Post-quantum security of cryptographic transformations in the random oracle model

Huang, Y.

Chapter 3

Fiat-Shamir and Hash-and-Sign with Aborts

3.1 Introduction

Fiat-Shamir-with-aborts (FSwA) and *probabilistic hash-and-sign with retry/abort* (HSwA) are important design principles for digital signature schemes (in the random oracle model), in particular for constructing quantum-secure signature schemes. Both have in common that the signature generation may require several trials until a "good" signature is obtained; informally speaking, the retrying is typically necessary in order to not leak unwanted information about the secret key via a "bad" choice of the signature. Examples of signature schemes that follow one or the other are Lyubashevsky's signature [Lyu09, Lyu12], GLP [GLP12], TESLA [ABB+17], Dilithium [DKL+18], SeaSign [DFG19a], and HAETAE [CCD+24] in the FSwA paradigm, and Hidden Field Equation (HFE) signatures [Pat96], Unbalanced Oil and Vinegar (UOV) [KPG99], the Courtois-Finiasz-Sendrier (CFS) signature [CFS01, Dal08], GeMSS [CFMR+17], Wave [DST19], MAYO [Beu21], and QR-UOV [FIKT21] in the HSwA paradigm.

The two design principles, FSwA and HSwA, also have in common that security is typically proven in two steps: first, the specific instantiation is exploited in order to show UF-NMA security, and then an argument that is generic for the design principle (and only requires some mild additional properties from the instantiation) is used to conclude full-fledged (strong or ordinary) UF-CMA security, i.e., security against chosen-message attacks.

Quite recently, a subtle but crucial flaw was independently and concurrently discovered by [BBD+23] and [DFPS23] in all prior UF-CMA-to-UF-NMA reductions of FSwA. The flaw applies both classically and quantum, and invalidates all prior security proofs of FSwA signature schemes, most notably leaving the future NIST PQC standard, Dilithium, without a valid proof. Moving on to

HSwA signature schemes, the authors of [KX24] followed up on the observation from [CDP23] that the original security reduction of HSwA in [SSH11] contains a similar flaw, and they provide the first correct such reduction in the ROM, covering both classical and quantum attacks. However, as will be seen later, their quantum analysis suffers a rather non-ideal security loss.

3.1.1 Our Contribution

Facing the aforementioned flaw, we re-establish the security analyses of FSwA and HSwA signature schemes in this chapter, to the extent possible. Specifically, we consider a unified, abstract class of digital signature schemes, which we call *generalized Fiat-Shamir with aborts* signature schemes, covering both FSwA and HSwA (see Section 3.2.1). We then describe the flaw in Section 3.3 under this abstract formalism, and provide a new generic UF-CMA-to-UF-NMA reduction, accompanied with various bounds that covers both classical and quantum attacks in the random oracle model.

A new security proof. In Section 3.4, we provide a new security proof in the ROM from scratch, reducing the UF-CMA security of a generalized FSwA scheme $\mathcal{S}$ to the UF-NMA security. On a very high level, our reduction relies on a simulator Sim that can efficiently simulate the signing procedure Sign without knowing the secret key. This may seem contradictory to the very definition of the UF-CMA security, but the catch here is that, as a part of our reduction, Sim is allowed to reprogram the random oracle $H(r, m) := y$, once it produces a suitable simulated transcript (r, y, z). With such a simulator in place, any UF-CMA attacker $\mathcal{A}$ can be transformed into a similarly efficient UF-NMA attacker $\mathcal{B}$, via forwarding each signing queries made by $\mathcal{A}$ to the simulator.

For this approach to work, it is sufficient to show that Sign and Sim are indistinguishable, even if one can query the random oracle H. Moreover, the UF-CMA-to-UF-NMA security loss is roughly upperbounded by the distinguishing advantage, i.e.

$$\mathbf{Adv}^{\mathsf{UF\text{-}CMA}}_{\mathcal{S}}(\mathcal{A}) - \mathbf{Adv}^{\mathsf{UF\text{-}NMA}}_{\mathcal{S}}(\mathcal{B}) \leq SD(\mathcal{A}^{H,\mathsf{Sign}}, \mathcal{A}^{H,\mathsf{Sim}}) \ ,$$

assuming $\mathcal{A}$ makes a few more queries for technical reasons. To prove this indistinguishability, our analyses proceed through a sophisticated hybrid argument, detailed in Section 3.4.2, with two additional intermediate oracles Prog and Trans come into the play. The main technical challenge is to show that $\mathcal{A}^{H,\mathsf{Sign}} \approx \mathcal{A}^{H,\mathsf{Prog}} \approx \mathcal{A}^{H,\mathsf{Trans}}$, while the closeness $\mathcal{A}^{H,\mathsf{Trans}} \approx \mathcal{A}^{H,\mathsf{Sim}}$ follows straightforwardly with advantage bounded by $q_S\zeta$, where q_S is the number of signing queries $\mathcal{A}$ can make, and ζ is a scheme-dependent parameter that is close to zero.

In the general case where $\mathcal{A}$ is allowed to make quantum queries, we obtain

$SD\left(\mathcal{A}^{H,\mathsf{Sign}}, \mathcal{A}^{H,\mathsf{Prog}}\right) \leq O\left(\frac{q_S\sqrt{q_H\epsilon}}{1-p}\right)$ and $SD\left(\mathcal{A}^{H,\mathsf{Prog}}, \mathcal{A}^{H,\mathsf{Trans}}\right) \leq O\left(q_H\sqrt{\frac{q_S\epsilon}{1-p}}\right)$, where q_H is the number of $\mathcal{A}$'s queries to the random oracle H, and p and ϵ are scheme-dependent parameters with $p \in [0,1)$ being not too close to 1, and ϵ being close-to-zero.[1] This leads to the following security loss

$$\mathbf{Adv}^{\mathsf{UF\text{-}CMA}}_{\mathcal{S}}(\mathcal{A}) - \mathbf{Adv}^{\mathsf{UF\text{-}NMA}}_{\mathcal{S}}(\mathcal{B}) \leq O\left(q_H\sqrt{\frac{q_S\epsilon}{1-p}} + \frac{q_S}{1-p}\sqrt{q_H\epsilon}\right) + q_S\zeta\ .$$

In the case where $\mathcal{A}$ is restricted to making classical queries only, via replacing relevant parts of the quantum analysis with a classical bound, we obtain a correspondingly tighter security loss

$$\mathbf{Adv}^{\mathsf{UF\text{-}CMA}}_{\mathcal{S}}(\mathcal{A}) - \mathbf{Adv}^{\mathsf{UF\text{-}NMA}}_{\mathcal{S}}(\mathcal{B}) \leq O\left(\frac{q_Hq_S\epsilon}{1-p} + \frac{q_S^2\epsilon}{(1-p)^2}\right) + q_S\zeta\ .$$

We note that, typically q_H is much bigger than q_S, e.g. in the NIST-2 security level, (q_S, q_H) can be all the way up to $(2^{64}, 2^{128})$. Therefore, the above quantum and classical bounds are dominated by $O\left(q_H\sqrt{\frac{q_S}{1-p}\epsilon}\right)$ and $O\left(\frac{q_Hq_S\epsilon}{1-p}\right)$ respectively (assuming $q_S\zeta$ is not too big). For the classical bound there is a matching attack where $SD(\mathcal{A}^{H,\mathsf{Sign}}, \mathcal{A}^{H,\mathsf{Sim}}) \geq \Omega(q_Hq_S\epsilon)$. However, (in the case where $q_S = 1$)[2] the best known quantum attack as presented in [GHHM21] only yields $SD(\mathcal{A}^{H,\mathsf{Sign}}, \mathcal{A}^{H,\mathsf{Sim}}) \geq \Omega(\sqrt{q_H\epsilon})$, with the q_H term inside of the square-root, suggesting that the quantum bound as stated above may still be suboptimal. This seemingly suboptimal loss also appears in prior analyses of HSwA [KX24] in the quantum setting.

Improving the quantum analysis. In Section 3.5, we further improve the hybrid step $\mathcal{A}^{H,\mathsf{Prog}} \approx \mathcal{A}^{H,\mathsf{Trans}}$ to $SD(\mathcal{A}^{H,\mathsf{Prog}}, \mathcal{A}^{H,\mathsf{Trans}}) \leq O\left(\frac{q_S}{1-p}\sqrt{q_H\epsilon}\right)$, which then leads to a tighter overall security loss in the quantum setting:

$$\mathbf{Adv}^{\mathsf{UF\text{-}CMA}}(\mathcal{A}) - \mathbf{Adv}^{\mathsf{UF\text{-}NMA}}(\mathcal{B}) \leq O\left(\frac{q_S}{1-p}\sqrt{q_H\epsilon}\right) + q_S\zeta\ .$$

Plugging in the numbers with $(q_S, q_H) = (2^{64}, 2^{128})$, and suppressing less relevant terms, we obtain $q_S\sqrt{q_H\epsilon} = 2^{128}\sqrt{\epsilon}$, an improvement against $q_H\sqrt{q_S\epsilon} = 2^{160}\sqrt{\epsilon}$ by a multiplicative factor of 2^{32}. This improved quantum bound as well as the above classical bound, matches the well-studied analyses of the original Fiat-Shamir signature schemes (without aborts).[3] Though it remains

[1] As a matter of fact, we allow these parameters to be key-dependent, which is omitted in this introduction for simplicity.

[2] The situation when q_S scales up remains unclear.

[3] In the atypical regime where q_S is much bigger than q_H, our quantum bound of order $O(q_S\sqrt{q_H\epsilon})$ even outperforms the standard bound of order $O(q_S\sqrt{(q_S+q_H)\epsilon})$.

open whether the quantum bound is optimal, any further improvement would also need to carry over to the better-understood setting without aborts.

At the core of our improvement is the following technical challenge (somewhat simplified here for the ease of exposition). Let $\mathcal{D}$ be a distribution over a set $\mathcal{R}$ with the promise that $\Pr[r = r_\circ] \leq \epsilon$ for any $r_\circ \in \mathcal{R}$ and $r \leftarrow \mathcal{D}$, and let f be an arbitrary (randomized or deterministic) function with domain $\mathcal{R} \times \mathcal{Y}$ and with a special symbol $\perp$ in its co-domain. Consider an arbitrary quantum algorithm $\mathcal{A}^H$ that gets a sample produced by one or the other of the following two procedures:

1: $r \leftarrow \mathcal{D}$		1: $r \leftarrow \mathcal{D}$
2: $H(r) := y \leftarrow \mathcal{Y}$		2: $y \leftarrow \mathcal{Y}$
3: $z \leftarrow f(r, y)$		3: $z \leftarrow f(r, y)$
4:	or	4: **if** $z \neq \perp$ **then** $H(r) := y$
5: **if** $z = \perp$ **then return** $\perp$		5: **if** $z = \perp$ **then return** $\perp$
6: **else return** (r, z)		6: **else return** (r, z)

and that can make superposition queries to the random oracle H *before* and *after* it gets the sample, say q_H in total. The goal now is to show that it is hard for $\mathcal{A}^H$ to decide from which of the two it got the sample.

We note that the only difference between the two is that the first procedure reprograms $H(r)$ to y no matter what, while the latter does so only in case of a non-$\perp$ output. Thus, intuitively it is clear that it is hard for $\mathcal{A}^H$ to distinguish the two: it can notice the difference only when the procedure outputs $\perp$ *and* $\mathcal{A}^H$ queries $H(r)$ after having received the sample (thus $\perp$); but the latter is unlikely to happen then due to the high entropy in r. However, making this a rigorous argument in the case of quantum queries results in a hybrid argument over all the queries to H (after having received the sample), where (for the sake of the argument) one would then measure for each query if the query is to r or not. This then leads to a distinguishing advantage of $q_H\sqrt{\epsilon}$, where the square-root comes from the Gentle-Measurement Lemma, used to argue that the measurements cause little disturbance, and the factor q_H is by quantifying over all queries. Thus, the real challenge is to prove a bound on the distinguishing advantage that scales as $\sqrt{q_H\epsilon}$ instead; this is what we aim for and achieve in this work.[4]

Concrete analyses of Dilithium. In Section 3.6, we perform a more elaborated concrete analysis on the NIST PQC standard Dilithium, where the parameter ϵ is better controlled. This is achieved both via a computer-aided

[4]One might also be tempted to argue that the two can only be distinguished by $\mathcal{A}^H$ if $\mathcal{A}^H$ has queried $H(r)$ *before* it receives the sample (and then use compressed-oracle techniques to get the q_H inside the square-root) — but then one falls for the same trap as earlier, faulty FSwA proofs: due to the *conditional* reprogramming of H in the second procedure, H may become non-uniformly random there. One expects this non-uniformity to be negligible and hard to notice for $\mathcal{A}^H$, but giving a concrete (and sufficiently good) bound appears difficult.

approach that yields better numerical results, and via a pen-and-paper version that is easier to verify, but with a slightly worse concrete bound.

The technical challenge that we address in order to control ϵ for Dilithium, is a rather innocent-looking mathematical problem. We consider a block diagonal square matrix $A^{\square} := D_1 \oplus \cdots \oplus D_n$ with each block $D_i \leftarrow \mathbb{F}_q^{\ell \times \ell}$ being an ℓ-by-ℓ random matrix over a finite field of order q, where concretely speaking $q \approx 2^{23}$ and $n\ell \in (1000, 2000)$. The goal here is to show that, with a cryptographically close-to-1 probability, the rank of $A^{\square}$ is very close to $n\ell$. We note that this is not as straightforward as it may seem. For instance, one may be tempted to argue that $A^{\square}$ is invertible with a probability close to 1, but this probability is in fact roughly $1 - n/q \approx 1 - n \cdot 2^{-23}$, which is not cryptographically close to 1.

3.2 Preliminaries

3.2.1 Generalized Fiat-Shamir with Aborts Signatures

Let $\mathcal{M}, \mathcal{R}, \mathcal{Y}$ and $\mathcal{Z}$ be arbitrary non-empty finite sets, and fix the domain and range of the random oracle to be $\mathcal{R} \times \mathcal{M}$ and $\mathcal{Y}$, respectively, i.e. $H : \mathcal{R} \times \mathcal{M} \to \mathcal{Y}$. Furthermore, let $\perp$ be some special symbol not contained in $\mathcal{Z}$. These sets may depend on a security parameter, but we leave this dependency — and the security parameter itself — implicit throughout most of the document.

We consider signature schemes $\left(\mathsf{KGen}, \mathsf{Sign}^H, \mathsf{Vrfy}^H\right)$ in the random oracle model of the following form. On input the security parameter, the key-generation algorithm KGen produces a key pair $(\mathsf{sk}, \mathsf{pk})$, which in turn specifies a distribution $\mathcal{D}$ over a set $\mathcal{R}$, an ensemble $\{f(r, y)\}_{r \in \mathcal{R}, y \in \mathcal{Y}}$ of distributions over $\mathcal{Z} \cup \{\perp\}$, and a predicate $\mathcal{V} : \mathcal{R} \times \mathcal{Y} \times \mathcal{Z} \to \{0, 1\}$. Signing of a message $m \in \mathcal{M}$ works as specified in Fig. 3.1 below, and the verification Vrfy, given a claimed signature $\sigma = (r, z)$ for a message $m \in \mathcal{M}$, accept if and only if $\mathcal{V}(r, H(r, m), z) = 1$.

Remark 3.1. *How the predicate $\mathcal{V}(r, y, z)$ works is irrelevant for this chapter, but a natural way is to check that z falls in the support of the distribution $f(r, y)$.*

$\mathsf{Sign}^H(\mathsf{sk}, m)$:
1: **repeat**
2: $\quad r \leftarrow \mathcal{D}$
3: $\quad y := H(r, m)$
4: $\quad z \leftarrow f(r, y)$
5: **until** $z \neq \perp$
6: **return** (r, z)

Figure 3.1: The signing procedure, where the hash function H in step 3 is modelled as a random oracle. The dependency of $\mathcal{D}$ and f on sk is left implicit.

Obviously, for signing to be efficient, it is necessary that $\mathcal{D}$ is efficiently sampleable when given the secret key sk, and $f(r, y)$ is efficiently sampleable (for any r and y) when given sk and the randomness used to sample $r \leftarrow \mathcal{D}$. For verification to be efficient, it needs to be efficiently testable given the public key (only) if z is in the support of $f(r, y)$, for any r, y, z. Similarly, for the scheme to be secure (against key-only attacks) it is necessary that signing should be computationally hard when only given the public key pk. However, these (in)efficiency aspects are not our concern; our security reductions also apply for non-efficient or insecure schemes (though they become somewhat pointless then). Therefore, we keep the public key pk and the secret key sk implicit — unless specified otherwise, we consider them arbitrary but fixed — and so we also keep the dependency of $\mathcal{D}$ and f on $(\mathsf{pk}, \mathsf{sk})$, implicit; the same for the parameters p and ϵ below.

