_{lossy}/t_{sig} \le 2^{-n-1}$, the overhead required to get n bits security with (q_s, q_h) queries is

$$oh_{\mathsf{SIG\text{-}MR}_{2f}}(n, q_s, q_h) \ge n + \log(q_h+q_s) \quad or \quad oh_{\mathsf{SIG\text{-}MR}_{2f}}(n) \ge 2n$$

assuming only the trivial $q_h + q_s \le 2^n$ bound on the number of queries.

Proof. We compute the overhead according to Definition 2.1. Using $2\varepsilon_{lossy}/t_{sig} \le 2^{-n-1}$ and $t_{sig} \ge q_s + q_h$ we obtain by (6)

$$\frac{\varepsilon_{sig}}{t_{sig}} \le \frac{(q_s+q_h)^2}{2\rho t_{sig}} + 2^{-n-1} \le \frac{q_s+q_h}{2\rho} + 2^{-n-1}$$

To get n bits of security we must set the overhead ρ such that $\varepsilon_{sig}/t_{sig} \le 2^{-n}$, which holds for $\log \rho := n + \log(q_s+q_h)$. ■

Note that $\mathsf{SIG\text{-}MR}_{2f}$ is the same as $\mathsf{PSS\text{-}MR}$ with modular addition instead of xor. Hence Corollary 4.2 similar essentially reproves a theorem of [9] about the security of $\mathsf{PSS\text{-}MR}$, expressed in our general framework.

The following lemma says that one cannot avoid the additional additive factor q_h^2/ρ in the security reduction, hence the overhead of $oh_{\mathsf{SIG\text{-}MR}_{2f}} = n + \log(q_h)$ bits is optimal for $\mathsf{SIG\text{-}MR}_{2f}$.

Lemma 4.3 *If there exists a one-way (lossy) TDP, then there exists a one-way (lossy) TDP' such that for all q, $\mathsf{SIG}_{\mathsf{PAD}_{2f},\mathsf{TDP}'}$ is not $(t_{sig} = O(q), q_h = q, q_s = 0, \varepsilon_{sig} = q^2/\rho)$-secure.*

Proof. We define the evaluation function of TDP' as $\mathsf{f}'(z,y) := (z, \mathsf{f}(y)) \in \mathbb{Z}_\mu \times \mathbb{Z}_\rho$. Clearly, one-wayness and lossiness are inherited. The attack on TDP' is as follows. First, F picks uniform $x_1, \ldots, x_q$ and computes $y_i := \mathsf{f}(x_i)$. Next, it makes q queries arbitrary distinct $m_1, \ldots, m_q$ to the $\mathcal{H}_1$ oracle. The probability that there exists indices $i, j \in \{1, \ldots, q\}$ such that $\mathcal{H}_1(m_i) = y_j$ is bounded by q^2/ρ. In case they exist, then $\tau := (m_i + \mathcal{H}_2(y_j), x_j)$ is a valid signature, i.e., $\mathsf{Recover}'(\tau) = m_i$. ■

4.2 The four round Feistel network

We now define our main padding scheme $\mathsf{PAD}_{4F}[\rho]$ based on an unbalanced four-round Feistel network as illustrated in Figure 1. ρ is a parameter controlling the overhead. We will later prove that $\mathsf{PAD}_{4F}[\rho]$ is $(q, q^{1+o(1)}/\rho)$ simulatable (as compared to the $(q, q^2/\rho)$ simulability of $\mathsf{PAD}_{2f}[\rho]$.)

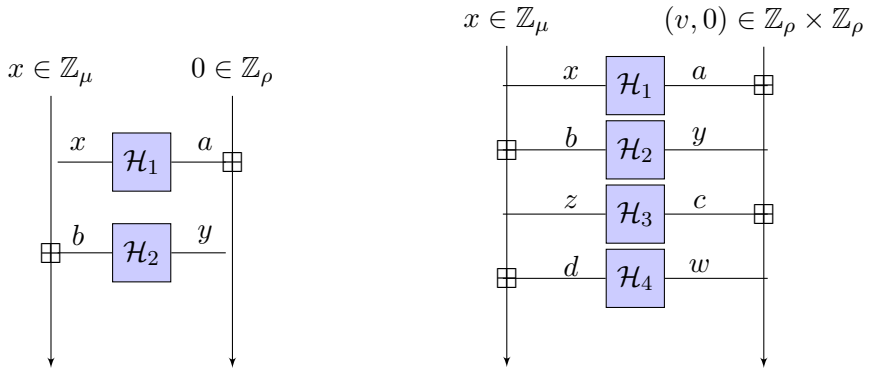

Figure 1: **(left)** The evaluation function $\pi(x) = \phi_{2f}(x, 0)$ of the padding $\mathsf{PAD}_{2f}[\rho]$. It is derived from an unbalanced two-round Feistel network ϕ_{2f}, instantiated with random oracles $\mathcal{H}_1, \mathcal{H}_2$. The redundancy is $\log_2(\rho)$ bits. $\boxplus$ denotes component-wise addition in the respective domains. **(right)** The evaluation function $\pi(x, v) = \phi_{4F}(x, v, 0)$ of the padding $\mathsf{PAD}_{4F}[\rho]$, also with $\log_2(\rho)$ bits redundancy.

Let $Q(\mu, q)$ denote the random variable that takes as value the number of edges of the densest $q \times q$ subgraph of an $\mu \times \mu$ random bipartite Cayley graph of degree q (we give a formal definition in Section 5.1) We'll prove the following theorem which bounds the simulability of PAD_{4F} in terms of the expected value of $Q(\mu, q)$.

Theorem 4.4 (Simulability of PAD_{4F}) *The padding scheme $\mathsf{PAD}_{4F}[\rho]$ with evaluation function $\pi : \mathbb{Z}_\mu \times \mathbb{Z}_\rho \to \mathbb{Z}_\mu \times \mathbb{Z}_\rho \times \mathbb{Z}_\rho$ as illustrated in Figure 1 is $(q_\mathcal{F}, q_\mathcal{H}, q_\mathsf{S}, t_{sim}, \varepsilon_{sim})$ simulatable for any $q_\mathcal{F}, q_\mathcal{H}$ and with $q_\mathsf{A} = q_\mathcal{F} + q_\mathcal{H}$*

$$q_\mathsf{S} \le q_\mathsf{A} \log(\rho) \qquad t_{sim} = q_\mathsf{S} \cdot polylog(\mu) \qquad \varepsilon_{sim} = \frac{2\mathsf{E}[Q(\mu, q_\mathsf{A})]}{\rho} + \frac{2q_\mathsf{A}^4}{\mu} + \frac{2q_\mathsf{A}^2}{\rho^2} \cdot \left(\frac{\log(\rho)}{\log(\rho/q_\mathsf{A})}\right)^2. \quad (7)$$

4.3 The Overhead of $\mathsf{SIG\text{-}MR}[\mathsf{PAD}_{4F}[\rho], \mathsf{RSA}]$

We will prove in Section 6 that for any $0 < a < 1/2$ we have an upper bound $Q(\mu, \mu^a) \le \mu^{a(1+a/(1-2a)}$ except with some extremely tiny probability $O(\exp(-\mu^\beta))$ for some $\beta > 0$ which we can safely ignore. As long as

$$(i) \quad (\log \rho - \log q_\mathsf{A})^2 > \log^2(\rho) \cdot 2 \cdot q_\mathsf{A}/\rho \qquad \text{and} \qquad (ii) \quad \mu \ge 2q_\mathsf{A}^3\rho \quad (8)$$