Two important parameters for such a signature scheme are

$$p := \Pr_{\substack{r \leftarrow \mathcal{D}, y \leftarrow \mathcal{Y} \\ z \leftarrow f(r,y)}} [\, z = \perp \,], \tag{3.1}$$

referred to as the *abort probability*, and

$$\epsilon := \mathsf{guess}(r) := \max_{r^\circ} \Pr_{r \leftarrow \mathcal{D}} [\, r = r^\circ \,], \tag{3.2}$$

which is the *guessing probability* of the distribution $\mathcal{D}$.

The above abstract signature scheme design is well motivated by the fact that it covers both *Fiat-Shamir with aborts* (FSwA) signature schemes, as well as *probabilistic hash-and-sign with retry/abort* (HSwA) signature schemes. In the case of FSwA signatures, f is usually a deterministic function, which can be efficiently computed given the secret key and the randomness used to sample r (the "first message" in the Σ-protocol); in the case of HSwA signatures, $\mathcal{D}$ is typically the uniform distribution over strings of a certain length, and f is the preimage-sampling algorithm of a so-called weak preimage-sampleable function.

As a matter of fact, a HSwA signature can be understood as the FSwA signature obtained from the Σ-protocol that chooses a random bit string r as the "first message", and samples the response z as a preimage of the (random) challenge under the weak preimage-sampleable function. We therefore call the general class of signature schemes considered here (covering FSwA and HSwA signatures) *generalized* FSwA signature schemes.

To formally capture that such an honestly generated signature leaks nothing about the secret key, we define the following *accepting honest-verifier zero-knowledge* (acHVZK) property.

Definition 3.2 (Accepting (Statistical) Honest-Verifier Zero-Knowledge)**.** *Let $\zeta > 0$ and $T \in \mathbb{Z}_{\geq 0}$. A generalized FSwA signature scheme $\Sigma = (\mathsf{KGen}, \mathcal{D}, f, \mathcal{V})$ is called (ζ, T)-acHVZK if there exists an algorithm acSim with runtime T, such that on input a public key pk it outputs a triple $(\hat{r}, \hat{y}, \hat{z})$ that (on average over the choice of pk generated by KGen) is ζ-close in statistical distance to (r, y, z) conditioned on $z \neq \bot$, where $r \leftarrow \mathcal{D}$, $y \leftarrow \mathcal{Y}$ and $z \leftarrow f(r, y)$.*

Remark 3.3. *In the above definition, we consider the statistical distance for fixed pk, then average it over the randomness of pk.*

3.2.2 A variant of the Adaptive Reprogramming Lemma

We consider the following, slightly extended variant of the Adaptive Reprogramming Lemma from [GHHM21]. It differs from the original variant in that, next to the quantum read queries, we allow the distinguisher to make (classical) write queries to H (with a bound on the *expected* number of write queries).

Lemma 3.4. *Let $\epsilon > 0$, and let $\{D_i\}_{i \in \mathcal{I}}$ be a family of distributions over $\mathcal{X}$ indexed by a finite set $\mathcal{I}$, such that*

$$\underset{x \leftarrow D_i}{\mathsf{guess}}(x) := \max_{x^\circ \in \mathcal{X}} \Pr_{x \leftarrow D_i}[x = x^\circ] \leq \epsilon$$

for all $i \in \mathcal{I}$. Let $\mathcal{A}^{\bar{H}, \blacksquare}$ be an oracle algorithm that makes one query to an oracle ($\blacksquare$), which is to be instantiated by $\mathcal{O}_0^H$ or $\mathcal{O}_1^{\bar{H}}$ as specified in Fig. 3.2; furthermore, prior to that query, $\mathcal{A}$ makes at most q_r quantum read queries to H, and in expectation *at most q_w classical write queries to H, for given positive numbers $q_r, q_w \in \mathbb{Z}$.[5] Then*

$$\left| \Pr\left[1 \leftarrow \mathcal{A}^{\bar{H}, \mathcal{O}_0^H}\right] - \Pr\left[1 \leftarrow \mathcal{A}^{\bar{H}, \mathcal{O}_1^{\bar{H}}}\right] \right| \leq \left(2q_w + \frac{q_r}{2}\right)\epsilon + \sqrt{q_r \epsilon}\,.$$

[5] We allow arbitrary many read/write H-queries *after* the query to $\mathcal{O}_0$ or $\mathcal{O}_1$.

$\mathcal{O}_0^H(i)$:	$\mathcal{O}_1^{\bar{H}}(i)$:
1: $x \leftarrow \mathcal{D}_i$	1: $x \leftarrow \mathcal{D}_i$
2: $y := H(x)$	2: $H(x) := y \leftarrow \mathcal{Y}$
3: **return** (x, y)	3: **return** (x, y)

Figure 3.2: Reprogramming or not reprogramming, that is the question.

The proof for this extended variant is a quite simple reduction to the original version [GHHM21], exploiting that we can simulate the write queries.

Proof. Consider $\bar{\mathcal{A}}^{H,\mathcal{O}_0^H} = \mathcal{A}^{S^H,\mathcal{O}_0^H}$ and $\bar{\mathcal{A}}^{H,\mathcal{O}_1^{\bar{H}}} = \mathcal{A}^{S^H,\mathcal{O}_1^{\bar{H}}}$, which run $\mathcal{A}$ and simulate the write queries locally, and let Ω be the event that $\mathcal{A}$ has not made a write query (prior to the oracle call) for the point s sampled by the oracle then. This event is well defined and has the same probability in any of the executions of $\mathcal{A}^{\bar{H},\mathcal{O}_0}, \bar{\mathcal{A}}^{H,\mathcal{O}_0}, \bar{\mathcal{A}}^{H,\mathcal{O}_1}, \mathcal{A}^{\bar{H},\mathcal{O}_1}$. Furthermore,

$$\bar{\mathcal{A}}^{H,\mathcal{O}_0^H}[\Omega] \simeq \mathcal{A}^{\bar{H},\mathcal{O}_0^H}[\Omega] \qquad \text{and} \qquad \bar{\mathcal{A}}^{H,\mathcal{O}_1^{\bar{H}}}[\Omega] \simeq \mathcal{A}^{\bar{H},\mathcal{O}_1^{\bar{H}}}[\Omega]\,.$$

It thus follows that

$$\begin{aligned}
&SD(\mathcal{A}^{\bar{H},\mathcal{O}_0^H}, \mathcal{A}^{\bar{H},\mathcal{O}_1^{\bar{H}}}) \\
&\leq SD(\mathcal{A}^{\bar{H},\mathcal{O}_0^H}, \bar{\mathcal{A}}^{H,\mathcal{O}_0^H}) + SD(\bar{\mathcal{A}}^{H,\mathcal{O}_0^H}, \bar{\mathcal{A}}^{H,\mathcal{O}_1^{\bar{H}}}) + SD(\bar{\mathcal{A}}^{H,\mathcal{O}_1^{\bar{H}}}, \mathcal{A}^{\bar{H},\mathcal{O}_1^{\bar{H}}}) \\
&\leq 2\Pr[\neg\Omega] + SD(\bar{\mathcal{A}}^{H,\mathcal{O}_0^H}, \bar{\mathcal{A}}^{H,\mathcal{O}_1^{\bar{H}}}) \leq 2q_w\epsilon + \frac{q_r}{2}\epsilon + \sqrt{q_r\epsilon}\,.
\end{aligned}$$

where the bound on $SD(\bar{\mathcal{A}}^{H,\mathcal{O}_0^H}, \bar{\mathcal{A}}^{H,\mathcal{O}_1^{\bar{H}}})$ follows from the standard adaptive reprogramming lemma (see Proposition 2 in [GHHM21]). □

3.3 A Gap in Prior Analyses of FSwA

A gap in prior analyses of FSwA occurs in the UF-CMA-to-UF-NMA reduction. In this step, signature queries made by the considered UF-CMA-attacker $\mathcal{A}$, which has query access to a signing oracle $\mathsf{Sign}(\mathsf{sk}, \cdot)$ and a random oracle H, must be answered without knowledge of the secret key, replacing real signatures with simulated ones produced by an Honest-Verifier Zero Knowledge (HVZK) simulator associated with the sigma protocol. To ensure that the attacker cannot detect that it is being given simulated signatures, it is also necessary to reprogram the random oracle to be consistent with the transcripts produced by the simulator. The crucial step boils down to replacing the oracle Sign by the oracle Trans (see Fig. 3.3).

$\mathsf{Sign}(\mathsf{sk}, m)$	$\mathsf{Trans}(\mathsf{sk}, m)$
1: **repeat**	1: **repeat**
2: $\quad r \leftarrow \mathcal{D}$	2: $\quad r \leftarrow \mathcal{D}$
3: $\quad y := H(r, m)$	3: $\quad y \leftarrow \mathcal{Y}$
4: $\quad z \leftarrow f(r, y)$	4: $\quad z \leftarrow f(r, y)$
5: **until** $z \neq \perp$	5: **until** $z \neq \perp$
6:	6: $H(r, m) := y$
7: **return** (r, z)	7: **return** (r, z)

Figure 3.3: Oracles Sign and Trans.

Clearly, the adversary $\mathcal{A}$ can attempt to guess r and query H on r before calling Sign/Trans, and then detect the inconsistency introduced by the reprogramming in case of Trans. However, even if the adversary makes no prior H-queries, the distribution of the random oracle changes, and this is where the gap lies. The reprogramming in Trans only reprograms the random oracle with accepting transcripts and thereby shifts the random oracle slightly towards pairs $((r, m), y)$ such that $f(r, y) \neq \perp$ with higher probability. Even though one expects this change in the distribution of the random oracle to be small, there is still a gap that needs to be properly bounded.

Both Lyubashevsky [Lyu12] and KLS [KLS18] miss the loss incurred by the bias in H in their analysis. In [Lyu12] this is missed in the hop from the real signing oracle to Hybrid 1 in the proof of Lemma 5.3 — note that the bound in [Lyu12] remains correct due to a loose analysis. In [KLS18] the gap is missed in the game hop from G_0 to G_1 in the proof of Theorem 3.2. Moreover, this oversight is not a problem limited to [Lyu12] and [KLS18], and it potentially affects all FS-based schemes involving rejection sampling. This includes a long list of works [LNP22, DKL$^+$18, DFG19b, BKP20, BDK$^+$22] on lattice-based and isogeny-based signature schemes (and non-interactive proof systems) that need to be re-examined carefully.

3.4 A UF-CMA-to-UF-NMA Reduction

Throughout this section, and the rest of the chapter, let Σ be an aborting sigma protocol, and $\mathcal{S} := \mathsf{FSwA}[\Sigma, H]$ be the corresponding FSwA signature scheme, as introduced in Sect. 3.1, with parameters ϵ and p defined as in (3.1) and (3.2). We assume $\mathcal{S}$ to be (ζ, T)-acHVZK for given ζ and T.

Following standard notation, we write $\mathbf{Adv}^{\mathsf{UF\text{-}CMA}}_{\mathcal{S}}(\mathcal{A})$ for the advantage of an attacker $\mathcal{A}$ of winning the standard UF-CMA security game (in the QROM) for the scheme $\mathcal{S}$, and similarly $\mathbf{Adv}^{\mathsf{UF\text{-}NMA}}_{\mathcal{S}}(\mathcal{B})$ for the advantage of an attacker $\mathcal{B}$ of winning the standard UF-NMA security game.

3.4.1 The Statements

Our main result is a UF-CMA-to-UF-NMA reduction. The reductions loss is in terms of (bounds on) the parameters p and ϵ. Since these parameters are in general key-dependent, we first state the improved reduction loss for a fixed choice of key pair $(\mathsf{sk}, \mathsf{pk})$, and we write p_{sk} and ϵ_{sk} when we want to make the dependency on the choice of the key explicit (where we assume without loss of generality that pk is determined by sk).

Similarly, we write ζ_{sk} for the statistical distance of the simulated transcript to the actual accepted transcript *for the specific choice* sk *of the key pair*; it then obviously holds that $\mathbb{E}[\zeta_{\mathsf{sk}}] = \zeta$, with the expectation taken over $(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KGen}$. Finally, we write $\mathbf{Adv}^{\mathsf{UF\text{-}CMA}}_{\mathcal{S}}(\mathcal{A}, \mathsf{sk})$ and $\mathbf{Adv}^{\mathsf{UF\text{-}NMA}}_{\mathcal{S}}(\mathcal{B}, \mathsf{sk})$ for the respective advantages when the key is chosen to be sk.

The proof of the following main theorem is presented in the subsequent subsections.

Theorem 3.5. *Let $\mathcal{S}$ be a generalized FSwA signature scheme. Then for every* UF-CMA *attacker $\mathcal{A}^{H,\bullet}$ making at most Q_H quantum queries to H and q_S classical queries to the signing oracle, there exists an* UF-NMA *attacker $\mathcal{B}^H$ making at most Q_H quantum queries to H such that for every fixed choice of key sk with $p_{\mathsf{sk}} < 1$, and for $q_H = Q_H + 1$ we have*

$$\mathbf{Adv}^{\mathsf{UF\text{-}CMA}}_{\mathcal{S}}(\mathcal{A}, \mathsf{sk}) \leq \mathbf{Adv}^{\mathsf{UF\text{-}NMA}}_{\mathcal{S}}(\mathcal{B}, \mathsf{sk}) + \frac{3q_S\sqrt{q_H\epsilon_{\mathsf{sk}}}}{1-p_{\mathsf{sk}}} + 2q_H\sqrt{\frac{q_S\epsilon_{\mathsf{sk}}}{1-p_{\mathsf{sk}}}} + q_S\zeta_{\mathsf{sk}} \,.$$

Moreover, if we count runtime in terms of the number of gates, except that each arithmetic operation on $\mathcal{X}$ and $\mathcal{Y}$ and every comparison among them (with respect to a strict total ordering) are counted as unit runtime, then $\mathsf{TIME}(\mathcal{B}) \leq \mathsf{TIME}(\mathcal{A}) + O\big(\, q_S T + q_H q_S + q_S^2 \,\big)$.

Remark 3.6. *Suppose $\mathcal{B}$ is allowed to use a QRAM, where each cell may contain an element of $\mathcal{X} \times \mathcal{Y}$, up to $O(1)$ many memory pointers and up to $O(1)$ many auxiliary bits. If we count each arithmetic operation and each comparison of the memory pointers as being unit runtime, then $\mathcal{B}$ can further achieve the runtime* $\mathsf{TIME}(\mathcal{B}) \leq \mathsf{TIME}(\mathcal{A}) + O\big(\, q_S T + (q_S + q_H)\log q_S \,\big)$ *using only $O(q_S)$ many cells.*

When taking the expectation over sk on both sides, in order to get the average reduction loss for a random key-pair, one can apply Jensen's inequality to $\mathbb{E}[\sqrt{\epsilon_{\mathsf{sk}}}]$ get a bound in terms of the expectation of ϵ_{sk} (over the choice of sk). Unfortunately, this does not work for the parameter p_{sk}, where Jensen's inequality goes the wrong way round. Hence, we need to have a bound $\bar{p}$ on p_{sk} that holds for all sk, or holds except with small probability (over the choice of sk).

Towards optimizing the bound, it may also make sense to avoid some bad, yet unlikely, choices of $(\mathsf{sk}, \mathsf{pk})$ that make ϵ_{sk} large, i.e. to consider a bound $\bar{\epsilon}$

on the sub-normalized conditional expectation $\Pr[\mathsf{sk} \in \Gamma_\epsilon]\mathbb{E}[\epsilon_{\mathsf{sk}}|\mathsf{sk} \in \Gamma_\epsilon]$, where Γ_ϵ is a subset of the keys for which $\Pr[\mathsf{sk} \notin \Gamma_\epsilon]$ is small.

Altogether, this then gives the following statement.

Corollary 3.7. *Let $\mathcal{S}$ be a generalized FSwA signature scheme that is (ζ, T)-acHVZK. Furthermore, let Γ_ϵ and Γ_p be subsets of keys sk such that $p_{\mathsf{sk}} \leq \bar{p}$ for all $\mathsf{sk} \in \Gamma_p$ and $\Pr[\mathsf{sk} \in \Gamma_\epsilon]\mathbb{E}[\epsilon_{\mathsf{sk}}|\mathsf{sk} \in \Gamma_\epsilon] \leq \bar{\epsilon}$ for parameters $0 < \bar{\epsilon}, \bar{p} < 1$. Then for every $\mathsf{UF\text{-}CMA}$ attacker $\mathcal{A}$ making at most Q_H quantum queries to H and q_S classical queries to the signing oracle, the $\mathsf{UF\text{-}NMA}$ attacker $\mathcal{B}$ (dependent on $\mathcal{A}$) as defined in Theorem 3.5 is such that the following holds for $q_H := Q_H + 1$:*

$$\mathbf{Adv}^{\mathsf{UF\text{-}CMA}}_{\mathcal{S}}(\mathcal{A}) \leq \mathbf{Adv}^{\mathsf{UF\text{-}NMA}}_{\mathcal{S}}(\mathcal{B}) + \frac{3q_S\sqrt{q_H\bar{\epsilon}}}{1-\bar{p}} + 2q_H\sqrt{\frac{q_S\bar{\epsilon}}{1-\bar{p}}} + q_S\zeta + \Pr[\mathsf{sk} \notin \Gamma_p] + \Pr[\mathsf{sk} \notin \Gamma_\epsilon] \,.$$

3.4.2 Proof Strategy

Towards proving the above claim, we consider an arbitrary $\mathsf{UF\text{-}CMA}$ attacker $\mathcal{A}^{H,\bullet}$, where by default the queries to the oracle $\bullet$ are answered by the signing algorithm/oracle Sign in the obvious way.[6] It will be convenient to assume that $\mathcal{A}$ makes one more query to H (which are $q_H = Q_H + 1$ queries in total) in order to check himself if the forged signature is valid under a new message that has not been queried, and then aborts (i.e. outputs $\perp$) if the check fails.

Our goal is to show that

$$\mathcal{A}^{H,\mathsf{Sign}^H(\mathsf{sk},\cdot)}(\mathsf{pk}) \approx \mathcal{A}^{H,\mathsf{Sim}^{\bar{H}}(\mathsf{pk},\cdot)}(\mathsf{pk}) \tag{3.3}$$

with an upper bound on the distance that is in line with the security loss in the theorem statement. Here, Sim simulates the signing oracle by exploiting the non-abort ZK property and the ability to reprogram H, as specified in Fig. 3.4 below.

$\mathsf{Sim}^{\bar{H}}(\mathsf{pk}, m)$:
1: $(\hat{r}, \hat{y}, \hat{z}) \leftarrow \mathsf{acSim}(\mathsf{pk})$
2: $H(\hat{r}, m) := \hat{y}$
3: **return** $(\hat{r}, \hat{z})$

Figure 3.4: Simulating the signing oracle by means of the acHVZK simulator and reprogramming H.

This then implies that the $\mathsf{UF\text{-}NMA}$ attacker $\bar{\mathcal{B}}^H(\mathsf{pk}) = \mathcal{B}^{\mathcal{S}^H}(\mathsf{pk})$ obtained by running $\mathcal{B}^{\bar{H}}(\mathsf{pk}) := \mathcal{A}^{H,\mathsf{Sim}^{\bar{H}}(\mathsf{pk},\cdot)}(\mathsf{pk})$ but simulating the write queries to H

[6] I.e., using the secret key that corresponds to the public key that is given to $\widetilde{\mathcal{A}}$ as input.

internally, is similarly successful in forging a signature as the original UF-CMA attacker $\mathcal{A}$. The crucial property of Sim is of course that it does not need the secret key, and so can indeed be simulated by $\mathcal{B}$ itself.

By assumption on $\mathcal{A}$ (to verify the forged signature before outputting it), we know that $\mathcal{B}^{\bar{H}}(\mathsf{pk})$ outputs a forgery σ^* for a message m^* that correctly verifies under the reprogrammed oracle H (or else outputs $\perp$). However, since H gets reprogrammed only at places $(\hat{r}, m)$ for $m^* \neq m$, σ^* also verifies under the original (unreprogrammed) choice of H. Consequently, whenever $\bar{\mathcal{B}}$ outputs non-$\perp$, it outputs a valid forgery. Thus, we have

$$\begin{aligned}\mathbf{Adv}^{\mathsf{UF\text{-}NMA}}(\bar{\mathcal{B}}) &= \Pr\left[\bar{\mathcal{B}}^{H}(\mathsf{pk}) \neq \perp\right] = \Pr\left[\mathcal{B}^{\bar{H}}(\mathsf{pk}) \neq \perp\right] = \Pr\left[\mathcal{A}^{H,\mathsf{Sim}^{\bar{H}}(\mathsf{pk},\cdot)}(\mathsf{pk}) \neq \perp\right] \\ &\geq \mathbf{Adv}^{\mathsf{UF\text{-}CMA}}(\mathcal{A}) - SD\Big(\mathcal{A}^{H,\mathsf{Sign}^{H}(\mathsf{sk},\cdot)}(\mathsf{pk}), \mathcal{A}^{H,\mathsf{Sim}^{\bar{H}}(\mathsf{sk},\cdot)}(\mathsf{pk})\Big)\,, \qquad (3.4)\end{aligned}$$

where the second and third equalities follow $\bar{\mathcal{B}}^{H}(\mathsf{pk}) \simeq \mathcal{B}^{\bar{H}}(\mathsf{pk}) = \mathcal{A}^{H,\mathsf{Sign}^{\bar{H}}(\mathsf{sk},\cdot)}(\mathsf{pk})$.

Remark 3.8. *If we aim for* strong *unforgeability, similar argument applies, but we additionally require what is known as the* computational unique-response property*, which prevents an efficient attacker to come up with two valid triples $(r, y, z_1), (r, y, z_2)$ with the same first message r and the same challenge y but distinct responses $z_1 \neq z_2$*

Towards showing the closeness (3.3), we introduce intermediate oracles Prog and Trans between Sign and Sim as in Fig. 3.5. The closeness (3.3) is to be shown by means of the following sequence of closeness results:

$$\mathcal{A}^{H,\mathsf{Sign}^{H}(\mathsf{sk},\cdot)}(\mathsf{pk}) \overset{(a)}{\approx} \mathcal{A}^{H,\mathsf{Prog}^{H}(\mathsf{sk},\cdot)}(\mathsf{pk}) \overset{(b)}{\approx} \mathcal{A}^{H,\mathsf{Trans}^{H}(\mathsf{sk},\cdot)}(\mathsf{pk}) \overset{(c)}{\approx} \mathcal{A}^{H,\mathsf{Sim}(\mathsf{pk})^{\bar{H}}}(\mathsf{pk})\,.$$

The closeness claims (a) and (b) are to be shown in Sections 3.4.3 and 3.4.4 respectively, and the closeness claim (c) follows directly from the defining properties of the non-abort ZK simulator; with distance ζ_{sk} for a fixed choice sk of the key, and with distance ζ on average.

$\mathsf{Sign}^{H}(\mathsf{sk}, m)$:	$\mathsf{Prog}^{\bar{H}}(\mathsf{sk}, m)$:	$\mathsf{Trans}^{\bar{H}}(\mathsf{sk}, m)$:
repeat	**repeat**	**repeat**
$r \leftarrow \mathcal{D}$	$r \leftarrow \mathcal{D}$	$r \leftarrow \mathcal{D}$
$y := H(r, m)$	$H(r, m) := y \leftarrow \mathcal{Y}$	$y \leftarrow \mathcal{Y}$
$z := f(r, y)$	$z := f(r, y)$	$z := f(r, y)$
until $z \neq \perp$	**until** $z \neq \perp$	**until** $z \neq \perp$
		$H(r, m) := y$
return (r, z)	**return** (r, z)	**return** (r, z)

Figure 3.5: The oracles Prog, Trans compared with Sign.

Looking ahead, both (a) and (b) hold even for fixed choices of sk and pk; we now consider them arbitrary but fixed in the remainder of this work, and we do not write them explicitly anymore as input to $\mathcal{A}, \mathsf{Sign}$ etc., and we leave the dependency of p and ϵ on the key implicit again.

3.4.3 From Sign to Prog

For $\mathcal{A}$ making q_S queries to Sign, let $\mathcal{G}_i^{\bar{H},\bullet} := \mathcal{A}^{H,[(\mathsf{Prog})^{i-1},\bullet,(\mathsf{Sign}^{\bar{H}})^{q_S-i}]}$ be a run of $\mathcal{A}$ such that the first $i-1$ signing queries are answered by Prog, the ith signing query is answered by an unspecified oracle (which will later be instantiated either by Sign or by Prog), and all remaining signing queries are answered by Sign. Our goal here, is to prove the closeness of $\mathcal{G}_i^{\bar{H},\mathsf{Sign}} \approx \mathcal{G}_i^{\bar{H},\mathsf{Prog}}$. For that purpose, we consider the following sequence of intermediate oracles $O_j^{\blacklozenge} := \mathsf{Loop}^{[(\mathsf{B_P})^{j-1},\blacklozenge,(\mathsf{B_S})^{\infty}]}$ for every $j \in \mathbb{Z}_{>0}$, where $O_1^{\mathsf{B_S}} = \mathsf{Loop}^{\mathsf{B_S}} = \mathsf{Sign}$ and $O_\infty^{\blacklozenge} = \mathsf{Loop}^{\mathsf{B_P}} = \mathsf{Prog}$ regardless of the instantiation of $\blacklozenge$, and so we might just denote the latter as O_∞. Our proof proceeds as outlined below:

$$
\begin{aligned}
\mathcal{A}^{H,\mathsf{Sign}} &= \mathcal{G}_1^{\bar{H},\mathsf{Sign}} = \mathcal{G}_1^{\bar{H},O_1^{\mathsf{B_S}}} \approx \cdots \approx \mathcal{G}_1^{\bar{H},O_k^{\mathsf{B_S}}} \approx \mathcal{G}_1^{\bar{H},O_\infty} \\
&= \mathcal{G}_2^{\bar{H},\mathsf{Sign}} = \mathcal{G}_2^{\bar{H},O_2^{\mathsf{B_S}}} \approx \cdots \approx \mathcal{G}_2^{\bar{H},O_k^{\mathsf{B_S}}} \approx \mathcal{G}_2^{\bar{H},O_\infty} \\
&\vdots \\
&= \mathcal{G}_{q_S}^{\bar{H},\mathsf{Sign}} = \mathcal{G}_{q_S}^{\bar{H},O_{q_S}^{\mathsf{B_S}}} \approx \cdots \approx \mathcal{G}_{q_S}^{\bar{H},O_k^{\mathsf{B_S}}} \approx \mathcal{G}_{q_S}^{\bar{H},O_\infty} = \mathcal{A}^{H,\mathsf{Prog}} ,
\end{aligned}
$$

where $k \in \mathbb{Z}_{>0}$ and the closenesses are to be shown in Lemmas 3.9 and 3.10.

$\mathsf{Loop}^{\blacklozenge}(m)$:	$\mathsf{B_S}^{H}(m)$:	$\mathsf{B_P}^{\bar{H}}(m)$:
1: **repeat**	1: $r \leftarrow \mathcal{D}$	1: $r \leftarrow \mathcal{D}$
2: $\quad out \leftarrow \blacklozenge(m)$	2: $y := H(r,m)$	2: $H(r,m) := y \leftarrow \mathcal{Y}$
3: **until** $out \neq \bot$	3: $z \leftarrow f(r,y)$	3: $z \leftarrow f(r,y)$
4: **return** out	4: **if** $z = \bot$ **return** $\bot$	4: **if** $z = \bot$ **return** $\bot$
	5: **return** (r,z)	5: **else return** (r,z)

Figure 3.6: The repetition loop $\mathsf{Loop}^{\blacklozenge}$ (left), and different instantiations of the body of the loop $(\mathsf{B_S}, \mathsf{B_P})$.

Lemma 3.9. *Let $\mathcal{A}^{H,\bullet}$ be given q_H quantum queries to H and q_S classical queries to $\bullet$. Then for every $i \in [q_S]$ and $j \in \mathbb{Z}_{\geq 0}$ we have*

$$
SD\left(\mathcal{G}_i^{\bar{H},O_j^{\mathsf{B_S}}}, \mathcal{G}_i^{\bar{H},O_\infty}\right) \leq p^{j-1} .
$$

Proof. Let E_j be the event that the jth iteration of O_j is reached. Conditioned on $\neg E_j$, the oracles behave identically, i.e. $O_j[\neg E_j] = O_{j+1}[\neg E_j]$. Hence we have

$$SD\left(\mathcal{G}_i^{\bar{H},O_j^{\mathsf{B_S}}}, \mathcal{G}_i^{\bar{H},O_\infty}\right) \leq \Pr[\neg E_j] \leq p^{j-1} .$$

□

Lemma 3.10. *Let $\mathcal{A}^{H,\bullet}$ be given q_H quantum queries to H and q_S classical queries to $\bullet$. Then for every $i \in [q_S]$ and $j \in \mathbb{Z}_{\geq 0}$ we have*

$$SD\left(\mathcal{G}_i^{\bar{H},O_j^{\mathsf{B_S}}}, \mathcal{G}_i^{\bar{H},O_{j+1}^{\mathsf{B_S}}}\right) \leq p^{j-1} \cdot \left(\left(\frac{2(i-1)}{1-p} + 2(j-1) + \frac{q_H}{2}\right)\epsilon + \sqrt{q_H\epsilon}\right) .$$

Proof. Let E_j be the event that the jth iteration of O_j is reached. Conditioned on $\neg E_j$, the oracles behave identically, i.e. $O_j[\neg E_j] = O_{j+1}[\neg E_j]$. We thus have

$$\begin{aligned} SD\left(\mathcal{G}_i^{\bar{H},O_j^{\mathsf{B_S}}}, \mathcal{G}_i^{\bar{H},O_j^{\mathsf{B_P}}}\right) &= \Pr[E_j] \cdot SD\left(\mathcal{G}_i^{\bar{H},O_j^{\mathsf{B_S}}[E_j]}, \mathcal{G}_i^{\bar{H},O_j^{\mathsf{B_P}}[E_j]}\right) \\ &\leq p^{j-1} \cdot \left(\left(\frac{2(i-1)}{1-p} + 2(j-1) + \frac{q_H}{2}\right)\epsilon + \sqrt{q_H\epsilon}\right) , \end{aligned}$$

where the last inequality is via a direct application of our adaptive reprogramming lemma (Lemma 3.4). □

Corollary 3.11. *Let $\mathcal{A}^{H,\bullet}$ make at most q_H quantum queries to H and q_S classical queries to $\bullet$. Then*

$$SD\left(\mathcal{A}^{H,\mathsf{Sign}^H}, \mathcal{A}^{H,\mathsf{Prog}^{\bar{H}}}\right) \leq \frac{2q_S^2\epsilon}{(1-p)^2} + \frac{3q_S\sqrt{q_H\epsilon}}{2(1-p)} \leq \frac{3q_S\sqrt{q_H\epsilon}}{1-p} ,$$

where the last inequality holds as long as $q_H > 0$ and the right-hand-side is at most 1.

Proof. Combining Lemmas 3.9 and 3.10, we obtain

$$\begin{aligned} SD\left(\mathcal{G}_i^{\bar{H},\mathsf{Sign}}, \mathcal{G}_{i+1}^{\bar{H},\mathsf{Prog}}\right) &\leq SD\left(\mathcal{G}_i^{\bar{H},\mathcal{O}_{k+1}^{\mathsf{B_S}}}, \mathcal{G}_i^{\bar{H},\mathcal{O}_\infty}\right) + \sum_{j\in[k]} SD\left(\mathcal{G}_i^{\bar{H},\mathcal{O}_j^{\mathsf{B_S}}}, \mathcal{G}_i^{\bar{H},\mathcal{O}_{j+1}^{\mathsf{B_S}}}\right) \\ &\leq p^k + \sum_{j\in[k]} p^{j-1} \cdot \left(\left(\frac{2(i-1)}{1-p} + 2(j-1) + \frac{q_H}{2}\right)\epsilon + \sqrt{q_H\epsilon}\right) \\ &\leq p^k + \frac{2q_S\epsilon}{(1-p)^2} + \frac{q_H\epsilon}{2(1-p)} + \frac{\sqrt{q_H\epsilon}}{1-p} , \end{aligned}$$

where the last inequality is via $p \leq 1$ and $i \leq q_S$ and the p^k term vanishes as $k \to \infty$. Summing the above over $i \in [q_H]$, the proof is concluded. □

3.4.4 From Prog to Trans

For the purpose of showing closeness of $\mathcal{A}^{\mathsf{Prog},H}$ and $\mathcal{A}^{\mathsf{Trans},H}$, we introduce a second instantiation H' of the random oracle, which is set to be equal to H at the beginning, and we modify Prog to Prog' so as to also reprogram H', but only on the accepted transcript (see Fig. 3.7 middle). Looking ahead, we notice that this detour via Prog' and H' is not done in the ROM proof; there, we have a (more) direct argument to go from $\mathcal{A}^{\mathsf{Prog},H}$ to $\mathcal{A}^{\mathsf{Trans},H}$, very similar to the one going from $\mathcal{A}^{\mathsf{Sign},H}$ to $\mathcal{A}^{\mathsf{Prog},H}$. The reason we do it this way here is that we obtain a better bound than when trying to mimic the reasoning that is used in the ROM proof.