The last two terms of ε_{sim} in (7) are $\le q_\mathsf{A}/\rho \le \mathsf{E}[Q(\mu, q_\mathsf{A})]/\rho$ and we can simplify $\varepsilon_{sig} \le 4\mathsf{E}[Q(\mu, q_\mathsf{A})]/\rho$. In order to bound ε_{sig}, we need to bound $a = \log q_\mathsf{A}/\log \mu$ (as $q_\mathsf{A} = \mu^a$). As we're interested in 80 bit security, we can assume $q_\mathsf{A} \le 2^{80}$. And as we're interested in an instantiation over an RSA modulus N with $\log N \ge 1024$, we can assume $\log \mu = \log N - 2\log \rho \ge 800$, and thus get an upper bound $a \le \log q_\mathsf{A}/\log \mu \le 0.1$. By Theorem 6.1

$$\mathsf{E}[Q(\mu, q_\mathsf{A} = \mu^a)] \le \mu^{a(1+a/(1-2a))} = q_\mathsf{A}^{1+a/(1-2a)} \le q_\mathsf{A}^{1.125}$$

We assume the parameters of the TDP are such that $2\varepsilon_{lossy}/t_{sim} \leq 2^{-n-1}$. Then using Theorem 3.2 in the first and $t_{sim} \geq q_{\mathsf{A}}$ in the last step

$$\frac{\varepsilon_{sim}}{t_{sim}} \leq \frac{\varepsilon_{sig} + 2\varepsilon_{lossy}}{t_{sim}} \leq \frac{\varepsilon_{sig}}{t_{sim}} + 2^{-n-1} \leq \frac{4\mathsf{E}[Q(\mu, q_{\mathsf{A}}])}{\rho t_{sim}} + 2^{-n-1} \leq \frac{4q_{\mathsf{A}}^{1.125}}{\rho t_{sim}} + 2^{-n-1} \leq \frac{4q_{\mathsf{A}}^{0.125}}{\rho} + 2^{-n-1}$$

Setting $\rho = 2^{1.125n+3}$ and using $q_{\mathsf{A}} \leq 2^n$ we further get

$$\frac{4q_{\mathsf{A}}^{0.125}}{\rho} + 2^{-n-1} \leq \frac{4}{2^{n+3}} + 2^{-n-1} = 2^{-n}$$

Thus an overhead of $oh_{\mathsf{SIG\text{-}MR}_{4F}}(n) = 1.125n + 3$ in sufficient for n bits security, in particular for $n = 80$ we achieve an overhead of $\log \rho = 1.125n + 3 = 93$ (note that the constraints in eq.(8) are indeed satisfied for this parameters, i.e. $\rho = 2^{93}, q_{\mathsf{A}} \leq 2^{80}, \mu \geq 2^{800}$).

We can push the overhead closer to the theoretical minimum of 80 by either setting a non-trivial upper bound $q_{\mathsf{A}} \ll 2^n$ on the number of oracle queries or increasing the domain of the TDP, say to 2048 instead 1024 as we did now. For example, for $q_{\mathsf{A}} \leq 2^{3n/4}$ (which is 2^{60} for $n = 80$) we get $a \leq \log q_{\mathsf{A}} / \log_{\mu} = 60/800 = 0.075$. Then

$$4q_{\mathsf{A}}^{0.075}/\rho \leq 2^{-n-1}$$

Holds for $\log \rho = 1.05625n + 3$, thus $oh_{\mathsf{SIG\text{-}MR}_{4F}}(n, q = 2^{3n/4}) = 1.05625n + 3$, which is $87.5 < 88$ for $n = 80$.

5 Proofs for the Four Round Feistel Padding

The proof is organized as follows. First, in Section 5.1 we define $Q(q, \mu)$. Next, we prove a technical lemma (Lemma 5.3) which informally bounds the advantage of any q query adversary in making a fresh query x to $\mathcal{H}_1$ (variables as in Figure 2, right) such that for some v, in the evaluation of $\phi_{4F}(x, v, 0)$ the input z to $\mathcal{H}_3$ has already been queried. Using Lemma 5.3, in Section 5.3 we prove the simulability of the evaluation function $\pi(\cdot) = \phi_{4F}(\cdot, \cdot, 0)$ of PAD_{4F} as claimed in Theorem 4.4. Finally, in Section 6 we prove a $q^{1+o(1)}/\rho$ upper bound on the expected value of $Q(q, \mu)$.

5.1 Density of Subgraphs of Random Cayley Graphs

For $\mu, q \in \mathbb{N}$ let $\mathcal{B}$ be a subset of $\mathbb{Z}_\mu$ of size q, we define the value

$$Q(\mu, q, \mathcal{B}) = \max_{\mathcal{X}, \mathcal{Z} \subset \mathbb{Z}_\mu, |\mathcal{X}| = |\mathcal{Z}| = q} |\{(x, z) \mid x \in \mathcal{X}, z \in \mathcal{Z}, z - x \in \mathcal{B}\}| \tag{9}$$

We will be interested in the random variable $Q(\mu, q, \mathcal{B})$ where $\mathcal{B}$ is a randomly chosen q element subset of $\mathbb{Z}_\mu$, we denote this variable by $Q(\mu, q)$.

It will be convenient to think of $Q(\mu, q)$ in terms of random Cayley graphs as we will explain now. For $\mathcal{B} \subset \mathbb{Z}_\mu, |\mathcal{B}| = q$ we denote with $\mathcal{C}(\mu, q, \mathcal{B})$ the bipartite graph with μ vertices on each side which we identify with the elements of $\mathbb{Z}_\mu$.The edge set is $e(\mathcal{C}(\mu, q, \mathcal{B})) = \{(x, z) \ : \ z - x \in \mathcal{B}\}$, that is, (x, z) is an edge if $x + b = z$ for some $b \in \mathcal{B}$. With $\mathcal{C}(\mu, q)$ we denote the random graph $\mathcal{C}(\mu, q, \mathcal{B})$ for a random $\mathcal{B} \subset \mathbb{Z}_\mu, |\mathcal{B}| = q$.

With this notion $Q(\mu, q, \mathcal{B})$ is the maximum number of edges in any subgraph of $\mathcal{C}(\mu, q, \mathcal{B})$ with q vertices on each side. Trivial lower and upper bounds on $Q(\mu, q, \mathcal{B})$ are

$$\forall \mathcal{B} \subset \mathbb{Z}_q, |\mathcal{B}| = q \ : \ 2q - 1 \leq Q(\mu, q, \mathcal{B}) \leq q^2$$

In the proposition below we observe that known results on the edge density of graphs without 4-cycles already give us an $q^{1.5}$ upper bound on the expected value of $Q(\mu, q, \mathcal{B}) \leftarrow Q(\mu, q)$. In Section 6 we will prove an upper bound of $q^{1+o(1)}$ almost matching the lower bound.

Proposition 5.1 *If $\mu \geq q^5$ then $\mathsf{E}[Q(\mu, q)] \leq q^{1.5} + 3q$.*

Proof. We first observe that $\mathcal{C}(\mu, q, \mathcal{B}) \leftarrow \mathcal{C}(\mu, q)$ has a 4-cycle with probability at most

$$\Pr[G \leftarrow \mathcal{C}(\mu, q) \ : \ G \text{ has a 4-cycle}] \leq q^4/\mu$$

The proposition now follows from a result by Naor and Verstraëte [11] who show that a bipartite graph with q vertices on each side that does not contain a 4-cycle has at most $q^{1.5} + 2q$ edges. With probability at most $q^4/\mu \leq 1/q$ we have a cycle, in which case we use the trivial q^2 upper bound which adds another $q = q^2/q$ to the expected value. ∎

5.2 A Game on Three Round Feistel

Figure 2: **(left)** An unbalanced three-round Feistel network ϕ_{3f} over $\mathbb{Z}_\mu \times \mathbb{Z}_\rho$. **(right)** The three-round Feistel network ϕ_{3F} with an extra $\mathbb{Z}_\rho$ domain on the right side. This permutations define evaluation functions of padding schemes $\pi_{3f}(x) = \phi_{3f}(x, 0)$ and $\pi_{3F}(x, v) = \phi_{3F}(x, v, 0)$ by fixing the rightmost $\mathbb{Z}_\rho$ part of the input to 0.

In this section we describe a game, where an attacker A can query three randomly chosen functions $\mathcal{H}_1, \mathcal{H}_3 : \mathbb{Z}_\mu \to \mathbb{Z}_\rho, \mathcal{H}_2 : \mathbb{Z}_\rho \to \mathbb{Z}_\mu$, which we think of as round functions of a Feistel network ϕ_{3f} as illustrated in Figure 2 (left). Informally, the adversary wins if she makes a fresh query x to $\mathcal{H}_1$, such that the input z to $\mathcal{H}_3$ in the evaluation of $\phi_{3f}(x, 0)$ has already been queried. We will call this game the z-collision game and prove (in Lemma 5.2 below) an $\mathsf{E}[Q(\mu, q)]/\rho$ upper bound for any q-query adversary for the z-collision game.

Next, we will show that the same bound on the winning advantage holds for a similar game on the Feistel-network ϕ_{3F} as illustrated in Figure 2 (right), where we have an extra $\mathbb{Z}_\rho$ domain on the right side. Here we say the adversary wins if she makes a query $x \in \mathbb{Z}_\mu$ such that there exists a $v \in \mathbb{Z}_\rho$ such that in the evaluation of $\phi_{3F}(x, v, 0)$ the input z is not fresh.

We will later use the bound on the winning advantage for the z-collision game on ϕ_{3F} as key technical lemma to prove the simulability of π_{4F}. As in the simulability proof the random functions are defined by lazy sampling (done by the simulator), we will already use lazy sampling in the proof of our upper bound for the z-collision game. More precisely, we will consider functions $\hat{\mathcal{H}}_1, \hat{\mathcal{H}}_2, \hat{\mathcal{H}}_3$ which initially are undefined on all inputs. The sets $\mathcal{X}, \mathcal{Z} \subset \mathbb{Z}_\mu, \mathcal{Y} \subset \mathbb{Z}_\rho$ denote the inputs to $\hat{\mathcal{H}}_1, \hat{\mathcal{H}}_3$ and $\hat{\mathcal{H}}_2$ on which the outputs have been defined, initially $\mathcal{X}, \mathcal{Y}, \mathcal{Z} = \emptyset$. Moreover we initialize a

variable $\mathsf{FAIL} := 0$, the adversary wins the game if at the end of the game $\mathsf{FAIL} > 0$. We will assume that $\hat{\mathcal{H}}_2$ is a random *injective* function as this will make the proofs a bit cleaner. One cannot distinguish a random from a random injective function with range $\mathbb{Z}_\mu$ making q queries with advantage better than q^2/μ. As we'll later set $\mu \geq \rho^3$, this term will be dominated by other terms $\Omega(q/\rho)$ and thus we will simply ignore it.

The z-collision game on ϕ_{3f}. Consider an adversary A who can make queries to the three functions (at most most q to each) which are answered as follows:

$\hat{\mathcal{H}}_2$ **query** $y \in \mathbb{Z}_\rho$: If $y \notin \mathcal{Y}$ sample $b \leftarrow \mathbb{Z}_\mu \setminus \mathcal{B}$ and set $\hat{\mathcal{H}}_2(y) := b$.[6] Output $\hat{\mathcal{H}}_2(y)$.

$\hat{\mathcal{H}}_3$ **query** $z \in \mathbb{Z}_\mu$: If $z \notin \mathcal{Z}$ sample $c \leftarrow \mathbb{Z}_\rho$ and set $\hat{\mathcal{H}}_3(z) := c$. Output $\hat{\mathcal{H}}_3(z)$.

$\hat{\mathcal{H}}_1$ **query** $x \in \mathbb{Z}_\mu$: If $x \notin \mathcal{X}$ then

1. sample $a \leftarrow \mathbb{Z}_\rho$ and set $\hat{\mathcal{H}}_1(x) := a$.
2. If $x + \hat{\mathcal{H}}_2(a) \in \mathcal{Z}$ then $\mathsf{FAIL} := \mathsf{FAIL} + 1$.

Output $\hat{\mathcal{H}}_1(x)$.

A wins the z-collision game if at the end $\mathsf{FAIL} > 0$. If we forget about the lazy sampling and consider an A who plays the game against ϕ_{3f} (i.e. instantiated with random function $\mathcal{H}_1, \ldots, \mathcal{H}_3$), then A wins the z-collision game if at same point she makes a query x s.t. the inputs y, z to $\mathcal{H}_2$ and $\mathcal{H}_3$ (as in Figure 2) in the evaluation of $\phi_{3f}(x, 0)$ have already been queried.

We will now upper bound the success probability of any adversary making at most q queries to each of the three function to win the z-collision game (i.e. achieve $\mathsf{FAIL} > 0$).

Trivial lower and upper bounds on the winning advantage of the z-collision game are q/ρ and q^2/r. We will now give an upper bound on the advantage in terms of $Q(\mu, q)$ introduced in Section 5.1.

Lemma 5.2 *The advantage of any q-query adversary A in winning the z-collision game on ϕ_{3f} (i.e. force $\mathsf{FAIL} > 0$) is at most*

$$\Pr[\mathsf{FAIL} > 0] \leq \mathsf{E}[Q(\mu, q)]/\rho$$

Proof. We will only sketch the proof, as this lemma is used to get an intuition for Lemma 5.3 below.