$\mathsf{Prog}(m)$:	$\mathsf{Prog}'(m)$:	$\mathsf{Trans}(m)$:
1: **repeat**	1: **repeat**	1: **repeat**
2: $\quad r \leftarrow \mathcal{D}(sk)$	2: $\quad r \leftarrow \mathcal{D}(sk)$	2: $\quad r \leftarrow \mathcal{D}(sk)$
3: $\quad H(r,m) := y \leftarrow \mathcal{Y}$	3: $\quad H(r,m) := y \leftarrow \mathcal{Y}$	3: $\quad y \leftarrow \mathcal{Y}$
4: $\quad z := f(r,z)$	4: $\quad z := f(r,z)$	4: $\quad z := f(r,z)$
5: **until** $z \neq \perp$	5: **until** $z \neq \perp$	5: **until** $z \neq \perp$
6:	6: $H'(r,m) := y$	6: $H(r,m) := y$
7: **return** (r,z)	7: **return** (r,z)	7: **return** (r,z)

Figure 3.7: The oracles Prog, Prog' and Trans.

Since the adversary $\mathcal{A}$ in an execution of $\mathcal{A}^{H,\mathsf{Prog}}$ has its random-oracle queries answered by H, and $\mathcal{A}$ has no access to H', we obviously have that $\mathcal{A}^{H,\mathsf{Prog}} = \mathcal{A}^{H,\mathsf{Prog}'}$. Similarly, $\mathcal{A}^{H',\mathsf{Prog}'} = \mathcal{A}^{H,\mathsf{Trans}}$. Thus, it remains to show closeness of $\mathcal{A}^{H,\mathsf{Prog}'}$ and $\mathcal{A}^{H',\mathsf{Prog}'}$. Towards this goal, we first settle the following properties of an execution of Prog'.

Proposition 3.12. *For an arbitrary but fixed message m_0, let (r,y,z) be the first non-$\perp$ transcript produced in an invocation of $\mathsf{Prog}'(m_0)$, and let S' be the set of r's sampled in the loop for which $z = \perp$. Then the following holds.*

- *The distribution of (S',r,y,z) is invariant to the choice of m_0.* (3.5)
- *S' is statistically independent of (r,y,z).* (3.6)
- *For every $r^\circ \in A$, $\Pr[r^\circ \in S'] \leq \frac{\epsilon}{1-p}$.* (3.7)

Proof. Let $t_i = (r_i, y_i, z_i)$ be the transcript sampled in the ith iteration of the loop. For the purpose of the analysis, we assume that t_i is sampled for *every* $i \in \mathbb{Z}_{>0}$, even if the loop stops before. Then, the t_i's are i.i.d. distributed, and S' equals $\{r_1, \ldots, r_{K-1}\}$, with K being minimal such that $z_K \neq \perp$ and $(r,y,z) = t_K$. As the sampling of (S',r,y,z) does not involve m_0 at all, (3.5) follows immediately.

For the analysis of (3.6), we consider the list $L := [t_1, \ldots, t_{K-1}]$; clearly showing independence of L and (r, y, z) implies independence of S' and (r, y, z). Further consider an arbitrary but fixed list $L^\circ = [t_1^\circ, \ldots, t_{k-1}^\circ]$ of transcripts $t_i^\circ = (r_i^\circ, y_i^\circ, z_i^\circ)$, and an arbitrary but fixed transcript $t^\circ = (r^\circ, y^\circ, z^\circ)$. With the goal to show that

$$\Pr\left[L = L^\circ \text{ and } (r, y, z) = t^\circ\right] = \Pr\left[L = L^\circ\right] \cdot \Pr\left[(r, y, z) = t^\circ\right] , \tag{3.8}$$

we may assume $z_1^\circ = \cdots = z_{k-1}^\circ = \bot$ and $z^\circ \neq \bot$, because otherwise both sides of Equation (3.8) vanish trivially. But then, by definition of L and (r, y, z),

$$\begin{aligned}
&\Pr\left[L = L^\circ \text{ and } (r, y, z) = t^\circ\right] = \Pr\left[\forall i < k : t_i = t_i^\circ \text{ and } t_k = t^\circ\right] \\
&= \Pr\left[\forall i < k : t_i = t_i^\circ \text{ and } t_k = t^\circ \text{ and } z_k \neq \bot\right] \\
&= \Pr\left[\begin{matrix} z_k \neq \bot \\ \forall i < k : t_i = t_i^\circ \end{matrix}\right] \cdot \Pr\left[t_k = t^\circ \,\middle|\, \begin{matrix} z_k \neq \bot \\ \forall i < k : t_i = t_i^\circ \end{matrix}\right] \\
&= \Pr\left[\begin{matrix} z_k \neq \bot \\ \forall i < k : t_i = t_i^\circ \end{matrix}\right] \cdot \Pr\left[t_k = t^\circ \middle| z_k \neq \bot\right] \\
&= \Pr\left[L = L^\circ\right] \cdot \Pr\left[t_k = t^\circ \middle| z_k \neq \bot\right] ,
\end{aligned}$$

where the fourth equality is due to independence between $(t_1, \ldots, t_{k-1})$ and t_k. Furthermore, summing up both sides of the above equality over all choices of L°, noting that $\Pr\left[t_k = t^\circ \middle| z_k \neq \bot\right]$ does not depend on k (since the t_i's are i.i.d.), we immediately get that $\Pr\left[t_k = t^\circ \middle| z_k \neq \bot\right] = \Pr\left[(r, y, z) = t^\circ\right]$, which shows (3.8) and thus (3.6).

Next, notice that $|L| \geq \ell$ implies $z_1 = \cdots = z_\ell = \bot$. Thus,

$$\begin{aligned}
&\Pr\left[a^\circ \in S'\right] \leq \sum_{\ell \geq 1} \Pr\left[r_\ell = r^\circ \text{ and } |L| \geq \ell\right] \\
&\leq \sum_{\ell \geq 1} \Pr\left[r_\ell = r^\circ \text{ and } z_1 = \cdots = z_{\ell-1} = \bot\right] \\
&= \sum_{\ell \geq 1} \Pr\left[r_\ell = r^\circ\right] \cdot \Pr\left[z_1 = \cdots = z_{\ell-1} = \bot\right] \leq \sum_{\ell \geq 1} p^{\ell-1}\epsilon \leq \frac{\epsilon}{1-p} ,
\end{aligned}$$

where the equality holds due to the independence between a_ℓ and $(z_1, \ldots, z_{\ell-1})$. This concludes (3.7). □

For this purpose, for every $0 \leq i \leq q_H$ we let $\mathcal{G}_i$ be the hybrid between $\mathcal{A}^{H,\mathsf{Prog}'}$ and $\mathcal{A}^{H',\mathsf{Prog}'}$ that has the first i queries to the random oracle answered by H', and the remaining ones by H. Obviously, $\mathcal{G}_0 = \mathcal{A}^{H,\mathsf{Prog}'}$, while $\mathcal{G}_{q_H} = \mathcal{A}^{H',\mathsf{Prog}'}$. Thus, considering an arbitrary but fixed $1 \leq i \leq q_H$ and setting

$\mathcal{G} := \mathcal{G}_{i-1}$ and $\mathcal{G}' := \mathcal{G}_i$, it is sufficient to show that $\mathcal{G}$ and $\mathcal{G}'$ are close. This is indeed the case:

Lemma 3.13. $SD(\mathcal{G}, \mathcal{G}') \leq 2\sqrt{\frac{q_S \epsilon}{1-p}}$.

Proof. Below, we refer to the ith query of $\mathcal{A}$ to the random oracle, i.e., the query on which $\mathcal{G}$ and $\mathcal{G}'$ differ, as the *crucial query*.

In the respective executions of $\mathcal{G}$ and $\mathcal{G}'$, we define S as the set of all the r's that Prog' sampled but for which the corresponding $z = \perp$, in all the invocations of Prog' before the crucial query. Thus, by construction, at the time of the crucial query, H and H' differ at most at the points in S. (They might agree on a point in S, if the freshly sampled value for H at this point equals the old value.)

For the sake of analysis, consider a binary projective measurement on the input query register for the crucial query, which measures whether or not the input (r, m) is such that $r \in S$. Let Γ be satisfied if $r \notin S$, and let $\tilde{\mathcal{G}}$ and $\tilde{\mathcal{G}}'$ be the two respective games obtained by performing this measurement. Since H and H' only differ at the places (r, m) where $r \in S$, conditioned on Γ, the two oracles behave identically, and thus do $\tilde{\mathcal{G}}$ and $\tilde{\mathcal{G}}'$. Furthermore, the probability $\Pr[\Gamma]$ is the same in both games.

Thus by a double application of the gentle measurement lemma [Wil11], we have

$$SD(\mathcal{G}, \mathcal{G}') \leq SD(\mathcal{G}, \tilde{\mathcal{G}}'[\Gamma]) + SD(\tilde{\mathcal{G}}'[\Gamma], \tilde{\mathcal{G}}[\Gamma]) + SD(\tilde{\mathcal{G}}[\Gamma], \mathcal{G}]) \leq 2\sqrt{\Pr[\neg\Gamma]}\,.$$

Hence, it remains to bound the probability $\Pr[\neg\Gamma]$. The intuition is that S collects those r's that Prog dismisses; thus, $\mathcal{A}$ does not get to see them, so it is hard for him to find an element in S, hence Γ is satisfied most likely. However, turning this intuition into a rigorous argument is not fully straightforward, since the set S, as a random variable, has a somewhat odd distribution.

Let Q be the random variable indicating the number of queries made to Prog prior to the crucial query; we have with certainty that $Q \leq q_S$.

A crucial observation that holds for both $\mathcal{G}, \mathcal{G}'$ is that, conditioned on $Q = q$ for an arbitrary but fixed q, the set S equals $S'_1 \cup \cdots \cup S'_q$ where every S'_j is the set S' that was produced in the jth query of Prog' as specified in Proposition 3.12. We note that, at the time the adversary $\mathcal{A}$ makes the crucial query, H has not been queried, and (r, y, z) in Proposition 3.12 is the only information that is dissipated to the adversary for every prior query to Prog'. It follows from (3.5) and (3.6) that every S'_j is independent from the view of adversary, and hence so is S.

Due to the independence, it suffices to bound $\Pr[r^\circ \in S | Q = q]$ for every $r^\circ \in \mathcal{R}$. Then it follows from the union bound and (3.7) that

$$\Pr[r^\circ \in S | Q = q] \leq \sum_{j \in [q]} \Pr[r^\circ \in S'_j] \leq \frac{q_S \epsilon}{1-p}\,.$$

Putting things together, the proof is concluded. □

Corollary 3.14. *Let $\mathcal{A}^{H,\bullet}$ make at most q_H quantum queries to H and at most q_S queries to $\bullet$. Then,*

$$SD\left(\mathcal{A}^{H,\mathsf{Prog}}, \mathcal{A}^{H,\mathsf{Trans}}\right) \leq 2q_H\sqrt{\frac{q_S\epsilon}{1-p}}\,.$$

3.4.5 Wrapping up the Proof of Theorem 3.5

Collecting the bounds in Corollaries 3.11 and 3.14, for a fixed choice of sk, we obtain

$$\begin{aligned}
&SD\left(\mathcal{A}^{H,\mathsf{Sign}}, \mathcal{A}^{H,\mathsf{Sim}}\right)\\
&\leq SD\left(\mathcal{A}^{H,\mathsf{Sign}}, \mathcal{A}^{H,\mathsf{Prog}}\right) + SD\left(\mathcal{A}^{H,\mathsf{Prog}}, \mathcal{A}^{H,\mathsf{Trans}}\right) + SD\left(\mathcal{A}^{H,\mathsf{Trans}}, \mathcal{A}^{H,\mathsf{Sim}}\right)\\
&\leq \frac{3q_S\sqrt{q_H\epsilon_{\mathsf{sk}}}}{1-p_{\mathsf{sk}}} + 2q_H\sqrt{\frac{q_S\epsilon_{\mathsf{sk}}}{1-p_{\mathsf{sk}}}} + q_S\zeta_{\mathsf{sk}}\,.
\end{aligned}$$

Plugging the above back to Eq. (3.4), we conclude the proof. □

3.4.6 Classical Bounds

Assuming now $\mathcal{A}$ makes only classical queries to H, we give a classical bound below. First, Lemma 3.10 can be replaced with

$$SD\left(\mathcal{G}_j^{\bar{H},O_j^{\mathsf{BS}}}, \mathcal{G}_i^{\bar{H},O_{j+1}^{\mathsf{BS}}}\right) \leq p^{j-1}\left(\frac{i-1}{1-p} + j - 1 + q_H\right)\epsilon\,,$$

leading to a tighter classical variant of Corollary 3.11 as follows

$$\begin{aligned}
SD\left(\mathcal{A}^{H,\mathsf{Sign}^H}, \mathcal{A}^{H,\mathsf{Prog}^{\bar{H}}}\right) &\leq SD\left(\mathcal{G}_i^{\bar{H},O_{k+1}^{\mathsf{BS}}}, \mathcal{G}_i^{\bar{H},O_\infty}\right) + \sum_{j\in[k]} SD\left(\mathcal{G}_i^{\bar{H},O_j^{\mathsf{BS}}}, \mathcal{G}_i^{\bar{H},O_{j+1}^{\mathsf{BS}}}\right)\\
&\leq p^k + \sum_{j\in[k]} p^{j-1}\left(\frac{i-1}{1-p} + j - 1 + q_H\right)\\
&\leq p^k + \left(\frac{q_S}{(1-p)^2} + \frac{q_H}{1-p}\right)\epsilon\,,
\end{aligned}$$

where the p^k term vanishes as $k \to \infty$ and the last inequality is again via $i \leq q_S$. Then, we replace Lemma 3.13 with

$$SD(\mathcal{G}, \mathcal{G}') = SD(\tilde{\mathcal{G}}', \tilde{\mathcal{G}}) \leq \Pr[\neg\Gamma]\cdot SD(\tilde{\mathcal{G}}'[\Gamma], \tilde{\mathcal{G}}[\Gamma]) \leq \Pr[\neg\Gamma] \leq \frac{q_S\epsilon}{1-p}\,,$$

where $\tilde{\mathcal{G}}$ and $\tilde{\mathcal{G}}'$ are as defined in the proof of Lemma 3.13, which gives us

$$SD\big(\mathcal{A}^{H,\mathsf{Prog}}, \mathcal{A}^{H,\mathsf{Trans}}\big) \leq \frac{q_H q_S \epsilon}{1-p} \ .$$

Collecting the bounds, for a fixed choice of sk, we obtain

$$\begin{aligned} &SD\big(\mathcal{A}^{H,\mathsf{Sign}}, \mathcal{A}^{H,\mathsf{Sim}}\big) \\ &\leq SD\left(\mathcal{A}^{H,\mathsf{Sign}}, \mathcal{A}^{H,\mathsf{Prog}}\right) + SD\left(\mathcal{A}^{H,\mathsf{Prog}}, \mathcal{A}^{H,\mathsf{Trans}}\right) + SD\left(\mathcal{A}^{H,\mathsf{Trans}}, \mathcal{A}^{H,\mathsf{Sim}}\right) \\ &\leq \left(\frac{q_S^2}{(1-p)^2} + \frac{2q_H q_S}{1-p}\right)\epsilon + q_S \zeta_{\mathsf{sk}} \ , \end{aligned}$$

Plugging into Eq. (3.4), we immediately obtain the following.