Consider the queries $\mathcal{X}, \mathcal{Y}, \mathcal{Z}$ made by A, and recall that $\mathcal{B} = \hat{\mathcal{H}}_2(\mathcal{Y})$. Consider the $(\mathcal{X}, \mathcal{Y})$ subgraph of the Cayley graph $\mathcal{C}(\mu, q, \mathcal{B})$. Now for any $x \in \mathcal{X}$, let $\mathcal{B}_x = \{b \in \mathcal{B} \ : \ x + b \in \mathcal{Z}\}$ be the number of edges from x to $\mathcal{Z}$. The x query will increase FAIL iff the $a \leftarrow \mathbb{Z}_\rho$ we sampled is a preimage of some $b \in \mathcal{B}_x$ (i.e. $\hat{\mathcal{H}}_2(a) = b \in \mathcal{B}_x$), this probability is $|\mathcal{B}_x|/\rho$. Summing over all $x \in \mathcal{X}$ we get $\mathsf{E}[\mathsf{FAIL}] = \sum_{x \in \mathcal{X}} |\mathcal{B}_x|/\rho = Q(\mu, q, \mathcal{B})/\rho$. This is the expectation after $\mathcal{B}$ has been fixed, as $\mathcal{B}$ is a random subset

$$\mathsf{E}[\mathsf{FAIL}] \leq \mathsf{E}[Q(\mu, q)]/\rho \tag{10}$$

Finally note that $\mathsf{E}[\mathsf{FAIL}] \geq \Pr[\mathsf{FAIL} = 1]$ as FAIL is an integer ≥ 0.

Let us mention that eq.(10) is tight and equality is achieved by the following attack strategy. First make any q queries to $\hat{\mathcal{H}}_2$ which gives us a random set $\mathcal{B}$ (as $\hat{\mathcal{H}}_2$ is injective, $|\mathcal{B}| = |\mathcal{Y}| = q$). Next, identify the sets $\mathcal{X}, \mathcal{Z}$ of size q s.t. the $(\mathcal{X}, \mathcal{Z})$ subgraph of $\mathcal{C}(\mu, q, \mathcal{B})$ has $Q(\mu, q, \mathcal{B})$ edges. Make all $\mathcal{Z}$ queries to $\hat{\mathcal{H}}_3$, followed by all $\mathcal{X}$ queries to $\hat{\mathcal{H}}_1$. ∎

[6] Note that we sample the output from $\mathbb{Z}_\mu \setminus \mathcal{B}$ because we want $\hat{\mathcal{H}}_2$ to behave like a random *injective* function.

Ultimately, our goal is to prove simulability of a padding scheme. The above lemma is a good start as it tells us that for the evaluation function $\pi_{3f}(x) = \phi_{3f}(x, 0)$ with probability $1 - \mathsf{E}[Q(\mu, q)]/\rho$ the following holds: whenever a q-query adversary makes an x query to $\hat{\mathcal{H}}_1$, the resulting z input to $\hat{\mathcal{H}}_3$ will be "fresh", and thus we will be able to program the output $c := \hat{\mathcal{H}}_3(z)$ such that it is consistent with $\mathcal{F}(x)$. By adding one more round to the Feistel network we will be able to program also the left $\mathbb{Z}_\mu$ part of the input. Unfortunately, this will only work as long as the inputs to this fourth function are fresh. As its inputs are over $\mathbb{Z}_\rho$, there's a $\Theta(q^2/\rho)$ chance we have a collision on these inputs and will not be able to program after all. Summing up, we are no better than the q^2/ρ bound we already got for the two round Feistel in Theorem 4.1. To overcome this problem, we will simply increase the domain on the right side of the Feistel to $\mathbb{Z}_\rho \times \mathbb{Z}_\rho$, but in order to not increase the redundancy space, this extra $\mathbb{Z}_\rho$ space is used for the message, not redundancy. We will now show that the z-collision game on this new $\pi_{3F}(x, v) = \phi_{3F}(x, v, 0)$ padding scheme is still hard.

The z-collision game on ϕ_{3F}.

$\hat{\mathcal{H}}_2$ **query** $y \in \mathbb{Z}_\rho \times \mathbb{Z}_\rho$: If $y \notin \mathcal{Y}$ sample $b \leftarrow \mathbb{Z}_\mu \setminus \mathcal{B}$ and set $\hat{\mathcal{H}}_2(y) := b$. Output $\hat{\mathcal{H}}_2(y)$.

$\hat{\mathcal{H}}_3$ **query** $z \in \mathbb{Z}_\mu$: If $z \notin \mathcal{Z}$ sample $c \leftarrow \mathbb{Z}_\rho \times \mathbb{Z}_\rho$ and set $\hat{\mathcal{H}}_3(z) := c$. Output $\hat{\mathcal{H}}_3(z)$.

$\hat{\mathcal{H}}_1$ **query** $x \in \mathbb{Z}_\mu$: If $x \notin \mathcal{X}$ then

1. sample $(a_0, a_1) \leftarrow \mathbb{Z}_\rho \times \mathbb{Z}_\rho$ and set $\hat{\mathcal{H}}_1(x) := (a_0, a_1)$.
2. For all $(y_0, y_1) \in \mathcal{Y}$ where $a_1 = y_1$ and $x + \hat{\mathcal{H}}_2(y_0, y_1) \in \mathcal{Z}$ set $\mathsf{FAIL} := \mathsf{FAIL} + 1$.

Output $\hat{\mathcal{H}}_1(x)$.

Lemma 5.3 *The advantage of any q-query adversary A in winning the z-collision game on ϕ_{3F} (i.e. force $\mathsf{FAIL} > 0$) is at most*

$$\Pr[\mathsf{FAIL} > 0] \leq \mathsf{E}[Q(\mu, q)]/\rho$$

Proof. Consider the queries $\mathcal{X}, \mathcal{Y}, \mathcal{Z}$ made by A, and recall that $\mathcal{B} = \hat{\mathcal{H}}_2(\mathcal{Y})$. For any query $x \in \mathcal{X}$, let $\mathcal{B}_x = \{b \in \mathcal{B} \ : \ x + b \in \mathcal{Z}\}$. Note that $|\mathcal{B}_x|$ is the number of edges from x to $\mathcal{Z}$ in the Cayley graph $\mathcal{C}(\mu, q, \mathcal{B})$. The expected value by which this x query will increase FAIL is

$$|\mathcal{B}_x|/\rho.$$

To see this, note that for any $b \in \mathcal{B}$, the probability that FAIL wil increase because of this b is 0 if $x + b \notin \mathcal{Z}$ (equivalently $b \notin \mathcal{B}_x$), and $1/\rho$ otherwise. More precisely, FAIL will increase if the (a_0, a_1) we sample and the preimage (y_0, y_1) of b (i.e. $\hat{\mathcal{H}}_2(y_0, y_1) = b$) satisfy $a_1 = y_1$, and as a_1 is uniform, $\Pr[a_1 = y_1] = 1/\rho$. By definition $\sum_{x \in \mathcal{X}} |\mathcal{B}_x|$ is the number of edges of the $(\mathcal{X}, \mathcal{Z})$ subgraph of $\mathcal{C}(\mu, q, \mathcal{B})$, we have for a fixed $\mathcal{B}$

$$\mathsf{E}[\mathsf{FAIL}] \leq Q(\mu, q, \mathcal{B})/\rho$$

and as $\mathcal{B}$ is sampled uniformly at random

$$\mathsf{E}[\mathsf{FAIL}] = \mathsf{E}[Q(\mu, q)]/\rho.$$

■

5.3 Simulating the Four-Round Feistel

In this section we prove Theorem 4.4. Recall that $q_{\mathsf{A}} = q_{\mathcal{H}} + q_{\mathcal{F}}$ denotes total number of queries made by A. As in the proof of Theorem 4.1 for the two-round Feistel our simulator $\mathsf{S}^{\mathcal{F}}$ (given access to a random function $\mathcal{F} : \mathbb{Z}_\mu \times \mathbb{Z}_\rho \to \mathbb{Z}_\mu \times \mathbb{Z}_\rho \times \mathbb{Z}_\rho$) will define fake random oracles $\hat{\mathcal{H}}_i, i = 1, \ldots, 4$ by lazy sampling. The sets $\mathcal{X}, \mathcal{Y}, \mathcal{Z}, \mathcal{W}$ define the inputs on which $\hat{\mathcal{H}}_1, \ldots, \hat{\mathcal{H}}_4$ have already been defined. The sets $\mathcal{A}, \mathcal{B}, \mathcal{C}, \mathcal{D}$ are the corresponding outputs, e.g. $\mathcal{A} = \hat{\mathcal{H}}_1(\mathcal{X})$. The simulator also initializes variables $\mathsf{FAIL}_i := 0$ for $i \in \{0, 1, 2, 3, 4\}$ which will only be used in the proof. Informally, whenever the simulator cannot define the $\hat{\mathcal{H}}_i$'s consistently, it sets $\mathsf{FAIL}_j := 1$ (for which j depends on the reason why it fails) and aborts, by this we mean it just stops giving any more outputs.

We now define how $\mathsf{S}^{\mathcal{F}}$ answers the $q_{\mathcal{H}}$ queries to $\hat{\mathcal{H}}_i, i \in \{1, 2, 3, 4\}$ and updates its state on the $q_{\mathcal{F}}$ queries to $\mathcal{F}$.

- $\hat{\mathcal{H}}_2$ **query** $y \in \mathbb{Z}_\rho \times \mathbb{Z}_\rho$**:** If $y \notin \mathcal{Y}$ sample $b \leftarrow \mathbb{Z}_\mu$ and set $\hat{\mathcal{H}}_2(y) := b$. Output $\hat{\mathcal{H}}_2(y)$.
  If $x + b = z$ for some $x \in \mathcal{X}, z \in \mathcal{Z}$ set $\mathsf{FAIL}_0 := 1$ and abort.
- $\hat{\mathcal{H}}_4$ **query** $w \in \mathbb{Z}_\rho$**:** If $w \notin \mathcal{W}$ sample $d \leftarrow \mathbb{Z}_\mu$ and set $\hat{\mathcal{H}}_4(w) := d$. Output $\hat{\mathcal{H}}_4(w)$.
- $\hat{\mathcal{H}}_3$ **query** $z \in \mathbb{Z}_\mu$**:** If $z \notin \mathcal{Z}$ and there exist *two* distinct pairs of messages $x, (y_0, y_1)$ and $x', (y_0', y_1')$ in $\mathcal{X} \times \mathcal{Y}$ s.t. with $(a_0, a_1) = \hat{\mathcal{H}}_1(x), (a_0', a_1') = \hat{\mathcal{H}}_1(x')$
  1. $a_1 = y_1$ and $a_1' = y_1'$
  2. $z = \hat{\mathcal{H}}_2(y_0, y_1) + x = \hat{\mathcal{H}}_2(y_0', y_1') + x'$

  then set $\mathsf{FAIL}_3 := 1$ and abort. (Here we fail as z appears in two queries $(x, y_0 - a_0, 0)$ and $(x', y_0' - a_0', 0)$ to $\pi_{4F}^{\hat{\mathcal{H}}}$, and we won't be able to program $\hat{\mathcal{H}}_3(z)$ to be consistent with both).
  Otherwise, if $z \notin \mathcal{Z}$ and there exists *exactly one* pair $x, (y_0, y_1)$ where $a_1 = y_1$ and $z = \hat{\mathcal{H}}_2(y_0, y_1) + x$, try to program $\hat{\mathcal{H}}_3, \hat{\mathcal{H}}_4$ s.t. $\pi_{4F}^{\hat{\mathcal{H}}}(x, y_0 - a_0, 0) = \mathcal{F}(x, y_0 - a_0)$ as follows:
  1. Query $(f_0, f_1, f_2) \leftarrow \mathcal{F}(x, y_0 - a_0)$
  2. $\hat{\mathcal{H}}_3(z) := (f_1 - y_0, f_2 - y_1)$ (program $\hat{\mathcal{H}}_3(z)$)
  3. If $(f_1, f_2) \in \mathcal{W}$ set $\mathsf{FAIL}_4 := 1$ and abort (fail due to collision on w value)
  4. set $\hat{\mathcal{H}}_4(f_1, f_2) := f_0 - z$ (program $\hat{\mathcal{H}}_4(w)$)

  Otherwise, if $z \notin \mathcal{Z}$ and no such pair exists, sample $c \leftarrow \mathbb{Z}_\rho \times \mathbb{Z}_\rho$ and set $\hat{\mathcal{H}}_3(z) := c$.
  Output $\hat{\mathcal{H}}_3(z)$.
- $\hat{\mathcal{H}}_1$ **query** $x \in \mathbb{Z}_\mu$**:** If $x \in \mathcal{X}$ output $\hat{\mathcal{H}}_1(x)$.
  Otherwise, if $x \notin \mathcal{X}$ sample $(a_0, a_1) \leftarrow \mathbb{Z}_\rho \times \mathbb{Z}_\rho$, set $\hat{\mathcal{H}}_1(x) := (a_0, a_1)$, output $\hat{\mathcal{H}}_1(x)$.
  Then for all $(y_0, y_1) \in \mathcal{Y}$ where $a_1 = y_1$ try to program $\hat{\mathcal{H}}_3, \hat{\mathcal{H}}_4$ s.t. $\pi_{4F}^{\hat{\mathcal{H}}}(x, y_0 - a_0, 0) = \mathcal{F}(x, y_0 - a_0)$ as follows:
  1. Query $(f_0, f_1, f_2) \leftarrow \mathcal{F}(x, y_0 - a_0)$
  2. If $\hat{\mathcal{H}}_2(y_0, y_1) + x \in \mathcal{Z}$ set $\mathsf{FAIL}_1 := 1$ and abort (fail due to collision on z value)
  3. for $z = \hat{\mathcal{H}}_2(y_0, y_1) + x$ set $\hat{\mathcal{H}}_3(z) := (f_1 - y_0, f_2 - y_1)$ (program $\hat{\mathcal{H}}_3(z)$)
  4. If $(f_1, f_2) \in \mathcal{W}$ set $\mathsf{FAIL}_2 := 1$ and abort (fail due to collision on w value)
  5. set $\hat{\mathcal{H}}_4(f_1, f_2) := f_0 - z$ (program $\hat{\mathcal{H}}_4(w)$)
- $\mathcal{F}$ **query** $(x, v) \in \mathbb{Z}_\mu$**:** Try to program the $\hat{\mathcal{H}}_i$ s.t. $\pi_{4F}^{\hat{\mathcal{H}}}(x, v, 0) = \mathcal{F}(x, v)$ as follows: query $(a_0, a_1) \leftarrow \hat{\mathcal{H}}_1(x)$ (as described above), then query $b \leftarrow \hat{\mathcal{H}}_2(v + a_0, a_1)$ and finally $(c_0, c_1) \leftarrow \hat{\mathcal{H}}_3(x + b)$ (note that we don't have to query $\hat{\mathcal{H}}_4$ explicitly as the $\hat{\mathcal{H}}_3$ query already programs $\hat{\mathcal{H}}_4$.)