Theorem 3.15. *Let $\mathcal{S}$ be a generalized FSwA signature scheme. Then, for every* UF-CMA *attacker $\mathcal{A}^{H,\bullet}$ making at most Q_H and q_S classical queries to H and the signing oracle respectively, the* UF-NMA *attacker $\mathcal{B}$ (dependent on $\mathcal{A}$) as defined in Theorem 3.5 is such that for every fixed choice of key sk with $p_{\mathsf{sk}} < 1$, and for $q_H = Q_H + 1$ we have*

$$\mathbf{Adv}^{\mathsf{UF\text{-}CMA}}_{\mathcal{S}}(\mathcal{A}, \mathsf{sk}) \leq \mathbf{Adv}^{\mathsf{UF\text{-}NMA}}_{\mathcal{S}}(\mathcal{B}, \mathsf{sk}) + \left(\frac{q_S^2}{(1-p_{\mathsf{sk}})^2} + \frac{2q_H q_S}{1-p_{\mathsf{sk}}}\right)\epsilon_{\mathsf{sk}} + q_S \zeta_{\mathsf{sk}} \ .$$

Corollary 3.16. *Let $\mathcal{S}$ be a generalized FSwA signature scheme that is (ζ, T)-acHVZK. Furthermore, let Γ_ϵ and Γ_p be subsets of keys sk such that $p_{\mathsf{sk}} \leq \bar{p}$ for all $\mathsf{sk} \in \Gamma_p$ and $\Pr[\mathsf{sk} \in \Gamma_\epsilon]\mathbb{E}[\epsilon_{\mathsf{sk}} | \mathsf{sk} \in \Gamma_\epsilon] \leq \bar{\epsilon}$ for parameters $0 < \bar{\epsilon}, \bar{p} < 1$. Then for every* UF-CMA *attacker $\mathcal{A}$ making at most Q_H and q_S classical queries to H and the signing oracle respectively, the* UF-NMA *attacker $\mathcal{B}$ (dependent on $\mathcal{A}$) as defined in Theorem 3.5 is such that the following holds for $q_H := Q_H + 1$:*

$$\begin{aligned} \mathbf{Adv}^{\mathsf{UF\text{-}CMA}}_{\mathcal{S}}(\mathcal{A}) \leq \mathbf{Adv}^{\mathsf{UF\text{-}NMA}}_{\mathcal{S}}(\mathcal{B}) + \left(\frac{q_S^2}{(1-\bar{p})^2} + \frac{2q_H q_S}{1-\bar{p}}\right)\bar{\epsilon} + q_S \zeta \\ + \Pr[\mathsf{sk} \notin \Gamma_p] + \Pr[\mathsf{sk} \notin \Gamma_\epsilon] \ . \end{aligned}$$

3.5 Tighter Bounds in QROM

In this section we provide tighter bounds in QROM. We recycle the same reduction and the same proving strategy as in Section 3.4. The only difference is that we provide a more efficient hybrid sequence to show the closeness $\mathcal{A}^{H,\mathsf{Prog}} \approx \mathcal{A}^{H,\mathsf{Trans}}$, compared to Section 3.4.4.

3.5.1 The Statements

Theorem 3.17. *Let $\mathcal{S}$ be a generalized FSwA signature scheme. Then for every* UF-CMA *attacker $\mathcal{A}^{H,\bullet}$ making at most Q_H quantum queries to H and q_S classical queries to the signing oracle, the* UF-NMA *attacker $\mathcal{B}$ (dependent on $\mathcal{A}$) as defined in Theorem 3.5 is such that is such that for every fixed choice of key* sk *with $p_{\mathsf{sk}} < 1$, and for $q_H = Q_H + 1$ we have*

$$\mathbf{Adv}^{\mathsf{UF\text{-}CMA}}_{\mathcal{S}}(\mathcal{A}, \mathsf{sk}) \leq \mathbf{Adv}^{\mathsf{UF\text{-}NMA}}_{\mathcal{S}}(\mathcal{B}, \mathsf{sk}) + \frac{8q_S\sqrt{q_H\epsilon_{\mathsf{sk}}}}{1-p_{\mathsf{sk}}} + q_S\zeta_{\mathsf{sk}} \ .$$

Corollary 3.18. *Let $\mathcal{S}$ be a generalized FSwA signature scheme that is (ζ, T)-acHVZK. Furthermore, let Γ_ϵ and Γ_p be subsets of keys* sk *such that $p_{\mathsf{sk}} \leq \bar{p}$ for all $\mathsf{sk} \in \Gamma_p$ and $\Pr[\mathsf{sk} \in \Gamma_\epsilon]\mathbb{E}[\epsilon_{\mathsf{sk}} | \mathsf{sk} \in \Gamma_\epsilon] \leq \bar{\epsilon}$ for parameters $0 < \bar{\epsilon}, \bar{p} < 1$. Then for every* UF-CMA *attacker $\mathcal{A}$ making at most Q_H quantum queries to H and q_S classical queries to the signing oracle, the* UF-NMA *attacker $\mathcal{B}$ (dependent on $\mathcal{A}$) as defined in Theorem 3.5 is such that the following holds for $q_H := Q_H + 1$:*

$$\mathbf{Adv}^{\mathsf{UF\text{-}CMA}}_{\mathcal{S}}(\mathcal{A}) \leq \mathbf{Adv}^{\mathsf{UF\text{-}NMA}}_{\mathcal{S}}(\mathcal{B}) + \frac{8q_S\sqrt{q_H\bar{\epsilon}}}{1-\bar{p}} + q_S\zeta + \Pr[\mathsf{sk} \notin \Gamma_p] + \Pr[\mathsf{sk} \notin \Gamma_\epsilon] \ .$$

3.5.2 From Prog to Trans

We recap the oracles Loop, $\mathsf{B_P}$ and introduce a new oracle $\mathsf{B_T}$ in Fig. 3.8. Our strategy for proving closeness of $\mathcal{A}^{H,\mathsf{Prog}}$ and $\mathcal{A}^{H,\mathsf{Trans}}$ is to replace, query by query and iteration by iteration, the body $\mathsf{B}^{\bar{H}}_{\mathsf{P}}$ of the repeat loop of $\mathsf{Prog}^{\bar{H}} = \mathsf{Loop}^{\mathsf{B}^{\bar{H}}_{\mathsf{P}}}$ by the body $\mathsf{B}^{\bar{H}}_{\mathsf{T}}$ of the repeat loop of $\mathsf{Trans}^{\bar{H}} = \mathsf{Loop}^{\mathsf{B}^{\bar{H}}_{\mathsf{T}}}$.

$\mathsf{Loop}^{\bullet}(\mathsf{sk}, m)$:	$\mathsf{B}^{\bar{H}}_{\mathsf{P}}(\mathsf{sk}, m)$:	$\mathsf{B}^{\bar{H}}_{\mathsf{T}}(\mathsf{sk}, m)$:
1: **repeat**	1: $r \leftarrow \mathcal{D}$	1: $r \leftarrow \mathcal{D}$
2: $\quad out \leftarrow \bullet(\mathsf{sk}, m)$	2: $H(r, m) := y \leftarrow \mathcal{Y}$	2: $y \leftarrow \mathcal{Y}$
3: **until** $out \neq \perp$	3: $z \leftarrow f(r, y)$	3: $z \leftarrow f(r, y)$
4: **return** out	4: **if** $z \neq \perp$, $H(r, m) := y$	4: **if** $z \neq \perp$, $H(r, m) := y$
	5: **if** $z = \perp$ **return** $\perp$	5: **if** $z = \perp$ **return** $\perp$
	6: **else return** (r, z)	6: **else return** (r, z)

Figure 3.8: The repetition loop (left), and different instantiations of the body of the loop $(\mathsf{B_P}, \mathsf{B_T})$. The greyed out line 4 in $\mathsf{B_P}$ is irrelevant and can be ignored.

In order to capture the corresponding hybrid game and hybrid step, we introduce the following game, played by an oracle algorithm $\mathcal{C}^{\bar{H},\bullet}$ with the following features (see Fig. 3.9). During its run, $\mathcal{C}$ is allowed to make multiple

quantum read and *classical write* queries to H, and moreover one single query to an unspecified oracle that is to be instantiated by $\mathsf{B_P}$ or $\mathsf{B_T}$.

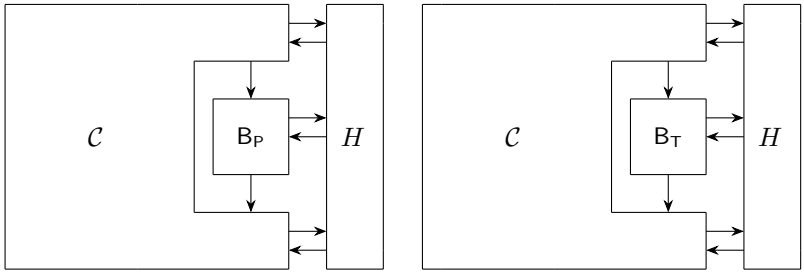

Figure 3.9: The oracle algorithm $\mathcal{C}$, which makes an fixed number of (at most) q_r quantum read queries to H, an expected number of (at most) q_w classical write queries to H, and one query to either $\mathsf{B_P}$ or $\mathsf{B_T}$.

For parameters q_r, q_w, we then define $\mathbf{Adv}(q_r, q_w)$ to be the maximal advantage of distinguishing the two games from Fig. 3.9, i.e.,

$$\mathbf{Adv}(q_r, q_w) := \sup_{\mathcal{C}} SD\left(\mathcal{C}^{\bar{H},\mathsf{B_P}}, \mathcal{C}^{\bar{H},\mathsf{B_T}}\right) , \tag{3.9}$$

takes the supremum over all $\mathcal{C}^{\bar{H},\blacklozenge}$ as above that make at most q_r quantum read queries to H *in the worst case* and at most q_w classical write queries *on average*, regardless of how the unspecified oracle ($\blacklozenge$) is instantiated.

The following allows us to control the closeness of $\mathcal{A}^{H,\mathsf{Prog}}$ and $\mathcal{A}^{H,\mathsf{Trans}}$ in terms of $\mathbf{Adv}(q_r, q_w)$.

Lemma 3.19. *Let $\mathcal{A}^{H,\bullet}$ be given q_H (read) queries to H and q_S signing queries. Then for p as in (3.1),*

$$SD\left(\mathcal{A}^{H,\mathsf{Prog}^{\bar{H}}}, \mathcal{A}^{H,\mathsf{Trans}^{\bar{H}}}\right) \leq \frac{q_S}{1-p} \cdot \mathbf{Adv}\left(q_H, \frac{q_S}{1-p}\right) .$$

At first glance, this is a straightforward hybrid argument, where we switch, one by one, the body of the jth iteration of the repeat loop in the ith signing query from $\mathsf{B_P}$ to $\mathsf{B_T}$; however, one needs to be careful since there is no fixed upper bound on the number of times the loop in Prog and Trans is repeated. However, via similar reasoning as in [BBD+23], we can exploit that it becomes less and less likely that the loop where we switch from $\mathsf{B_P}$ to $\mathsf{B_T}$ is reached, and so we can bound the distinguishing advantage by an infinite geometric series, which can be controlled. For this to work it is crucial that $\mathcal{A}$ cannot influence the number of loop repetitions (by the way of choosing m); whether the loop is repeated or not depends solely on the random choice of r.

For completeness, we work out the details in the formal proof below; it is somewhat tedious but altogether rather straightforward.

Proof. Let

$$\mathcal{G}_i^{\bar{H},\bullet} := \mathcal{A}^{H,[(\mathsf{Trans}^{\bar{H}})^{i-1},\bullet,(\mathsf{Prog}^{\bar{H}})^{q_S-i}]}$$

be a run of $\mathcal{A}$ such that the first $i-1$ signing queries are answered by Trans, the ith signing query is answered by an unspecified oracle (which will later be instantiate either by Trans or by Prog), and all remaining signing queries are answered by Prog. By construction, $\mathcal{G}_i^{\bar{H},\bullet}$ makes q_H quantum read queries to H, an expected number of at most $q_S/(1-p)$ classical write queries to H (the ones made by the runs of Trans and Prog), and one query to the unspecified oracle. Furthermore,

$$\mathcal{A}^{H,\mathsf{Prog}^{\bar{H}}} = \mathcal{G}_1^{\bar{H},\mathsf{Prog}^{\bar{H}}}\,, \quad \mathcal{G}_i^{\bar{H},\mathsf{Trans}} = \mathcal{G}_{i+1}^{\bar{H},\mathsf{Prog}} \quad \text{and} \quad \mathcal{G}_{q_S}^{\bar{H},\mathsf{Trans}^{\bar{H}}} = \mathcal{A}^{H,\mathsf{Trans}^{\bar{H}}}\,.$$

Our goal is to show the closeness of $\mathcal{G}_i^{\bar{H},\mathsf{Prog}^{\bar{H}}}$ and $\mathcal{G}_i^{\bar{H},\mathsf{Trans}^{\bar{H}}}$ for every $i \in \{1,\dots,q_S\}$, with error at most $\frac{1}{1-p}\mathbf{Adv}\big(q_H, \frac{q_S}{1-p}\big)$, which then implies the claim via

$$\mathcal{A}^{H,\mathsf{Prog}^{\bar{H}}} = \mathcal{G}_1^{\bar{H},\mathsf{Prog}^{\bar{H}}} \approx \mathcal{G}_1^{\bar{H},\mathsf{Trans}^{\bar{H}}} = \mathcal{G}_2^{\bar{H},\mathsf{Prog}^{\bar{H}}} \approx \cdots \approx \mathcal{G}_{q_S}^{\bar{H},\mathsf{Trans}^{\bar{H}}} = \mathcal{A}^{H,\mathsf{Trans}^{\bar{H}}}\,.$$

To show the claimed closeness, we do a similar hybrid argument as above, but now over the different iterations of the repeat loop in Trans and Prog. Concretely, we consider

$$\mathsf{Loop}_j^{\bar{H},\blacklozenge} := \mathsf{Loop}^{[(\mathsf{B}_\mathsf{T}^{\bar{H}})^{j-1},\blacklozenge,(\mathsf{B}_\mathsf{P}^{\bar{H}})^{\infty}]}$$

and observe that

$$\mathsf{Loop}_1^{\bar{H},\mathsf{B}_\mathsf{P}} = \mathsf{Prog}^{\bar{H}}\,,\ \ \mathsf{Loop}_j^{\bar{H},\mathsf{B}_\mathsf{T}} = \mathsf{Loop}_{j+1}^{\bar{H},\mathsf{B}_\mathsf{P}} \ \text{ and } \ \mathsf{Loop}_\infty^{\bar{H},\mathsf{B}_\mathsf{P}} = \mathsf{Loop}_\infty^{\bar{H},\mathsf{B}_\mathsf{T}} = \mathsf{Trans}^{\bar{H}}\,,$$

and so it remains to show the following closeness claims:

$$\begin{aligned}\mathcal{G}_i^{\bar{H},\mathsf{Prog}^{\bar{H}}} &= \mathcal{G}_i^{\bar{H},\mathsf{Loop}_1^{\bar{H},\mathsf{B}_\mathsf{P}}} \approx \mathcal{G}_i^{\bar{H},\mathsf{Loop}_1^{\bar{H},\mathsf{B}_\mathsf{T}}} = \mathcal{G}_i^{\bar{H},\mathsf{Loop}_2^{\bar{H},\mathsf{B}_\mathsf{P}}} \\ &\approx \cdots \approx \mathcal{G}_i^{\bar{H},\mathsf{Loop}_k^{\bar{H},\mathsf{B}_\mathsf{P}}} \approx \mathcal{G}_i^{\bar{H},\mathsf{Loop}_\infty^{\bar{H},\mathsf{B}_\mathsf{P}}} = \mathcal{G}_i^{\bar{H},\mathsf{Trans}^{\bar{H}}}\end{aligned} \tag{3.10}$$

for sufficiently large $k > 1$.