We will bound the probability that $\mathsf{FAIL}_i = 1$ for $i \in \{0, 1, 2, 3, 4\}$, and then can use standard arguments show that the advantage of distinguishing $\mathcal{F}, \mathsf{S}^{\mathcal{F}}$ from $\pi^{\mathcal{H}}, \mathcal{H}$ is upper bounded by the sum of these probabilities.

We start with bounding the sizes of $\mathcal{X}, \mathcal{Y}, \mathcal{Z}, \mathcal{W}$ which will give the same upper bounds on $\mathcal{A}, \mathcal{B}, \mathcal{C}$ and $\mathcal{D}$, respectively. As $\mathcal{X}$ and $\mathcal{Y}$ can only increase by at most 1 on a $\hat{\mathcal{H}}_1, \hat{\mathcal{H}}_2$ or $\mathcal{F}$ query,

we have

$$|\mathcal{X}|\ ,\ |\mathcal{Y}| \leq q_{\mathsf{A}} \tag{11}$$

Bounding $|\mathcal{Z}|$ and $|\mathcal{W}|$ is less trivial, as a $\hat{\mathcal{H}}_1$ query can increase $|\mathcal{Z}|$ and $|\mathcal{W}|$ by as much as $|\mathcal{Y}|$, and thus we only get a q_{A}^2 upper bound. Fortunately the *expected* increase of $|\mathcal{Z}|, |\mathcal{W}|$ on a query x to $\hat{\mathcal{H}}_1$ is only $|\mathcal{Y}|/\rho \leq q_{\mathsf{A}}/\rho \leq 1$ as $(y_0, y_1) \in \mathcal{Y}$ will increase $|\mathcal{Z}|, |\mathcal{W}|$ if $y_1 = a_1$ for the randomly chosen $a_1 \in \mathbb{Z}_\rho$. Moreover a $\hat{\mathcal{H}}_3$ or $\hat{\mathcal{H}}_4$ query can increase $|\mathcal{Z}|$ and $|\mathcal{W}|$ by at most 1. We get a bound on the expected size of $\mathcal{Z}, \mathcal{W}$ of (we have a factor 2 below, as an $\mathcal{F}$ query invokes a $\hat{\mathcal{H}}_1$ and $\hat{\mathcal{H}}_3$ query.)

$$\mathsf{E}[|\mathcal{Z}|]\ ,\ \mathsf{E}[|\mathcal{W}|] \leq 2q_{\mathsf{A}} \tag{12}$$

Below, we will also need an upper bound on $\mathsf{E}[|\mathcal{W}|^2]$. As mentioned, the expected increase of $|\mathcal{W}|$ with every x query is $|\mathcal{Y}|/\rho$, the case which maximizes $\mathsf{E}[|\mathcal{W}|^2]$ is when this increase is either $|\mathcal{Y}|$ (with probability ρ^{-1}) or 0, as this maximizes the variance of $|\mathcal{W}|$, and thus also the expectation $\mathsf{E}[|\mathcal{W}|^2]$. The probability that we increase by $|\mathcal{Y}|$ more than t times is at most $(q_{\mathsf{A}}/\rho)^t$. Setting $t := -\log\rho/\log(q_{\mathsf{A}}/\rho) = \log\rho/\log(\rho/q_{\mathsf{A}})$ this is $(q_{\mathsf{A}}/\rho)^t = 2^{-\log\rho} = 1/\rho$ we get $\Pr\left[|\mathcal{W}| \geq q_{\mathsf{A}} \cdot \frac{\log\rho}{\log(\rho/q_{\mathsf{A}})}\right] \leq 1/\rho$. The same bound holds for $|\mathcal{Z}|$, and from now on we'll assume

$$|\mathcal{Z}|\ ,\ |\mathcal{W}| \leq q_{\mathsf{A}} \cdot \log\rho/\log(\rho/q_{\mathsf{A}}) \tag{13}$$

We can safely ignore the tiny $1/\rho$ probability that this fails to hold.

We will now bound the probability that $\mathsf{FAIL}_0 := 1$ will be set in any of the queries to $\hat{\mathcal{H}}_2$. There are at most $|\mathcal{X}||\mathcal{Z}|$ possible $b \in \mathbb{Z}_\mu$ s.t. $x + b = z$ for some $x \in \mathcal{X}, z \in \mathcal{Z}$, thus a random b will fall in this set with probability at most $|\mathcal{X}||\mathcal{Z}|/\mu$. Taking the union bound over all $\leq q_{\mathsf{A}}$ queries to $\hat{\mathcal{H}}_2$

$$\Pr[\mathsf{FAIL}_0 = 1] \leq \frac{q_{\mathsf{A}}\mathsf{E}[|\mathcal{X}||\mathcal{Z}|]}{\mu} \leq \frac{q_{\mathsf{A}}^2\mathsf{E}[|\mathcal{Z}|]}{\mu} \leq \frac{2q_{\mathsf{A}}^3}{\mu} \tag{14}$$

Next, we will bound the probability that $\mathsf{FAIL}_2 := 1$ will be set in any of the at most q_{A} queries to $\hat{\mathcal{H}}_1$. We set $\mathsf{FAIL}_2 := 1$ if the uniformly random (f_1, f_2) falls into the set $\mathcal{W}$, which happens with probability $|\mathcal{W}|/\rho^2$. Taking the union bound over the at most $|\mathcal{W}|$ times we will go into step 4. of the $\hat{\mathcal{H}}_1$ query and (13) in the second step

$$\Pr[\mathsf{FAIL}_2 = 1] \leq \frac{\mathsf{E}[|\mathcal{W}|^2]}{\rho^2} \leq \frac{q_{\mathsf{A}}^2}{\rho^2} \cdot \left(\frac{\log\rho}{\log(\rho/q_{\mathsf{A}})}\right)^2 \tag{15}$$

We get the same bound with the same argument for $\Pr[\mathsf{FAIL}_4 = 1]$.

To bound the probability that $\mathsf{FAIL}_1 := 1$ will be set, we observe that from an adversary A who manages to set $\mathsf{FAIL} := 1$ when interacting with $\mathcal{F}, \mathsf{S}^{\mathcal{F}}$, we can get an adversary $\tilde{\mathsf{A}}$ which wins the z-collision game on ϕ_{3F} with at least the same probability. Using this observation with the bound from Lemma 5.3 and the upper bound on $|\mathcal{Z}|$ as given (13) would give $\Pr[\mathsf{FAIL}_1 = 1] \leq \mathsf{E}[Q(\mu, q_{\mathsf{A}}\log\rho/\log(\rho/q_{\mathsf{A}}))]/\rho$. We can get a better bound using $\mathsf{E}[|\mathcal{Z}|] \leq 2q_{\mathsf{A}}$ and the fact that for any $c \geq 1$, $Q(\mu, q)$ increases by at most a factor c if we allow the $\mathcal{Z}$ in (9) to have size cq instead of q (this follows from the maximality of $\mathcal{Z}$.) In particular this gives

$$\Pr[\mathsf{FAIL}_1 = 1] \leq 2\mathsf{E}[Q(\mu, q_{\mathsf{A}})]/\rho \tag{16}$$