The last closeness claim is rather straightforward. Indeed, $\mathsf{Loop}_\infty^{\bar{H},\mathsf{B}_\mathsf{P}}$ and $\mathsf{Loop}_k^{\bar{H},\mathsf{B}_\mathsf{P}}$ behave (potentially) differently only if the repeat loop, which is at the core of the two algorithms, is repeated at least k times, which happens

only if all the k prior calls to B_T produce $\perp$.[7] Formally, we write E_k for this event that the loop is repeated at least k times, and we observe that it happens with probability $\Pr[E_k] = p^{k-1}$ only. Then $\mathsf{Loop}_k[\neg E_k] = \mathsf{Loop}_\infty[\neg E_k]$, and therefore, exploiting that $\mathcal{G}_i$ makes only one call to (whatever version of) Loop,

$$\begin{aligned} SD\left(\mathcal{G}_i^{\bar{H},\mathsf{Loop}_k^{\bar{H},\mathsf{B_P}}}, \mathcal{G}_i^{\bar{H},\mathsf{Loop}_\infty^{\bar{H},\mathsf{B_P}}}\right) &\le SD\left(\mathcal{G}_i^{\bar{H},\mathsf{Loop}_k^{\bar{H},\mathsf{B_P}}[E_k]}, \mathcal{G}_i^{\bar{H},\mathsf{Loop}_\infty^{\bar{H},\mathsf{B_P}}[E_k]}\right)\Pr[E_k] \\ &\le \Pr[E_k] \ \le p^{k-1}\,. \end{aligned}$$

It remains to show that $\mathcal{G}_i^{\bar{H},\mathsf{Loop}_j^{\bar{H},\mathsf{B_P}}} \approx \mathcal{G}_i^{\bar{H},\mathsf{Loop}_j^{\bar{H},\mathsf{B_T}}}$ for every j. Similar to above, we note and exploit that

$$\mathsf{Loop}_j^{\bar{H},\mathsf{B_P}}[\neg E_j] = \mathsf{Loop}_j^{\bar{H},\mathsf{B_T}}[\neg E_j]\,,$$

i.e., the two behave identically if the jth iteration is not reached. Thus,

$$\begin{aligned} SD\left(\mathcal{G}_i^{\bar{H},\mathsf{Loop}_j^{\bar{H},\mathsf{B_P}}}, \mathcal{G}_i^{\bar{H},\mathsf{Loop}_j^{\bar{H},\mathsf{B_T}}}\right) &\le SD\left(\mathcal{G}_i^{\bar{H},\mathsf{Loop}_j^{\bar{H},\mathsf{B_P}}[E_j]}, \mathcal{G}_i^{\bar{H},\mathsf{Loop}_j^{\bar{H},\mathsf{B_T}}[E_j]}\right)\Pr[E_j] \\ &\le SD\left(\mathcal{C}_{i,j}^{\bar{H},\mathsf{B_P}}, \mathcal{C}_{i,j}^{\bar{H},\mathsf{B_T}}\right)p^{j-1} \le \mathbf{Adv}\Big(q_H, \tfrac{q_S}{1-p}\Big)p^{j-1}\,, \end{aligned}$$

where the second inequality is obtained by letting $\mathcal{C}_{i,j}^{\bar{H},\blacklozenge}$ to be the oracle algorithm $\mathcal{G}_i^{\bar{H},\mathsf{Loop}_j^{\bar{H},\blacklozenge}[E_j]}$, which performs the run of $\mathsf{Loop}_j[E_j]$ (which is promised to reach the jth iteration) internally, but forwards the oracle query. Noting that $\mathcal{C}_{i,j}^{\bar{H},\blacklozenge}$ is as required, with at most q_H quantum read queries to H and an average of at most $\frac{q_S}{1-p}$ classical write queries, the final upper bound applies.

Adding up all the error terms in (3.10), we get that

$$\begin{aligned} SD\left(\mathcal{G}_i^{\bar{H},\mathsf{Prog}^{\bar{H}}}, \mathcal{G}_i^{\bar{H},\mathsf{Trans}^{\bar{H}}}\right) &\le \mathbf{Adv}\Big(q_H, \frac{q_S}{1-p}\Big)\sum_{j=1}^{k-1} p^{j-1} + p^{k-1} \\ &\longrightarrow \ \frac{1}{1-p}\mathbf{Adv}\Big(q_H, \frac{q_S}{1-p}\Big) \end{aligned}$$

for $k \to \infty$. This concludes the proof. □

The main technical challenge, and so the main innovation of this work, lies in establishing the following claim.

Proposition 3.20. *For any positive $q_r, q_w \in \mathbb{Z}$ and for* $\mathbf{Adv}$ *as specified in* (3.9)

$$\mathbf{Adv}(q_r, q_w) \le (5q_w + q_r)\epsilon + 2\sqrt{q_r\epsilon}\,.$$

[7] It is actually necessary to loop for $k+1$ iterations for the two to behave differently, but we do not need to be tight here.

We note that the only difference between B_P and B_T is whether H gets reprogrammed or not in case $z = \perp$ (in which case r remains unknown). This difference can only be detected when $\mathcal{C}$ makes a future query to H on input r; but due to the assumed high entropy in r, this is unlikely to happen. Turning this intuition into a proof when $\mathcal{C}$ can make *quantum* queries to H results in a hybrid argument over the q_r queries to H, which in turn results in a bound on the distinguishing advantage of the order $q_r\sqrt{\epsilon}$. Thus, the actual challenge lies in finding a hybrid argument that shows that the advantage actually scales as $\sqrt{q_r\epsilon}$ (plus neglectable terms).

Proof. For the sake of the analysis, we introduce the following aborting variants of B_P and B_T, defined in Fig. 3.10. The only different to the non-aborting variants is line 6., where aB_P and aB_T instruct to abort instead of returning (r, z) in case $z \neq \perp$. We stress that the abort command is a *global* abort, causing the ambient game ($\mathcal{C}^{\bar{H},\mathsf{aB}_\mathsf{P}}$ or $\mathcal{C}^{\bar{H},\mathsf{aB}_\mathsf{T}}$) to abort if $z \neq \perp$, instead of returning (r, z) to the ambient game then.

$\mathsf{aB}_\mathsf{P}^{\bar{H}}(m)$:	$\mathsf{aB}_\mathsf{T}(m)$:
1: $r \leftarrow \mathcal{D}$	1: $r \leftarrow \mathcal{D}$
2: $H(r, m) := y \leftarrow \mathcal{Y}$	2: $y \leftarrow \mathcal{Y}$
3: $z \leftarrow f(r, y)$	3: $z \leftarrow f(r, y)$
4: **if** $z \neq \perp$ **then** $H(r, m) := y$	4: **if** $z \neq \perp$ **then** $H(r, m) := y$
5: **if** $z = \perp$ **then return** $\perp$	5: **if** $z = \perp$ **then return** $\perp$
6: **else abort**	6: **else abort**

Figure 3.10: Aborting variants of B_P and B_T, which cause the ambient game to abort if $z \neq \perp$, instead of returning (r, z). Line 4. then becomes irrelevant also for aB_T.

Since $\mathsf{B}_\mathsf{P}^{\bar{H}}$ and $\mathsf{B}_\mathsf{T}^{\bar{H}}$ behave identically anyway if $z \neq \perp$ (both have reprogrammed $H(r, m)$ and return (r, m)), asking to abort in that a case does not affect the distinguishing advantage, i.e.,

$$\Pr\left[1 \leftarrow \mathcal{C}^{\bar{H},\mathsf{B}_\mathsf{P}^{\bar{H}}}\right] - \Pr\left[1 \leftarrow \mathcal{C}^{\bar{H},\mathsf{B}_\mathsf{T}^{\bar{H}}}\right] = \Pr\left[1 \leftarrow \mathcal{C}^{\bar{H},\mathsf{aB}_\mathsf{P}^{\bar{H}}}\right] - \Pr\left[1 \leftarrow \mathcal{C}^{\bar{H},\mathsf{aB}_\mathsf{T}}\right] .$$

In order to show that the right hand side is small, we proceed through the following sequence of hybrid games $\mathcal{G}_0$ to $\mathcal{G}_5$, given in Fig. 3.11. We refer to the $\mathcal{G}_i$'s as "games" but after all these are just algorithms $\mathcal{G}_i^{\bar{H}}$ with write access to H, and so the concepts from Sect. 2.2 readily apply.

We also note that $\mathcal{G}_0^{\bar{H}}$ is semantically equal to $\mathcal{C}^{\bar{H},\mathsf{aB}_\mathsf{P}^{\bar{H}}}$, i.e. $\mathcal{G}_0^{\bar{H}} = \mathcal{C}^{\bar{H},\mathsf{aB}_\mathsf{P}^{\bar{H}}}$; we merely have split $\mathcal{C}$ into two parts, which respectively captures $\mathcal{C}$'s behavior before and after the call to $\mathsf{aB}_\mathsf{P}^{\bar{H}}$, and we have spelled out $\mathsf{aB}_\mathsf{P}^{\bar{H}}$. Correspondingly for $\mathcal{G}_5^{\bar{H}}$ and $\mathcal{C}^{\bar{H},\mathsf{aB}_\mathsf{T}^{\bar{H}}}$.

$\mathcal{G}_0$:
1: $(m, \mathsf{st}) \leftarrow \mathcal{C}_0^{\bar{H}}$
2:
3: $r \leftarrow \mathcal{D}$
4: $H(r, m) := y \leftarrow \mathcal{Y}$
5: $z \leftarrow f(r, y)$
6: **if** $z \neq \perp$ **abort**
7: $b \leftarrow \mathcal{C}_1^{\bar{H}}(\mathsf{st}, \perp)$
8: **return** b

$\mathcal{G}_1$:
1: $(m, \mathsf{st}) \leftarrow \mathcal{C}_0^{\bar{H}}$
2:
3: $r \leftarrow \mathcal{D}$
4: $y := H(r, m)$
5: $z \leftarrow f(r, y)$
6: **if** $z \neq \perp$ **abort**
7: $b \leftarrow \mathcal{C}_1^{\bar{H}}(\mathsf{st}, \perp)$
8: **return** b

$\mathcal{G}_2$:
1: $(m, \mathsf{st}) \leftarrow \mathcal{C}_0^{\bar{H}}$
2: $b \leftarrow \mathcal{C}_1^{\bar{H}}(\mathsf{st}, \perp)$
3: $r \leftarrow \mathcal{D}$
4: $y := H(r, m)$
5: $z \leftarrow f(r, y)$
6: **if** $z \neq \perp$ **abort**
7:
8: **return** b

$\mathcal{G}_3$:
1: $(m, \mathsf{st}) \leftarrow \mathcal{C}_0^{\bar{H}}$
2: $b \leftarrow \mathcal{C}_1^{\bar{H}}(\mathsf{st}, \perp)$
3: $r \leftarrow \mathcal{D}$
4: $H(r, m) := y \leftarrow \mathcal{Y}$
5: $z \leftarrow f(r, y)$
6: **if** $z \neq \perp$ **abort**
7:
8: **return** b

$\mathcal{G}_4$:
1: $(m, \mathsf{st}) \leftarrow \mathcal{C}_0^{\bar{H}}$
2: $b \leftarrow \mathcal{C}_1^{\bar{H}}(\mathsf{st}, \perp)$
3: $r \leftarrow \mathcal{D}$
4: $y \leftarrow \mathcal{Y}$
5: $z \leftarrow f(r, y)$
6: **if** $z \neq \perp$ **abort**
7:
8: **return** b

$\mathcal{G}_5$:
1: $(m, \mathsf{st}) \leftarrow \mathcal{C}_0^{\bar{H}}$
2:
3: $r \leftarrow \mathcal{D}$
4: $y \leftarrow \mathcal{Y}$
5: $z \leftarrow f(r, y)$
6: **if** $z \neq \perp$ **abort**
7: $b \leftarrow \mathcal{C}_1^{\bar{H}}(\mathsf{st}, \perp)$
8: **return** b

Figure 3.11: The hybrid games.

Game hop $\mathcal{G}_0$ to $\mathcal{G}_1$. The game $\mathcal{G}_1$ is obtained from $\mathcal{G}_0$ by replacing the reprogramming step $H(r, m) := y \leftarrow \mathcal{Y}$ to the hash evaluation $y := H(r, m)$ in line 4. Therefore, recalling the bound q_r on the number of quantum read queries to H and the bound q_w on the expected number of (classical) write queries of $\mathcal{C}$, from directly applying our variant of the adaptive reprogramming lemma (Lemma 3.4) we obtain

$$\left| \Pr\left[1 \leftarrow \mathcal{G}_0\right] - \Pr\left[1 \leftarrow \mathcal{G}_1\right] \right| \leq \left(2q_w + \frac{q_r}{2}\right)\epsilon + \sqrt{q_r \epsilon}\,.$$

As a quick remark, considering the bigger context, we note that y is now computed as in the original signing oracle; thus, at first glance it seems that we are making a step back again, towards $\mathcal{A}^{H,\mathsf{Sign}}$ instead of $\mathcal{A}^{H,\mathsf{Trans}}$. However, it is a crucial step in this delicate sequence of hybrids.

Game hop $\mathcal{G}_1$ to $\mathcal{G}_2$. The game $\mathcal{G}_2$ is identical to $\mathcal{G}_1$ except that the run of $\mathsf{aB_P}$ (including the decision to abort) is delayed to the very end of the game. Conditioned on the event that the execution of $\mathcal{C}_1^{\bar{H}}(\mathsf{st}, \perp)$ does not reprogram H at the point (r, m), for the r sampled in step 3., the two games $\mathcal{G}_1$ and $\mathcal{G}_2$

behave identically. I.e., using our formalism,

$$\mathcal{G}_1\big[\mathcal{C}_1^{\bar{H}}(\mathsf{st},\perp) \text{ does not program } H(r,m)\big] = \mathcal{G}_2[\,\mathcal{C}_1^{\bar{H}}(\mathsf{st},\perp) \text{ does not program } H(r,m)\,]\,.$$

Due to the min-entropy requirement (3.2) on r, and due to the bound q_w on the expected number of write queries that $\mathcal{C}$ performs, the probability that $\mathcal{C}_1^{\bar{H}}(\mathsf{st},\perp)$ does reprogram $H(r,m)$ is at most $q_w\epsilon$ (and it is the same probability in both games). Therefore,

$$\big|\ \Pr\left[1 \leftarrow \mathcal{G}_1\right] - \Pr\left[1 \leftarrow \mathcal{G}_2\right]\ \big| \leq q_w\epsilon\,.$$

Game hop $\mathcal{G}_2$ to $\mathcal{G}_3$. The game $\mathcal{G}_3$ is defined from $\mathcal{G}_2$ by replacing the hash evaluation $y := H(r,m)$ in line 4 to reprogramming $H(r,m) := y \leftarrow \mathcal{Y}$. This is again a direct application of the adaptive reprogramming lemma, and so

$$\big|\ \Pr\left[1 \leftarrow \mathcal{G}_2\right] - \Pr\left[1 \leftarrow \mathcal{G}_3\right]\ \big| \leq \Big(2q_w + \frac{q_r}{2}\Big)\epsilon + \sqrt{q_r\epsilon}\,.$$

Game hop $\mathcal{G}_3$ to $\mathcal{G}_4$. The game $\mathcal{G}_4$ is obtained from $\mathcal{G}_3$ by dropping the reprogramming $H(r,m) := y$ in line 4. Since there are no further queries to H after that point, this change has no effect on the output b, and so

$$\Pr\left[1 \leftarrow \mathcal{G}_3\right] = \Pr\left[1 \leftarrow \mathcal{G}_4\right]\ .$$

Game hop $\mathcal{G}_4$ to $\mathcal{G}_5$. The game $\mathcal{G}_5$ is the same as $\mathcal{G}_4$, but the run of $\mathcal{C}_1^{\bar{H}}(\mathsf{st},\perp)$ is moved to the end again. This is just a syntactic change, which only affects *when* the abort decision is made, but does not affect the actual outcome of the game. Hence

$$\Pr\left[1 \leftarrow \mathcal{G}_4\right] = \Pr\left[1 \leftarrow \mathcal{G}_5\right]\ .$$

Collecting the upperbounds, the proof is concluded. □

3.5.3 Wrapping up the Proof of Theorem 3.17

For a fixed choice of sk, collecting the bounds in Lemma 3.19 and *Proposition* 3.20, we obtain

$$SD(\mathcal{A}^{H,\mathsf{Prog}}, \mathcal{A}^{H,\mathsf{Trans}}) \leq \frac{q_S}{1-p} \cdot \mathbf{Adv}\left(q_H, \frac{q_S}{1-p}\right)$$
$$\leq \left(\frac{5q_S^2}{(1-p)^2} + \frac{q_S q_H}{1-p}\right)\epsilon + \frac{2q_S}{1-p}\sqrt{q_H\epsilon} \leq \sqrt{\frac{6q_S^2 q_H}{(1-p)^2}} + \frac{2q_S}{1-p}\sqrt{q_H\epsilon} \leq \frac{5q_S}{1-p}\sqrt{q_H\epsilon}\,,$$

where the third inequality holds as long as $q_H > 0$ (which is satisfied because $q_H = Q_H + 1 > 0$) and the right-hand side is at most 1. Combined with

Corollary 3.11, we obtain

$$\begin{aligned}
&SD\left(\mathcal{A}^{H,\mathsf{Sign}}, \mathcal{A}^{H,\mathsf{Sim}}\right) \\
&\leq SD\left(\mathcal{A}^{H,\mathsf{Sign}}, \mathcal{A}^{H,\mathsf{Prog}}\right) + SD\left(\mathcal{A}^{H,\mathsf{Prog}}, \mathcal{A}^{H,\mathsf{Trans}}\right) + SD\left(\mathcal{A}^{H,\mathsf{Trans}}, \mathcal{A}^{H,\mathsf{Sim}}\right) \\
&\leq \frac{3q_S}{1-p_{\mathsf{sk}}}\sqrt{\frac{q_H\epsilon_{\mathsf{sk}}}{1-p_{\mathsf{sk}}}} + \frac{5q_S}{1-p_{\mathsf{sk}}}\sqrt{\frac{q_H\epsilon_{\mathsf{sk}}}{1-p_{\mathsf{sk}}}} + q_S\zeta_{\mathsf{sk}} \leq \frac{8q_S}{1-p_{\mathsf{sk}}}\sqrt{\frac{q_H\epsilon_{\mathsf{sk}}}{1-p_{\mathsf{sk}}}} + q_S\zeta_{\mathsf{sk}} \,.
\end{aligned}$$

Plugging the above back to Eq. (3.4), we conclude Theorem 3.17.

3.6 Concrete Analysis of Dilithium

In this section, we show, partly computer aided, a lower bound on the min-entropy, i.e., an upper bound on the guessing probability, of the first message $\mathsf{highBits}(\mathbf{Ay}, 2\gamma_2)$ in the sigma protocol underlying Dilithium for some relevant choices of the parameters.

We briefly recall relevant details of Dilithium here. Let $R_q \cong \mathbb{F}_q[X]/(X^n+1)$ be a cyclotomic ring over the finite field $\mathbb{F}_q$ of order q, where X^n+1 splits completely in $\mathbb{F}_q$, i.e. there exist pair-wise distinct $\alpha_1, \ldots, \alpha_n \in \mathbb{F}_q$ such that $X^n+1 = (X-\alpha_1)\cdots(X-\alpha_n)$. Let $k, \ell, \gamma_1, \gamma_2 \in \mathbb{Z}_{>0}$ be so that $k \geq \ell$, $\gamma_1 > \gamma_2$, and let $S_{\gamma_1-1} \subseteq R_q$ be of size $2\gamma_1 - 1$. In addition, let $\mathsf{highBits} : R_q \times \mathbb{Z}_{>0} \to R_q$ be a function so that each preimage of $\mathsf{highBits}(\cdot, 2\gamma_2)$ (restricting the second input to be $2\gamma_2$) is of size at most $2\gamma_2 + 1$.[8] What we are interested in, is to show that, with overwhelming probability over the choice of $\mathbf{A} \leftarrow R_q^{k\times\ell}$, the min-entropy of $\mathsf{highBits}(\mathbf{Ay}, 2\gamma_2)$ is large over the randomness of $\mathbf{y} \leftarrow S_{\gamma_1-1}^{\ell}$. The concrete parameters are specified in Fig. 3.12 below.

	n	ℓ	q	γ_1	γ_2
NIST2	256	4	8380417	2^{17}	$(q-1)/88$
NIST3	256	5	8380417	2^{19}	$(q-1)/32$
NIST5	256	7	8380417	2^{19}	$(q-1)/32$

Figure 3.12: Concrete parameters of Dilithium.

3.6.1 Controlling the Min-Entropy via the Rank of A

First, note that, for the top-most square $\mathbf{A}^{\square} \in R_q^{\ell\times\ell}$ of $\mathbf{A} \in R_q^{k\times\ell}$,

$$\mathsf{H}_\infty(\mathsf{highBits}(\mathbf{Ay}, 2\gamma_2)) \geq \mathsf{H}_\infty(\mathsf{highBits}(\mathbf{A}^{\square}\mathbf{y}, 2\gamma_2)) \geq \mathsf{H}_\infty(\mathbf{A}^{\square}\mathbf{y}) - n\ell\log(2\gamma_2+1)\,.$$

[8] We are going to slightly abuse the notation: when $\mathsf{highBits}(\cdot, 2\gamma_2)$ is applied to a tuple of elements in R_q, we take it as understood that it is applied componentwisely to the tuple.

Furthermore,

$$\mathsf{H}_\infty(\mathbf{A}^\square \mathbf{y}) \geq \mathsf{H}_\infty(\mathbf{y}) - (n\ell - \mathsf{rank}(\mathbf{A}^\square)) \log(q)\,,$$

where $\mathsf{rank}(\mathbf{A}^\square)$ is the rank of $\mathbf{A}^\square$ acting on R_q^ℓ as a $\mathbb{F}_q$-linear space. Finally, by the choice of $\mathbf{y}$, $\mathsf{H}_\infty(\mathbf{y}) \geq n\ell \log(2\gamma_1 - 1)$. Thus, altogether,

$$\mathsf{H}_\infty(\mathsf{highBits}(\mathbf{A}\mathbf{y}, 2\gamma_2)) \geq n\ell \log\left(\frac{2\gamma_1 - 1}{2\gamma_2 + 1}\right) - (n\ell - \mathsf{rank}(\mathbf{A}^\square)) \log(q)\,. \quad (3.11)$$

Therefore, it suffices to have good enough control over $\mathsf{rank}(\mathbf{A}^\square)$.

3.6.2 Numerically Controlling the Rank of A

The following is a direct consequence of the requirement that $X^n + 1$ splits completely in $\mathbb{F}_q$.

Lemma 3.21. *There is a $\mathbb{F}_q$-algebra isomorphism between*

$$\phi : R_q^{\ell\times\ell} \xrightarrow{\sim} \bigoplus_{1\leq i\leq n} \mathbb{F}_q^{\ell\times\ell}\,.$$

Furthermore, ϕ is $\mathbb{F}_q$-rank-preserving, i.e. for every $\mathbf{A}^\square$, with $\phi(\mathbf{A}^\square) = \bigoplus_i \mathbf{D}_i$, we have $\mathsf{rank}(\mathbf{A}^\square) = \mathsf{rank}(\phi(\mathbf{A}^\square)) = \sum_i \mathsf{rank}(\mathbf{D}_i)$.

From Lemma 3.21, we now know that the distribution of $\mathsf{rank}(\mathbf{A}^\square)$ equals the distribution of $\sum_i \mathsf{rank}(\mathbf{D}_i)$ for random and independent $\mathbf{D}_1, \ldots, \mathbf{D}_n \leftarrow \mathbb{F}_q^{\ell\times\ell}$.

The rank of a random matrix. Below, we thus consider a uniformly random $\mathbf{D} \leftarrow \mathbb{F}_q^{\ell\times\ell}$, and we work out the distribution of $\mathsf{rank}(\mathbf{D})$. For this purpose, let

$$\mathbf{D} = \begin{pmatrix}\mathbf{D}^1 & \ldots & \mathbf{D}^\ell\end{pmatrix}\,,$$

where each $\mathbf{D}^j \in \mathbb{F}_q^\ell$ is the jth column of $\mathbf{D}$. Define the *rank sequence* $r_1, \ldots, r_\ell$ given by

$$r_j := \mathsf{rank}\begin{pmatrix}\mathbf{D}^1 & \ldots & \mathbf{D}^j\end{pmatrix}\,.$$

With the convention that $r_0 := 0$, define their difference sequence $d_1, \ldots, d_\ell$ as $d_j := r_j - r_{j-1} \in \{0, 1\}$. In other words, d_j indicates whether the jth column increases the rank or not.

Lemma 3.22 below gives the distribution of the difference sequence $(d_1, \ldots, d_n)$ for a random $\mathbf{D}$.

Lemma 3.22. *The probability that a random matrix $\mathbf{D} \in \mathbb{F}_q^{\ell\times\ell}$ has a given*

difference sequence $(d_1, \ldots, d_\ell) \in \{0,1\}^\ell$ *is*

$$\pi_\ell(q, d_1, \ldots, d_\ell) := \prod_{1 \le j \le \ell} \left(d_j + (-1)^{d_j} q^{-(\ell - r_{j-1})} \right) ,$$

where $r_1, \ldots, r_\ell$ *is naturally defined as* $r_j = d_1 + \cdots + d_j$*, with* $r_0 = 0$*.*

Proof. For any $j \in \{1, \ldots, \ell\}$, conditioned on the columns $\mathbf{D}^1, \ldots, \mathbf{D}^{j-1}$, the matrix $\mathbf{D}$ has $d_j = 0$ if and only if the jth column $\mathbf{D}^j$ lies in $\mathsf{span}_{\mathbb{F}_q}\{\mathbf{D}^1, \ldots, \mathbf{D}^{j-1}\}$, which happens with probability

$$\Pr\left[\mathbf{D}^j \in \mathsf{span}_{\mathbb{F}_q}\{\mathbf{D}^1, \ldots, \mathbf{D}^{j-1}\}\right] = \frac{\left|\mathsf{span}_{\mathbb{F}_q}\{\mathbf{D}^1, \ldots, \mathbf{D}^{j-1}\}\right|}{q^\ell} = q^{-(\ell - r_{j-1})} ,$$

and it has $d_j = 1$ with complementary probability $1 - q^{-(\ell - r_{j-1})}$. The probability of a particular difference sequence $d_1, \ldots, d_\ell$ is then the product of the respective probabilities above, which matches the claim. □

Since $\mathsf{rank}(\mathbf{D}) = d_1 + \cdots + d_\ell$, we have

$$\Pr\left[\mathsf{rank}(\mathbf{D}) = r\right] = \sum_{d \in S_r^\ell} \pi_\ell(q, d) , \tag{3.12}$$

where $S_r^\ell := \{(d_1, \ldots, d_\ell) \in \{0,1\}^\ell \mid d_1 + \cdots + d_\ell = r\}$. Note that, by using Lemma 3.22, one can show that $\Pr\left[\mathsf{rank}(\mathbf{D}) = r\right] = p_r(1/q)$ for an (ℓ-dependent) integer polynomial p_r of degree at most ℓ^2. The equality (3.12) gives rise to an algorithm that computes the distribution of $\Pr\left[\mathsf{rank}(\mathbf{D}) = r\right]$ (as polynomials in $1/q$) in time $2^{O(\ell)}$. For $\ell = 5$ (which is in line with the NIST3 parameters of Dilithium) one obtains the polynomials given in Fig. 3.13.[9]

[9] For such a small choice of ℓ the exponential run time is no issue. As a matter of fact, this could still be worked out by hand.

$$
\begin{aligned}
p_0(1/q) &= q^{-25} \\
p_1(1/q) &= -q^{-25} - q^{-24} - q^{-23} - q^{-22} - q^{-21} + q^{-20} + q^{-19} + q^{-18} + q^{-17} + q^{-16} \\
p_2(1/q) &= q^{-24} + q^{-23} + 2q^{-22} + 2q^{-21} + q^{-20} - q^{-19} - 2q^{-18} - 4q^{-17} - 4q^{-16} \\
&\quad - 2q^{-15} - q^{-14} + q^{-13} + 2q^{-12} + 2q^{-11} + q^{-10} + q^{-9} \\
p_3(1/q) &= -q^{-22} - q^{-21} - 2q^{-20} - q^{-19} + 3q^{-17} + 4q^{-16} + 5q^{-15} + 3q^{-14} \\
&\quad - 3q^{-12} - 5q^{-11} - 4q^{-10} - 3q^{-9} + q^{-7} + 2q^{-6} + q^{-5} + q^{-4} \\
p_4(1/q) &= q^{-19} + q^{-18} - q^{-16} - 2q^{-15} - 3q^{-14} - 2q^{-13} + q^{-12} + 3q^{-11} + 4q^{-10} \\
&\quad + 3q^{-9} + q^{-8} - 2q^{-7} - 3q^{-6} - 2q^{-5} - q^{-4} + q^{-2} + q^{-1} \\
p_5(1/q) &= -q^{-15} + q^{-14} + q^{-13} - q^{-10} - q^{-9} - q^{-8} + q^{-7} + q^{-6} + q^{-5} \\
&\quad - q^{-2} - q^{-1} + 1 \,.
\end{aligned}
$$

Figure 3.13: The polynomials $p_r(1/q) = \Pr[\mathsf{rank}(\mathbf{D}) = r]$ for $\ell = 5$.

The rank of a random block-diagonal matrix. Towards controlling the distribution of $\mathsf{rank}(\mathbf{A}^{\square})$, we consider the generating function for the distribution of $\mathsf{rank}(\mathbf{D})$, given by

$$f(z) := \sum_{0 \le r \le \ell} \Pr[\mathsf{rank}(\mathbf{D}) = r] \cdot z^r = \sum_{0 \le r \le \ell} p_r(1/q) \cdot z^r \,. \tag{3.13}$$

Then, the nth power $f^n(z)$ of the above generating function generates the distribution of $\mathsf{rank}(\mathbf{A}^{\square}) = \mathsf{rank}(\mathbf{D}_1) + \cdots + \mathsf{rank}(\mathbf{D}_n)$, i.e.

$$f^n(z) = \sum_{0 \le r \le \ell n} \Pr\left[\mathsf{rank}(\mathbf{A}^{\square}) = r\right] \cdot z^r \,.$$

On input $f(z)$, as a polynomial in z with degree ℓ, with coefficient being polynomials in $1/q$ with degree (at most) ℓ^2, the nth power $f^n(z)$ can be computed in time polynomial in n and ℓ. $\Pr\left[\mathsf{rank}(\mathbf{A}^{\square}) = r\right]$ can then be obtained by reading out the coefficient of the degree-r term of $f^n(z)$, which is again an integer polynomial in $1/q$, and evaluating it (as a rational number) for the considered choice of $q \in \mathbb{Z}$.

For any $i \in \mathbb{Z}_{>0}$, we can then compute

$$\Pr\left[\mathsf{rank}(\mathbf{A}^{\square}) < n\ell - i\right] = \sum_{r < n\ell - i} \Pr\left[\mathsf{rank}(\mathbf{A}^{\square}) = r\right]$$

as a rational number $\frac{a}{b}$ with $a \le b \in \mathbb{Z}$, which we can then upper bound by writing $b = ad + e$ for $e \in \{0, \ldots, a-1\}$, and noting that

$$\Pr\left[\mathsf{rank}(\mathbf{A}^{\square}) < n\ell - i\right] = \frac{a}{b} = \frac{a}{ad+e} \le \frac{1}{d} \,.$$

Finally, counting the number of bits in the bit representation of d excluding the most significant bit gives us the largest $\delta \in \mathbb{Z}$ so that $2^\delta \leq d$, and thus $\Pr\left[\mathsf{rank}(\mathbf{A}^\square) < n\ell - i\right] \leq 2^{-\delta}$.