We will now bound $\Pr[\mathsf{FAIL}_3 = 1]$. To have $\mathsf{FAIL}_3 = 1$ it must be the case that at some point the adversary makes a fresh query $y \notin \mathcal{Y}$ s.t. the random output $\hat{\mathcal{H}}_3(y) = b$ satisfies $x + b = x' + b'$ for some $(x, x', b') \in \mathcal{X} \times \mathcal{X} \times \mathcal{B}$ (note that if the x query was made after the y query, then $\hat{\mathcal{H}}(z)$ would already be defined). As there are at most $|\mathcal{X}|^2|\mathcal{Y}| \leq q_{\mathsf{A}}^3$ triples, (x', b', x), each giving rise to

one possible "target" $b = x' + b' - x$, the probability to hit any of them in the at most q_{A} queries is at most

$$\Pr[\mathsf{FAIL}_3 = 1] \leq \frac{q_{\mathsf{A}}^4}{\mu}$$

Finally, we can bound

$$\begin{aligned} & |\Pr[\mathsf{A}^{\pi^{\mathcal{H}},\mathcal{H}}(1^n) = 1] - \Pr[\mathsf{A}^{\mathcal{F},\mathsf{S}^{\mathcal{F}}}(1^n) = 1]| \\ \leq & \sum_{i=0}^{4} \Pr[\mathsf{FAIL}_i = 1] \qquad (17) \\ \leq & \frac{2\mathsf{E}[Q(\mu, q_{\mathsf{A}})]}{\rho} + \frac{2q_{\mathsf{A}}^4}{\mu} + \frac{2q_{\mathsf{A}}^2}{\rho^2} \cdot \left(\frac{\log \rho}{\log(\rho/q_{\mathsf{A}})}\right)^2 \end{aligned}$$

using standard arguments like in the proof of Theorem 4.1. ■

6 Size of Subgraphs of Random Cayley Graphs

In this section we shall give a tail estimate for the random variable $Q(\mu, q)$ introduced in Section 5.1. Recall that $Q(\mu, q)$ ranges over randomly chosen $\mathcal{B} \subset \mathbb{Z}_\mu, |\mathcal{B}| = q$, and takes value

$$Q(\mu, q, \mathcal{B}) = \max_{\mathcal{X},\mathcal{Z} \subset \mathbb{Z}_\mu, |\mathcal{X}| = |\mathcal{Z}| = q} |\{(x, z) \mid x \in \mathcal{X}, z \in \mathcal{Z}, z - x \in \mathcal{B}\}| \qquad (18)$$

Theorem 6.1 *For $0 < a < 1/2$ and $\alpha > 2a^2$, there is some $\beta > 0$ such that*

$$\Pr[Q(\mu, \mu^a) \geq \mu^{a+\alpha}] \leq O(\exp(-\mu^\beta))$$

Intuition. The proof of Lemma 6.1 is by a compression argument. We will show that a set $\mathcal{B}$ satisfying $Q(\mu, q, \mathcal{B}) \geq \mu^{a+\alpha}$ has a lot of constant size linear relation between its elements, which allows us to describe $\mathcal{B}$ with less than $\log \binom{\mu}{\mu^a}$ bits (the latter is the number of bits needed to describe $\mathcal{B}$ without the condition). The number of bits saved will be μ^β for some $\beta > 0$, implying the bound in Lemma 6.1 on the probability. In fact, we are going to encode only a subset $\mathcal{D} \subseteq \mathcal{B}$, $|\mathcal{D}| = \mu^\gamma$ more efficiently than usual, while we encode the rest, $\overline{\mathcal{D}} = \mathcal{B} \setminus \mathcal{D}$, in the usual way.

Assume that $\overline{\mathcal{D}}$ is already given, and there are some fixed $x, z \in \mathbb{Z}_\mu$ such that all elements $b \in \mathcal{D}$ can be expressed as $b = z - x - \epsilon_1 b_1 - \ldots - \epsilon_l b_l$, where $b_1, \ldots, b_l \in \overline{\mathcal{D}}$ and $\epsilon_1, \ldots, \epsilon_l \in \{1, -1\}$. Then, after the minimal overhead of specifying z and x (once for the entire $\mathcal{D}$), we can describe b with only $l \times (\log |\overline{\mathcal{D}}| + 1)$ bits instead of $(1 - a) \log \mu$ bits. This gives us an advantage, when $l \leq \frac{1-a}{2a}$, since $\log |\overline{\mathcal{D}}| \leq \log |\overline{\mathcal{B}}| = a \log \mu$. The gain will be $|\mathcal{D}| \frac{1-a}{2} (\log \mu - \frac{1}{a})$ bits.

Proof. Consider a $\mathcal{B}$ that satisfies $Q(\mu, q, \mathcal{B}) \geq \mu^{a+\alpha}$ via $\mathcal{X}, \mathcal{Z} \subseteq \mathbb{Z}_\mu$ (there may be numerous such set-pair certificates; we pick one). Let G be the bipartite graph with bipartition $\mathcal{X}, \mathcal{Z}$ and edge set

$$e(G) = \{(x, z) \mid \ x \in \mathcal{X}, z \in \mathcal{Z}, z - x \in \mathcal{B}\}.$$

It is easy to see that G has an induced subgraph G' of G such that G' still has $\Omega(n^{a+\alpha})$ edges, moreover all degrees are at least $n^\alpha/4$. Consider any point $x \in \mathcal{X} \cap V(G')$. For our proof the following definition is crucial:

Definition 6.2 *Let P_i for $i = 1, 2, \ldots$ be the set of all those paths π of length i that satisfy:*

1. *π starts at x*

2. *No two edges edges of π have identical labels, where a label of an edge (x, z) is by definition $z - x$.*

Note that every path in P_i has at most i continuations that are not in P_{i+1}. Thus the overwhelming majority of paths from x of length i belongs to P_i. In fact, because every degree in G' is at least $\mu^\alpha/4$, the number of different path in P_i is at least

$$\prod_{j=0}^{i-1}\left(\frac{\mu^\alpha}{4} - j\right) > \frac{1}{2}\frac{\mu^{i\alpha}}{4^i}$$

if μ is sufficiently large. Set i to be $a/\alpha + 1$, assuming this is an integer. (If not, we set it to $\lceil a/\alpha + 1\rceil$.) So there must be a $z \in \mathcal{X}$, if $i = a/\alpha + 1$ is even, or $z \in \mathcal{Z}$, if $i = a/\alpha + 1$ is odd, such that at least $\frac{1}{2}\frac{\mu^\alpha}{4^i}$ paths from P_i end in z (since $|\mathcal{X}|, |\mathcal{Z}| \le \mu^a$). Let T be the set of the paths that end in this z. For a path π let $\ell(\pi)$ denote the set of labels that occur on its edges. Notice that the number of path π for which $\ell(\pi)$ is a given fixed set of i different labels is at most $i!$, a constant. This implies that the size of the set system