3.6.3 Plugging in the Numbers

For parameters as described in Fig. 3.12, we present the relevant quantities, obtained by following the above computation steps using Sage. In Fig. 3.14 below are the obtained upper bounds δ_i such that $\Pr[\mathsf{rank}(\mathbf{A}^\square) < n\ell - i] \leq \delta_i$ for selective choices of i's, together with the resulting bounds $\mathsf{H}_\infty(\mathsf{highBits}(\mathbf{Ay}, 2\gamma_2)) \geq \eta_i$ obtained via (3.11). For the NIST3 parameter, in particular, we see that except with probability at most 2^{-440} the matrix $\mathbf{A}^\square$ has corank at most 23, and then

$$\mathsf{H}_\infty(\mathsf{highBits}(\mathbf{Ay}, 2\gamma_2)) \geq 752\,,$$

which means that the guessing probability is at most

$$\mathsf{guess}\left(\mathsf{highBits}(\mathbf{Ay}, 2\gamma_2)\right) \leq 2^{-752}\,,$$

where the randomness is over the choice of $\mathbf{y}$.

i	0	1	5	6	8	12	23	33	44	63
$-\log \delta_i$	14	31	99	117	153	227	440	641	867	1268
$\eta_i^{(2)}$	471	448	356	333	287	195	0	0	0	0
$\eta_i^{(3)}$	1281	1258	1166	1143	1097	1005	752	522	269	0
$\eta_i^{(5)}$	1794	1771	1679	1656	1610	1518	1265	1035	782	345

Figure 3.14: $\Pr\left[\mathsf{H}_\infty(\mathsf{highBits}(\mathbf{Ay}, 2\gamma_2)) \geq \eta_i^{(\iota)}\right] \leq \delta_i$ for NISTι parameters, $\iota \in \{2, 3, 5\}$

One can obtain a slightly better bound by considering the *average* guessing probability over the choice of $\mathbf{A}$, averaged over the non-normalized distribution of $\mathbf{A}$ conditioned on $\mathbf{A}^\square$ having corank at most 12. Concretely, letting Γ_{23} be the event that $\mathbf{A}^\square$ has corank at most 23, where we know that $\Pr[\neg\Gamma_{23}] \leq 2^{-752}$, we obtain

$$\begin{aligned}
&\mathop{\mathbb{E}}_{A}\left[\,\mathsf{guess}\left(\mathsf{highBits}(\mathbf{Ay}, 2\gamma_2)\right) \;\middle|\; \Gamma_{23}\right] \cdot \Pr\left[\,\Gamma_{23}\,\right] \\
&\leq \sum_{0 \leq i \leq 23} \Pr[\mathsf{rank}(\mathbf{A}^\square) = n\ell - i \;\wedge\; \Gamma_{23}] \cdot 2^{-\eta_i^{(3)}} \leq \sum_{0 \leq i \leq 23} \delta_{i-1} \cdot 2^{-\eta_i^{(3)}} \leq 2^{-1172}\,,
\end{aligned}$$

with the convention that $\delta_{-1} = 1$. Similarly, we work out the (average-case) entropy bounds for all parameters. In Fig. 3.15, for a suitable choice of $a \in \mathbb{Z}_{\geq 0}$, and the corresponding event $\Gamma_\epsilon : \mathsf{rank}(\mathbf{A}^\square) \geq n\ell - a$, we provide an upperbound

for $\Pr[\neg\Gamma_\epsilon]$, and an upperbound on

$$\bar{\epsilon} := \mathop{\mathbb{E}}_{A}\left[\,\mathsf{guess}\left(\mathsf{highBits}(\mathbf{Ay}, 2\gamma_2)\right) \mid \Gamma_\epsilon\,\right] \cdot \Pr\left[\,\Gamma_\epsilon\,\right] \ .$$

Then, we compare the obtained classical security loss in Corollary 3.16

$$L = \left(\frac{q_S^2}{(1-\bar{p})^2} + \frac{2q_H q_S}{1-\bar{p}}\right)\bar{\epsilon} + q_S\zeta + \Pr\left[\,\neg\Gamma_p\,\right] + \Pr\left[\,\neg\Gamma_\epsilon\,\right] \ ,$$

with the corresponding quantum security loss in Corollary 3.7

$$L^* = \frac{8q_S\sqrt{q_H\bar{\epsilon}}}{1-\bar{p}} + q_S\zeta + \Pr\left[\,\neg\Gamma_p\,\right] + \Pr\left[\,\neg\Gamma_\epsilon\,\right] \ ,$$

taking $\zeta = 0$, Γ_p as always satisfied, and a heuristic choice for $\bar{p}$, as in [KLS18].

	$\bar{p}$	q_S	q_H	a	$\Pr[\neg\Gamma_\epsilon]$	$\bar{\epsilon}$	security loss
NIST2	$\leq \frac{49}{64}$	2^{64}	2^{128}	5	$\leq 2^{-99}$	$\leq 2^{-437}$	$L^* \leq 2^{-85}$
				12	$\leq 2^{-227}$	$\leq 2^{-404}$	$L \leq 2^{-209}$
			2^{64}	6	$\leq 2^{-117}$	$\leq 2^{-432}$	$L^* \leq 2^{-114}$
			1	8	$\leq 2^{-153}$	$\leq 2^{-422}$	$L^* \leq 2^{-141}$
NIST3	$\leq \frac{103}{128}$	2^{64}	2^{192}	23	$\leq 2^{-440}$	$\leq 2^{-1172}$	$L^* \leq 2^{-421}$
				44	$\leq 2^{-867}$	$\leq 2^{-1115}$	$L \leq 2^{-856}$
NIST5	$\leq \frac{759}{1024}$	2^{64}	2^{256}	33	$\leq 2^{-641}$	$\leq 2^{-1655}$	$L^* \leq 2^{-630}$
				63	$\leq 2^{-1268}$	$\leq 2^{-1591}$	$L \leq 2^{-1267}$

Figure 3.15: Concrete security loss of Dilithium, worked out via numeric calculation as described above.

3.6.4 Analytically Controlling ϵ_{sk}

Next, we give an analytic bound controlling $\mathsf{rank}(\mathbf{A}^\square)$ and hence ϵ_{sk} via (3.11). Crucially, by Lemma 8 in the full version of [BBD+23], over the random choice of the key pair $(\mathsf{sk}, \mathsf{pk})$, the distribution of $\mathsf{rank}(\mathbf{A}^\square)$ is identical to that of $\sum_{i\in[n]} \mathsf{rank}(\mathbf{D}_i)$ where $\mathbf{D}_1, \ldots, \mathbf{D}_n \leftarrow \mathbb{F}_q^{\ell\times\ell}$ are sampled uniformly and independently, which we bound below.

Theorem 3.23. *Let $\ell, n \in \mathbb{Z}_{>0}$ and q be a prime. Then for $\mathbf{D}_1, \ldots, \mathbf{D}_n \leftarrow \mathbb{F}_q^{\ell\times\ell}$*

and for every $a \in \mathbb{Z}_{\geq 0}$,

$$\Pr\left[\sum_{i\in[n]} \mathsf{rank}(\mathbf{D}_i) \leq n\ell - a\right] \leq e^{4/3}\,(n/q)^a \cdot (1-1/q)^{-n\ell}\,.$$

Proof. Let $\mathsf{corank}(\mathbf{D}_i) := \ell - \mathsf{rank}(\mathbf{D}_i)$ for each $i \in [n]$, we first work out the probability that $\mathsf{corank}(\mathbf{D}_i) = r$ for every $r \in \{0,\ldots,\ell\}$. Via one of the isomorphism theorems of linear spaces, once $\ker(\mathbf{D}_i) := \{v \in \mathbb{F}_q^\ell \mid \mathbf{D}_i v = 0\}$ and $\mathsf{Im}(\mathbf{D}_i) := \{\mathbf{D}_i v \mid v \in \mathbb{F}_q^\ell\}$ are fixed, the linear mapping $\mathbf{D}_i/\ker(\mathbf{D}_i) : \mathbb{F}_q^\ell/\ker(\mathbf{D}_i) \to \mathsf{Im}(\mathbf{D}_i)$ with $v + \ker(\mathbf{D}_i) \mapsto \mathbf{D}_i v$ uniquely determines $\mathbf{D}_i$. Note that for $\mathsf{corank}(\mathbf{D}_i) = r$, the spaces $\ker(\mathbf{D}_i)$ and $\mathsf{Im}(\mathbf{D}_i)$ can be (and only be) any sub-spaces of dimensions r and $n-r$ respectively, and $\mathbf{D}_i/\ker(\mathbf{D}_i)$ is bijective, and hence uniquely determined by an $(\ell-r)\times(\ell-r)$ invertible matrix (once the two spaces are fixed). Hence, we have the following chain of bijective correspondences:

$$\begin{gathered}
\left\{D \in \mathbb{F}_q^{\ell\times\ell} \;\middle|\; \mathsf{corank}(D) = r\right\} \\
\updownarrow \\
\left\{K \leq \mathbb{F}_q^\ell \;\middle|\; \dim(K) = r\right\} \times \left\{V \leq \mathbb{F}_q^\ell \;\middle|\; \dim(V) = \ell - r\right\} \times \mathsf{GL}(\ell-r, \mathbb{F}_q) \\
\updownarrow \\
\left\{K \leq \mathbb{F}_q^\ell \;\middle|\; \dim(K) = r\right\}^2 \times \mathsf{GL}(\ell-r, \mathbb{F}_q)\,,
\end{gathered}$$

where we denote the $\mathbb{F}_q$-subspace relation by $\leq$, and the second correspondence is via identifying the dual space $V^\perp := \{u \in \mathbb{F}_q^\ell \mid u^T v = 0\ \forall v \in V\}$ so that $\dim V^\perp + \dim V = \ell$ for every subspace $V \leq \mathbb{F}_q^\ell$. Working out the above numbers via counting, we obtain

$$\begin{aligned}
\Pr\left[\mathsf{corank}(\mathbf{D}_i) = r\right] &= \left|\left\{D \in \mathbb{F}_q^{\ell\times\ell} \;\middle|\; \mathsf{corank}(D) = r\right\}\right| \cdot q^{-\ell^2} \\
&= \left|\left\{K \leq \mathbb{F}_q^\ell \;\middle|\; \dim(V) = r\right\}\right|^2 \cdot \left|\,\mathsf{GL}(\ell-r, \mathbb{F}_q)\,\right| \cdot q^{-\ell^2} \\
&= \left(\frac{\prod_{i\in[r]}\left(q^\ell - q^{i-1}\right)}{\prod_{i\in[r]}\left(q^r - q^{i-1}\right)}\right)^2 \cdot \left(\prod_{i\in[\ell-r]}\left(q^{\ell-r} - q^{i-1}\right)\right) \cdot q^{-\ell^2} \\
&\leq q^{-r^2} \cdot \left(1 - \frac{1}{q}\right)^{-\ell}\,, \qquad (3.14)
\end{aligned}$$

where the last inequality is via pulling out the leading factors from the products, and simplifying the remaining terms.

What we are interested in is the event where $\sum_i \mathsf{rank}(\mathbf{D}_i) \leq n\ell - a$, which is equivalent to the event where $\sum_i \mathsf{corank}(\mathbf{D}_i) \geq a$, and can be bounded as

below, for every $t > 0$:

$$\begin{aligned}\Pr\left[\sum_{i\in[n]} \mathsf{corank}(\mathbf{D}_i) \geq a\right] &\leq e^{-at} \cdot \mathbb{E}\left[e^{\sum_i \mathsf{corank}(\mathbf{D}_i)\cdot t}\right] = e^{-at} \prod_{i\in[n]} \mathbb{E}\left[e^{\mathsf{corank}(\mathbf{D}_i)\cdot t}\right] \\ &\leq e^{-at} \cdot \left(\sum_{r\geq 0} e^{rt} \cdot q^{-r^2}\right)^n \cdot \left(1-\frac{1}{q}\right)^{-n\ell} \\ &\leq e^{-at} \cdot \exp\left(\sum_{r>0} n \cdot e^{rt} \cdot q^{-r^2}\right) \cdot \left(1-\frac{1}{q}\right)^{-n\ell}, \end{aligned} \tag{3.15}$$

where the first inequality is via Markov's bound, the first equality is via noticing that $\mathbf{D}_1, \ldots, \mathbf{D}_n$ are mutually independent, the second inequality is via (3.14), and the last inequality is via the fact that $1 + x \leq \exp(x)$ for every $x \in \mathbb{R}$. Plugging in $t = \ln(q) - \ln(n)$, we immediately obtain

$$\begin{aligned}(3.15) &\leq \left(\frac{n}{q}\right)^a \cdot \exp\left(\sum_{r>0} n^{-(r-1)} \cdot q^{-r(r-1)}\right) \cdot \left(1-\frac{1}{q}\right)^{-n\ell} \\ &\leq \left(\frac{n}{q}\right)^a \cdot \exp\left(\sum_{r\geq 0} n^{-r} \cdot q^{-2r}\right) \cdot \left(1-\frac{1}{q}\right)^{-n\ell} \\ &\leq \left(\frac{n}{q}\right)^a \cdot \exp\left(\frac{1}{1-1/(nq^2)}\right) \cdot \left(1-\frac{1}{q}\right)^{-n\ell} \\ &\leq \left(\frac{n}{q}\right)^a \cdot e^{4/3} \cdot \left(1-\frac{1}{q}\right)^{-n\ell}, \end{aligned}$$

where the last inequality is via the fact that $q \geq 2$. This concludes the proof. □

Combining the above and (3.11) with $\Gamma_\epsilon : \mathsf{corank}(A^{\square}) \leq a$, we get

$$\begin{aligned}\Pr[\Gamma_\epsilon] \cdot \mathbb{E}[\epsilon_{\mathsf{sk}} \mid \Gamma_\epsilon] &\leq \sum_{0\leq r\leq a} \Pr\left[\mathsf{corank}(\mathbf{A}^{\square}) = r\right] \cdot \mathbb{E}\left[\epsilon_{\mathsf{sk}} \;\middle|\; \mathsf{corank}(\mathbf{A}^{\square}) = r\right] \\ &\leq \sum_{0\leq r\leq a} e^{4/3}(n/q)^r \cdot (1-1/q)^{-n\ell} \cdot \left(\frac{2\gamma_2+1}{2\gamma_1-1}\right)^{n\ell} \cdot q^r \\ &\leq a \cdot e^{4/3} \cdot n^a \cdot \left(\frac{2\gamma_2+1}{2\gamma_1-1}\right)^{n\ell} \cdot (1-1/q)^{-n\ell}. \end{aligned}$$

Corollary 3.24. *Let Dilithium with relevant parameters $n, q, \ell, \gamma_1, \gamma_2$ be as*

described in [BBD+23], and in addition $q \geq n\ell$. Then for every $a \in \mathbb{Z}_{>0}$ there is an event Γ_ϵ of the key sk such that $\Pr[\neg\Gamma_\epsilon] \leq e^{7/3}(n/q)^{a+1}$, and

$$\Pr[\Gamma_\epsilon] \cdot \mathbb{E}[\epsilon_{\mathsf{sk}} \mid \Gamma_\epsilon] \leq a \cdot e^{7/3} \cdot n^a \cdot \left(\frac{2\gamma_2+1}{2\gamma_1-1}\right)^{n\ell} .$$

Combining the above bound with Corollary 3.18, and simplifying the bound under a suitable choice of a, yields Corollary 3.25, and the corresponding concrete bounds in Table 3.16 (taking Γ_p as always satisfied, and a heuristic choice of $\bar{p}$, as in [KLS18]).

	$\bar{p}$	q_S	q_H	a	security loss
NIST2	$\leq \frac{49}{64}$	2^{64}	2^{128}	5	$\leq 2^{-79}$
			2^{64}	7	$\leq 2^{-103}$
			1	9	$\leq 2^{-127}$
NIST3	$\leq \frac{103}{128}$	2^{64}	2^{192}	25	$\leq 2^{-371}$
NIST5	$\leq \frac{759}{1024}$	2^{64}	2^{256}	37	$\leq 2^{-547}$

Figure 3.16: Concrete security loss of Dilithium from Corollary 3.25.

Corollary 3.25. *Let Dilithium with relevant parameters $n, q, \ell, \gamma_1, \gamma_2$ be as described in this section above, and in addition $q \geq n\ell$. Let $0 < \bar{p} < 1$ and Γ_p be an event on $(\mathsf{sk}, \mathsf{pk}) \leftarrow \mathsf{KGen}$ that implies $p_{\mathsf{sk}} \leq \bar{p}$. Then for every $\mathsf{UF\text{-}CMA}$ attacker $\mathcal{A}$ making at most Q_H quantum queries to H and q_S classical queries to the signing oracle, the $\mathsf{UF\text{-}NMA}$ attacker $\mathcal{B}$ (dependent on $\mathcal{A}$) as defined in Theorem 3.5 is such that for $q_H := Q_H + 1$ we have*

$$\mathbf{Adv}^{\mathsf{UF\text{-}CMA}}_{\mathsf{Dilithium}}(\mathcal{A}) \leq \mathbf{Adv}^{\mathsf{UF\text{-}NMA}}_{\mathsf{Dilithium}}(\mathcal{B}) + \frac{37 \cdot q_S}{1-\bar{p}} \sqrt{a \cdot n^a \cdot q_H \left(\frac{2\gamma_2+1}{2\gamma_1-1}\right)^{n\ell}} + \Pr[\neg\Gamma_p] ,$$

whenever

$$a := \left\lceil \frac{n\ell \cdot \log\left(\frac{2\gamma_1-1}{2\gamma_2+1}\right) - 2\log(q/n) - \log\left(q_S^2 q_H\right)}{2\log q - \log n} \right\rceil > 0 .$$