$$\mathcal{T} = \{\ell(\pi) \mid \pi \in T\}$$

is at least $\frac{1}{i!}\frac{1}{2}\frac{\mu^\alpha}{4^i}$, which further implies that its basic set (i.e. $\cup_{\ell(\pi)\in\mathcal{T}}\ell(\pi)$) is of size at least $\left(\frac{1}{i!}\frac{1}{2}\frac{\mu^\alpha}{4^i}\right)^{1/i} = \Omega(\mu^{\alpha/i})$. Let $\mathcal{D}_0$ be this basic set. We are going to define a subdivision $\mathcal{B} = \mathcal{D} \cup \overline{\mathcal{D}}$ with the property that $|\mathcal{D}| = \Omega(\mu^{\alpha/i})$ and every element in $\mathcal{D}$ can be written as $b = z - x - \epsilon_1 b_1 - \ldots - \epsilon_{i-1}b_{i-1}$, where $b_1, \ldots, b_{i-1} \in \overline{\mathcal{D}}$, $\epsilon_1, \ldots, \epsilon_{i-1} \in \{1, -1\}$, and x and z are the fixed starting- and end-point of the paths in T. First select every element of $\mathcal{B}$ that are not in $\mathcal{D}_0$ into $\overline{\mathcal{D}}$. The existence of the desired $\mathcal{D}$ is now going to be guaranteed by a probabilistic construction. Recall that in a probabilistic construction all we have to show is that the desired event occurs with non-zero probability. We throw every element of $\mathcal{D}_0$ with probability $1 - 1/i$ also into $\overline{\mathcal{D}}$. For every element $b \in \mathcal{D}_0$ there is a $\pi \in T$ such that $b \in \ell(\pi)$ (we designate this π to b *before* doing the random drawing with the understanding that there can be more bs whom we designate the same π). The probability that all elements of $\ell(\pi)$ with the exception of b is thrown into $\overline{\mathcal{D}}$ is $\frac{1}{i}\left(1 - \frac{1}{i}\right)^{i-1} > \frac{1}{4i}$. We put every such b into $\mathcal{D}$, and all the rest into $\overline{\mathcal{D}}$. Notice that now every such b is of the form

$$b = z - x - \epsilon_1 b_1 - \ldots - \epsilon_{i-1}b_{i-1},$$

where $b_1, \ldots, b_{i-1}$ are the labels on the path $\ell(\pi)$ other than b. This probabilistic construction shows that there is choice for $\mathcal{D}$ of size at least $\frac{1}{4i}|\mathcal{D}_0|$ with the desired property.

In summary, what we have proven:

Lemma 6.3 *If $Q(\mu, q, \mathcal{B}) \ge \mu^{a+\alpha}$ there is a subdivision $\mathcal{D}, \overline{\mathcal{D}}$ of $\mathcal{B}$ such that $|\mathcal{D}| = c\mu^{\alpha/i}$ for $i = a/\alpha + 1$ and for some constant $c = c(a)$ such, that every element in $\mathcal{D}$ can be written as $b = z - x - \epsilon_1 b_1 - \ldots - \epsilon_{i-1}b_{i-1}$ for some $b_1, \ldots, b_{i-1} \in \overline{\mathcal{D}}$, $\epsilon_1, \ldots, \epsilon_{i-1} \in \{1, -1\}$ with fixed z and x.*

From the lemma we get that the number of $\mathcal{B}$s for which $Q(\mu, q, \mathcal{B}) \ge \mu^{a+\alpha}$ is at most

$$\mu^2\binom{\mu}{\mu^a - c\mu^{\alpha/i}} \times \left((2\mu^a)^{a/\alpha}\right)^{c\mu^{\alpha/i}}.$$

Here μ^2 comes from specifying x and z; $\binom{\mu}{\mu^a - c\mu^{\alpha/i}}$ comes from specifying $\overline{\mathcal{D}}$; $2\mu^a$ in the second term comes from specifying ϵ_j and b_j, where $1 \leq j \leq a/\alpha = i - 1$, and the outer exponent of the second term comes from that we have to do this for all $b \in \mathcal{D}$. The expression can be estimated from above as

$$\mu^2 \binom{\mu}{\mu^a} \times \frac{2\mu^{ac\mu^{\alpha/i}}}{\mu^{c\mu^{\alpha/i}}} \times \left((2\mu^a)^{a/\alpha}\right)^{c\mu^{\alpha/i}}.$$

This is significantly smaller than $\binom{\mu}{\mu^a}$ when $2a^2 < \alpha$.

References

[1] N. Alon, T. Kaufman, M. Krivelevich, and D. Ron. Testing triangle-freeness in general graphs. *SIAM J. Discrete Math.*, 22(2):786–819, 2008. 4

[2] M. Bellare and T. Ristenpart. Simulation without the artificial abort: Simplified proof and improved concrete security for Waters' IBE scheme. In A. Joux, editor, *EUROCRYPT 2009*, volume 5479 of *LNCS*, pages 407–424. Springer, Apr. 2009. 1

[3] M. Bellare and P. Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. In V. Ashby, editor, *ACM CCS 93*, pages 62–73. ACM Press, Nov. 1993. 2, 4

[4] M. Bellare and P. Rogaway. The exact security of digital signatures: How to sign with RSA and Rabin. In U. M. Maurer, editor, *EUROCRYPT'96*, volume 1070 of *LNCS*, pages 399–416. Springer, May 1996. 2, 3, 10

[5] D. Boneh, B. Lynn, and H. Shacham. Short signatures from the Weil pairing. In C. Boyd, editor, *ASIACRYPT 2001*, volume 2248 of *LNCS*, pages 514–532. Springer, Dec. 2001. 2

[6] J.-S. Coron. On the exact security of full domain hash. In M. Bellare, editor, *CRYPTO 2000*, volume 1880 of *LNCS*, pages 229–235. Springer, Aug. 2000. 7

[7] J.-S. Coron. Optimal security proofs for PSS and other signature schemes. In L. R. Knudsen, editor, *EUROCRYPT 2002*, volume 2332 of *LNCS*, pages 272–287. Springer, Apr. / May 2002. 2

[8] S. Goldwasser, S. Micali, and R. L. Rivest. A digital signature scheme secure against adaptive chosen-message attacks. *SIAM Journal on Computing*, 17(2):281–308, Apr. 1988. 4

[9] S. A. Kakvi and E. Kiltz. Optimal security proofs for full domain hash, revisited. In D. Pointcheval and T. Johansson, editors, *EUROCRYPT 2012*, volume 7237 of *LNCS*, pages 537–553. Springer, Apr. 2012. 2, 3, 5, 6, 11

[10] J. Katz and N. Wang. Efficiency improvements for signature schemes with tight security reductions. In S. Jajodia, V. Atluri, and T. Jaeger, editors, *ACM CCS 03*, pages 155–164. ACM Press, Oct. 2003. 7

[11] A. Naor and J. Verstraëte. A note on bipartite graphs without 2k-cycles. *Comb. Probab. Comput.*, 14(5-6):845–849, Nov. 2005. 14

[12] P. Paillier. Public-key cryptosystems based on composite degree residuosity classes. In J. Stern, editor, *EUROCRYPT'99*, volume 1592 of *LNCS*, pages 223–238. Springer, May 1999. 5

[13] C. Peikert and B. Waters. Lossy trapdoor functions and their applications. In R. E. Ladner and C. Dwork, editors, *40th ACM STOC*, pages 187–196. ACM Press, May 2008. 3, 5